

---

# **TOPAS Documentation**

***Release 3.1***

**TOPAS MC Inc.**

**Jul 27, 2021**



|          |                                       |           |
|----------|---------------------------------------|-----------|
| <b>1</b> | <b>Introduction to TOPAS</b>          | <b>3</b>  |
| <b>2</b> | <b>Authors</b>                        | <b>5</b>  |
| 2.1      | TOPAS Software . . . . .              | 5         |
| 2.2      | TOPAS Documentation . . . . .         | 6         |
| <b>3</b> | <b>Citation</b>                       | <b>7</b>  |
| <b>4</b> | <b>User Support and Contributions</b> | <b>9</b>  |
| 4.1      | User Support . . . . .                | 9         |
| 4.2      | User Contributions . . . . .          | 9         |
| <b>5</b> | <b>History</b>                        | <b>11</b> |
| 5.1      | 3.5 Series . . . . .                  | 11        |
| 5.1.1    | 3.5.0 (2020-06-21) . . . . .          | 11        |
| 5.2      | 3.4 Series . . . . .                  | 12        |
| 5.2.1    | 3.4.0 (2020-6-21) . . . . .           | 12        |
| 5.3      | 3.3 Series . . . . .                  | 15        |
| 5.3.1    | 3.3.1 (2020-03-31) . . . . .          | 15        |
| 5.3.2    | 3.3.0 (2020-03-25) . . . . .          | 15        |
| 5.4      | 3.2 Series . . . . .                  | 17        |
| 5.4.1    | 3.2.2 (2019-08-15) . . . . .          | 17        |
| 5.4.2    | 3.2.1 (2019-07-07) . . . . .          | 17        |
| 5.4.3    | 3.2.0 (2019-05-02) . . . . .          | 17        |
| 5.5      | 3.1 Series . . . . .                  | 22        |
| 5.5.1    | 3.1.2 (2017-10-07) . . . . .          | 22        |
| 5.5.2    | 3.1.1 (2017-06-02) . . . . .          | 23        |
| 5.5.3    | 3.1.0 (2017-05-09) . . . . .          | 23        |
| 5.6      | 3.0 Series . . . . .                  | 28        |
| 5.6.1    | 3.0.1 (2016-06-02) . . . . .          | 28        |
| 5.6.2    | 3.0.0 (2016-06-01) . . . . .          | 28        |
| 5.7      | 2.0 Series . . . . .                  | 34        |
| 5.7.1    | 2.0.3 (2016-01-12) . . . . .          | 34        |
| 5.7.2    | 2.0.2 (2015-11-18) . . . . .          | 34        |
| 5.7.3    | 2.0.1 (2015-11-13) . . . . .          | 34        |
| 5.7.4    | 2.0.0 (2015-11-04) . . . . .          | 34        |
| 5.8      | 1.X Series . . . . .                  | 36        |

|           |   |           |
|-----------|---|-----------|
| 5.8.1     | 1.3.0 (2015-10-21)  | 36        |
| 5.8.2     | 1.2.2 (2015-06-12)  | 37        |
| 5.8.3     | 1.2.1 (2015-05-15)  | 37        |
| 5.8.4     | 1.2.0 (2015-04-20)  | 38        |
| 5.8.5     | 1.1.0 (2015-02-23)  | 39        |
| 5.8.6     | 1.0.0 (2015-02-11)  | 39        |
| <b>6</b>  | <b>Introduction to Parameter System</b>                   | <b>41</b> |
| 6.1       | Design Philosophy   | 41        |
| 6.2       | Syntax  | 42        |
| 6.2.1     | Complete Set of Allowed Syntax for any one Parameter Line | 44        |
| 6.3       | Relative Parameters                                       | 46        |
| 6.4       | Hierarchical Control                                      | 47        |
| 6.4.1     | Parameter File Chains                                     | 47        |
| 6.4.2     | Parameter File Graphs                                     | 48        |
| 6.5       | Controlling Multiple Batch Jobs                           | 49        |
| <b>7</b>  | <b>Default Parameters</b>                                 | <b>51</b> |
| 7.1       | Overall program control                                   | 51        |
| 7.2       | Overall timeline control                                  | 52        |
| 7.3       | Optional checks on correctness of geometry                | 52        |
| 7.4       | Top level geometry component, the World Volume            | 52        |
| 7.5       | Demo Beam position  | 53        |
| 7.6       | Demo Particle Source                                      | 53        |
| 7.7       | Physics   | 53        |
| 7.8       | Scoring   | 53        |
| 7.9       | Graphics  | 54        |
| 7.10      | Elements  | 54        |
| 7.11      | Materials   | 55        |
| 7.12      | Colors  | 57        |
| <b>8</b>  | <b>Overall Control</b>                                    | <b>59</b> |
| 8.1       | Time mode   | 59        |
| 8.1.1     | Fixed Time Mode   | 59        |
| 8.1.2     | Sequential Time Mode                                      | 59        |
| 8.1.3     | Random Time Mode  | 60        |
| 8.1.4     | Fixed Time but with Very Large Number of Histories        | 61        |
| 8.2       | Multithreading  | 61        |
| 8.3       | Random Number Seed  | 61        |
| 8.4       | Miscellaneous   | 62        |
| 8.4.1     | Interactive Geant4 Sessions                               | 62        |
| 8.4.2     | Dump Parameter Values                                     | 62        |
| 8.4.3     | Verbosity   | 62        |
| 8.4.4     | Other   | 63        |
| 8.4.5     | Quick Ways to Deactivate Parts of the Parameters Files    | 63        |
| <b>9</b>  | <b>Materials</b>  | <b>65</b> |
| 9.1       | Elements and Isotopes                                     | 66        |
| <b>10</b> | <b>Geometry Components</b>                                | <b>69</b> |
| 10.1      | Introduction  | 69        |
| 10.2      | Placement of Components                                   | 70        |
| 10.2.1    | Overlap Checking  | 71        |
| 10.2.2    | Pre-Defining Values                                       | 72        |
| 10.3      | Parallel Worlds   | 72        |

|           |   |            |
|-----------|---|------------|
| 10.3.1    | Layered Mass Geometry   | 73         |
| 10.4      | Electromagnetic Fields  | 73         |
| 10.5      | Visualization Attributes  | 75         |
| 10.6      | Dividable Components  | 76         |
| 10.7      | Generic Components  | 77         |
| 10.8      | Group Component   | 79         |
| 10.9      | Specialized Components  | 79         |
| 10.9.1    | Range Modulator Wheel   | 80         |
| 10.9.2    | Propeller   | 84         |
| 10.9.3    | Ridge Filter  | 86         |
| 10.9.4    | Multi Wire Chamber  | 88         |
| 10.9.5    | Multi Leaf Collimator   | 89         |
| 10.9.6    | CAD (Computer Aided Design)   | 90         |
| 10.9.7    | Aperture  | 92         |
| 10.9.8    | Compensator   | 94         |
| 10.9.9    | BrachyApplicator  | 96         |
| 10.10     | Patient Components  | 97         |
| 10.10.1   | Common Parameters   | 98         |
| 10.10.2   | Patient in DICOM Format   | 99         |
| 10.10.3   | Patient in ImageCube Format (handles XCAT, XiO, MaterialTagNumber and more) | 104        |
| 10.10.4   | Imaging to Material Conversion  | 105        |
| 10.10.5   | XCAT  | 105        |
| 10.10.6   | MaterialTagNumber   | 106        |
| 10.10.7   | Schneider   | 106        |
| <b>11</b> | <b>Particle Sources</b>   | <b>109</b> |
| 11.1      | Introduction  | 109        |
| 11.1.1    | Particle Names  | 110        |
| 11.2      | Beam Sources  | 110        |
| 11.3      | Emittance Sources   | 113        |
| 11.4      | Isotropic Sources   | 115        |
| 11.5      | volumetric Sources  | 116        |
| 11.6      | Phase Space Sources   | 116        |
| 11.7      | Miscellaneous   | 119        |
| 11.7.1    | Additional Control of Number of Histories                                   | 119        |
| 11.7.2    | Filtering Sources   | 119        |
| <b>12</b> | <b>Physics</b>  | <b>121</b> |
| 12.1      | Introduction  | 121        |
| 12.2      | Modular Physics Lists   | 122        |
| 12.2.1    | Physics Regions   | 122        |
| 12.2.2    | List of Available Modules   | 123        |
| 12.3      | Reference Physics Lists   | 124        |
| 12.4      | Optical Physics   | 125        |
| 12.4.1    | Optical Photons   | 125        |
| 12.4.2    | Optical Surfaces  | 126        |
| 12.5      | Miscellaneous   | 127        |
| 12.5.1    | User-Supplied Physics Lists   | 127        |
| 12.5.2    | Multiple Physics Lists  | 127        |
| 12.5.3    | Production Thresholds   | 128        |
| 12.5.4    | Step Size   | 128        |
| <b>13</b> | <b>Scoring</b>  | <b>129</b> |
| 13.1      | Introduction  | 129        |

|           |   |            |
|-----------|---|------------|
| 13.2      | Volume Scorers . . . . .                                | 129        |
| 13.2.1    | ProtonLET Scorer . . . . .                              | 131        |
| 13.3      | Surface Scorers . . . . .                               | 132        |
| 13.4      | Phase Space Scorer . . . . .                            | 133        |
| 13.4.1    | Phase Space Format . . . . .                            | 135        |
| 13.5      | Filtering Scorers . . . . .                             | 137        |
| 13.6      | Output Specification . . . . .                          | 141        |
| 13.6.1    | Common Parameters . . . . .                             | 141        |
| 13.6.2    | DICOM Output . . . . .                                  | 142        |
| 13.6.3    | Histogram Output . . . . .                              | 143        |
| 13.6.4    | DVH Output . . . . .                                    | 143        |
| 13.6.5    | Phasespace Output . . . . .                             | 144        |
| 13.7      | Miscellaneous . . . . .                                 | 144        |
| 13.7.1    | Binning in Dividable Components . . . . .               | 144        |
| 13.7.2    | Binning by Energy . . . . .                             | 145        |
| 13.7.3    | Binning by Time . . . . .                               | 145        |
| 13.7.4    | Splitting by Time Feature . . . . .                     | 146        |
| 13.7.5    | Statistical Information . . . . .                       | 147        |
| 13.7.6    | Change Component Color Based on Scoring . . . . .       | 147        |
| 13.7.7    | Toggling a Scorer Off and On . . . . .                  | 148        |
| 13.7.8    | Restoring Results from Files . . . . .                  | 148        |
| <b>14</b> | <b>Graphics</b>   | <b>149</b> |
| <b>15</b> | <b>Time Features</b>                                    | <b>155</b> |
| 15.1      | Linear, Sine, Cosine and Sqrt Functions . . . . .       | 156        |
| 15.2      | Step Function . . . . .                                 | 156        |
| 15.3      | Combining Time Features for Complex Behaviors . . . . . | 157        |
| <b>16</b> | <b>Variance Reduction</b>                               | <b>159</b> |
| 16.1      | Introduction . . . . .                                  | 159        |
| 16.1.1    | Specify the Split Geometry . . . . .                    | 159        |
| 16.1.2    | Define the Splitting Technique . . . . .                | 160        |
| 16.2      | Geometrical Particle Splitting . . . . .                | 160        |
| 16.3      | Importance Sampling . . . . .                           | 161        |
| 16.4      | Weight Window . . . . .                                 | 162        |
| 16.5      | Tracking Only Specific Particles . . . . .              | 163        |
| 16.6      | Secondary Biasing . . . . .                             | 163        |
| 16.7      | Forced interaction . . . . .                            | 164        |
| 16.8      | Cross section enhancement . . . . .                     | 165        |
| <b>17</b> | <b>Outcome Modeling</b>                                 | <b>167</b> |
| <b>18</b> | <b>The TOPAS Graphical User Interface</b>               | <b>169</b> |
| 18.1      | Mouse Control . . . . .                                 | 170        |
| 18.2      | The Parameter Control Table . . . . .                   | 170        |
| 18.3      | Additional Functions . . . . .                          | 171        |
| 18.4      | Scene Tree . . . . .                                    | 171        |
| <b>19</b> | <b>Basic</b>  | <b>173</b> |
| 19.1      | AllParameterForms.txt . . . . .                         | 173        |
| 19.2      | DividedComponents.txt . . . . .                         | 176        |
| 19.3      | DNAModelByRegions.txt . . . . .                         | 178        |
| 19.4      | Emittance_Gaussian.txt . . . . .                        | 178        |
| 19.5      | Emittance_Twiss.txt . . . . .                           | 180        |

|           |  |            |
|-----------|--|------------|
| 19.6      | EmModelByRegions.txt                     | 180        |
| 19.7      | FlatteningFilter.txt                     | 181        |
| 19.8      | Isotope.txt                              | 182        |
| 19.9      | LayeredMassGeometry.txt                  | 183        |
| 19.10     | OneBox.txt                               | 185        |
| 19.11     | OneBoxRotate.txt                         | 185        |
| 19.12     | OneBoxTranslate.txt                      | 185        |
| 19.13     | PhysicsSetting.txt                       | 186        |
| 19.14     | ShapeTestWithAllParameters.txt           | 187        |
| 19.15     | ShapeTestWithOnlyRequiredParameters.txt  | 196        |
| 19.16     | Spectrum.txt                             | 203        |
| 19.17     | TwoBeams.txt                             | 203        |
| 19.18     | VolumetricSource.txt                     | 204        |
| <b>20</b> | <b>Graphics</b>                          | <b>207</b> |
| 20.1      | ColorByCreatorProcess.txt                | 207        |
| 20.2      | ColorByEnergy.txt                        | 209        |
| 20.3      | ColorByGeneration.txt                    | 210        |
| 20.4      | ColorByMomentum.txt                      | 212        |
| 20.5      | ColorByOriginComponent.txt               | 214        |
| 20.6      | ColorByOriginComponentOrSubComponent.txt | 215        |
| 20.7      | ColorByOriginVolume.txt                  | 217        |
| 20.8      | ColorByParticleType.txt                  | 219        |
| 20.9      | ColorTest.txt                            | 220        |
| 20.10     | FilterByCreatorProcess.txt               | 223        |
| 20.11     | FilterByEnergy.txt                       | 225        |
| 20.12     | FilterByMomentum.txt                     | 226        |
| 20.13     | FilterByOriginComponent.txt              | 228        |
| 20.14     | FilterByParticleCharge.txt               | 230        |
| 20.15     | NanoMeterScaleTest.txt                   | 231        |
| 20.16     | QtTest.txt                               | 232        |
| 20.17     | SolidSphere.txt                          | 233        |
| 20.18     | TestGraphics.txt                         | 234        |
| 20.19     | TwoProjections.txt                       | 236        |
| <b>21</b> | <b>Nozzle</b>                            | <b>237</b> |
| 21.1      | ConstantsForNozzles.txt                  | 237        |
| 21.2      | RasterScanningPattern.txt                | 239        |
| 21.3      | ScanningNozzle.txt                       | 239        |
| 21.4      | ScanningStationaryTarget.txt             | 250        |
| 21.5      | ScanningTargetMovingHorizontal.txt       | 251        |
| 21.6      | ScanningTargetMovingInDepth.txt          | 251        |
| 21.7      | ScatteringNozzle.txt                     | 252        |
| 21.8      | ScanningNozzle_QtTest.txt                | 263        |
| 21.9      | ScatteringNozzle_run.txt                 | 263        |
| 21.10     | ScanningNozzle_run_QtTest.txt            | 264        |
| 21.11     | TargetInWaterPhantom.txt                 | 264        |
| 21.12     | TsAnalysisWaterPhantom.txt               | 264        |
| 21.13     | Viewer.txt                               | 265        |
| <b>22</b> | <b>Optical</b>                           | <b>267</b> |
| 22.1      | OpticalMaterialSample.txt                | 267        |
| 22.2      | OpticalPhotonCount.txt                   | 271        |
| 22.3      | PlasticScintillator.txt                  | 273        |

|           |                                 |            |
|-----------|---------------------------------|------------|
| 22.4      | Rotating_Surfaces.txt           | 274        |
| 22.5      | Surfaces.txt                    | 275        |
| 22.6      | WavelengthShifter.txt           | 277        |
| <b>23</b> | <b>Outcome</b>                  | <b>281</b> |
| 23.1      | CEModelBurman.txt               | 281        |
| 23.2      | CVModelStavrev.txt              | 281        |
| 23.3      | Phantom.txt                     | 282        |
| 23.4      | PoissonModelOkuneff.txt         | 283        |
| 23.5      | TestOutcomeModel.txt            | 285        |
| 23.6      | TestRestoreModel.txt            | 287        |
| <b>24</b> | <b>Patient</b>                  | <b>291</b> |
| 24.1      | Applicator.txt                  | 291        |
| 24.2      | DoseTo4DCT.txt                  | 296        |
| 24.3      | DoseToCT.txt                    | 298        |
| 24.4      | HUtoMaterialSchneiber.txt       | 300        |
| 24.5      | HUtoMaterialSchneider2.txt      | 302        |
| 24.6      | Implant.txt                     | 304        |
| 24.7      | TwoDicoms.txt                   | 306        |
| 24.8      | ViewAbdomen_rtdose.txt          | 308        |
| 24.9      | ViewAbdomen.txt                 | 311        |
| 24.10     | XCAT_Materials.txt              | 313        |
| 24.11     | XCAT.txt                        | 315        |
| <b>25</b> | <b>PhaseSpace</b>               | <b>317</b> |
| 25.1      | MultiRun_Write_ASCII.txt        | 317        |
| 25.2      | ReadASCII.txt                   | 318        |
| 25.3      | ReadBinary.txt                  | 319        |
| 25.4      | ReadLimited.txt                 | 320        |
| 25.5      | WriteASCII.txt                  | 321        |
| 25.6      | WriteBinary.txt                 | 322        |
| 25.7      | WriteIonsASCII.txt              | 323        |
| 25.8      | WriteLimited.txt                | 324        |
| 25.9      | WriteROOT.txt                   | 326        |
| <b>26</b> | <b>Scoring</b>                  | <b>329</b> |
| 26.1      | ChargeInSphere.txt              | 329        |
| 26.2      | Complex.txt                     | 330        |
| 26.3      | Dose.txt                        | 333        |
| 26.4      | DoseInVoxelMaterials.txt        | 334        |
| 26.5      | DoseToMediumVsWater.txt         | 335        |
| 26.6      | DoseVolumeHistogram.txt         | 337        |
| 26.7      | EnergyAndDose.txt               | 338        |
| 26.8      | EnergyDepositBinnedByEnergy.txt | 339        |
| 26.9      | EnergyDivisions.txt             | 341        |
| 26.10     | EnergyInBinnedCylinder.txt      | 342        |
| 26.11     | Filters.txt                     | 344        |
| 26.12     | FoilToBox.txt                   | 357        |
| 26.13     | FoilToCylinder.txt              | 358        |
| 26.14     | FoilToSphere.txt                | 360        |
| 26.15     | Gated.txt                       | 361        |
| 26.16     | GeometryDivisions.txt           | 363        |
| 26.17     | Histograms.txt                  | 364        |
| 26.18     | Inactive.txt                    | 366        |



|           |   |            |
|-----------|---|------------|
| 26.19     | Ion.txt . . . . .                                 | 367        |
| 26.20     | Origin.txt . . . . .                              | 368        |
| 26.21     | ParallelBoxRebinned.txt . . . . .                 | 370        |
| 26.22     | SplitByTimeFeature.txt . . . . .                  | 371        |
| 26.23     | Surfaces.txt . . . . .                            | 372        |
| 26.24     | VoxelMaterials.txt . . . . .                      | 374        |
| <b>27</b> | <b>SpecialComponents</b>                          | <b>379</b> |
| 27.1      | CAD.txt . . . . .                                 | 379        |
| 27.2      | DipoleMagnet.txt . . . . .                        | 380        |
| 27.3      | MultiLeafCollimator.txt . . . . .                 | 381        |
| 27.4      | MultiLeafCollimator_sequence.txt . . . . .        | 383        |
| 27.5      | MultiWire_Chamber.txt . . . . .                   | 384        |
| 27.6      | Propeller_ContinuousRotation.txt . . . . .        | 385        |
| 27.7      | Propeller_StepRotation.txt . . . . .              | 385        |
| 27.8      | Propeller.txt . . . . .                           | 386        |
| 27.9      | PurgingMagnet_move.txt . . . . .                  | 387        |
| 27.10     | QuadAndDipoleMagnets.txt . . . . .                | 390        |
| 27.11     | QuadInMovingNozzle.txt . . . . .                  | 393        |
| 27.12     | QuadrupoleMagnet.txt . . . . .                    | 395        |
| 27.13     | RangeModulator_ConstantBeam.txt . . . . .         | 396        |
| 27.14     | RangeModulator_CurrentModulatedBeam.txt . . . . . | 396        |
| 27.15     | RangeModulator.txt . . . . .                      | 397        |
| 27.16     | RidgeFilter.txt . . . . .                         | 399        |
| 27.17     | RotatingMagnet.txt . . . . .                      | 400        |
| 27.18     | UniformElectroMagneticField.txt . . . . .         | 403        |
| <b>28</b> | <b>TimeFeature</b>                                | <b>405</b> |
| 28.1      | BoxWithinBox.txt . . . . .                        | 405        |
| 28.2      | CameraRotateAndZoom.txt . . . . .                 | 407        |
| 28.3      | ChangingKEFilterByTimeFeature.txt . . . . .       | 408        |
| 28.4      | ColorChange.txt . . . . .                         | 409        |
| 28.5      | CylinderGrowingInPhi.txt . . . . .                | 411        |
| 28.6      | Darkening.txt . . . . .                           | 413        |
| 28.7      | Logo.txt . . . . .                                | 419        |
| 28.8      | RotateResizeRecolor.txt . . . . .                 | 425        |
| 28.9      | Rotation.txt . . . . .                            | 427        |
| 28.10     | RunRandom_Mode.txt . . . . .                      | 430        |
| 28.11     | RunSequential_Mode.txt . . . . .                  | 430        |
| <b>29</b> | <b>UCSFETF</b>                                    | <b>433</b> |
| 29.1      | UCSF_Beamline_Base.txt . . . . .                  | 434        |
| 29.2      | UCSF_Beamline_Propeller10.txt . . . . .           | 434        |
| 29.3      | UCSF_Beamline_Propeller15.txt . . . . .           | 434        |
| 29.4      | UCSF_Beamline_Propeller20.txt . . . . .           | 434        |
| 29.5      | UCSF_Beamline_Propeller24.txt . . . . .           | 434        |
| 29.6      | UCSF_DefaultParameters.txt . . . . .              | 434        |
| 29.7      | UCSF_WaterPhantom.txt . . . . .                   | 434        |
| 29.8      | User_Beamline_WC_Viewer.txt . . . . .             | 434        |
| 29.9      | User_BP_R28_WaterPhantom.txt . . . . .            | 434        |
| 29.10     | User_SOBP24_R28_WaterPhantom.txt . . . . .        | 434        |
| 29.11     | User_SOBP24_Viewer.txt . . . . .                  | 434        |
| 29.12     | Viewer.txt . . . . .                              | 434        |
| <b>30</b> | <b>VarianceReduction</b>                          | <b>435</b> |

|           |  |            |
|-----------|--|------------|
| 30.1      | AmbientDoseEquivalent.txt . . . . .                                    | 435        |
| 30.2      | CrossSectionEnhancement.txt . . . . .                                  | 440        |
| 30.3      | ForcedInteraction.txt . . . . .  | 442        |
| 30.4      | GeometricalParticleSplit.txt . . . . .                                 | 444        |
| 30.5      | ImportanceSampling.txt . . . . .                                       | 447        |
| 30.6      | ImportanceSamplingMassGeometry.txt . . . . .                           | 451        |
| 30.7      | ImportanceSamplingSecondaryBiasing.txt . . . . .                       | 456        |
| 30.8      | SecondaryBiasing.txt . . . . .   | 461        |
| 30.9      | WeightWindow.txt . . . . .   | 463        |
| <b>31</b> | <b>Introduction to Extensions</b>                                      | <b>469</b> |
| 31.1      | Extra Classes . . . . .  | 471        |
| 31.2      | Changeable Parameters . . . . .  | 471        |
| 31.3      | Transient Parameters . . . . .   | 472        |
| <b>32</b> | <b>Custom Geometry Components</b>                                      | <b>473</b> |
| <b>33</b> | <b>Custom Particle Sources</b>   | <b>477</b> |
| <b>34</b> | <b>Custom Physics Lists and Physics Modules</b>                        | <b>479</b> |
| <b>35</b> | <b>Custom Scorers</b>  | <b>481</b> |
| <b>36</b> | <b>Custom Outcome Models</b>   | <b>485</b> |
| <b>37</b> | <b>Custom Filters</b>  | <b>487</b> |
| <b>38</b> | <b>Custom Fields</b>   | <b>489</b> |
| <b>39</b> | <b>Custom Imaging to Material Conversion</b>                           | <b>491</b> |
| <b>40</b> | <b>Additional User Hooks</b>   | <b>493</b> |
| <b>41</b> | <b>Proton RBE Scorers</b>  | <b>495</b> |
| 41.1      | Important notes . . . . .  | 496        |
| 41.2      | Normalizing simulations to prescriptions . . . . .                     | 496        |
| 41.3      | Using tissue-specific RBE model parameters . . . . .                   | 497        |
| 41.4      | Reusing sub-scorers to reduce simulation memory requirements . . . . . | 497        |
| 41.5      | Content . . . . .  | 497        |
| 41.6      | Scoring . . . . .  | 498        |
| 41.7      | References . . . . .   | 499        |
|           | <b>Bibliography</b>  | <b>501</b> |

---

**Note:** A PDF version of the documentation is found by clicking the “Read the Docs” panel in the bottom-left corner of the website.

---



---

## Introduction to TOPAS

---

TOPAS wraps and extends the [Geant4 Simulation Toolkit](#) to provide an easier-to-use application for the medical physicist. TOPAS's unique parameter control system lets you assemble and control a rich library of simulation objects (geometry components, particle sources, scorers, etc.) with no need to write C++ code.

Advanced users remain free to implement their own simulation objects in C++ code, and add them to TOPAS via an *extension mechanism*. While user-written objects benefit from underlying functionality of TOPAS base classes and the TOPAS parameter system, they can exploit the full flexibility of Geant4.

To discover the Geant4 version used by a specific version of TOPAS, please consult the [History](#).

---

**Note:** Users should carefully read the *Introduction to Parameter System*, as well as the introductory parts of the sections *Geometry Components*, *Particle Sources*, *Physics* and *Scoring*. The rest of the documentation provides a detailed reference that you may just want to skim initially.

---

Installation instructions for the latest TOPAS version are provided at the top of the [Code Repository](#) page of the TOPAS web site. This page is accessible only to licensed TOPAS users.



### 2.1 TOPAS Software

- Joseph Perl
- Jungwook Shin
- Jan Schuemann
- Bruce Faddegon
- Harald Paganetti
- Jose Ramos
- Aimee McNamara
- David Hall

## 2.2 TOPAS Documentation

|                    |   |
|--------------------|---|
| Website Creator    | David Hall  |
| Lead Author        | Joseph Perl   |
| Additional Authors | <ul style="list-style-type: none"><li>• Jungwook Shin</li><li>• Jan Schuemann</li><li>• Jose Ramos</li><li>• David Hall</li></ul> |
| Editors            | <ul style="list-style-type: none"><li>• Joseph Perl</li><li>• David Hall</li></ul>  |
| User Contributions | <ul style="list-style-type: none"><li>• Aleksandra Biegun</li><li>• Fada Guan</li></ul>   |



If you use TOPAS, please be sure to cite the following paper ([PubMed](#)):

Perl J, Shin J, Schumann J, Faddegon B, Paganetti H. TOPAS: an innovative proton Monte Carlo platform for research and clinical applications. Med Phys. 2012; 39(11):6818-37.

and the follow-up paper emphasizing the latest developments:

Faddegon B, Ramos-Mendez J, Schuemann J, McNamara A, Shin J, Perl J, Paganetti H, The TOPAS Tool for Particle Simulation, a Monte Carlo Simulation Tool for Physics, Biology and Clinical Research, Physica Medica, doi:10.1016/j.ejmp.2020.03.019. Here is a BibTeX entry to aid use within LaTeX:

To discover the Geant4 version used by a specific version of TOPAS, please consult the [History](#).

```
@article{TOPAS,
  author = {Perl, J. and Shin, J. and Schumann, J. and Faddegon, B. and Paganetti,
↪H.},
  title = {{TOPAS: An innovative proton Monte Carlo platform for research and
↪clinical applications}},
  journal = {Medical Physics},
  pages = {6818},
  volume = {39},
  year = {2012},
  pmid = {23127075},
  doi = {10.1118/1.4758060}
}

@article{TOPAS,
  author = {Faddegon, B. and Ramos-Mendez, J. and Schuemann, J. and McNamara, A.
↪and Shin, J. and Perl, J. and Paganetti H.},
  title = {{The TOPAS Tool for Particle Simulation, a Monte Carlo Simulation Tool
↪for Physics, Biology and Clinical Research}},
  journal = {Physica Medica},
  year = {2020},
  doi = {10.1016}
}
```



---

# User Support and Contributions

---

## 4.1 User Support

Help can be requested from TOPAS developers and other users at the [User Forum](#). Note that you must login as a registered user before you can access the forums.

## 4.2 User Contributions

- Feedback and feature requests are best communicated via the User Forum described above.
- If you would like to share your TOPAS [extensions](#) or parameter files with other users, please get in touch
- Contributions to this documentation website are also most welcome, but does require a GitHub account:
  - Click “Edit on GitHub” in the top-right corner of the page you wish to edit
  - This takes you to the page source at GitHub.com
  - Click the pencil icon to [fork the repository](#) and edit the page source
  - The source format is [reStructuredText](#)
  - After making your changes, open a [pull request](#)



TOPAS wraps and extends the [Geant4 Simulation Toolkit](#). Geant4 is built directly into TOPAS (the user does not need to themselves install Geant4). The following table lists which Geant4 version is built into each TOPAS version.

| TOPAS version | Geant4 version |
|---------------|----------------|
| 3.5.0         | 10.06.p01      |
| 3.4.0         | 10.05.p02      |
| 3.3.0         | 10.05.p01      |
| 3.2.0         | 10.05.p01      |
| 3.1.0         | 10.03.p01      |
| 3.0.0         | 10.02.p01      |
| 2.0.0         | 10.01.p02      |
| 1.0.0         | 9.06.4         |

A detailed list of changes introduced by each TOPAS version can be found below:

## 5.1 3.5 Series

This release is almost exactly the same TOPAS code as TOPAS Version 3.4, but with a new underlying Geant4 version. TOPAS 3.5 uses Geant4.10.06.p1, whereas TOPAS 3.4 used Geant4.10.05.p2. Our tests show no major difference in patient dose calculations, however, as this is a new Geant4 version, you should be on the lookout for changes that may affect your simulations. Our intention is that by comparing your TOPAS 3.5 simulation against a TOPAS 3.4 simulation, you can see how your results are affected by the underlying Geant4 version. There is one new feature:

### 5.1.1 3.5.0 (2020-06-21)

**Added: Users of Topas 3.5 can take advantage of a new Drawing Style** This new drawing style represents a solid as a random cloud of points within the solid:

```
S:Ge/MyComponent/DrawingStyle = "Cloud" # Previous options were "WireFrame",  
  ~"FullWireFrame" and "Solid"
```

It can be particularly useful in visualization of Boolean solids, as such solids are not always handled correctly by the polygon-generator underlying the other drawing styles. By default, the point cloud will be made of ten thousand points. You can adjust this by setting the new optional parameter:

```
i:Ge/MyComponent/NumberOfCloudPoints
```

## 5.2 3.4 Series

The main features introduced by the 3.4 series were:

- significant reduction of the memory footprint of TOPAS
- redefined variable density materials
- graphics options

### 5.2.1 3.4.0 (2020-6-21)

**Improved: Significant reduction to the memory footprint** The memory savings are substantial, in particular for the most challenging full patient simulations, when there are many different materials, many voxels and potentially many worker threads.

**Fixed: Significant memory leak in defining patient materials** We identified some C++ objects involved in materials creation that did not need to be retained. For a typical patient simulation involving thousands of materials, the savings from this fix are on the order of a full GB.

**Improved: Scorers now require much less memory** We found that our scoring code had been allocating all the maps needed for all of the different reporting options (Sum, Mean, Minimum, Maximum, etc.) even if the user only wanted to report a single one of these quantities. The result is an up to fivefold reduction in memory use by scorers. And as these memory costs are repeated once per worker thread, they can be even more substantial if your simulation has a high `Ts/NumberOfThreads`. Savings may be on the order of several GB.

**Improved: Memory-intensive Mott Scattering correction tables are now optional** We found, for our default physics settings, most of the memory devoted to physics lookup tables was to support one specific process detail, the Mott Scattering correction. This correction is on by default in our default physics settings, as those settings include the EM module `g4em-standard_opt4`. Users may save significant memory by either changing their physics settings to use a different EM option, or by staying with `Opt4` but turning off this correction factor:

```
b:Ph/Default/MottCorrection = "False" # Defaults to "True"
```

**Added: Options for Variable Density Materials** We hoped to save additional memory by changing how we define materials. Our most commonly used imaging to material converter, the Schneider method, defines thousands of different materials (to correspond to the thousands of different Hounsfield numbers in the CT). Within this large set of materials, there are many cases where hundreds of materials share the same elemental composition, with only a variation of density. Up to now, we have used the original Geant4 method of material definition, defining each one of these thousands of materials as an entirely separate `G4Material`. However, Geant4 eventually added an option to just define a single material of a given composition, and then define many other materials as different density variants of this “Base Material”. In the hopes that this new method would give us comparable physics results while using less memory (as it would have to create fewer full material tables), we added a new option:

```
b:Ge/Patient/SchneiderUseVariableDensityMaterials = "True"
```

While this did result in memory savings, it also gave some alarming differences in a few of our detailed patient dose tests. For some test cases, there was a good match, but for other cases we saw significant dose differences in some areas with a high density gradient (such as the interface of patient head to the surrounding air). Our hypothesis is that the variable density material technique is not fully implemented for all Geant4 physics processes. We will continue to study this issue, but didn't want to hold off on this release any longer. Users may want to try this option at their discretion, and depending on their choices of physics options, but we are leaving it off by default.

**Added: Other features for Variable Density Materials** You can now define a TOPAS material as a variant of some other TOPAS material:

```
s:Ma/MyMaterial/BaseMaterial = "SomeOtherMaterialName"
d:Ma/MyMaterial/Density = 1.5 g/cm3
```

And if you are a C++ expert writing your own `ImagingToMaterial` converter, you can create a material based on a Base Material by calling the new method:

```
TsVImagingToMaterial: G4int DefineNewMaterial (G4String materialName, G4double_
↳density, G4String baseMaterialName, G4String colorName)
```

**Improved: ImagingToMaterial converter options now include ByIntegerID** Some of our users have `TsImageCube` components where each voxel is represented not as a CT number but as an integer “tag number,” a 16-bit integer (C++ short) that corresponds to a particular material name. A new `ImagingToMaterialConverter` will interpret these tag numbers based on a lookup table created by two additional TOPAS vector parameters, `MaterialTagNumbers` and `MaterialNames`. For example:

```
s:Ge/Patient/Type = "TsImageCube"
s:Ge/Patient/ImagingToMaterialConverter = "ByIntegerID"
iv:Ge/Patient/MaterialTagNumbers = 6 0 3 42 43 100 110
sv:Ge/Patient/MaterialNames = 6 "Air" "G4_BLOOD_ICRP" "G4_BONE_CORTICAL_ICRP" "G4_
↳BONE_COMPACT_ICRU" "G4_BRAIN_ICRP" "G4_MUSCLE_SKELETAL_ICRP"
```

- Where the voxel is tagged with the number 0, the converter will interpret this as “Air”
- Where the voxel is tagged with the number 3, the converter will interpret this as ” G4\_BLOOD\_ICRP “
- Where the voxel is tagged with the number 42, the converter will interpret this as ” G4\_BONE\_CORTICAL\_ICRP “

etc.

**Improved: Ancestor Track and Vertex information has been extended in TsTrackInformation** This feature allows extension fillers and scorers to access more information about a track's history. For each of the track's ancestors, it now provides the track pointer (rather than just the track ID), the vertex momentum direction and the vertex kinetic energy (rather than just the vertex ID). Specifically, the new method signatures in `TsTrackInformation.hh` are:

```
std::vector<const G4Track*> TsTrackInformation::GetParentTracks()
std::vector<G4ThreeVector>_
↳TsTrackInformation::GetParentTrackVertexMomentumDirections()
std::vector<G4double> TsTrackInformation::GetParentTrackVertexKineticEnergies()
```

**Improved: Ions of any charge can now work as Particle Sources** TOPAS previously only supported generation of fully stripped ions. All ion charges are now supported. Specify the charge by giving the appropriate Charge value in the particle name `GenericIon(Z, A, Charge)`.

**Improved: 4D CT Materials loading logic** The materials definition process has always been a little complicated for 4D CT cases, since Geant4 requires that we define all materials before the first particle history is created. If 4D behavior causes a new CT image to be loaded later in the simulation, and this image contains a material that was not present in the initial CT image, this could cause TOPAS to have to quit. Our previous method of handling this was to have you add the parameter:

```
b:Ge/Patient/PreLoadAllMaterials = "True"
```

This worked by defining every material in your HU conversion table, whether that material is ever actually used or not. This had the unwanted side effect of wasting initialization time and memory on some materials that were never used. We suggest you no longer use the `PreLoadAllMaterials` feature, and instead use the feature:

```
sv:Ge/Patient/PreLoadMaterialsFrom = 3 "CT_Image_1" "CT_Image_2" "CT_Image_3"
```

where you specify the names of all the different CT images that will eventually be loaded. TOPAS will then define all the materials in all of these images, but not waste time and memory on materials that are not used in any of the images.

**Added: Multiple DICOM components can now be included in the same simulation** While this is an unusual situation, we had a user who wanted to represent two separate DICOM components in a single simulation. We found a small change we could make so that this is now supported, and we added a new example to demonstrate this capability `examples/Patient/TwoDicoms.txt`. The new example puts both the Abdomen DICOM and the DICOM\_Box into two different locations in the same simulation. Note: each DICOM component in this example needs its own separate copy of the HU conversion parameters, so you will see that the example has `includeFile` statements for both `HUtoMaterialSchneider.txt` and `HUtoMaterialSchneider2.txt`. A more elegant solution would allow one to use the same conversion parameters in both DICOMs, but this would require syntax changes that would break previous use cases, so such a change will have to wait for a future new major release.

**Improved: Particle Killing Variance Reduction options** This feature did not previously accept particle names that were of the form `GenericIon`. It now works for all particle types. We also added an inverted version of this option. So, the available options are now:

```
sv:Vr/KillOtherParticles/OnlyTrackParticlesNamed
sv:Vr/KillOtherParticles/OnlyTrackParticlesNotNamed
```

**Added: Graphics Options** Graphics Options have been added to control Background Color, Line Width and Step Point Size:

```
s:Gr/MyOpenGL/BackgroundColor = "white" # Set background color
i:Gr/MyOpenGL/Linewidth = 8 # Set line width for both geometry and trajectories
i:Gr/MyOpenGL/StepPointSize = 16 # Set size of step points
```

**Added: New Graphics Examples** New examples have been added to demonstrate various settings of the trajectory `ColorBy` options. Look in `examples/Graphics` to see the many new examples.

**Fixed: Bug previously reported as “Incorrect results from dose scorer in certain unusual setups”** This bug affected dose scoring in a component that has a subcomponent, such as scoring on a box that has another box inside it. The issue is now fixed. Any users who have written their own extension scorers that rely on the volume of the scoring component are advised to copy our fix by replacing any calls to `fSolid->GetCubicVolume()` with the new method now accessible directly from the scorer’s base class `GetCubicVolume(aStep)`.

**Fixed: Bug that was preventing one of the physics parameters from working** This should now work correctly:

```
d:Ph/Default/SetProductionCutHighEdge
```

**Fixed: Bug that was making Step Points drawing not work in some cases.** This was only working for some of the Trajectory `ColorBy` options. It now works for all cases:



```
b:Gr/MyView/IncludeStepPoints
```

## 5.3 3.3 Series

The main features introduced in the 3.3 series were:

- Graphical User Interface (GUI)
- option to declare neutrons stable
- surface name `AnySurface`

### 5.3.1 3.3.1 (2020-03-31)

**Fixed: Jobs crashing** Jobs were crashing if they had ALL THREE of the following characteristics in the same job:

- 1) Geometry changing over time in a way that was more than just changing position
- 2) Multiple threads
- 3) Scoring

This has been fixed. If your job was not crashing, this issue did not affect your job and there is no need to re-do those simulations.

### 5.3.2 3.3.0 (2020-03-25)

**Added: Graphical User Interface (GUI)** The key features of the GUI are as follows:

- You can rotate and zoom the graphics window, and you can click on geometry and trajectories to get details about them
- You can adjust parameters and see the effects immediately in the graphics window.
- You can see and correct geometry issues such as overlaps for rapid prototyping (you will find that the “Run” button will not work until all overlaps are resolved).
- You can add geometry components, scorers, particle sources, etc., from the GUI
- Once you have a set of parameters that you like, click the “Save” button to save the changes to a new parameter file (for subsequent runs either with or without the GUI).
- To activate the GUI, your parameter file needs: `Ts/UseQt = "True"`
- To make a parameter appear in the GUI’s parameter control window, just add “c” (for changeable) right after the parameter type, for example, where your parameter file previously had `d:Ge/MyBox/HLX` change it to `dc:Ge/MyBox/HLX`
- A simple Qt example has been added `topas/examples/Graphics/QtTest.txt`
- MacOS 10.15 Catalina users will find that the GUI turns on automatically if they request OpenGL graphics even if they have not set `Ts/UseQt`. This is because the Qt GUI is the only way that we can activate OpenGL on that operating system.
- The GUI is supported on all platforms except for CentOS6 and RHEL 6.

**Improved: Color code trajectories** Options to color code trajectories have been extensively revised.

**Improved: Coloring and filtering options for particle trajectory visualization** In response to an error reported by a user, we carried out a complete review of all the coloring and filtering options for particle trajectory visualization. We found and fixed many issues (this part of the TOPAS code has not been reviewed in a long time). We believe that all of the `Gr/*/ColorBy...` parameters and `Gr/*/OnlyInclude...` parameters are now working correctly. This includes making sure they understand all of the defined colors (previously only worked for a subset of the Topas colors that happened to also have the same name in the underlying Geant4 color defaults), and making sure they work for `GenericIon` as well as other particle types. We also added many new `ColorBy` and `FilterBy` examples in `examples/Graphics`.

**Improved: Scoring with `OutputAfterRun`** When scoring with `OutputAfterRun`, there is now a way to avoid clearing the scorers between runs. Thus each run's output will contain the new time interval's scores plus all scores accumulated during previous runs. To activate this feature, set the parameter:

```
b/Sc/MyScorer/OutputAfterRunShouldAccumulate = "True"
```

**Fixed: Scoring option `OutputAfterRun`** A user found that scorers gave incorrect output if they set `OutputAfterRun` and the scorer had been set to report more than just the sum. Scoring option `OutputAfterRun` now works correctly in all cases.

**Fixed: Scoring in Geometry Components** Scoring now works correctly in geometry components that are changing size under time features. A user noted that if the scoring component changed size, it would no longer receive any hits. This has now been fixed.

**Fixed: Variance Reduction** Variance Reduction now works correctly in components that are changing size under time features.

**Fixed: Magnetic and Electric fields** Magnetic and Electric fields now work correctly in components that are changing size under time features.

**Improved: `Vr/KillOtherParticles/OnlyTrackParticlesNamed` now supports `GenericIons`** It previously only worked for other particle names.

**Added: Declare neutrons stable** You can now declare that neutrons should be considered stable. This can be used to work around error conditions such as "Can not determine decay channel ...". Set parameter:

```
b:Ph/SetNeutronToStable = "True"
```

**Added: New surface name `AnySurface`** A new surface name `AnySurface` lets you make a surface scorer be sensitive on any surface. You could use this, for example, to capture phase space of particles passing through all surfaces of a cylinder (instead of the current `ZPlusSurface`, `OuterCurvedSurface`, etc.)

**Improved: Surface names are no longer case sensitive** So where we previously required exactly `ZPlusSurface`, we now accept `zplussurface`, etc.

**Improved: Mapped Magnetic Field handling** A user provided some corrections to the mapped field handling.

**Fixed: DICOM RT Structure filter** The DICOM RT Structure filter now survives hits with invalid indices. A rare Geant4 bug used to cause this filter to crash. The filter now survives this condition, and the relevant hits are appropriately trapped and reported.

**Improved: Protection against use of cylinders or spheres parameterized in `r`, `phi` or `theta` in the presence of parallel worlds** We have found that Geant4's parallel world navigation system has problems in these conditions and can cause hits to be associated with wrong bins (often piling up in bin zero). We had some protection against this previously, but have learned that more cases are affected, so we now completely forbid use of these parameterizations when there are parallel worlds anywhere in the setup. You can override this limitation if you wish by setting the following parameter, but if you do so, we suggest you check that your results seem sensible:

```
b:Ge/AllowDividedCylinderOrSphereWithParallelWorld = "True"
```

**Fixed: Scoring option `PropagateToChildren`** This option was not working correctly in cases where other scorers were also attached to those child components.

**Improved: Forbid attaching a scorer to the World component** We have seen some examples where users set the World as the scoring component. This was something we had not anticipated in the TOPAS design, and it gives incorrect results in some cases (such as if you add binning options in the scorers). It is now simply forbidden.

**Improved: Uniformity check on DICOM `DoseGridOffset`** The uniformity check on DICOM `DoseGridOffset` now allows for up to 2% difference.

## 5.4 3.2 Series

The main features introduced by the 3.2 series were:

- Option to Score Recursively in Subcomponents
- Volumetric Particle Source
- Auger Cascade
- New Physics Constructors and lists
- Material can now be 4D
- Centering Graphics on a Given Component
- Scaling Graphics by Different Amounts in X, Y and Z
- “power-based” history count

We also updated the underlying Geant4 version to Geant4.10.5.p01.

### 5.4.1 3.2.2 (2019-08-15)

**Fixed: Reading of XCAT Phantoms and other simple ImageCube implementations** We have fixed these readings, and `examples/Patient/XCAT.txt` now works correctly.

### 5.4.2 3.2.1 (2019-07-07)

**Fixed: Incorrect placement of the RT Dose Grid** A bug was introduced in release 3.2 that caused the RT Dose Grid to be improperly positioned. This is the grid that is automatically created when one uses the parameter `CloneRTDoseGridFrom`. Placement should now be correct.

**Fixed: Crash if volumetric source is close to the size of the World** In some cases, Topas could crash if a volumetric source was close to the edge of the World. Such placement will no longer cause a problem.

### 5.4.3 3.2.0 (2019-05-02)

**Improved: Simplified Setting of Environment Variables** Where you previously had to set a separate environment variable for each of Geant4’s many data files, you now just tell TOPAS the location of the overall G4Data directory. You can do this by either one of the following two ways:

Set one of these two environment variables:

```
export TOPAS_G4_DATA_DIR = "/PathToDataDir/G4Data"
setenv TOPAS_G4_DATA_DIR "/PathToDataDir/G4Data"
```

Or set a TOPAS parameter:

```
Ts/G4DataDirectory = "/PathToDataDir/G4Data"
```

As always, your syntax will depend on where you actually chose to put your data files, and on whether you are using a system that expects “export” versus “setenv” to set environment variables.

**Improved: DICOM Slice Thickness Attribute is Now Mostly Ignored** We had previously refused to load a DICOM series if the DICOM slice thickness attribute did not match the actual separation between the individual slice positions. However, we have seen that in many DICOM data sets, this attribute does not correspond to the actual spacing of the slices. We now just ignore this slice thickness attribute, and the only case in which we use the slice thickness attribute is if you only load a single slice (so that there is no possibility for us to compute the actual separation between slices).

**Improved: Control the Caching of Material Maps for 4D CT** For 4D CT, the previous TOPAS version always cached all the material maps (a separate map for each CT phase). We did this because we assumed 4D CT users would always want to return to each breathing phase multiple times. However, a user had a case where he had a very large number of maps and only wanted to use each map once. The caching caused him to run out of memory, so a new parameter lets you decide whether such maps should be cached (as before) or not:

```
b:Ge/CacheMaterialMapForEachTimeSlice defaults to True
```

**Fixed: ShowSpecificSlicesVoxel Numbering** The following parameters limit which patient voxels are shown in graphics views. Where slice numbering previously started from 0, it now starts from 1:

```
Ge/ShowSpecificSlicesVoxelsX
Ge/ShowSpecificSlicesVoxelsY
Ge/ShowSpecificSlicesVoxelsZ
```

**Improved: Option to Load a Truncated DICOM, XiO or ImageCube** The previous parameters, `ShowSpecificSlicesVoxels`, affected only the graphical output of the voxels. The voxels were still all loaded, and all affected the physics results, but only specific voxels were shown in graphics. These new parameters, `RestrictVoxels`, have a much more significant effect: they cause TOPAS to not even load some parts of the patient. This can save a lot of time and allow you to test things on DICOM or other image formats quickly, as you only have to load whatever part of the DICOM you actually want to use. But beware: if you use this option, you will change your results. It does not only affect graphics, it actually avoids creating the given voxels at all. Though our main motivation for this work was to get a way to do more rapid development (testing on real patients without the slowdown of loading everything), there may be a value also for our brachytherapy users. They may only need to load the part of the image that is near the brachy source. New parameters allow you to restrict which voxels of a patient image are loaded:

```
i:Ge/*/RestrictVoxelsXMin
i:Ge/*/RestrictVoxelsYMin
i:Ge/*/RestrictVoxelsZMin
i:Ge/*/RestrictVoxelsXMax
i:Ge/*/RestrictVoxelsYMax
i:Ge/*/RestrictVoxelsZMax
```

**Improved: ViewAbdoment.txt Example has been Sped Up** We have modified our simplest example of viewing a DICOM, `examples/Patient/ViewAbdomen.txt` by applying the new `RestrictVoxels` parameters described above.

**Fixed: XCAT Phantom Reading** We fixed an issue with attenuation parameters not being read properly.

**Improved: Energy Binning** We have understood that some users want to use our energy binning (EBins) for a different thing than we designed. From our proton therapy dose calculation roots, the energy binning that we currently do is based not on the energy of the final particle at hit deposition time but instead on the incident particle energy. This is the energy of the final scored particle, or its ancestor, when that particle or ancestor was first incident on

the scoring volume. However, users who have been trying to use this feature to get a spectrum instead need the particle's energy at the current step. So we have now added a parameter to control what kind of energy we use for this binning:

```
s:Sc/*/EBinEnergy = "IncidentTrack" # "IncidentTrack", "PreStep" or
↳ "DepositedInStep"
```

- “IncidentTrack” is the behavior we have had in the past, the energy that the particle or its ancestor had when it first was incident on the scoring component. This remains the default.
- “PreStep” is the track’s energy at the start of the current step.
- “DepositedInStep” is the amount of energy deposited in the current step.
- A new example shows the effect of the three different choices `examples/Scoring/EnergyDepositBinnedByEnergy.txt`

**Added: Option to Score Recursively in Subcomponents** When your scoring component is the parent of other components, you now have the option to set scoring to record particles note only in the parent component, but also in its children. This action is recursive to all levels of subcomponents:

```
Sc/MyScorer/PropagateToChildren = "True"
```

**Improved: Allow nTuple Scorers to Get Binning Information** A helper method, `GetBin`, tells scorers which bin was hit within a divided component. This method was previously only available to scorers that implement `TsVBinnedScorer`. At request of a user, this method has now also been made accessible to scorers that implement `TsVNTupleScorer`.

**Fixed: Binning in Variable Density Materials** A user pointed out that when we generate Variable Density Materials, our binning was slightly off such that the top bin was not quite getting the maximum intended density value. This has been corrected.

**Improved: Scoring Sensitivity in Some Rare Cases of Complex Geometry Components** In the past, when components other than `TsBox`, `TsCylinder` and `TsSphere` were made up of more than one volume, scoring was only sensitive in the mother volume. Scoring is now sensitive in all volumes, however this change is unlikely to affect most users. The scoring was already correct in `TsBox`, `TsCylinder` and `TsSphere`.

**Improved: Redesigned Filtering** We modified the design of our filters to avoid an issue that was bothering the newer compiler included in Debian 9. The only impact on users is that if you have written your own extension filter, you should modify your filter’s constructor to replace `TsVFiltered*` filter with `TsVGenerator*` generator, `TsVScorer*` scorer.

**Added: Volumetric Particle Source** A new Particle Source type, `Volumetric`, has been designed for Brachytherapy applications (though there may be other applications as well). It randomly samples particle starting positions from within the radioactive volume of a given component. When you have a Source of Type “Volumetric”, Topas will expect a second parameter:

```
s:Sc/*/ActiveMaterial
```

to specify which material within the given component should be considered radioactive. So, for example, if you have:

```
s:So/Example/Type = "Volumetric"
s:So/Example/Component = "ActiveSource"
sc:So/Example/ActiveMaterial = "G4_Ir"
```

particles will start from randomly sampled positions within the Iridium parts of the component named `ActiveSource`. See the new example `examples/Basic/VolumetricSource.txt`

**Added: New Geometry Component `TsBrachyApplicator`** This component represents a gynecological applicator. Parameters control overall dimensions and the number of holes into which one can guide a source wire. See `examples/Patient/Applicator.txt` which simulates a Varian VS2000 source wire moving through various dwell positions within a brachytherapy applicator (that is in turn placed within a phantom).

**Improved: EM Physics Models per Region:** You can now select different Electromagnetic Physics models in different regions:

```
s:Ph/Default/ForRegion/MyRegion2/ActiveG4EmModelFromModule = "g4em-livermore"
```

New example for EmPhysics models per region `examples/Basic/EmModelByRegions.txt`

**Added: Auger Cascade** Auger Cascade is now supported.

**Added: New Physics Constructors** New Physics Constructors for G4DNA available in Geant4.10.05 have been added to our modular physics list:

- `g4em-dna_opt6`
- `g4em-dna_opt7`
- `g4em-dna_opt8`

**Added: New Physics Lists** New Physics Lists that artificially maintain constant the incident particle kinetic energy (frozen-velocity approximation), ensuring the correct calculation of stopping power:

- `g4em-dna-stationary`
- `g4em-dna-stationary_opt2`
- `g4em-dna-stationary_opt4`
- `g4em-dna-stationary_opt6`

**Improved: Variance Reduction** We added several new features to Variance Reduction:

- The capability to change VR parameters (e.g. split number) using time features.
- The capability of have more than one VR per simulation (e.g. uniform splitting + imp sampling)
- Forced interaction VR

**Added: New manager for generic biasing process** A new virtual class `TsVBiassingProcess` facilitates the integration of new VRs. Current parameters for VR are still functional, but now they can be renamed different than `Vr/ParticleSplit/XYZ`

**Improved: Examples for secondary biasing** We added several new examples for secondary biasing

- New example for forced interaction `examples/VarianceReduction/ForcedInteraction.txt`
- New example showing cross-section enhancement for bremsstrahlung `examples/VarianceReduction/CrossSectionEnhancement.txt`
- New example showing importance sampling and bremsstrahlung splitting `examples/VarianceReduction/ImportanceSamplingSecondaryBiassing.txt`
- Also shows the capabilities of having more than one VR in the same simulation.

**Fixed: Obscure Problem with Component Placement** Components were incorrectly placed when the first component in a parallel world was a Group Component. Interestingly, we never had any reports of this problem from users. It was only found during some of our own special tests.

**Added: Forbid Use of Parallel Worlds with Chemistry** Since Geant4 DNA's chemistry processes do not support parallel worlds, we have added a check to forbid use of parallel worlds if chemistry is turned on in your physics list.

**Fixed: Diagnostic Printout of TsPropeller Component** The TsPropeller component has a diagnostic that prints out its angle. Where this was meant to print out the full angle, it was actually printing the half-angle. This has been corrected.

**Added: Material can now be 4D** You can now change a component's material over time. You can also change the material that is specified in the DoseToMaterial scorer (though we don't know why you might want to do this).

**Added: Centering Graphics on a Given Component** A new graphics parameter allows you to center the view on a given component:

```
s:Gr/*/CenterOn = "MyComponent"
```

Topas will then ignore any parameters of the form: Gr/\*/TransX or Gr/\*/TransY

**Added: Scaling Graphics by Different Amounts in X, Y and Z** New parameters allow you to scale by different amounts in each dimension:

```
uv:Gr/*/Scale = 3 1. 2. 3.
```

would scale by 1 in X, by 2 in Y and by 3 in Z.

**Improved: Zooming to the Nanometer Scale** With very large Gr/\*/Zoom values, as one might use to zoom into the nano scale, floating point errors in the graphics card lead to discontinuous jumps in the image position. This makes it difficult to center or size the view as one wishes, but an easy workaround has been understood. If one simply makes the world invisible, then because the initial graphics extent only includes the nanometer scale components, one doesn't need much zoom. This effect and its solution are demonstrated in the new example `examples/Graphics/NanoMeterScaleTest.txt`

**Improved: Controlling Console Output** You can add time stamps to the history count:

```
b:Ts/IncludeTimeInHistoryCount = "True"
```

**Added: You can have a “power-based” history count** `b:Ts/ShowHistoryCountLessFrequentlyAsSimulationProgresses`

After first ten histories, output will change to once for every 10, then to once for every 100, etc. An additional optional parameter, `i:Ts/MaxShowHistoryCountInterval`, puts an upper limit on how high the `ShowHistoryCountInterval` can be. For example:

```
b:Ts/ShowHistoryCountLessFrequentlyAsSimulationProgresses
i:Ts/MaxShowHistoryCountInterval = 100
```

Gives: 1, ,2, 3...9, 10, 20, 30...100 but from there always keep counting by 100 (rather than going on to counting by 1000, 10,000, etc.)

**Improved: Handling of Mapped Magnetic Fields** Thanks to several users, we have identified and improved some aspects of the MappedMagnet field reader. The system now works for a wider variety of field maps and allows for 4D changes in the map.

**Improved: Increased the Default Ts/MaxStepNumber** TOPAS watches the number of steps on tracks to try to detect tracks that are stuck endlessly looping. The limit is set by the parameter `Ts/MaxStepNumber`. We have found cases where legitimate tracks take more than 100K steps, so we have increased the default limit from 100K to 1M.

**Improved: Made Parameter File Reading be More Robust** We now catch and replace smart quotes (single and double) with simple double quotes. We also now catch and replace various Unicode hyphen characters with minus sign.



## 5.5 3.1 Series

The main features introduced by the 3.1 series were:

- improved support for DICOM input and output
- added XCAT phantom geometry component
- outcome modeling (e.g. TCP and NTCP)

We also updated the underlying Geant4 version to Geant4.10.3.p01.

### 5.5.1 3.1.2 (2017-10-07)

**Fixed: time varying magnetic or electric field when using more than one thread** If you ran with more than one thread, that is `Ts/NumberOfThreads` not set to its default value of 1, only the first thread was correctly updating fields over time. In other threads, the field never changed from its initial setting. This bug did not affect any other time features, only magnetic and electric fields.

**Fixed: DICOM RT structure sets** Structure contours were incorrectly read from at least some structure set files. This affected both scoring (filtering by structure) and graphics (coloring by structure). Also, graphics was erratic if any structures that were being filtered on were not also being colored by.

**Added: Protection against inconsistent DICOM FrameOfReferenceUID** In a properly constructed DICOM data set, all files from a given study (slices, structures and dose files) will have a consistent `FrameOfReferenceUID`. We now test on these UIDs and give an informative error message if they do not match. You can override this test if you need to by specifying:

```
b:Ge/Patient/IgnoreInconsistentFrameOfReferenceUID = "True"
```

**Improved: ReferencedDicomPatient now inferred from component ancestry** If a scorer does not have the `ReferencedDicomPatient` parameter set, it will automatically be inferred from the ancestry of the scorer's Component.

**Added: Protection against attempts to run with more than one thread in random time mode** Attempting to make a simulation with `Tf/RandomizeTimeDistribution = "True"` faster by using multiple threads actually makes the simulation slower. Geant4 multithreading only saves time if a run has many histories to share among the threads. Our random mode has only one history per run, randomizing the time for the next run. With only one history per run, the overhead from extra, never used threads just slows things down. We now give an informative error message if you try to run in this inefficient way.

**Added: Protection against accidentally using Sc/Type when you should use Sc/Quantity** We have seen cases where users thought the parameter to select a scorer was `Sc/*/Type` (it is actually `Sc/*/Quantity`). We should have made it `Sc/*/Type` since that would be consistent with other parameter names such as `Ge/*/Type` and `Gr/*/Type`, but we can't change it now without breaking existing parameter files (something we only allow ourselves to do for a Major release). We now give an informative error message if we see any cases of `Ge/*/Type`.

**Improved: Protection against two chains of parameter files affecting the same component, scorer, etc** TOPAS includes tests to make sure that no two parameter chains try to affect the same component, scorer, etc. This is to insure that you can safely take parameter chains from some other work group and incorporate them into your own simulation without unexpected effects. This protects against, for example, two groups each defining different components that happen to have the same name. The test was overly tight, making the following situation needlessly forbidden. One chain has: `Ge/MainCylinder/...` Other chain has: `Ge/MainCylinder/ThingInsideIt/...` The code considered these both to be the same component, "MainCylinder". But really they are two different components, "MainCylinder" and "MainCylinder/ThingInsideIt".



### 5.5.2 3.1.1 (2017-06-02)

**Improved: Reduce sensitivity of test for stuck tracks** In Topas 3.1, we introduced a test to catch a rare problem in which Geant4 stops propagating any tracks but still runs to completion, resulting in misleading results. The test looks for cases where a history has only a single track, and that track has only two steps and that track does not exit the world. In the case where Geant4 got stuck, we saw that once the job was in trouble all histories had this signature. A user has since shown us a case where this test was triggered when the simulation was not actually stuck (thanks Eelco Lens for sharing your example with us - it involved a particle with nearly zero energy emitted by a phase space source).

With this patch, we make the test less sensitive and make it controllable by two new parameters. Only quits now if 10 such histories are found in a row. The number of such histories allowed in a row can be adjusted by:

```
i:Ts/NumberOfAnomalousHistoriesToAllowInARow = 10
```

And the test can be disabled completely by setting:

```
b:Ts/QuitIfManyHistoriesSeemAnomalous = "False"
```

### 5.5.3 3.1.0 (2017-05-09)

**Improved: Updated Geant4 to latest version 10.03.p01** Includes new physics modules such as g4em-standard\_SS and new EM options: EMBins and EMBinsPerDecade.

**Improved: Extensions Interface** You can now add your own physics modules for use in modular physics lists. You can have more than one extension directory: semicolon-separated paths in cmake argument TOPAS\_EXTENSIONS\_DIR. TOPAS now knows to ignore any hidden directories within your extensions directories (for example .DS\_Store directories). Added support for gcc compiler versions later than 5.0.

**Improved: DICOM tags of output files set appropriately** Some metadata tags (Study Instance UID, Frame of Reference UID) are copied from input DICOM (TsDicomPatient) to output DICOM (the scorer), which is important for data provenance:

- The metadata source can be specified by the new parameter: ReferencedDicomPatient. This is helpful when scoring on a TsBox.
- Otherwise, the metadata is copied from the scorer's Component (if it is a TsDicomPatient)
- Otherwise, the metadata is generated by TOPAS

Other metadata tags (SOP Instance UID, Series Instance UID, Series Description, Manufacturer, Manufacturer's Model Name, Dates and Times) are now set appropriately. It is also possible to set a custom Series Description using the new SeriesDescription parameter.

**Added: Automatically create a Scoring Grid that exactly matches a provided RTDOSE file in your DICOM dataset**

This makes it easier to compare TOPAS results to Treatment Planning System results. Tell TOPAS which RTDOSE file to use by providing a "CloneRTDoseGridFrom" parameter, such as:

```
s:Ge/Patient/CloneRTDoseGridFrom = Ge/Patient/DicomDirectory + "/RTDOSE.dcm"
```

TOPAS will then automatically create a scoring volume in a parallel world to overlay your grid, and will name this component with the same name as your patient, plus "/RTDoseGrid". You can then score on this component just like on any other component:

```
s:Sc/Dose/Component = "Patient/RTDoseGrid"
```

**Added: Automatically set DicomOrigin parameters to help with patient positioning** If you define a set of DicomOrigin parameters for your patient:

```
dc:Ge/Patient/DicomOriginX = 0.0 mm
dc:Ge/Patient/DicomOriginY = 0.0 mm
dc:Ge/Patient/DicomOriginZ = 0.0 mm
```

then when you read in a `TsDicomPatient`, TOPAS will update these parameters on the fly to provide the origin of the DICOM coordinate system specified in the TOPAS coordinate system. You can combine this information with other information you may have about your isocenter to get your patient properly positioned.

For example, if you just wanted to center your patient in its parent component, such as `PatientGroup`, you would do:

```
s:Ge/Patient/Type = "TsDicomPatient"
s:Ge/Patient/Parent = "PatientGroup"
d:Ge/Patient/TransX = 0.0 mm
d:Ge/Patient/TransY = 0.0 mm
d:Ge/Patient/TransZ = 0.0 mm
```

If you also had isocenter information from at RT-Plan in DICOM coordinates:

```
d:Rt/plan/IsoCenterX = 0.0 mm
d:Rt/plan/IsoCenterY = -99.9904 mm
d:Rt/plan/IsoCenterZ = -14.0 mm
```

you could adjust the patient to isocenter by doing:

```
d:Ge/Patient/TransX = Ge/Patient/DicomOriginX - Rt/plan/IsoCenterX mm
d:Ge/Patient/TransY = Ge/Patient/DicomOriginY - Rt/plan/IsoCenterY mm
d:Ge/Patient/TransZ = Ge/Patient/DicomOriginZ - Rt/plan/IsoCenterZ mm
```

A new example shows how to use the new features: [ViewAbdomen\\_rtdose.txt](#).

**Added: Read XCAT Phantoms and any other user-defined simple cube of imaging values** This new component type, `TsImageCube`, replaces and extends our previous `TsXiOPatient`. Data can be either short, int or float values. Conversion of imaging values to materials can use `XCAT_Attenuation` or `XCAT_Activity` values from an XCAT log file, or some other conversion class you provide yourself through our extensions mechanism, or can take values that you set directly in TOPAS Parameters. See [documentation](#) and example in [XCAT.txt](#).

**Fixed: Adjusted the names of some of the materials automatically created during HU conversion** Where the HU number was negative, automatically generated material names were such as:

```
Ma/PatientTissueFromHU-295
```

But we've said that the minus sign is a reserved character, not allowed in parameter names. So we now create material names such as:

```
Ma/PatientTissueFromHUNegative295
```

Should have no impact on users as these material names are both created and used automatically.

#### Improved: Other DICOM improvements

- Updated DICOM toolkit `gdcm` to latest version 2.6.8
- DICOM scoring output is now stored in RTDOSE files (previously stored in CT files)
- Corrected the voxel size in DICOM output
- Automated scaling so that `DICOMOutputScaleFactor` parameter is no longer needed
- Added support for negative values in DICOM output (e.g. charge scoring)

- Gave DICOM output a TOPAS-specific root UID: 1.2.826.0.1.3680043.9.5871.
- Fixed coordinate systems of DICOM input and output, which also affected RTStruct filtering
- Added check that where DICOM output is specified, the scoring component is appropriate (e.g. TsBox or TsPatient)
- Renamed examples/DICOM to examples/Patient. The change was made since we support not just DICOM but also XiO, XCAT and arbitrary image cubes.
- TsDicomPatient no longer requires overall material parameter

**Added: Outcome modeling** TOPAS can now directly perform Outcome Modeling such as calculating Tumor Control Probabilities and Normal Tissue Complication Probabilities. Expanding on TOPAS previous capability to directly produce a Dose Volume Histogram, TOPAS can now directly apply outcome models to the DVH. We provide a variety of standard outcome models from the literature, for each of which you can adjust various parameters. See [documentation](#) and [examples](#).

We also allow you to read back in a previously created DVH to have TOPAS apply new outcome models without having to re-do the Monte Carlo simulation phase of the job.

You can also supply your own outcome model via the TOPAS extensions interface.

**Added: New options when reading phasespace files** Phase space source can now scale the particle start positions:

```
u:So/MyPhaseSpaceSource/PhaseSpaceScaleXPosBy = 0.1
u:So/MyPhaseSpaceSource/PhaseSpaceScaleYPosBy = 0.1
u:So/MyPhaseSpaceSource/PhaseSpaceScaleZPosBy = 0.1
```

To ignore a position, scale to zero, as in:

```
u:So/MyPhaseSpaceSource/PhaseSpaceScaleXPosBy = 0.
```

The previous way to ignore a position was:

```
b:So/MyPhaseSpaceSource/PhaseSpaceIgnoreXPos = "True"
```

is still supported, but is deprecated and will be removed at the next major release.

**Added: Reading Limited Phase Space can now handle files that lack NewHistory flags** We found that some files from other vendors that are supposed to have NewHistory flags do not have them, so that TOPAS could not tell which particles were new histories. These files seemed to have the assumption that all photons are new histories. To support this, we added a parameter:

```
b:So/MyPhaseSpaceSource/LimitedAssumePhotonIsNewHistory = "True"
```

**Added: Phasespace scorer can output creator process** Phase space output can now include Creator Process Name:

```
b:So/MyPhaseSpaceSource/IncludeCreatorProcessName = "True"
```

**Improved: Made PhaseSpace source accept time-varying NumberOfHistoriesInRun** This option is only allowed if you have MultipleUse = 0, meaning you intend to explicitly say how many histories to use, rather than running through the entire phase space file 1 or more times.

**Added: Protection against unreasonable setups of phase space input** Do not allow PhaseSpaceMultipleUse is negative.

Do not allow PhaseSpaceMultipleUse to be controlled by a time feature.

Do not allow phase space source with empty histories when there are time features since we can't tell where in the time sequence these empty histories are supposed to occur (we will address this with an addition to the TOPAS phase space formats in the next release).

**Improved: Extension scorers C++ interface** When developing extension scorers, the C++ class interface has changed when using sub-scorers. This does not affect the parameter interface. Sub-scorers are now assigned names, which are used for identification in the `CombineSubScorers()` method. This also allows sub-scorers to be shared between scorers, which reduces the memory used by a simulation (e.g. multiple RBE scorers can reuse Dose and LET scorers). The new C++ interface is demonstrated in `ExtensionExamples/More/MyScoreProtonLET`. See the [Custom Scorers](#) for details.

**Improved: Improved handling of empty runs in scoring** When scoring has `OutputAfterRun`, but a particular run has zero histories (as may happen during beam current modulation), TOPAS was not producing any output file for the run. This was occurring because Geant4 itself does not actually increment its `RunID` for such empty runs. However this is not what users want when they have `OutputAfterRun`. TOPAS now maintains its own `RunID` that includes empty runs. Empty runs will now have output files just as non-empty runs will.

**Improved: Other scoring improvements**

- Added example of `OpticalPhotonCount` scorer: [OpticalPhotonCount.txt](#)
- Added a more complex example of a user-written Ntuple Scorer
- Fixed bug in `SplitByTimeFeature` with step function of a double parameter. Reported by Weiguang Yao in user forum.
- Added Number of Entries to Ntuple headers
- Protected against attempts to score on a Group Component
- Added track vertex 4-momentum to information in `TsTrackInfo`. Users can access this information when writing custom scorers or filters.

**Improved: Default color definitions to match HTML 4.01 standard** Some of the color values we had were strange leftovers from some very very old code. We now use the standard values from the HTML 4.01 standard as described [here](#). As with any parameters, you are free to redefine these in your own parameter files.

**Added: Support for transparency** Where color parameters used to take just three integer values (0 to 255) for the three color components, they now allow an optional fourth integer value (0 to 255) for the alpha value. So, for example:

```
iv:Gr/Color/TransparentYellow = 4 255 255 0 50
```

If the alpha value is omitted, the color is fully opaque.

**Improved: Increased default value of `Gr/SwitchOGLtoOGLIfVoxelCountExceeds`** This parameter controls when TOPAS switches from using the “Stored” mode of OpenGL to using the “Immediate” mode. Stored allows for faster re-rendering when the view changes. Immediate mode does not re-render as quickly, but uses less RAM. Value was increased from 3 million to 70 million voxels.

**Improved: Other graphics improvements**

- Made division lines in divided components match color of the overall component. These were previously showing as white no matter what color the component was.
- Fixed various issues with having multiple graphics views at same time.

**Improved: Faster overlap checking** Overlap checking previously spent some unnecessary time checking whether the divisions within a divided component (such as the voxels in a `TsBox`) overlapped each other. Since these voxels are generated automatically by TOPAS, you can trust that they do not overlap. If you really want to turn this overlap checking back on, set:

```
b:Ge/CheckInsideEnvelopesForOverlaps = "True"
```

**Added: More control over precision of overlap check** Geant4's overlap checking works by randomly placing points on the surface of a solid, and then checking whether any of these points are inside another solid that is not a mother. You can now control the number of such points:

```
i:Ge/CheckForOverlapsResolution = 1000
```

And you can check the tolerance for overlap:

```
d:Ge/CheckForOverlapsTolerance = 0. mm
```

You can also set these in a more granular fashion, per Component (overrides the above parameters for this particular component):

```
i:Ge/MyComponent/CheckForOverlapsResolution = 1000
d:Ge/MyComponent/CheckForOverlapsTolerance = 0. mm
```

#### **Improved: Other geometry improvements**

- Added ability for World to be a Sphere or a Cylinder. World was previously always a Box.
- Removed limitation on number of parallel worlds. Previously had limit of 8 parallel worlds. There is no longer any limit (but you should avoid using more than necessary as they may slow performance).

**Improved: Trap anomaly in which Geant4 appears to run through entire job, but no particles propagate** This is a rare anomaly. We do not yet understand what causes it, but we have seen it start happening in otherwise reasonable setups after some random large number of histories. Once the anomaly sets in, the job would appear to run successfully to completion, but no particles actually propagate, and thus nothing is scored. We have found that a signature of this condition is that the entire history has only one step and the endpoint of that step is not on the world volume. We now watch all histories for this signature, and exit the job if this signature is detected. The workaround we have for now is to then try the same job with a different random seed.

#### **Improved: Other miscellaneous improvements**

- Added checks that integer parameters are within bounds. Covers input values up to 9223372036854775807.
- Added protection against missing parameter type letter before first colon. Catches mistake such as:

```
:Ge/MyBox/Type = "TsBox"
```

- Improved various error messages about inappropriate parameter formatting
- When dumping parameters to file per run, file names now have underscore after "Run". This makes parameter dump file naming consistent with scoring file names.
- Fixed bug that made some particle sources give too many histories when there were multiple sources in multithreaded mode
- Improved error trapping for case of invalid particle name set for variance reduction
- Corrected units in information printout from propeller
- All MaxStepSize in components that have parallel scoring copies. TOPAS was previously trying to apply the MaxStepSize to the parallel copy. This is not appropriate and was causing Geant4 to fail.
- NumberOfHistoriesInRandomJob is now a required parameter when running in random mode. This fixes a bug where users of RandomizeTimeDistribution saw the Demo source produce histories even if it was not wanted.

## 5.6 3.0 Series

The main features introduced by the 3.0 series were:

- n-tuple output from custom scorers
- new extensions mechanism
- redesigned beam source parameter interface
- generalized imaging to material conversion
- electric fields
- faster graphics performance

We also updated the underlying Geant4 version to Geant4.10.2.p01.

We took the opportunity of a major release to revise some parameter names with the benefit or hindsight and looking towards future releases. Consequently, we provided a `topas2to3` script to help in upgrading parameter files.

### 5.6.1 3.0.1 (2016-06-02)

**Fixed: drawing to multiple OpenGL graphics views not working correctly** We have been struggling with some issues in OpenGL graphics since we moved to Geant4.10.02. In particular, there were issues if you drew more than one view at a time. We now believe this is fixed, with the exception of the Debian8 build. Behavior is highly dependent on the specific operating system, and we (and Geant4) are still working to get this just right.

**Fixed: compiler issues when adding extensions** Building with extensions should now work on all operating systems. It is no longer necessary to run `cmake` twice in a row, and our `cmake` does a better job now of setting the required C++11 flags. You will still need to update your compiler if your default compiler does not support C++11. This is now discussed in the last section of the README.

**Fixed: bug in filter extensions** Can now add multiple filter extensions simultaneously.

**Improved: geometry component types now case-insensitive** It no longer matters what case you use in `Ge/*Type` values. So, for example, you could use `"tsBox"`, `"TsBox"` or `"tsbox"`. This also applies to user-supplied extension components.

### 5.6.2 3.0.0 (2016-06-01)

**Added: n-tuple output from custom scorers** If you're not familiar with n-tuples, you may be interested to know that you've already been using a version of one if you were outputting particle information to phase space.

Our new design gives you very fine control over what information your scorer will write.

In your scorer's constructor you define each column and its data type. For floating point columns you also specify the unit string, similar to how the `SetUnit()` method is used in other scorers. In the `ProcessHits()` method, you then compute the variables you want to output and then call `fNtuple->Fill()`. Buffered file writing is automatically handled. Output can be to ASCII, binary and ROOT data formats.

**Added: phase space scorers can be output to ROOT format** This is a binary data format associated with the ROOT data analysis framework.

**Improved: overhaul of extension mechanism** No more editing the `CMakeLists` file or adding a clause into the `TsExtensionManager.cc`.

Just create a folder anywhere on your computer, put your extensions into it (you can even organize them by subfolders within this folder), add one special comment line to the top of each `cc` file, and run `CMake`, telling

it where your folder is with the `-DTOPAS_EXTENSIONS_DIR` option. Our CMake scripts will then automatically interweave your code in with ours, and all will work. Please note that, with the introduction of n-tuple scorers, existing scorers will require some additional modifications to distinguish them as “binned scorers”.

**Improved: revised parameter interface for beam and isotropic sources** You can now provide a cutoff (like a collimation) to the angular spread distribution. This improvement was sparked by a request from Christian Sommer.

The overall parameter set has been revised to be, we think, easier to understand and remember.

- `Type = {Beam, Phasespace, Emittance, Isotropic}`
- **BeamPositionDistribution = {None, Flat, Gaussian} (if Beam)**
  - `BeamPositionCutoffShape = {Rectangle, Ellipse} (if Flat/Gaussian)`
  - `BeamPositionCutoffX/Y (if Flat/Gaussian)`
  - `BeamPositionSpreadX/Y (if Gaussian)`
- **BeamAngularDistribution = {None, Flat, Gaussian} (if Beam)**
  - `BeamAngularCutoffX/Y (if Flat/Gaussian)`
  - `BeamAngularSpreadX/Y (if Gaussian)`

The previous option `BeamShape = "Point"` is now chosen via `BeamPositionDistribution = "None"`, and the previous option `BeamShape = "Isotropic"` is now chosen via `Type = "Isotropic"`.

Here is the list of parameters that already existed, but now have new names:

- `BeamXYDistribution` → `BeamPositionDistribution`
- `BeamShape` → `BeamPositionCutoffShape`
- `BeamHWX` → `BeamPositionCutoffX`
- `BeamHWY` → `BeamPositionCutoffY`
- `BeamStandardDeviationX` → `BeamPositionSpreadX`
- `BeamStandardDeviationY` → `BeamPositionSpreadY`

**Improved: default source removed** The default particle source parameters, `So/Default/*`, have been removed. While revising particle sources, we changed our mind about whether we should provide any default particle source parameters. We see so many differences in what different users need, that we now believe it is important that every user carefully consider all of their particle source parameters. Accordingly, we removed the default parameters `So/Default/*`. All examples now include a full specification of their particle source parameters.

We still bundle a built-in source, but we have renamed it `So/Demo` to emphasize that it should be used only for demonstration and teaching purposes. Users who write their own parameter files should fully specify their Particle Source.

**Improved: changed the default water material** We removed our custom Water (mean excitation energy 75 eV) in favor of `G4_WATER` (78 eV).

For some time, we had been using a custom material, Water, with a mean excitation energy of 75 eV, instead of Geant4’s pre-defined `G4_WATER`, which has a mean excitation energy of 78 eV. With newer Geant4 physics, we find we get better agreement at MGH when we use 78 eV. Accordingly, we have removed our material Water and switched all of our examples to `G4_WATER`.

We created a new material named `Water_75eV` in case you have calibrated with this material and really want to stick with that.

**Added: support for additional imaging to material conversion** Support Dual Energy CT, Multi-Energy CT and other complex ways of assigning material. We have generalized our imaging to material conversion. You can now provide your own extension class to control how TOPAS assigns materials in the patient. Input can come from one or more image files and the imaging modality is not limited to CT. TOPAS pulls the information out of the images for you, and gives your class the one or more values per voxel. Your class then determines what material to then assign for this voxel.

New parameters are:

```
i:Ge/Patient/NumberOfEnergies = 1 # defaults to 1
sv:Ge/Patient/DicomModalityTags = 1 "CT" # defaults to just CT
s:Ge/Patient/ImagingtoMaterialConverter = "Schneider"
```

You will note that we renamed the parameter `HUtoMaterialConversionMethod` to `ImagingToMaterialConverter` to emphasize that input need not necessarily be Hounsfield Units.

**Added: field handling extended to include electric fields** Where before we had:

```
Ge/MyComponent/MagneticField = "Dipole" # or "Quadrupole", "Map"
```

We now have:

```
Ge/MyComponent/Field = "DipoleMagnet" # or "QuadrupoleMagnet", "MappedMagnet",
↪ "UniformElectroMagnetic"
```

The latter can have both magnetic and electric components. To specify a pure electric field, use “UniformElectroMagnetic” while specifying a zero `MagneticFieldStrength`. You can also write your own extension field class that provides any other Electric, Magnetic or ElectroMagnetic field.

To make way for ElectricField parameters, we renamed some parameters:

- DirectionX → MagneticFieldDirectionX
- DirectionY → MagneticFieldDirectionY
- DirectionZ → MagneticFieldDirectionZ
- Strength → MagneticFieldStrength
- GradientX → MagneticFieldGradientX
- GradientY → MagneticFieldGradientY
- 3DTable → MagneticField3DTable
- Stepper → FieldStepper
- StepMinimum → FieldStepMinimum
- DeltaChord → FieldDeltaChord

**Improved: magnetic field setup from 3D field maps** The new code does a much better job of handling various field maps. Thanks to Eric Able of Varian Medical Systems for his extensive prototype work.

**Added: support for variable density materials** Driven by needs from imaging to material conversion, we have added a way that you can easily define a set of materials that differ only in density:

```
i:Ma/MyMaterial/VariableDensityBins = 100
u:Ma/MyMaterial/VariableDensityMin = .1
u:Ma/MyMaterial/VariableDensityMax = 10.
```

will generate 100 versions of `MyMaterial`, with densities varying from .1 x normal to 10. x normal.

The material names will then be like:



```
MyMaterial_VariableDensityBin_0
MyMaterial_VariableDensityBin_1
...
MyMaterial_VariableDensityBin_99
```

Note that numbering starts at zero.

**Improved: support for multiple slice thicknesses in TsDicomPatient** This capability was previously restricted to the TsXioPatient.

**Improved: OpenGL graphics are dramatically faster** Especially true for patient geometry. If you were avoiding displaying patient geometry, it's time to try it again.

**Added: new export options for OpenGL Graphics** Where before we had:

```
Gr/MyView/CopyOpenGLToEPS
```

we now have that plus three new options:

```
Gr/MyView/CopyOpenGLToPDF
Gr/MyView/CopyOpenGLToSVG
Gr/MyView/CopyOpenGLToPS
```

**Added: new syntax for specifying vectors** New parameter expressions let you set vector parameters from other vector parameters:

```
dv = name_of_double_vector_parameter unit
dv = number * name_of_double_vector_parameter unit
dv = name_of_unitless_or_integer_parameter * name_of_double_vector_parameter unit
uv = name_of_unitless_vector_parameter
uv = number * name_of_unitless_vector_parameter
uv = name_of_unitless_or_integer_parameter * name_of_unitless_vector_parameter
iv = name_of_integer_vector_parameter
iv = integer * name_of_integer_vector_parameter
iv = name_of_integer_parameter * name_of_integer_vector_parameter
bv = name_of_boolean_vector_parameter
sv = name_of_string_vector_parameter
```

This makes it easier to adjust existing vector parameters in file hierarchies.

**Improved: G4Box, G4Tubs and G4Sphere components removed** You should instead use TsBox, TsCylinder and TsSphere. These provide all the same functionality, but also support divisions. We have done this both to simplify the underlying TOPAS code (simpler means less likelihood for bugs), and because we have seen many cases where someone tried to apply divisions to G4Box, G4Tubs or G4Sphere, and had a hard time figuring out why this wasn't working (this has even happened to us during live demos).

**Added: specify material per voxel for any divided component** This means you can create complex phantoms directly from the parameter system:

```
sv:Ge/Phantom/VoxelMaterials = 100 "G4_WATER" "G4_WATER" "Air" "Air" "G4_WATER" ..
→.
```

Works for all three kinds of divided components: TsBox, TsCylinder and TsSphere.

**Improved: TsBox and TsCylinder allow parameterizations in parallel worlds** The underlying limitation that was preventing this has been resolved in the new Geant4 version. The restriction against using parameterization within a parallel world now only applies to TsSphere (as the underlying issue for this case is still present in Geant4).

**Improved: support larger numbers of histories** Various Counters have been changed from `int` to `long` to accommodate larger numbers of histories. With the move to multi-threading, we now have users running so many histories in a single session that various counters exceeded the size of our internal counters. There remain some limits within Geant4 itself, so we enforce a maximum of  $10^9$  histories per run.

We also found a way to allow you to have more than  $10^9$  histories in a single TOPAS session. The solution is to break these histories into multiple Geant4 runs. Originally, the parameters `Tf/TimeLineStart`, `Tf/TimeLineEnd` and `Tf/NumberOfSequentialTimes` were intended to let you have different runs at different times (TOPAS Time Features). But if you leave `TimeLineEnd` the same as `TimeLineStart` (and by default they are both 0), and just set `Tf/NumberOfSequentialTimes` to some value greater than 1, you will have multiple runs, and each can have up to  $10^9$  histories, but the total can be much larger.

**Improved: ProtonLET scorer extended to very low density materials** Our current LET scorer gives values that are too high in air, where the mean path length between discrete processes can be larger than the voxel size. This can be avoided by neglecting secondary electrons, so we introduce the `NeglectSecondariesBelowDensity` parameter, whose default value is 0.1 g/cm<sup>3</sup>.

Even when you do this, rare events that produce very low energy protons (e.g. a recoiling hydrogen nucleus) will produce spikes in LET. This is also seen in the `PreStepLookup` version of the scorer. They are not seen in the fluence-averaged version of the scorer, since they are rare events. For this reason we introduce a `UseFluenceWeightedBelowDensity` parameter, whose default value is zero. We disable this by default because it is strange to mix both types of LET in a single distribution, and could be significantly wrong at the end of range. We expect users to want to enable this when making a pretty plot of LET to overlay on a CT scan, without spikes in cavities and outside the patient.

**Improved: convenience method GetIndex for custom scorers** Scorers can now easily obtain the voxel indices from hits in divided or parameterized components. The base class `TsVScorer` now provides a convenience method, `G4int GetIndex(G4Step*)`. This is convenient for some expert users and also hides the `GetIndex` method that we don't want people trying to use from the `G4VPrimitiveScorer` (since the latter doesn't perform as the user would expect).

**Improved: DoseToWater and DoseToWaterBinned scorers are unified** `DoseToWaterBinned` was a way of scoring dose to water that improved speed at some cost to accuracy by pre-calculating stopping power ratios. We now offer only one scorer, `DoseToWater`. To get the previous behavior of `DoseToWaterBinned`, add the optional parameter:

```
b:Sc/MyScorer/PreCalculateStoppingPowerRatios = "True" # defaults to "False"
```

The same parameter is also available for the `DoseToMaterial` scorer.

**Improved: removed our custom EM Physics Module from our Default Physics List** In TOPAS 2.0 we provided a custom EM physics module, `tsem-standard_opt3_WVI`, that attempted to use the new WentzelVI model of multiple Coulomb scattering (MCS). Now that this MCS model has been fully incorporated into the Geant4 built-in physics module, `g4em-standard_opt4`, we switch to using this in our default physics list and remove the custom module.

**Improved: removed the G4RadioactiveDecay module from our Default Physics List** We have found that the `G4RadioactiveDecay` process sometimes causes errors such as:

```
G4Exception : de0001 issued by : G4AtomicTransitionManager::Shell()
No de-excitation for Z= 3  shellIndex= 2>=  numberOfShells= 2 AtomicShell not_
↳found
```

Since this module is not needed for most simulations, we have removed it from the default. If you really want this process, you can add it back to `Ph/Default/Modules`.

**Fixed: renamed surfaces of TsCylinder and TsSphere** We have revised the names of Surfaces to have a more consistent overall design. `Phi/Theta` now have `Plus/Minus` afterwards, like `X/Y/Z`.

- PlusPhiSurface → PhiPlusSurface
- MinusPhiSurface → PhiMinusSurface
- PlusThetaSurface → ThetaPlusSurface
- MinusThetaSurface → ThetaMinusSurface

**Fixed: corrected some surface area calculations**

- TsCylinder: calculation was wrong for area of Z surfaces and curved surfaces.
- TsSphere: calculation was wrong for area of curved surfaces when there was a phi cut.
- TsSphere: calculation was wrong for area of phi and theta cut surfaces.

We are sorry to have allowed these errors to slip through our testing process. Thanks to Christian Sommer for alerting us to the first of these (which led to a full review).

**Added: other minor features**

- We added a way to have TOPAS list all processes in the currently selected physics list:

```
b:Ph/ListProcesses = "True"
```

- Topas can now tell you it's version information: just type: `topas --version`

**Fixed: other minor bugs**

- Removed need for BeamEnergy and BeamEnergySpread when source is spectrum.
- Removed the particle source type Twiss as source type Emittance does the same and more.
- The angular generation for Beam sources has been corrected so that it is valid beyond the small-angle approximation.
- The base class for scorers, `TsVScorer.hh`, now includes `G4SystemOfUnits.hh`. Scoring often uses units, and this should make everyone's life easier.
- We removed the requirement that some water be present in the simulation when using the DoseToWater and DoseToMaterial scorers.
- In time feature random time mode, `Ts/ShowHistoryCountAtInterval` now counts runs rather than events.
- The material name `Flourine` has been corrected to `Fluorine`.
- The parameter `Ph/*/LamdaBins` has been corrected to `Ph/*/LambdaBins`.
- `PhaseSpaceBufferSize` is now called `OutputBufferSize`. We renamed this parameter as it now applies not just to phase space but also to n-tuples.
- Sources now move correctly when the source component's parent component is moving. This situation used to work only when the source component's parent was a group component. It now works correctly for all cases. Thanks to Christian Sommer for showing us this bug.
- Phase space source now correctly handles all ions. Some ions were previously being forbidden in the phase space source. Thanks to Vadim Moskvina for reporting this bug.
- Solved bug that was causing part of phase space file to be used by two separate threads. When a multi-threaded session was using a phase space source, histories were being incorrectly assigned to the worker threads, causing some histories at the end of the file to be used more than once. Thanks to Hugo Moreira for showing us this bug.

## 5.7 2.0 Series

The main feature introduced by the 2.0 series was multithreading. In doing so we changed the underlying Geant4 to version Geant4.10.1.p02. This upgrade required extensive recoding within TOPAS, but was done in a way that required almost no changes to Parameter Files.

### 5.7.1 2.0.3 (2016-01-12)

**Fixed: bug affecting scoring in divided TsBoxes, TsCylinders and TsSpheres** This caused some of the dose due to secondary particles to be assigned to wrong divisions. This bug was introduced when we switched to the Geant4.10 series, so began at Topas 2.0.

**Fixed: bug in DICOM output** Dose value per voxel index was correct, but the entire structure was being drawn too small.

**Fixed: bug using “Map” magnetic field type** In multi-threaded mode, field only took effect after the first run.

**Fixed: bug in Phase Space output** Was crashing for cases of `Ts/NumberOfThreads = 0` (meaning use all threads).

**Fixed: a type in the header of DoseToWaterBinned scorer** Header had some extraneous text left over from an earlier design.

**Improved: restored two convenience methods to TsVScorer** `GetRunID` and `GetEventID` are once again available for use by user-written scorers.

### 5.7.2 2.0.2 (2015-11-18)

**Fixed: segfaults from secondary biasing** Filtering on whether a particle interacted in a given component was failing to notice the interactions. Thanks to Gray Lu and Christian Sommer for reporting these bugs.

**Fixed: first few histories could ignore filters when using multiple threads** In testing our fix for the particle interaction filter, we also found a more subtle bug that could cause the first few histories to ignore any filter when running on more than one thread.

### 5.7.3 2.0.1 (2015-11-13)

**Fixed: scorers report incorrect “Sum” in a specific case** Bug triggered when you have more than one bin (that is, where XBin, YBin, ZBin, RBin, PhiBin or ThetaBin is more than one) **and** the scorer’s Report options are set to exactly “Sum” and “Mean”.

### 5.7.4 2.0.0 (2015-11-04)

**Added: multithreading support** Set `i:Ts/NumberOfThreads` to the number of CPU threads you want to use. If set to a positive integer, TOPAS will use that number of threads. If set to 0, TOPAS will use all of your computer’s threads (may be number of hardware cores or number of virtual cores (which includes hyper-threading cores) depending on your hardware architecture. If set to a negative number, TOPAS will use all BUT this number of threads, leaving you some threads reserved for other tasks (email, web browsing, etc.).

**Improved: updated physics lists for Geant4.10.1.p02** We have upgraded the default physics list to what we believe is the best option for proton therapy dose calculation in this Geant4 release. Remember that the default physics settings in TOPAS may not be the best settings for your own work. The only assurance we can give you is that the default settings are what we have currently chosen for proton therapy dose calculation research at our home

institutions of MGH and UCSF. Geant4 physics changes in each release, and it is always the user's responsibility to perform any validations and adjustments that may be required.

**Improved: magnetic field is now just a parameter on any component** You no longer specify special components for magnetic fields. Instead, the field is just an extra parameter that you can set for any of the standard components. So where you used to have:

```
Ge/MyComponent/Type = "TsDipoleMagnet" # or "TsQuadrupoleMagnet",
↳ "TsTabulated3DField"
```

you now use:

```
Ge/MyComponent/Type = "TsBox" # or "TsCylinder", etc
Ge/MyComponent/MagneticField = "Dipole" # or "Quadrupole", "Map"
```

which gives more flexibility.

**Added: quadrupole magnetic more flexible** Can now have separate GradientX and GradientY

**Added: visualization of magnetic fields** Field intensity and direction are represented as a set of arrows. The arrow density is controlled by:

```
i:Gr/ViewA/MagneticFieldArrowDensity = 10
```

Use with caution. When combined with rotation it sometimes causes crashes in polycone drawing (involved in drawing the arrowheads).

**Added: magnetic fields to extensions mechanism** [See here](#) for how to add custom magnetic fields.

**Improved: overlap checking is more strict** Geometry Overlaps previously caused only a warning. This meant that users might not even notice that their simulation had this dangerous geometry problem. TOPAS now is set to quit if any overlap is detected. If you really want TOPAS to continue, you can set:

```
Ge/QuitIfOverlapDetected = "False"
```

You will then still get a warning when the overlap is detected, and another warning at the end of the session. As before, you can turn off overlap checking entirely with:

```
Ge/CheckForOverlaps = "False"
```

This saves a small amount of time at startup, but is only recommended in cases where you are running a setup that you have already extensively tested.

**Added: “Twiss” source type replaced by “Emittance”** The new source type provides more flexibility than the previous Twiss ([see here](#)).

**Improved: user extensions require updating**

- If you have written your own Scorer in C++, you will need to add one additional argument to the constructor and pass this argument on to the TsVScorer.
- If you have written your own Particle Source in C++, you will need to redesign this to have separate TsSource and TsGenerator.

**Fixed: rare bug affecting some apertures** We have found rare cases in which an Aperture leaked dose (particles passed through one part of the aperture as material as if there was no material present). This was traced to an underlying bug in Geant4's TessellatedSolid. While the bug is not yet fixed in Geant4, we now trap it and interrupt the relevant history. If this bug affects your session you will see warning message each time it occurs, plus a summary about this bug at the end of the console. An additional parameter aborts the session if this bug is found more than a specified number of times:

```
i:/Ts/MaxInterruptedHistories = 10 # defaults to 10
```

We do not recommend setting this to be a significant fraction of the total number of simulated histories.

**Added: new examples** Random versus Sequential Time Feature modes are demonstrated in two new examples ([Run-Random\\_Mode.txt](#) and [RunSequential\\_Mode.txt](#)). Bremsstrahlung splitting is demonstrated in a new example ([SecondaryBiasing.txt](#)).

## 5.8 1.X Series

### 5.8.1 1.3.0 (2015-10-21)

**Improved: scoring report option “Min\_Max” is split into “Min” and “Max”** If you have previously used the option “Min\_Max”, you will need to replace it with “Min” “Max”.

**Added: new ways to dump parameters to text files** `Ts/DumpParametersToSimpleFile` takes a list of parameter names and dumps their types, names and values to a simple, human-readable file. `Ts/DumpParametersToSemicolonSeparatedFile` takes a list of parameter names and dumps their types, names and values to a semicolon separated file, suitable for easy import into other applications.

**Added: method of defining elements** Elements can now be defined from specific combinations of isotopes (rather than from natural abundance).

**Improved: zero padding of RunID in output file names** For example, `MyScorerOutput_Run_0001.csv` rather than `MyScorerOutput_Run_1.csv`, so that they will sort naturally in various file viewers. The number of padding places can be modified by the parameter `i:/Ts/RunIDPadding`.

**Improved: scoring filters can now specify ion charges** Source ions must be fully stripped (as before), but this restriction is no longer applied to scoring filters. For example:

```
sv:Sc/IonsWithCharge3to5/OnlyIncludeParticlesNamed = 1 "GenericIon(6,12,5)"
```

will only score those carbon ions that have charge of 5.

**Improved: console output of phase space scorers** Additional information reported to the console.

**Improved: filtering example** Now includes `InitialParticle` filters (see [examples/Scoring/Filters.txt](#)).

**Improved: replaced the LET scorer** The new version has a new name, `ProtonLET`, to clarify that this calculation technique is only appropriate for protons. It uses a new “subscorer” mechanism by which the energy deposit and step length are internally scored by two separate scorers which are created for you on the fly and then divided to produce the final value.

**Fixed: phi divisions in `TsCylinder` and `TsSphere`** The phi order of the divisions was previously backwards. For cases where the Cylinder or Sphere covered the full two Pi, this was only a numbering issue. But for cases where the Cylinder or Sphere was cut to less than the full two Pi, this caused the phi divisions to extend outside of the mother volume.

**Fixed: excessive generation of parallel worlds** Some jobs were unnecessarily hitting the maximum number of parallel worlds (8). TOPAS was generating parallel worlds where they were not needed for one very specific situation: when a component’s name was a subset of a surface scorer’s target component’s name. So for example, when there was a component named `MyComponent` while a scorer was using `MyComponent/SubComponentName`. (Yes, never mind the details, it’s fixed.)

**Fixed: other minor bugs**

- The phase space might not read all particles in the phase space file.

- Setting AssignToRegionNamed for a component that is unused creates a segfault.
- Using verbose scoring output from unsegmented components creates a segfault.
- Histogram IDs in xml and root files had incorrect run numbers.
- Quadrupole magnetic field incorrectly respected rotation of the component.

### 5.8.2 1.2.2 (2015-06-12)

**Improved: TOPAS now prevents Geant4 from hanging due to stuck particles** We check the particle's `GetCurrentStepNumber` at every step, killing the particle in the very rare case that the step numbers exceeds 100,000 (reasonable particles generally take no more than a few thousand steps). The limit can be adjusted by the parameter `Ts/MaxStepNumber`, but there is unlikely to be any reason that you should do so (stuck particles are rare, and even taking 100,000 steps doesn't take very long). In principle, stuck particles should never occur, as Geant4 has a mechanism in place to gently nudge stuck particles, but this mechanism appears to have occasional failings (on the order a few per ten million histories). Because this mechanism now requires us to include a stepping action for every step in every region, we have checked carefully to evaluate its cost in CPU time. The cost appears to be only about 1%, well worth paying to avoid the workflow disruption that comes from stuck jobs. A message is printed at the bottom of the console if any particles have been killed by this new check. It tells the total number of particles affected and the total kinetic energy involved. More detailed information on each killed particle is provided earlier in the console log.

**Improved: testing of complex parameter file graphs** These are complex arrangements with multiple `includeFile`. This fixes an issue in which our checks to prevent ambiguous parameter file graphs had been rejecting some valid graphs.

**Fixed: correctly handle non-zero timeline starting value** This fixes an issue in which geometry setup failed or reported overlaps at time zero, when time zero was not actually going to be used for any of the runs.

**Fixed: placement within parallel worlds** Components that are children of a parallel world group component are now placed correctly.

**Improved: error messages from sources** Particle sources now have improved checks and more helpful error messages when required parameters are missing.

**Fixed: bug in DVH output** DVH header files now give the correct name for their DVH data file. This fixes a minor issue in which the wrong file extension was shown in the header.

**Improved: specifying ions** Ion handling has been updated to reflect that Geant4 only handles fully stripped ions. The third parameter in `GenericIon(Z, A, Charge)` is now optional. If present, it must equal Z since Geant4 only handles fully stripped ions. Since this argument is redundant, you can also just specify `GenericIon(Z, A)`.

**Added: IncludeCharge option for phase space output** This may be informative for ions that can lose charge as they move through material. Not available for Limited output format.

### 5.8.3 1.2.1 (2015-05-15)

**Fixed: bug in range modulator wheel component** The bug caused a mis-alignment in the position of the modulator wheel blocks. The magnitude of the effect depends on ones exact simulation (and hence was missed in our own SOBP tests). A review of changes shows that we introduced this bug just before we upgraded from our Beta releases to our Release 1.0. Thanks to Benjamin Lutz for identifying this bug.

**Fixed: incorrect RunID output by parameter dump** Triggered by diagnostic parameter dumps from `Ts/DumpParametersToCsvFile`, etc.

**Fixed: remove extraneous debugging messages** Removed diagnostic output that was left over from recent development of the beam source spectrum feature.



### 5.8.4 1.2.0 (2015-04-20)

**Added: energy spectrum parameters to beam source** Beam particle source can sample energy from a user-defined spectrum ([see here](#)).

**Added: particle sources to extension mechanism** [See here](#) for how to add custom particle sources.

**Added: new user hooks to extension mechanism** [See here](#) for how to add new methods: `BeginSession`, `BeginRun`, `BeginHistory`, `EndHistory`, `EndRun`, `EndSession`.

**Improved: column ordering in scorer output can be customized** Whereas previously, the `Report` parameter just indicated which quantities to report (Sum, Mean, etc.), the parameter is now also used to tell the order of these output columns. Thus: `sv:Sc/MyScorer/Report = 2 "Sum" "Mean"` now gives a different column order than `sv:Sc/MyScorer/Report = 2 "Mean" "Sum"`.

**Improved: removed trailing comma in csv output format** The trailing comma on each line caused problems for some Matlab users.

**Added: ability to read back and manipulate scorer output** Scoring Output can be read back in, so that one can then write out with different Report options:

```
Ts/RestoreResultsFromFile = "True" # defaults to "False"
```

will then expect each scorer to have:

```
s:Sc/MyScorer1/InputFile = "MySavedFileName"  
s:Sc/MyScorer1/InputType = "csv" # "csv" or "binary"
```

Reads header of input file to check that input is appropriate for the given scorer. Can be used to output with different Report options than the original file, such as different columns, different column order or creating a DVH. Can also be used to translate from csv to binary or back again.

**Added: option to dump parameter values to csv file** Dumps the requested parameters to a txt file: `TopasParameter-Dump_Run0.txt`. Triggered by:

```
sv:Ts/DumpParametersToCsv = 2 "SomeParameter" "SomeOtherParameter"
```

When multiple runs are involved, makes a new file for each run.

**Fixed: by default, phase spaces do not output empty histories** `PhaseSpaceIncludeEmptyHistories` default value changed to False to match what was already in the User Guide.

**Improved: meaning of EBins = 1 and TimeBins = 1 has changed** Previously, setting these values to 1 meant “do not use binning”. But since even when there is only one bin the data in the underflow and overflow bins can be useful, we now have `EBins = 1` mean create one bin, plus underflow and overflow bins. To have no binning at all, set `EBins = 0` (or don’t set `EBins` at all). A similar change was made for `TimeBins = 1`.

**Fixed: conflict between energy binning and some filters** Energy Binning was not working in presense of `FilterByAtomicMass`, `FilterByAtomicNumber` and the example user-supplied filter, `TsMyFilter1`.

**Fixed: memory bug in energy binning** Previously caused segfault on some systems when `EBins` was used.

**Improved: check vector parameter length is non-negative** Previously protected against this number being zero but not against this being negative.

**Added: example of rotation directions** [See examples/TimeFeature/Rotation.txt](#).

**Fixed: size of parallel scoring component for DICOM and XiO patient** This is the parallel world `TsBox` that is created if you attempt to score a DICOM or XiO Patient in a different grid than the original input file.



**Fixed: visualization of parallel scoring component for DICOM and XiO patient** These parallel scoring components were often invisible. They now show correctly.

### 5.8.5 1.1.0 (2015-02-23)

**Fixed: bug in particle type filters** It corrects behavior of the following four types of filters: `OnlyIncludeParticlesNamed`, `OnlyIncludeParticlesNotNamed`, `OnlyIncludeIfParticleOrAncestorNamed`, `OnlyIncludeIfParticleOrAncestorNotNamed`,

### 5.8.6 1.0.0 (2015-02-11)

First version of TOPAS was released! (Uses Geant4.9.6.p04)

Prior to the release of version 1.0, TOPAS had a long public beta phase.



---

## Introduction to Parameter System

---

### 6.1 Design Philosophy

TOPAS follows a consistent set of design paradigms. Understanding these paradigms will make TOPAS more intuitive to you.

All control is through the *TOPAS Parameters System*. Use of Geant4 macros or interactive commands is not supported as it does not give you the reliability and repeatability that comes from the parameters system.

- TOPAS Parameter files are not Geant4 macro files. TOPAS is specifically designed to avoid the kind of order-dependence risks that Geant4 macro files create.
- TOPAS Parameter files are not XML files. Those too involve the kind of order dependence that we explicitly avoid.

To keep OpenGL graphics from vanishing from the screen, you have the option to have TOPAS pause at the Geant4 command line by including the option:

```
b:Ts/PauseBeforeQuit = "True" # defaults to "False"
```

To exit the Geant4 command line, and continue with the TOPAS session, type `exit`. For simulations that do not involve OpenGL graphics, just leave this option at "False".

All positions are set relative to *Geometry Components*. If you want to place a particle source or a scorer, you place it relative to a particular Component. You may choose to do your placement relative to the center of the World component, in which case you have essentially used the global coordinate system, but you will likely choose a more directly relevant Component. For example, a source that represents a particle beam might be placed at the beamline exit window. Doing so means that the source position will move appropriately with any nozzle movement.

All time dependent behaviors are controlled through the *Time Features* system.

TOPAS fully supports the Multi-Threaded simulation capability of Geant4. By default, TOPAS will occupy just one CPU thread, but you can spread the simulation over multiple threads, distributing the load over your entire computer, by adjusting the parameter `i:Ts/NumberOfThreads`. See details in the section *Multithreading*.

## 6.2 Syntax

The TOPAS Parameter System is a control structure for applications in which a large number of complex inter-related parameters are controllable by designers and end-users, in a manner that is absolutely flexible but simultaneously easy to use. The system is designed with safety and repeatability as top priorities. A key error-checking strategy is strict type checking, in which every parameter must have a specific declared type (string, boolean, integer, etc.) and the provided values are checked to ensure they are appropriate to the given type.

The system takes a set of “Parameters Files,” simple text files made up of lines of key/value pairs:

```
Parameter_Type : Parameter_Name = Parameter_Value # Optional comment
```

When you edit parameter files, be careful to use a Plain Text editor. TOPAS will not understand the various hidden characters created by complex word processors (such as Word or Keynote). Whatever your editor, turn off advanced features such as “Smart quotes”, “Smart dashes” and “Smart links”.

Ten example parameter settings are given below:

```
d:Ge/Phantom/HLX                = 10. cm      # Dimensioned Double
u:Ge/Magnet/Dipole/MagneticFieldDirectionX = 1.0    # Unitless Double
i:Sc/DoseScorer/ZBins            = 100        # Integer
b:Sc/DoseScorer/Active           = "True"     # Boolean
s:Ge/Phantom/Material            = "G4_WATER" # String
dv:Ge/RMW_Track1/Angles          = 4 69.1 92.2 111.0 126.0 deg # Dimensioned_
↪Double Vector
uv:Ma/Phantom_Plastic/Fractions  = 3 0.05549 0.75575 0.18875 # Unitless Double_
↪Vector
iv:Gr/Color/yellow              = 3 225 255 0    # Integer Vector
bv:Tf/ScoringOnOff/Values       = 4 "true" "false" "true" "false" # Boolean Vector
sv:Ma/MyPlastic/Components      = 3 "Hydrogen" "Carbon" "Oxygen" # String Vector
```

---

**Note:** The order of lines within a parameter file does not matter.

---

A Parameter\_Name can be almost any string, but we have prefix conventions to keep things clear:

- Ma/ for Materials
- El/ for Elements
- Is/ for Isotopes
- Ge/ for Geometry Components
- So/ for Particle Sources
- Ph/ for Physics
- Vr/ for Variance Reduction
- Sc/ for Scoring
- Gr/ for Graphics
- Tf/ for Time Features
- Ts/ for TOPAS overall control

The Parameter\_Type tells TOPAS what type of data will be in this parameter:

- d for Dimensioned Double

- `u` for Unitless Double
- `i` for Integer
- `b` for Boolean
- `s` for String
- `dv` for Dimensioned Double Vector
- similarly for `uv`, `iv`, `bv` and `sv`

**Warning:** The only forbidden characters in a parameter name are: `= + - * " ' ` TAB NEWLINE` and `RETURN`. The only forbidden characters in a parameter value are: `= ' ``.

TOPAS uses this `Parameter_Type` to perform “strict type checking,” checking that the `Parameter_Value` is appropriate and complete for the given `Parameter_Type`.

A **String** parameter must be in quotes and may take any value.

A **Boolean** parameter must be in quotes and may be either:

- `"True"`, `"t"` or `"1"` (in any case) to mean true
- `"False"`, `"f"` or `"0"` (in any case) to mean false

An **Integer** parameter must be something that can be interpreted as an integer.

- The value may not contain any decimal part, as this can lead to ambiguity as to the employed rounding strategy.
- These are 32 bit integers, thus the values can range from 0 to 2147483647.

A **Dimensioned Double** parameter requires both a value and a unit.

- We require the unit to avoid misunderstandings.
- The value must be something that can be interpreted as a floating point number.

A **Vector of Dimensioned Doubles** parameter requires an integer (larger than zero) to indicate how many values are expected, then the values themselves, then a unit.

- Vector of Dimensioned Doubles is useful when the definition of a single shape, motion, etc. requires multiple dimensioned double values.
- Our usage of the term “vector” may be unfamiliar to some readers but is the standard term for such structures in modern programming languages.

**Vectors of Unitless, Integer, Boolean and String** again require an integer to indicate how many values are expected, then the values themselves. The individual strings in a Vector of Strings can not contain spaces (this requirement will be relaxed in a subsequent TOPAS release).

The comment character is `#`. Anything to the right of the comment character is taken as a comment. Comments can span as many lines as desired, until a new line is found that contains the equals sign.

**Warning:** A given parameter name may not be defined more than once in a single file.

Blank lines are ignored.

Parameter names may use mixed case, but their interpretation is not case sensitive. That is, `"myParameter"` is considered the same as `"myparameter"` or `"myPaRaMeter"`, etc.

## 6.2.1 Complete Set of Allowed Syntax for any one Parameter Line

**Warning:** Note that in all of the expressions below, there must be a space before and after any +, - or \*.

Dimensioned Double parameters:

```
d:parameterName = number unit
d:parameterName = number unit + name_of_dimensioned_double_parameter
d:parameterName = number unit - name_of_dimensioned_double_parameter
d:parameterName = number unit * name_of_unitless_or_integer_parameter
d:parameterName = number * name_of_dimensioned_double_parameter unit
d:parameterName = name_of_dimensioned_double_parameter unit
d:parameterName = name_of_dimensioned_double_parameter unit * number
d:parameterName = name_of_dimensioned_double_parameter unit * name_of_unitless_or_
↳integer_parameter
d:parameterName = name_of_unitless_or_integer_parameter * number unit
d:parameterName = name_of_dimensioned_double_parameter + number unit
d:parameterName = name_of_dimensioned_double_parameter - number unit
d:parameterName = name_of_dimensioned_double_parameter + name_of_dimensioned_double_
↳parameter unit
d:parameterName = name_of_dimensioned_double_parameter - name_of_dimensioned_double_
↳parameter unit
```

Unitless parameters:

```
u:parameterName = number
u:parameterName = number + name_of_unitless_or_integer_parameter
u:parameterName = number - name_of_unitless_or_integer_parameter
u:parameterName = number * name_of_unitless_or_integer_parameter
u:parameterName = name_of_unitless_or_integer_parameter
u:parameterName = name_of_unitless_or_integer_parameter + number
u:parameterName = name_of_unitless_or_integer_parameter - number
u:parameterName = name_of_unitless_or_integer_parameter * number
u:parameterName = name_of_unitless_or_integer_parameter + name_of_unitless_or_integer_
↳parameter
u:parameterName = name_of_unitless_or_integer_parameter - name_of_unitless_or_integer_
↳parameter
u:parameterName = name_of_unitless_or_integer_parameter * name_of_unitless_or_integer_
↳parameter
```

Integer parameters:

```
i:parameterName = integer
i:parameterName = integer + name_of_integer_parameter
i:parameterName = integer - name_of_integer_parameter
i:parameterName = integer * name_of_integer_parameter
i:parameterName = name_of_integer_parameter
i:parameterName = name_of_integer_parameter + integer
i:parameterName = name_of_integer_parameter - integer
i:parameterName = name_of_integer_parameter * integer
i:parameterName = name_of_integer_parameter + name_of_integer_parameter
i:parameterName = name_of_integer_parameter - name_of_integer_parameter
i:parameterName = name_of_integer_parameter * name_of_integer_parameter
```

Boolean parameters:

```
b:parameterName = value
b:parameterName = name_of_boolean_parameter
b:parameterName = name_of_boolean_parameter * name_of_boolean_parameter
```

String parameters:

```
s:parameterName = string
s:parameterName = string + name_of_integer_or_string_parameter
s:parameterName = name_of_integer_or_string_parameter
s:parameterName = name_of_integer_or_string_parameter + string
s:parameterName = name_of_integer_or_string_parameter + name_of_integer_or_string_
↳parameter
```

Dimensioned Double Vector parameters:

```
dv:parameterName = number_of_values value1 value2 ... valueN unit
dv:parameterName = number_of_values value1 value2 ... valueN unit + name_of_
↳dimensioned_double_or_double_vector_parameter
dv:parameterName = number_of_values value1 value2 ... valueN unit - name_of_
↳dimensioned_double_or_double_vector_parameter
dv:parameterName = number_of_values value1 value2 ... valueN unit * name_of_unitless_
↳or_integer_or_unitless_vector_or_integer_vector
dv:parameterName = number_of_values value1 value2 ... valueN * name_of_dimensioned_
↳double_or_double_vector_parameter unit
dv:parameterName = name_of_dimensioned_double_vector_parameter unit
dv:parameterName = number * name_of_dimensioned_double_vector_parameter unit
dv:parameterName = name_of_unitless_or_integer_parameter * name_of_dimensioned_double_
↳vector_parameter unit
# value1, value2, etc. can be a numeric value or the name of a dimensioned double_
↳parameter.
```

Unitless Vector parameters:

```
uv:parameterName = number_of_values value1 value2 ... valueN
uv:parameterName = number_of_values value1 value2 ... valueN + name_of_unitless_or_
↳integer_or_unitless_vector_or_integer_vector
uv:parameterName = number_of_values value1 value2 ... valueN - name_of_unitless_or_
↳integer_or_unitless_vector_or_integer_vector
uv:parameterName = number_of_values value1 value2 ... valueN * name_of_unitless_or_
↳integer_or_unitless_vector_or_integer_vector
uv:parameterName = name_of_unitless_vector_parameter
uv:parameterName = number * name_of_unitless_vector_parameter
uv:parameterName = name_of_unitless_or_integer_parameter * name_of_unitless_vector_
↳parameter
# value1, value2, etc. can be a numeric value or the name of a unitless parameter
```

Integer Vector parameters:

```
iv:parameterName = number_of_values value1 value2 ... valueN
iv:parameterName = number_of_values value1 value2 ... valueN + name_of_integer_or_
↳integer_vector_parameter
iv:parameterName = number_of_values value1 value2 ... valueN - name_of_integer_or_
↳integer_vector_parameter
iv:parameterName = number_of_values value1 value2 ... valueN * name_of_integer_or_
↳integer_vector_parameter
iv:parameterName = name_of_integer_vector_parameter
iv:parameterName = integer * name_of_integer_vector_parameter
```

(continues on next page)

(continued from previous page)

```
iv:parameterName = name_of_integer_parameter * name_of_integer_vector_parameter
# value1, value2, etc. can be a numeric value or the name of an integer parameter
```

Boolean Vector parameters:

```
bv:parameterName = number_of_values value1 value2 ... valueN
bv:parameterName = name_of_boolean_vector_parameter
# value1, value2, etc. can be a numeric value or the name of a boolean parameter
```

String Vector parameters:

```
sv:parameterName = number_of_values value1 value2 ... valueN
sv:parameterName = number_of_values value1 value2 ... valueN + name_of_integer_or_
↳string_or_integer_vector_or_string_vector
sv:parameterName = name_of_string_vector_parameter
# value1, value2, etc. can be a numeric value or the name of a string parameter
```

Other operations are intentionally not supported since their behavior might be unclear. Such things can be done in user C++ code, generating new parameters on the fly (see [Transient Parameters](#)).  $d * d$  is forbidden because can create new units that we don't recognize. Division is forbidden because of divide by zero issues, etc.

## 6.3 Relative Parameters

TOPAS supports “relative parameters”, wherein one parameter may be set relative to another, as in:

```
s:Ge/Phantom/Material = SomeOtherParameterName
```

**Note:** The many uses of this relative parameter syntax become more clear once one understands the entirety of the TOPAS design, including [hierarchical control](#) files and [time features](#).

With relative dimensioned double parameters, we must protect against a user setting a parameter relative to some other parameter that does not have appropriate units. The solution is to insist that a unit be included on the right side of the expression. In the example below, the unit of `cm` indicates that `SomeOtherParameter` must itself have units of length. If that other parameter's unit is of the entirely wrong unit category (mass, angle, etc.), TOPAS will refuse to run. If the unit is of the right category but a different exact unit (m, mm, etc.), TOPAS will perform appropriate unit conversion:

```
d:Ge/Phantom/HLX = SomeOtherParameterName cm
```

TOPAS has a grammar for operations such as adding or multiplying parameters:

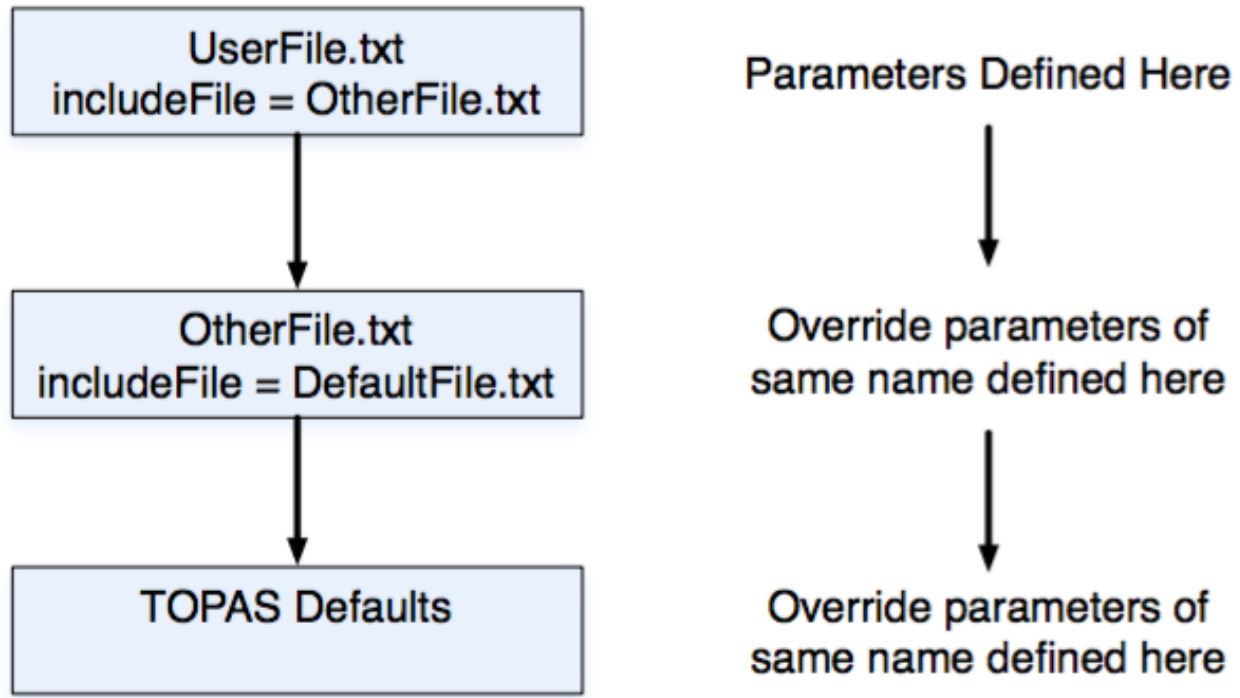
```
Ge/Compensator/TransZ = Ge/Aperture/DistalEdge + Ge/Compensator/HLZ mm
```

**Warning:** Note that there must be a space before and after the plus sign.

Relative parameters allow only a limited number of functions, intentionally not a full math library, since other math functions may be ambiguous, requiring too much prior understanding of the mathematical syntax. The complete set of allowed syntax for any one parameter line is shown [here](#).



## 6.4 Hierarchical Control



Parameter files may pull in other parameters through `includeFile` statements, such as:

```
includeFile = someOtherParameterFile
```

On most operating systems, the case of the file name matters - `MyIncludeFile.txt` is not considered the same file as `MYIncludeFile.txt` - so take care to match the exact case.

`includeFile` lines may appear anywhere in the parameter file.

`includeFile` can use either absolute or relative file positions, such as:

```
includeFile = /Applications/topas/someDirectory/MyIncludeFile.txt
# or
includeFile = ../someDirectory/MyIncludeFile.txt
```

A file inherits all settings from its `includeFile` statements, and can override any of those included settings by setting the same parameter name to a new value.

Type and kind of units has to match type and kind of units for the same parameter name in any `includeFile`.

Type can be omitted if the same parameter name has already been defined with a type in an `includeFile`.

### 6.4.1 Parameter File Chains

When a parameter file includes another parameter file, and this in turn includes another parameter file, we refer to this as a “parameter file chain”.

Parameter file chains fit nicely into research workflow. You can define most of your standard settings in one file, while a file higher on the chain overrides just those values that you want to change today.

To define a parameter in terms of the value of the same parameter in an `includeFile`, set the value to the same parameter name (always interpreted to mean this parameter value from an `includeFile`) or use the shortcut value, `inheritedValue`. The following example would set a foil to be twenty percent thicker than in its included file:

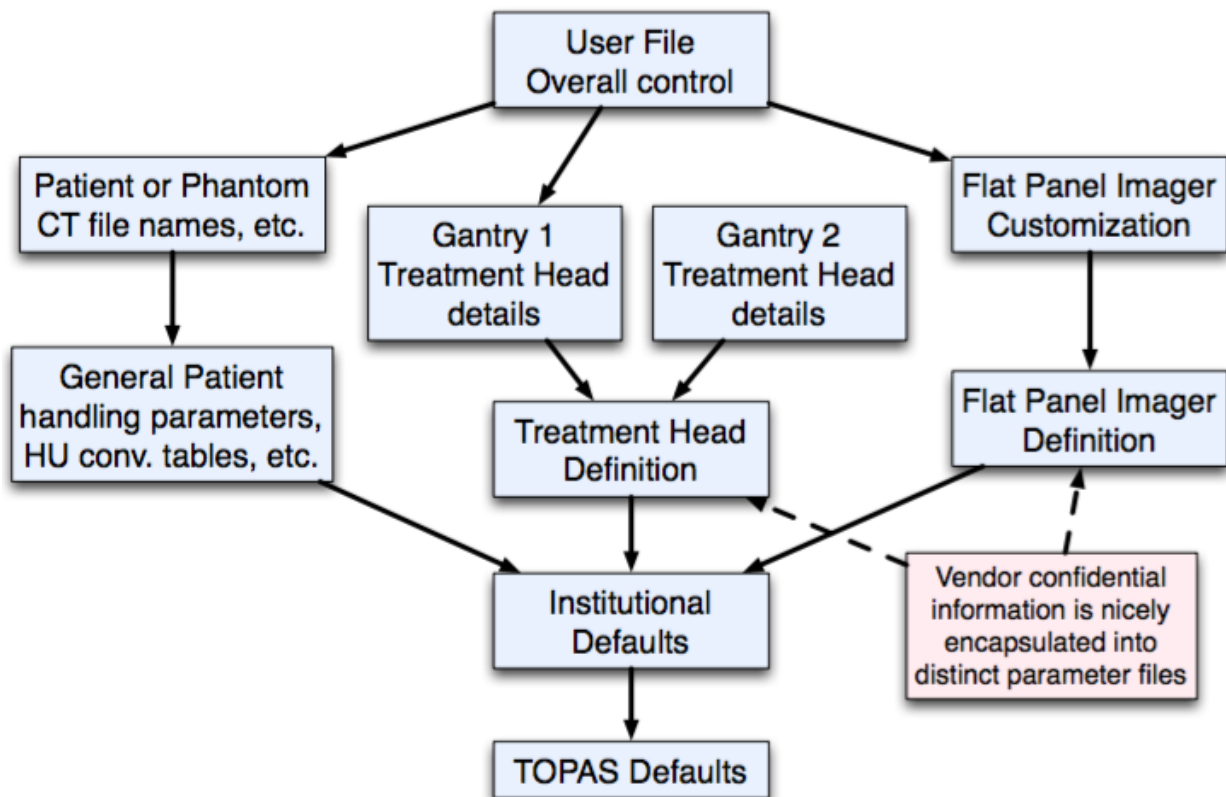
```
Ge/IonChamber/Layer2/Foil/HLZ = Ge/IonChamber/Layer2/Foil/HLZ mm * 1.2
# or
Ge/IonChamber/Layer2/Foil/HLZ = inheritedValue mm * 1.2
```

**Warning:** Note that there must be a space before and after the multiplication sign.

A basic set of TOPAS default parameters are built into the system (see [Default Parameters](#)). You may override these defaults in your own parameter files if you wish.

## 6.4.2 Parameter File Graphs

TOPAS is designed to facilitate multiple independent workgroups focused on separate aspects such as treatment head design, patient handling and imaging devices. To this end, a parameter file may inherit settings from more than one other parameter file, forming a structure that is more complex than just a single chain. We refer to such a structure as a “parameter file graph.” An example is shown below:



To implement such designs, a parameter file allows any number of `includeFile` statements, the statements may be located anywhere in the file, and you may specify one or more include files in a single `includeFile` statement, as in:

```
includeFile = someFile someOtherFile someOtherOtherFile
includeFile = stillAnotherFile
```

There is no significance to the left to right arrangement. That is:

```
includeFile = fileA fileB fileC
```

will behave the same as:

```
includeFile = fileC fileB fileA
```

or even the same as if this was broken up into multiple `includeFile` statements:

```
includeFile = fileB
includeFile = fileC fileA
```

To keep this order from mattering, TOPAS does not allow you to set up a simulation object (a Geometry Component, a Scorer, etc.) in one chain but modify it in a different chain. All parameters on a given Geometry Component need to be handled in the same chain. All parameters on a given Scorer need to be handled in the same chain.

TOPAS also checks to make sure that no two parameter file chains modify the same parameter in a way that is ambiguous. If, for example, the material `MySpecialTungstenAlloy` has been defined in the imaging chain, it cannot also be defined differently in the treatment head chain (unless the top level file, the user file, itself defines this parameter in an absolute way, that is, not relative to any other parameter).

The basic paradigm controlling use of multiple parameter chains is that nothing you do in one chain should magically change the behavior of anything in another chain. Think of the simulation world as an actual physical room. We don't want the behavior of one thing in the room to magically change just because something else entered the room. So when you have multiple chains, TOPAS will check that neither chain modifies anything from the other chain.

This also means that no chain can redefine any of the *Default Parameters*, since all of the Default Parameters effectively belong at the base of every chain. The only place you can redefine the Default Parameters is in the top parameter file. This can feel like a heavy requirement, but it is essential. We want TOPAS to be a great tool for use in collaborative research environments where several people or teams may be contributing their own parts of the simulation setup.

It can take some experience to design complex parameter file chains. If you get stuck, feel free to ask for help on the [TOPAS User Forum](#).

## 6.5 Controlling Multiple Batch Jobs

The hierarchical nature of parameter files makes it easy to control multiple batch jobs.

Make up a parameter file (or hierarchy of files) that has most of your settings:

- `MostOfMySettings.txt`

Then make small additional parameter files for each job you want to submit:

- `Job1.txt`
- `Job2.txt`
- `Job3.txt`

where each of these files has:

```
includeFile = MostOfMySettings.txt
Ts/Seed = 1 # Set this differently for each of Job1, Job2, Job3, ...
Sc/MyScorer/OutputFile = "Job1Output" # Set this differently for each of Job1, Job2, ↵
↵Job3,
```

Each job will thus have a unique starting random number seed (and hence produce a statistically distinct sample) and a unique output file specification, but all other aspects of the simulation will be identical from one job to the next.

---

## Default Parameters

---

The following parameters are built-in by default. They are actually compiled into the code rather than set from a parameter file, so that all users will always have the same starting set of defaults. You can override any of these parameters in your own files.

### 7.1 Overall program control

```
i:Ts/Seed = 1 # starting random seed
i:Ts/MaxStepNumber = 1000000 # limit on number of steps before a track is killed
i:Ts/MaxInterruptedHistories = 10 # limit on how many histories can throw rare Geant4
↳errors
b:Ts/DumpParameters = "False" # Set true to dump full set of parameters to html file
↳TopasParameterDump_Run0.html
b:Ts/DumpNonDefaultParameters = "False" # Like above but omits defaults
b:Ts/ListUnusedParameters = "False" # Set true to list unused parameters on the
↳console
i:Ts/ShowHistoryCountAtInterval = 1 # How often to print history count to the console
b:Ts/ShowHistoryCountLessFrequentlyAsSimulationProgresses = "False" # Counts by 1,
↳then by 10, then by 100, etc.
i:Ts/MaxShowHistoryCountInterval = "2147483647" # Stops increasing count interval
↳after this limit
b:Ts/ShowHistoryCountOnSingleLine = "False" # Set true to make history count reuse
↳same line of console
b:Ts/IncludeTimeInHistoryCount = "False" # Adds time stamp to history count
i:Ts/RunIDPadding = 4 # pad Run ID numbers to this many places in file names
b:Ts/PauseBeforeInit = "False" # Pause for Geant4 commands before initialization
b:Ts/PauseBeforeSequence = "False" # Pause for Geant4 commands before run sequence
b:Ts/PauseBeforeQuit = "False" # Pause for Geant4 commands before quitting
i:Ts/RunVerbosity = 0 # Set to larger integer to see details of run. Maximum is 2
i:Ts/EventVerbosity = 0 # Set to larger integer to see details of event. Maximum is 5
i:Ts/TrackingVerbosity = 0 # Set to larger integer to see details of tracking
i:Ts/SequenceVerbosity = 0 # Set to larger integer to see details of TOPAS run
↳sequence
```

(continues on next page)

(continued from previous page)

```
b:Ts/QuitIfManyHistoriesSeemAnomalous = "True" # Quits if Geant4 warnings issued on
↳too many histories
i:Ts/NumberOfAnomalousHistoriesToAllowInARow = 10000 # Limit for above
b:Ts/RestoreResultsFromFile = "False" # Re-reads previous results to allow new output
↳format or outcome modeling
i:Ts/NumberOfThreads = 1 # Number of CPU threads to which work will be distributed
b:Ts/BufferThreadOutput = "False" # Causes console output to be show one thread at a
↳time
b:Ts/TreatExcitedIonsAsGroundState = "False" # Allows you to read back in excited
↳ions in a phase space file
s:Ts/G4DataDirectory = "" # Specify path to Geant4 Data files (instead of having to
↳set environment variable)
```

## 7.2 Overall timeline control

```
b:Tf/RandomizeTimeDistribution = "False" # Causes each history to be at a different
↳time sampled from timeline
d:Tf/TimelineStart = 0. s
d:Tf/TimelineEnd = Tf/TimelineStart s
i:Tf/NumberOfSequentialTimes = 1
i:Tf/Verbosity = 0 # set to 1 to generate time log, set to 2 to get detailed update
↳messages
```

## 7.3 Optional checks on correctness of geometry

```
b:Ge/CheckForOverlaps = "True"
b:Ge/CheckInsideEnvelopesForOverlaps = "False" # Speeds up checking by assuming inner
↳parts of components are OK
i:Ge/CheckForOverlapsResolution = 1000
d:Ge/CheckForOverlapsTolerance = 0. mm
b:Ge/QuitIfOverlapDetected = "True"
i:Ge/NumberOfPointsPerOverlapCheck = 100
b:Ge/CheckForUnusedComponents = "True"
```

## 7.4 Top level geometry component, the World Volume

```
s:Ge/World/Type = "TsBox"
s:Ge/World/Material = "Air"
d:Ge/World/HLX = 5. m # Half Length
d:Ge/World/HLY = 5. m
d:Ge/World/HLZ = 5. m
d:Ge/World/TransX = 0. m
d:Ge/World/TransY = 0. m
d:Ge/World/TransZ = 0. m
d:Ge/World/RotX = 0. deg
d:Ge/World/RotY = 0. deg
d:Ge/World/RotZ = 0. deg
```

## 7.5 Demo Beam position

```
s:Ge/BeamPosition/Parent = "World"
s:Ge/BeamPosition/Type = "Group"
d:Ge/BeamPosition/TransX = 0. m
d:Ge/BeamPosition/TransY = 0. m
d:Ge/BeamPosition/TransZ = Ge/World/HLZ m
d:Ge/BeamPosition/RotX = 180. deg
d:Ge/BeamPosition/RotY = 0. deg
d:Ge/BeamPosition/RotZ = 0. deg
```

## 7.6 Demo Particle Source

```
s:So/Demo/Type = "Beam" # Beam, Isotropic, Emittance or PhaseSpace
s:So/Demo/Component = "BeamPosition"
s:So/Demo/BeamParticle = "proton"
d:So/Demo/BeamEnergy = 169.23 MeV
u:So/Demo/BeamEnergySpread = 0.757504
s:So/Demo/BeamPositionDistribution = "Gaussian" # Flat or Gaussian
s:So/Demo/BeamPositionCutoffShape = "Ellipse" # Point, Ellipse, Rectangle or Isotropic
d:So/Demo/BeamPositionCutoffX = 10. cm
d:So/Demo/BeamPositionCutoffY = 10. cm
d:So/Demo/BeamPositionSpreadX = 0.65 cm
d:So/Demo/BeamPositionSpreadY = 0.65 cm
s:So/Demo/BeamAngularDistribution = "Gaussian" # Flat or Gaussian
d:So/Demo/BeamAngularCutoffX = 90. deg
d:So/Demo/BeamAngularCutoffY = 90. deg
d:So/Demo/BeamAngularSpreadX = 0.0032 rad
d:So/Demo/BeamAngularSpreadY = 0.0032 rad
i:So/Demo/NumberOfHistoriesInRun = 0
i:So/Demo/NumberOfHistoriesInRandomJob = 0
```

## 7.7 Physics

```
s:Ph/LastName = "Default"
b:Ph/ListProcesses = "False" # Set true to dump list of active physics processes to_
↳console
s:Ph/Default/Type = "Geant4_Modular"
sv:Ph/Default/Modules = 6 "g4em-standard_opt4" "g4h-phy_QGSP_BIC_HP" "g4decay" "g4ion-
↳binarycascade" "g4h-elastic_HP" "g4stopping"
d:Ph/Default/EMRangeMin = 100. eV
d:Ph/Default/EMRangeMax = 500. MeV
```

## 7.8 Scoring

```
b:Sc/AddUnitEvenIfItIsOne = "False" # If unit is 1, rather than, say, Gy, default is_
↳to leave out unit in header.
s:Sc/RootFileName = "topas" # name for root output files
s:Sc/XMLFileName = "topas" # name for xml output files
```

## 7.9 Graphics

```
b:Gr/Enable = "True" # Set False to avoid instantiating any part of Geant4
↳visualization system (useful for running on batch machines that lack the OpenGL
↳graphics library)
i:Gr/Verbosity = 0 # Set to higher integer to increase verbosity of Geant4
↳visualization system
s:Gr/RefreshEvery = "Run" # "History", "Run" or "Session"
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 8000 # Above this limit, only show outer box
i:Gr/SwitchOGLtoOGLIifVoxelCountExceeds = 70000000 # Above this limit, switch OpenGL
↳Graphics to Immediate mode
```

## 7.10 Elements

```
s:El/Hydrogen/Symbol = "H"
s:El/Helium/Symbol = "He"
s:El/Lithium/Symbol = "Li"
s:El/Beryllium/Symbol = "Be"
s:El/Boron/Symbol = "B"
s:El/Carbon/Symbol = "C"
s:El/Nitrogen/Symbol = "N"
s:El/Oxygen/Symbol = "O"
s:El/Fluorine/Symbol = "F"
s:El/Neon/Symbol = "Ne"
s:El/Sodium/Symbol = "Na"
s:El/Magnesium/Symbol = "Mg"
s:El/Aluminum/Symbol = "Al"
s:El/Silicon/Symbol = "Si"
s:El/Phosphorus/Symbol = "P"
s:El/Sulfur/Symbol = "S"
s:El/Chlorine/Symbol = "Cl"
s:El/Argon/Symbol = "Ar"
s:El/Potassium/Symbol = "K"
s:El/Calcium/Symbol = "Ca"
s:El/Scandium/Symbol = "Sc"
s:El/Titanium/Symbol = "Ti"
s:El/Vanadium/Symbol = "V"
s:El/Chromium/Symbol = "Cr"
s:El/Manganese/Symbol = "Mn"
s:El/Iron/Symbol = "Fe"
s:El/Cobalt/Symbol = "Co"
s:El/Nickel/Symbol = "Ni"
s:El/Copper/Symbol = "Cu"
s:El/Zinc/Symbol = "Zn"
s:El/Gallium/Symbol = "Ga"
s:El/Germanium/Symbol = "Ge"
s:El/Arsenic/Symbol = "As"
s:El/Selenium/Symbol = "Se"
s:El/Bromine/Symbol = "Br"
s:El/Krypton/Symbol = "Kr"
s:El/Rubidium/Symbol = "Rb"
s:El/Strontium/Symbol = "Sr"
s:El/Yttrium/Symbol = "Y"
s:El/Zirconium/Symbol = "Zr"
```

(continues on next page)



(continued from previous page)

```

s:El/Niobium/Symbol = "Nb"
s:El/Molybdenum/Symbol = "Mo"
s:El/Technetium/Symbol = "Tc"
s:El/Ruthenium/Symbol = "Ru"
s:El/Rhodium/Symbol = "Rh"
s:El/Palladium/Symbol = "Pd"
s:El/Silver/Symbol = "Ag"
s:El/Cadmium/Symbol = "Cd"
s:El/Indium/Symbol = "In"
s:El/Tin/Symbol = "Sn"
s:El/Antimony/Symbol = "Sb"
s:El/Tellurium/Symbol = "Te"
s:El/Iodine/Symbol = "I"
s:El/Xenon/Symbol = "Xe"
s:El/Caesium/Symbol = "Cs"
s:El/Barium/Symbol = "Ba"
s:El/Lanthanum/Symbol = "La"
s:El/Cerium/Symbol = "Ce"
s:El/Praseodymium/Symbol = "Pr"
s:El/Neodymium/Symbol = "Nd"
s:El/Promethium/Symbol = "Pm"
s:El/Samarium/Symbol = "Sm"
s:El/Europium/Symbol = "Eu"
s:El/Gadolinium/Symbol = "Gd"
s:El/Terbium/Symbol = "Tb"
s:El/Dysprosium/Symbol = "Dy"
s:El/Holmium/Symbol = "Ho"
s:El/Erbium/Symbol = "Er"
s:El/Thulium/Symbol = "Tm"
s:El/Ytterbium/Symbol = "Yb"
s:El/Lutetium/Symbol = "Lu"
s:El/Hafnium/Symbol = "Hf"
s:El/Tantalum/Symbol = "Ta"
s:El/Tungsten/Symbol = "W"
s:El/Rhenium/Symbol = "Re"
s:El/Osmium/Symbol = "Os"
s:El/Iridium/Symbol = "Ir"
s:El/Platinum/Symbol = "Pt"
s:El/Gold/Symbol = "Au"
s:El/Mercury/Symbol = "Hg"
s:El/Thallium/Symbol = "Tl"
s:El/Lead/Symbol = "Pb"
s:El/Bismuth/Symbol = "Bi"
s:El/Polonium/Symbol = "Po"
s:El/Astatine/Symbol = "At"
s:El/Radon/Symbol = "Rn"
s:El/Francium/Symbol = "Fr"
s:El/Radium/Symbol = "Ra"

```

## 7.11 Materials

```

s:Ma/DefaultColor = "white"
i:Ma/Verbosity = 0 # Set to 1 to report each time a material is defined

```

(continues on next page)

(continued from previous page)

```
sv:Ma/Vacuum/Components = 4 "Carbon" "Nitrogen" "Oxygen" "Argon"
uv:Ma/Vacuum/Fractions = 4 0.000124 0.755268 0.231781 0.012827
d:Ma/Vacuum/Density = 1.0E-25 g/cm3
s:Ma/Vacuum/State = "Gas"
d:Ma/Vacuum/Temperature = 2.73 kelvin
d:Ma/Vacuum/Pressure = 3.0E-18 pascal
s:Ma/Vacuum/DefaultColor = "skyblue"
```

```
sv:Ma/Carbon/Components = 1 "Carbon"
uv:Ma/Carbon/Fractions = 1 1.0
d:Ma/Carbon/Density = 1.867 g/cm3
d:Ma/Carbon/MeanExcitationEnergy = 78 eV
s:Ma/Carbon/DefaultColor = "green"
```

```
sv:Ma/Aluminum/Components = 1 "Aluminum"
uv:Ma/Aluminum/Fractions = 1 1.0
d:Ma/Aluminum/Density = 2.6989 g/cm3
s:Ma/Aluminum/DefaultColor = "skyblue"
i:Ma/Aluminum/AtomicNumber = 13
d:Ma/Aluminum/AtomicMass = 26.98154 g/mole
```

```
sv:Ma/Nickel/Components = 1 "Nickel"
uv:Ma/Nickel/Fractions = 1 1.0
d:Ma/Nickel/Density = 8.902 g/cm3
s:Ma/Nickel/DefaultColor = "indigo"
```

```
sv:Ma/Copper/Components = 1 "Copper"
uv:Ma/Copper/Fractions = 1 1.0
d:Ma/Copper/Density = 8.96 g/cm3
s:Ma/Copper/DefaultColor = "orange"
```

```
sv:Ma/Iron/Components = 1 "Iron"
uv:Ma/Iron/Fractions = 1 1.0
d:Ma/Iron/Density = 7.87 g/cm3
s:Ma/Iron/DefaultColor = "skyblue"
```

```
sv:Ma/Tantalum/Components = 1 "Tantalum"
uv:Ma/Tantalum/Fractions = 1 1.0
d:Ma/Tantalum/Density = 16.654 g/cm3
s:Ma/Tantalum/DefaultColor = "indigo"
```

```
sv:Ma/Lead/Components = 1 "Lead"
uv:Ma/Lead/Fractions = 1 1.0
d:Ma/Lead/Density = 11.35 g/cm3
i:Ma/Lead/AtomicNumber = 82
d:Ma/Lead/AtomicMass = 207.19 g/mole
d:Ma/Lead/MeanExcitationEnergy = 823 eV
s:Ma/Lead/DefaultColor = "brown"
```

```
sv:Ma/Air/Components = 4 "Carbon" "Nitrogen" "Oxygen" "Argon"
uv:Ma/Air/Fractions = 4 0.000124 0.755268 0.231781 0.012827
d:Ma/Air/Density = 1.20484 mg/cm3
d:Ma/Air/MeanExcitationEnergy = 85.7 eV
s:Ma/Air/DefaultColor = "lightblue"
```

```
sv:Ma/Brass/Components = 2 "Copper" "Zinc"
uv:Ma/Brass/Fractions = 2 0.7 0.3
```

(continues on next page)

(continued from previous page)

```

d:Ma/Brass/Density = 8.550 g/cm3
d:Ma/Brass/MeanExcitationEnergy = 324.4 eV
s:Ma/Brass/DefaultColor = "grass"

sv:Ma/Lexan/Components = 3 "Hydrogen" "Carbon" "Oxygen"
uv:Ma/Lexan/Fractions = 3 0.055491 0.755751 0.188758
d:Ma/Lexan/Density = 1.2 g/cm3
d:Ma/Lexan/MeanExcitationEnergy = 73.1 eV
s:Ma/Lexan/DefaultColor = "grey"

sv:Ma/Lucite/Components = 3 "Hydrogen" "Carbon" "Oxygen"
uv:Ma/Lucite/Fractions = 3 0.080538 0.599848 0.319614
d:Ma/Lucite/Density = 1.190 g/cm3
d:Ma/Lucite/MeanExcitationEnergy = 74.0 eV
s:Ma/Lucite/DefaultColor = "grey"

sv:Ma/Mylar/Components = 3 "Hydrogen" "Carbon" "Oxygen"
uv:Ma/Mylar/Fractions = 3 0.041959 0.625017 0.333025
d:Ma/Mylar/Density = 1.40 g/cm3
s:Ma/Mylar/DefaultColor = "red"

sv:Ma/Mylon/Components = 4 "Hydrogen" "Carbon" "Nitrogen" "Oxygen"
uv:Ma/Mylon/Fractions = 4 0.097976 0.636856 0.123779 0.141389
d:Ma/Mylon/Density = 1.140 g/cm3
s:Ma/Mylon/DefaultColor = "purple"

sv:Ma/Kapton/Components = 4 "Hydrogen" "Carbon" "Nitrogen" "Oxygen"
uv:Ma/Kapton/Fractions = 4 0.026362 0.691133 0.073270 0.209235
d:Ma/Kapton/Density = 1.420 g/cm3
s:Ma/Kapton/DefaultColor = "purple"

sv:Ma/Water_75eV/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/Water_75eV/Fractions = 2 0.111894 0.888106
d:Ma/Water_75eV/Density = 1.0 g/cm3
d:Ma/Water_75eV/MeanExcitationEnergy = 75.0 eV
s:Ma/Water_75eV/DefaultColor = "blue"

sv:Ma/Titanium/Components = 1 "Titanium"
uv:Ma/Titanium/Fractions = 1 1.0
d:Ma/Titanium/Density = 4.54 g/cm3
s:Ma/Titanium/DefaultColor = "blue"

sv:Ma/Steel/Components = 8 "Carbon" "Silicon" "Phosphorus" "Sulfur" "Chromium"
↪ "Manganese" "Iron" "Nickel"
uv:Ma/Steel/Fractions = 8 0.0015 0.01 0.00045 0.0003 0.19 0.02 0.67775 0.1
d:Ma/Steel/Density = 8.027 g/cm3
s:Ma/Steel/DefaultColor = "lightblue"

```

## 7.12 Colors

```

iv:Gr/Color/White =      3 255 255 255
iv:Gr/Color/Silver =    3 191 191 191
iv:Gr/Color/Gray =      3 127 127 127
iv:Gr/Color/Grey =      3 127 127 127

```

(continues on next page)

(continued from previous page)

```
iv:Gr/Color/Black = 3 0 0 0
iv:Gr/Color/Red = 3 255 0 0
iv:Gr/Color/Maroon = 3 127 0 0
iv:Gr/Color/Yellow = 3 255 255 0
iv:Gr/Color/Olive = 3 127 127 0
iv:Gr/Color/Lime = 3 0 255 0
iv:Gr/Color/Green = 3 0 127 0
iv:Gr/Color/Aqua = 3 0 255 255
iv:Gr/Color/Teal = 3 0 127 127
iv:Gr/Color/Blue = 3 0 0 255
iv:Gr/Color/Navy = 3 0 0 127
iv:Gr/Color/Fuchsia = 3 255 0 255
iv:Gr/Color/Purple = 3 127 0 127

iv:Gr/Color/Lightblue = 3 175 255 255
iv:Gr/Color/Skyblue = 3 175 124 255
iv:Gr/Color/Magenta = 3 255 0 255
iv:Gr/Color/Violet = 3 224 0 255
iv:Gr/Color/Pink = 3 255 0 222
iv:Gr/Color/Indigo = 3 0 0 190
iv:Gr/Color/Grass = 3 0 239 0
iv:Gr/Color/Orange = 3 241 224 0
iv:Gr/Color/Brown = 3 225 126 66

iv:Gr/Color/grey020 = 3 20 20 20
iv:Gr/Color/grey040 = 3 40 40 40
iv:Gr/Color/grey060 = 3 60 60 60
iv:Gr/Color/grey080 = 3 80 80 80
iv:Gr/Color/grey100 = 3 100 100 100
iv:Gr/Color/grey120 = 3 120 120 120
iv:Gr/Color/grey140 = 3 140 140 140
iv:Gr/Color/grey160 = 3 160 160 160
iv:Gr/Color/grey180 = 3 180 180 180
iv:Gr/Color/grey200 = 3 200 200 200
iv:Gr/Color/grey220 = 3 220 220 220
iv:Gr/Color/grey240 = 3 240 240 240
```

## 8.1 Time mode

If you do nothing special, TOPAS will do a single run with no time variation. We call this “Fixed Time Mode”. Other available modes are “Sequential” and “Random”.

### 8.1.1 Fixed Time Mode

To run in Fixed Time Mode, just set your source’s `NumberOfHistoriesInRun`, as in:

```
i:So/MySource/NumberOfHistoriesInRun = 100
```

If your parameter files include *Time Features*, they will all be evaluated with time equals zero. To instead have them evaluated at a different fixed time, specify `TimelineStart`, as in:

```
d:Tf/TimelineStart = 10. s # defaults to zero
```

If you have more than one source, the run will continue until all sources have run all of their histories. For each Geant4 “beamOn”, each source will get called, but only those that have more histories left to produce will actually produce any.

### 8.1.2 Sequential Time Mode

To have TOPAS do several runs at fixed time intervals, specify the start time, end time and number of sequential times, as in:

```
d:Tf/TimelineStart = 0. s # defaults to zero
d:Tf/TimelineEnd = 10. s # must be larger than TimelineStart
i:Tf/NumberOfSequentialTimes = 100 # defaults to 1
```

TOPAS will divide the overall time, `TimelineEnd - TimelineStart`, by `NumberOfSequentialTimes` and perform runs at each of these intervals.

- The first run will be at time = TimelineStart.
- The last run will be at time = TimelineEnd minus one interval. That is, TOPAS will stop *before* it reaches TimelineEnd.

So, in the example above:

- Run 0 will have Time = 0. s
- Run 1 will have Time = 0.1 s
- ...
- Run 99 will have Time = 9.9 s

At each of these intervals, your source will generate your indicated NumberOfHistoriesInRun:

```
i:So/MySource/NumberOfHistoriesInRun = 10
```

So, for example, if you have 100 intervals, and NumberOfHistoriesInRun = 10, you will generate a total of  $100 \times 10 = 1000$  histories.

To have TOPAS print time feature information to a log file and to the console:

```
i:Tf/Verbosity = 2 # defaults to zero.  
# set to 1 to get time log (NbParticlesInTime.txt)  
# set to 2 to get detailed update messages
```

To implement **beam current modulation**, have your source's NumberOfHistoriesInRun get its value from a time feature, as in:

```
i:So/MySource/NumberOfHistoriesInRun = Tf/MyBCMTimeFeature/Value
```

By default, scorers will output just once, after the entire session. But if you wish to have separate output from specific runs:

```
b:Sc/MyScorer/OutputAfterRun = "True" # set True to trigger output of scorer after_  
↳this run
```

- If this is False, or not defined, we just output at the end of the simulation.
- If this is True, we output after every run.

### 8.1.3 Random Time Mode

Random Time Mode generates one history per run, with a randomly sampled time at each run. This has several uses.

- It allows one to sample time in a continuous fashion, so may show features that are obscured by Sequential Mode
- It provides a way to do a lower statistics run of what would have been a very long Sequential Mode job, yet still see aspects of the entire time interval, rather than just the first subset of the sequential times

To run in Random Time Mode, specify the TimelineStart and TimelineEnd, turn on RandomizeTimeDistribution, and set your source's NumberOfHistoriesInRandomJob, as in:

```
b:Tf/RandomizeTimeDistribution = "True" # defaults to "False"  
d:Tf/TimelineStart = 0. s # defaults to zero  
d:Tf/TimelineEnd = 10. s # must be larger than TimelineStart  
i:So/MySource/NumberOfHistoriesInRandomJob = 1000 # defaults to 100
```

For each history, a random time will be sampled between `TimelineStart` and `TimelineEnd`.

We keep the parameters that control random mode (`NumberOfHistoriesInRandomJob`) separate from those that control sequential mode (`NumberOfHistoriesInRun` and `NumberOfSequentialTimes`) so that you can easily switch between the two modes (by just switching `RandomizeTimeDistribution`).

To implement **beam current modulation**, give your source a time-dependent `ProbabilityOfUsingAGivenRandomTime`, as in:

```
d:So/MySource/ProbabilityOfUsingAGivenRandomTime = Tf/MyBCMTimeFeature/Value
```

### 8.1.4 Fixed Time but with Very Large Number of Histories

The maximum number of histories possible per run is limited by the size of some of Geant4's internal counters. If you need more than  $10^9$  histories at a fixed time, you can work around this limitation by breaking your session into multiple runs:

- Set `Tf/NumberOfSequentialTimes` to some value greater than 1
- No need to actually set `TimelineStart` or `TimelineEnd` (they both default to 0)

Your total number of histories will then be `NumberOfSequentialTimes * NumberOfHistoriesInRun`.

## 8.2 Multithreading

TOPAS fully supports the Multi-Threaded simulation capability of Geant4.

By default, TOPAS will occupy just one CPU thread. To use more, adjust:

```
i:Ts/NumberOfThreads = 4 # defaults to 1
```

- If set to a positive integer, TOPAS will use that number of threads
- If set to 0, TOPAS will use all of your computer's threads. This may be number of hardware cores or number of virtual cores (which includes hyper-threading cores) depending on your hardware architecture.
- If set to a negative number, TOPAS will use all BUT this number of threads, leaving you some threads reserved for other tasks (email, web browsing, etc.).

By default, console output from various threads will be interleaved. Output from each worker thread will have a distinctive prefix, such as:

```
G4WT0 >
G4WT1 >
```

To instead make Geant4 buffer the output, showing first everything from one thread and then everything from another thread, set:

```
b:Ts/BufferThreadOutput = "True" # Causes console output to be show one thread at a_
↪time
```

## 8.3 Random Number Seed

To set the random seed:

```
i:Ts/Seed = 1 # default is 1
```

To generate several statistically independent runs, give each run a different `Ts/Seed`. A typical solution to produce 10 independent runs would be to give starting seeds of 1 to 10. The allowed range is 0 to 2147483647 (the maximum 32 bit integer).

For more details see the discussion in the [Geant4 Application Developer's Guide](#).

We use the random engine called `RanecuEngine` and the seed given to TOPAS is passed to the engine through `CLHEP::HepRandom::setTheSeed`.

## 8.4 Miscellaneous

### 8.4.1 Interactive Geant4 Sessions

To have TOPAS pause and wait for interactive Geant4 commands:

```
b:Ts/PauseBeforeInit = "True"
b:Ts/PauseBeforeSequence = "True"
b:Ts/PauseBeforeQuit = "True"
```

After each pause, type the Geant4 command `exit` to return control to TOPAS.

- Most users will only use `PauseBeforeQuit`, typically to make a graphics window stay open at the end of the session (graphics windows close when Geant4 quits).
- The other two options, `PauseBeforeInit` and `PauseBeforeSequence`, provide the ability to enter Geant4 commands by hand, which may be useful in certain tests, but invalidates the basic TOPAS concept that the behavior of your simulation should be perfectly defined by TOPAS parameters.

### 8.4.2 Dump Parameter Values

Additional overall control parameters are:

```
b:Ts/DumpParameters = "True" # dump full list of params to file TopasParameterDump_
    ↳Run0.html
b:Ts/DumpNonDefaultParameters = "False" # Like above but omits defaults
sv:Ts/DumpParametersToSimpleFile = 2 "SomeParameter" "SomeOtherParameter" # Dumps the_
    ↳requested parameter types, names and values to a simple, human-readable file,_
    ↳TopasParameterDump_Run0.txt
sv:Ts/DumpParametersToSemicolonSeparatedFile = 2 "SomeParameter" "SomeOtherParameter"
    ↳# Dumps the requested parameter types, names and values to a semicolon separated_
    ↳file, TopasParameterDumpSSF_Run0.txt. This file is suitable for easy import into_
    ↳other applications
```

### 8.4.3 Verbosity

Additional overall control parameters are:

```
i:Ts/ShowHistoryCountAtInterval = 1 # how often to print history count to the console
# If set to 0, history count will never be printed
```

(continues on next page)



(continued from previous page)

```
b:Ts/ShowHistoryCountOnSingleLine = "False" # Make count reuse a single line of
→console
i:Ts/TrackingVerbosity = 0 # Set to larger integer to see details of tracking
```

You can add time stamps to the history count:

```
b:Ts/IncludeTimeInHistoryCount = "True"
```

You can have a “power-based” history count:

```
b:Ts/ShowHistoryCountLessFrequentlyAsSimulationProgresses
```

After first ten histories, output will change to once for every 10, then to once for every 100, etc.

An additional optional parameter:

```
i:Ts/MaxShowHistoryCountInterval
```

puts an upper limit on how high the ShowHistoryCountInterval can be.

For example:

```
b:Ts/ShowHistoryCountLessFrequentlyAsSimulationProgresses i:Ts/MaxShowHistoryCountInterval =
100
```

Gives:

```
1 2 3 ... 9 10 20 30 ... 100
```

but from there always keep counting by 100 (rather than going on to counting by 1000, 10,000, etc.)

## 8.4.4 Other

Additional overall control parameters are:

```
b:Ts/ShowCPUTime = "True" # Show CPU time used in various phases of the simulation
i:Ts/RunIDPadding = 4 # Run numbers are padded in output files, such as
→MyScoringOutput_Run_0001.csv, so that they will sort naturally in various file
→viewers. This parameter sets how many places of padding are used.
```

## 8.4.5 Quick Ways to Deactivate Parts of the Parameters Files

For most parameter categories, there is one key kind of parameter that triggers creation:

| Parameter object type    | Triggering pattern  |
|--------------------------|---------------------|
| Element                  | El/.../Symbol =     |
| Material                 | Ma/.../Components = |
| Component                | Ge/.../Parent =     |
| Particle Source          | So/.../Type =       |
| Physics setup            | Ph/.../Type =       |
| Scorer                   | Sc/.../Quantity =   |
| Graphic View             | Gr/.../Type =       |
| Variance Reduction setup | Vr/.../Type =       |
| Time Feature             | Tf/.../Function =   |

Thus you could effectively comment out the entire Component, Element, Material, Particle Source, etc. by just commenting out that line. But this way of turning something off can get you into trouble since you may then inherit behavior from a parent parameter file.

A better way to handle this is by setting a specific parameter designed for this purpose:

```
Ge/MyComponent/Include = "False"  
So/MySource/NumberOfHistoriesInRun = 0  
Sc/MyScorer/Active = "False"  
Gr/MyGraphics/Active = "False"
```

Such a parameter can then even be controlled by a time feature.

We have *pre-defined a few materials*. You are free to define additional materials, as in:

```
sv:Ma/Air/Components=4 "Carbon" "Nitrogen" "Oxygen" "Argon" # names of elements
uv:Ma/Air/Fractions=4 0.000124 0.755268 0.231781 0.012827 # fractions of elements
d:Ma/Air/Density=1.2048 mg/cm3
d:Ma/Air/MeanExcitationEnergy=85.7 eV
s:Ma/Air/DefaultColor="lightblue"
```

All *Elements have been pre-defined* with natural isotope abundance from the NIST database. You will only need to create your own Elements if you need something other than natural isotope abundance. For that, see *Elements and Isotopes* below.

Fractions are by weight.

MeanExcitationEnergy is the I parameter in the Bethe equation, which not only includes ionization, but also inner-atomic excitations, etc.

In the *Default Parameters* section, we show the complete list of pre-defined materials. This basically covers those materials that are used in our included examples.

You can also create a new material from a combination of other materials. The key is that you provide a additional parameter called BuildFromMaterials and set this to True. Then instead of Components being element names, it will expect material names.

Here's an example of making a material that is a combination of water and air:

```
b:Ma/MyMixture/BuildFromMaterials = "True"
sv:Ma/MyMixture/Components = 2 "G4_WATER" "Air"
uv:Ma/MyMixture/Fractions = 2 .5 .5
d:Ma/MyMixture/Density = .5 g/cm3
```

You may also use any of the Materials and Compounds that are defined by default in Geant4. The names start with the prefix, G4\_, such as: G4\_Al, G4\_Ti, G4\_MUSCLE\_SKELETAL\_ICRP, etc. The complete list of these materials and compounds can be found [here](#).

- NIST material names must be specified with exact case.

- As of this writing, the mean excitation energy listed in the above reference for G4\_WATER is incorrect. It lists G4\_WATER mean excitation energy as 75.0 eV but it is actually set to 78.0 eV.

**Note:** The Geant4-DNA physics processes have special behavior for G4\_WATER. They take into account the material's molecular properties rather than just the atomic properties. Accordingly, you should use G4\_WATER rather than defining your own Water, unless you have some other reason to make a specific change (such as changing the mean excitation energy to something other than 78.0 eV).

It is up to you to define any additional materials that you want in your own parameter files. If you make your own material, make sure to pick a new material name (the string after the `Ma/`) and make sure that any other parameter file that uses this material includes the file where you defined this material (either directly or through *Parameter File Chains*). The usual rules of *Parameter File Graphs* govern parameter conflicts.

**Warning:** Do not use the prefix `G4_` for the materials that you add. This prefix is reserved for materials and compounds from the pre-defined NIST database.

Where a pre-defined material definition exists, it is generally better to use that definition rather than making your own material. The pre-defined material may provide extra benefit by triggering specific corrections to ionization models.

If you have a set of materials that differ only in density, you can define them all at once (this is commonly needed for *imaging to material conversion*):

```
i:Ma/MyMaterial/VariableDensityBins = 100
u:Ma/MyMaterial/VariableDensityMin = .1
u:Ma/MyMaterial/VariableDensityMax = 10.
```

will generate 100 versions of MyMaterial, with densities varying from .1 x normal to 10. x normal. The material names will then be like:

```
MyMaterial_VariableDensityBin_0
MyMaterial_VariableDensityBin_1
...
MyMaterial_VariableDensityBin_99
```

## 9.1 Elements and Isotopes

All *Elements have been pre-defined* with natural isotope abundance from the NIST database. You will only need to create your own Elements if you need something other than natural Isotope abundance. You can define additional elements as follows:

Define each isotope that you will use, specifying Z, N and A:

```
i:Is/U235/Z = 92
i:Is/U235/N = 235
d:Is/U235/A = 235.01 g/mole
i:Is/U238/Z = 92
i:Is/U238/N = 238
d:Is/U238/A = 238.03 g/mole
```

Define your element with your desired proportion of these isotopes:

```
s:El/MyElU/Symbol = "MyElU"  
sv:El/MyElU/IsotopeNames = 2 "U235" "U238"  
uv:El/MyElU/IsotopeAbundances = 2 90. 10.
```

See *Isotope.txt* example.



# CHAPTER 10

---

## Geometry Components

---

### 10.1 Introduction

All Geometry Components must have at least the following parameters:

```
s:Ge/MyComponent/Parent = "World"
s:Ge/MyComponent/Type = "TsBox"
d:Ge/MyComponent/TransX=0.0 cm # defaults to 0
d:Ge/MyComponent/TransY=0.0 cm # defaults to 0
d:Ge/MyComponent/TransZ=0.0 cm # defaults to 0
d:Ge/MyComponent/RotX=0.0 deg # defaults to 0
d:Ge/MyComponent/RotY=0.0 deg # defaults to 0
d:Ge/MyComponent/RotZ=0.0 deg # defaults to 0
```

The Parent, Trans and Rot parameters place a component within its “mother” as described in *Placement of Components*.

Each Type has its own set of additional required parameters, discussed elsewhere for each specific component type.

The World can be either a TsBox, TsSphere or TsCylinder.

The component name can include the forward slash character /, and this is used in many examples to give some hints about component hierarchy, such as:

```
s:Ge/VBox2/Dipole/Parent = "Nozzle"
```

This bit of hierarchy in the component name, such as VBox2/Dipole, does NOT actually control how the components are assembled. The actual control is from the Parent parameter (discussed [here](#)). The forward slash is just another character here. You could just as well use VBox2\_Dipole or VBox2Dipole, as long as you use the same exact string whenever you refer to this component.

Components that are in the real world (as opposed to *Parallel Worlds*) must also have a material:

```
s:Ge/MyComponent/Material = "Air"
```

To deactivate a Component (and all its children), you can either comment out the parameter that sets its `Parent`, or set its `Include` parameter to false, as in:

```
b:Ge/MyComponent/Include = "False" # defaults to "True"
```

While it is not forbidden to have unused components (components that are never assigned a `Parent`), this can often be a sign that you have not correctly assigned the parents in your geometry. Accordingly, we check for unused components on startup and given a warning message if any are found. You can disable this warning message by setting:

```
Ge/CheckForUnusedComponents = "False"
```

In some cases you may want to keep unused components around. This can be like keeping extra pieces of unused laboratory equipment handy on a shelf. They will have no effect on your simulation, but remain available to quickly plug in when needed by assigning a parent and setting placement parameters.

Physics control for a specific component is done as part of the `Ge/` parameters for that component rather than in the `Ph/` parameters, such as:

```
d:Ge/MyComponent/MaxStepSize = 1. mm # sets maximum step size used in this component
```

## 10.2 Placement of Components

A component's `Parent` parameter tells which other component the current one is a child of. In this way, one can build a hierarchy of components:

```
s:Ge/MyComponent/Parent = SomeOtherComponent
```

The one component that is always provided automatically for you, into which you plug the rest of your hierarchy, is called `World`.

Each component has three translation and three rotation parameters. These give the position of the component in the coordinate system of its parent component.

The following defines a box of air with half width of 5 m on each side placed at the center of the world:

```
s:Ge/MyBox/Type="TsBox"
s:Ge/MyBox/Parent = "World"
s:Ge/MyBox/Material="Air"
d:Ge/MyBox/HLX=5. m # Half Length
d:Ge/MyBox/HLY=5. m
d:Ge/MyBox/HLZ=5. m
d:Ge/MyBox/TransX=0. m
d:Ge/MyBox/TransY=0. m
d:Ge/MyBox/TransZ=0. m
d:Ge/MyBox/RotX=0. deg
d:Ge/MyBox/RotY=0. deg
d:Ge/MyBox/RotZ=0. deg
```

If you set more than one rotation, note that rotation happens first around X, then the Y rotation is applied around the now-rotated axes, and then the Z rotation is applied around those rotated axes. In general, a way to keep rotations more clear is to use intermediate Group components as follows:

- Place your component inside a Group component.
- Rotate your component around one axis.



- Rotate the group component around the other axis.

While the direction of rotation can be confusing, the correctness of rotations in TOPAS has been double, triple and quadruple checked and found consistent with the definitions in Geant4. The [Rotation.txt](#) example shows an object being rotated first in the positive X direction, then in the positive Y direction, then in the positive Z direction.

For Geant4 experts, be advised that the rotation angles you provide to TOPAS are passed into `G4RotationMatrix() -> rotateX/Y/Z`. Further discussion of Geant4 rotations can be found [here](#).

The following overrides the size definition of the `World` box that was inherited from the *built-in default parameters* and then inserts a box into this world and another box into the first box:

```
# Overrides the world size that was set in built-in defaults:
Ge/World/HLX=10. m
Ge/World/HLY=10. m
Ge/World/HLZ=10. m

# Box inserted into the World
s:Ge/TextBox/Material="Air"
s:Ge/TextBox/Parent="World"
s:Ge/TextBox/Type="TsBox"
d:Ge/TextBox/HLX=400. cm
d:Ge/TextBox/HLY=300. cm
d:Ge/TextBox/HLZ=200. cm
d:Ge/TextBox/TransX=0. m
d:Ge/TextBox/TransY=0. cm
d:Ge/TextBox/TransZ=0. m
d:Ge/TextBox/RotX=0. deg
d:Ge/TextBox/RotY=0. deg
d:Ge/TextBox/RotZ=0. deg

# Another box inserted into the first box
s:Ge/TextBox2/Material="Carbon"
s:Ge/TextBox2/Parent="TextBox"
s:Ge/TextBox2/Type="TsBox"
d:Ge/TextBox2/HLX=180. cm
d:Ge/TextBox2/HLY=120. cm
d:Ge/TextBox2/HLZ=80. cm
d:Ge/TextBox2/TransX=0. m
d:Ge/TextBox2/TransY=0. cm
d:Ge/TextBox2/TransZ=150. cm
d:Ge/TextBox2/RotX=0. deg
d:Ge/TextBox2/RotY=30. deg
d:Ge/TextBox2/RotZ=0. deg
```

### 10.2.1 Overlap Checking

Because accidental overlaps of geometry volumes are a serious issue for all Monte Carlo simulations, Geant4 provides tools to automatically check for such overlaps. Overlap checking is not perfect, it works by testing a random set of points on each boundary, to see if they are inside any other boundary. Thus it will not necessarily find all overlaps. By default TOPAS checks 100 points on each volume. Overlap checking has a speed cost at initialization time, so once you are confident that your geometry has no overlaps, you may choose to turn this feature off (though most users never find this necessary):

```
Ge/CheckForOverlaps = "False"
```

TOPAS will save time by avoiding overlap checking for individual parts within a divided component (such as the

voxels within a patient) since these subdivisions are generated automatically by our own code. But if you ever want to turn these back on:

```
b:Ge/CheckInsideEnvelopesForOverlaps = "True"
```

You can control the number of points used in the overlap check:

```
i:Ge/CheckForOverlapsResolution = 1000
```

And you can check the tolerance for overlap:

```
d:Ge/CheckForOverlapsTolerance = 0. mm
```

You can also set these in a more granular fashion, per Component (overrides the above parameters for this particular component):

```
i:Ge/MyComponent/CheckForOverlapsResolution = 1000  
d:Ge/MyComponent/CheckForOverlapsTolerance = 0. mm
```

## 10.2.2 Pre-Defining Values

It may be useful to pre-define a range of named-values, such that you can easily access the values later. For example, we pre-define the angles at which certain scatterers are stored on a scatterer selection wheel:

```
d:Ge/Gantry1/Scatterer2/RotZForSS0 = 0. deg  
d:Ge/Gantry1/Scatterer2/RotZForSS8 = 270. deg  
d:Ge/Gantry1/Scatterer2/RotZForSS2 = 180. deg  
d:Ge/Gantry1/Scatterer2/RotZForSS3 = 90. deg
```

And then in our user file, the user doesn't have to know these actual angles, but can just rotate to one of the named scatterers:

```
Ge/Scatterer2/Holder/RotZ = Ge/Gantry1/Scatterer2/RotZForSS3 deg
```

## 10.3 Parallel Worlds

Components can be assigned to “parallel worlds” rather than the standard, mass world. Such components have no effect on physics (other than usually very minor step limitation effects) but can still be used for scoring. Such components can arbitrarily overlay the mass world. Their volumes can overlap any other volumes in other mass or parallel worlds. To assign a component to a parallel world, include the line:

```
b:Ge/MyComponent/IsParallel = "True"
```

Due to limitations in the Geant4-DNA chemistry design, parallel worlds may not be used if Geant4-DNA chemistry is included in your physics list. TOPAS will issue an error message in this situation.

- Parallel world components may be children of other parallel world components.
- Parallel world components may be children of mass world components.
- Mass world components may not be children of parallel world components.

A new parallel world will be created each time you specify `IsParallel`, with an automatically generated parallel world name based on the component name. If you want to explicitly assign multiple components to the same parallel world, provide the additional parameter:

```
s:Ge/MyComponent/ParallelWorldName = "SomeParallelWorldName"
```

There is no limit on the total number of parallel worlds, but each additional world can cause some performance penalty.

**Warning:** In certain cases, TOPAS must represent a geometry by using a Geant4 technique called “parameterized volumes.” However we have found that Geant4 behaves unreliably if parameterized sphere is placed in a parallel world. Accordingly, TOPAS applies a safety restriction:

- TsSphere can not be in a parallel world if it has any divisions.

### 10.3.1 Layered Mass Geometry

Components that are in a parallel world can have material or not. If they have material, and they are listed in the `LayeredMassGeometryWorlds` parameter, this material will take precedence over any real world material found in that location.

In Geant4 this is called Layered Mass Geometry. It is further described in ([PubMed](#)):

Enger S et al, “Layered mass geometry: a novel technique to overlay seeds and applicators onto patient geometry in Geant4 brachytherapy simulations,” *Phys Med Biol.* 2012 Oct 7;57(19):6269-77.

Any time a component in a parallel world has material, that world must be listed in the `LayeredMassGeometryWorlds` parameter. The parameter is a string vector because any number of parallel worlds can have material. The order of the worlds in this parameter is significant. Material from worlds listed later in this list take precedence over material in worlds listed earlier. Thus, in the following example, material in the world `Seed` will take precedence over material in the world `Tumor` which will take precedence over material in the regular world:

```
sv:Ph/Default/LayeredMassGeometryWorlds = 2 "Tumor" "Seed"
```

A simple example is provided in *LayeredMassGeometry.txt*.

## 10.4 Electromagnetic Fields

You can assign an electric, magnetic or combined electromagnetic field to any geometry component (with exception of Group components, which have no intrinsic extent). The field will extend into any child components unless they themselves have their own field.

To assign a field, add the parameter `Field`, as in:

```
s:Ge/MyComponent/Field = "DipoleMagnet" # "DipoleMagnet", "QuadrupoleMagnet",  
↪ "MappedMagnet", "UniformElectroMagnetic" or your own definition
```

For "DipoleMagnet", specify dipole field and strength, as in (see *DipoleMagnet.txt*):

```
u:Ge/MyComponent/MagneticFieldDirectionX = 0.0  
u:Ge/MyComponent/MagneticFieldDirectionY = 1.0  
u:Ge/MyComponent/MagneticFieldDirectionZ = 0.0  
d:Ge/MyComponent/MagneticFieldStrength = 3.0 tesla
```

For "QuadrupoleMagnet", specify the two components of the gradient, as in (see *QuadrupoleMagnet.txt*):

```
d:Ge/MyComponent/MagneticFieldGradientX = 1.0 tesla
d:Ge/MyComponent/MagneticFieldGradientY = 1.0 tesla
```

For "MappedMagnet ", specify a field map in the Opera 3D format, as in (see [PurgingMagnet\\_move.txt](#)):

```
s:Ge/MyComponent/MagneticField3DTable = "PurgMag3D.TABLE"
```

For "UniformElectroMagnetic", specify electric field and dipole magnetic field, as in (see [UniformElectro-MagneticField.txt](#)):

```
u:Ge/MyComponent/ElectricFieldDirectionX = 1.0
u:Ge/MyComponent/ElectricFieldDirectionY = 1.0
u:Ge/MyComponent/ElectricFieldDirectionZ = 0.0
d:Ge/MyComponent/ElectricFieldStrength = 5000 kV/cm
u:Ge/MyComponent/MagneticFieldDirectionX = 0.0
u:Ge/MyComponent/MagneticFieldDirectionY = 1.0
u:Ge/MyComponent/MagneticFieldDirectionZ = 0.0
d:Ge/MyComponent/MagneticFieldStrength = 5.0 tesla
```

If you have any other value in Field, TOPAS will look in your extensions to find your own class that defines this field. See [Custom Fields](#) for details on writing extension fields.

Field orientation is set by rotating the component.

You can visualize magnetic fields, with field intensity and direction depicted through a set of arrows:

```
i:Gr/ViewA/MagneticFieldArrowDensity = 10
```

Use with caution. When combined with rotation seems to sometimes cause crashes in polycone drawing (involved in drawing the arrowheads).

As with almost any TOPAS parameter, the Electric Field Strength, Dipole Magnet Strength, Quadrupole Magnet Gradient or Mapped Magnetic Field file can be set to change over time by using [Time Features](#) such as (see [QuadAnd-DipoleMagnets.txt](#)):

```
d:Ge/MyComponent/MagneticFieldStrength = Tf/BField1st/Value tesla
```

Fine control of the stepping algorithm can be done by changing the following parameters from their default values:

```
s:Ge/MyComponent/FieldStepper = "ClassicalRK4"
d:Ge/MyComponent/FieldStepMinimum = 1.0 mm
d:Ge/MyComponent/FieldDeltaChord = 1.0e-1 mm
```

See the Geant4 Application Developers Guide on the [Geant4 Documentation Page](#) for detailed discussion of these options.

Stepper choices for purely magnetic fields are:

- "ExplicitEuler"
- "ImplicitEuler"
- "SimpleRunge"
- "SimpleHeum"
- "HelixExplicitEuler"
- "HelixImplicitEuler"
- "HelixSimpleRunge"

- “CashKarpRK45”
- “RKG3”
- “ClassicalRK4”

Steeper choices for electromagnetic fields are:

- “ExplicitEuler”
- “ImplicitEuler”
- “SimpleRunge”
- “SimpleHeum”
- “ClassicalRK4”

## 10.5 Visualization Attributes

By default, Components are colored according to their materials, based on the default color that was assigned to the material, such as:

```
s:Ma/Air/DefaultColor="lightblue"
```

Parameters of the Component let you override this color and set other visualization attributes:

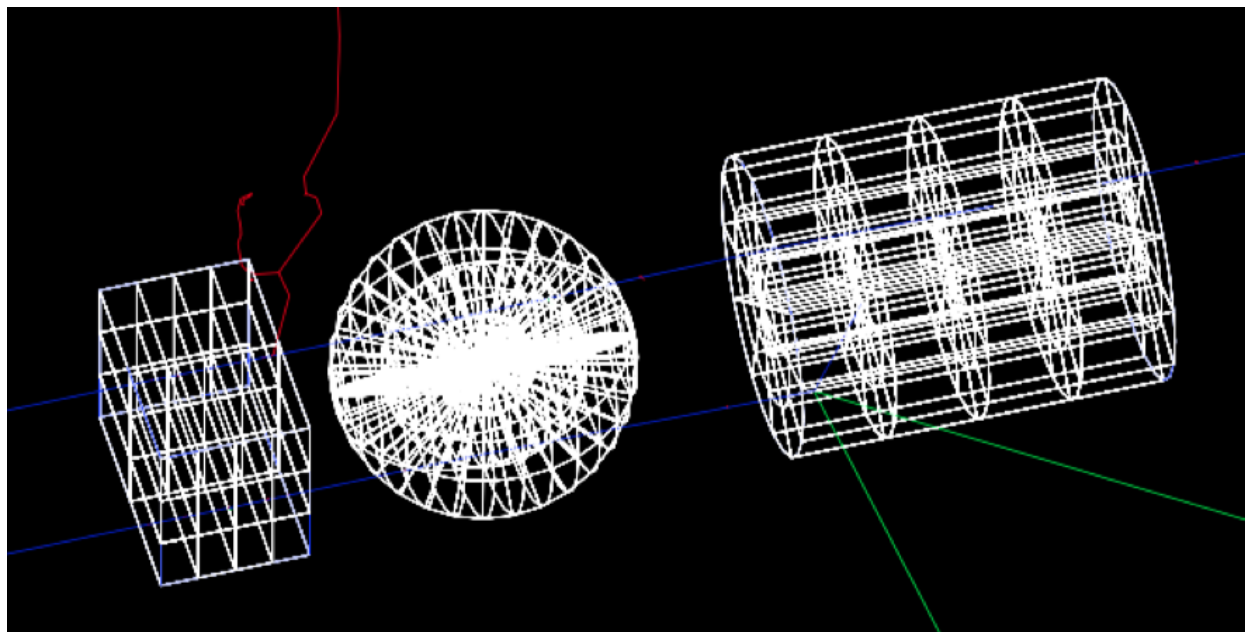
```
s:Ge/MyComponent/Color = "red"
s:Ge/MyComponent/DrawingStyle = "Solid" # "Solid", "Wireframe" or "FullWireFrame"
i:Ge/MyComponent/VisSegsPerCircle = 100 # Number of line segments to use to
→approximate a circle, defaults to 24
b:Ge/MyComponent/Invisible = "True" # defaults to False meaning visible
```

FullWireFrame includes drawing of additional edge lines that Geant4 calls “soft edges” - on many graphics devices WireFrame and FullWireFrame give the same result.

Increase VisSegsPerCircle if you want a smoother curve.

## 10.6 Dividable Components

| Type       | Parameters                    | Type        | Default value |
|------------|-------------------------------|-------------|---------------|
| TsBox      | XBins<br>YBins<br>ZBins       | i<br>i<br>i | 1<br>1<br>1   |
| TsCylinder | RBins<br>PhiBins<br>ZBins     | i<br>i<br>i | 1<br>1<br>1   |
| TsSphere   | RBins<br>PhiBins<br>ThetaBins | i<br>i<br>i | 1<br>1<br>1   |



Scorers associated with the divisible components may use the same or different divisions (thus one can do things like represent the patient with CT resolution but score with other resolutions). See [here](#) for details.

You cannot place child components inside a divided component, but if the only reason for dividing this component is to have fine-grained scoring, you can easily work around this limitation. Use an undivided parent component. Place the children into this undivided parent component. Then when you specify that you want to score on this parent component, specify divided scoring (see [here](#)). TOPAS will automatically create a parallel world version of your component to handle the divided scoring.

You can optionally specify different materials for each voxel, overriding the value set in the regular `Ge/.../Material` parameter:

```
sv:Ge/Phantom/VoxelMaterials = 100 "G4_WATER" "G4_WATER" "Air" "Air" "G4_WATER" ...
```

This means you can create complex phantoms directly from the parameter system. `VoxelMaterials` works for all three kinds of divided components: `TsBox`, `TsCylinder` and `TsSphere`. See the [DoseInVoxelMaterials.txt](#) example.

## 10.7 Generic Components

You can create a Geometry Component for any of the standard solids defined in the geometry section of the [Geant4 Application Developers Guide](#).

The [ShapeTestWithAllParameters.txt](#) example demonstrates how to build each of the solids.



Below we list the parameters for each Geant4 solid. Further details about the parameters, along with helpful diagrams, can be found in the [Geant4 Application Developers Guide](#). For most solids, sizes are specified in Half Lengths, denoted with an HL, such as HLX. For a few solids, sizes are specified in full Lengths, denoted with just L, such as LX.

---

**Note:** The `TsBox`, `TsCylinder` and `TsSphere` should be used instead of `G4Box`, `G4Tubs` and `G4Sphere` respectively. See [Dividable Components](#).

---

Some examples of components that can be built just from Generic Components:

- Scatterer
- Collimator
- Mirror
- Water Tank
- Rando Phantom (as constructive solid geometry rather than DICOM import)
- Pin Diode Chamber
- Flat Panel Imaging Device
- Standard Ion Chamber
- Segmented Ion Chamber
- Faraday Cup

We have built some complex things just from combinations of the Generic Components (such as the STAR radiosurgery beamline at MGH).

| Type      | Parameters  | Type                                      | Default value                            |
|-----------|---|---|--|
| G4CutTubs | RMin<br>RMax<br>HL<br>SPhi<br>DPhi<br>LowNorm<br>HighNorm | d<br>d<br>d<br>d<br>d<br>uv (3)<br>uv (3) | 0 cm<br><br>0 deg<br>360 deg             |
| G4Cons    | RMin1<br>RMax1<br>RMin2<br>RMax2<br>HL<br>SPhi<br>DPhi    | d<br>d<br>d<br>d<br>d<br>d<br>d           | 0 cm<br><br>0 cm<br><br>0 deg<br>360 deg |
| G4Para    | HLX<br>HLY<br>HLZ<br>Alpha<br>Theta<br>Phi                | d<br>d<br>d<br>d<br>d<br>d                |  |
| G4Trd     | HLX1<br>HLX2<br>HLY1<br>HLY2<br>HLZ                       | d<br>d<br>d<br>d<br>d                     |  |
| G4RTrap   | LZ<br>LY<br>LX<br>LTX                                     | d<br>d<br>d<br>d                          |  |
| G4GTrap   | HLZ<br>Theta<br>Phi<br>HLY1<br>HLX1<br>HLX2               | d<br>d<br>d<br>d<br>d<br>d                |  |
| 78        | Alp1<br>HLY2<br>HLX3<br>HLX4                              | d<br>d<br>d<br>d                          | <b>Chapter 10. Geometry Components</b>   |



## 10.8 Group Component

Creates no actual solid, but still has a placement (Trans and Rot). Other components placed within this Group component are affected by this placement just as if the group were an enclosing box component.

The following defines a group component called MyGroup:

```
s:Ge/MyGroup/Type="Group"
s:Ge/MyGroup/Parent = "World"
d:Ge/MyGroup/TransX=2. m
d:Ge/MyGroup/TransY=2. m
d:Ge/MyGroup/TransZ=0. m
d:Ge/MyGroup/RotX=0. deg
d:Ge/MyGroup/RotY=0. deg
d:Ge/MyGroup/RotZ=30. deg
```

The following example shows how a Group Component, Jaws, placed in a nozzle, allows one to position two individual movable collimator blocks, Jaw\_Upper and Jaw\_Lower, without the creation of an extraneous mother volume:

```
s:Ge/Jaws/Type = "Group"
s:Ge/Jaws/Parent = "Nozzle"
d:Ge/Jaws/TransZ = 0. m
...
s:Ge/Jaw_Upper/Type = "TsBox"
s:Ge/Jaw_Upper/Parent = "Jaws"
s:Ge/Jaw_Upper/Material = "Tungsten"
d:Ge/Aperture/TransY = 2. cm
...
s:Ge/Jaw_Lower/Type = "TsBox"
s:Ge/Jaw_Lower/Parent = "Jaws"
s:Ge/Jaw_Lower/Material = "Tungsten"
d:Ge/Aperture/TransY = -2. cm
...
```

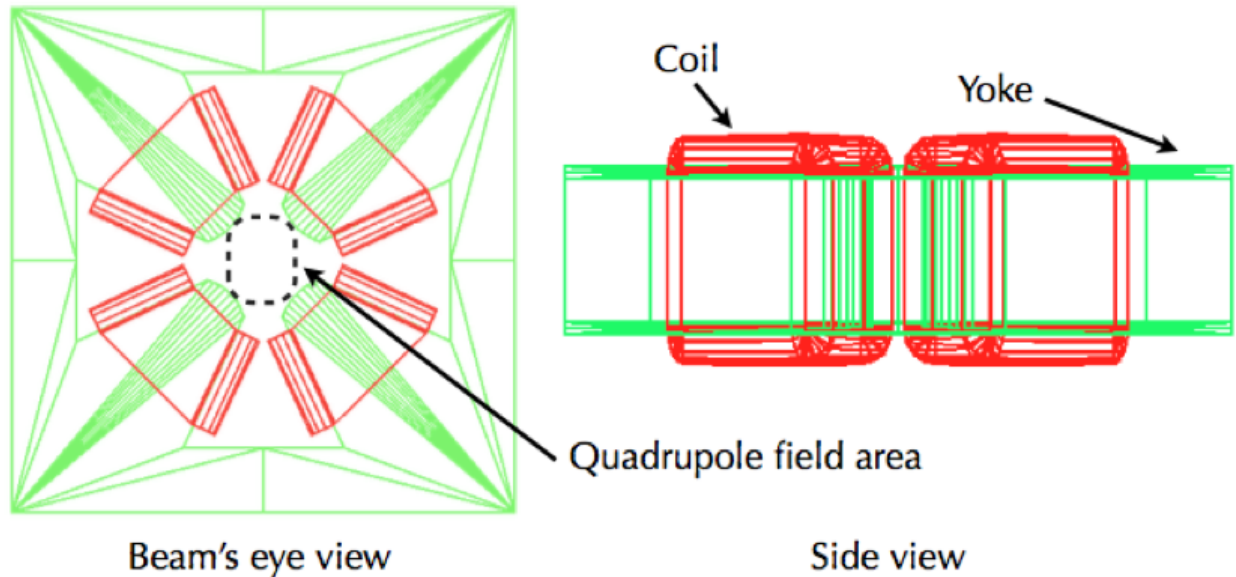
## 10.9 Specialized Components

| Geometry Component                 | Type                  |
|------------------------------------|-----------------------|
| <i>Range Modulator Wheel</i>       | TsRangeModulator      |
| <i>Propeller</i>                   | TsPropeller           |
| <i>Ridge Filter</i>                | TsRidgeFilter         |
| <i>Multi Wire Chamber</i>          | TsMultiWireChamber    |
| <i>Multi Leaf Collimator</i>       | TsMultiLeafCollimator |
| <i>CAD (Computer Aided Design)</i> | TsCAD                 |
| <i>Aperture</i>                    | TsAperture            |
| <i>Compensator</i>                 | TsCompensator         |
| <i>BrachyApplicator</i>            | TsBrachyApplicator    |

Each of the specialized components has its own set of special parameters. Usage is best learned by studying the relevant examples parameter files included in TOPAS.

You may write your own additional components (see *Custom Geometry Components*).

The following figure from Samsung Medical Center shows how their very specific quadrupole magnet system was coded as a TOPAS geometry.



### 10.9.1 Range Modulator Wheel

TOPAS Range modulator is designed to accommodate various specifications from a vendor. We suggest modeling your Range Modulator Wheel (RMW) by the following procedure:

- Define the dimension of RMW drum, such as thickness and material of shell and hub (see figure below). Tracks will be placed in between the hub and the shell.
- This space (in between hub and shell) is vertically divided into three sections named, “Upper”, “Middle”, and “Bottom” so that each section can have its own tracks. You can adjust heights of these sections. The sum of these heights is the total height of your RMW.
- In order to reserve spaces for tracks, divide radially each section into as many as tracks you want by using the parameter, `RadialDivision`
- Using vector parameters, configure the tracks individually such as each block’s height, span angle, and material. Then assign vector parameter to the parameter, called `Pattern`

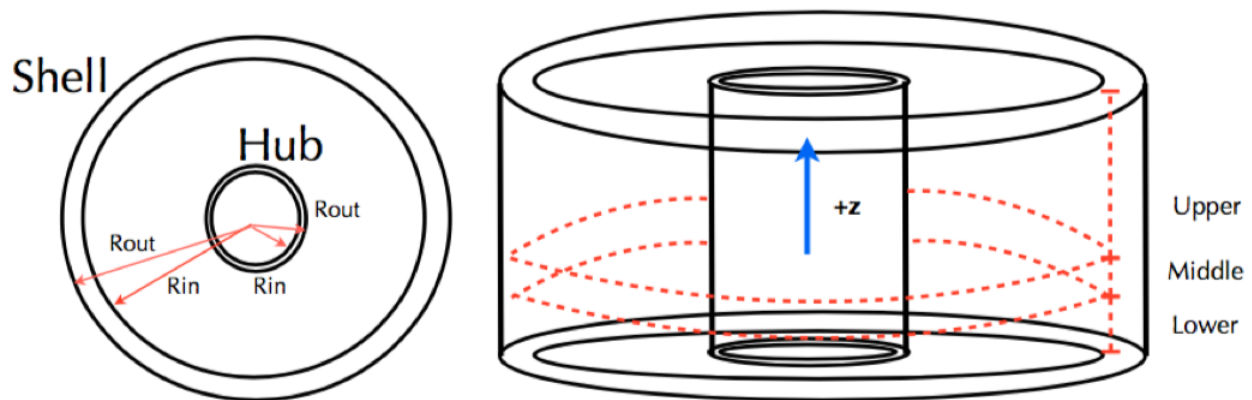
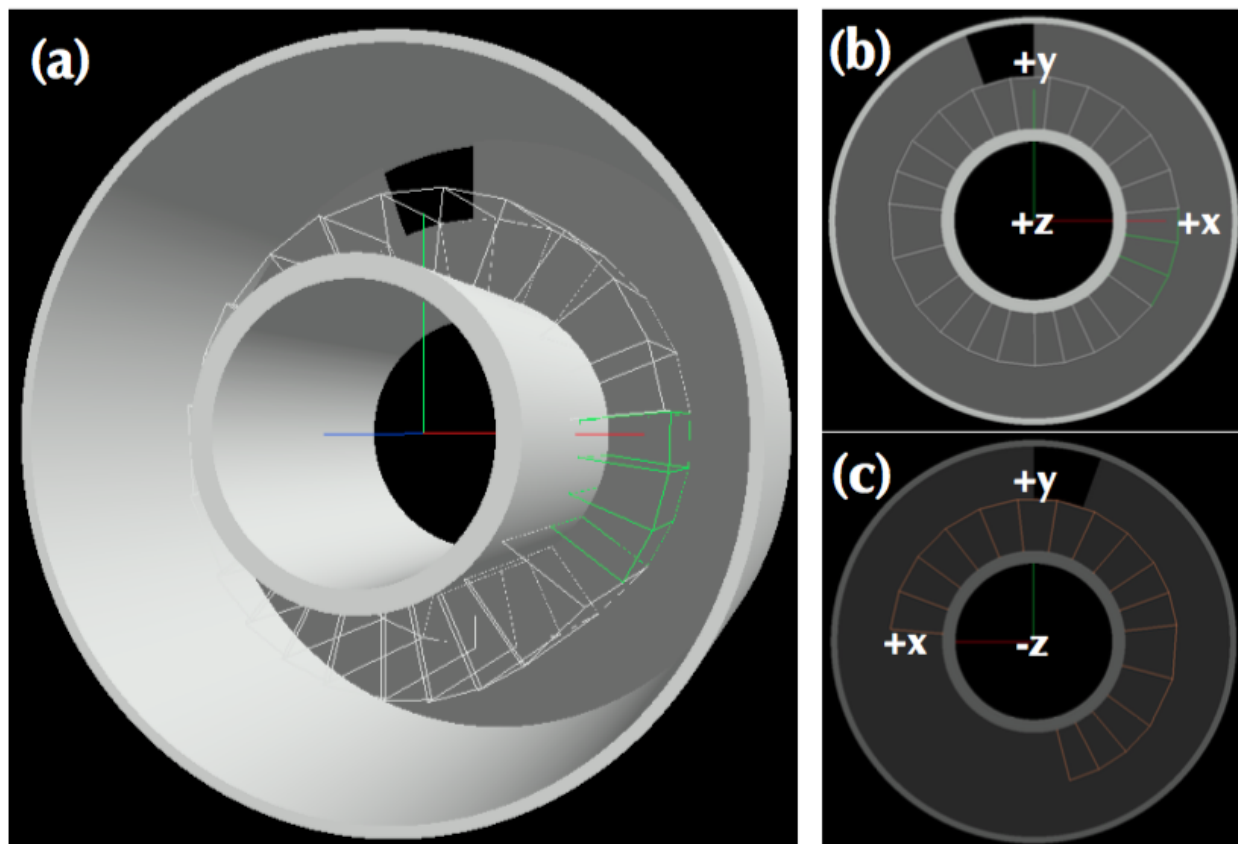


Illustration for TOPAS RMW dimensions. Tracks are placed in between Rout of Hub and Rin of Shell and this area

is to be radially divided in case of placing multiple tracks. There are three vertical rooms, so it is possible to make double sided RMWs with an interface disk.



An example of RMW; (a) Perspective view. Upper section is divided into two but only inner radial division has a track pattern. In middle section, two track patterns are used to make a hole. (b) X-Y view from +z of RMW and (c) X-Y view from -z of RMW. Tracks are drawn in wireframe style, so more lines on the tracks are shown than the number of blocks.

Here is the complete set of the parameters for the above RMW (see [RangeModulator.txt](#) example):

```
# Common parameters: type of geometry, position, and rotation
s:Ge/RangeModulatorA/Type = "TsRangeModulator"
s:Ge/RangeModulatorA/Material = "Parent"
s:Ge/RangeModulatorA/Parent = "World"
d:Ge/RangeModulatorA/TransX = 10.0 cm
d:Ge/RangeModulatorA/TransY = 0.0 cm
d:Ge/RangeModulatorA/TransZ = 0.0 cm
d:Ge/RangeModulatorA/RotX = 0.0 deg
d:Ge/RangeModulatorA/RotY = 0.0 deg
d:Ge/RangeModulatorA/RotZ = 0.0 deg
b:Ge/RangeModulatorA/Invisible = "TRUE"

# Set height of each sections and total height = 160.0 mm
d:Ge/RangeModulatorA/HeightOfUpper = 150 mm
d:Ge/RangeModulatorA/HeightOfMiddle = 1.0 mm
d:Ge/RangeModulatorA/HeightOfLower = 9.0 mm

# Shell dimensions, material, color, etc.
```

(continues on next page)

(continued from previous page)

```

d:Ge/RangeModulatorA/Shell/Rin = 15.0 cm
d:Ge/RangeModulatorA/Shell/Rout = 15.5 cm
s:Ge/RangeModulatorA/Shell/Material = "Aluminum"
s:Ge/RangeModulatorA/Shell/Color = "grey"
s:Ge/RangeModulatorA/Shell/DrawingStyle = "Solid"
i:Ge/RangeModulatorA/Shell/VisSegsPerCircle = 360

# Hub dimensions, material, color, etc.
d:Ge/RangeModulatorA/Hub/Rin = 6.0 cm
d:Ge/RangeModulatorA/Hub/Rout = 7.0 cm
s:Ge/RangeModulatorA/Hub/Material = "Aluminum"
s:Ge/RangeModulatorA/Hub/Color = "grey"
s:Ge/RangeModulatorA/Hub/DrawingStyle = "Solid"
i:Ge/RangeModulatorA/Hub/VisSegsPerCircle = 360

# Setting tracks on Upper area
# Two tracks Track1 (7.0 cm ~ 11.0 cm) and Track2 ( 11.0 cm ~ 15.0 cm)
dv:Ge/RangeModulatorA/Upper/RadialDivisions=1 11.0 cm

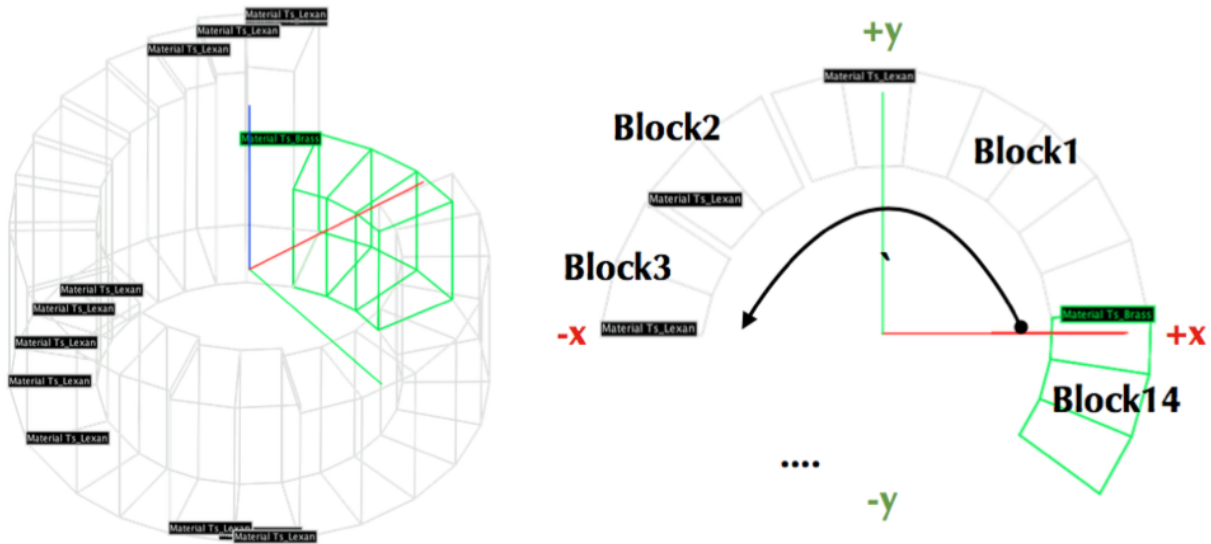
# Assignment of track pattern of Upper area
# Track1 refers the pattern named "LexanBlockT1" whose vector parameters are defined_
→elsewhere (see below).
s:Ge/RangeModulatorA/Upper/Track1/Pattern = "LexanBlockT1"
s:Ge/RangeModulatorA/Upper/Track2/Pattern = "NULL" #NULL means empty track.

# Track1 pattern: 14 blocks of Lexan.
# Numbers of Angles, Heights, and Materials should be same.
d:Ge/LexanBlockT1/Offset=0.0 deg #means shift of zero-angle
# Angle divisions. The first block's spans from 5.0 deg to 115.0 deg.
# The last block starting at 324.0 deg spans to the first block's boundary.
# This case last block spans from 324.0 deg to 360.0 + 5.0 deg
dv:Ge/LexanBlockT1/Angles=14
5.00 115.00 146.50 173.2 195.07
216.15 230.14 243.00 255.5 270.60
282.20 294.60 306.20 324.00 deg
# Height of each block.
# Note that zero height means that no block in that angle range.
dv:Ge/LexanBlockT1/Heights=14
77.0 82.0 87.0 92.15 95.0
100.4 106.0 110.2 115.3 119.5
124.0 128.8 132.00 60.0 mm

# Material of each block.
sv:Ge/LexanBlockT1/Materials=14
"Lexan" "Lexan" "Lexan" "Lexan" "Lexan"
"Lexan" "Lexan" "Lexan" "Lexan" "Lexan"
"Lexan" "Lexan" "Lexan" "Brass"

```

In the same way, you can configure other tracks. Then the track1 on upper area looks like following figure.



A track pattern from the parameter above; (left) a complete set of the track view. (right) blocks are constructed in counterclockwise.

```
b:Ge/RangeModulatorA/PrintInformation = "True" #Print out specification, see below
```

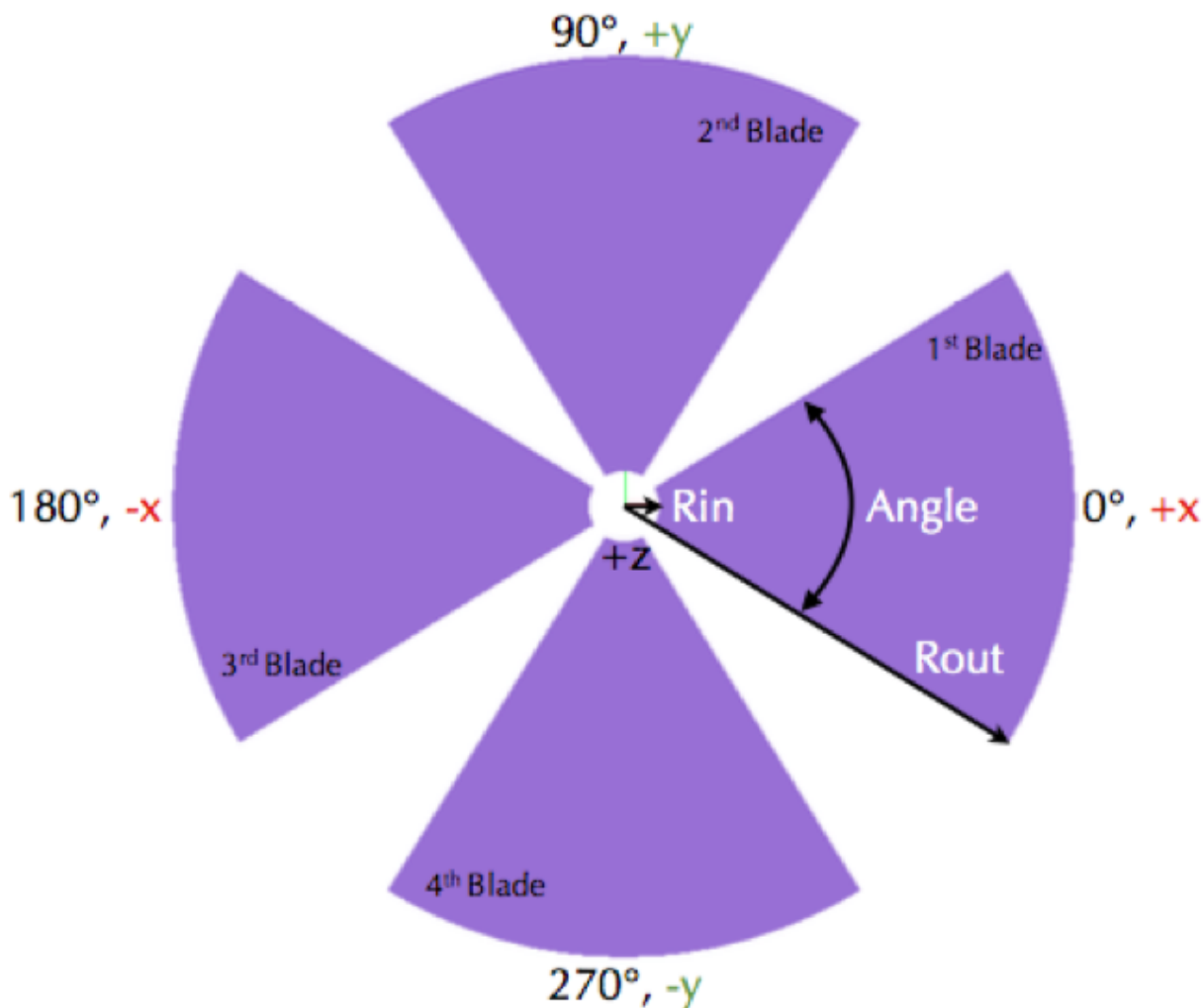
When TOPAS builds the geometry, you will see the numbers are input properly from console output as:

```
---UpperTrack1 , # of Blocks: 14
0th Block
  Angle : 5, 115 deg
  Height : 7.7 cm
  Material: Ts_Lexan
1st Block
  Angle : 115, 146.5 deg
  Height : 8.2 cm
  Material: Ts_Lexan
2nd Block
  Angle : 146.5, 173.2 deg
  Height : 8.7 cm
  Material: Ts_Lexan
3rd Block
  Angle : 173.2, 195.07 deg
  Height : 9.215 cm
  Material: Ts_Lexan
4th Block
  Angle : 195.07, 216.15 deg
  Height : 9.5 cm
  Material: Ts_Lexan
...
```

TOPAS RMW is a specialized geometry and so allows only the rotation around z-axis as well as the propeller rotation. Two examples demonstrate how to rotate RMW and modulate beam current using *Time Features* (*RangeModulator\_ConstantBeam.txt* and *RangeModulator\_CurrentModulatedBeam.txt*).

## 10.9.2 Propeller

A propeller is a component widely used to modulate the range of Bragg peaks. TOPAS currently supports a symmetrical propeller, i.e., each blade has the same shape but in different placements. Users can specify the number of blades with a spanning angle, thickness and materials of each layer. Here is an example of a single-layer propeller having 4 blades.



Each blade is constructed in the counterclockwise order. The figure shows its shape with coordination system.

Here is the complete set of the parameters for the above Propeller:

```
# Common parameters: type of geometry, position, and rotation
s:Ge/PropellerA/Type = "TsPropeller" #TsPropeller as type of geometry
s:Ge/PropellerA/Parent = "World"
s:Ge/PropellerA/Material = "Parent" #This is required to be set as "Parent"
d:Ge/PropellerA/TransX = 0.0 cm
d:Ge/PropellerA/TransY = 0.0 cm
d:Ge/PropellerA/TransZ = 0.0 cm
d:Ge/PropellerA/RotX = 0.0 deg
d:Ge/PropellerA/RotY = 0.0 deg
d:Ge/PropellerA/RotZ = 0.0 deg
b:Ge/PropellerA/Invisible = "true" #To avoid visualize propeller's mother volume.
```

(continues on next page)

(continued from previous page)

```

i:Ge/PropellerA/NbOfBlades = 4 #Number of blades
d:Ge/PropellerA/Rin = 10.0 mm #Inner radius, Rin in the figure
d:Ge/PropellerA/Rout = 127.5 mm #Outer radius, Rout in the figure

dv:Ge/PropellerA/Thickness = 1 0.356 mm #thickness.
dv:Ge/PropellerA/Angles = 1 63.15 deg #span angle of each blade.
sv:Ge/PropellerA/Materials = 1 "G4_POLYVINYL_ACETATE" #material.

# Visualization attributes setting
s:Ge/PropellerA/Blade/Material = Ge/PropellerA/Material
s:Ge/PropellerA/Blade/Color = "skyblue"
s:Ge/PropellerA/Blade/DrawingStyle = "Solid"
i:Ge/PropellerA/Blade/VisSegsPerCircle = 360
# you can apply a propeller specific step size
d:Ge/PropellerA/Blade/MaxStepSize = 10 mm
b:Ge/PropellerA/PrintInformation = "True" #Print out specification, see below

```

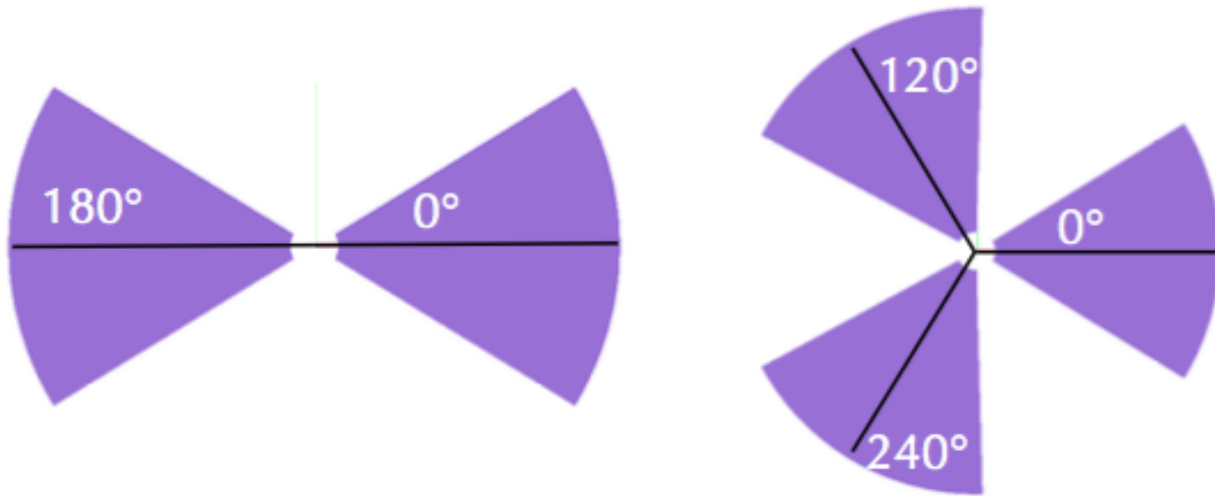
While TOPAS starts to build geometries, you can confirm whether the numbers you put are input properly from console output as:

```

Layer: "0" , Thickness: 0.0356 (cm), Angle: 63.15 (deg), Material: G4_POLYVINYL_
->ACETATE
Blade "0", Angle (-31.575 deg, 31.575deg)
Blade "1", Angle (88.425 deg, 151.575deg)
Blade "2", Angle (208.425 deg, 271.575deg)

```

With different numbers of blades, the angle of each blade will look like:



(Left) `Ge/PropellerA/NbOfBlades = 2`, (Right) `Ge/PropellerA/NbOfBlades = 3`.

You can model a multiple layered propeller just by extending the vector parameters, such as `Thickness`, `Angles`, and `Materials` (for more detail, see [Propeller.txt](#)):

```

dv:Ge/PropellerA/Thickness=10
0.5 0.6 0.7 0.8 0.9 1.0 1.1 1.2 1.3 1.4 mm
dv:Ge/PropellerA/Angles=10
63.15 54.15 46.450 40.65 35.85 31.8 28.1 24.725 21.8 19.1 deg

```

(continues on next page)

(continued from previous page)

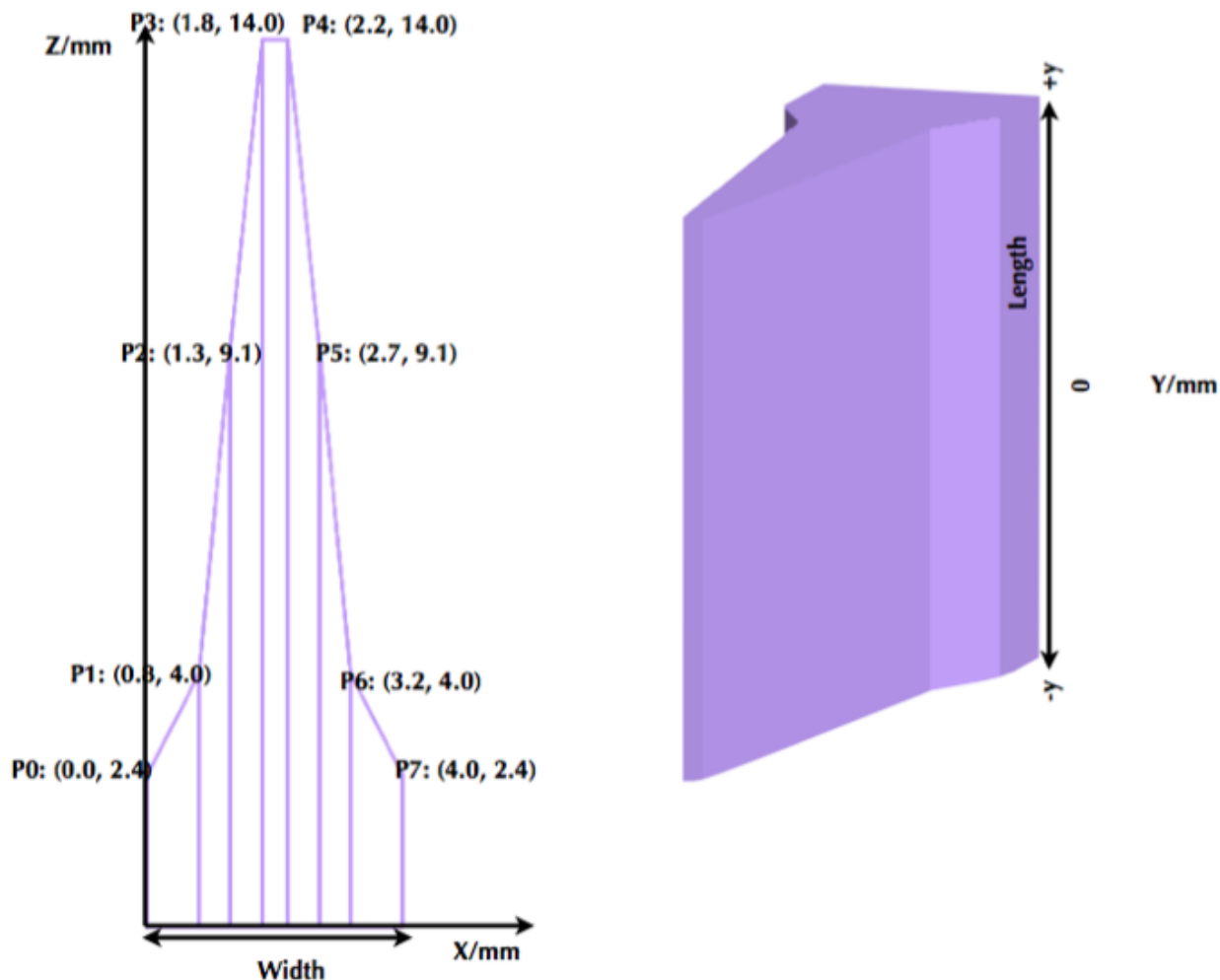
```
sv:Ge/PropellerA/Materials=10
"Lexan" "G4_WATER" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE"
"G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE"
"G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE"
```

Layers are created in the order of the parameter vector, i.e., Lexan is the bottom layer in this case, Water layer is the next, and so on. It is possible to make each layer with different thickness, angles, and materials. Note that these three vector parameters have same number of elements.

TOPAS propeller allows only the rotation around z-axis, which means that you can only assign rotation *Time Feature* to RotZ, such as `Ge/PropellerA/RotZ = Tf/ContinuousRotation/Value`. Two examples demonstrate how to handle propeller rotations (*Propeller\_ContinuousRotation.txt* and *Propeller\_StepRotation.txt*).

### 10.9.3 Ridge Filter

A ridge filter is an energy modulation component used in proton therapy. TOPAS offers a generic way to model an arbitrary shape of a ridge and place the replica. The shape of a ridge is defined in the x-z plane and then it becomes a volume by extending in the y direction.



(left) A ridge shape in X-Z plane, represented by points-connection. Because the connection starts at the origin and ends at the last point, (width, 0), so users need to define the width of a ridge first. Depending on the topology of points,



the arbitrary shape can be constructed. (right) A complete ridge by extending the shape along with y axis.

Here is a complete set of the parameters for the above ridge filter (see *RidgeFilter.txt* example):

```
# Common parameters: type of geometry, position, and rotation
s:Ge/RidgeFilterA/Type = "TsRidgeFilter"
s:Ge/RidgeFilterA/Parent = "RidgeGroup"
s:Ge/RidgeFilterA/Material = "Aluminum"
d:Ge/RidgeFilterA/TransX = 0.0 cm
d:Ge/RidgeFilterA/TransY = 0.0 cm
d:Ge/RidgeFilterA/TransZ = 0.0 cm
d:Ge/RidgeFilterA/RotX = 0.0 deg
d:Ge/RidgeFilterA/RotY = 0.0 deg
d:Ge/RidgeFilterA/RotZ = 0.0 deg
s:Ge/RidgeFilterA/DrawingStyle = "Solid"

#Ridge Filter-specific parameters;
#Width definition
#Note that points are sequentially connected.
#Number of XPoints and YPoints should be same.
dv:Ge/RidgeFilterA/XPoints = 8
0.0 0.8 1.3 1.8 2.2 2.7 3.2 4.0 mm
dv:Ge/RidgeFilterA/ZPoints = 8
2.4 4.0 9.1 14.0 14.0 9.1 4.0 2.4 mm
d:Ge/RidgeFilterA/Width = 4.0 mm
d:Ge/RidgeFilterA/Length = 1.0 cm
```

To check that the numbers are input properly:

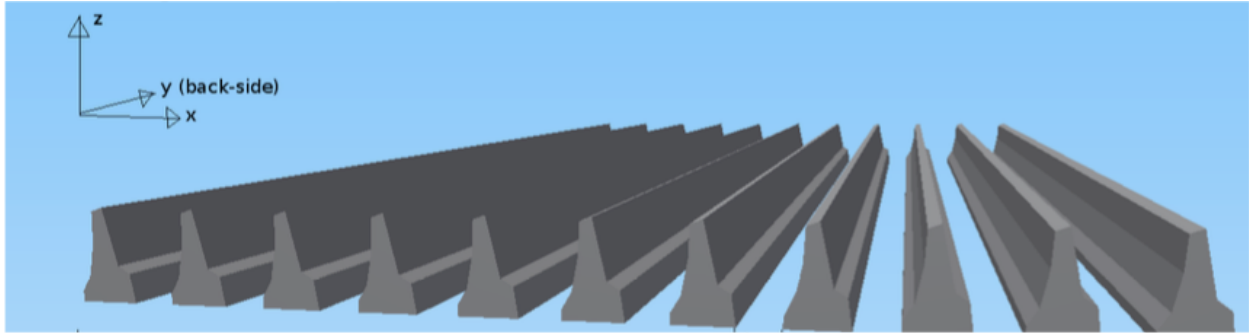
```
b:Ge/RidgeFilterA/PrintInformation = "True"
```

will generate console output as:

```
Ridge points (x,z) --- :8
P initial : (0, 0) cm
P 0th     : (0, 0.24) cm
P 1st     : (0.08, 0.4) cm
P 2nd     : (0.13, 0.91) cm
P 3rd     : (0.18, 1.4) cm
P 4th     : (0.22, 1.4) cm
P 5th     : (0.27, 0.91) cm
P 6th     : (0.32, 0.4) cm
P 7th     : (0.4, 0.24) cm
P final   : (0.4, 0) cm
```

You can make replicas of the ridge and their positions along the x-axis. A total of 3 replicas of the ridge and placed at -5.0, 0.0, 5.0. Each point represents the x-coordinate of the center of ridge width:

```
dv:Ge/RidgeFilterA/Displacement = 3 -5.0 0.0 5.0 mm
```

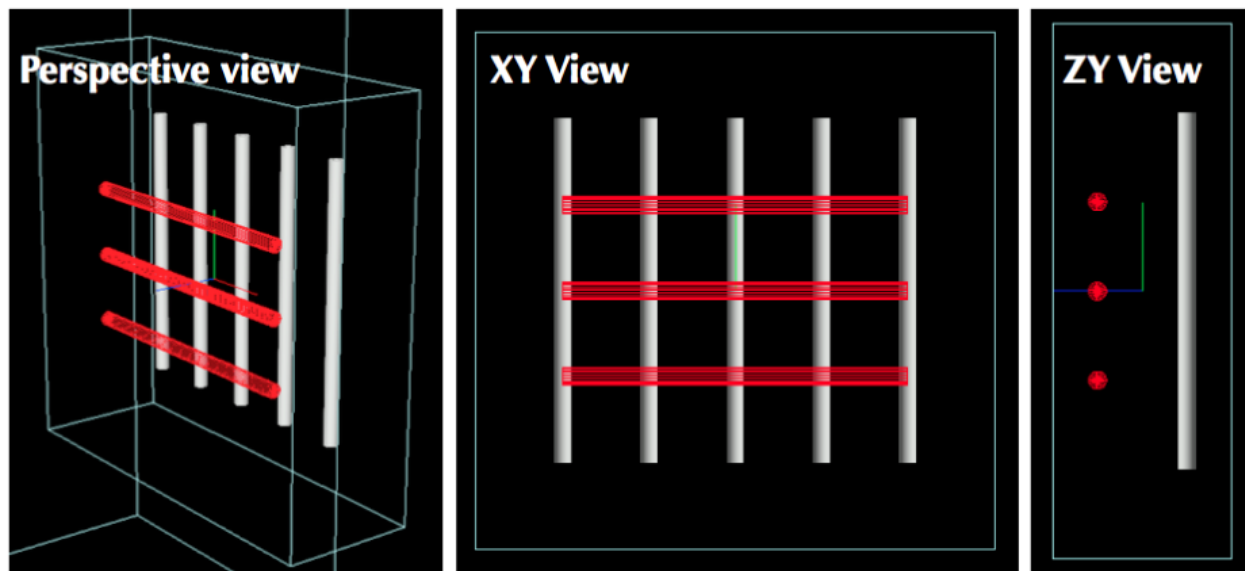


An example of replica set.

### 10.9.4 Multi Wire Chamber

A multi wire chamber may be built from many of geometry primitives such as `TsBox` and `TsCylinder`. However, It is quite cumbersome to place many wires individually and adjust their dimension on any request. So TOPAS multi wire chamber (`TsMultiWireChamber`) allows to instantiate many wires and to place them efficiently. `TsMultiWireChamber` is a box consisting of multiple sets of wires. Each set can have its own configuration, such as the dimension and material of the wires, spaces between wires, alignment axis, Z-positions, and drawing-style.

Here is an example of `TsMultiWireChamber` (see [MultiWire\\_Chamber.txt](#) example).



TOPAS multi wire chamber consists of two wire sets aligned along the X and Y axes. Three red wires are aligned to X axis while four gray wires are aligned to Y axis. These two sets of wires are placed within their mother box (gas filled).

The following parameters show how to model the above multi wire chamber:

```
s:Ge/WireChamberA/Parent = "World"
s:Ge/WireChamberA/Type = "TsMultiWireChamber" #Type of geometry
d:Ge/WireChamberA/HLX=30.0 cm #Chamber dimension
d:Ge/WireChamberA/HLY=30.0 cm
d:Ge/WireChamberA/HLZ=10.0 cm
s:Ge/WireChamberA/Material="Air" #Chamber is filled with this material.
d:Ge/WireChamberA/TransX=0.0 cm
```

(continues on next page)

(continued from previous page)

```

d:Ge/WireChamberA/TransY=0.0 cm
d:Ge/WireChamberA/TransZ=0.0 cm
d:Ge/WireChamberA/RotX=0.0 deg
d:Ge/WireChamberA/RotY=0.0 deg
d:Ge/WireChamberA/RotZ=0.0 deg
i:Ge/WireChamberA/NbOfLayers=2 #Number of wire sets.

# Parameters for specifying each wire set add 'Layer#' to geometry name.
# So Layer1 to Layer'NumberOfLayers'.
d:Ge/WireChamberA/Layer1/RMin=0.0 cm #Wire's inner radius
d:Ge/WireChamberA/Layer1/RMax=1.0 cm #Wire's outer radius
s:Ge/WireChamberA/Layer1/Material="Brass" #Wire material
d:Ge/WireChamberA/Layer1/HL=20.0 cm #Wire's half length
s:Ge/WireChamberA/Layer1/Align= "X" #Align axis
dv:Ge/WireChamberA/Layer1/Displacement=3 -10 0 10 cm #number of wires and its
    ↪ position in Y, because wires are aligned to X axis)
d:Ge/WireChamberA/Layer1/PosZ=5.0 cm #Z-position w.r.t its mother box.
s:Ge/WireChamberA/Layer1/DrawingStyle="FullWireFrame"
s:Ge/WireChamberA/Layer1/Color="red"

d:Ge/WireChamberA/Layer2/RMin=0.5 cm
d:Ge/WireChamberA/Layer2/RMax=1.0 cm
s:Ge/WireChamberA/Layer2/Material="Lexan"
d:Ge/WireChamberA/Layer2/HL=20.0 cm
s:Ge/WireChamberA/Layer2/Align= "Y"
dv:Ge/WireChamberA/Layer2/Displacement=5 -20 -10 0 10 20 cm
d:Ge/WireChamberA/Layer2/PosZ=-5.0 cm
s:Ge/WireChamberA/Layer2/DrawingStyle="Solid"
b:Ge/WireChamberA/PrintInformation = "True" #Print out specification, see below

```

When TOPAS starts to build geometries, you will see the numbers are input properly from console output as:

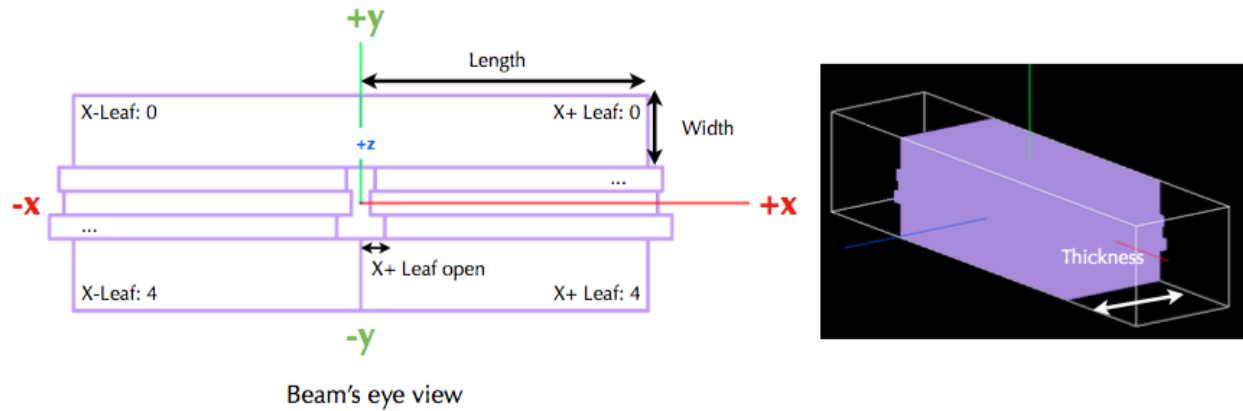
```

Layer: "0" , # of Wires: 3, Alignment: X, Wire (Rmin= 0 cm, Rmax= 1 cm, HL= 20 cm) ,
    ↪ Z Position in the Chamber: 5 (cm)
  Wire "0", Position (0 cm, -10 cm)
  Wire "1", Position (0 cm, 0 cm)
  Wire "2", Position (0 cm, 10 cm)
Layer: "1" , # of Wires: 5, Alignment: Y, Wire (Rmin= 0.5 cm, Rmax= 1 cm, HL= 20 cm) ,
    ↪ Z Position in the Chamber: -5 (cm)
  Wire "0", Position (-20 cm, 0 cm)
  Wire "1", Position (-10 cm, 0 cm)
  Wire "2", Position (0 cm, 0 cm)
  Wire "3", Position (10 cm, 0 cm)
  Wire "4", Position (20 cm, 0 cm)

```

### 10.9.5 Multi Leaf Collimator

Due to the design variations of Multi Leaf Collimator (MLC) from manufacturers, TOPAS provides a simplified MLC model instead of a generic design. With TOPAS MLC's minimal set of parameters, users can define various width of each leaf and opening of each leaf.



Illustrations for TOPAS MLC dimensions. The user can define an arbitrary number of leaves with different width of each leaf. TOPAS detects leaf collision when it is built and leaves are repositioned by *Time Features* operations.

Here is a complete set of the parameters for the above TOPAS MLC (see *MultiLeafCollimator.txt* example):

```
# Common parameters: type of geometry, position, and rotation
s:Ge/MultiLeafCollimatorA/Type = "TsMultiLeafCollimator"
s:Ge/MultiLeafCollimatorA/Parent = "World"
s:Ge/MultiLeafCollimatorA/Material = "Aluminum"
d:Ge/MultiLeafCollimatorA/TransX = 0.0 cm
d:Ge/MultiLeafCollimatorA/TransY = 0.0 cm
d:Ge/MultiLeafCollimatorA/TransZ = 0.0 cm
d:Ge/MultiLeafCollimatorA/RotX = 0.0 deg
d:Ge/MultiLeafCollimatorA/RotY = 0.0 deg
d:Ge/MultiLeafCollimatorA/RotZ = 0.0 deg
s:Ge/MultiLeafCollimatorA/DrawingStyle = "Solid"
b:Ge/MultiLeafCollimatorA/PrintInformation = "True"

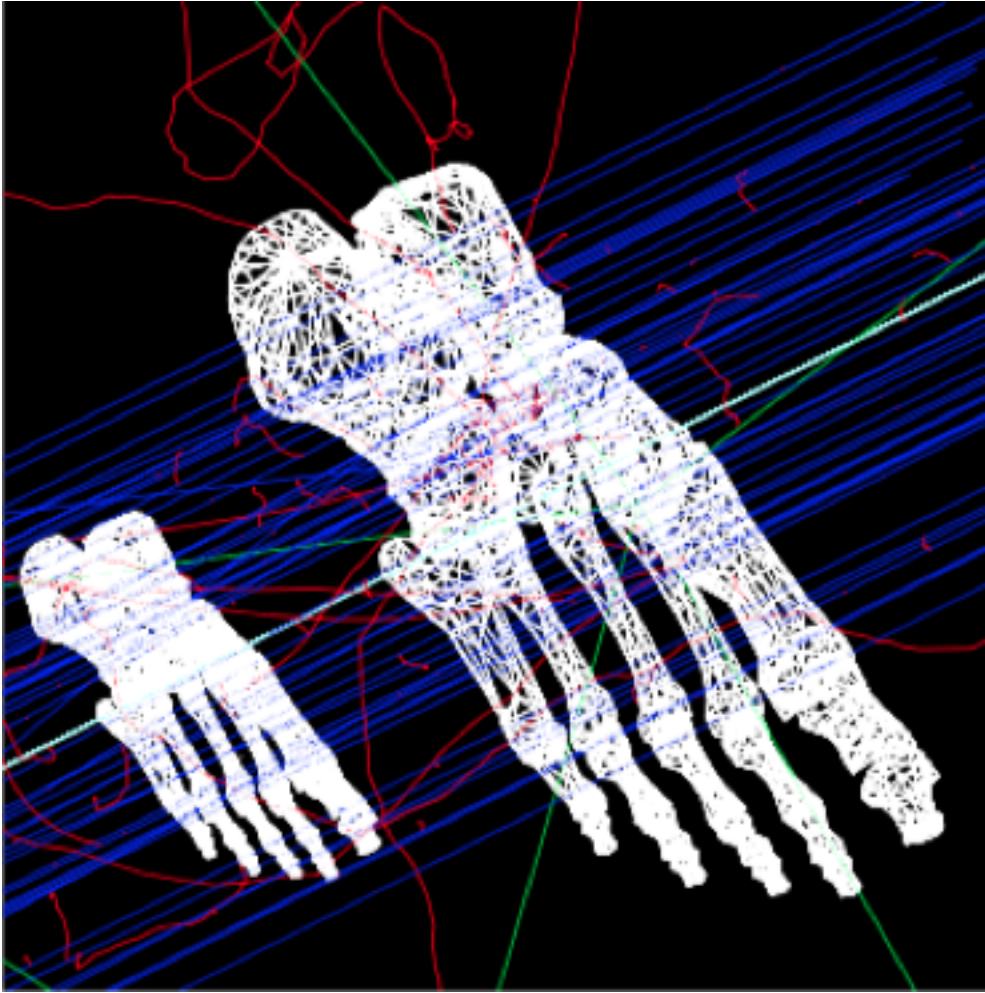
# MLC-specific parameters:
# Limits Leaf opening. Any of absolute values from X+- Leaf can't exceed this value.
d:Ge/MultiLeafCollimatorA/MaximumLeafOpen = 5.0 cm
d:Ge/MultiLeafCollimatorA/Thickness = 5.0 cm #Leaf thickness (z)
d:Ge/MultiLeafCollimatorA/Length = 6.0 cm #Leaf length (y)
dv:Ge/MultiLeafCollimatorA/Widths = 5 1.5 0.5 0.5 0.5 1.5 cm #Leaves width

# Each leaf's opening distance from Y axis.
# XMinusLeavesOpen means the x position of X- leaf's right edge.
# XPlusLeavesOpen means the x position of X+ leaf's left edge.
dv:Ge/MultiLeafCollimatorA/XMinusLeavesOpen = 5 0.0 -0.3 -0.2 -0.5 0.0 cm
dv:Ge/MultiLeafCollimatorA/XPlusLeavesOpen = 5 0.0 0.3 0.2 0.5 0.0 cm
```

TOPAS MLC is a specialized geometry and so allows only the reposition of each leaf as a function of time, using *Time Features* (see *MultiLeafCollimator\_sequence.txt* example).

### 10.9.6 CAD (Computer Aided Design)

The TsCAD component allows you to turn any geometry that has been designed in a CAD system into a TOPAS Component. This allows you to incorporate arbitrarily complex geometries.



The supported CAD formats are:

- STL - Stereolithography binary format
- PLY - Polygon ASCII format

STL and PLY files describe a geometry as a tessellation, providing a set of vertices and faces of triangular or quadrangular surfaces to approximate the volume. While some STL and PLY files also contain additional information such as material and color, TOPAS does not currently accept such information. The STL and PLY files you provide to TOPAS must contain only the tessellation information. Internally, TOPAS represents this component as a `G4TessellatedSolid`.

Most CAD systems allow direct export of parts to the above formats. If your CAD system does not support one of those formats, you may be able to convert from some other CAD format by using a free conversion tool such as [MeshLab](#).

```
# Common parameters: type of geometry, position, and rotation
s:Ge/MyPartFromCAD/Type = "TsCAD"
s:Ge/MyPartFromCAD/Parent = "World"
s:Ge/MyPartFromCAD/Material = "G4_WATER"
d:Ge/MyPartFromCAD/TransX = 0.0 cm
d:Ge/MyPartFromCAD/TransY = 0.0 cm
d:Ge/MyPartFromCAD/TransZ = 0.0 cm
d:Ge/MyPartFromCAD/RotX = 0.0 deg
```

(continues on next page)

(continued from previous page)

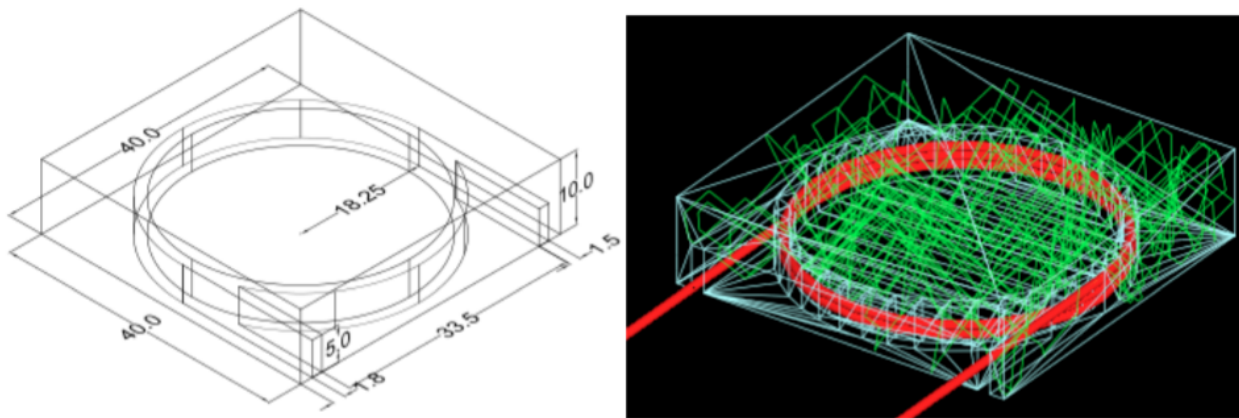
```

d:Ge/MyPartFromCAD/RotY = 0.0 deg
d:Ge/MyPartFromCAD/RotZ = 0.0 deg
s:Ge/MyPartFromCAD/DrawingStyle = "Wireframe"

# CAD-specific parameters:
s:Ge/MyPartFromCAD/InputFile = "Foot" # file name, without extensions. Match exact_
→case
s:Ge/MyPartFromCAD/FileFormat = "ply" # file extension
d:Ge/MyPartFromCAD/Units = 1.0 cm # how to interpret dimension numbers in the file.
→Changing this value will re-scale the component

```

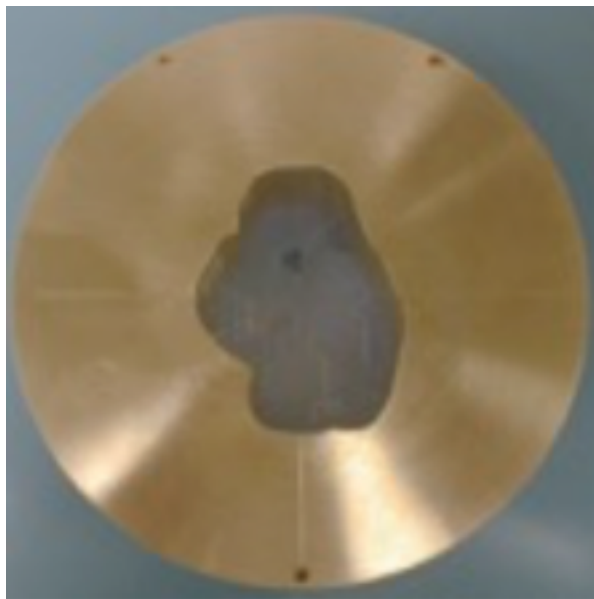
TOPAS does not automatically know where the center of your CAD component will be. This is affected by how your CAD system manages coordinates. For example, some CAD software exports the STL by relocating the volume to the first positive octant of its coordinate system. You may have to adjust the `TransX/Y/Z` parameters of your component to center it as desired.



Above, A plastic scintillator with customized groove. Left: CAD, Right: TOPAS

### 10.9.7 Aperture

An aperture is a component used to shape the lateral penumbra of a (generally) double-scattered proton beam. It is basically a block of brass with a hole cut out from the middle, in the shape of the treatment volume. The purpose is to block the beam outside the desired irradiation path. TOPAS models the aperture by connecting the aperture file points to create a polygon and then extruding this polygon in Z to cut out the aperture hole.



A typical implementation of an apertures in TOPAS is given below followed by a more detailed description of each option (see *ScatteringNozzle.txt* example):

```
s:Ge/Aperture/Type = "TsAperture"
s:Ge/Aperture/Parent = "Snout"
s:Ge/Aperture/Material= "Brass"
d:Ge/Aperture/RMax =4.5 cm
d:Ge/Aperture/HL= 2.5 cm
d:Ge/Aperture/TransX = 0.0 cm
d:Ge/Aperture/TransY = 0.0 cm
d:Ge/Aperture/TransZ = -13.0 cm
d:Ge/Aperture/RotX = 0.0 deg
d:Ge/Aperture/RotY = 0.0 deg
d:Ge/Aperture/RotZ = 0.0 deg
s:Ge/Aperture/InputFile = "ApertureFileIn.ap" # Match exact case
s:Ge/Aperture/FileFormat = "XYCoordinates" # XYCoordinates or MGH
b:Ge/Aperture/PrintPoints = "True" # Print points to console
```

FileFormat has two options:

- "XYCoordinates" takes is a simple list of points. The first line defines how many points there are in the file, each following line in the file is a comma separated x,y pair, such as:

- numberOfPoints
- x1,y1
- x2,y2
- ...
- xN,yN

N = numberOfPoints is the number of data points (xi, yi). This is a required condition. The units of the coordinates are millimeter.

- "MGH" takes the milling data produced by the MGH machine shop. It consists of the same information as the "XYCoordinates" option, but with more overhead, such as:
  - patientName

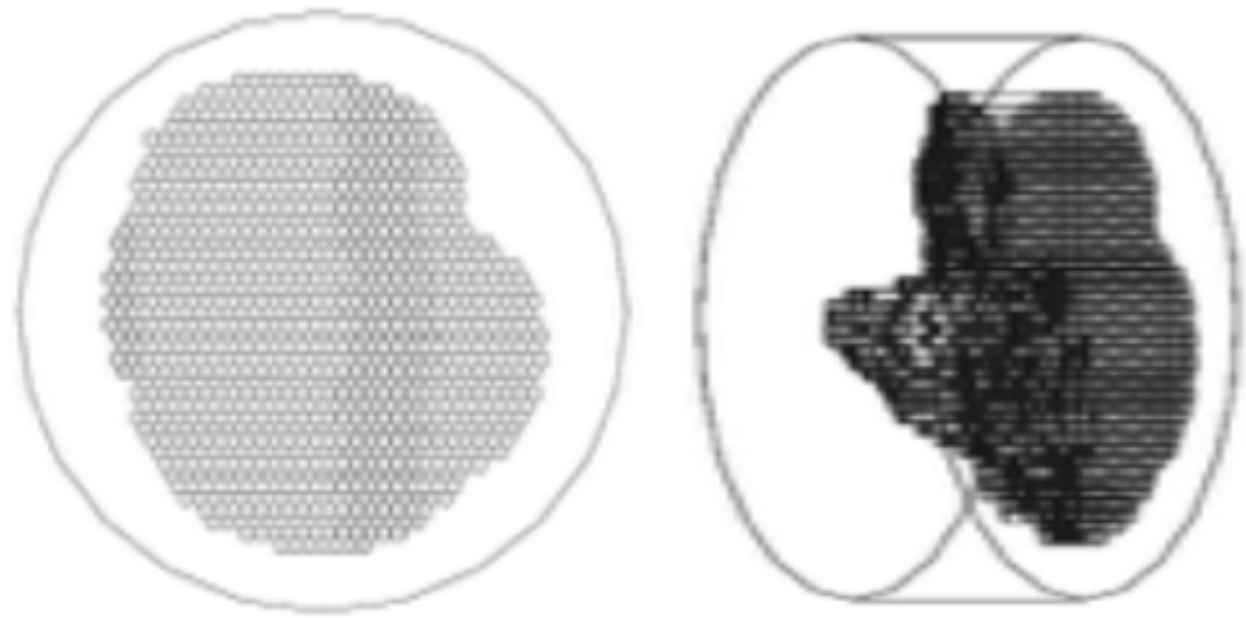


- Warning message about not fabricating this file
- someDoubleValue
- someIntValue
- M (this is a number of dummy points, this amount of points will be skipped)
- x1 y1 x2 y2 ... xM yM
- N
- x1 y1 x2 y2 ... xN yN

Data pairs are listed in a simple space-separated list. The units are in centimeter.

### 10.9.8 Compensator

A compensator is a component that is used to shape the distal edge of a proton beam by placing a varying amount of material in the beam path, usually behind the aperture. An example compensator is shown below in top and side views. The compensator consists of a material that is to be placed in the beam to attenuate the beam (usually lexan) and a number of drill holes that were drilled into the compensator.



A typical compensator has the following parameters (see *ScatteringNozzle.txt* example):

```
s:Ge/Compensator/Type = "TsCompensator"  
s:Ge/Compensator/Parent = "Snout"  
s:Ge/Compensator/Material = "CompensatorLucite"  
d:Ge/Compensator/RMax = 5.5. cm  
d:Ge/Compensator/TransX = 0. cm  
d:Ge/Compensator/TransY = 0. cm  
dc:Ge/Compensator/Thickness = 0. cm # will be reset to actual thickness when_  
→compensator is read in. This allows other parameter files to access this variable_  
→thickness.  
d:Ge/Compensator/InvHLL = -0.5 * Ge/Compensator/Thickness cm  
d:Ge/Compensator/TransZ = -15.5 cm + Ge/Compensator/InvHLL # Allows centering_  
→regardless of thickness
```

(continues on next page)



(continued from previous page)

```

d:Ge/Compensator/RotX = 0. deg
d:Ge/Compensator/RotY = 0. deg
d:Ge/Compensator/RotZ = 0. deg
s:Ge/Compensator/InputFile = "CompensatorFileInRowsDepths.rc" # match exact case
s:Ge/Compensator/FileFormat = "RowsAndDepths" # RowsAndDepths or MGH
s:Ge/Compensator/Method = "ExtrudedSolid" # Polyhedra, ExtrudedSolid,
↳SubtractionCylinders or UnionCylinders
b:Ge/Compensator/PrintPoints = "True"

```

Thickness has the special parameter type, *dc*, where the *c* means this dimensioned double is *changeable*, that is, it can change on the fly based on what exact compensator is read in. Other parameters can then take this thickness into account when they perform placements.

FileFormat has two options:

- "RowsAndDepths": all sizes are in millimeters:
  - numberOfRows
  - MainCylinderThickness
  - DrillHoleDiameter
  - n1 deltaX1 X1 Y1
  - D1 D2 ... Dn1
  - n2 deltaX2 X2 Y2
  - D1 D2 ... Dn2
  - ...
  - nN deltaXn Xn Yn
  - D1 D2 ... DnN

NumberOfRows = N defines how many rows of drill holes there are (in Y), the MainCylinderThickness. The DrillHoleDiameter is the diameter of the drill hole, we approximate this by a hexagon. The values *ni* are the number of drill holes in X for each row of drill holes in Y, *deltaXi* defines the step size (and direction) and *Xi* and *Yi* are the starting position of the drilling for this row.

- "MGH": all sizes are in inches:
  - Some line
  - numberOfRows
  - dummyDouble dummyDouble dummyDouble dummyDouble
  - MainCylinderThickness
  - dummyDouble dummyDouble dummyDouble dummyDouble dummyDouble dummyDouble dummyDouble
  - DrillHoleDiameter
  - n1 deltaX1 X1 Y1
  - D1 D2 ... Dn1
  - n2 deltaX2 X2 Y2
  - D1 D2 ... Dn2
  - ...

- $nN \Delta X_n X_n Y_n$
- $D1 D2 \dots DnN$

Method has four options:

- "ExtrudedSolid" builds the compensator from a set of extruded solids. This is the most reliable and efficient technique.
- "Polyhedra" carves hexagon shapes out of the compensator. This method has an extra check to adjust the position of each starting drill hole of each row to compensate for rounding inaccuracies produced by the drilling machine:

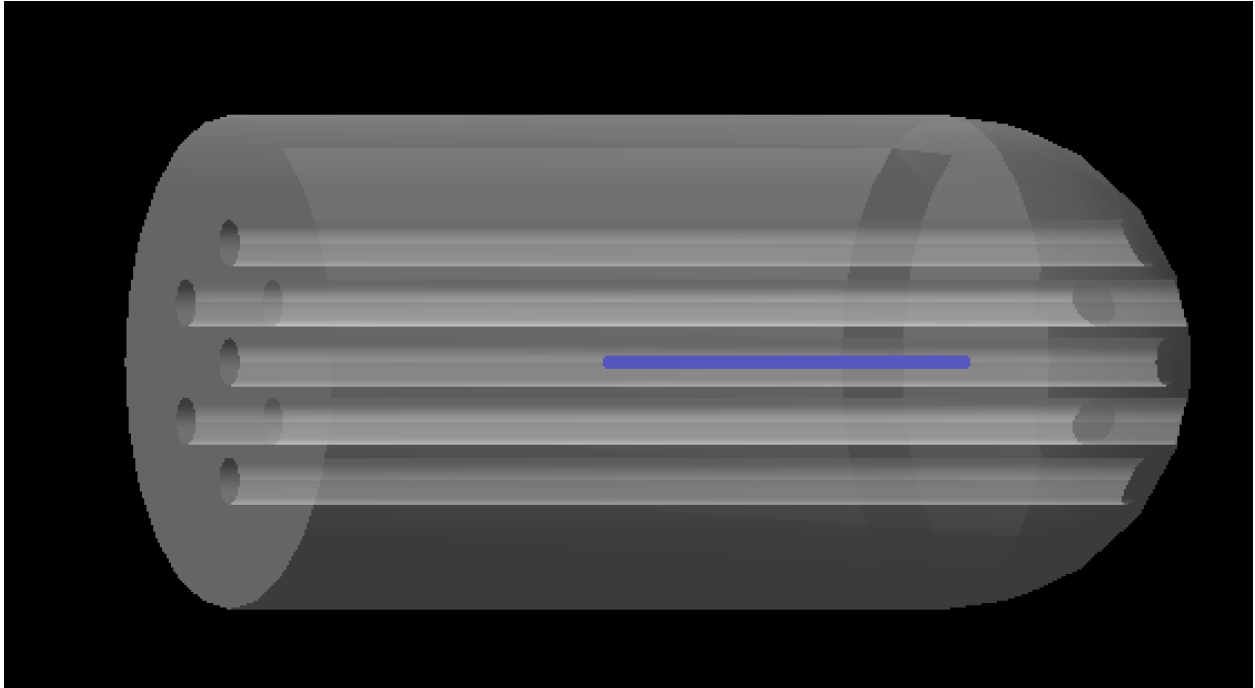
```
d:Ge/Compensator/XTolerance = 1. mm  
d:Ge/Compensator/YTolerance = 1. mm
```

- "SubtractionCylinders" builds the compensator by subtracting drill hole cylinders from the overall compensator cylinder. This technique gives the most perfect representation of the drilling process, however the added precision is insignificant, while particle navigation time is increased. Note that if you want to visualize this form of compensator, you should use RayTracer, as this is the only Geant4 visualization drivers that can correctly render boolean operations.
- "UnionCylinders" builds the compensator by first creating a union solid of all the holes, and then subtracting this union solid from the overall compensator cylinder. This technique is similar to "SubtractionCylinders" but slightly more efficient. Note that if you want to visualize this form of compensator, you should use RayTracer, as this is the only Geant4 visualization drivers that can correctly render boolean operations.

### 10.9.9 BrachyApplicator

A BrachyApplicator is a component that is used to precisely place source wires for brachytherapy applications. It is the first of what we intend to become a large library of new components for brachytherapy applications.

The overall shape of the BrachyApplicator is a cylinder with a hemispherical cap on one end. There is one hole in the center into which a source wire can be driven, plus a configurable number of other holes located radially around this center. The component generates additional parameters at run-time to represent the x and y translations of these holes relative to the applicator center line. These parameters can be used to easily position the source wire into these holes.



The following example parameters are taken from the new example: `examples/Patient/Applicator.txt` This example places the applicator inside of a patient, and then uses time features to drive a source wire to various dwell positions within this applicator.

**The parameters you set are as follows.** `s:Ge/Applicator/Type = "TsBrachyApplicator"` `s:Ge/Applicator/Parent = "Patient"` `b:Ge/Applicator/IsParallel = "True"` `s:Ge/Applicator/Material = "G4_WATER"` `d:Ge/Applicator/CylinderLength = 40. mm` `d:Ge/Applicator/Radius = 12.5 mm` `i:Ge/Applicator/NumberOfRadialHoles = 6` `d:Ge/Applicator/HoleOffset = 6 mm` # Distance of radial holes from center `d:Ge/Applicator/HoleRadius = 1.2 mm` `s:Ge/Applicator/DrawingStyle = "Solid"` `s:Ge/Applicator/Color = "transparentgrey"` `iv:Gr/Color/transparentgrey = 4 255 255 255 90`

The following parameters are updated automatically by the applicator component to show true hole centers. They need to be defined here, but these initial values are not important. They must run from `Hole0` (for the central hole) to `HoleN`, for the Nth radial hole.

```

d:Ge/Applicator/Hole0/TransX = 0. mm d:Ge/Applicator/Hole0/TransY = 0. mm
d:Ge/Applicator/Hole1/TransX = 0. mm d:Ge/Applicator/Hole1/TransY = 0. mm
d:Ge/Applicator/Hole2/TransX = 0. mm d:Ge/Applicator/Hole2/TransY = 0. mm
d:Ge/Applicator/Hole3/TransX = 0. mm d:Ge/Applicator/Hole3/TransY = 0. mm
d:Ge/Applicator/Hole4/TransX = 0. mm d:Ge/Applicator/Hole4/TransY = 0. mm
d:Ge/Applicator/Hole5/TransX = 0. mm d:Ge/Applicator/Hole5/TransY = 0. mm
d:Ge/Applicator/Hole6/TransX = 0. mm d:Ge/Applicator/Hole6/TransY = 0. mm

```

## 10.10 Patient Components

TOPAS currently supports the following Patient Component types:

| Geometry Component   | Type                                     |
|--|--|
| <i>Patient in DICOM Format</i>   | TsDicomPatient                           |
| <i>Patient in ImageCube Format (handles XCAT, XiO, MaterialTagNumber and more)</i> | TsImageCube (handles XCAT, XiO and more) |

It is also necessary to define how to convert the imaging data to material data, following a *Imaging to Material Conversion* scheme.

### 10.10.1 Common Parameters

Many of the parameters for Patient Components are common to both TsDicomPatient and TsImageCube. These are described here.

To perform Monte Carlo simulation, TOPAS needs to map each voxel of the patient image to a material, density and, for useful graphics, a color. You specify how to do this by telling TOPAS which *Imaging to Material Conversion* to use.

To dump your file's raw imaging values to the console:

```
b:Ge/Patient/DumpImagingValues = "True"
```

Set any parent you like, but it is often convenient to place patient into a group component which can then be rotated to represent couch setup:

```
s:Ge/Patient/Parent = "PatientGroup"
```

Even though a large number of materials are defined in your HU conversion file, TOPAS will only create those materials that are actually used in your CT image. In the 4DCT case, if any image introduces new materials that were not in the first image, Geant4 will be unable to proceed (it cannot load new materials after physics has initialized). TOPAS will exit with a warning message advising you to set the parameter:

```
b:Ge/Patient/PreLoadAllMaterials = "True"
```

Startup will then be slower, since TOPAS will preload the full set of materials defined in your HU conversion file, but your 4DCT will then work.

For single slice thickness images, scoring will use the same voxel divisions as your CT image. For multiple slice thicknesses, scoring will not know what divisions to use unless you explicitly specify these in your scoring parameters, such as:

```
i:Sc/MyScorer/XBins = 512
i:Sc/MyScorer/YBins = 512
i:Sc/MyScorer/ZBins = 256
```

The built-in Geant4 visualization tools do not perform well when a complex voxel structure is loaded. To make visualization more successful, several additional parameters are provided.

There is generally little value in showing all pixels of the image at once. Each slice just covers up the last slice. To instead show only a specific set of slices in any dimension:

```
iv:Gr/Patient/ShowSpecificSlicesZ = 4 1 3 9 12 # will only show slices 1, 3, 9 and 12.
```

Number of slices begins at number 1.

Similar slicing is allowed in X and Y. Three special values are also allowed:

```
iv:Gr/Patient/ShowSpecificSlicesZ = 1 0 # means show all slices
iv:Gr/Patient/ShowSpecificSlicesZ = 1 -1 # means only show center slice
iv:Gr/Patient/ShowSpecificSlicesZ = 1 -2 # means only first, center and last slice
```

The following will result in a display that shows 27 pixels comprising the boundaries and center of the image. This allows you to see the overall placement of the image and see the individual voxel size:

```
iv:Gr/Patient/ShowSpecificSlicesX = 1 -2 # means only show center slice
iv:Gr/Patient/ShowSpecificSlicesY = 1 -2 # means only show center slice
iv:Gr/Patient/ShowSpecificSlicesZ = 1 -2 # means only show center slice
```

You have the Option to Load a Truncated DICOM, XiO or ImageCube. Parameters to restrict which voxels of a patient image are loaded are:

```
i:Ge//RestrictVoxelsXMin          i:Ge//RestrictVoxelsYMin          i:Ge//RestrictVoxelsZMin
i:Ge//RestrictVoxelsXMax i:Ge//RestrictVoxelsYMax i:Ge//RestrictVoxelsZMax
```

The previously mentioned parameters, ShowSpecificSlicesVoxels, affect only the graphical output of the voxels. The voxels were still all loaded, and all affected the physics results, but only specific voxels were shown in graphics.

The RestrictVoxels... parameters have a much more significant effect: they cause TOPAS to not even load some parts of the patient. This can save a lot of time and allow you to test things on DICOM or other image formats quickly, as you only have to load whatever part of the DICOM you actually want to use. Beware though: if you use this option, you will change your results. It is not just a graphics thing. It actually avoids creating the given voxels at all.

Though our main motivation for this work was to get a way to do more rapid development (testing on real patients without the slowdown of loading everything), there may be a value also for our brachytherapy users. They may only need to load the part of the image that is near the brachy source.

Another option allows you to specify the maximum number of voxels to show. If the total number of voxels is greater than this limit, TOPAS will just draw the overall DICOM outline:

```
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 8000
```

## 10.10.2 Patient in DICOM Format

DICOM import is handled through the [GDCM](#) package, which is pre-built into TOPAS.

See the [DoseToCT.txt](#) and [DoseTo4DCT.txt](#) examples of how to use TsDicomPatient. Note that before running this example, you must unzip the included DICOM files.

You specify the name of a directory containing one or more dcm files (one for each slice):

```
s:Ge/Patient/DicomDirectory = "DICOM_Box"
```

To specify 4DCT, you can have DicomDirectory change under control of a [Time Feature](#).

Files of other types in this directory will be ignored. Exact titles of the dcm files are not important as TOPAS will re-order them based on the slice ordering information inside the DICOM headers.

By default, Topas will only consider dcm files that are from CT. This can be adjusted by:

```
sv:Ge/Patient/DicomModalityTags = 1 "CT" # defaults to just CT
```

Other modality tags are, for example, "MR" for Magnetic Resonance and "US" for Ultrasound. A complete list can be found [here](#).

Patient positioning information from the DICOM file is not currently used. You must position as you would for any TOPAS component:

```
d:Ge/Patient/TransX=0. m
d:Ge/Patient/TransY=0. m
d:Ge/Patient/TransZ=0. m
d:Ge/Patient/RotX=0. deg
```

(continues on next page)

(continued from previous page)

```
d:Ge/Patient/RotY=0. deg
d:Ge/Patient/RotZ=0. deg
```

TOPAS can read DICOM RT Structure Sets. A structure set is an extra file in the DICOM directory that provides information on structures such as organs, tumors, PTVs, etc. that have been outlined (contoured) in the planning process. The data is stored as a set of polygons, up to one per slice for each contoured structure. TOPAS can color code DICOM components according to this structure information and can filter scoring based on these structures (see the filter: OnlyIncludeIfInRTStructure).

To make TOPAS color the voxels by structure:

```
sv:Ge/Patient/ColorByRTStructNames = 2 "R_LUNG" "L_LUNG"
sv:Ge/Patient/ColorByRTStructColors = 2 "yellow" "red"
```

- If the structure name includes a space, substitute an underscore in the parameter. So, for example, if the structure name is “R LUNG”, you should supply the parameter as “R\_LUNG”.
- If you don’t actually know what structures are included in your DICOM, just try providing in ColorByRTStructNames. TOPAS will give you an error message that includes a list of the known structure names.
- To allow easy testing of this feature in simple DICOM examples that don’t really have any structures, the following parameter will “fake” an RT structure set, assigning the given structure to all voxels in the lower XY quadrant:

```
b:Ge/Patient/FakeStructures = "True"
```

TOPAS can automatically set DicomOrigin parameters to help with patient positioning.

If you define a set of DicomOrigin parameters for your patient:

```
dc:Ge/Patient/DicomOriginX = 0.0 mm
dc:Ge/Patient/DicomOriginY = 0.0 mm
dc:Ge/Patient/DicomOriginZ = 0.0 mm
```

then when you read in a TsDicomPatient, TOPAS will update these parameters on the fly to provide the origin of the DICOM coordinate system specified in the TOPAS coordinate system.

You can combine this information with other information you may have about your isocenter to get your patient properly positioned. For example, if you just wanted to center your patient in its parent component, such as PatientGroup, you would do:

```
s:Ge/Patient/Type = "TsDicomPatient"
s:Ge/Patient/Parent = "PatientGroup"
d:Ge/Patient/TransX = 0.0 mm
d:Ge/Patient/TransY = 0.0 mm
d:Ge/Patient/TransZ = 0.0 mm
```

If you also had isocenter information from an RT-Plan in DICOM coordinates:

```
d:Rt/plan/IsoCenterX = 0.0 mm
d:Rt/plan/IsoCenterY = -99.9904 mm
d:Rt/plan/IsoCenterZ = -14.0 mm
```

you could adjust the patient to isocenter by doing:

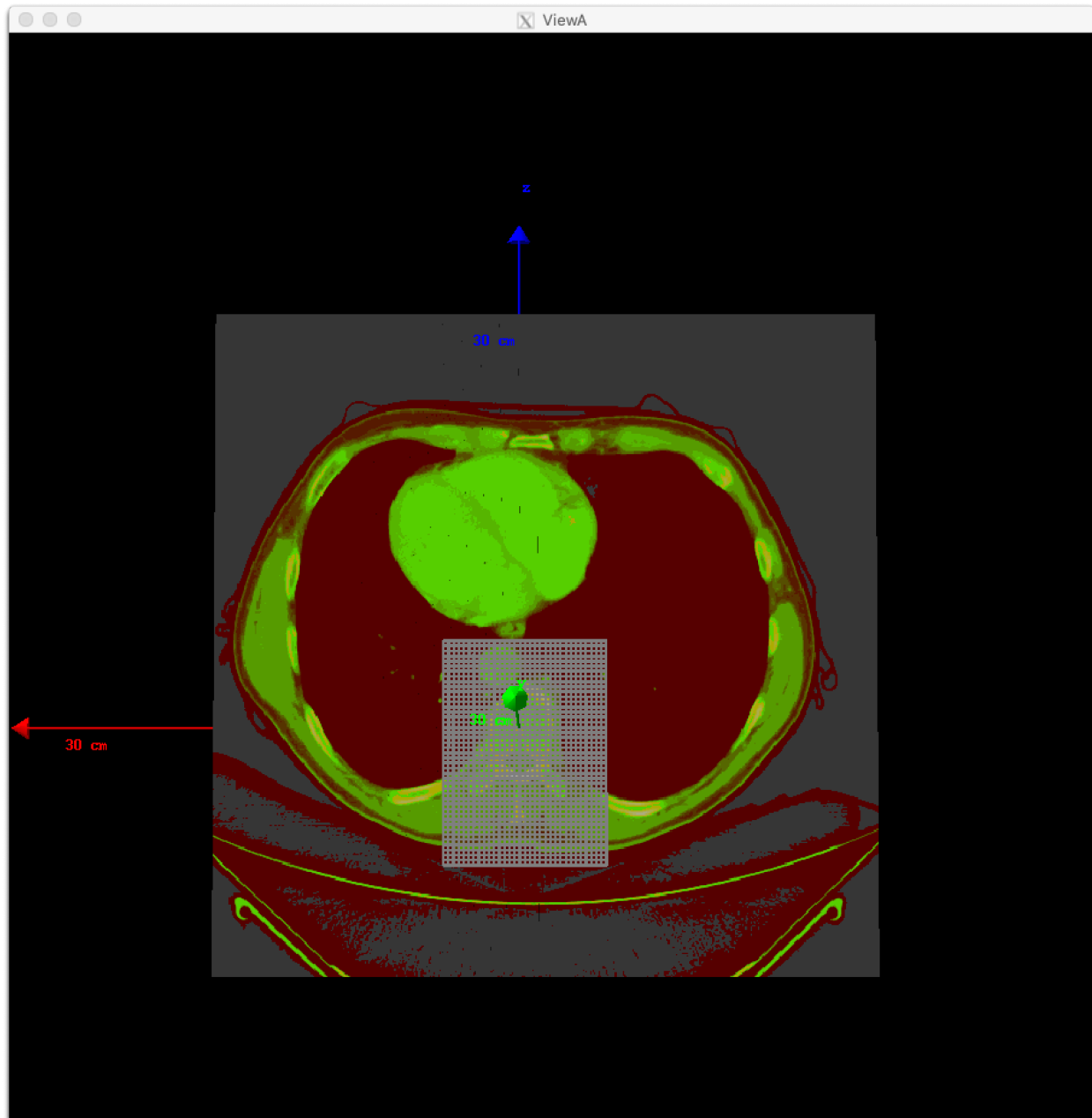
```
d:Ge/Patient/TransX = Ge/Patient/DicomOriginX - Rt/plan/IsoCenterX mm
d:Ge/Patient/TransY = Ge/Patient/DicomOriginY - Rt/plan/IsoCenterY mm
d:Ge/Patient/TransZ = Ge/Patient/DicomOriginZ - Rt/plan/IsoCenterZ mm
```

See [ViewAbdomen\\_rtdose.txt](#) for an example of how to use these patient-positioning features.

TOPAS can automatically create a Scoring Grid that exactly matches a provided RTDOSE file in your DICOM dataset. This makes it easier to compare TOPAS results to Treatment Planning System results.

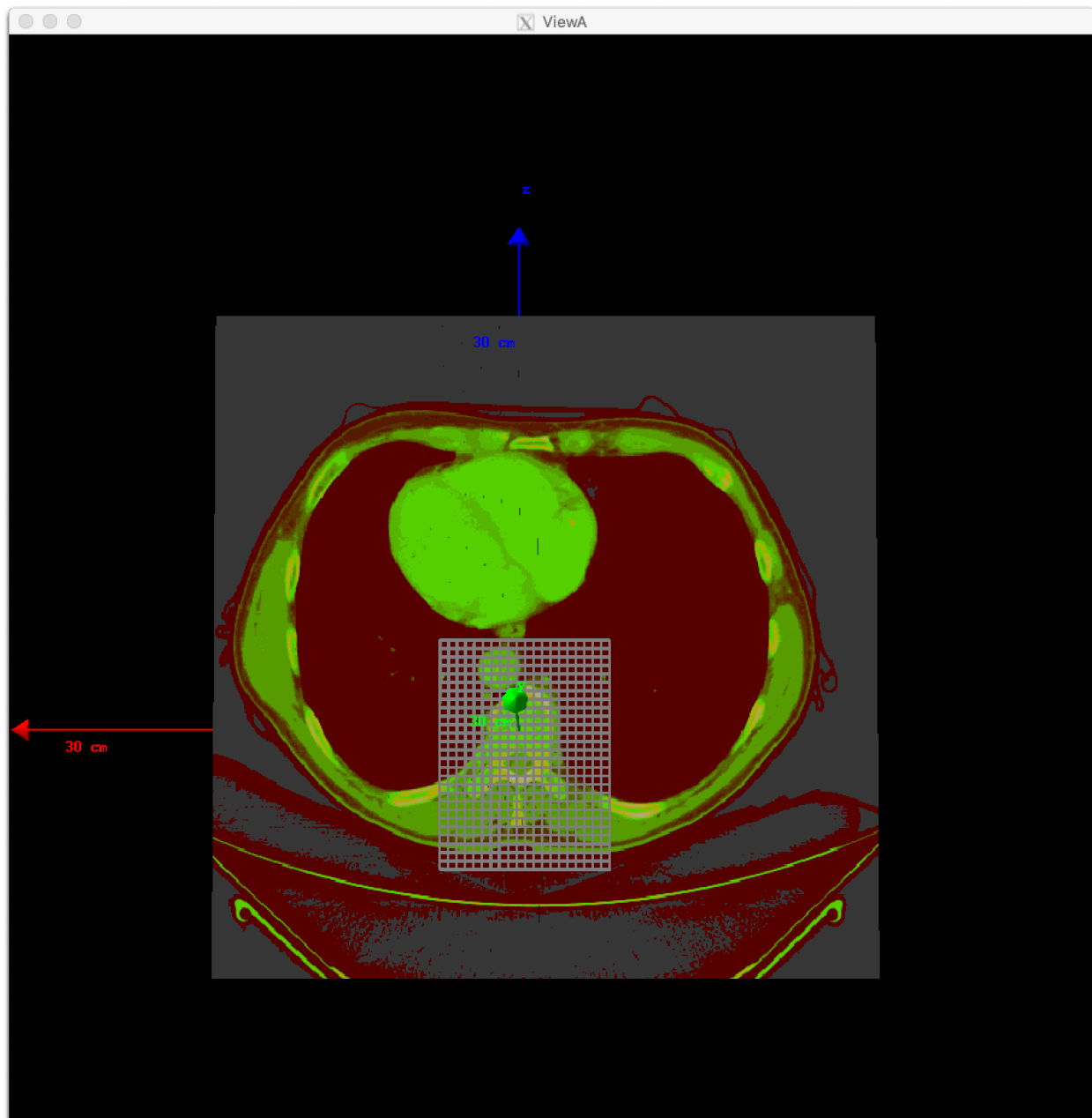
Tell TOPAS which RTDOSE file to use by providing “CloneRTDoseGridFrom” parameter, such as:

```
s:Ge/Patient/CloneRTDoseGridFrom = Ge/Patient/DicomDirectory + "/RTDOSE.dcm"
```



You can adjust pixel sizes while keeping the position, such as:

```
s:Ge/Patient/CloneRTDoseGridFrom = Ge/Patient/DicomDirectory + "/rtdose.dcm"  
dv:Ge/Patient/CloneRTDoseGridSize = 3 5 5 8 mm
```

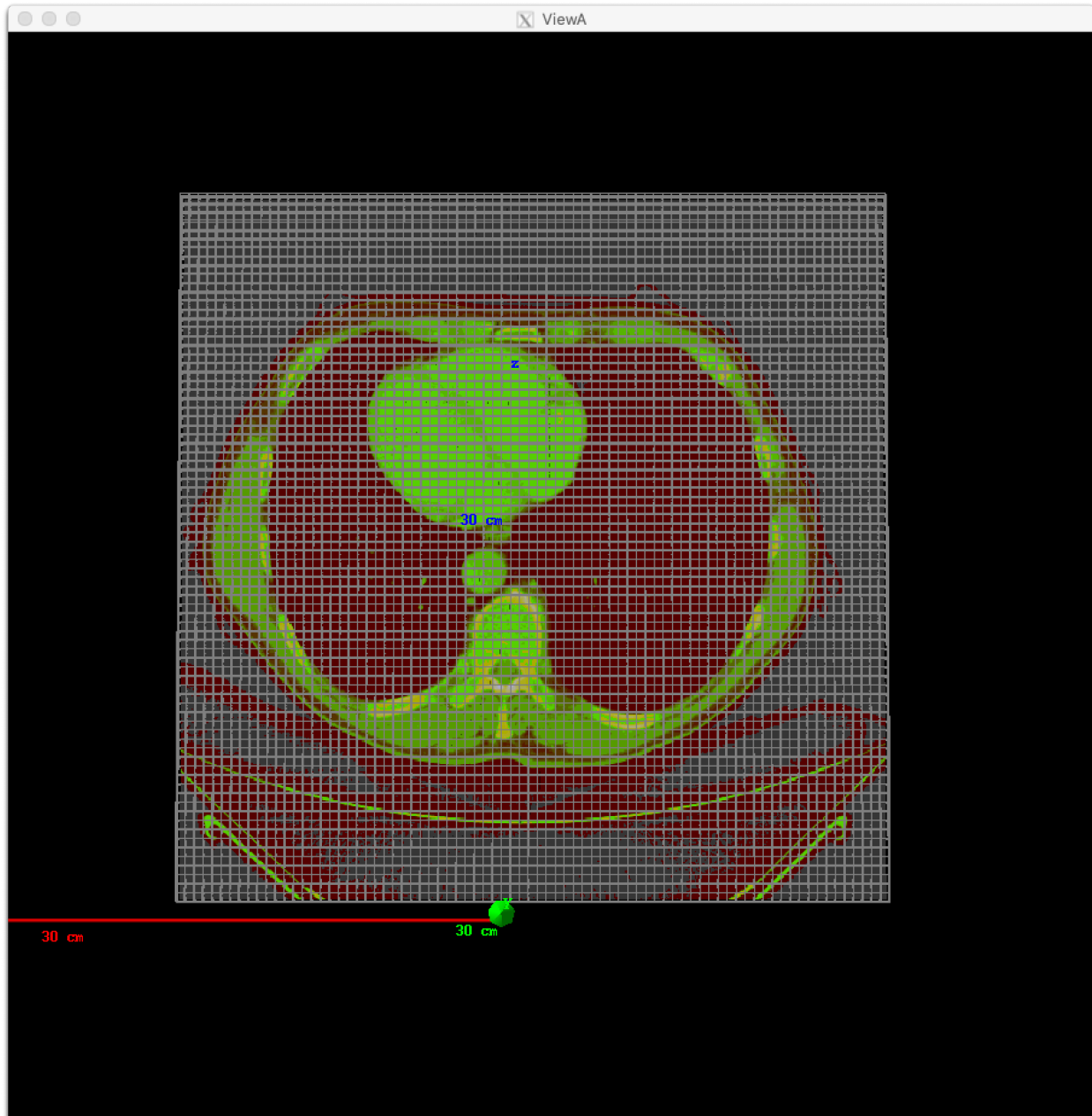


In addition, you can define a dose grid without a RTDOSE file by specifying its grid size such as:

```
dv:Ge/Patient/CloneRTDoseGridSize = 3 5 5 5 mm
```

A dose grid that fits the volume of a TsDicomPatient will be created.





TOPAS will then automatically create a scoring volume in a parallel world to overlay your grid, and will name this component with the same name as your patient, plus “/RTDoseGrid”. You can then score on this component just like on any other component:

```
s:Sc/Dose/Component = "Patient/RTDoseGrid"
```

TOPAS Scoring can use information from your DICOM dataset so that scored results can be more easily compared to those from treatment planning systems, by using a consistent coordinate system. See [DICOM Output](#) for more information.

### 10.10.3 Patient in ImageCube Format (handles XCAT, XiO, MaterialTagNumber and more)

We refer to a patient input file as an “Image Cube” if it is a simple binary file that contains one value for each voxel. These values may be Housefield units or any other sort of imaging information that you have. Elsewhere you will tell TOPAS how to convert a given value from this file into a specific material for that voxel.

- For the case of an XCAT phantom, the binary file will contain, for each voxel, an activation or attenuation value as a float
- For the case of an XiO patient, the binary file will contain, for each voxel, a Hounsfield value as a short
- For the case of a MaterialTagNumber file, the binary file will contain, for each voxel, a material index as a short
- For other cases, you can provide a binary file that contains, for each voxel, any float, int or short  
(and for XCAT phantoms, there may be an additional file, an XCAT log, that provides metadata)

An ImagingToMaterialConverter, described in the next section, handles the details of how each voxel’s information is converted to a material specification.

See the *XCAT.txt* example of how to read an XCAT file. See the *DoseToCT.txt* example of how to read an XiO file.

Specify file directory and file name:

```
s:Ge/Patient/InputDirectory = "./"
s:Ge/Patient/InputFile = "ctvolume.dat" # match exact case
```

To specify 4DCT, you can have `InputDirectory` or `InputFile` change under control of a *Time Feature*.

For 4D CT, a parameter lets you decide whether the material maps for the various CT phases will be cached (saving time if they are reused again) or not (saving memory):

b:Ge/CacheMaterialMapForEachTimeSlice defaults to True

You must position as you would for any TOPAS component:

```
d:Ge/Patient/RotX = 0. deg
d:Ge/Patient/RotY = 90. deg
d:Ge/Patient/RotZ = 0. deg
d:Ge/Patient/TransX = 1.5 mm
d:Ge/Patient/TransY = 3.3 mm
d:Ge/Patient/TransZ = 4.2 mm
```

TOPAS then needs some metadata: specifically it needs to know:

- how many voxels there are in each dimension
- how large the voxels are in each dimension
- what data type is involved (float, int or short)
- how to convert the given value to a material

For XCAT phantoms, all of this metadata can come from an XCAT log file:

```
s:Ge/Patient/MetaDataFile = "XCAT_FullMouse_86x86x161_atn_1.log"
```

If you had some other form of Image Cube (not XCAT), or you don’t want to read this information from an XCAT log file, you can provide this meta data as TOPAS parameters:

```
s:Ge/Patient/DataType = "FLOAT" # "SHORT", "INT" or "FLOAT"
i:Ge/Patient/NumberOfVoxelsX = 86
i:Ge/Patient/NumberOfVoxelsY = 86
i:Ge/Patient/NumberOfVoxelsZ = 161
d:Ge/Patient/VoxelSizeX = .5 mm
d:Ge/Patient/VoxelSizeY = .5 mm
d:Ge/Patient/VoxelSizeZ = .5 mm
```

If there are multiple slice thicknesses in your image, use vectors to specify number and thickness of voxels in each section. For example, a 30 slice image that has 10 slices of 2.5 mm and then 20 slices of 1.25 mm:

```
iv:Ge/Patient/NumberOfVoxelsZ = 2 10 20
dv:Ge/Patient/VoxelSizeZ = 2 2.5 1.25 mm
```

If you are using XCAT without providing metadata from an XCAT log file, you should also provide parameters to tell TOPAS what material to use for a given value found in the XCAT binary file, such as:

```
u:Ge/Patient/AttenuationForMaterial_XCAT_Air = 0.
u:Ge/Patient/AttenuationForMaterial_XCAT_Muscle = 195.2515
u:Ge/Patient/AttenuationForMaterial_XCAT_Lung = 57.5347
```

## 10.10.4 Imaging to Material Conversion

You are free to write your own converter, including approaches that use alternative imaging modalities (e.g. MRI, pCT, ultrasound), or that use more than one image (e.g. Dual Energy CT, Multi-Energy CT). To write your own converter, see [Custom Imaging to Material Conversion](#).

## 10.10.5 XCAT

TOPAS provides two built-in converters for XCAT and other Image Cube data:

```
s:Ge/Patient/ImagingToMaterialConverter = "XCAT_Attenuation" # "XCAT_Activity"
```

These converters assume the value found in the binary file for a given voxel is either an Attenuation or an Activity. They then convert the given value to a material name from either the metadata file (the XCAT log file) or from explicit parameters you have specified such as:

```
u:Ge/Patient/AttenuationForMaterial_XCAT_Air = 0.
u:Ge/Patient/AttenuationForMaterial_XCAT_Muscle = 195.2515
u:Ge/Patient/AttenuationForMaterial_XCAT_Lung = 57.5347
```

The actual material name that TOPAS will expect you to define somewhere is the part after "AttenuationForMaterial\_", such as XCAT\_Air and XCAT\_Muscle. You need to make sure that these material names have been defined somewhere in your TOPAS parameters. In our XCAT example we defined these in the file XCAT\_Materials.txt. Two notes on this example XCAT\_Materials file:

- We faked the definitions, defining all the materials as different colors of what is really just water. You could edit this file to provide the real elemental compositions of the various materials.
- We only defined the materials used in the attenuation part of the XCAT log file. If you instead want to use the materials used in the activity part of the XCAT log file, you'll need to define some additional materials (the activity part of that XCAT log file had more materials than the attenuation part).

### 10.10.6 MaterialTagNumber

Some of our users have TsImageCube components where each voxel is represented not as a CT number but as an integer “tag number,” a 16-bit integer (C++ short) that corresponds to a particular material name. The ImagingToMaterialConverter called MaterialTagNumber will interpret these tag numbers based on a lookup table created by two additional TOPAS vector parameters, MaterialTagNumbers and MaterialNames. For example:

```
s:Ge/Patient/Type = "TsImageCube"
s:Ge/Patient/ImagingToMaterialConverter = "MaterialTagNumber"
iv:Ge/Patient/MaterialTagNumbers = 6 0 3 42 43 100 110
sv:Ge/Patient/MaterialNames = 6 "Air" "G4_BLOOD_ICRP" "G4_BONE_CORTICAL_ICRP" "G4_
→BONE_COMPACT_ICRU" "G4_BRAIN_ICRP" "G4_MUSCLE_SKELETAL_ICRP"
```

Thus:

- Where the voxel is tagged with the number 0, the converter will interpret this as “Air”
- Where the voxel is tagged with the number 3, the converter will interpret this as ” G4\_BLOOD\_ICRP “
- Where the voxel is tagged with the number 42, the converter will interpret this as ” G4\_BONE\_CORTICAL\_ICRP “
- etc.

### 10.10.7 Schneider

TOPAS provides a built-in converter that follows the most common method used in proton therapy for DICOM or XiO patient data ([PubMed](#)):

- Schneider W, Bortfeld T and Schlegel W. Correlation between CT numbers and tissue parameters needed for Monte Carlo simulations of clinical dose distributions. Phys. Med. Biol. 2000; 45(2):459-78.

This converter follows the technique developed by Schneider to assign materials based on a single CT image file containing Hounsfield Unit (HU) values. It is selected using:

```
s:Ge/Patient/ImagingtoMaterialConverter = "Schneider"
```

The HU conversion parameters are typically stored in a separate parameter file:

```
includeFile = HUtoMaterialSchneider.txt
```

An example of such a HU conversion parameter file is examples/DICOM/HUtoMaterialSchneider.txt.

The first set of parameters in the HU file are used to calculate density:

```
dv:Ge/Patient/DensityCorrection = 3996 9.35212 5.55269 4.14652 ...1.06255 1.00275 g/
→cm3
iv:Ge/Patient/SchneiderHounsfieldUnitSections = 8 -1000 -98 15 23 101 2001 2995 2996
uv:Ge/Patient/SchneiderDensityOffset = 7 0.00121 1.018 1.03 1.003 1.017 2.201 4.54
uv:Ge/Patient/SchneiderDensityFactor = 7 0.00103 0.00089 0.0 0.00117 0.00059 0.0005 0.
→0
uv:Ge/Patient/SchneiderDensityFactorOffset = 7 1000. 0. 1000. 0. 0. -2000. 0.
```

DensityCorrection:

- One value for every possible HU value.
- Values start from Ge/Patient/MinImagingValue which defaults to -1000

SchneiderHounsfieldUnitSections:

- Specifies how to break up the entire set of HU units into several density calculation sections. The HU conversion formula then uses different correction factors for each of these sections.
- The total range (last value minus first value) must equal the number of values in `DensityCorrection`.
- In the above example, the 8 values define 7 sections:
  - Section 1: -1000 to -99
  - Section 2: -98 to 14
  - ...
  - Section 7: 2995 to 2996

`SchneiderDensityOffset`, `SchneiderDensityFactor` and `SchneiderDensityFactorOffset`:

- Must have one value for each of the density calculation sections, so length must be one less than the length of `SchneiderHounsfieldUnitSections`

Thus, for any specific HU number, we can extract the appropriate:

- `DensityCorrection`
- `SchneiderDensityOffset`
- `SchneiderDensityFactor`
- `SchneiderDensityFactorOffset`

And use these in the Schneider formula:

- $\text{Density} = (\text{Offset} + (\text{Factor} * (\text{FactorOffset} + \text{HU}[-1000,2995]))) * \text{DensityCorrection}$

The second set of parameters in the HU file are used to calculate material name and graphics color:

```
iv:Ge/Patient/SchneiderHUToMaterialSections = 26 -1000 -950 -120 -83 ... 1500 2995_
↪2996
sv:Ge/Patient/SchneiderElements = 13 "Hydrogen" "Carbon" "Nitrogen" "Oxygen" ...
uv:Ge/Patient/SchneiderMaterialsWeight1 = 13 0.0 0.0 0.755 0.232 ...
uv:Ge/Patient/SchneiderMaterialsWeight2 = 13 0.103 0.105 0.031 0.749 ...
...
iv:Gr/Color/PatientTissue1 = 3 63 63 63
iv:Gr/Color/PatientTissue2 = 3 100 0 0
...
```

`iv:SchneiderHUToMaterialSections`:

- Specifies how to break up the entire set of HU units into several material name assignment sections.
- The total range (last value minus first value) must equal the number of values in `DensityCorrection`.
- In the above example, the 26 values define 25 material name assignment sections:
  - Section 1: -1000 to -949
  - Section 2: -50 to -119
  - ...
  - Section 25: 2995 to 2996

`sv:SchneiderElements`:

- Specifies all of the elements that will be used in the patient.
- All patient materials must be composed from combinations of this set of elements.

uv:SchneiderMaterialsWeight1 through SchneiderMaterialsWeight25:

- There should be one of these parameters for each of the material name assignment sections. The length of `SchneiderMaterialsWeight` must equal the length of `SchneiderElements`.
- Each value in `SchneiderMaterialsWeight` tells what proportion of the given element in `SchneiderElements` to use in this material.
- In our `SchneiderMaterialsWeight2` parameter, the values: 0.103 0.105 0.031 0.749 mean:
  - 10.3 percent of the first element, Hydrogen
  - 10.5 percent of the second element, Carbon
  - 3.1 percent of the second element, Nitrogen
  - 74.9 percent of the second element, Oxygen
- You may optionally provide this parameter to override the default mean excitation energies of some or all of the materials.
- There should be one value for each material name assignment section.
- To use the default mean excitation energy for a particular material, enter that value as 0.
- For example, the following just overrides defaults for two out of 25 assignment sections:

iv:Gr/Color/PatientTissue1:

- Specifies what colors should be assigned to each of the materials.
- There should be one of these parameters for each of the `SchneiderHUToMaterialSections`.
- The three values specify the Red, Green and Blue components of the color.

Putting it all together, we have now specified density, material name, color and, optionally, mean excitation energy, for each of the Hounsfield numbers in the patient.

You can review the materials definitions that TOPAS created based on your patient file and the HU conversion settings. The following parameter tells TOPAS to dump parameters to a file:

```
Ts/DumpNonDefaultParameters = "True"
```

For each HU number that was used in the patient file, you will see a set of parameters starting with `Ma/PatientTissueFromHU` followed by an HU number. For example, for HU number -295, you may see:

```
Ma/PatientTissueFromHU-295/Component = 9 Hydrogen Carbon Nitrogen Oxygen Phosphorus_
-Sulfur Chlorine Sodium Potassium
Ma/PatientTissueFromHU-295/Fractions = 9 0.103 0.105 0.031 0.749 0.002 0.003 0.003 0.
-0.002 0.002
Ma/PatientTissueFromHU-295/Density = 0.707487 g/cm3
Ma/PatientTissueFromHU-295/DefaultColor = PatientTissue2
```

where you then follow the `DefaultColor` parameter named `PatientTissue2` to see that `Gr/Color/PatientTissue2` is 3 100 0 0 which means a mixture of 100 percent Red, 0 percent green, 0 percent blue.

### 11.1 Introduction

We allow any number of particle sources (zero, one or many) with no limitation on how they can be mixed.

We provide four different types of particle sources, each with many options:

- *Beam Parameterization*
- *Emittance Parameterization*
- *Isotropic*
- *Volumetric*
- *Phase Space*

And you may also write your own entirely new particle source (see *Custom Particle Sources*).

The position of the source is always the center of an associated Geometry Component. This is in keeping with the general TOPAS paradigm that all geometrical information resides in Geometry Components. We know that this may feel odd to experienced Geant4 users who are used to setting beam directions irrespective of any geometry volumes, but the TOPAS paradigm enables sources, components, scorers and even fields to all move together in an internally consistent manner.

The Geometry Component associated with a Particle Source is often a Group Component. Such components have a center position and orientation but no actual shape or extent. The Particle Source is placed at this center position and orientation. If the Component is some other Type, such as a TsBox or TsCylinder, the Particle Source still only takes center position and orientation from this Component. None of the other aspects of the Component, such as the Component's shape or size, have any impact on the Particle Source. So, for example, the shape and size of a Beam source is set by various `BeamPosition` parameters, not by the Component's shape or size.

Some examples place the source at a vacuum window at the entrance to a nozzle. The source then moves as the nozzle moves.

### 11.1.1 Particle Names

Throughout TOPAS, particle names can take the following forms (case does not matter):

- A simple string such as
  - “proton”
- A string describing an ion with arguments Z, A, and optionally Charge, such as:
  - “GenericIon(6,12,6)”
  - “GenericIon(6,12)” - Charge defaults to Z, that is, the ion is fully stripped
  - When used to filter sources, ions must be fully stripped (this is the only kind of ion that Geant4’s primary particle generation supports).
  - When used to filter scoring, ions can have any Charge, and any of the arguments can have wildcard value \* so, for example, “GenericIon(6,\*,\*)” will score any Carbon ion (any A and any Charge).
- An integer PDG ID code, though still contained in a string parameter, such as
  - “11”
  - PDG ID codes are as defined by the [Particle Data Group](#)
- When PDG code has 10 digits and starts with 100, this is passed to Geant4 either as the appropriate Geant4 light ion name (“alpha”, “deuteron”, “He3” or “triton”) or as GenericIon(Z,A) where:
  - Characters 4-6 give Z
  - Characters 7-9 give A
  - Character 10 gives Isomer level (not used)

The full set of known particles depends on the physics you have defined. Here are some common values, with associated PDG codes:

| Particle          | PDG code  |
|-------------------|---|
| “proton”          | “2212”  |
| “neutron”         | “2112”  |
| “e-”              | “11”  |
| “e+”              | “-11”   |
| “gamma”           | “22”  |
| “He3”             | “100002003”   |
| “alpha”           | “100002004”   |
| “deuteron”        | “100001002”   |
| “triton”          | “100001003”   |
| “opticalphoton”   | “0” (PDG group has no code for this particle)                   |
| “geantino”        | “0” (sees transportation processes but no physics, no PDG code) |
| “chargedgeantino” | “0” (same as above but with charge, no PDG code)                |

## 11.2 Beam Sources

By default there is a single source named `Demo` centered on a Component named `BeamPosition` that is placed at one end of the `World`. The beam shape is an Ellipse. Each of these parameters is described in detail below:



```
s:So/Demo/Type = "Beam" # Beam, Isotropic, Emittance or PhaseSpace
s:So/Demo/Component = "BeamPosition"
s:So/Demo/BeamParticle = "proton"
d:So/Demo/BeamEnergy = 169.23 MeV
u:So/Demo/BeamEnergySpread = 0.757504
s:So/Demo/BeamPositionDistribution = "Gaussian" # None, Flat or Gaussian
s:So/Demo/BeamPositionCutoffShape = "Ellipse" # Rectangle or Ellipse (if Flat or
↳Gaussian)
d:So/Demo/BeamPositionCutoffX = 10. cm # X extent of position (if Flat or Gaussian)
d:So/Demo/BeamPositionCutoffY = 10. cm # Y extent of position (if Flat or Gaussian)
d:So/Demo/BeamPositionSpreadX = 0.65 cm # distribution (if Gaussian)
d:So/Demo/BeamPositionSpreadY = 0.65 cm # distribution (if Gaussian)
s:So/Demo/BeamAngularDistribution = "Gaussian" # None, Flat or Gaussian
d:So/Demo/BeamAngularCutoffX = 90. deg # X cutoff of angular distrib (if Flat or
↳Gaussian)
d:So/Demo/BeamAngularCutoffY = 90. deg # Y cutoff of angular distrib (if Flat or
↳Gaussian)
d:So/Demo/BeamAngularSpreadX = 0.0032 rad # X angular distribution (if Gaussian)
d:So/Demo/BeamAngularSpreadY = 0.0032 rad # Y angular distribution (if Gaussian)
i:So/Demo/NumberOfHistoriesInRun = 0
```

Where the default definition of BeamPosition is:

```
s:Ge/BeamPosition/Parent="World"
s:Ge/BeamPosition/Type="Group"
d:Ge/BeamPosition/TransX=0. m
d:Ge/BeamPosition/TransY=0. m
d:Ge/BeamPosition/TransZ= Ge/World/HLZ m
d:Ge/BeamPosition/RotX=180. deg
d:Ge/BeamPosition/RotY=0. deg
d:Ge/BeamPosition/RotZ=0. deg
```

Details on BeamEnergySpread:

- The number is unitless because we find it more convenient generally to speak of the spread in terms of percentage of the mean energy, rather than as an absolute number. We could have chosen either representation, but this one seemed most consistent with what we see from other beam modeling applications.
- This is a standard deviation. So the code we have is:

```
fEnergySpread = BeamEnergySpread * fEnergy / 100.;
p.kEnergy = CLHEP::RandGauss::shoot(fEnergy, fEnergySpread);
```

- So, for example, if you want a spread of 0.2 MeV, and your energy is 153 MeV, set BeamEnergySpread to:

```
0.2 MeV / 153 MeV * 100 = 0.13
```

To run generate histories using this demo source, set its number of histories to some value:

```
So/Demo/NumberOfHistoriesInRun = 10
```

We recommend that you not use So/Demo for any serious work. This demonstration source is just there for simple demonstrations. For any serious work, please define your own source so that you do not just accidentally inherit any of the characteristics of our Demo source. Source characteristics vary greatly from one application to another. There is no meaningful “default” value that we can set for you.

So when you set out on your own work, define a new source name, such as:

```
s:So/MySource/BeamParticle = "proton"
d:So/MySource/BeamEnergy = 200. MeV
i:So/MySource/NumberOfHistoriesInRun = 100
...
```

You can provide an energy spectrum instead of a fixed energy by setting the following to "Discrete" or "Continuous":

```
s:So/MySource/BeamEnergySpectrumType = "Continuous" # Either "None", "Discrete" or
→"Continuous"
```

and providing energies and weights as:

```
dv:So/MySource/BeamEnergySpectrumValues = 3 50. 100. 150. MeV
uv:So/MySource/BeamEnergySpectrumWeights = 3 .20 .60 .20
```

An example is in *Spectrum.txt*.

Any source that has `NumberOfHistoriesInRun` greater than zero will contribute primary particles.

The beam is emitted along the Z axis of the beam's `Component` and may have some spread along the X and Y axes.

For `Type = "Beam"`, the beam shape can be further described by a set of parameters that control the position distribution of the start of the beam:

```
s:So/Demo/BeamPositionDistribution = "Gaussian" # None, Flat or Gaussian
s:So/Demo/BeamPositionCutoffShape = "Ellipse" # Rectangle or Ellipse (if Flat or
→Gaussian)
d:So/Demo/BeamPositionCutoffX = 10. cm # X extent of position (if Flat or Gaussian)
d:So/Demo/BeamPositionCutoffY = 10. cm # Y extent of position (if Flat or Gaussian)
d:So/Demo/BeamPositionSpreadX = 0.65 cm # X standard deviation (used only if Gaussian)
d:So/Demo/BeamPositionSpreadY = 0.65 cm # Y standard deviation (used only if Gaussian)
```

and a set of parameters that control how the beam spreads out from that start position:

```
s:So/Demo/BeamAngularDistribution = "Gaussian" # None, Flat or Gaussian
d:So/Demo/BeamAngularCutoffX = 90. deg # X cutoff of angular distrib (if Flat or
→Gaussian)
d:So/Demo/BeamAngularCutoffY = 90. deg # Y cutoff of angular distrib (if Flat or
→Gaussian)
d:So/Demo/BeamAngularSpreadX = 0.0032 rad # X standard deviation of angular
→distribution (used only if Gaussian)
d:So/Demo/BeamAngularSpreadY = 0.0032 rad # Y standard deviation of angular
→distribution (used only if Gaussian)
```

The `Cutoff` parameter is applied symmetrically.

You will note that for Gaussian beams, the position and angular distribution are controlled both by `Spread` and by `Cutoff` parameters. The `Spread` control the standard deviation of the Gaussian with zero mean (keep in mind that the position and orientation of the source is controlled by the parameter `So//Component`), while the `Cutoff` cut off the tails (which would otherwise be infinite). Inside TOPAS, when the Gaussian formula generates a starting point outside of this cutoff, that starting point is rejected and instead the random function is thrown again until a value is found that is within the specified cutoff.

## 11.3 Emittance Sources

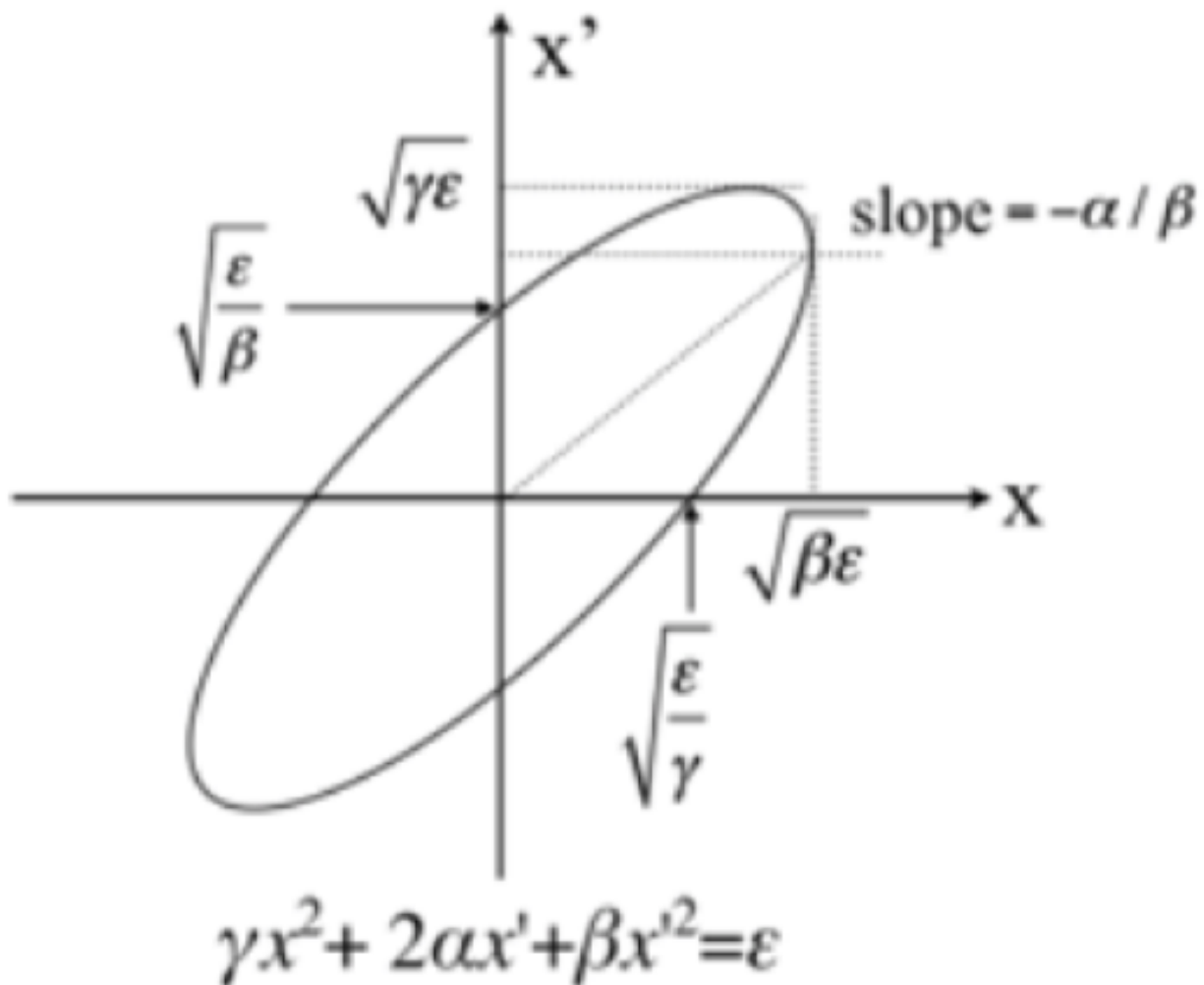
Emittance sources provide 4 ways of sampling particles' position (X and Y) and momentum direction (X' and Y') on a plane. Specify source type as:

```
s:So/MySource/Type = "emittance"
```

To sample particle position and momentum direction from “Gaussian” (bivariate 2D gaussian) per axis (see the *Emittance\_Gaussian.txt* example):

```
s:So/MySource/Distribution = "BiGaussian" # distribution name
d:So/MySource/SigmaX = 0.2 mm # std of x positions
u:So/MySource/SigmaXprime = 0.032 # std of x', note that it's unitless. 1 equals to 1.
  → 0 rad.
u:So/MySource/CorrelationX = -0.9411 # correlation of x and x'
d:So/MySource/SigmaY = 0.2 mm # std of y positions
u:So/MySource/SigmaYprime = 0.032 # std of y'
u:So/MySource/CorrelationY = 0.9411 # correlation of y and y'
```

Alternative distributions are TWISS Gaussian, TWISS Kapchinskij-Vladimirskij (KV), and TWISS Waterbag (based on the Courant-Snyder invariant ellipse in the following figure for X axis only):

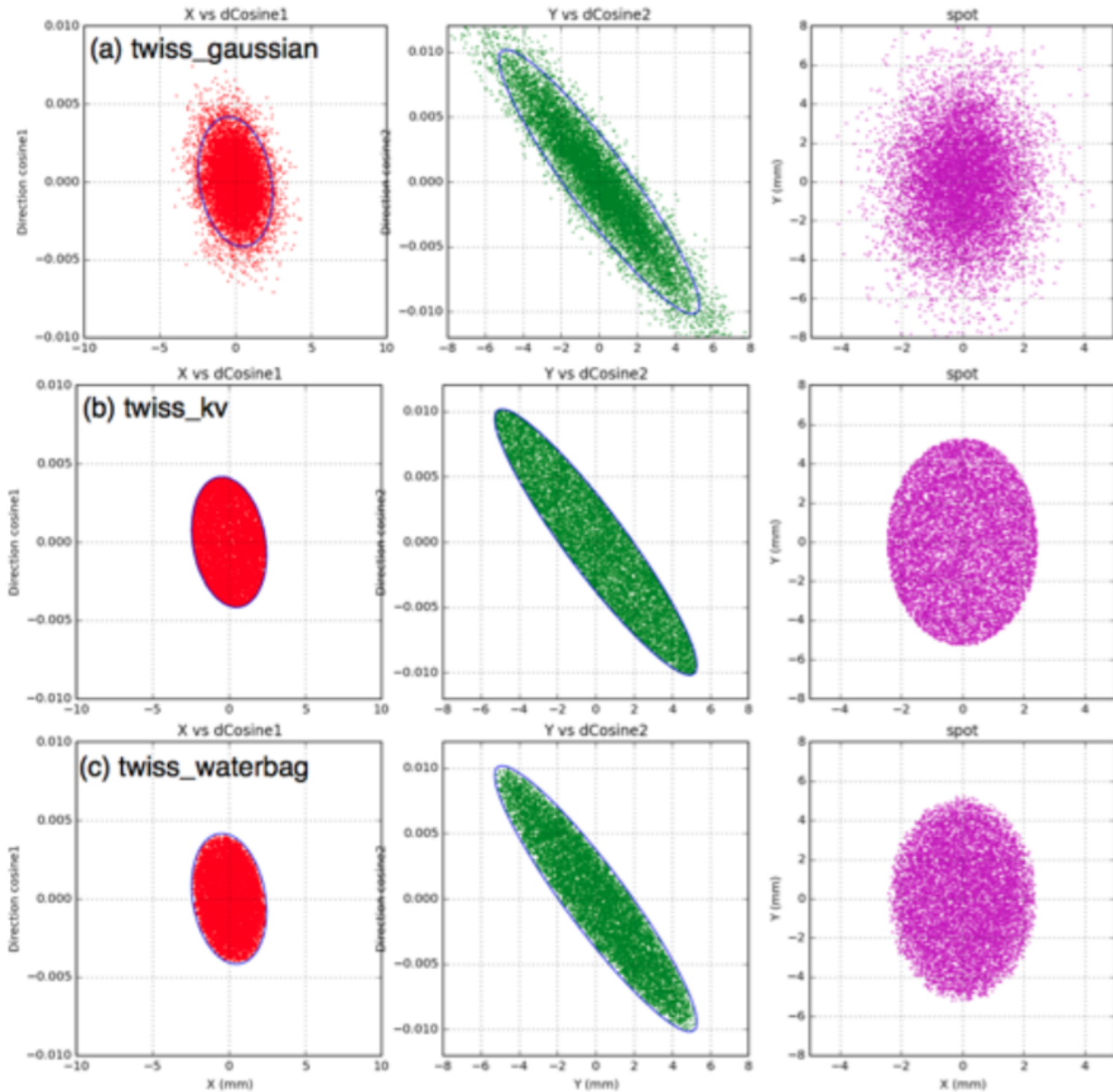


Additional parameters specify the TWISS alpha, beta and gamma as shown in the following from *Emittance\_Twiss.txt*:

```
s:So/MySource/Distribution = "twiss_gaussian" # "twiss_gaussian", "twiss_kv" or
↳ "twiss_waterbag"
u:So/MySource/AlphaX = 0.2
d:So/MySource/BetaX = 600.0 mm
d:So/MySource/EmittanceX = 0.01 mm # we don't multiply pi intrinsically.
u:So/MySource/AlphaY = 2.5
d:So/MySource/BetaY = 1400.0 mm
d:So/MySource/EmittanceY = 0.02 mm
# below for twiss gaussian option only
#0.9 means that 90 % particles will be included in ellipse circle
u:So/MySource/ParticleFractionX = 0.90
u:So/MySource/ParticleFractionY = 0.90
```

The energies and species of the emitted particles can be specified using the same parameters available to the *Beam Sources*.

The following images based on the *Emittance\_Twiss.txt* example show how particle position and momentum directions are sampled and the shape of the beam spots (purple). The blue solid lines in X (red dots) and Y (green dots) represent the Courant-Snyder invariant ellipse. For example, 90% of particles are sampled from the given emittance ellipse in (a).



## 11.4 Isotropic Sources

Isotropic sources emit particles uniformly from the center of the specified Component.

Specify source type as:

```
s:So/MySource/Type = "Isotropic"
```

The energies and species of the emitted particles can be specified using the same parameters available to the *Beam Sources*.

## 11.5 volumetric Sources

volumetric sources emit particles from randomly sampled starting positions from within the radioactive volume of a given component.

This source type has been designed for Brachytherapy applications (though there may be other applications as well).

Specify source type as:

```
s:So/MySource/Type = "Volumetric"
```

And then add an additional required parameter:

```
s:So/*/ActiveMaterial
```

to specify which material within the given component should be considered radioactive.

So, for example, if you have:

```
s:So/Example/Type      = "Volumetric"      s:So/Example/Component      = "ActiveSource"
sc:So/Example/ActiveMaterial = "G4_Ir"
```

particles will start from randomly sampled positions within the Iridium parts of the component named ActiveSource.

See the new example: `examples/Basic/VolumetricSource.txt`

The energies and species of the emitted particles can be specified using the same parameters available to the [Beam Sources](#).

## 11.6 Phase Space Sources

Phase Space refers to the technique of saving or replaying a set of particles crossing a given surface.

- When one saves a phase space, one defines a surface and then saves the position, particle type, energy and momentum of some or all particles crossing that surface.
- When one replays a phase space, one starts a set of particles from the saved positions, with the saved particle types, energy and momentum.

Phase Space enables separating two parts of a simulation or analysis job, and can be used to transfer sets of particles among different codes.

Each phase space must come as two related files (with same file name but different file extensions):

- A .header file tells the number of histories, the number of saved particles and the order of information in the .phsp file
- A .phsp file contains all the details of all the saved particles

We support three formats for Phase Space (and TOPAS automatically figures out the format of your .phsp file by studying the related .header file):

- ASCII provides particle information in a human-readable text file, which data encoded as a series of columns of text. The header file tells the contents and column order per particle.
- Binary provides the same information as ASCII, but in a much more compact format, with data encoded in a stream of bytes. The header file tells the contents and byte order per particle. Use Binary in cases where the ASCII format produces excessively large files.

- Limited is an alternate binary format compatible with some legacy codes. It has fewer options for what data can be expressed, but is compatible with codes such as that used by Varian for their TrueBeam phase space files. Use Limited format only when you need to exchange phase space with legacy codes.

Some users have found legacy phase space files that were unreadable in the Limited format because, though they were supposed to contain information about which particles represent a new history, there was in fact no new history information. In such cases, it seems that all photons were to be considered new histories. To read such files, use the Limited format with the additional TOPAS parameter:

```
b:So/MyPhaseSpaceSource/LimitedAssumePhotonIsNewHistory = "true"
```

Note that while our Phase Space Scorer lets you also write phase space to [ROOT files](#), we do not provide the capability read phase space back in from this format. For more details, see [Phase Space Format](#).

Phase Space sources ignore the parameters starting with “Beam” and instead use:

```
s:So/MySource/Type = "PhaseSpace"
s:So/MySource/Component = "World" # coordinate system of phase space. Usually "World"
s:So/MySource/PhaseSpaceFileName = "ASCIIOutput" # match exact case
```

TOPAS will look for header and phsp files with the given PhaseSpaceFileName.

You can generate some sample data by running any of the examples: [WriteASCII.txt](#), [WriteBinary.txt](#) or [WriteLimited.txt](#).

When using phase space sources, it is important to decide how you want to handle a special case we call “Empty Histories.” Recall that when a phase space is first recorded, for a given Original History, the set of resulting particles that cross the phase space surface:

- may include the primary particles, or
- may include a mix of primary and secondary particles, or
- may include only secondary particles, or
- may include no particles at all. We refer to this last case as an “Empty History.”

When you subsequently use this file as a Phase Space Source, you need to decide how you want TOPAS to handle Empty Histories. If you’re just calculating sums, it doesn’t matter. The Empty Histories contribute nothing to the sum anyway. But if you’re calculating statistical quantities, such as Mean, then these Empty Histories matter. Imagine you want to know the mean dose per Original History. If half of the Original Histories never made it to the phase space file, the decision of whether or not to include these Empty Histories will give a factor of two difference in the calculated Mean Dose per History.

Depending on your use case you may or may not want to include these Empty Histories. It comes down to whether the statistics you want to calculate are:

- per Original History, or
- per Original Histories that Reached Phase Space

You control this with:

```
b:So/MySource/PhaseSpaceIncludeEmptyHistories = "False" # defaults to false
```

TOPAS ASCII and Binary phase space format headers show all of the relevant information:

- Number of Original Histories
- Number of Original Histories that Reached Phase Space
- Number of Scored Particles

Limited phase space format header does not give:

- Number of Original Histories that Reached Phase Space
- so the only way to get that in Limited format is to first read through the entire phsp file and count how many histories contributed there.

TOPAS provides an option to check that the values in the header match what is in the file:

```
b:So/MySource/PhaseSpacePreCheck = "True" # defaults to true
```

For TOPAS ASCII and Binary formats, this is a thorough safety check. It will catch any cases where the files have somehow become corrupted (as could happen, for example, if you are doing a very long phase space writing job and the output disk becomes full during some part of the job). For Limited format, the check is still helpful but less thorough as the header file provides incomplete information. In Limited format, if you want to include Empty Histories, the check is required as it is the only way TOPAS can figure out how many Empty Histories there were.

If the phase space you are replaying came from a TOPAS job, the particle starting positions in that file will have been defined relative to the `World` Component. Set the `Component` parameter above to `"World"`. If you want to offset these particles to some other center or orientation, choose a `Component` that has the new desired center and orientation (reuse some existing `Component`, or define a new `Group Component` just for this purpose). If the phase space you are replaying did not come from TOPAS, there is no automatic way to know what coordinate system was used. It will be up to you to choose a `Component` that has this appropriate coordinate system.

You can optionally tell the phase space source to scale its position information:

```
u:So/MySource/PhaseSpaceScaleXPosBy = 0.1 # adjust starting point on X axis by factor,
↳of 0.1
u:So/MySource/PhaseSpaceScaleYPosBy = 0.1 # adjust starting point on Y axis by factor,
↳of 0.1
u:So/MySource/PhaseSpaceScaleZPosBy = 0.1 # adjust starting point on Z axis by factor,
↳of 0.1
```

You can tell the phase space source to ignore parts of its position information by scaling by zero:

```
u:So/MySource/PhaseSpaceScaleXPosBy = 0.
u:So/MySource/PhaseSpaceScaleYPosBy = 0.
u:So/MySource/PhaseSpaceScaleZPosBy = 0.
```

That coordinate of the particle position then just exactly matches the `Component` center.

You can optionally invert any of the phase space axes by:

```
b:So/MySource/PhaseSpaceInvertXAxis = "True"
b:So/MySource/PhaseSpaceInvertYAxis = "True"
b:So/MySource/PhaseSpaceInvertZAxis = "True"
```

In most cases you will instead want to just rotate the source component. However if the handedness of your source phase space is incorrect, one of these invert options will be necessary.

By default, a `PhaseSpace` source will run all of the histories in the file. To run all of the histories multiple times:

```
i:So/MySource/PhaseSpaceMultipleUse = 2 # reuse this phase space multiple times
```

If you set `PhaseSpaceMultipleUse` to zero, the number of histories in the file will be ignored, and we will instead run the exact number from:

```
i:So/MySource/NumberOfHistoriesInRun
```

This may mean only partial use of the phase space file, or partial reuse to get the right number of histories.



- If your data was generated with time dependence, partial reuse of phase space may not give valid results (you may be playing back only a part of the time sequence). Many more details on controlling number of histories are found in *Time mode*.
- Partial reuse of phase space can not include Empty Histories. There is no statistically valid way to handle these empty histories when the phase space file is only partially used (since one does not know where in the phase space order these Empty Histories would have occurred).

For efficiency, the phase space file will be read in chunks of 10,000 particles at a time. Advanced users may find some reason to adjust this buffer size (though I can't think of any):

```
i:So/MySource/PhaseSpaceBufferSize = 1000000
```

Take care when mixing Phase Space Sources with *Time Features*. While TOPAS can save the current TOPAS time to a phase space file, this time is not automatically applied when reading particles back in from phase space. Thus, if you want to correctly replay source particles that were recorded with time features, it is your responsibility to apply the identical time features during the play back simulation. Some additional notes:

- Do not attempt to change the name of the phase space file over time. Save and replay all particles from a single phase space file.
- Do not use *Random Time Mode*. The randomly generated times during playback will not necessarily match the randomly generated times that were saved to the phase space. Only use *Fixed Time Mode* or *Sequential Time Mode*.
- If your intention is to play back with exactly the same sequence as you had when you generated the phase space file, make sure to set:

```
b:So/MySource/PhaseSpaceIncludeEmptyHistories = "True"
```

otherwise empty histories will put the playback job out of synch with the original job.

A future version of TOPAS will provide more tools to synchronize and check playback time features.

## 11.7 Miscellaneous

### 11.7.1 Additional Control of Number of Histories

Because TOPAS supports both sequential and random time, there are additional parameters that can control the number of histories in random mode. Read *Time mode* before using these parameters:

```
i:So/Demo/NumberOfHistoriesInRandomJob = 100
d:So/Demo/ProbabilityOfUsingAGivenRandomTime = 1.
```

### 11.7.2 Filtering Sources

Optionally filter what comes from the source. This is mainly intended for use with saved PhaseSpace, but is applied uniformly to all sources. Syntax is identical to that used for *filtering in Scorers*.

You may write your own additional filters (see *Custom Filters*).

Filter by Charge. Accepts one or more of "Positive", "Negative" or "Neutral":

```
sv:So/MySource/OnlyIncludeParticlesCharged = 1 "Negative"
sv:So/MySource/OnlyIncludeParticlesNotCharged = 1 "Negative"
```

Filter by Atomic Mass or Number:

```
i:So/MySource/OnlyIncludeParticlesOfAtomicMass = 10 # allow all ions of atomic mass 10
i:So/MySource/OnlyIncludeParticlesNotOfAtomicMass = 10
i:So/MySource/OnlyIncludeParticlesOfAtomicNumber = 6 # allow all ions of Carbon
i:So/MySource/OnlyIncludeParticlesNotOfAtomicNumber = 6
```

Filter by Particle's Initial Kinetic Energy:

```
d:So/MySource/OnlyIncludeParticlesWithInitialKEBelow = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialKENotBelow = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialKE = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialKENot = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialKEAbove = 10. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialKENotAbove = 10. MeV
```

Filter by Particle's Initial Momentum:

```
d:So/MySource/OnlyIncludeParticlesWithInitialMomentumBelow = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialMomentumNotBelow = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialMomentum = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialMomentumNot = 1. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialMomentumAbove = 10. MeV
d:So/MySource/OnlyIncludeParticlesWithInitialMomentumNotAbove = 10. MeV
```

Filter by Particle Name:

```
sv:So/MySource/OnlyIncludeParticlesNamed = 2 "proton" "neutron"
sv:So/MySource/OnlyIncludeParticlesNotNamed = 2 "proton" "neutron"
```

Particle names are as described [here](#).

You may specify more than one filter. For example, to emit protons with initial KE over 100 MeV:

```
sv:So/MySource/OnlyIncludeParticlesNamed = 1 "proton"
d:So/MySource/OnlyIncludeParticlesWithInitialKEAbove = 100. MeV # minimum energy
```

You can invert the results of all previous filters. The following would score only particles that are 1. not protons AND 2. not neutrons AND 3. have initial KE smaller or equal to 100 MeV:

```
sv:So/MySource/OnlyIncludeParticlesNamed = 2 "proton" "neutron"
d:So/MySource/OnlyIncludeParticlesWithInitialKEAbove = 100. MeV # minimum energy
b:So/MySource/InvertFilter = "True"
```

Any filter property can be set by *Time Features* if you wish, to produce time-dependent filtering.

## 12.1 Introduction

In Geant4, physics options are set in pieces of code called “Physics Lists”. A physics list specifies what particles and physics processes are defined, plus various cuts and options. By default, we set TOPAS physics to a list that has been shown to work well for proton therapy research at the Massachusetts General Hospital. This list includes models that handle not only protons but also all secondary particles (neutrons, helium ions, deuterons, tritons, photons, electrons, etc.). The default gives results that closely match a previous custom list that was described in:

- C. Zacharatou Jarlskog and H. Paganetti, “Physics settings for using the Geant4 toolkit in proton therapy,” IEEE Trans. Nucl. Sci. 55, 1018-1025 (2008)

but which can no longer be used since that list corresponded to a much earlier Geant4 release.

The choice of physics list generally depends on the accuracy required for the scored quantities and the speed of the calculation. The Geant4 Medical Simulation Benchmark Group (G4MSBG) has collected a comprehensive set of mainly measured benchmarks that are run at the time of each Geant4 release. Results may be viewed at <https://geant-val.cern.ch/> (see Arce et al, “Report on G4-Med, a Geant4 benchmarking system for medical physics applications developed by the Geant4 Medical Physics Benchmarking Group,” In press, Med Phys, April 2020). Future benchmarks will include the trade-off with speed.

The default list we provide is a Modular physics list. It is specified by the parameters described [here](#).

Advanced users can set their own parameters to override some of these default settings, or can specify entirely different physics lists.

You can choose from two general types of physics lists:

- *Reference Physics Lists* are pre-made, complete lists provided by Geant4.
- *Modular Physics Lists* are lists where you mix and match a set of modules to create a customized complete list.

You can also provide your own physics list using *Custom Physics Lists and Physics Modules* (not recommended unless you have significant Geant4 expertise).

You can get a list of what processes are in your currently selected physics list by:

```
b:Ph/ListProcesses = "True"
```

## 12.2 Modular Physics Lists

The default list we provide is a Modular physics list. It is specified by the parameters described [here](#).

The Geant4 EM physics group recommends against setting `EMRangeMin` too low:

- Set to 100. eV or greater when using standard Geant4 EM physics
- Set to 10. eV or greater when using Geant4-DNA physics

If you want to run with no physics, but only the transportation process (useful for some demos and tests), specify the modules in the following special way:

```
sv:Ph/Default/Modules = 1 "Transportation_Only"
```

If you want to simplify physics so that only basic electromagnetic physics is done (useful for making basic geometry tests run faster), specify just the one module:

```
sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
```

Below is a [List of Available Modules](#) with the corresponding Geant4 class names. Users who are advanced experts in Geant4 physics can also write their own Geant4 physics modules.

The remaining options for the "Geant4\_Modular" physics type are:

```
d:Ph/Default/CutForAllParticles = 0.05 mm # single range cut to use for all particles
d:Ph/Default/CutForGamma = 0.05 mm # overrides CutForAllParticles for Gamma
d:Ph/Default/CutForElectron = 0.05 mm # overrides CutForAllParticles for Electron
d:Ph/Default/CutForPositron = 0.05 mm # overrides CutForAllParticles for Positron
d:Ph/Default/CutForProton = 0.05 mm # overrides CutForAllParticles for Proton
d:Ph/Default/CutForAlpha = 0.05 mm # overrides CutForAllParticles for Alpha
d:Ph/Default/CutForDeuteron = 0.05 mm # overrides CutForAllParticles for Deuteron
d:Ph/Default/CutForTriton = 0.05 mm # overrides CutForAllParticles for Triton
d:Ph/Default/EMRangeMin = 100. eV # minimum for EM tables
d:Ph/Default/EMRangeMax = 500. MeV # maximum for EM tables
i:Ph/Default/EMBins = 77 # number of bins for EM tables
i:Ph/Default/EMBinsPerDecade = 7 # number of bins per decade for EM tables
b:Ph/Default/Fluorescence = "False" # Set to true to turn on Fluorescence
b:Ph/Default/Auger = "False" # Set to true to turn on Auger
b:Ph/Default/AugerCascade = "False" # Set to true to turn on AugerCascade
b:Ph/Default/DeexcitationIgnoreCut = "False" # Set to true to implement_
↳DeexcitationIgnoreCut
b:Ph/Default/PIXE = "False" # Set to true to turn on PIXE
```

### 12.2.1 Physics Regions

By default, cuts affect the entire world, but you can optionally divide the world into several regions and can specify different cuts in each region. First, specify which components belong to a given region:

```
s:Ge/MyComponent/AssignToRegionNamed = "MyRegion"
```

- All children of this component will also be assigned to that region, unless the child has its own `AssignToRegionNamed` parameter.

- There is no requirement that all of the components in a given region be contiguous.

Then assign cuts per region by including the region name in the parameter name as in:

```
d:Ph/Default/ForRegion/MyRegion/CutForGamma = 0.05 mm
d:Ph/Default/ForRegion/MyRegion/CutForElectron = 0.05 mm
d:Ph/Default/ForRegion/MyRegion/CutForPositron = 0.05 mm
d:Ph/Default/ForRegion/MyRegion/CutForProton = 0.05 mm
```

Note: `CutForAllParticles` does not have effect on physics regions. The cuts in physics regions should be specifically assigned to each particle. On the contrary, a cut of 0.05 mm will be assigned for the particles without specified cuts in the named region.

Cuts do not affect all processes, but only those listed below:

- Energy thresholds for gamma are used in Bremsstrahlung
- Energy thresholds for electrons are used in ionization and e+e- pair production processes Energy thresholds for positrons are used in e+e- pair production process
- Energy thresholds for gamma and electrons are used optionally in all discrete processes
  - Photoelectric effect
  - Compton
  - gamma conversion
- Energy thresholds for protons are used in processes of elastic scattering for hadrons and ions defining the threshold for kinetic energy of nuclear recoil

## 12.2.2 List of Available Modules

Users who are advanced experts in Geant4 physics can also write their own Geant4 physics modules and plug these into TOPAS through the Extensions Interface.

| TOPAS Module Name        | Geant4 Class Name                 |
|--------------------------|-----------------------------------|
| g4h-chargeexchange       | G4ChargeExchangePhysics           |
| g4decay                  | G4DecayPhysics                    |
| g4em-dna                 | G4EmDNAPhysics                    |
| g4em-dna_opt1            | G4EmDNAPhysics_option1            |
| g4em-dna_opt2            | G4EmDNAPhysics_option2            |
| g4em-dna_opt3            | G4EmDNAPhysics_option3            |
| g4em-dna_opt4            | G4EmDNAPhysics_option4            |
| g4em-dna_opt5            | G4EmDNAPhysics_option5            |
| g4em-dna_opt6            | G4EmDNAPhysics_option6            |
| g4em-dna_opt7            | G4EmDNAPhysics_option7            |
| g4em-dna_opt8            | G4EmDNAPhysics_option8            |
| g4em-dna-stationary      | G4EmDNAPhysics_stationary         |
| g4em-dna-stationary_opt2 | G4EmDNAPhysics_stationary_option2 |
| g4em-dna-stationary_opt4 | G4EmDNAPhysics_stationary_option4 |
| g4em-dna-stationary_opt6 | G4EmDNAPhysics_stationary_option6 |
| g4em-dna-chemistry       | G4EmDNAChemistry                  |
| g4em-standard_GS         | G4EmStandardPhysicsGS             |
| g4em-standard_SS         | G4EmStandardPhysicsSS             |
| g4em-standard_WVI        | G4EmStandardPhysicsWVI            |

Continued on next page

Table 1 – continued from previous page

| TOPAS Module Name      | Geant4 Class Name             |
|------------------------|-------------------------------|
| g4h-phy_QGSP_BIC_AllHP | G4HadronPhysicsQGSP_BIC_AllHP |
| g4em-extra             | G4EmExtraPhysics              |
| g4em-livermore         | G4EmLivermorePhysics          |
| g4em-polarized         | G4EmLivermorePolarizedPhysics |
| g4em-lowep             | G4EmLowEPPhysics              |
| g4em-penelope          | G4EmPenelopePhysics           |
| g4em-standard_opt0     | G4EmStandardPhysics           |
| g4em-standard_opt1     | G4EMStandardPhysics_option1   |
| g4em-standard_opt2     | G4EMStandardPhysics_option2   |
| g4em-standard_opt3     | G4EMStandardPhysics_option3   |
| g4em-standard_opt4     | G4EMStandardPhysics_option4   |
| g4h-elastic_D          | G4HadronDElasticPhysics       |
| g4h-elastic            | G4HadronElasticPhysics        |
| g4h-elastic_HP         | G4HadronElasticPhysicsHP      |
| g4h-elastic_LEND       | G4HadronElasticPhysicsLEND    |
| g4h-elastic_XS         | G4HadronElasticPhysicsXS      |
| g4h-elastic_H          | G4HadronHElasticPhysics       |
| g4h-inelastic_QBBC     | G4HadronInelasticQBBC         |
| g4h-phy_FTFP_BERT      | HadronPhysicsFTFP_BERT        |
| g4h-phy_FTFP_BERT_HP   | HadronPhysicsFTFP_BERT_HP     |
| g4h-phy_FTFP_BERT_TRV  | HadronPhysicsFTFP_BERT_TRV    |
| g4h-phy_FTF_BIC        | HadronPhysicsFTF_BIC          |
| g4h-phy_QGSP_BERT      | HadronPhysicsQGSP_BERT        |
| g4h-phy_QGSP_BERT_HP   | HadronPhysicsQGSP_BERT_HP     |
| g4h-phy_QGSP_BIC       | HadronPhysicsQGSP_BIC         |
| g4h-phy_QGSP_BIC_HP    | HadronPhysicsQGSP_BIC_HP      |
| g4h-phy_QGSP_FTFP_BERT | HadronPhysicsQGSP_FTFP_BERT   |
| g4h-phy_QGS_BIC        | HadronPhysicsQGS_BIC          |
| g4h-phy_Shielding      | HadronPhysicsShielding        |
| g4ion-binarycascade    | G4IonBinaryCascadePhysics     |
| g4ion-inclxx           | G4IonINCLXXPhysics            |
| g4ion                  | G4IonPhysics                  |
| g4ion-QMD              | G4IonQMDPhysics               |
| g4n-trackingcut        | G4NeutronTrackingCut          |
| g4optical              | G4OpticalPhysics              |
| g4radioactivedecay     | G4RadioactiveDecayPhysics     |
| g4stopping             | G4StoppingPhysics             |

## 12.3 Reference Physics Lists

Reference physics lists are pre-made, complete lists provided by Geant4.

One complication with reference lists is that they do not support use of *Parallel Worlds*. This means that you cannot place components into a parallel world, and, for the *Dividable Components* (TsBox, TsCylinder and TsSphere), you cannot score with a different set of divisions than you have set for the component itself (we handle such complex scoring by creating parallel worlds). TOPAS will give an error if you attempt to use a reference list in a situation where parallel worlds are needed. In such situations, use *Modular Physics Lists*.

The names of the reference physics lists, and their detailed descriptions, are [here](#).

To use a reference physics list, specify the list name in the `Type` parameter, such as:

```
s:Ph/Default/Type = "QGSP_BERT_HP"
```

Reference physics lists allow only one additional option:

```
d:Ph/Default/CutForAllParticles = 0.05 mm # single range cut to use for all particles
```

## 12.4 Optical Physics

### 12.4.1 Optical Photons

A full description of the tracking of optical photons is available in the [Geant4 Physics Reference Manual](#) and the [Geant4 Guide for Applications Developers](#).

TOPAS allows to include optical physics by means of the `g4optical` module in the physics list. The available optical processes included in the `g4optical` module are: scintillation, Cerenkov radiation, wavelength shifting, optical absorption, Rayleigh scattering and boundary processes. However, the optical properties of the material of the volumes must also be defined (at the least the refractive index must to be defined). There exist two types of variables to define the optical properties: a vector based and constant based. The vector-based parameter allows to define a property (refractive index for example) as a function of the photon's energy. While the constant-based parameters allows to define an scalar (scintillation yield for example).

To activate the optical properties in a material one must to set:

```
b:Ma/MyMaterial/EnableOpticalProperties = "True"
```

To set a property based on a vector, one must to define the energy of reference. For example to include the refractive index one must to define two parameters:

```
dv:Ma/MyMaterial/RefractiveIndex/Energies = 3 2.0 2.5 3.0 eV
uv:Ma/MyMaterial/RefractiveIndex/Values = 3 1.58 1.58 1.58
```

To set a property based on a scalar only one parameter is needed, for example:

```
u:Ma/MyMaterial/ScintillationYield = 1120 # in ph/MeV
d:Ma/MyMaterial/FastTimeConstant = 2.1 ns
```

The full list of parameters available is listed in the next table.

| Type | Parameter name                       |
|------|--------------------------------------|
| uv   | RefractiveIndex                      |
| uv   | ImaginaryRefractiveIndex             |
| uv   | RealRefractiveIndex                  |
| dv   | AbsLength                            |
| uv   | FastComponent                        |
| uv   | SlowComponent                        |
| uv   | Miehg                                |
| uv   | SpecularLobeConstant                 |
| uv   | SpecularSpikeConstant                |
| uv   | BackScatterConstant                  |
| uv   | WLSAbsLength                         |
| uv   | WLSComponent                         |
| uv   | Reflectivity                         |
| uv   | Efficiency                           |
| uv   | Transmittance                        |
| u    | ScintillationYield ( in photons/MeV) |
| u    | ResolutionScale                      |
| d    | FastTimeConstant                     |
| d    | SlowTimeConstant                     |
| u    | YieldRatio                           |
| u    | MiehgForward                         |
| u    | MiehgBackward                        |
| u    | MiehgForwardRatio                    |
| u    | WLSTimeConstant                      |
| u    | BirksConstant (in mm/MeV)            |

## 12.4.2 Optical Surfaces

If a perfect smooth interface is between two dielectric materials, the user only needs to provide the refractive index. In all other cases, a surface or optical boundary needs to be defined. There exist two kinds of surfaces: the border surface that delimits the boundary between two components; and the skin surface which surrounds one single component. Border surface is ordered in the sense that the order of the components matters, two border surfaces can exists between a pair of components. Thus, the follow parameters define two surfaces for a pair of components:

```
s:Ge/MyComponent1/OpticalBehaviorTo/MyComponent2 = "MySurface1"  
s:Ge/MyComponent2/OpticalBehaviorTo/MyComponent1 = "MySurface2"
```

For skin surface only one surface can be defined per component:

```
s:Ge/MyComponent1/OpticalBehavior = "MySurface1"
```

Surfaces can be defined as follows (see next table for description):

```
s:Su/MySurfaceName/Type = "dielectric_dielectric" # or dielectric_metal
```

Next, choose the model for optical surfaces:

```
s:Su/MySurfaceName/Model = "Glisur " # Or Unified
```

Finally the finish:



```
s:Su/MySurfaceName/Finish = "Polished"
```

In addition, more detailed properties can be added by parameters described in the table below. In such a case, the way to define would be for example (with prefix Su instead of Ma):

```
dv:Su/MySurfaceName/Energies = 2 1.0 4.0 eV
uv:Su/MySurfaceName/Reflectivity = 2 0.8 0.8
```

| Type     | Parameter name | Possible values   |
|----------|----------------|---|
| string   | Type           | dielectric_dielectric<br>dielectric_metal   |
| string   | Finish         | polished: <i>smooth perfectly polished surface</i><br>polishedfrontpainted: <i>smooth top-layer (front) paint</i><br>polishedbackpainted: <i>same as polished but with a back-paint</i><br>ground: <i>rough surface</i><br>groundfrontpainted: <i>rough top-layer (front) paint</i><br>groundbackpainted: <i>same as ground but with a back-paint</i> |
| string   | Model          | Unified: <a href="#">reference</a><br>Glisur: original GEANT3.21 model  |
| unitless | SigmaAlpha     | Between 0 and 1. By default 0   |

## 12.5 Miscellaneous

### 12.5.1 User-Supplied Physics Lists

See *Custom Physics Lists and Physics Modules* for details on how to provide your own physics list. This option is not recommended unless you have significant Geant4 expertise.

### 12.5.2 Multiple Physics Lists

You can have more than one list defined at the same time, but only the one specified in Ph/ListName will actually be in effect:

```
s:Ph/ListName = "MyList1"
s:Ph/MyList1/Type= "QGSP_BERT_HP" # This list is in effect now
d:Ph/MyList1/CutForAllParticles = 0.05 mm
```

(continues on next page)

(continued from previous page)

```
...
s:Ph/MyList2/Type= "Geant4_Modular" # This list goes into effect if Ph/ListNode set_
→to MyList2
sv:Ph/MyList2/Modules = 1 "g4em-standard_opt3"
d:Ph/MyList2/CutForGamma = 0.04 mm
```

### 12.5.3 Production Thresholds

Production Thresholds and range cuts are discussed in detail in the [Geant4 Application Developers Guide](#). By default, appropriate limits are set by the physics list. You can override these defaults with:

```
d:Ph/MyPhysics/SetProductionCutLowerEdge = 200 eV
d:Ph/MyPhysics/SetProductionCutHighEdge = 30 MeV
```

### 12.5.4 Step Size

The selection of step size is a complex issue in Monte Carlo tracking. Geant4 has its own complex logic for automatically selecting what it thinks will be an appropriate step size, based on local geometry and physics, and the user will not generally need to override this automatic behavior. However, your applications may be sensitive to this behavior, and you may therefore want to set a maximum step size in certain components. In general, larger step sizes give faster performance, but smaller step sizes may give better accuracy.

To limit Geant4's maximum step size in a given component:

```
d:Ge/MyComponent/MaxStepSize = 1. mm # sets maximum step size used in this component
```

Step size settings do not affect other Components placed within this Component. You must explicitly set the step size for any subcomponents that you want to affect.

The choice of maximum step size is highly dependent on your exact simulation problem. If you think you need to set a maximum step size, try running with several values, and pick one for which a small variation up or down does not cause a significant change in results.

## 13.1 Introduction

There are two basic classes of scorers:

- *Volume Scorers* (e.g. Energy or Dose)
- *Surface Scorers* (e.g. Track Count or Phase Space)

Most scorers output overall quantities that are accumulated over many particles (counts and averages), but other scorers can output specific information per particle (in an n-tuple format).

You can have any number of scorers. A scorer is defined when you have a line that ends with `Quantity`, such as:

```
s:Sc/MyScorer/Quantity = "DoseToMedium"
```

When your scoring component is the Parent of other components, you have the option to set scoring to record particles not only in the parent component, but also in its children:

```
b:Sc/MyScorer/PropagateToChildren = "True"
```

This action is recursive to all levels of subcomponents.

You may write your own additional scorers (see *Custom Scorers*).

## 13.2 Volume Scorers

Here are the available volume scorers:

| Quantity              | Description  |
|-----------------------|--|
| AmbientDoseEquivalent | sum of fluence times fluence-to-effective dose conversion coefficients |
| DoseToMedium          | sum of energy deposits divided by mass                                 |
| DoseToWater           | from energy-dependent stopping power conversion (see below)            |
| DoseToMaterial        | from energy-dependent stopping power conversion (see below)            |
| EnergyDeposit         | sum of energy deposits   |
| Fluence               | sum of step lengths divided by volume                                  |
| EnergyFluence         | sum of step lengths times energy divided by volume                     |
| StepCount             | counting number of Geant4 steps in the volume                          |
| OpticalPhotonCount    | fills an ntuple with information about optical photons seen in volume  |
| Charge                | counting method described below  |
| EffectiveCharge       | counting method described below  |
| ProtonLET             | various methods described below  |

Volume Scorers must indicate the relevant Component:

```
s:Sc/MyScorer/Component = "Phantom"
```

When your scoring component is the Parent of other components, you have the option to set scoring to record particles not only in the parent component, but also in its children:

```
b:Sc/MyScorer/PropagateToChildren = "True"
```

This action is recursive to all levels of subcomponents.

For DoseToMaterial, you must also specify the Material:

```
s:Sc/MyScorer/Material = "SomeMaterial"
```

Note that in this case, the material name must exactly match the case defined in Geant4. To check what materials have been defined, add the parameter:

```
i:Ma/Verbosity = 1
```

For AmbientDoseEquivalent, the scoring is performed per single particle:

```
s:Sc/MyScorer/Quantity = "AmbientDoseEquivalent"
s:Sc/MyScorer/Component = "MyDetectorComponent"
s:Sc/MyScorer/GetAmbientDoseEquivalentForParticleNamed = 1 "neutron"
```

The scorer uses a track-length estimator, a variance reduction technique that consists of retrieving the absorbed dose at the scoring regions (voxels) located along the particle path until the point of interaction. That improves the computational efficiency substantially for neutral particles. To that end, the ambient dose equivalent is obtained by folding the incident particle fluence, defined as the particle's track-length divided by the scorer's volume [1], with linearly-interpolated fluence-to-effective dose conversion coefficients. Then, the user needs to provide the corresponding fluence-to-effective dose conversion coefficients for a range of incident energy values using a couple of dimensioned double vectors:

```
dv:Sc/MyScorer/FluenceToDoseConversionEnergies = 30 ... MeV
dv:Sc/MyScorer/FluenceToDoseConversionValues = 30 ... Sv*mm2
```

The example AmbientDoseEquivalent.txt provides a complete example for neutron particles. It uses the fluence-to-effective dose conversion coefficients from reference [2], downloaded from reference [3].

[1] Attix FH, Introduction to radiological physics and radiation dosimetry, 1986 Wiley-VCH, Chapter 1, Section III.D.

[2] Pelliccioni, M. Overview of Fluence-to-Effective Dose and Fluence-to-Ambient Dose Equivalent Conversion Coefficients for High Energy Radiation Calculated Using the FLUKA Code, Radiat. Prot. Dosim. 88(4), 279-297 (2000).

[3] <http://www.fluka.org/fluka.php?id=examples&sub=example4> Accessed on March 10, 2021.

For DoseToWater and DoseToMaterial, we use energy-dependent stopping power conversion as in:

```
dose_to_new_material = dose_to_medium * ( density_of_medium / density_of_new_material_
→) * ( dEdX_in_new_material / dEdX_in_medium )
```

The dEdX comes from the Geant4 EmCalculator utility.

The DoseToWater and DoseToMaterial scorers are somewhat slow since, for every hit, they need to compute stopping power ratios based on the current energy of the particle. You can obtain better speed by adding the option:

```
b:Sc/MyScorer/PreCalculateStoppingPowerRatios = "True" # defaults to "False"
```

- False gives the best accuracy, calculating stopping power on-the-fly for the exact energy.
- True gives the best speed, looking up stopping power from a pre-calculated table binned by energy. It is about 50% faster than the default option for typical patient simulations. The difference in accuracy is not significant for most studies.

For PreCalculateStoppingPowerRatios, the table of stopping power ratios can be tuned by:

```
Sc/MyScorer/ProtonEnergyBinSize # default is 1 MeV
Sc/MyScorer/MinProtonEnergyForStoppingPowerRatio # default is 1 MeV
Sc/MyScorer/MaxProtonEnergyForStoppingPowerRatio # default is 500 MeV
Sc/MyScorer/ElectronEnergyBinSize # default is 1 keV
Sc/MyScorer/MinElectronEnergyForStoppingPowerRatio # default is 1 keV
Sc/MyScorer/MaxElectronEnergyForStoppingPowerRatio # default is 1 MeV
```

For Charge and EffectiveCharge:

- If a particle reaches zero kinetic energy in the scoring volume, its charge is accumulated
- If a particle is generated in the scoring volume, its charge is subtracted
- For Charge, the charge comes from the particle definition's PDGCharge
- For EffectCharge, the charge comes from the PreStepPoint's Charge

### 13.2.1 ProtonLET Scorer

The ProtonLET scorer gives the LET of primary and secondary protons, including the energy deposited by associated secondary electrons. It uses techniques discussed in two recent articles on best practices to score LET in Geant4:

- Phys. Med. Biol. 60 (2015) 2645–2669 by MA Cortes-Giraldo and A Carabe
- Phys. Med. Biol. 60 (2015) N283–N291 by DA Granville and GO Sawakuchi

In particular, we adopt the methods developed by Granville and Sawakuchi. We compute dose-averaged LET, but you may instead request track-averaged:

```
s:Sc/MyScorer/WeightBy = "Track" # defaults to "Dose"
```

By default, the LET is computed by dividing the energy deposited by the step length. Such distributions can feature spurious spikes, caused by events where the step length is severely constrained by a voxel boundary crossing. Three solutions to this issue are provided:

- By default, a step-by-step upper cut-off is set, such that steps contributing greater than this value are not be scored:

```
d:Sc/MyScorer/MaxScoredLET = 100 MeV/mm/(g/cm3) # default 100 MeV/mm/(g/cm3)
```

- Alternatively, you can set the LET computation to look up the electronic stopping power for the pre-step proton energy:

```
b:Sc/MyScorer/UsePreStepLookup = "True" # defaults to "False"
```

- Or you can increase the electron production threshold:

```
d:Ph/Default/CutForElectron = 1 mm # defaults to 0.05 mm
```

The ProtonLET Scorer can give values that are too high in air, where the mean path length between discrete processes can be larger than the voxel size. This can be avoided by neglecting secondary electrons, with:

```
d:Sc/MyScorer/NeglectSecondariesBelowDensity = 0.1 g/cm3
```

Even when you do this, rare events that produce very low energy protons (e.g. a recoiling hydrogen nucleus) will produce spikes in LET. This is also seen in the `PreStepLookup` version of the scorer. They are not seen in the fluence-averaged version of the scorer, since they are rare events. For this reason we introduce the parameter:

```
d:Sc/MyScorer/UseFluenceWeightedBelowDensity = 0. g/cm3
```

We set this to zero by default because it is strange to mix both types of LET in a single distribution, and could be significantly wrong at the end of range. We expect users to want to enable this when making a pretty plot of LET to overlay on a CT scan, without spikes in cavities and outside the patient.

## 13.3 Surface Scorers

Surface Scorer Quantities are:

- SurfaceCurrent
- SurfaceTrackCount
- PhaseSpace

Surface Scorers must indicated the relevant `Component` and `Surface` name:

```
s:Sc/MyScorer/Surface = "Phantom/ZMinusSurface"
```

where the surface name refers to the coordinate system of the `Component`.

When your scoring component is the Parent of other components, you have the option to set scoring to record particles not only in the parent component, but also in its children:

```
b:Sc/MyScorer/PropagateToChildren = "True"
```

The syntax to specify surface depends on which shape component is involved.

- For `TsBox`:
  - `XMinusSurface`
  - `XPlusSurface`
  - `YMinusSurface`

- YPlusSurface
- ZMinusSurface
- ZPlusSurface
- AnySurface
- For TsCylinder:
  - ZMinusSurface
  - ZPlusSurface
  - InnerCurvedSurface
  - OuterCurvedSurface
  - PhiMinusSurface (if cut or divided along Phi)
  - PhiPlusSurface (if cut or divided along Phi)
  - AnySurface
- For TsSphere:
  - InnerCurvedSurface
  - OuterCurvedSurface
  - PhiMinusSurface (if cut or divided along Phi)
  - PhiPlusSurface (if cut or divided along Phi)
  - ThetaMinusSurface (if cut or divided along Theta)
  - ThetaPlusSurface (if cut or divided along Theta)
  - AnySurface
- For any other Component Type, TOPAS does not have a syntax to tell one surface from another, so the only allowable option is:
  - AnySurface

If you are scoring on a *divided component* (TsBox, TsCylinder or TsSphere), all surfaces of the divided component then become sensitive for scoring. So, for example, ZMinusSurface will mean to accumulate hits on every ZMinusSurface of every voxel in the divided TsBox.

Creators of parameter files can pre-define more user-friendly synonyms through relative parameters, such as:

```
s:Ge/WaterTank/Water/UpstreamSurface = Ge/WaterTank/Water/ZMinusSurface
```

so that users can then score using the named Surface, as in:

```
s:Sc/MyScorer/Surface = Ge/WaterTank/Water/UpstreamSurface
```

## 13.4 Phase Space Scorer

Phase Space refers to the technique of saving or replaying a set of particles crossing a given surface. It is the only one of our built-in scorers that saves data to n-tuple format, rather than storing accumulated overall data (counts or averages). However you can write extension scorers to use this generalized n-tuple capability to store other information on a per-particle basis (see *Custom Scorers*).

- When one saves a phase space, one defines a surface and then saves the position, particle type, energy and momentum of some or all particles crossing that surface.
- When one replays a phase space, one starts a set of particles from the saved positions, with the saved particle types, energy and momentum.

Phase Space enables separating two parts of a simulation or analysis job, and can be used to transfer sets of particles among different codes.

If your *Surface Scorer* has `Quantity = "PhaseSpace"`, the output will be a pair of Phase Space files:

- A .header file tells the number of histories, the number of saved particles and the order of information in the .phsp file
- A .phsp file contains all the details of all the saved particles

Note that if your scoring Component is a divided component, such as a voxelized patient, phase space will be scored on the specified surface of Every Voxel (this is the same scoring behavior you will see from any other Surface Scorer in TOPAS).

We support three formats for Phase Space:

- ASCII provides particle information in an easy to read simple text file, which data encoded as a series of columns of text. The header file tells the contents and column order per particle.
- Binary provides the same information as ASCII, but in a much more compact format, with data encoded in a stream of bytes. The header file tells the contents and byte order per particle. Use Binary in cases where the ASCII format produces excessively large files.
- Limited is an alternate binary format compatible with some legacy codes. It has fewer options for what data can be expressed, but is compatible with codes such as that used by Varian for their TrueBeam phase space files. Use Limited format only when you need to exchange phase space with legacy codes.

You can additionally write phase space to *ROOT files*, however there is no corresponding ability to read phase space back in from these files.

You tell TOPAS what format to write out by setting:

```
s:Sc/MyScorer/OutputType = "ASCII" # "Binary", "ASCII", "Limited" or "ROOT"
```

All formats provide at least ten quantities for each scored particle:

- X position
- Y position
- Z position
- U (direction cosine of momentum with respect to X)
- V (direction cosine of momentum with respect to Y)
- Energy in MeV
- Weight
- Particle ID
- Flag to tell if Third Direction Cosine is Negative (1 means true)
- Flag to tell if this is the First Scored Particle from this History (1 means true) (Note that this may or may not be the primary, as the primary may or may not have made it all the way to the scoring plane).

The positions are relative to the center of the *World*.

For the ASCII and Binary formats, you can turn on additional columns of phase space output:



```

b:Sc/MyScorer/IncludeTOPASTime = "True" # Time used by TimeFeatures for this history
b:Sc/MyScorer/IncludeTimeOfFlight = "True" # Time of Flight of this particle from_
↳start of history to scoring plane
b:Sc/MyScorer/IncludeRunID = "True"
b:Sc/MyScorer/IncludeEventID = "True"
b:Sc/MyScorer/IncludeTrackID = "True"
b:Sc/MyScorer/IncludeParentID = "True" # Track ID of parent particle
b:Sc/MyScorer/IncludeCharge = "True"
b:Sc/MyScorer/IncludeCreatorProcess = "True"
b:Sc/MyScorer/IncludeVertexInfo = "True" # Initial KE, Position and Momentum
b:Sc/MyScorer/IncludeSeed = "True"

```

The last of these gives the four variable parts of a random seed. Replaying this random seed will get you the same event back later. The full random seed should be a file of the form:

```

Uvec
1878463799
3
1425618182
1466214412

```

To reuse a saved seed, create a file with the above five lines, replacing the four numeric parts with the four integers in the phase space file. Assuming you name that file `event1.rndm`, you can then make TOPAS start from this random seed by having TOPAS wake up at the Geant4 command line, by using:

```

Ts/PauseBeforeSequence = "True"

```

And then typing:

```

/random/resetEngineFrom event1.rndm
exit

```

The phase space scorer and any custom n-tuple scorers buffer output to avoid excessive disk access. You will not generally need to adjust this buffering value, but can adjust if if you wish:

```

i:Sc/MyScorer/OutputBufferSize = 1000 # Number of particles in phase space buffer

```

### 13.4.1 Phase Space Format

Phase Space refers to the technique of saving or replaying a set of particles crossing a given surface.

- When one saves a phase space, one defines a surface and then saves the position, particle type, energy and momentum of some or all particles crossing that surface.
- When one replays a phase space, one starts a set of particles from the saved positions, with the saved particle types, energy and momentum.

Phase Space enables separating two parts of a simulation or analysis job, and can be used to transfer sets of particles among different codes.

A Phase Space is stored as a pair of related files:

- A `.header` file tells the number of histories, the number of saved particles and the order of information in the `.phsp` file
- A `.phsp` file contains all the details of all the saved particles

We support three formats for Phase Space:

- Binary is a compact format, with data encoded in a stream of bytes. The header file tells the contents and byte order per particle.
- ASCII provides the same information as Binary, but presents it as a much less compact, but easier to read simple text file, which data encoded as a series of columns of text. The header file tells the contents and column order per particle.
- Limited is an alternate binary format compatible with some legacy codes. It has fewer options for what data can be expressed, but is compatible with codes such as that used by Varian for their TrueBeam phase space files.

You can additionally write phase space to [ROOT files](#), however there is no corresponding ability to read phase space back in from these files.

For the Binary and ASCII formats, Particle ID is encoded using the large set of integer codes specified by the Particle Data Group (PDG):

- 22 = photon
- 11 = electron
- -11 = positron
- 2112 = neutron
- 2212 = proton
- Additional codes go all the way up to ten digit ion codes of the form  $\pm 10\text{LZZZAAAI}$ .
- See the [PDG web site](#) for a full explanation

For the Limited format, only a few particle codes are supported, while other particle types are not scored at all (and so this format is only recommended if you need to interface with legacy codes):

- 1 = photon
- 2 = electron
- 3 = positron
- 4 = neutron
- 5 = proton

The Binary and ASCII formats are self-describing, with the complete column or byte order documented in the associated header file. The exact set of columns will depend on which options are used to create the phase space file. Run the [WriteASCII.txt](#) and [WriteBinary.txt](#) examples to see these headers.

If you are attempting to create TOPAS Binary or ASCII phase space from some application other than TOPAS, be advised that the formatting requirements are very specific. It is best to compare your phase space header and phsp files to those produced by the TOPAS examples listed above. Some things to watch out for:

- First line of header has to be exactly as produced by TOPAS, with no extra spaces, tabs, etc.
- Integer values in the ASCII phase space must not contain decimal points

The Limited format uses the following byte order (the format is not self-describing):

| Size    | Quantity   |
|---------|--|
| 1 byte  | Particle ID<br>Absolute value gives the particle code<br>Sign of this value encodes the direction of the 3rd direction cosine                    |
| 4 bytes | Energy<br>Absolute value gives the energy in MeV<br>Sign of this value is set to negative if this is the first scored particle from this history |
| 4 bytes | X position   |
| 4 bytes | Y position   |
| 4 bytes | Z position   |
| 4 bytes | U (direction cosine of momentum with respect to X)   |
| 4 bytes | V (direction cosine of momentum with respect to Y)   |
| 4 bytes | Weight   |

Direction cosines are consistent between Binary, ASCII and Limited formats. Descriptions can be found [on Wikipedia](#) and [on MathWorld](#). Direction cosines U, V and W correspond to direction cosines alpha, beta and gamma on those sites.

## 13.5 Filtering Scorers

You may add filters to limit what is scored.

You may assign more than one filter to the same scorer. When you have more than one filter, they work as an “AND”.

You may write your own additional filters (see [Custom Filters](#)).

Filter by Generation. Accepts either "Primary" or "Secondary":

```
s:Sc/MyScorer/OnlyIncludeParticlesOfGeneration = "Primary"
```

Filter by Charge. Accepts one or more of "Positive", "Negative" or "Neutral":

```
sv:Sc/MyScorer/OnlyIncludeParticlesCharged = 1 "Negative"
sv:Sc/MyScorer/OnlyIncludeParticlesNotCharged = 1 "Negative"
```

Filter by Atomic Mass or Number:

```
i:Sc/MyScorer/OnlyIncludeParticlesOfAtomicMass = 10      # allow all ions of atomic_
↳mass 10
i:Sc/MyScorer/OnlyIncludeParticlesNotOfAtomicMass = 10
i:Sc/MyScorer/OnlyIncludeParticlesOfAtomicNumber = 6     # allow all ions of Carbon
i:Sc/MyScorer/OnlyIncludeParticlesNotOfAtomicNumber = 6
```

Filter by Particle's Initial Kinetic Energy:

```
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKEBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKENotBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKE = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKENot = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKEAbove = 10. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKENotAbove = 10. MeV
```

When designing energy or momentum filters, keep in mind that since no vacuum is perfect in Geant4 (density can be low but cannot be exactly zero), even particles traveling through "Vacuum" will experience some energy loss.

Filter by Particle's Initial Momentum:

```
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialMomentumBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialMomentumNotBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialMomentum = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialMomentumNot = 1. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialMomentumAbove = 10. MeV
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialMomentumNotAbove = 10. MeV
```

Filter by Kinetic Energy of Particle or its Ancestor when it hit the Scoring Component (excludes any particles descended from primaries that originated in the component):

```
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKEBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKENotBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKE = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKENot = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKEAbove = 10. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKENotAbove = 10. MeV
```

Filter by Initial Momentum of Particle or its Ancestor when it hit the Scoring Component (excludes any particles descended from primaries that originated in the component):

```
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleMomentumBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleMomentumNotBelow = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleMomentum = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleMomentumNot = 1. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleMomentumAbove = 10. MeV
d:Sc/MyScorer/OnlyIncludeIfIncidentParticleMomentumNotAbove = 10. MeV
```

Filter by Process that created the particle. Allows one or more process name:

```
sv:Sc/MyScorer/OnlyIncludeParticlesFromProcess = 2 "hIoni" "eBrem"
sv:Sc/MyScorer/OnlyIncludeParticlesNotFromProcess = 2 "hIoni" "eBrem"
```

Filter by Process that created the particle or any of its ancestors:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorFromProcess = 2 "hIoni" "eBrem"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotFromProcess = 2 "hIoni" "eBrem"
```

Filter by Particle Name:

```
sv:Sc/MyScorer/OnlyIncludeParticlesNamed = 2 "proton" "neutron"
sv:Sc/MyScorer/OnlyIncludeParticlesNotNamed = 2 "proton" "neutron"
```

Filter by Particle Name or the name of any of the particle's ancestors. Use this to, for example, score all charge that results from neutrons, even if the final particle is not a neutron:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNamed = 1 "neutron"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotNamed = 1 "neutron"
```

Particle names are as described [here](#).

Filter by Particle's Origin Volume, Component, or Component and Subcomponents:

```
sv:Sc/MyScorer/OnlyIncludeParticlesFromVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeParticlesNotFromVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeParticlesFromComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeParticlesNotFromComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeParticlesFromComponentOrSubComponentsOf = 1 "Nozzle"
sv:Sc/MyScorer/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf = 1 "Nozzle"
```

If you specify multiple Volume or Component names, this is interpreted as an “OR”, not “AND”.

You will see that you can specify a single Volume or a Component.

```
sv:Sc/OnlyIncludeParticlesFromVolume/OnlyIncludeParticlesFromVolume = 2 "World" "Foil"
sv:Sc/OnlyIncludeParticlesFromComponent/OnlyIncludeParticlesFromComponent = 2 "World" "Foil"
```

If the Component is a simple one, like a box or a sphere, then the component only has a single G4Volume, so the two are functionally identical. But if the Component is a more complicated one, like a Range Modulator Wheel, then there are multiple G4Volumes in the Component, and the latter would allow one to do things like tell just which particles interacted in the Wheel's “stop block”.

Filter by Particle or its Ancestor's Origin Volume, Component, or Component and Subcomponents:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorFromVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotFromVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorFromComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotFromComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf = 1
↪ "Nozzle"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf = 1
↪ "Nozzle"
```

Filter by whether Particle Interacted in Volume, Component, or Component and Subcomponents:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleInteractedInVolume = 1 "Propeller20/Leaf"
↪ "
sv:Sc/MyScorer/OnlyIncludeIfParticleNotInteractedInVolume = 1 "Propeller20/
↪ Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleInteractedInComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleNotInteractedInComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf = 1 "Nozzle"
↪ "
sv:Sc/MyScorer/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf = 1
↪ "Nozzle"
```

Filter by Particle or its Ancestor Interacted in Volume, Component, or Component and Subcomponents:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorInteractedInVolume = 1 "Propeller20/Leaf"
↪ "
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume = 1 "Propeller20/
↪ Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorInteractedInComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf =
↪ 1 "Nozzle"
sv:Sc/MyScorer/
↪ OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf (continues on next page)
↪ "
```

(continued from previous page)

Filter by whether Particle Traversed Volume, Component, or Component and Subcomponents:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleTraversedVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleNotTraversedVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleTraversedComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleNotTraversedComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf = 1 "Nozzle"
sv:Sc/MyScorer/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf = 1 "Nozzle"
→"
```

Filter by Particle or its Ancestor Traversed Volume, Component, or Component and Subcomponents:

```
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorTraversedVolume = 1 "Propeller20/Leaf"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotTraversedVolume = 1 "Propeller20/Leaf"
→"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorTraversedComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotTraversedComponent = 1 "Jaws"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf = 1
→"Nozzle"
sv:Sc/MyScorer/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf =
→ 1 "Nozzle"
```

Filter by material:

```
sv:Sc/MyScorer/OnlyIncludeIfInMaterial = 2 "G4_WATER" "Air"
sv:Sc/MyScorer/OnlyIncludeIfNotInMaterial = 2 "G4_WATER" "Air"
```

Note that in this case, the material name must exactly match the case defined in Geant4. To check what materials have been defined, add the parameter:

```
i:Ma/Verbosity = 1
```

Filter on DICOM RT Structure Sets: A structure set is an extra file in the DICOM directory that provides information on structures such as organs, tumors, PTVs, etc. that have been outlined (contoured) in the planing process. The data is stored as a set of polygons, up to one per slice for each contoured structure. TOPAS can color code DICOM components according to this structure information (see *Patient in DICOM Format*) and can filter scoring based on these structures:

```
sv:Sc/MyScorer/OnlyIncludeIfInRTStructure = 2 "R_LUNG" "L_LUNG"
```

If the structure name includes a space, substitute an underscore in the parameter. So, for example, if the structure name is "R LUNG", you should supply the parameter as "R\_LUNG".

The scored value is set to -1 if the given voxel is not in one of the named structures.

For Surface Scorers, you can also filter by whether particle is going "In" or "Out" of scoring surface. Omit this filter to allow either option:

```
s:Sc/MyScorer/OnlyIncludeParticlesGoing = "In"
```

You may specify more than one filter. For example, to score protons with initial KE over 100 MeV:

```
sv:Sc/MyScorer/OnlyIncludeParticlesNamed = 1 "proton"
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKEAbove = 100. MeV # minimum energy
```

You can invert the results of all previous filters by adding the InvertFilter:

```
b:Sc/MyScorer/InvertFilter = "True"
```

The InvertFilter inverts the final result of the full set of other filters. Thus, the following would score only particles that are NOT ( (proton OR neutron) AND have initial KE larger than 100. MeV ). Note the double parenthesis after the NOT. The NOT applies to the overall result from the set of other filters:

```
sv:Sc/MyScorer/OnlyIncludeParticlesNamed = 2 "proton" "neutron"
d:Sc/MyScorer/OnlyIncludeParticlesWithInitialKEAbove = 100. MeV # minimum energy
b:Sc/MyScorer/InvertFilter = "True"
```

Just to be more clear here: the InvertFilter is applied After all the other filters have been evaluated. It inverts the Total result of the other filters (rather than inverting them one at a time).

So, if you have filters:

```
A
B
```

Then the result is to allow particles that pass A AND B.

If you have filters:

```
A
B
Invert
```

Then the result is to allow particles that DO NOT PASS (A AND B) which is logically equivalent to:

```
(NOT A) OR (NOT B)
```

(see how the And was replaced by an Or here)

And, as with any TOPAS parameter statement, it does not matter what order these lines are in the parameter file.

Any filter property can be set by *Time Features* if you wish, to produce time-dependent filtering.

## 13.6 Output Specification

Scored quantities can be output to simple files (csv or binary formats), data files for use in analysis systems (ROOT or XML format) or to a DICOM file. There are also options to directly produce Volume Histograms (such as DVH).

### 13.6.1 Common Parameters

To specify output file name:

```
s:Sc/MyScorer/OutputFile = "myOutputFileName" # if null, use scorer name, e.g.
↳ "MyScorer"
```

Note that this can be more than just a file name - it can include a relative or absolute file path, as in:

```
s:Sc/MyScorer/OutputFile = "../myOutputFileName" # one directory above current_
↳ directory
s:Sc/MyScorer/OutputFile = "~/SomeSubdirectory/myOutputFileName"
d
```

If you are outputting to ROOT or XML, the above only affects the name of a particular histogram within your overall ROOT or XML file. But the name of the overall ROOT or XML file is set by a different parameter:

```
s:Sc/RootFileName = "topas" # name for ROOT output file
s:Sc/XmlFileName = "topas" # name for XML output file
```

To specify output file type for all except the *Phase Space Scorer*:

```
s:Sc/MyScorer/OutputType = "csv" # "csv", "binary", "Root", "Xml" or "DICOM"
```

For binary output of 3D data, such as from scoring in a water phantom or a patient, the following table shows the correspondence between TOPAS divisions and common 3D data viewing applications:

| TOPAS    | fNi (X/R/R)                                    | fNj (Y/Phi/Phi) | fNk (Z/Z/Theta) |
|----------|--|-----------------|-----------------|
| ImageJ   | Width  | Height          | Images          |
| ParaView | X  | Y               | Z               |
| Amide    | X  | Y               | Z               |
| Numpy    | Use the python module <code>topas2numpy</code> |                 |                 |

By default, output will occur just once, after the entire session. But if you are using *Time Features* and wish to have separate output from specific runs:

```
b:Sc/MyScorer/OutputAfterRun = "True" # set True to trigger output of scorer after
→this run
# If this is always set False, or not defined, we just output at the end of the
→session.
# If this is always set True, we output after every run.
```

Additional output control options:

```
b:Sc/MyScorer/OutputToConsole = "True" # control whether output is also dumped to
→console
s:Sc/MyScorer/IfOutputFileAlreadyExists = "Increment" # "Exit", "Overwrite" or
→"Increment"
```

We keep name and type separate in the above so that one can do things like change all output from csv to AIDA by just changing a single parameter (by setting many OutputType parameters equal to a common relative string paramater).

### 13.6.2 DICOM Output

DICOM output is handled through the package *GDCM*, which is pre-built into TOPAS.

DICOM output is in the form of .dcm files that contains DICOM header information (voxel size, spacing, etc.) and then a block of binary image data representing the 3D output. We use the DICOM output format called RTDOSE. Each pixel is represented by a 16 or 32 bit integer.

- 16 bit is the default.
- For 32 bit, specify:

```
b:Sc/MyScorer/DICOMOutput32BitsPerPixel = "True"
```

TOPAS DICOM output will have a TOPAS-specific root UID:

- 1.2.826.0.1.3680043.9.5871.



TOPAS can use information from your DICOM dataset so that scored results can be more easily compared to those from treatment planning systems. Some metadata tags (e.g. Study Instance UID, Frame of Reference UID) are copied from input DICOM (TsDicomPatient) to output DICOM (the scorer), which is important for data provenance:

- The metadata source can be specified by the parameter: `ReferencedDicomPatient`. This is helpful when scoring on a TsBox.
- Otherwise, the metadata is copied from the scorer's Component (if it is a TsDicomPatient)
- Otherwise, the metadata is generated by TOPAS

Other metadata tags (SOP Instance UID, Series Instance UID, Series Description, Manufacturer, Manufacturer's Model Name, Dates and Times) are set appropriately.

It is also possible to set a custom Series Description using the `SeriesDescription` parameter:

```
s:Sc/MyScorer/SeriesDescription = "Custom description here"
```

TOPAS can automatically create a Scoring Grid that exactly matches a provided RTDOSE file in your DICOM dataset. This makes it easier to compare TOPAS results to Treatment Planning System results. See [Patient in DICOM Format](#) for more details.

### 13.6.3 Histogram Output

“Root” and “Xml” will generate histogram files. Specify the binning of the scored quantity as follows:

```
i:Sc/MyScorer/HistogramBins = 100 # number of bins
d:Sc/MyScorer/HistogramMin = 0. MeV # with unit appropriate to scored quantity
d:Sc/MyScorer/HistogramMax = 100. MeV # with unit appropriate to scored quantity
```

Histograms will be either 1D or 2D depending on how the scoring geometry is divided or the energy is binned (see [Binning by Energy](#)).

- If the geometry is undivided and there is no energy binning, a 1D histogram is produced.
- If the geometry is undivided and there is energy binning, the second histogram axes will be energy.
- If the geometry is divided, it can only be divided in one dimension (such as either X, Y or Z for TsBox geometries) and there can be no energy binning. The second histogram axes will be the axes of the geometry division.

All histogram output is combined into a single file, such as `topas.root` or `topas.xml`. The histogram file name can be adjusted by:

```
s:Sc/RootFileName = "topas" # name for ROOT output file
s:Sc/XmlFileName = "topas" # name for XML output file
```

### 13.6.4 DVH Output

Physicists often report the quality of a treatment plan by showing Dose Volume Histograms (DVHs). Such histograms represent what fractional volume of a given structure has received a given Dose.

- In a differential DVH, the bin value indicates what percentage of the structure volume received the given dose.
- In a cumulative DVH, the bin value indicates what percentage of the structure volume received at least the given dose (the zeroth bin will always have a value of 1, since all bins receive at least zero dose).
- If you combine TOPAS DVH options with the filtering option `OnlyIncludeIfInRTStructure` (see [Filtering Scorers](#)), you can generate a DVH for a specific contoured structure (such as DVH to R\_LUNG).

TOPAS can generate a Volume Histogram for any scored quantity, not just Dose. Just set the scorer's report parameter to include either "DifferentialVolumeHistogram" or "CumulativeVolumeHistogram", as in:

```
sv:Sc/DoseAtPhantom/Report = 1 "CumulativeVolumeHistogram"
```

You cannot specify both types of volume histograms in a single scorer, but you can specify other reporting options, such as the following, which will give a basic histogram of "Sum" and "Mean", plus a "CumulativeVolumeHistogram":

```
sv:Sc/DoseAtPhantom/Report = 3 "Sum" "Mean" "CumulativeVolumeHistogram"
```

As with any histogram, you also need to specify HistogramBins, HistogramMin and HistogramMax. For an example, see [DoseVolumeHistogram.txt](#).

If your results has a 1 in the first bin and zero in the other bins, it probably means your HistogramMax was set too high, and thus none of the voxels had enough dose to get beyond the zeroth bin.

## 13.6.5 Phasespace Output

To specify output file type for the phase space scorer:

```
s:Sc/MyScorer/OutputType = "ASCII" # "ASCII" or "Binary"
```

ASCII format has the advantage that it is human-readable text. Binary format has the advantage that it is much more compact, hence suitable for large files.

## 13.7 Miscellaneous

### 13.7.1 Binning in Dividable Components

When scoring in *Dividable Components* (TsBox, TsCylinder or TsSphere), you have many binning options. By default, binning will match the divisions of the volume. So if you have divided the component, the score will be divided in the same manner.

You are also free to specify some other binning.

- In a TsBox, you can specify binning in X, Y and Z:

```
i:Sc/MyScorer/XBins = 512
i:Sc/MyScorer/YBins = 512
i:Sc/MyScorer/ZBins = 256
```

- In a TsCylinder, you can specify binning in R, Phi and Z:

```
i:Sc/MyScorer/RBins = 100
i:Sc/MyScorer/PhiBins = 20
i:Sc/MyScorer/ZBins = 1
```

- In a TsSphere, you can specify binning in R, Phi and Theta:

```
i:Sc/MyScorer/RBins = 20
i:Sc/MyScorer/PhiBins = 20
i:Sc/MyScorer/ThetaBins = 1
```

Behind the scenes, TOPAS uses Geant4's parallel worlds system to support this binning flexibility. When scoring binning is different from the component's natural binning, TOPAS actually scores in a parallel world copy of the component. This is all done automatically.

Because TOPAS is a fully 3D code, letting you design beams to come from any side, bin 0 may be the first bin hit by the beam, but may also be the last bin hit by the beam. So do not be surprised if beam profiles are the opposite of what you might have expected. If it is important to you that bin 0 be the first bin hit, you may need to change your beam position and direction, or rotate your scoring component by 180 degrees.

### 13.7.2 Binning by Energy

Any scorer can be binned by particle energy, by adding the following parameters:

```
i:Sc/MyScorer/EBins = 10 # defaults to 1, that is, un-binned
d:Sc/MyScorer/EBinMin = 0. MeV # defaults to zero
d:Sc/MyScorer/EBinMax = 100. MeV # must be specified if EBins is greater than 1
```

Note that there are several options for what we mean here by “particle energy.”

From our proton therapy dose calculation roots, the energy binning that we do by default is based not on the energy of the final particle at hit deposition time but instead on the incident particle energy. This is the energy of the final scored particle, or its ancestor, when that particle or ancestor was first incident on the scoring volume.

However, users who have been trying to use this feature to get a spectrum instead need the particle's energy at the current step.

So we have now have a parameter to control what kind of Energy we use for this binning.:

```
s:Sc/*/EBinEnergy = "IncidentTrack" # "IncidentTrack", "PreStep" or "DepositedInStep"
```

- “IncidentTrack” is the behavior we have had in the past, the energy that the particle or its ancestor had when it first was incident on the scoring component. This remains the default.
- “PreStep” is the track's energy at the start of the current step.
- “DepositedInStep” is the amount of energy deposited in the current step.

An example shows the effect of the three different choices:

examples/Scoring/EnergyDepositBinnedByEnergy.txt

The output will include two extra bins, one for underflow (energy < EBinMin), one for overflow (energy > EBinMax). And if you have set EBinEnergy to IncidentTrack, there will be one more bin to hold those deposits for which there is no incident track (the primary particle was created already inside the scoring component, so neither it nor any ancestor of it was ever incident upon the scoring component).

### 13.7.3 Binning by Time

Any scorer can be binned by time-of-flight, the elapsed time since the history was generated (in Geant4 this is called “global time”):

```
i:Sc/MyScorer/TimeBins = 10 # defaults to 0, that is, un-binned
d:Sc/MyScorer/TimeBinMin = 0. ns # defaults to zero
d:Sc/MyScorer/TimeBinMax = 100. ns # must be specified if TimeBins is greater than 1
```

The output will include two extra bins, one for underflow (time < TimeBinMin) and one for overflow (time > TimeBinMax). Note that this time-of-flight is not the same as the TOPAS time feature time. To split results based on that TOPAS time, see [Splitting by Time Feature](#).

When radioactive decay is present, some very large times can occur, as decay may be delayed for hours or days. Thus it is not unusual to have some times exceed the `TimeBinMax`. To get an interesting report on what particles and processes exceed `TimeBinMax`, set `Ts/TrackingVerbosity > 0`.

### 13.7.4 Splitting by Time Feature

To split a scorer into separate scorers depending on the current value of any selected *Time Feature*:

```
s:Sc/MyScorer/SplitByTimeFeature = some_time_feature_name
```

If the time feature is a Step function, one split scorer is made for each of the time feature's values. If the time feature is a Continuous function, another parameter is expected to specify split values. This will be either a dimensioned double vector, unitless vector or integer vector, depending on the type of controlling time feature, such as:

```
dv:Sc/DoseAtPhantom/SplitByTimeFeatureValues = 5 0. 90. 180. 270. 360. deg
```

#### Example 1 - Splitting under control of a Step Time Feature

To split up a 4D CT simulation's dose output depending on the CT time slice, where the CT time slice is controlled by:

```
s:Tf/ImageName/Function = "Step"  
sv:Tf/ImageName/Values = 3 "image1" "image2" "image3"
```

The following will make the scorer `DoseAtPhantom` split by current value of `Tf/ImageName/Value`:

```
s:Sc/DoseAtPhantom/SplitByTimeFeature = "ImageName"
```

creating one scorer for each value of `ImageName`:

```
Sc/DoseAtPhantom-image1  
Sc/DoseAtPhantom-image2  
Sc/DoseAtPhantom-image3
```

#### Example 2 - Splitting under control of a Continuous Time Feature

To split up a simulation's dose output depending on the position of a propeller, where the propeller position is controlled by:

```
s:Tf/PropellerRotation/Function = "Linear deg"
```

The following will make `DoseAtPhantom` split by current value of `Tf/PropellerRotation/Value`:

```
s:Sc/DoseAtPhantom/SplitByTimeFeature = "PropellerRotation"  
dv:Sc/DoseAtPhantom/SplitByTimeFeatureValues = 5 0. 90. 180. 270. 360. deg
```

creating one scorer for each defined range of `PropellerRotation`:

```
Sc/DoseAtPhantom-0.-90.deg  
Sc/DoseAtPhantom-90.-180.deg  
Sc/DoseAtPhantom-180.-270.deg  
Sc/DoseAtPhantom-270.-360.deg
```

See the *SplitByTimeFeature.txt* and *DoseTo4DCT.txt* examples.

### 13.7.5 Statistical Information

By default, scorers will report the sum of the scored quantity over all histories, but many additional reporting options are available:

```
sv:Sc/MyScorer/Report = 1 "Sum" # One or more of Sum, Mean, Histories, Count_In_Bin,
↳Second_Moment, Variance, Standard_Deviation, Min, Max
```

Output columns will be in the same order as the values in the Report parameter.

When there is binning by energy or time, and there is more than one Report option (such as "Sum" and "Mean"), the output will be ordered as:

- Sum (underflow), Mean (underflow), Sum (bin 1), Mean (bin 1), Sum (bin 2), Mean (bin 2), etc.

"Histories" is the total number of histories that were simulated while this scorer was active (that is, excludes any histories that were produced when this scorer was gated to inactive).

"Count\_In\_Bin" is the number of histories that contributed to this bin (that is, excludes any histories for which no particles hit this bin).

If only "Sum" is requested, simple accumulation is used. If "Mean", "Second\_Moment", "Variance" or "Standard\_Deviation" is requested, accumulation uses a numerically stable algorithm from: Donald E. Knuth (1998). The Art of Computer Programming, volume 2: Seminumerical Algorithms, 3rd edn., p. 232. Boston: Addison-Wesley:

```
for x in data:
    n = n+1
    delta = x - mean
    mean = mean + delta/n
    M2 = M2 + delta*(x - mean)
sum = n * mean
variance = M2/(n - 1)
standard deviation = sqrt(variance)
```

Note that if your geometry has many divisions (such as the 70M voxels of a 512 x 512 x 256 CT), and you ask for "Mean", "Second\_Moment", "Variance" or "Standard\_Deviation", you will see a speed penalty. This occurs because any bin that has ever been hit will then have to recalculate its mean or second moment to account for the new history (even if the current history doesn't hit this bin).

TOPAS calculates the variance (and hence the standard deviation) associated with the distribution of the quantity of interest (dose, fluence, etc).

- For the standard deviation of the mean value, divide the standard deviation from TOPAS by the square root of the total number of histories.
- For the standard deviation of the sum, multiply the standard deviation from TOPAS by the square root of the total number of histories.

### 13.7.6 Change Component Color Based on Scoring

You can make TOPAS recolor a component during simulation to reflect a scored value. Using this technique, you can, for example, make a box become darker as it accumulates dose. See the [Darkening.txt](#) example.

To activate this feature:

```
s:Sc/EnergyInPhantom/ColorBy = "Sum" # sum, mean, histories, standard_deviation, min,
↳max
```

You must then provide a list of colors, and cutoff values, such as:

```
sv:Sc/EnergyInPhantom/ColorNames = 5 "white" "grey240" "grey220" "grey200" "grey180"  
dv:Sc/EnergyInPhantom/ColorValues = 4 1. 1000 2000 3000 MeV
```

In the above example:

- if the total energy is from 0 to 1, the phantom will be colored "White".
- if the total energy is from 1 to 1000, the phantom will be colored "grey240".
- if the total energy is from 1000 to 2000, the phantom will be colored "grey220".
- etc.

This feature must be used in conjunction with *Time Features*, as the color will only update after each run. And your scorer must be set to output after each run:

```
b:Sc/EnergyInPhantom/OutputAfterRun = "True"
```

This technique does not currently work in the *Dividable Components* (TsBox, TsCylinder and TsSphere). We will add this capability in a future TOPAS release. For now it only works in simple components made of single Geant4 solids.

### 13.7.7 Toggling a Scorer Off and On

To turn off a scorer:

```
b:Sc/MyScorer/Active = "False" # defaults to "True"
```

This feature can be combined with boolean *Time Features* to produce gated scoring. If the scorer skipped any values due to being set inactive at any time, the total number of skipped values is written out at in the scoring summary.

### 13.7.8 Restoring Results from Files

TOPAS provides an option to read back scored values so that you can then redo the scoring output with different options. Set the parameter:

```
Ts/RestoreResultsFromFile = "True" # defaults to "False"
```

With this set, simulation will not be run, but instead the scored values will be restored from the output of previous TOPAS simulations. For each scorer, there must be an appropriate file to read back, specified by name and type:

```
s:Sc/MyScorer1/InputFile = "MySavedFileName" # match exact case  
s:Sc/MyScorer1/InputType = "csv"
```

The file to read back in must contain the appropriate scored quantity, the appropriate binning, and sufficient information to provide the new Report options. So, for example, if you previously scored "Sum" and "Histories", you could now report "Sum", "Mean", "Histories", and a DVH.

This option is particularly handy if you have been using Outcome Modeling. You can run additional Outcome Model calculations, or repeat previous calculations with different model parameters, without having to repeat the full simulation.

This option can also be used to read in binary output and write out csv, or vice versa.

# CHAPTER 14

---

## Graphics

---

You may have zero, one or more graphics windows active at same time:

```
s:Gr/MyGraphic1/Type = "OpenGL" # OpenGL, HepRep, VRML, DAWN, RayTracer, RayTracerX
```

Note that the file-based graphics systems, HepRep, VRML and DAWN may not show any image until at least one history is run. We will revisit this issue when we move to the next Geant4 version.

HepRep files are designed to be viewed in a Java application called HepRApp. Details can be found [here](#).

Note that graphics can be one of the slowest parts of a simulation, so should be disabled if you are running a long simulation. To disable graphics, do one of the following three things:

- Comment out all of the Gr/\*/Type parameters
- Set all Gr/\*/Active to "False"
- Disable graphics entirely, by setting: b:Gr/Enable = "False"

This last option is essential if you want to run on a batch system that does not contain any OpenGL graphics drivers.

File-based graphics systems will also expect a file name:

```
s:Gr/MyGraphic1/FileName = "MyFileName" # Defaults to name of view (which here is _  
→MyGraphic1).
```

Will use this filename plus an \_n where n increments with each refresh. Due to limitations in Geant4, FileName only affects OpenGL and HepRep. For other cases, the file name is a fixed value, g4\_ followed by a file number.

This can be more than just a file name - it can include a relative or absolute file path, as in:

```
s:Gr/MyGraphic1/FileName = "../MyFileName" # one directory above current directory  
s:Gr/MyGraphic1/FileName = "~/SomeSubdirectory/MyFileName"
```

Basic options:

```
b:Gr/MyGraphic1/IncludeGeometry = "True" # defaults to "True"
b:Gr/MyGraphic1/IncludeTrajectories = "True" # defaults to "True"
b:Gr/MyGraphic1/IncludeStepPoints = "True" # Show trajectory step points, defaults to
↪ "False"
```

Colors are defined by specifying their red, green, blue components, each on a scale of 0 to 255, as in:

```
iv:Gr/Color/lightblue = 3 175 255 255
```

You can optionally provide a fourth value to make colors transparent. This value is called Alpha, with 0 being completely transparent (no color at all) and 255 being fully opaque (the default). So for example, to be 50% transparent:

```
iv:Gr/Color/transparentred = 3 255 255 255 126
```

By default, trajectories will be drawn as what Geant4 calls “Smooth Trajectories”, which means they include additional points to make them curve smoothly in a magnetic field. Geant4 does not actually use these “auxiliary points” in its simulation results, they are just present to make visualization in a field look better. In some cases, Geant4 has trouble handling these auxiliary points, and reports:

```
``!!!!!!! Filter: auxiliary points are being memory leaked !!!!!``
```

To work around this, turn off trajectory drawing or tell Geant4 not to making the trajectories smooth:

```
b:Gr/MyGraphic1/UseSmoothTrajectories = "False" # defaults to "True"
```

You can add axes to the display. Axes lines are colored red for X, green for Y, blue for Z:

```
b:Gr/MyGraphic1/IncludeAxes = "True" # defaults to "False"
s:Gr/MyGraphic1/AxesComponent = "World" # Component in which to center the axes.
↪ Defaults to World.
d:Gr/MyGraphic1/AxesSize = 3. m # size of axes
```

Note that on most OpenGL graphics systems, the shadowing on the arrowheads allows you to tell whether a given axis is coming towards or away from you.

You can visualize magnetic fields, with field intensity and direction depicted through a set of arrows:

```
i:Gr/ViewA/MagneticFieldArrowDensity = 10
```

Use with caution. When combined with rotation seems to sometimes cause crashes in polycone drawing (involved in drawing the arrowheads).

By default, graphics views will refresh after every run. But you can change this to show each history individually or to accumulate all histories for the entire session (multiple runs). This applies globally to all graphics views:

```
s:Gr/RefreshEvery = "History" # "History", "Run" or "Session"
```

If parallel worlds are present, by default they will be visible. If you instead want to see only the main world, specify:

```
sv:Gr/MyGraphic1/VisibleWorlds = 1 "World" # "World", "All" or one or more specific
↪ world names
```

To turn off a graphic:

```
b:Gr/MyGraphic1/Active = "False" # defaults to "True"
```

Extra options used by OpenGL:



```

u:Gr/MyGraphic1/Zoom = 2. # increase to zoom in, decrease to zoom out
d:Gr/MyGraphic1/Theta = 45. deg # view angle as in /vis/viewer/set/viewpointThetaPhi
d:Gr/MyGraphic1/Phi = 45. deg # view angle as in /vis/viewer/set/viewpointThetaPhi
u:Gr/MyGraphic1/TransX = 0. # move left or right in the view window
d:Gr/MyGraphic1/TransY = 0. # move up or down in the view window
s:Gr/MyGraphic1/Projection = "Perspective" # Defaults to "Orthogonal"
d:Gr/MyGraphic1/PerspectiveAngle = 10. deg # Increase for stronger perspective effect
i:Gr/MyGraphic1/WindowSizeX = 600
i:Gr/MyGraphic1/WindowSizeY = 600
i:Gr/MyGraphic1/WindowPosX = 0
i:Gr/MyGraphic1/WindowPosY = 0
b:Gr/MyGraphic1/HiddenLineRemovalForGeometry = "False" # Remove hidden lines from
↳wireframe geometries, like Geant4's /vis/viewer/set/hiddenEdge
b:Gr/MyGraphic1/HiddenLineRemovalForTrajectories = "False" # Remove hidden
↳trajectories lines from within geometries, like Geant4's /vis/viewer/set/
↳hiddenMarker

```

You can tell TOPAs to center the graphics on a given component:

```
s:Gr/*/CenterOn = "MyComponent"
```

Topas will then ignore any parameters of the form:

```
Gr//TransX or Gr//TransY.
```

You can scale the image by different amounts in each dimension:

```
uv:Gr/*/Scale = 3 1. 2. 3.
```

would scale by 1 in X, by 2 in Y and by 3 in Z.

Zooming to the Nanometer Scale: With very large Gr/\*/Zoom values, as one might use to zoom into the nano scale, floating point errors in the graphics card lead to discontinuous jumps in the image position. This makes it difficult to center or size the view as one wishes. An easy workaround has been understood. If one simply makes the world invisible, then because the initial graphics extent only includes the nanometer scale components, one doesn't need much zoom. This effect and its solution are demonstrated in the example:

```
examples/Graphics/NanoMeterScaleTest.txt
```

You can set Topas so that for OpenGL views, the view is copied to a file at the end of each run:

```

b:Gr/MyGraphic1/CopyOpenGLToPDF = "True" # save to PDF
b:Gr/MyGraphic1/CopyOpenGLToSVG = "True" # save to Scalable Vector Graphics
b:Gr/MyGraphic1/CopyOpenGLToEPS = "True" # save to Encapsulated PostScript
b:Gr/MyGraphic1/CopyOpenGLToPS = "True" # save to PostScript

```

Some views may result in one of the following warning messages from Geant4 Visualization. These messages are just informational and can be safely ignored. Note that even if you have set Gr/RefreshEvery = "History", the view will only be copied to a file at the end of the run (not per history).

```

"WARNING: Viewpoint direction is very close to the up vector direction.
Consider setting the up vector to obtain definable behavior."

```

```

"G4PhysicalVolumeSearchScene::FindVolume:
Required volume "Phantom3_10x10x1", copy no. 0, found more than once.
This function is not smart enough to distinguish identical physical volumes which
have different parentage. It is tricky to specify in general. This function gives
you access to the first occurrence only."

```

To create movies, Zoom, Theta, Phi, TransX, TransY, Projection and PerspectiveAngle can be controlled by *Time Features*.

## Trajectory Coloring:

```
s:Gr/MyGraphic1/ColorBy = "Charge" # "Charge", "ParticleType", "OriginComponent",
↳ "Energy", "Momentum", "Generation", "CreatorProcess"
```

For ColorBy = "Charge", trajectories default to red, green, blue for negative, neutral and positive. You can override these defaults with:

```
sv:Gr/MyGraphic1/ColorByChargeColors = 3 "blue" "green" "red" # colors for neg,
↳ neutral, pos
```

For ColorBy = "ParticleType", colors are Geant4 defaults:

| Particle Species | Color   |
|------------------|---------|
| gamma            | green   |
| e-               | red     |
| e+               | blue    |
| pi+              | magenta |
| proton           | cyan    |
| neutron          | yellow  |
| other            | gray    |

You can override these settings with (particle names are described [here](#)):

```
sv:Gr/MyGraphic1/ColorByParticleTypeNames = 4 "e-" "gamma" "proton" "neutron" # any
↳ number of particle names
sv:Gr/MyGraphic1/ColorByParticleTypeColors = 4 "red" "green" "blue" "yellow" # for
↳ each particle type above. All other particles will be set to grey.
```

For ColorBy = "OriginVolume", trajectories are grey unless they come from a named volume in:

```
sv:Gr/MyGraphic1/ColorByOriginVolumeNames = 1 "Propeller20/Leaf" # one or more volume
sv:Gr/MyGraphic1/ColorByOriginVolumeColors = 1 "red" # one color for each name above
```

For ColorBy = "OriginComponent", trajectories are grey unless they come from a named component in:

```
sv:Gr/MyGraphic1/ColorByOriginComponentNames = 1 "jaws" # one or more component names
sv:Gr/MyGraphic1/ColorByOriginComponentColors = 1 "red" # one color for each name
↳ above
```

For ColorBy = "ColorByOriginComponentOrSubComponentOf", trajectories are grey unless they come from a named component or any of its subcomponents in:

```
sv:Gr/MyGraphic1/ColorByOriginComponentNames = 1 "Nozzle" # one or more components
sv:Gr/MyGraphic1/ColorByOriginComponentColors = 1 "red" # one color for each name
↳ above
```

For ColorBy = "Energy":

```
dv:Gr/MyGraphic1/ColorByEnergyRanges = 3 1. 4. 8. MeV # limits of energy ranges
sv:Gr/MyGraphic1/ColorByEnergyColors = 4 "red" "green" "blue" "yellow" # one for
↳ every energy interval that is defined by those ranges - one more value than number
↳ of ranges since includes less than first range value and greater than first range
↳ value
```

For ColorBy = "Momentum":

```
dv:Gr/MyGraphic1/ColorByMomentumRanges = 3 1. 4. 8. MeV # limits of momentum ranges
sv:Gr/MyGraphic1/ColorByMomentumColors = 4 "red" "green" "blue" "yellow" # one for
↳every energy interval that is defined by those ranges - one more value than number
↳of ranges since includes less than first range value and greater than first range
↳value
```

For ColorBy = "Generation":

```
sv:Gr/MyGraphic1/ColorByGenerationColors = 2 "red" "green" # colors for primary and
↳secondaries
```

For ColorBy = "CreatorProcess":

```
sv:Gr/MyGraphic1/ColorByCreatorProcessNames = 5 "eBrem" "annihil" "Decay" "eIoni"
↳"hIoni" # one or more process name
sv:Gr/MyGraphic1/ColorByCreatorProcessColors = 5 "red" "green" "blue" "yellow"
↳"magenta" # one for every process name
```

To filter what trajectories will be in the graphics, use similar syntax to that used for *Filtering Scorers* and *Filtering Sources* (applies globally to all graphics views):

```
sv:Gr/OnlyIncludeParticlesNamed = 2 "proton" "neutron" # one or more particle names
sv:Gr/OnlyIncludeParticlesCharged = 1 "negative" # one or more "positive", "negative"
↳or "neutral"
sv:Gr/OnlyIncludeParticlesFromVolume = 1 "Propeller20/Leaf" # one or more volume
sv:Gr/OnlyIncludeParticlesFromComponent = 1 "Jaws" # one or more component
sv:Gr/OnlyIncludeParticlesFromComponentOrSubComponentsOf = 1 "Nozzle" one or more
d:Gr/OnlyIncludeParticlesWithInitialKEBelow = 1. MeV # maximum energy
d:Gr/OnlyIncludeParticlesWithInitialKEAbove = 10. MeV # minimum energy
d:Gr/OnlyIncludeParticlesWithInitialMomentumBelow = 1. MeV # maximum momentum
d:Gr/OnlyIncludeParticlesWithInitialMomentumAbove = 10. MeV # minimum momentum
sv:Gr/OnlyIncludeParticlesFromProcess = 1 "hIoni" # one or more process name
```

Note that the following three filters may cause a crash if the particle origin is at the world boundary:

```
sv:Gr/OnlyIncludeParticlesFromVolume
sv:Gr/OnlyIncludeParticlesFromComponent
sv:Gr/OnlyIncludeParticlesFromComponentOrSubComponentsOf
```

We will study this issue again when we move to the next Geant4 version.

Visualization control for a specific component is done as part of the Ge/ parameters for that component rather than in the Gr/ parameters:

```
s:Ge/MyComponent/Color = "red"
s:Ge/MyComponent/DrawingStyle = "Solid" # "Solid", "Wireframe" or "FullWireFrame".
# FullWireFrame includes drawing of additional edge lines that Geant4 calls "soft
↳edges"
# - on many graphics devices WireFrame and FullWireFrame give the same result
i:Ge/MyComponent/VisSegsPerCircle = 100 # Number of line segments to use to
↳approximate a circle, defaults to 24. Set to a larger number if you want a smoother
↳curve
b:Ge/MyComponent/Invisible = "True" # defaults to False meaning visible
```

We sometimes see error messages from visualization of the following form:

```
G4PhysicalVolumeSearchScene::FindVolume:
Required volume "PhantomCentralDose_1x1x40", copy no. 0, found more than once...
```

Such messages can be ignored. They do not affect the simulation results. We will revisit how to solve these error messages once we move to the next Geant4 version.

# CHAPTER 15

---

## Time Features

---

While the repeatability requirements of the TOPAS parameter system require that parameter definitions be well specified, there is still a need to define time-dependent behaviors (such as motion, beam current modulation, starting and stopping of scoring activities). The TOPAS Time Feature system allows such time-dependence to be specified in a manner that is both flexible and repeatable.

A Time Feature is a set of parameters that ultimately describes the change of a time feature `Value`. You provide parameters that define the time function, such as a linear change over time. TOPAS automatically creates a `Value` parameter for this function (a parameter you don't define). TOPAS continually updates this `Value` parameter to the appropriate value for a given time.

---

**Note:** If you're doing complex things with parameter file chains, you may want to know where in chain this automatically added `Tf/.../Value` parameter goes: the answer is that it goes into the same virtual file as the `Tf/.../Function` parameter.

---

In addition to specifying the time features, you need to specify the overall time sequence for *Sequential Time Mode*.

### First example

Here is an example, a Time Feature called `ArmRot` that describes a constant rotation:

```
s:Tf/ArmRot/Function = "Linear deg"
d:Tf/ArmRot/Rate = 2. deg/ms
d:Tf/ArmRot/StartValue = 0.0 deg
d:Tf/ArmRot/RepetitionInterval = 50. ms
```

TOPAS automatically creates another parameter:

```
s:Tf/ArmRot/Value
```

and updates this parameter to the appropriate value for a given time.

You can then use this value to affect a component position through a statement such as:

```
d:Ge/Arm/RotX = 0. deg + Tf/ArmRot/Value
```

## 15.1 Linear, Sine, Cosine and Sqrt Functions

For **Dimensioned Double** or **Unitless** values, the `Function` can be any one of:

| Function | Value(Time)   |
|----------|---|
| Linear   | $\text{StartValue} + \text{Rate} * \text{Time}$                   |
| Sine     | $\text{Sine} ( \text{StartValue} + \text{Rate} * \text{Time} )$   |
| Cosine   | $\text{Cosine} ( \text{StartValue} + \text{Rate} * \text{Time} )$ |
| Sqrt     | $\text{Sqrt} ( \text{StartValue} + \text{Rate} * \text{Time} )$   |

If the value is **Dimensioned Double**, you must also provide a unit, such as the `deg` in:

```
s:Tf/ArmRot/Function = "Linear deg"
```

You must provide appropriate `StartValue` and `Rate` parameters, such as:

```
d:Tf/ArmRot/Rate = 2. deg/ms
d:Tf/ArmRot/StartValue = 0.0 deg
```

You must also provide a `RepetitionInterval`, the time interval after which the function will return to the `StartValue`.

## 15.2 Step Function

With a Step function, you can control **any** type of parameter value. You define a set of times at which to change value, and a value for each of those times. The first value you provide specifies the starting value (the value at time zero).

Here is an example of a Step time feature that controls a String:

```
s:Tf/ImageName/Function = "Step"
dv:Tf/ImageName/Times = 3 10 20 30 ms
sv:Tf/ImageName/Values = 3 "lung-1" "lung-2" "lung-3"
```

- The first value is used for times `Tf/TimelineStart` to 10 ms.
- If `Tf/TimelineStart` was not specified, it defaults to 0, so that first value will be used for times 0 to 10 ms.
- The second value is used for times 10 to 20 ms.
- The third value is used for times 20 to 30 ms.
- After 30 ms, the value cycles back to the first value.

Note that the first member of `Values` is not the time that the first `Value` should be used. It is the time that the first Change of value should be made. The first `Value` will be used for the time between `Tf/TimelineStart` and that first member of `Tf/*/Times`.

Note that whereas continuous functions (Linear, Sine, Cosine and Sqrt) include a `RepetitionInterval`, Step Functions do not. They just cycle back to the first `Value` after the last of the `Times` is reached.

Here is an example of a Step time feature that controls a Boolean:

```
s:Tf/ScoringOnOff/Function="Step"
dv:Tf/ScoringOnOff/Times =10 10 20 30 40 50 60 70 80 90 100 ms
bv:Tf/ScoringOnOff/Values=10 "true" "false" "true" "false" "false" "true" "true" "true"
↪ " "false" "true"
```

Note that:

- Tf/.../Times is always of type dv: and has unit of time.
- Tf/.../Values is a vector of whatever type the function controls.

Any individual member of the Values parameter vector can itself be a parameter, such as:

```
bv:Tf/ScoringOnOff/Values=4 "true" "false" Some_Other_Boolean_Parameter_Name "false"
```

## 15.3 Combining Time Features for Complex Behaviors

You can add or multiply time feature Value parameters just as you can add or multiply any other kind of parameter. For example, here is how the number of histories in a run can be controlled by both a beam current and a beam weight:

```
s:Tf/BeamCurrent/Function = "Step"
dv:Tf/BeamCurrent/Times = 1 10 ms
iv:Tf/BeamCurrent/Values = 1 10

s:Tf/BeamWeight/Function = "Step"
dv:Tf/BeamWeight/Times = 10 1 2 3 4 5 6 7 8 9 10 ms
iv:Tf/BeamWeight/Values = 10 1 1 1 2 2 2 2 4 4 4

i:Tf/BCM/Value = Tf/BeamWeight/Value * Tf/BeamCurrent/Value
i:So/MySource/NumberOfHistoriesInRun = Tf/BCM/Value
```

By combining Step time features with other time features, you can control complex sequences. The following from *PurgingMagnet\_move.txt* moves a box first in one direction and then in the other:

```
s:Tf/BackForward/Function = "Step"
dv:Tf/BackForward/Times = 2 100.0 200.0 ms
dv:Tf/BackForward/Values = 2 Tf/BackStep/Value Tf/ForwardStep/Value mm

s:Tf/BackStep/Function = "Linear mm"
d:Tf/BackStep/Rate = 3 mm/ms
d:Tf/BackStep/StartValue = 0.0 mm
d:Tf/BackStep/RepetitionInterval = 100.0 ms

s:Tf/ForwardStep/Function = "Linear mm"
d:Tf/ForwardStep/Rate = -3 mm/ms
d:Tf/ForwardStep/StartValue = 300.0 mm
d:Tf/ForwardStep/RepetitionInterval = 100.0 ms
```

Some complex examples of time features are in examples/Nozzle. While we have had examples of double scattering and pencil beam scanning for some time, those examples have included proprietary IBA information, so could not be generally shared. The examples found in examples/Nozzle have no vendor confidential information.

| Example                                    | Description  |
|--|--|
| <i>RasterScanningPattern.txt</i>           | Time Features for controlling the dipole magnets are implemented. The time varying magnet will scan rectangle fields in a raster pattern.  |
| <i>ScanningStationaryTarget.txt</i>        | In addition to RasterScanningPattern.txt, a water phantom including a plane target is added.   |
| <i>Scanning-TargetMovingHorizontal.txt</i> | The perpendicularly moving target is defined. In order to make protons follow the moving target, compensated Time Features for the dipole magnets are implemented. The execution of this file will show the moving target in horizontal direction and the proton beams tracking the moving target. |
| <i>Scanning-Target-MovingIn-Depth.txt</i>  | To trace the target moving along with the depth, the changes of proton's incident energy should be synchronized with the motion.   |
| <i>Scanning-Nozzle.txt</i>                 | All geometry for the scanning nozzle is defined. The nozzle consists of magnet systems, for example, two quadrupole magnets and two dipole magnets in helium gas filled beam pipe and various monitoring chambers. Magnet fields are set to zero in this parameter file.                           |
| <i>Scattering-Nozzle.txt</i>               | All geometry for the scattering nozzle is defined.   |
| <i>ScatteringNozzle_run.txt</i>            | Range Modulator Wheel rotates over time and scatterers move in and out of the beam.  |

**Warning:** Take care when mixing Phase Space Sources with Time Features. While TOPAS can save the current TOPAS time to a phase space file, this time is not automatically applied when reading particles back in from phase space. Thus, if you want to correct replay source particles that were recorded with time features, it is your responsibility to apply the identical time features during the play back simulation. Some additional notes:

- Do not attempt to change the name of the phase space file over time. Save and replay all particles from a single phase space file.
- Do not use *Random Time Mode*. The randomly generated times during playback will not necessarily match the randomly generated times that were saved to the phase space. Only use *Fixed Time Mode* or *Sequential Time Mode*.

A future version of TOPAS will provide more tools to synchronize and check playback time features.



### 16.1 Introduction

Variance reduction (VR) techniques in TOPAS can be combined to create a very sophisticated setup. However, please note that VR is highly dependent on your specific geometry. Approach these features with caution and test all variance reduced setups against an equivalent setup without variance reduction.

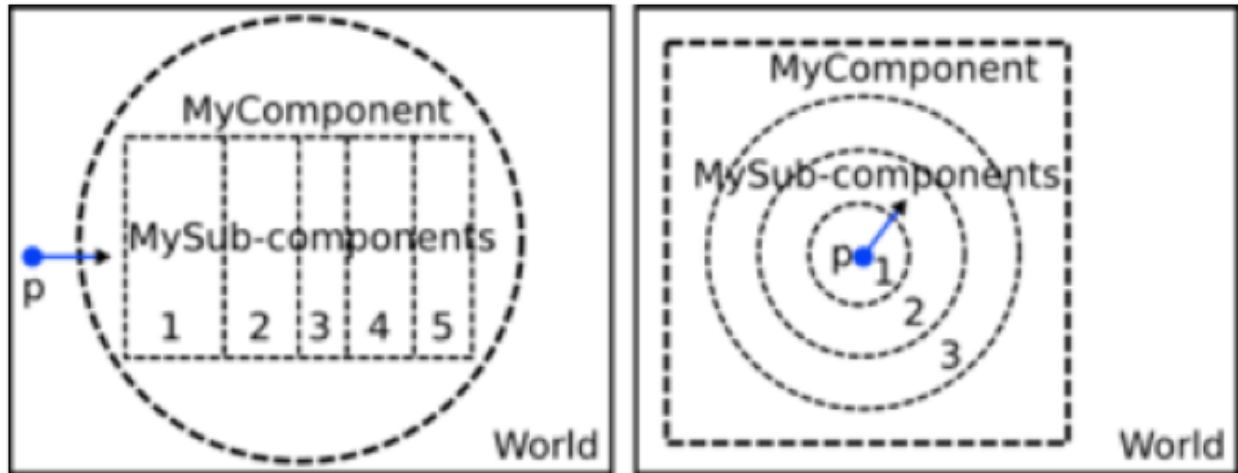
You should also review the Geant4 document that describes which cases are problematic [here](#).

To enable the particle split applied to protons:

```
b:Vr/UseVarianceReduction = "true"  
b:Vr/ParticleSplit/Active = "true"  
sv:Vr/ParticleSplit/ParticleName = 1 "proton"
```

#### 16.1.1 Specify the Split Geometry

The geometry for variance reduction must be in a *parallel world*. The type of component can be any standard solid (*Dividable Components* or *Generic Components*). The geometry must consist of a geometry component with a set of geometry sub-components as daughters. The sub-components must be located in such a way that the boundaries coincide. The split process or Russian roulette will occur at these boundaries. In the next figure a simple scheme is shown.



*Time Features* can be used to move or rotate the component or sub-components. But there is a restriction: the implementation of VR does not allow you to change the dimensions of the component and sub-components.

To set the geometry for VR:

```
s:Vr/ParticleSplit/Component = "MyComponent"
sv:Vr/ParticleSplit/SubComponents = n "MySubComp_1" ... "MySubComp_n"
```

### 16.1.2 Define the Splitting Technique

There are three variance reduction techniques available:

- *GeometricalParticleSplit*
- *ImportanceSampling*
- *WeightWindow*

To choose a technique, use for example:

```
s:Vr/ParticleSplit/Type = "GeometricalParticleSplit"
```

## 16.2 Geometrical Particle Splitting

TOPAS variance reduction is further described in:

Ramos-Mendez et al, "Geometrical splitting technique to improve the computational efficiency in Monte Carlo calculations for proton therapy," Med. Phys. 40, 041718 (2013)

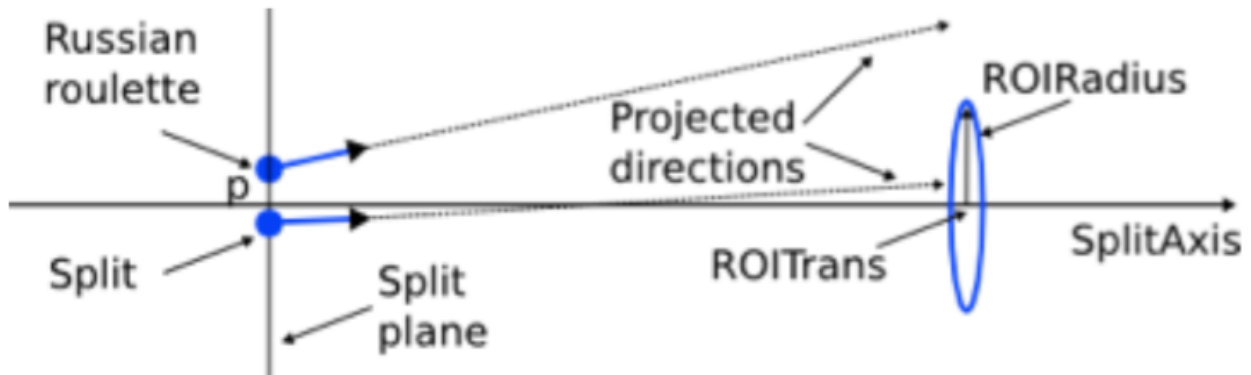
This technique was designed for heavy charged particles. In this implementation, you must specify whether the beam entering into the sub-component has cylindrical symmetry or not. This is because the particles may or may not be randomly redistributed around the `SplitAxis`.

The Russian roulette is applied in a particular direction. That is, at the split plane and prior to being split, the particle is subject to the Russian roulette if its direction does not point towards a Region of Interest (ROI). Then the radius of the ROI and its position on the `SplitAxis` must be defined too. Further, the Russian roulette can be turned on/off at specific surfaces between sub-components.

```
s:Vr/ParticleSplit/Type = "GeometricalParticleSplit"
s:Vr/ParticleSplit/SplitAxis = "zaxis"
d:Vr/ParticleSplit/RussianRoulette/ROIRadius = 7.8 cm
d:Vr/ParticleSplit/RussianRoulette/ROITrans = 10 cm
bv:Vr/ParticleSplit/RussianRoulette = 2 "false" "true"
```

To set whether the region at each sub-component is symmetric or not and to define the corresponding split number:

```
bv:Vr/ParticleSplit/Symmetric = 2 "false" "true"
uv:Vr/ParticleSplit/SplitNumber = 2 8 8
```



In addition for this technique, geometrical Russian roulette will be played if a particle leaves the component or the world in a scheme similar to the *Importance Sampling* technique.

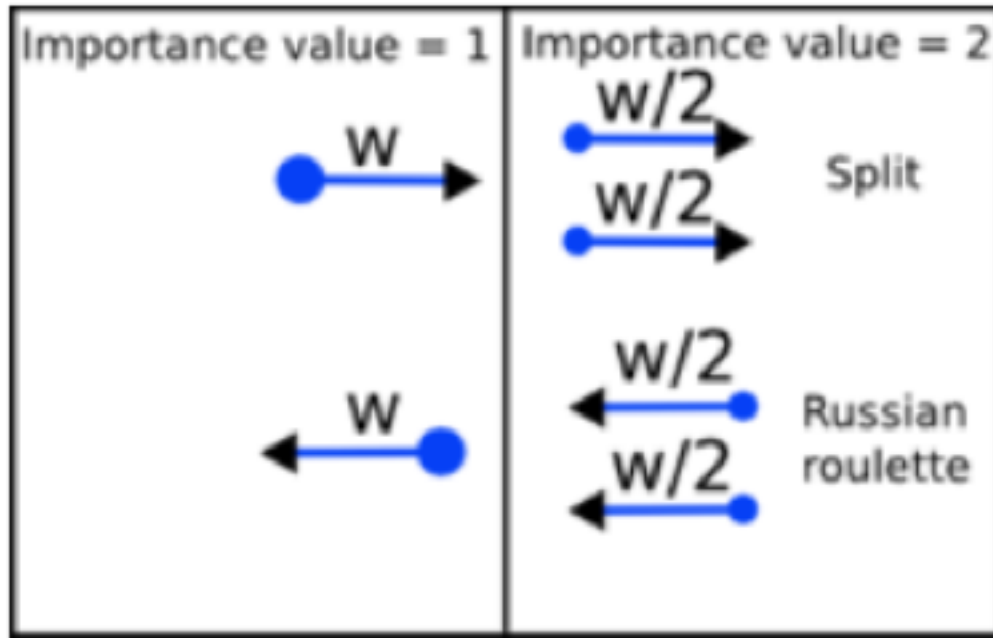
## 16.3 Importance Sampling

In this technique, an importance value is assigned to each sub-component. If a particle is transported into a sub-component with a higher importance, then the particle is split. If it is transported into a sub-component with a lower importance, then Russian roulette is played. By default an importance value of 1 is automatically assigned to the parent component and to the world.

**Warning:** It is desirable for the thickness of the sub-components to be similar to the mean free path of the physical process to be biased.

The sub-component importance values are defined by hand. For example, to split the particles by a factor of 2 between subsequent sub-components, one must to define:

```
s:Vr/ParticleSplit/Type = "ImportanceSampling"
uv:Vr/ParticleSplit/ImportanceValues = 5 1 2 4 8 16
```



## 16.4 Weight Window

In this technique, the split process or Russian roulette will be applied depending on the statistical weight of the particle. Every time that a particle crosses from a sub-component to the next one, the statistical weight is evaluated.

- Particles with weights greater than a lower bound and smaller than an upper bound will be tracked normally.
- Particles with weights smaller than a lower bound will be subject to Russian roulette. If it survives, the particle is tracked normally but its weight is increased.
- Particles with weights greater than an upper bound will be split, and the new particles will be assigned lower weights.

The split number and Russian roulette criteria are internally calculated from an energy map, a weight map, an upper limit factor and a survival factor. In simple geometries the maps can be input by hand.

The user must provide a double vector with upper energy bounds and a unitless vector with lower weight bounds for every sub-component: `WeightMap` and `EnergyMap`. The inverse of a parameter named `MaximumSplitNumber` (100 by default) is used to specify the minimum survival probability to be used in Russian roulette. The parameter `PlaceOfAction` states whether the split process (or Russian roulette) will occur at the sub-component boundaries, at physics interactions or at both.

The follow configuration is equivalent to importance sampling with importance generator of 2:

```
s:Vr/ParticleSplit/Type = "WeightWindow"
uv:Vr/ParticleSplit/WeightMap = 4 1. 1. 0.125 0.0615
dv:Vr/ParticleSplit/EnergyMap = 4 1. 1. 1. 1. GeV
u:Vr/ParticleSplit/UpperLimitFactor = 1
u:Vr/ParticleSplit/SurvivalFactor = 1
i:Vr/ParticleSplit/MaximumSplitNumber = 100
s:Vr/ParticleSplit/PlaceOfAction = "onBoundary"
#Others options of PlaceOfAction: "OnCollision" and "OnBoundaryAndCollision"
```



## 16.5 Tracking Only Specific Particles

In this option the particles are eliminated just after they were created. The user can choose which particles will be tracked in all components. Nevertheless, the user can specify if a component is going to be neglected. That is, all particles are tracked in such components. This option can be useful when the contribution of certain particles is negligible to the final scored quantity. But it must to be used with caution. This option is not a variance reduction.

To eliminate particles other than protons and electrons in all components but in component named `WaterPhantom`:

```
b:Vr/KillOtherParticles/Active = "True"
sv:Vr/KillOtherParticles/HaveNoEffectInComponentsNamed = 1 "WaterPhantom"
sv:Vr/KillOtherParticles/OnlyTrackParticlesNamed = 2 "proton" "e-"
```

## 16.6 Secondary Biasing

The split of secondary particles created after an electromagnetic interaction is also supported. A common example is the split of secondary photons created in the bremsstrahlung process for conventional radiotherapy simulations.

This variance reduction works per electromagnetic physical process per region. *Physics Regions* allow multiple components to have specific production cuts. This is useful in complex geometry setups to improve the computational speed by assigning high production cuts in regions where detailed simulation is unimportant. To assign a region to a component:

```
s:Ge/MyComponent/AssignToRegionNamed = "MyRegion"
```

The region `MyRegion` is automatically created if it does not exist. The next step is to set the secondary biasing option:

```
s:Vr/ParticleSplit/Type = "SecondaryBiasing"
```

Then three vectors must be defined. One with the name of the electromagnetic processes, one with the split number for each process and one with the maximum energies for each processes. The biased particles with energies larger than these values are subject to Russian roulette:

```
sv:Vr/ParticleSplit/ForRegion/MyRegion/ProcessesNamed = 2 "eBrem" "compt"
uv:Vr/ParticleSplit/ForRegion/MyRegion/SplitNumber = 2 100 10
dv:Vr/ParticleSplit/ForRegion/MyRegion/MaximumEnergy = 2 6.0 0.511 MeV
```

If suitable, further CPU time can be saved with a directional Russian roulette for secondary particles created with split (analogous to *Geometrical Particle Splitting*). For that, a new VRT must be set:

```
s:Vr/DirectionalFilter/Type = "DirectionalRussianRoulette"
```

Later, a reference component must be chosen:

```
s:Vr/DirectionalFilter/ReferenceComponent = "Target"
```

And the directional filter is applied:

```
dv:Vr/DirectionalFilter/ForRegion/MyRegion/DirectionalSplitLimits = 2 1.0 1.0 m
dv:Vr/DirectionalFilter/ForRegion/MyRegion/DirectionalSplitRadius = 2 5.0 5.0 cm
sv:vr/DirectionalFilter/ForRegion/MyRegion/processesNamed = 2 "eBrem" "compt"
```

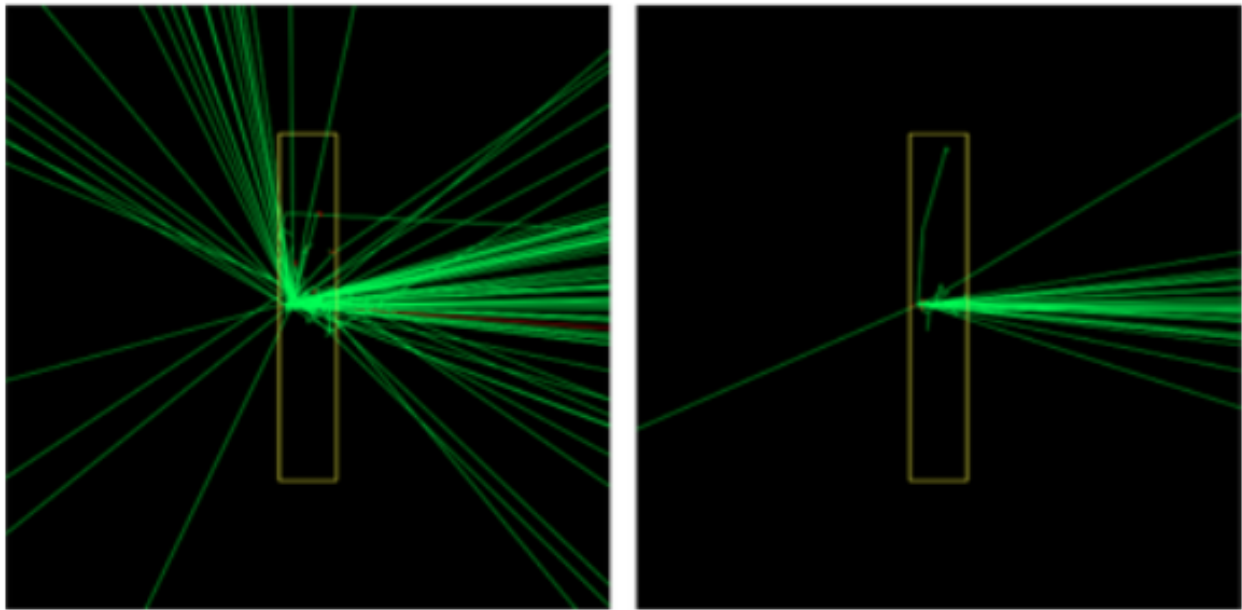


Figure: Biasing particle of secondary photons after a bremsstrahlung process. On the left, no directional Russian roulette is applied. On the right, a directional Russian roulette is applied.

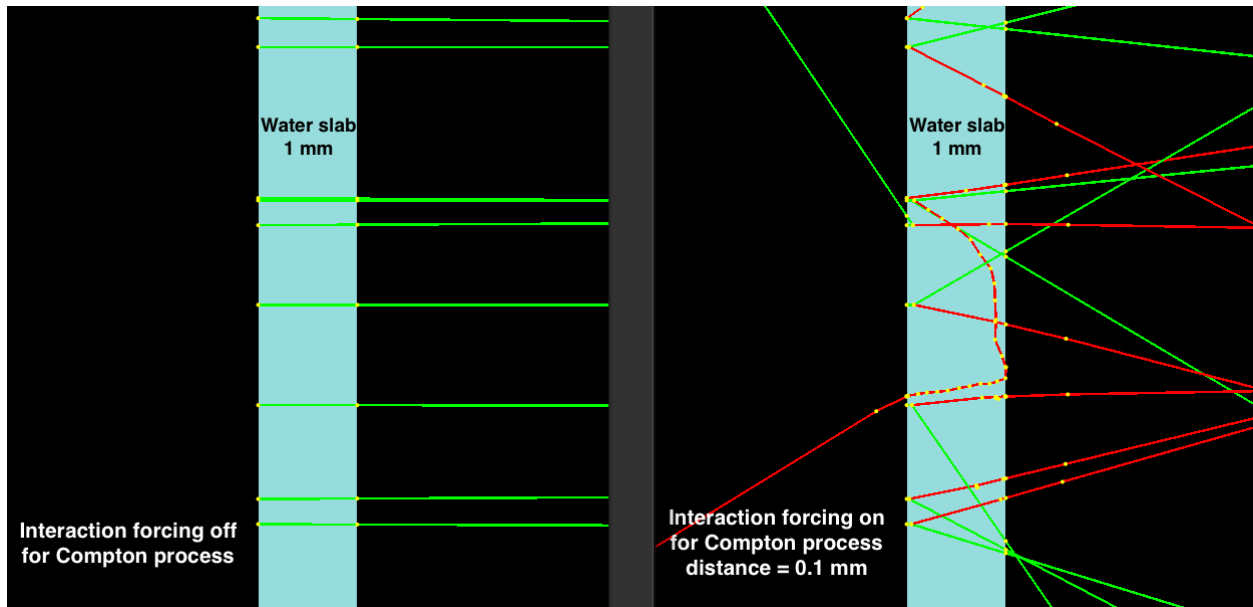
## 16.7 Forced interaction

In this technique, photons are forced to interact by resampling the interaction probability using an scaling distance factor.

**Warning:** The distance must be smaller than the thickness of the component assigned to the region.

This technique is applied per region, then the user is responsible to create and assign the components desired to a region. In addition, a scaling distance and physical process needs to be provided:

```
s:Vr/PhotonInteractionForcing/Type = "ForcedInteraction"
sv:Vr/PhotonInteractionForcing/ForRegion/BiasingRegion/ProcessesNamed = 1 "compt"
dv:Vr/PhotonInteractionForcing/ForRegion/BiasingRegion/ForcedDistances = 1 0.1 mm
```

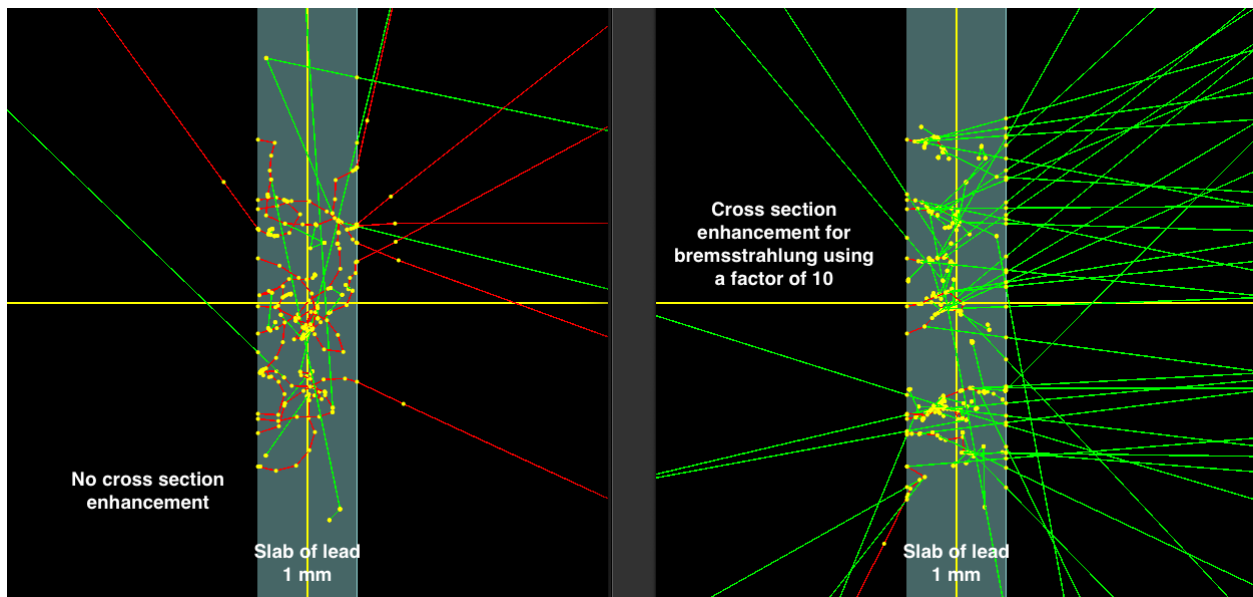


## 16.8 Cross section enhancement

In this technique, particle's interaction in a material is enhanced by rescaling the cross-section of a process provided by the user.

This technique is applied at every interaction of the particle. The user needs to provide a unitless scaling factor and the physical processes of interest:

```
s:Vr/CrossSectionEnhancement/Type = "CrossSectionEnhancement"
sv:Vr/CrossSectionEnhancement/ProcessesNamed = 1 "eBrem"
uv:Vr/CrossSectionEnhancement/EnhancementFactor = 1 20
```







# CHAPTER 17

---

## Outcome Modeling

---

TOPAS can now directly perform Outcome Modeling such as calculating Tumor Control Probabilities and Normal Tissue Complication Probabilities.

Expanding on TOPAS previous capability to directly produce a Dose Volume Histogram, TOPAS can now directly apply outcome models to the DVH. We provide a variety of standard outcome models from the literature, for each of which you can adjust various parameters. You can also write your own outcome models using the TOPAS Extensions interface.

Starting from an existing scorer, if a differential or cumulative histogram will be scored, TOPAS will takes the corresponding bins (dose and volume) as input of the biological models:

```
s:Sc/ScorerName/Report= "differentialvolumehistogram" # or "cumulativevolumehistogram"
```

An example that runs several different outcome models on a patient dose can be seen at [TestOutcomeModel.txt](#)

We also allow you to read back in a previously created DVH to have TOPAS apply new outcome models without having to re-do the Monte Carlo simulation phase of the job. Just set the parameter that tells TOPAS to restore results from a previously created file:

```
Ts/RestoreResultsFromFile = "True"
```

See the example [TestRestoreModel.txt](#)

If no volume histogram is required, as input of the biological models, TOPAS will takes the final full dose distribution in the organ, and by assuming that all voxels have the same dimension, the volume input will be a vector of ones. This assumption relies in the fact that volume bins are internally converted to fractional volume.

To activate the biological models calculation:

```
b:Sc/ScorerName/CalculateProbabilityOfOutcome = "True"
```

User may want to scale the dose distribution:

```
u:Sc/ScorerName/BiologyOutputScaleFactor = 1E6
```

Set the number and name of models to be calculated:

```
sv:Sc/ScorerName/ModelName = 2 "LKB" "CriticalElement"
```

Set the input parameters of the corresponding model:

```
u:Sc/ScorerName/LKB/n = 0.25
u:Sc/ScorerName/LKB/m = 0.15
u:Sc/ScorerName/LKB/td50 = 60
u:Sc/ScorerName/CriticalElement/m = 0.15
u:Sc/ScorerName/CriticalElement/td50 = 60
s:Sc/ScorerName/CriticalElement/Function="probit" #"logistic"
```

For LKB, critical element, critical volume and Poisson models, the parameters can be set from an internal data base (see references below) by input the organ name instead of the model parameters as follows:

```
s:Sc/ScorerName/IncludeParametersFromOrganNamed="brain"
```

If the organ name is not found, the full list of available names is displayed and TOPAS execution is stopped.

The output (NTCP or TCP in percent) will be displayed on the screen for every model for every scorer. If CSV or Binary DVHs output is chosen, the output will be at the header too.

The follow references contains tables with parameters for several organs for several models:

```
C. Burman, G. J. Kutcher, B. Emami and M. Goitein, "Fitting of normal tissue
→tolerance data to an analytic function", Int. J. Radiation Oncology Biol. Phys. 21,
→123-135. (1991)
B. Emami, J. Lyman, A. Brown, L. Coia, M. Goitein, J. E. Munzenrider, B. Shank. L. J.
→Solin and M Wesson "Tolerance of normal tissue to therapeutic irradiation", Int. J.
→Radiation Oncology Biol. Phys. 21, 109-122. (1991)
P. Okunieff, D. Morgan, A. Niemerko and D. Suit "Radiation dose-response of human
→tumors", Int. J. Radiation Oncology Biol. Phys. 32(4), 1227-1237. (1995)
P. Stavrev, A. Niemerko, N. Stavreva and M. Goitein, "The application of biological
→models to clinical data", Physica Medica, 17(2), 2-13. (2001)
```

---

### The TOPAS Graphical User Interface

---

TOPAS includes an optional Graphical User Interface (GUI). The GUI allows the user to easily manipulate graphics (rotate, zoom and click on graphical elements to get more information about the selected geometry element or particle track) and inspect and adjust TOPAS parameters.

The GUI is intended as a rapid prototyping extension of TOPAS. It allows you to easily see and adjust geometry and review the behavior of a small set of particle histories. The GUI then gives you the ability to save the new parameters, such that these saved values can later be loaded back in to a new TOPAS session, with or without the GUI.

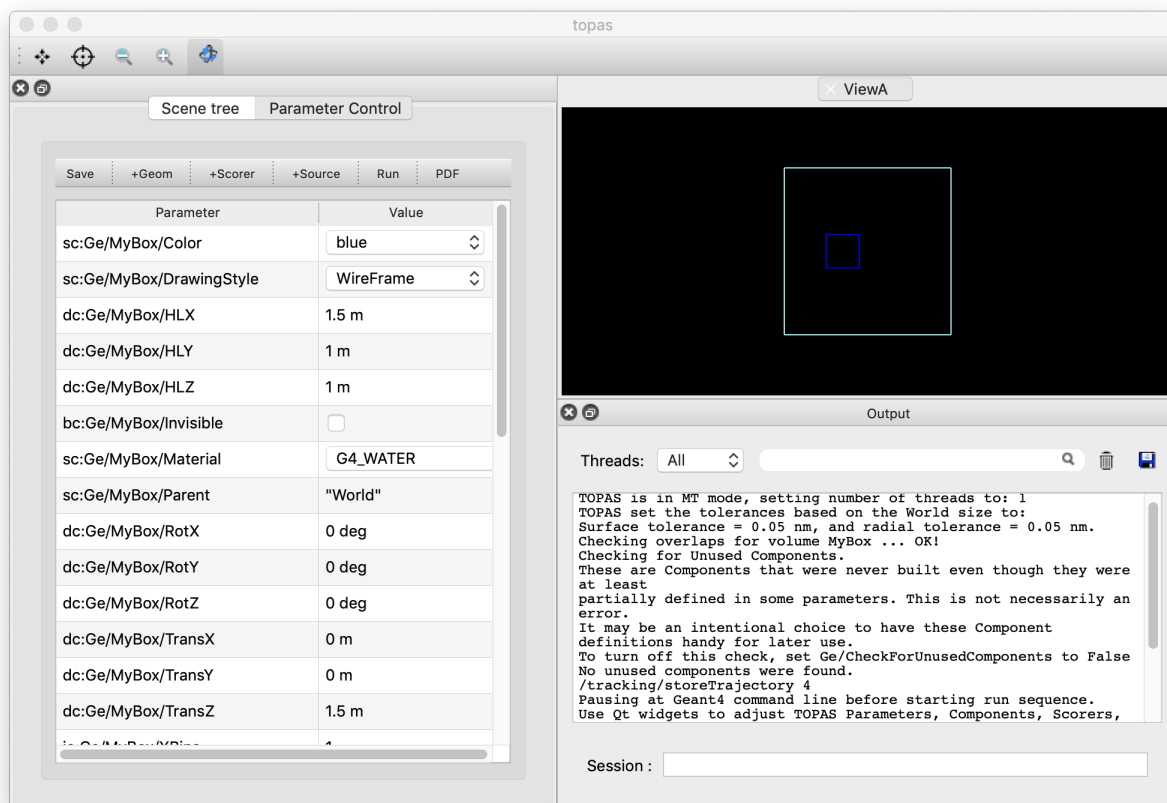
For larger, high precision runs, with larger numbers of histories, the user will generally run on a batch system (cluster or cloud) without the GUI.

Future developments will extend the system such that users will be able to directly launch cluster or cloud jobs from the GUI, with the GUI serving to monitor the progress of those jobs and display their results.

The GUI is based on the Qt Toolkit. To activate the GUI, include the following parameter in your TOPAS Parameter Control File:

```
Ts/UseQt = "True"
```

TOPAS will then bring up the GUI.



## 18.1 Mouse Control

The top row of icons determine the behavior of the mouse when one clicks in the graphics window. Going from left to right:

- Pick to Translate the Image. Moving the mouse will move the image left right up or down
- Pick to Inquire. A pop-up window will give details about the selected object within the image (name of geometry, material, density etc., or type of particle track, particle energy, etc.).
- Pick to Zoom out
- Pick to Zoom in
- Pick to Rotate. Moving the mouse will rotate the image.

## 18.2 The Parameter Control Table

The Parameter Control Table includes one row for each of the parameters that you can adjust. Click on the given value to adjust. There may be a slider or a text input box, depending on the parameter type. If you type an invalid entry, a warning message will appear in the session console area on the lower right, and the previous value will be restored.

You choose which of your TOPAS parameters will be displayed here by adding the prefix character “c” (for “change-able”) to the parameter type. So, for example, to have the Z Half Length of a component named MyBox be included in this control area, specify it in the parameter control file as:

```
i c:Ge/MyBox/HLX = 2
```

Or to have this NOT be present in the control area:

```
i:Ge/MyBox/HLX = 2
```

## 18.3 Additional Functions

A set of buttons above the parameter control table support additional functions:

- **Save:** saves the current set of parameters to a new file in your current directory. The file name will be displayed in the session console area on the lower right. The new file will NOT overwrite your original parameter file. Rather, it will create a new parameter file that includes only your changes, and that then uses the TOPAS includeFile mechanism to bring in your original parameter file for the other, unchanged values.
- **+Geom:** Brings up a window to let you add a new Geometry Component
- **+Scorer:** Brings up a window to let you add a new Scorer. Note that due to constraints in the way Geant4 initializes various functions, you can only add scorers Before the first run.
- **+Source:** Brings up a window to let you add a new Particle Source. Note that due to constraints in the way Geant4 initializes various functions, you can only add scorers Before the first run.
- **Run:** Runs your simulation. You can issue this as many times as you wish, changing parameters and re-running to see new results.
- **PDF:** Saved the current graphics window to a PDF file in your current directory.

## 18.4 Scene Tree

You can switch from the Parameter Control widget to another widget called the Scene Tree. Scene Tree shows you the hierarchy of geometry volumes. You can turn on or off various volumes to better understand your geometry.



## 19.1 AllParameterForms.txt

```
# Demonstrates all allowed parameter forms
# and dumps results to TsDumpParameters.html

b:Ts/DumpNonDefaultParameters = "True"

sv:Ph/Default/Modules = 1 "Transportation_Only"

# Parameters used in expressions below
d:A_Double1      = 100. cm
d:A_Double5      = 500. cm
d:A_Double10     = 1000. cm
d:A_Double20     = 2000. cm
u:A_Unitless1     = 1.
u:A_Unitless5     = 5.
u:A_Unitless10    = 10.
u:A_Unitless20    = 20.
i:A_Integer1      = 1
i:A_Integer5      = 5
i:A_Integer10     = 10
i:A_Integer20     = 20
s:A_StringTen     = "Ten"
b:A_BooleanTrue   = "True"
dv:A_DoubleVector1 = 3 1. 1. 1. m
uv:A_UnitlessVector1 = 3 1. 1. 1.
iv:A_IntegerVector1 = 3 1 1 1
sv:A_StringVector1 = 3 "One" "One" "One"

# Double parameters: correct answer is always 10m
d:A_DoubleFromValue      = 10. m
d:A_DoubleFromValuePlusDouble = 0. m + A_Double10
d:A_DoubleFromValueMinusDouble = 20. m - A_Double10
```

(continues on next page)

(continued from previous page)

```

d:A_DoubleFromValueTimesUnitless = 10. m * A_Unitless1
d:A_DoubleFromValueTimesInteger   = 10. m * A_Integer1
d:A_DoubleFromValueTimesDouble    = 1. * A_Double10 m
d:A_DoubleFromDouble              = A_Double10 m
d:A_DoubleFromDoubleTimesValue    = A_Double10 m * 1.
d:A_DoubleFromDoubleTimesUnitless = A_Double10 m * A_Unitless1
d:A_DoubleFromDoubleTimesInteger  = A_Double10 m * A_Integer1
d:A_DoubleFromUnitlessTimesValue  = A_Unitless1 * 10. m
d:A_DoubleFromIntegerTimesValue   = A_Integer1 * 10. m
d:A_DoubleFromDoublePlusValue     = A_Double10 + 0. m
d:A_DoubleFromDoubleMinusValue    = A_Double20 - 10. m
d:A_DoubleFromDoublePlusDouble    = A_Double5 + A_Double5 m
d:A_DoubleFromDoubleMinusDouble   = A_Double20 - A_Double10 m

# Unitless parameters: correct answer is always 10
u:A_UnitlessFromValue             = 10.
u:A_UnitlessFromValuePlusUnitless = 0. + A_Unitless10
u:A_UnitlessFromValueMinusUnitless = 20. - A_Unitless10
u:A_UnitlessFromValueTimesUnitless = 1. * A_Unitless10
u:A_UnitlessFromValuePlusInteger  = 0. + A_Integer10
u:A_UnitlessFromValueMinusInteger  = 20. - A_Integer10
u:A_UnitlessFromValueTimesInteger  = 1. * A_Integer10
u:A_UnitlessFromUnitless          = A_Unitless10
u:A_UnitlessFromInteger           = A_Integer10
u:A_UnitlessFromUnitlessPlusValue  = A_Unitless10 + 0.
u:A_UnitlessFromUnitlessMinusValue = A_Unitless20 - 10.
u:A_UnitlessFromUnitlessTimesValue = A_Unitless10 * 1.
u:A_UnitlessFromIntegerPlusValue   = A_Integer10 + 0.
u:A_UnitlessFromIntegerMinusValue  = A_Integer20 - 10.
u:A_UnitlessFromIntegerTimesValue  = A_Integer10 * 1.
u:A_UnitlessFromUnitlessPlusUnitless = A_Unitless5 + A_Unitless5
u:A_UnitlessFromUnitlessMinusUnitless = A_Unitless20 - A_Unitless10
u:A_UnitlessFromUnitlessTimesUnitless = A_Unitless10 * A_Unitless1
u:A_UnitlessFromUnitlessPlusInteger  = A_Unitless5 + A_Integer5
u:A_UnitlessFromUnitlessMinusInteger  = A_Unitless20 - A_Integer10
u:A_UnitlessFromUnitlessTimesInteger  = A_Unitless10 * A_Integer1
u:A_UnitlessFromIntegerPlusUnitless  = A_Integer5 + A_Unitless5
u:A_UnitlessFromIntegerMinusUnitless  = A_Integer20 - A_Unitless10
u:A_UnitlessFromIntegerTimesUnitless  = A_Integer10 * A_Unitless1
u:A_UnitlessFromIntegerPlusInteger    = A_Integer5 + A_Integer5
u:A_UnitlessFromIntegerMinusInteger    = A_Integer20 - A_Integer10
u:A_UnitlessFromIntegerTimesInteger    = A_Integer10 * A_Integer1

# Integer parameters: correct answer is always 10
i:A_IntegerFromValue             = 10
i:A_IntegerFromValuePlusInteger  = 0 + A_Integer10
i:A_IntegerFromValueMinusInteger  = 20 - A_Integer10
i:A_IntegerFromValueTimesInteger  = 1 * A_Integer10
i:A_IntegerFromInteger           = A_Integer10
i:A_IntegerFromIntegerPlusValue   = A_Integer10 + 0
i:A_IntegerFromIntegerMinusValue  = A_Integer20 - 10
i:A_IntegerFromIntegerTimesValue  = A_Integer10 * 1
i:A_IntegerFromIntegerPlusInteger  = A_Integer5 + A_Integer5
i:A_IntegerFromIntegerMinusInteger = A_Integer20 - A_Integer10
i:A_IntegerFromIntegerTimesInteger = A_Integer10 * A_Integer1

# Boolean parameters: correct answer is always 1 (= True)

```

(continues on next page)



(continued from previous page)

```

b:A_BooleanFromValue           = "True"
b:A_BooleanFromBoolean         = A_BooleanFromValue
b:A_BooleanFromBooleanTimesBoolean = A_BooleanFromValue * A_BooleanTrue

# String parameters: correct answer is always Ten, 10
# or some combination of the two such as TenTen, Ten10, 1010, etc.
s:A_StringFromValue           = "Ten"
s:A_StringFromValuePlusInteger = "Ten" + A_Integer10
s:A_StringFromValuePlusString  = "Ten" + A_StringTen
s:A_StringFromInteger          = A_Integer10
s:A_StringFromString           = A_StringTen
s:A_StringFromIntegerPlusValue = A_Integer10 + "Ten"
s:A_StringFromStringPlusValue  = A_StringTen + "Ten"
s:A_StringFromIntegerPlusInteger = A_Integer10 + A_Integer10
s:A_StringFromIntegerPlusString = A_Integer10 + A_StringTen
s:A_StringFromStringPlusInteger = A_StringTen + A_Integer10
s:A_StringFromStringPlusString  = A_StringTen + A_StringTen

# Double Vectors: correct answer is always 10. 5. 1. m
dv:A_DoubleVectorFromValue           = 3 10. 5. 1. m
dv:A_DoubleVectorFromValuePlusDouble = 3 9. 4. 0. m + A_Double1
dv:A_DoubleVectorFromValueMinusDouble = 3 11. 6. 2. m - A_Double1
dv:A_DoubleVectorFromValuePlusDoubleVector = 3 9. 4. 0. m + A_DoubleVector1
dv:A_DoubleVectorFromValueMinusDoubleVector = 3 11. 6. 2. m - A_DoubleVector1
dv:A_DoubleVectorFromValueTimesUnitless = 3 10. 5. 1. m * A_Unitless1
dv:A_DoubleVectorFromValueTimesInteger = 3 10. 5. 1. m * A_Integer1
dv:A_DoubleVectorFromValueTimesUnitlessVector = 3 10. 5. 1. m * A_UnitlessVector1
dv:A_DoubleVectorFromValueTimesIntegerVector = 3 10. 5. 1. m * A_IntegerVector1
dv:A_DoubleVectorFromValueTimesDouble = 3 10. 5. 1. * A_Double1 m
dv:A_DoubleVectorFromValueTimesDoubleVector = 3 10. 5. 1. * A_DoubleVector1 m
dv:A_DoubleVectorFromDoubleVector = A_DoubleVectorFromValue m
dv:A_DoubleVectorFromScaleTimesDoubleVector = 1. * A_DoubleVectorFromValue m
dv:A_DoubleVectorFromIntegerTimesDoubleVector = A_IntegerFromValue * A_
↪DoubleVectorFromValue m
dv:A_DoubleVectorFromUnitlessTimesDoubleVector = A_UnitlessFromValue * A_
↪DoubleVectorFromValue m

# Unitless Vectors: correct answer is always 10 5 1
uv:A_UnitlessVectorFromValue           = 3 10. 5. 1.
uv:A_UnitlessVectorFromValuePlusUnitless = 3 9. 4. 0. + A_Unitless1
uv:A_UnitlessVectorFromValueMinusUnitless = 3 11. 6. 2. - A_Unitless1
uv:A_UnitlessVectorFromValueTimesUnitless = 3 10. 5. 1. * A_Unitless1
uv:A_UnitlessVectorFromValuePlusInteger = 3 9. 4. 0. + A_Integer1
uv:A_UnitlessVectorFromValueMinusInteger = 3 11. 6. 2. - A_Integer1
uv:A_UnitlessVectorFromValueTimesInteger = 3 10. 5. 1. * A_Integer1
uv:A_UnitlessVectorFromValuePlusUnitlessVector = 3 9. 4. 0. + A_UnitlessVector1
uv:A_UnitlessVectorFromValueMinusUnitlessVector = 3 11. 6. 2. - A_UnitlessVector1
uv:A_UnitlessVectorFromValueTimesUnitlessVector = 3 10. 5. 1. * A_UnitlessVector1
uv:A_UnitlessVectorFromValuePlusIntegerVector = 3 9. 4. 0. + A_IntegerVector1
uv:A_UnitlessVectorFromValueMinusIntegerVector = 3 11. 6. 2. - A_IntegerVector1
uv:A_UnitlessVectorFromValueTimesIntegerVector = 3 10. 5. 1. * A_IntegerVector1
uv:A_UnitlessVectorFromUnitlessVector = A_UnitlessVectorFromValue
uv:A_UnitlessVectorFromScaleTimesUnitlessVector = 1. * A_UnitlessVectorFromValue
uv:A_UnitlessVectorFromUnitlessTimesUnitlessVector = A_UnitlessFromValue * A_
↪UnitlessVectorFromValue
uv:A_UnitlessVectorFromIntegerTimesUnitlessVector = A_IntegerFromValue * A_
↪UnitlessVectorFromValue

```

(continues on next page)

(continued from previous page)

```
# Integer Vectors: correct answer is always 10 5 1
iv:A_IntegerVectorFromValue           = 3 10 5 1
iv:A_IntegerVectorFromValuePlusInteger = 3 9 4 0 + A_Integer1
iv:A_IntegerVectorFromValueMinusInteger = 3 11 6 2 - A_Integer1
iv:A_IntegerVectorFromValueTimesInteger = 3 10 5 1 * A_Integer1
iv:A_IntegerVectorFromValuePlusIntegerVector = 3 9 4 0 + A_IntegerVector1
iv:A_IntegerVectorFromValueMinusIntegerVector = 3 11 6 2 - A_IntegerVector1
iv:A_IntegerVectorFromValueTimesIntegerVector = 3 10 5 1 * A_IntegerVector1
iv:A_IntegerVectorFromIntegerVector      = A_IntegerVectorFromValue
iv:A_IntegerVectorFromScaleTimesIntegerVector = 1 * A_IntegerVectorFromValue
iv:A_IntegerVectorFromIntegerTimesIntegerVector = A_IntegerFromValue * A_
IntegerVectorFromValue

# Boolean Vectors: correct answer is always 1 0 1 (= True False True)
bv:A_BooleanVectorFromValue           = 3 "True" "False" "True"
bv:A_BooleanVectorFromBooleanVector    = A_BooleanVectorFromValue

# String Vectors: correct answers vary by case
sv:A_StringVectorFromValue             = 3 "Ten" "Five" "One"
sv:A_StringVectorFromValuePlusInteger  = 3 "Ten" "Five" "One" + A_Integer10
sv:A_StringVectorFromValuePlusString   = 3 "Ten" "Five" "One" + A_StringTen
sv:A_StringVectorFromStringVector       = A_StringVectorFromValue
sv:A_StringVectorFromValuePlusIntegerVector = 3 "Ten" "Five" "One" + A_IntegerVector1
sv:A_StringVectorFromValuePlusStringVector = 3 "Ten" "Five" "One" + A_StringVector1
```

## 19.2 DividedComponents.txt

```
# Demonstrates use of the Divided Components,
# TsBox, TsCylinder and TsSphere.

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

# Save time by turning off overlap check
b:Ge/CheckForOverlaps = "False"

s:Ge/TestBox/Parent   = "World"
s:Ge/TestBox/Type     = "TsBox"
s:Ge/TestBox/Material = "G4_WATER"
d:Ge/TestBox/HLX      = 7.0 cm
d:Ge/TestBox/HLY      = 7.0 cm
d:Ge/TestBox/HLZ      = 9.0 cm
d:Ge/TestBox/TransX    = 0. cm
d:Ge/TestBox/TransY    = 0. cm
d:Ge/TestBox/TransZ    = 0. cm
d:Ge/TestBox/RotX      = 0. deg
d:Ge/TestBox/RotY      = 0. deg
d:Ge/TestBox/RotZ      = 0. deg
s:Ge/TestBox/Color     = "blue"
i:Ge/TestBox/XBins     = 3
i:Ge/TestBox/YBins     = 4
```

(continues on next page)

(continued from previous page)

```

i:Ge/TestBox/ZBins      = 5

s:Ge/TestCylinder/Parent    = "World"
s:Ge/TestCylinder/Type      = "TsCylinder"
s:Ge/TestCylinder/Material  = "G4_WATER"
d:Ge/TestCylinder/RMin      = 0.0 cm
d:Ge/TestCylinder/RMax      = 8.0 cm
d:Ge/TestCylinder/HL        = 10.0 cm
d:Ge/TestCylinder/SPhi      = 0. deg
d:Ge/TestCylinder/DPhi      = 360. deg
d:Ge/TestCylinder/TransX    = 0. cm
d:Ge/TestCylinder/TransY    = 0. cm
d:Ge/TestCylinder/TransZ    = 25. cm
d:Ge/TestCylinder/RotX      = 0. deg
d:Ge/TestCylinder/RotY      = 0.0 deg
d:Ge/TestCylinder/RotZ      = 0. deg
s:Ge/TestCylinder/Color     = "blue"
i:Ge/TestCylinder/RBins     = 2
i:Ge/TestCylinder/PhiBins   = 8
i:Ge/TestCylinder/ZBins     = 2

s:Ge/TestSphere/Parent     = "World"
s:Ge/TestSphere/Type       = "TsSphere"
s:Ge/TestSphere/Material   = "G4_WATER"
d:Ge/TestSphere/RMin       = 0.0 cm
d:Ge/TestSphere/RMax       = 10.0 cm
d:Ge/TestSphere/SPhi       = 0. deg
d:Ge/TestSphere/DPhi       = 360. deg
d:Ge/TestSphere/STheta     = 0. deg
d:Ge/TestSphere/DTheta     = 180. deg
d:Ge/TestSphere/TransX     = 0. cm
d:Ge/TestSphere/TransY     = 0. cm
d:Ge/TestSphere/TransZ     = 50. cm
d:Ge/TestSphere/RotX       = 0. deg
d:Ge/TestSphere/RotY       = 0. deg
d:Ge/TestSphere/RotZ       = 0. deg
s:Ge/TestSphere/Color      = "blue"
i:Ge/TestSphere/RBins      = 3
i:Ge/TestSphere/PhiBins    = 4
i:Ge/TestSphere/ThetaBins  = 12

s:Gr/ViewA/Type            = "OpenGL"
i:Gr/ViewA/WindowSizeX     = 1024
i:Gr/ViewA/WindowSizeY     = 768
d:Gr/ViewA/Theta           = 55 deg
d:Gr/ViewA/Phi             = 20 deg
s:Gr/ViewA/Projection       = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom             = 1.8
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

sv:Ph/Default/Modules      = 1 "g4em-standard_opt0"

b:Ts/PauseBeforeQuit       = "True"

```

## 19.3 DNAModelByRegions.txt

```
# Demonstrates having G4DNA model defined in specific regions.
# While only one electromagnetic model can be used in a given TOPAS
# physics list, you are free to define a dna-electromagnetic model in
# a user specific region.
# The em models currently supported are: g4em-livermore, g4em-polarized
# g4em-lowep, g4em-penelope and g4em-standard_opt(0-4)
# The dna-electromagnetic model names are: opt0, opt1 and opt2.

# A simple geometry. The material must be G4_WATER.
s:Ge/Target/Parent      = "World"
s:Ge/Target/Type        = "TsBox"
s:Ge/Target/Material    = "G4_WATER"
d:Ge/Target/HLX         = 0.5 um
d:Ge/Target/HLY         = 0.5 um
d:Ge/Target/HLZ         = 0.25 um
s:Ge/Target/Color       = "grass"
# Assign a region
s:Ge/Target/AssignToRegionNamed = "target"

d:Ge/World/HLX          = Ge/Target/HLX um
d:Ge/World/HLY          = Ge/Target/HLY um
d:Ge/World/HLZ          = 1.5 * Ge/Target/HLZ um
s:Ge/World/Material     = "G4_WATER"
s:Ge/World/Color        = "blue"

# Start with a physics list
sv:Ph/Default/Modules = 6 "g4em-penelope" "g4h-phy_QGSP_BIC_HP" "g4decay" "g4ion-
→binarycascade" "g4h-elastic_HP" "g4stopping"
# Set the regions where g4em-penelope is switch to g4-dna
s:Ph/Default/ForRegion/target/ActiveG4EmModelNamed = "g4em-dna"

# Set a source
s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "proton"
d:So/Example/BeamEnergy        = 5 MeV
u:So/Example/BeamEnergySpread  = 0.757504
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

s:Gr/view/Type = "OpenGL"
b:Ts/PauseBeforeQuit = "True"
```

## 19.4 Emittance\_Gaussian.txt

```
s:Ge/World/Material = "Vacuum"
```

(continues on next page)

(continued from previous page)

```

#--- Beam
s:So/Example/Component           = "BeamPosition"
s:So/Example/BeamParticle        = "proton"
d:So/Example/BeamEnergy          = 94.00 MeV
u:So/Example/BeamEnergySpread    = 1.0
i:So/Example/NumberOfHistoriesInRun = 10000

#----- Primary: Emittance beam
s:So/Example/Type                = "Emittance"

#1. Bivariate Gaussian: X,X',correlation and Y,Y',correlation
s:So/Example/Distribution        = "BiGaussian"
d:So/Example/SigmaX             = 2.0 mm
u:So/Example/SigmaXprime        = 0.0032 #32 mrad
u:So/Example/CorrelationX       = -0.8
d:So/Example/SigmaY             = 2.0 mm
u:So/Example/SigmaYprime        = 0.0032
u:So/Example/CorrelationY       = 0.4

# Beam position (S)
s:Ge/BeamPosition/Parent        = "World"
s:Ge/BeamPosition/Type          = "Group"
d:Ge/BeamPosition/TransX        = 0. m
d:Ge/BeamPosition/TransY        = 0. m
d:Ge/BeamPosition/TransZ        = 3.001 m
#flipped cause the beam flies 0 to +z.
d:Ge/BeamPosition/RotX          = 180. deg
d:Ge/BeamPosition/RotY          = 0. deg
d:Ge/BeamPosition/RotZ          = 0. deg

#---- Verbosity
i:Ts/TrackingVerbosity = 0

#---- Sequence
b:Ge/CheckForUnusedComponents = "False"
b:Ts/ShowCPUtime = "true"
i:Ts/ShowHistoryCountAtInterval = 1000
i:Ts/Seed = 10

#--- Physics --
sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

#---- Constants
d:Ge/MyBeamSizeInX = 20.0 cm
d:Ge/MyBeamSizeInY = 20.0 cm

#--- Beam Box
s:Ge/PlaneAtBeamPosition/Type    = "TsBox"
s:Ge/PlaneAtBeamPosition/Material = "Vacuum"
s:Ge/PlaneAtBeamPosition/Parent  = "World"
d:Ge/PlaneAtBeamPosition/HLX      = 0.5 cm + Ge/MyBeamSizeInX
d:Ge/PlaneAtBeamPosition/HLY      = 0.5 cm + Ge/MyBeamSizeInY
d:Ge/PlaneAtBeamPosition/HLZ      = 0.05 mm
d:Ge/PlaneAtBeamPosition/TransX    = 0. m
d:Ge/PlaneAtBeamPosition/TransY    = 0. m
d:Ge/PlaneAtBeamPosition/TransZ    = 3.0 m - Ge/PlaneAtBeamPosition/HLZ
d:Ge/PlaneAtBeamPosition/RotX      = 0. deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/PlaneAtBeamPosition/RotY      = 0. deg
d:Ge/PlaneAtBeamPosition/RotZ      = 0. deg

#--- Phase space scoring at Beam geometry
s:Sc/Beam/Quantity                  = "PhaseSpace"
b:Sc/Beam/OutputToConsole           = "True"
s:Sc/Beam/Surface                   = "PlaneAtBeamPosition/ZPlusSurface"
s:Sc/Beam/OutputType                 = "ASCII"
i:Sc/Beam/OutputBufferSize          = 1000
b:Sc/Beam/IncludeRunID              = "True"
b:Sc/Beam/IncludeEventID            = "True"
b:Sc/Beam/IncludeTrackID            = "True"
b:Sc/Beam/IncludeTime               = "True"
b:Sc/Beam/IncludeSeed               = "True"
s:Sc/Beam/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/Beam/OutputFile                = "BeamGaussian"

```

## 19.5 Emittance\_Twiss.txt

```

includeFile = Emittance_Gaussian.txt

#!!! You can choose one from these three options
s:So/Example/Distribution = "twiss_gaussian"
#s:So/Example/Distribution = "twiss_kv"
#s:So/Example/Distribution = "twiss_waterbag"

u:So/Example/AlphaX      = 0.2
d:So/Example/BetaX       = 600.0 mm #
d:So/Example/EmittanceX  = 0.01 mm #we don't multiply pi

u:So/Example/AlphaY      = 2.5
d:So/Example/BetaY       = 1400.0 mm
d:So/Example/EmittanceY  = 0.02 mm

#+ twiss gaussian option only
#0.9 means that 90 % particles will be included in ellipse circle
u:So/Example/ParticleFractionX = 0.90
u:So/Example/ParticleFractionY = 0.90

s:Sc/Beam/OutputFile = "BeamTwiss"

```

## 19.6 EmModelByRegions.txt

```

# Demonstrates having electromagnetic model defined in specific regions.
# While only one electromagnetic model can be used in a given TOPAS
# physics list, you are free to define a electromagnetic model in
# a user specific region.
# The em models currently supported are: g4em-livermore, g4em-polarized
# g4em-lowep, g4em-penelope and g4em-standard_opt(0-4)

# A simple geometry. The material must be G4_WATER.

```

(continues on next page)

(continued from previous page)

```

s:Ge/Target/Parent      = "World"
s:Ge/Target/Type        = "TsBox"
s:Ge/Target/Material    = "G4_WATER"
d:Ge/Target/HLX         = 0.5 cm
d:Ge/Target/HLY         = 0.5 cm
d:Ge/Target/HLZ         = 0.25 cm
s:Ge/Target/Color       = "grass"
# Assign a region
s:Ge/Target/AssignToRegionNamed = "target"

d:Ge/World/HLX          = Ge/Target/HLX um
d:Ge/World/HLY          = Ge/Target/HLY um
d:Ge/World/HLZ          = 1.5 * Ge/Target/HLZ um
s:Ge/World/Material     = "G4_WATER"
s:Ge/World/Color        = "blue"

# Start with a physics list
sv:Ph/Default/Modules = 6 "g4em-penelope" "g4h-phy_QGSP_BIC_HP" "g4decay" "g4ion-
  ↪binarycascade" "g4h-elastic_HP" "g4stopping"
s:Ph/Default/ForRegion/target/ActiveG4EmModelFromModule = "g4em-livermore"

# Set a source
s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "proton"
d:So/Example/BeamEnergy        = 80 MeV
u:So/Example/BeamEnergySpread  = 0.757504
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

s:Gr/view/Type = "OpenGL"
b:Ts/PauseBeforeQuit = "True"

```

## 19.7 FlatteningFilter.txt

```

# This is not any particular actual flattening filter.
# It just demonstrates the general principle of how to
# combine two different kinds of polycones to create
# a shape that no single polycone can represent.

b:Ge/World/Invisible = "True"

s:Gr/MyViewA/Type = "OpenGL"
d:Gr/MyViewA/Phi = 45. deg
d:Gr/MyViewA/Theta = 45. deg

b:Ts/PauseBeforeQuit = "True"

s:Ge/FlatteningFilter/Type = "Group"

```

(continues on next page)

(continued from previous page)

```

s:Ge/FlatteningFilter/Parent = "World"
#d:Ge/FlatteningFilter/RotX = 90. deg

# Inner section is implemented as a Solid Polycone
s:Ge/InnerCone/Type          = "G4SPolycone"
s:Ge/InnerCone/Parent        = "FlatteningFilter"
s:Ge/InnerCone/Material      = "G4_Ti"
s:Ge/InnerCone/Color         = "grey"
s:Ge/InnerCone/DrawingStyle  = "Solid"
dv:Ge/InnerCone/R            = 7 0 9 9 6 5 2 0 mm
dv:Ge/InnerCone/Z            = 7 0 0 1 2 6 8 10 mm

# Outer section is implemented as a Hollow Polycone
s:Ge/OuterCone/Type          = "G4HPolycone"
s:Ge/OuterCone/Parent        = "FlatteningFilter"
s:Ge/OuterCone/Material      = "G4_Ti"
s:Ge/OuterCone/Color         = "brown"
s:Ge/OuterCone/DrawingStyle  = "Solid"
dv:Ge/OuterCone/ROuter       = 4 15 15 15 15 mm
dv:Ge/OuterCone/RInner      = 4 9 9 10 13 mm
dv:Ge/OuterCone/Z            = 4 0 1 2 3 mm

```

## 19.8 Isotope.txt

```

# Example of creating an element from specific abundances

sv:Ma/MyUranium/Components = 1 "MyElU"
uv:Ma/MyUranium/Fractions   = 1 1.0
d:Ma/MyUranium/Density      = 18.95 g/cm3
s:Ma/MyUranium/DefaultColor = "green"

s:El/MyElU/Symbol           = "MyElU"
sv:El/MyElU/IsotopeNames    = 2 "U235" "U238"
uv:El/MyElU/IsotopeAbundances = 2 90. 10.

i:Is/U235/Z = 92
i:Is/U235/N = 235
d:Is/U235/A = 235.01 g/mole
i:Is/U238/Z = 92
i:Is/U238/N = 238
d:Is/U238/A = 238.03 g/mole

s:Ge/MyBox/Type          = "TsBox"
s:Ge/MyBox/Material      = "MyUranium"
s:Ge/MyBox/Parent        = "World"
d:Ge/MyBox/HLX           = 2.5 m
d:Ge/MyBox/HLY           = 2. m
d:Ge/MyBox/HLZ           = 10. mm
d:Ge/MyBox/TransX        = 2. m
d:Ge/MyBox/TransY        = 0. m
d:Ge/MyBox/TransZ        = 0. m
d:Ge/MyBox/RotX          = 0. deg
d:Ge/MyBox/RotY          = 0. deg
d:Ge/MyBox/RotZ          = 0. deg

```

(continues on next page)



(continued from previous page)

```

s:Gr/ViewA/Type           = "OpenGL"
i:Gr/ViewA/WindowSizeX    = 1024
i:Gr/ViewA/WindowSizeY    = 768
d:Gr/ViewA/Theta          = 55 deg
d:Gr/ViewA/Phi            = 20 deg
s:Gr/ViewA/Projection     = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom           = 2.

s:So/Example/Type         = "Beam"
s:So/Example/Component    = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy   = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX    = 10. cm
d:So/Example/BeamPositionCutoffY    = 10. cm
d:So/Example/BeamPositionSpreadX     = 0.65 cm
d:So/Example/BeamPositionSpreadY     = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX      = 90. deg
d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 10

b:Ts/PauseBeforeQuit = "True"

```

## 19.9 LayeredMassGeometry.txt

```

# Demonstrates use of Layered Mass Geometry,
# a Geant4 feature whereby there can be material in a parallel world.

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

sv:Ph/Default/LayeredMassGeometryWorlds = 2 "WaterBox" "IronBox"

s:Ge/AirBox/Parent   = "World"
s:Ge/AirBox/Type     = "TsBox"
s:Ge/AirBox/Material = "Air"
d:Ge/AirBox/HLX      = 1. m
d:Ge/AirBox/HLY      = 1. m
d:Ge/AirBox/HLZ      = 1. m
d:Ge/AirBox/TransX    = 0. m
d:Ge/AirBox/TransY    = 0. m
d:Ge/AirBox/TransZ    = 0. m
d:Ge/AirBox/RotX      = 0. deg
d:Ge/AirBox/RotY      = 0. deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/AirBox/RotZ      = 0. deg

s:Ge/WaterBox/Parent  = "World"
s:Ge/WaterBox/Type    = "TsBox"
s:Ge/WaterBox/Material = "G4_WATER"
d:Ge/WaterBox/HLX     = .6 m
d:Ge/WaterBox/HLY     = .6 m
d:Ge/WaterBox/HLZ     = .6 m
d:Ge/WaterBox/TransX  = 0. m
d:Ge/WaterBox/TransY  = 0. m
d:Ge/WaterBox/TransZ  = 0. m
d:Ge/WaterBox/RotX    = 0. deg
d:Ge/WaterBox/RotY    = 0. deg
d:Ge/WaterBox/RotZ    = 0. deg
b:Ge/WaterBox/isParallel = "True"

s:Ge/IronBox/Parent   = "World"
s:Ge/IronBox/Type     = "TsBox"
s:Ge/IronBox/Material = "Iron"
d:Ge/IronBox/HLX      = .4 m
d:Ge/IronBox/HLY      = .4 m
d:Ge/IronBox/HLZ      = .4 m
d:Ge/IronBox/TransX   = 0. m
d:Ge/IronBox/TransY   = 0. m
d:Ge/IronBox/TransZ   = 0. m
d:Ge/IronBox/RotX     = 0. deg
d:Ge/IronBox/RotY     = 0. deg
d:Ge/IronBox/RotZ     = 0. deg
b:Ge/IronBox/isParallel = "True"

b:Ts/PauseBeforeQuit = "True"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 270. MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 5 cm
d:So/Example/BeamPositionCutoffY = 20 cm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 10

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:Gr/ViewA/Type      = "OpenGL"
sv:Gr/ViewA/VisibleWorlds = 1 "All"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta      = 90 deg
d:Gr/ViewA/Phi        = 0 deg

```

## 19.10 OneBox.txt

```
# Simplest TOPAS example.
# A box in a beam with EM physics.

s:Ge/MyBox/Type      = "TsBox"
s:Ge/MyBox/Material  = "Air"
s:Ge/MyBox/Parent    = "World"
d:Ge/MyBox/HLX       = 2.5 m
d:Ge/MyBox/HLY       = 2. m
d:Ge/MyBox/HLZ       = 1. m
d:Ge/MyBox/TransX     = 2. m
d:Ge/MyBox/TransY     = 0. m
d:Ge/MyBox/TransZ     = 0. m
d:Ge/MyBox/RotX       = 0. deg
d:Ge/MyBox/RotY       = 0. deg
d:Ge/MyBox/RotZ       = 0. deg

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSize = 1024
i:Gr/ViewA/WindowSize = 768
b:Gr/ViewA/IncludeAxes = "True"
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi        = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 2.

b:Ts/PauseBeforeQuit = "True"
```

## 19.11 OneBoxRotate.txt

```
# Demonstrates use of includeFile.
# Overrides the RotX value from OneBox.txt

includeFile = OneBox.txt

d:Ge/MyBox/RotX = 45. deg
```

## 19.12 OneBoxTranslate.txt

```
# Demonstrates use of includeFile.
# Translates box by a value of -0.5 times the value
# of TransX that was inherited from the includeFile.

includeFile = OneBox.txt

d:Ge/MyBox/TransX = inheritedValue m * -0.5
```

## 19.13 PhysicsSetting.txt

```
# Demonstrates having several physics lists defined.
# While only one physics list can be used in a given TOPAS session,
# you are free to define several different sets in your parameter files.
# The parameter Ph/ListName tells which of these lists is actually used.

s:Ph/ListName = "MyList1"

s:Ph/MyList1/Type          = "Geant4_Modular"
sv:Ph/MyList1/Modules      = 3 "g4decay" "g4em-standard_opt3" "g4h-elastic"
d:Ph/MyList1/CutForGamma   = 0.04 mm
d:Ph/MyList1/CutForElectron = 0.03 mm
d:Ph/MyList1/CutForPositron = 0.02 mm
d:Ph/MyList1/CutForProton  = 0.01 mm

s:Ph/MyList2/Type          = "Geant4_Modular"
sv:Ph/MyList2/Modules      = 1 "g4em-standard_opt0"
d:Ph/MyList2/CutForGamma   = 0.04 mm
d:Ph/MyList2/CutForElectron = 0.03 mm
d:Ph/MyList2/CutForPositron = 0.02 mm
d:Ph/MyList2/CutForProton  = 0.01 mm

s:Ph/MyList3/Type          = "Geant4_Modular"
sv:Ph/MyList3/Modules      = 1 "Transportation_Only"
d:Ph/MyList3/CutForGamma   = 0.04 mm
d:Ph/MyList3/CutForElectron = 0.03 mm
d:Ph/MyList3/CutForPositron = 0.02 mm
d:Ph/MyList3/CutForProton  = 0.01 mm

s:Ph/MyList4/Type          = "QGSP_BERT_HP"
d:Ph/MyList4/CutForAllParticles = 0.03 mm
i:Ph/MyList4/Verbosity     = 2

s:Ge/Phantom/Type          = "TsBox"
s:Ge/Phantom/Parent        = "World"
s:Ge/Phantom/Material      = "G4_WATER"
d:Ge/Phantom/HLX           = 80.0 cm
d:Ge/Phantom/HLY           = 80.0 cm
d:Ge/Phantom/HLZ           = 20.0 cm
d:Ge/Phantom/TransX        = 0. cm
d:Ge/Phantom/TransY        = 0. cm
d:Ge/Phantom/TransZ        = -30. cm
d:Ge/Phantom/RotX          = 0. deg
d:Ge/Phantom/RotY          = 0.0 deg
d:Ge/Phantom/RotZ          = 0. deg
s:Ge/Phantom/Color         = "blue"
i:Ge/Phantom/XBins         = 2

s:Sc/EnergyDepAtPhantom/Quantity = "DoseToWater"
s:Sc/EnergyDepAtPhantom/Component = "Phantom"
b:Sc/EnergyDepAtPhantom/OutputToConsole = "TRUE"
s:Sc/EnergyDepAtPhantom/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type = "OpenGL"

s:So/Example/Type = "Beam"
```

(continues on next page)

(continued from previous page)

```

s:So/Example/Component           = "BeamPosition"
s:So/Example/BeamParticle        = "proton"
d:So/Example/BeamEnergy          = 169.23 MeV
u:So/Example/BeamEnergySpread    = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX  = 10. cm
d:So/Example/BeamPositionCutoffY  = 10. cm
d:So/Example/BeamPositionSpreadX  = 0.65 cm
d:So/Example/BeamPositionSpreadY  = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX   = 90. deg
d:So/Example/BeamAngularCutoffY   = 90. deg
d:So/Example/BeamAngularSpreadX   = 0.0032 rad
d:So/Example/BeamAngularSpreadY   = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 19.14 ShapeTestWithAllParameters.txt

```

# Demonstrates all of the standard Geant4 solids.
# Shows all parameters, including optional ones
# (the ones such as TransX or RMin that have default values).
# Details are given in the TOPAS User Guide
# and in chapter 4 of the Geant4 Application Developer's Guide
# http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/
# ForApplicationDeveloper/html/ch04.html#sect.Geom.Solids

# Box
s:Ge/DemoBox/Type           = "TsBox"
s:Ge/DemoBox/Parent         = "World"
s:Ge/DemoBox/Material        = "Air"
d:Ge/DemoBox/TransX         = 0 cm
d:Ge/DemoBox/TransY         = 0 cm
d:Ge/DemoBox/TransZ         = 0 cm
d:Ge/DemoBox/RotX           = 0 deg
d:Ge/DemoBox/RotY           = 0 deg
d:Ge/DemoBox/RotZ           = 0 deg
d:Ge/DemoBox/HLX            = 30. mm
d:Ge/DemoBox/HLY            = 40. mm
d:Ge/DemoBox/HLZ            = 60. mm
s:Ge/DemoBox/Color          = "white"

# Cylinder
s:Ge/DemoCylinder/Type      = "TsCylinder"
s:Ge/DemoCylinder/Parent    = "World"
s:Ge/DemoCylinder/Material   = "Air"
d:Ge/DemoCylinder/TransX    = 0 cm
d:Ge/DemoCylinder/TransY    = 0 cm
d:Ge/DemoCylinder/TransZ    = 12 cm
d:Ge/DemoCylinder/RotX      = 0 deg
d:Ge/DemoCylinder/RotY      = 0 deg
d:Ge/DemoCylinder/RotZ      = 0 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/DemoCylinder/RMin          = 10 mm
d:Ge/DemoCylinder/RMax          = 15 mm
d:Ge/DemoCylinder/HL            = 20 mm
d:Ge/DemoCylinder/SPhi          = 1. rad
d:Ge/DemoCylinder/DPhi          = 1.5 rad
s:Ge/DemoCylinder/Color          = "lightblue"
s:Ge/DemoCylinder/DrawingStyle   = "FullWireFrame"

# CutTubs
s:Ge/DemoCutTubs/Type            = "G4CutTubs"
s:Ge/DemoCutTubs/Parent          = "World"
s:Ge/DemoCutTubs/Material        = "Air"
d:Ge/DemoCutTubs/TransX          = 0 cm
d:Ge/DemoCutTubs/TransY          = 0 cm
d:Ge/DemoCutTubs/TransZ          = 24 cm
d:Ge/DemoCutTubs/RotX            = 0 deg
d:Ge/DemoCutTubs/RotY            = 0 deg
d:Ge/DemoCutTubs/RotZ            = 0 deg
d:Ge/DemoCutTubs/RMin            = 12 mm
d:Ge/DemoCutTubs/RMax            = 20 mm
d:Ge/DemoCutTubs/HL              = 30 mm
d:Ge/DemoCutTubs/SPhi            = 0 rad
d:Ge/DemoCutTubs/DPhi            = 1.5 rad
uv:Ge/DemoCutTubs/LowNorm         = 3 0. -0.7 -0.71
uv:Ge/DemoCutTubs/HighNorm        = 3 0.7 0. 0.71
s:Ge/DemoCutTubs/Color            = "skyblue"
s:Ge/DemoCutTubs/DrawingStyle     = "FullWireFrame"

# Cons
s:Ge/DemoCons/Type                = "G4Cons"
s:Ge/DemoCons/Parent              = "World"
s:Ge/DemoCons/Material            = "Air"
d:Ge/DemoCons/TransX              = 0 cm
d:Ge/DemoCons/TransY              = 0 cm
d:Ge/DemoCons/TransZ              = 36 cm
d:Ge/DemoCons/RotX                = 0 deg
d:Ge/DemoCons/RotY                = 0 deg
d:Ge/DemoCons/RotZ                = 0 deg
d:Ge/DemoCons/RMin1               = 5 mm
d:Ge/DemoCons/RMax1               = 10 mm
d:Ge/DemoCons/RMin2               = 20 mm
d:Ge/DemoCons/RMax2               = 25 mm
d:Ge/DemoCons/HL                  = 40 mm
d:Ge/DemoCons/SPhi                = 0 rad
d:Ge/DemoCons/DPhi                = 1.34 rad
s:Ge/DemoCons/Color                = "red"
s:Ge/DemoCons/DrawingStyle         = "FullWireFrame"

# Para
s:Ge/DemoPara/Type                = "G4Para"
s:Ge/DemoPara/Parent              = "World"
s:Ge/DemoPara/Material            = "Air"
d:Ge/DemoPara/TransX              = 0 cm
d:Ge/DemoPara/TransY              = 0 cm
d:Ge/DemoPara/TransZ              = 48 cm
d:Ge/DemoPara/RotX                = 0 deg
d:Ge/DemoPara/RotY                = 0 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/DemoPara/RotZ           = 0 deg
d:Ge/DemoPara/HLX           = 30. mm
d:Ge/DemoPara/HLY           = 40. mm
d:Ge/DemoPara/HLZ           = 60. mm
d:Ge/DemoPara/Alpha         = 0.3 rad
d:Ge/DemoPara/Theta         = 0 rad
d:Ge/DemoPara/Phi           = 0 rad
s:Ge/DemoPara/Color         = "magenta"

# Trd
s:Ge/DemoTrd/Type           = "G4Trd"
s:Ge/DemoTrd/Parent         = "World"
s:Ge/DemoTrd/Material       = "Air"
d:Ge/DemoTrd/TransX         = 0 cm
d:Ge/DemoTrd/TransY         = 0 cm
d:Ge/DemoTrd/TransZ         = 66 cm
d:Ge/DemoTrd/RotX           = 0 deg
d:Ge/DemoTrd/RotY           = 0 deg
d:Ge/DemoTrd/RotZ           = 0 deg
d:Ge/DemoTrd/HLX1           = 30. mm
d:Ge/DemoTrd/HLX2           = 10. mm
d:Ge/DemoTrd/HLY1           = 40. mm
d:Ge/DemoTrd/HLY2           = 15. mm
d:Ge/DemoTrd/HLZ           = 60. mm
s:Ge/DemoTrd/Color         = "violet"

# RTrap - Right Angular Wedge Trapezoid
s:Ge/DemoRTrap/Type         = "G4RTrap"
s:Ge/DemoRTrap/Parent       = "World"
s:Ge/DemoRTrap/Material     = "Air"
d:Ge/DemoRTrap/TransX       = 0 cm
d:Ge/DemoRTrap/TransY       = 0 cm
d:Ge/DemoRTrap/TransZ       = 84 cm
d:Ge/DemoRTrap/RotX         = 0 deg
d:Ge/DemoRTrap/RotY         = 0 deg
d:Ge/DemoRTrap/RotZ         = 0 deg
d:Ge/DemoRTrap/LZ           = 120. mm
d:Ge/DemoRTrap/LY           = 80. mm
d:Ge/DemoRTrap/LX           = 60. mm
d:Ge/DemoRTrap/LTX          = 30. mm
s:Ge/DemoRTrap/Color        = "pink"

# GTrap - General Trapezoid
s:Ge/DemoGTrap/Type         = "G4GTrap"
s:Ge/DemoGTrap/Parent       = "World"
s:Ge/DemoGTrap/Material     = "Air"
d:Ge/DemoGTrap/TransX       = 0 cm
d:Ge/DemoGTrap/TransY       = 0 cm
d:Ge/DemoGTrap/TransZ       = 104 cm
d:Ge/DemoGTrap/RotX         = 0 deg
d:Ge/DemoGTrap/RotY         = 0 deg
d:Ge/DemoGTrap/RotZ         = 0 deg
d:Ge/DemoGTrap/HLZ          = 60. mm
d:Ge/DemoGTrap/Theta        = 20 deg
d:Ge/DemoGTrap/Phi          = 5 deg
d:Ge/DemoGTrap/HLY1         = 40. mm
d:Ge/DemoGTrap/HLX1         = 30. mm

```

(continues on next page)

(continued from previous page)

```

d:Ge/DemoGTrap/HLX2           = 40. mm
d:Ge/DemoGTrap/Alp1           = 10 deg
d:Ge/DemoGTrap/HLY2           = 16. mm
d:Ge/DemoGTrap/HLX3           = 10. mm
d:Ge/DemoGTrap/HLX4           = 14. mm
d:Ge/DemoGTrap/Alp2           = 10 deg
s:Ge/DemoGTrap/Color           = "indigo"

# Sphere
s:Ge/DemoSphere/Type           = "TsSphere"
s:Ge/DemoSphere/Parent         = "World"
s:Ge/DemoSphere/Material       = "Air"
d:Ge/DemoSphere/TransX         = 0 cm
d:Ge/DemoSphere/TransY         = 0 cm
d:Ge/DemoSphere/TransZ         = 124 cm
d:Ge/DemoSphere/RotX           = 0 deg
d:Ge/DemoSphere/RotY           = 0 deg
d:Ge/DemoSphere/RotZ           = 0 deg
d:Ge/DemoSphere/RMin           = 100 mm
d:Ge/DemoSphere/RMax           = 120 mm
d:Ge/DemoSphere/SPhi           = 0 deg
d:Ge/DemoSphere/DPhi           = 180 deg
d:Ge/DemoSphere/STheta         = 0 deg
d:Ge/DemoSphere/DTheta         = 180 deg
s:Ge/DemoSphere/DrawingStyle    = "FullWireFrame"
s:Ge/DemoSphere/Color           = "grass"

# Orb
s:Ge/DemoOrb/Type              = "G4Orb"
s:Ge/DemoOrb/Parent            = "World"
s:Ge/DemoOrb/Material          = "Air"
d:Ge/DemoOrb/TransX            = 0 cm
d:Ge/DemoOrb/TransY            = 0 cm
d:Ge/DemoOrb/TransZ            = 150 cm
d:Ge/DemoOrb/RotX              = 0 deg
d:Ge/DemoOrb/RotY              = 0 deg
d:Ge/DemoOrb/RotZ              = 0 deg
d:Ge/DemoOrb/R                 = 100 mm
s:Ge/DemoOrb/Color              = "orange"
s:Ge/DemoOrb/DrawingStyle       = "FullWireFrame"

# Torus
s:Ge/DemoTorus/Type            = "G4Torus"
s:Ge/DemoTorus/Parent          = "World"
s:Ge/DemoTorus/Material        = "Air"
d:Ge/DemoTorus/TransX          = 0 cm
d:Ge/DemoTorus/TransY          = 0 cm
d:Ge/DemoTorus/TransZ          = 176 cm
d:Ge/DemoTorus/RotX            = 0 deg
d:Ge/DemoTorus/RotY            = 0 deg
d:Ge/DemoTorus/RotZ            = 0 deg
d:Ge/DemoTorus/RMin            = 40 mm
d:Ge/DemoTorus/RMax            = 60 mm
d:Ge/DemoTorus/RTor            = 200 mm
d:Ge/DemoTorus/SPhi            = 0 deg
d:Ge/DemoTorus/DPhi            = 90 deg
s:Ge/DemoTorus/Color            = "purple"

```

(continues on next page)



(continued from previous page)

```

s:Ge/DemoTorus/DrawingStyle      = "FullWireFrame"

# HPolycone - Hollow Polycone
s:Ge/DemoHPolycone/Type          = "G4HPolycone"
s:Ge/DemoHPolycone/Parent        = "World"
s:Ge/DemoHPolycone/Material      = "Air"
d:Ge/DemoHPolycone/TransX        = 0 cm
d:Ge/DemoHPolycone/TransY        = 0 cm
d:Ge/DemoHPolycone/TransZ        = 180 cm
d:Ge/DemoHPolycone/RotX          = 0 deg
d:Ge/DemoHPolycone/RotY          = 0 deg
d:Ge/DemoHPolycone/RotZ          = 0 deg
d:Ge/DemoHPolycone/PhiStart      = 0.25 rad
d:Ge/DemoHPolycone/PhiTotal      = 1.5 rad
dv:Ge/DemoHPolycone/RInner       = 9 0 1. 1. 1. 2. 2. 3. .5 .2 mm
dv:Ge/DemoHPolycone/ROuter       = 9 0 10 10 5 5 10 10 2 2 mm
dv:Ge/DemoHPolycone/Z            = 9 5 7 9 11 25 27 29 31 35 mm
s:Ge/DemoHPolycone/Color         = "brown"
s:Ge/DemoHPolycone/DrawingStyle  = "FullWireFrame"

# SPolycone - Solid Polycone
s:Ge/DemoSPolycone/Type          = "G4SPolycone"
s:Ge/DemoSPolycone/Parent        = "World"
s:Ge/DemoSPolycone/Material      = "Air"
d:Ge/DemoSPolycone/TransX        = 0 cm
d:Ge/DemoSPolycone/TransY        = 0 cm
d:Ge/DemoSPolycone/TransZ        = 186 cm
d:Ge/DemoSPolycone/RotX          = 0 deg
d:Ge/DemoSPolycone/RotY          = 0 deg
d:Ge/DemoSPolycone/RotZ          = 0 deg
d:Ge/DemoSPolycone/PhiStart      = 0.25 rad
d:Ge/DemoSPolycone/PhiTotal      = 1.5 rad
dv:Ge/DemoSPolycone/R            = 10 0 10 10 5 5 10 10 2 2 0 mm
dv:Ge/DemoSPolycone/Z            = 10 5 7 9 11 25 27 29 31 35 35 mm
s:Ge/DemoSPolycone/Color         = "grey"
s:Ge/DemoSPolycone/DrawingStyle  = "FullWireFrame"

# HPolyhedra - Hollow Polyhedra
s:Ge/DemoHPolyhedra/Type         = "G4HPolyhedra"
s:Ge/DemoHPolyhedra/Parent       = "World"
s:Ge/DemoHPolyhedra/Material     = "Air"
d:Ge/DemoHPolyhedra/TransX       = 0 cm
d:Ge/DemoHPolyhedra/TransY       = 0 cm
d:Ge/DemoHPolyhedra/TransZ       = 192 cm
d:Ge/DemoHPolyhedra/RotX         = 0 deg
d:Ge/DemoHPolyhedra/RotY         = 0 deg
d:Ge/DemoHPolyhedra/RotZ         = 0 deg
d:Ge/DemoHPolyhedra/PhiStart     = -0.25 rad
d:Ge/DemoHPolyhedra/PhiTotal     = 1.25 rad
i:Ge/DemoHPolyhedra/NSides       = 3
dv:Ge/DemoHPolyhedra/Z           = 7 0 5 8 13 30 32 35 mm
dv:Ge/DemoHPolyhedra/RInner      = 7 0 2 2 3 1 1 2 mm
dv:Ge/DemoHPolyhedra/ROuter      = 7 0 15 15 4 4 10 10 mm
s:Ge/DemoHPolyhedra/Color        = "blue"

# SPolyhedra - Solid Polyhedra
s:Ge/DemoSPolyhedra/Type         = "G4SPolyhedra"

```

(continues on next page)

(continued from previous page)

```

s:Ge/DemoSPolyhedra/Parent          = "World"
s:Ge/DemoSPolyhedra/Material        = "Air"
d:Ge/DemoSPolyhedra/TransX          = 0 cm
d:Ge/DemoSPolyhedra/TransY          = 0 cm
d:Ge/DemoSPolyhedra/TransZ          = 198 cm
d:Ge/DemoSPolyhedra/RotX            = 0 deg
d:Ge/DemoSPolyhedra/RotY            = 0 deg
d:Ge/DemoSPolyhedra/RotZ            = 0 deg
d:Ge/DemoSPolyhedra/PhiStart        = -0.25 rad
d:Ge/DemoSPolyhedra/PhiTotal        = 1.25 rad
i:Ge/DemoSPolyhedra/NSides          = 100
dv:Ge/DemoSPolyhedra/R              = 5 0 4 12 8 0 mm
dv:Ge/DemoSPolyhedra/Z              = 5 0 0 8 16 16 mm
s:Ge/DemoSPolyhedra/Color           = "green"

# EllipticalTube
s:Ge/DemoEllipticalTube/Type        = "G4EllipticalTube"
s:Ge/DemoEllipticalTube/Parent      = "World"
s:Ge/DemoEllipticalTube/Material    = "Air"
d:Ge/DemoEllipticalTube/TransX      = 0 cm
d:Ge/DemoEllipticalTube/TransY      = 0 cm
d:Ge/DemoEllipticalTube/TransZ      = 204 cm
d:Ge/DemoEllipticalTube/RotX        = 0 deg
d:Ge/DemoEllipticalTube/RotY        = 0 deg
d:Ge/DemoEllipticalTube/RotZ        = 0 deg
d:Ge/DemoEllipticalTube/HLX         = 5 mm
d:Ge/DemoEllipticalTube/HLY         = 10 mm
d:Ge/DemoEllipticalTube/HLZ         = 20 mm
s:Ge/DemoEllipticalTube/Color       = "yellow"
s:Ge/DemoEllipticalTube/DrawingStyle = "FullWireFrame"

# Ellipsoid
s:Ge/DemoEllipsoid/Type              = "G4Ellipsoid"
s:Ge/DemoEllipsoid/Parent            = "World"
s:Ge/DemoEllipsoid/Material          = "Air"
d:Ge/DemoEllipsoid/TransX            = 0 cm
d:Ge/DemoEllipsoid/TransY            = 0 cm
d:Ge/DemoEllipsoid/TransZ            = 210 cm
d:Ge/DemoEllipsoid/RotX              = 0 deg
d:Ge/DemoEllipsoid/RotY              = 0 deg
d:Ge/DemoEllipsoid/RotZ              = 0 deg
d:Ge/DemoEllipsoid/HLX               = 10 mm
d:Ge/DemoEllipsoid/HLY               = 20 mm
d:Ge/DemoEllipsoid/HLZ               = 50 mm
d:Ge/DemoEllipsoid/ZBottom           = -10 mm
d:Ge/DemoEllipsoid/ZTop              = 40 mm
s:Ge/DemoEllipsoid/Color             = "white"
s:Ge/DemoEllipsoid/DrawingStyle      = "FullWireFrame"

# EllipticalCone
s:Ge/DemoEllipticalCone/Type         = "G4EllipticalCone"
s:Ge/DemoEllipticalCone/Parent       = "World"
s:Ge/DemoEllipticalCone/Material     = "Air"
d:Ge/DemoEllipticalCone/TransX       = 0 cm
d:Ge/DemoEllipticalCone/TransY       = 0 cm
d:Ge/DemoEllipticalCone/TransZ       = 222 cm
d:Ge/DemoEllipticalCone/RotX         = 0 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/DemoEllipticalCone/RotY      = 0 deg
d:Ge/DemoEllipticalCone/RotZ      = 0 deg
d:Ge/DemoEllipticalCone/RMin      = 10 mm
d:Ge/DemoEllipticalCone/RMax      = 15 mm
d:Ge/DemoEllipticalCone/HLX       = .4 mm
d:Ge/DemoEllipticalCone/HLY       = .8 mm
d:Ge/DemoEllipticalCone/ZMax      = 50 mm
d:Ge/DemoEllipticalCone/ZTop      = 25 mm
s:Ge/DemoEllipticalCone/Color     = "lightblue"
s:Ge/DemoEllipticalCone/DrawingStyle = "FullWireFrame"

# Paraboloid
s:Ge/DemoParaboloid/Type          = "G4Paraboloid"
s:Ge/DemoParaboloid/Parent        = "World"
s:Ge/DemoParaboloid/Material      = "Air"
d:Ge/DemoParaboloid/TransX        = 0 cm
d:Ge/DemoParaboloid/TransY        = 0 cm
d:Ge/DemoParaboloid/TransZ        = 234 cm
d:Ge/DemoParaboloid/RotX          = 0 deg
d:Ge/DemoParaboloid/RotY          = 0 deg
d:Ge/DemoParaboloid/RotZ          = 0 deg
d:Ge/DemoParaboloid/HLZ           = 20 mm
d:Ge/DemoParaboloid/R1            = 20 mm
d:Ge/DemoParaboloid/R2            = 35 mm
s:Ge/DemoParaboloid/Color         = "skyblue"
s:Ge/DemoParaboloid/DrawingStyle  = "FullWireFrame"

# Hype
s:Ge/DemoHype/Type                = "G4Hype"
s:Ge/DemoHype/Parent              = "World"
s:Ge/DemoHype/Material            = "Air"
d:Ge/DemoHype/TransX              = 0 cm
d:Ge/DemoHype/TransY              = 0 cm
d:Ge/DemoHype/TransZ              = 248 cm
d:Ge/DemoHype/RotX                = 0 deg
d:Ge/DemoHype/RotY                = 0 deg
d:Ge/DemoHype/RotZ                = 0 deg
d:Ge/DemoHype/IR                  = 20 mm
d:Ge/DemoHype/OR                  = 30 mm
d:Ge/DemoHype/IS                  = .7 rad
d:Ge/DemoHype/OS                  = .7 rad
d:Ge/DemoHype/HLZ                 = 50 mm
s:Ge/DemoHype/Color               = "red"
s:Ge/DemoHype/DrawingStyle        = "FullWireFrame"

# Tet
s:Ge/DemoTet/Type                 = "G4Tet"
s:Ge/DemoTet/Parent               = "World"
s:Ge/DemoTet/Material             = "Air"
d:Ge/DemoTet/TransX               = 0 cm
d:Ge/DemoTet/TransY               = 0 cm
d:Ge/DemoTet/TransZ               = 262 cm
d:Ge/DemoTet/RotX                 = 0 deg
d:Ge/DemoTet/RotY                 = 0 deg
d:Ge/DemoTet/RotZ                 = 0 deg
dv:Ge/DemoTet/Anchor              = 3 0 0 17.3 mm
dv:Ge/DemoTet/P2                  = 3 0 16.3 -5.8 mm

```

(continues on next page)

(continued from previous page)

```

dv:Ge/DemoTet/P3                = 3 -14.1 -8.2 -5.8 mm
dv:Ge/DemoTet/P4                = 3 14.1 -8.2 -5.8 mm
s:Ge/DemoTet/Color              = "magenta"

# ExtrudedSolid
s:Ge/DemoExtrudedSolid/Type     = "G4ExtrudedSolid"
s:Ge/DemoExtrudedSolid/Parent   = "World"
s:Ge/DemoExtrudedSolid/Material = "Air"
d:Ge/DemoExtrudedSolid/TransX   = 0 cm
d:Ge/DemoExtrudedSolid/TransY   = 0 cm
d:Ge/DemoExtrudedSolid/TransZ   = 270 cm
d:Ge/DemoExtrudedSolid/RotX     = 0 deg
d:Ge/DemoExtrudedSolid/RotY     = 0 deg
d:Ge/DemoExtrudedSolid/RotZ     = 0 deg
dv:Ge/DemoExtrudedSolid/Polygons = 16 -30 -30 -30 30 30 30 30 -30 15 -30 15 15 -
  -15 15 -15 -30 mm
d:Ge/DemoExtrudedSolid/HLZ      = 20 mm
dv:Ge/DemoExtrudedSolid/Off1    = 2 10. 10. mm
u:Ge/DemoExtrudedSolid/Scale1   = 1.
dv:Ge/DemoExtrudedSolid/Off2    = 2 -10. -10. mm
u:Ge/DemoExtrudedSolid/Scale2   = 0.6
s:Ge/DemoExtrudedSolid/Color    = "violet"

# TwistedBox
s:Ge/DemoG4TwistedBox/Type      = "G4TwistedBox"
s:Ge/DemoG4TwistedBox/Parent    = "World"
s:Ge/DemoG4TwistedBox/Material  = "Air"
d:Ge/DemoG4TwistedBox/TransX    = 0 cm
d:Ge/DemoG4TwistedBox/TransY    = 0 cm
d:Ge/DemoG4TwistedBox/TransZ    = 284 cm
d:Ge/DemoG4TwistedBox/RotX      = 0 deg
d:Ge/DemoG4TwistedBox/RotY      = 0 deg
d:Ge/DemoG4TwistedBox/RotZ      = 0 deg
d:Ge/DemoG4TwistedBox/Twist     = 30 deg
d:Ge/DemoG4TwistedBox/HLX       = 30. mm
d:Ge/DemoG4TwistedBox/HLY       = 40. mm
d:Ge/DemoG4TwistedBox/HLZ       = 60. mm
s:Ge/DemoG4TwistedBox/Color     = "pink"
s:Ge/DemoG4TwistedBox/DrawingStyle = "FullWireFrame"

# RTwistedTrap - Right Angular Wedge Twisted Trapezoid
s:Ge/DemoRTwistedTrap/Type      = "G4RTwistedTrap"
s:Ge/DemoRTwistedTrap/Parent    = "World"
s:Ge/DemoRTwistedTrap/Material  = "Air"
d:Ge/DemoRTwistedTrap/TransX    = 0 cm
d:Ge/DemoRTwistedTrap/TransY    = 0 cm
d:Ge/DemoRTwistedTrap/TransZ    = 302 cm
d:Ge/DemoRTwistedTrap/RotX      = 0 deg
d:Ge/DemoRTwistedTrap/RotY      = 0 deg
d:Ge/DemoRTwistedTrap/RotZ      = 0 deg
d:Ge/DemoRTwistedTrap/Twist     = 30 deg
d:Ge/DemoRTwistedTrap/HLX1      = 30. mm
d:Ge/DemoRTwistedTrap/HLX2      = 40. mm
d:Ge/DemoRTwistedTrap/HLY       = 40. mm
d:Ge/DemoRTwistedTrap/HLZ       = 60. mm
s:Ge/DemoRTwistedTrap/Color     = "indigo"
s:Ge/DemoRTwistedTrap/DrawingStyle = "FullWireFrame"

```

(continues on next page)

(continued from previous page)

```

# GTwistedTrap - General Twisted Trapezoid
s:Ge/DemoGTwistedTrap/Type           = "G4GTwistedTrap"
s:Ge/DemoGTwistedTrap/Parent         = "World"
s:Ge/DemoGTwistedTrap/Material       = "Air"
d:Ge/DemoGTwistedTrap/TransX        = 0 cm
d:Ge/DemoGTwistedTrap/TransY        = 0 cm
d:Ge/DemoGTwistedTrap/TransZ        = 320 cm
d:Ge/DemoGTwistedTrap/RotX          = 0 deg
d:Ge/DemoGTwistedTrap/RotY          = 0 deg
d:Ge/DemoGTwistedTrap/RotZ          = 0 deg
d:Ge/DemoGTwistedTrap/Twist         = 30 deg
d:Ge/DemoGTwistedTrap/HLZ           = 60. mm
d:Ge/DemoGTwistedTrap/Theta         = 20 deg
d:Ge/DemoGTwistedTrap/Phi           = 5 deg
d:Ge/DemoGTwistedTrap/HLY1          = 40. mm
d:Ge/DemoGTwistedTrap/HLX1          = 30. mm
d:Ge/DemoGTwistedTrap/HLX2          = 40. mm
d:Ge/DemoGTwistedTrap/HLY2          = 16. mm
d:Ge/DemoGTwistedTrap/HLX3          = 10. mm
d:Ge/DemoGTwistedTrap/HLX4          = 14. mm
d:Ge/DemoGTwistedTrap/Alpha         = 10 deg
s:Ge/DemoGTwistedTrap/Color         = "grass"
s:Ge/DemoGTwistedTrap/DrawingStyle  = "FullWireFrame"

# TwistedTrd
s:Ge/DemoTwistedTrd/Type             = "G4TwistedTrd"
s:Ge/DemoTwistedTrd/Parent           = "World"
s:Ge/DemoTwistedTrd/Material         = "Air"
d:Ge/DemoTwistedTrd/TransX           = 0 cm
d:Ge/DemoTwistedTrd/TransY           = 0 cm
d:Ge/DemoTwistedTrd/TransZ           = 336 cm
d:Ge/DemoTwistedTrd/RotX             = 0 deg
d:Ge/DemoTwistedTrd/RotY             = 0 deg
d:Ge/DemoTwistedTrd/RotZ             = 0 deg
d:Ge/DemoTwistedTrd/HLX1             = 30. mm
d:Ge/DemoTwistedTrd/HLX2             = 10. mm
d:Ge/DemoTwistedTrd/HLY1             = 40. mm
d:Ge/DemoTwistedTrd/HLY2             = 15. mm
d:Ge/DemoTwistedTrd/HLZ             = 60. mm
d:Ge/DemoTwistedTrd/Twist            = 30 deg
s:Ge/DemoTwistedTrd/Color            = "orange"
s:Ge/DemoTwistedTrd/DrawingStyle     = "FullWireFrame"

# GenericTrap
s:Ge/DemoGenericTrap/Type            = "G4GenericTrap"
s:Ge/DemoGenericTrap/Parent          = "World"
s:Ge/DemoGenericTrap/Material        = "Air"
d:Ge/DemoGenericTrap/TransX          = 0 cm
d:Ge/DemoGenericTrap/TransY          = 0 cm
d:Ge/DemoGenericTrap/TransZ          = 350 cm
d:Ge/DemoGenericTrap/RotX            = 0 deg
d:Ge/DemoGenericTrap/RotY            = 0 deg
d:Ge/DemoGenericTrap/RotZ            = 0 deg
d:Ge/DemoGenericTrap/HLZ             = 25 mm
dv:Ge/DemoGenericTrap/Vertices       = 16 -30 -30 -30 30 30 30 30 -30 -5 -20 -20 20
→20 20 20 -20 mm

```

(continues on next page)

(continued from previous page)

```

s:Ge/DemoGenericTrap/Color          = "purple"
s:Ge/DemoGenericTrap/DrawingStyle   = "FullWireFrame"

# TwistedTubs
s:Ge/DemoTwistedTubs/Type           = "G4TwistedTubs"
s:Ge/DemoTwistedTubs/Parent         = "World"
s:Ge/DemoTwistedTubs/Material       = "Air"
d:Ge/DemoTwistedTubs/TransX         = 0 cm
d:Ge/DemoTwistedTubs/TransY         = 0 cm
d:Ge/DemoTwistedTubs/TransZ         = 358 cm
d:Ge/DemoTwistedTubs/RotX           = 0 deg
d:Ge/DemoTwistedTubs/RotY           = 0 deg
d:Ge/DemoTwistedTubs/RotZ           = 0 deg
d:Ge/DemoTwistedTubs/Twist          = 60 deg
d:Ge/DemoTwistedTubs/EndInnerRad     = 10 mm
d:Ge/DemoTwistedTubs/EndOuterRad     = 15 mm
d:Ge/DemoTwistedTubs/HLZ            = 20 mm
d:Ge/DemoTwistedTubs/Phi            = 90 deg
s:Ge/DemoTwistedTubs/Color          = "brown"
s:Ge/DemoTwistedTubs/DrawingStyle    = "FullWireFrame"

b:Ge/World/Invisible = "True"

s:Gr/MyViewA/Type = "OpenGL"
b:Ts/PauseBeforeQuit = "True"

```

## 19.15 ShapeTestWithOnlyRequiredParameters.txt

```

# Demonstrates all of the standard Geant4 solids.
# Shows only the required parameters, omitting the optional ones
# (the ones such as TransX or RMin that have default values).
# Details are given in the TOPAS User Guide
# and in chapter 4 of the Geant4 Application Developer's Guide
# http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/
# ↪ForApplicationDeveloper/html/ch04.html#sect.Geom.Solids

# Box
s:Ge/DemoBox/Type          = "TsBox"
s:Ge/DemoBox/Parent        = "World"
s:Ge/DemoBox/Material      = "Air"
d:Ge/DemoBox/HLX           = 30. mm
d:Ge/DemoBox/HLY           = 40. mm
d:Ge/DemoBox/HLZ           = 60. mm
s:Ge/DemoBox/Color         = "white"

# Tubs
s:Ge/DemoCylinder/Type     = "TsCylinder"
s:Ge/DemoCylinder/Parent   = "World"
s:Ge/DemoCylinder/Material = "Air"
d:Ge/DemoCylinder/TransZ   = 12 cm
d:Ge/DemoCylinder/RMax     = 15 mm
d:Ge/DemoCylinder/HL       = 20 mm
s:Ge/DemoCylinder/Color    = "lightblue"
s:Ge/DemoCylinder/DrawingStyle = "FullWireFrame"

```

(continues on next page)

(continued from previous page)

```

# CutTubs
s:Ge/DemoCutTubs/Type           = "G4CutTubs"
s:Ge/DemoCutTubs/Parent         = "World"
s:Ge/DemoCutTubs/Material       = "Air"
d:Ge/DemoCutTubs/TransZ         = 24 cm
d:Ge/DemoCutTubs/RMax           = 20 mm
d:Ge/DemoCutTubs/HL             = 30 mm
uv:Ge/DemoCutTubs/LowNorm       = 3 0. -0.7 -0.71
uv:Ge/DemoCutTubs/HighNorm      = 3 0.7 0. 0.71
s:Ge/DemoCutTubs/Color          = "skyblue"
s:Ge/DemoCutTubs/DrawingStyle   = "FullWireFrame"

# Cons
s:Ge/DemoCons/Type              = "G4Cons"
s:Ge/DemoCons/Parent            = "World"
s:Ge/DemoCons/Material          = "Air"
d:Ge/DemoCons/TransZ            = 36 cm
d:Ge/DemoCons/RMax1             = 10 mm
d:Ge/DemoCons/RMax2             = 25 mm
d:Ge/DemoCons/HL                = 40 mm
s:Ge/DemoCons/Color             = "red"
s:Ge/DemoCons/DrawingStyle      = "FullWireFrame"

# Para
s:Ge/DemoPara/Type              = "G4Para"
s:Ge/DemoPara/Parent            = "World"
s:Ge/DemoPara/Material          = "Air"
d:Ge/DemoPara/TransX            = 0 cm
d:Ge/DemoPara/TransY            = 0 cm
d:Ge/DemoPara/TransZ            = 48 cm
d:Ge/DemoPara/RotX              = 0 deg
d:Ge/DemoPara/RotY              = 0 deg
d:Ge/DemoPara/RotZ              = 0 deg
d:Ge/DemoPara/HLX               = 30. mm
d:Ge/DemoPara/HLY               = 40. mm
d:Ge/DemoPara/HLZ               = 60. mm
d:Ge/DemoPara/Alpha             = 0.3 rad
d:Ge/DemoPara/Theta             = 0 rad
d:Ge/DemoPara/Phi               = 0 rad
s:Ge/DemoPara/Color             = "magenta"

# Trd
s:Ge/DemoTrd/Type               = "G4Trd"
s:Ge/DemoTrd/Parent             = "World"
s:Ge/DemoTrd/Material           = "Air"
d:Ge/DemoTrd/TransX             = 0 cm
d:Ge/DemoTrd/TransY             = 0 cm
d:Ge/DemoTrd/TransZ             = 66 cm
d:Ge/DemoTrd/RotX               = 0 deg
d:Ge/DemoTrd/RotY               = 0 deg
d:Ge/DemoTrd/RotZ               = 0 deg
d:Ge/DemoTrd/HLX1               = 30. mm
d:Ge/DemoTrd/HLX2               = 10. mm
d:Ge/DemoTrd/HLY1               = 40. mm
d:Ge/DemoTrd/HLY2               = 15. mm
d:Ge/DemoTrd/HLZ               = 60. mm

```

(continues on next page)

(continued from previous page)

```

s:Ge/DemoTrd/Color          = "violet"

# RTrap - Right Angular Wedge Trapezoid
s:Ge/DemoRTrap/Type         = "G4RTrap"
s:Ge/DemoRTrap/Parent       = "World"
s:Ge/DemoRTrap/Material     = "Air"
d:Ge/DemoRTrap/TransX       = 0 cm
d:Ge/DemoRTrap/TransY       = 0 cm
d:Ge/DemoRTrap/TransZ       = 84 cm
d:Ge/DemoRTrap/RotX         = 0 deg
d:Ge/DemoRTrap/RotY         = 0 deg
d:Ge/DemoRTrap/RotZ         = 0 deg
d:Ge/DemoRTrap/LZ           = 120. mm
d:Ge/DemoRTrap/LY           = 80. mm
d:Ge/DemoRTrap/LX           = 60. mm
d:Ge/DemoRTrap/LTX          = 30. mm
s:Ge/DemoRTrap/Color        = "pink"

# GTrap - General Trapezoid
s:Ge/DemoGTrap/Type         = "G4GTrap"
s:Ge/DemoGTrap/Parent       = "World"
s:Ge/DemoGTrap/Material     = "Air"
d:Ge/DemoGTrap/TransX       = 0 cm
d:Ge/DemoGTrap/TransY       = 0 cm
d:Ge/DemoGTrap/TransZ       = 104 cm
d:Ge/DemoGTrap/RotX         = 0 deg
d:Ge/DemoGTrap/RotY         = 0 deg
d:Ge/DemoGTrap/RotZ         = 0 deg
d:Ge/DemoGTrap/HLZ          = 60. mm
d:Ge/DemoGTrap/Theta        = 20 deg
d:Ge/DemoGTrap/Phi          = 5 deg
d:Ge/DemoGTrap/HLY1         = 40. mm
d:Ge/DemoGTrap/HLX1         = 30. mm
d:Ge/DemoGTrap/HLX2         = 40. mm
d:Ge/DemoGTrap/Alp1         = 10 deg
d:Ge/DemoGTrap/HLY2         = 16. mm
d:Ge/DemoGTrap/HLX3         = 10. mm
d:Ge/DemoGTrap/HLX4         = 14. mm
d:Ge/DemoGTrap/Alp2         = 10 deg
s:Ge/DemoGTrap/Color        = "indigo"

# Sphere
s:Ge/DemoSphere/Type        = "TsSphere"
s:Ge/DemoSphere/Parent      = "World"
s:Ge/DemoSphere/Material    = "Air"
d:Ge/DemoSphere/TransZ      = 124 cm
d:Ge/DemoSphere/RMax        = 120 mm
s:Ge/DemoSphere/DrawingStyle = "FullWireFrame"
s:Ge/DemoSphere/Color       = "grass"

# Orb
s:Ge/DemoOrb/Type           = "G4Orb"
s:Ge/DemoOrb/Parent         = "World"
s:Ge/DemoOrb/Material       = "Air"
d:Ge/DemoOrb/TransX         = 0 cm
d:Ge/DemoOrb/TransY         = 0 cm
d:Ge/DemoOrb/TransZ         = 150 cm

```

(continues on next page)



(continued from previous page)

```

d:Ge/DemoOrb/RotX           = 0 deg
d:Ge/DemoOrb/RotY           = 0 deg
d:Ge/DemoOrb/RotZ           = 0 deg
d:Ge/DemoOrb/R              = 100 mm
s:Ge/DemoOrb/Color          = "orange"
s:Ge/DemoOrb/DrawingStyle    = "FullWireFrame"

# Torus
s:Ge/DemoTorus/Type          = "G4Torus"
s:Ge/DemoTorus/Parent        = "World"
s:Ge/DemoTorus/Material      = "Air"
d:Ge/DemoTorus/TransZ        = 176 cm
d:Ge/DemoTorus/RotX          = 0 deg
d:Ge/DemoTorus/RMax          = 60 mm
d:Ge/DemoTorus/RTor          = 200 mm
s:Ge/DemoTorus/Color         = "purple"
s:Ge/DemoTorus/DrawingStyle  = "FullWireFrame"

# HPolycone - Hollow Polycone
s:Ge/DemoHPolycone/Type      = "G4HPolycone"
s:Ge/DemoHPolycone/Parent    = "World"
s:Ge/DemoHPolycone/Material  = "Air"
d:Ge/DemoHPolycone/TransZ    = 180 cm
dv:Ge/DemoHPolycone/RInner   = 9 0 1. 1. 1. 2. 2. 3. .5 .2 mm
dv:Ge/DemoHPolycone/ROuter   = 9 0 10 10 5 5 10 10 2 2 mm
dv:Ge/DemoHPolycone/Z        = 9 5 7 9 11 25 27 29 31 35 mm
s:Ge/DemoHPolycone/Color     = "brown"
s:Ge/DemoHPolycone/DrawingStyle = "FullWireFrame"

# SPolycone - Solid Polycone
s:Ge/DemoSPolycone/Type      = "G4SPolycone"
s:Ge/DemoSPolycone/Parent    = "World"
s:Ge/DemoSPolycone/Material  = "Air"
d:Ge/DemoSPolycone/TransZ    = 186 cm
dv:Ge/DemoSPolycone/R        = 10 0 10 10 5 5 10 10 2 2 0 mm
dv:Ge/DemoSPolycone/Z        = 10 5 7 9 11 25 27 29 31 35 35 mm
s:Ge/DemoSPolycone/Color     = "grey"
s:Ge/DemoSPolycone/DrawingStyle = "FullWireFrame"

# HPolyhedra - Hollow Polyhedra
s:Ge/DemoHPolyhedra/Type     = "G4HPolyhedra"
s:Ge/DemoHPolyhedra/Parent   = "World"
s:Ge/DemoHPolyhedra/Material = "Air"
d:Ge/DemoHPolyhedra/TransZ   = 192 cm
i:Ge/DemoHPolyhedra/NSides   = 3
dv:Ge/DemoHPolyhedra/Z       = 7 0 5 8 13 30 32 35 mm
dv:Ge/DemoHPolyhedra/RInner  = 7 0 2 2 3 1 1 2 mm
dv:Ge/DemoHPolyhedra/ROuter  = 7 0 15 15 4 4 10 10 mm
s:Ge/DemoHPolyhedra/Color    = "blue"

# SPolyhedra - Solid Polyhedra
s:Ge/DemoSPolyhedra/Type     = "G4SPolyhedra"
s:Ge/DemoSPolyhedra/Parent   = "World"
s:Ge/DemoSPolyhedra/Material = "Air"
d:Ge/DemoSPolyhedra/TransZ   = 198 cm
i:Ge/DemoSPolyhedra/NSides   = 100
dv:Ge/DemoSPolyhedra/R       = 5 0 4 12 8 0 mm

```

(continues on next page)

(continued from previous page)

```

dv:Ge/DemoSPolyhedra/Z           = 5 0 0 8 16 16 mm
s:Ge/DemoSPolyhedra/Color        = "green"

# EllipticalTube
s:Ge/DemoEllipticalTube/Type      = "G4EllipticalTube"
s:Ge/DemoEllipticalTube/Parent    = "World"
s:Ge/DemoEllipticalTube/Material  = "Air"
d:Ge/DemoEllipticalTube/TransZ    = 203 cm
d:Ge/DemoEllipticalTube/HLX       = 5 mm
d:Ge/DemoEllipticalTube/HLY       = 10 mm
d:Ge/DemoEllipticalTube/HLZ       = 20 mm
s:Ge/DemoEllipticalTube/Color     = "yellow"
s:Ge/DemoEllipticalTube/DrawingStyle = "FullWireFrame"

# Ellipsoid
s:Ge/DemoEllipsoid/Type           = "G4Ellipsoid"
s:Ge/DemoEllipsoid/Parent         = "World"
s:Ge/DemoEllipsoid/Material       = "Air"
d:Ge/DemoEllipsoid/TransZ        = 210 cm
d:Ge/DemoEllipsoid/HLX           = 10 mm
d:Ge/DemoEllipsoid/HLY           = 20 mm
d:Ge/DemoEllipsoid/HLZ           = 50 mm
s:Ge/DemoEllipsoid/Color         = "white"
s:Ge/DemoEllipsoid/DrawingStyle   = "FullWireFrame"

# EllipticalCone
s:Ge/DemoEllipticalCone/Type      = "G4EllipticalCone"
s:Ge/DemoEllipticalCone/Parent    = "World"
s:Ge/DemoEllipticalCone/Material  = "Air"
d:Ge/DemoEllipticalCone/TransZ    = 222 cm
d:Ge/DemoEllipticalCone/HLX       = .4 mm
d:Ge/DemoEllipticalCone/HLY       = .8 mm
d:Ge/DemoEllipticalCone/ZMax      = 50 mm
s:Ge/DemoEllipticalCone/Color     = "lightblue"
s:Ge/DemoEllipticalCone/DrawingStyle = "FullWireFrame"

# Paraboloid
s:Ge/DemoParaboloid/Type          = "G4Paraboloid"
s:Ge/DemoParaboloid/Parent        = "World"
s:Ge/DemoParaboloid/Material      = "Air"
d:Ge/DemoParaboloid/TransZ        = 234 cm
d:Ge/DemoParaboloid/HLZ           = 20 mm
d:Ge/DemoParaboloid/R1            = 20 mm
d:Ge/DemoParaboloid/R2            = 35 mm
s:Ge/DemoParaboloid/Color         = "skyblue"
s:Ge/DemoParaboloid/DrawingStyle   = "FullWireFrame"

# Hype
s:Ge/DemoHype/Type                = "G4Hype"
s:Ge/DemoHype/Parent              = "World"
s:Ge/DemoHype/Material            = "Air"
d:Ge/DemoHype/TransZ              = 248 cm
d:Ge/DemoHype/OR                  = 30 mm
d:Ge/DemoHype/OS                  = .7 rad
d:Ge/DemoHype/HLZ                 = 50 mm
s:Ge/DemoHype/Color               = "red"
s:Ge/DemoHype/DrawingStyle        = "FullWireFrame"

```

(continues on next page)

(continued from previous page)

```

# Tet
s:Ge/DemoTet/Type           = "G4Tet"
s:Ge/DemoTet/Parent         = "World"
s:Ge/DemoTet/Material       = "Air"
d:Ge/DemoTet/TransZ         = 262 cm
dv:Ge/DemoTet/Anchor        = 3 0 0 17.3 mm
dv:Ge/DemoTet/P2            = 3 0 16.3 -5.8 mm
dv:Ge/DemoTet/P3            = 3 -14.1 -8.2 -5.8 mm
dv:Ge/DemoTet/P4            = 3 14.1 -8.2 -5.8 mm
s:Ge/DemoTet/Color          = "magenta"

# ExtrudedSolid
s:Ge/DemoExtrudedSolid/Type = "G4ExtrudedSolid"
s:Ge/DemoExtrudedSolid/Parent = "World"
s:Ge/DemoExtrudedSolid/Material = "Air"
d:Ge/DemoExtrudedSolid/TransZ = 270 cm
dv:Ge/DemoExtrudedSolid/Polygons = 16 -30 -30 -30 30 30 30 30 -30 15 -30 15 15 -
→15 15 -15 -30 mm
d:Ge/DemoExtrudedSolid/HLZ    = 20 mm
dv:Ge/DemoExtrudedSolid/Off1  = 2 10. 10. mm
u:Ge/DemoExtrudedSolid/Scale1 = 1.
dv:Ge/DemoExtrudedSolid/Off2  = 2 -10. -10. mm
u:Ge/DemoExtrudedSolid/Scale2 = 0.6
s:Ge/DemoExtrudedSolid/Color  = "violet"

# TwistedBox
s:Ge/DemoG4TwistedBox/Type    = "G4TwistedBox"
s:Ge/DemoG4TwistedBox/Parent  = "World"
s:Ge/DemoG4TwistedBox/Material = "Air"
d:Ge/DemoG4TwistedBox/TransZ  = 284 cm
d:Ge/DemoG4TwistedBox/Twist   = 30 deg
d:Ge/DemoG4TwistedBox/HLX     = 30. mm
d:Ge/DemoG4TwistedBox/HLY     = 40. mm
d:Ge/DemoG4TwistedBox/HLZ     = 60. mm
s:Ge/DemoG4TwistedBox/Color   = "pink"
s:Ge/DemoG4TwistedBox/DrawingStyle = "FullWireFrame"

# RTwistedTrap - Right Angular Wedge Twisted Trapezoid
s:Ge/DemoRTwistedTrap/Type    = "G4RTwistedTrap"
s:Ge/DemoRTwistedTrap/Parent  = "World"
s:Ge/DemoRTwistedTrap/Material = "Air"
d:Ge/DemoRTwistedTrap/TransZ  = 302 cm
d:Ge/DemoRTwistedTrap/Twist   = 30 deg
d:Ge/DemoRTwistedTrap/HLX1    = 30. mm
d:Ge/DemoRTwistedTrap/HLX2    = 40. mm
d:Ge/DemoRTwistedTrap/HLY     = 40. mm
d:Ge/DemoRTwistedTrap/HLZ     = 60. mm
s:Ge/DemoRTwistedTrap/Color   = "indigo"
s:Ge/DemoRTwistedTrap/DrawingStyle = "FullWireFrame"

# GTwistedTrap - General Twisted Trapezoid
s:Ge/DemoGTwistedTrap/Type    = "G4GTwistedTrap"
s:Ge/DemoGTwistedTrap/Parent  = "World"
s:Ge/DemoGTwistedTrap/Material = "Air"
d:Ge/DemoGTwistedTrap/TransZ  = 320 cm
d:Ge/DemoGTwistedTrap/Twist   = 30 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/DemoGTwistedTrap/HLZ           = 60. mm
d:Ge/DemoGTwistedTrap/Theta         = 20 deg
d:Ge/DemoGTwistedTrap/Phi           = 5 deg
d:Ge/DemoGTwistedTrap/HLY1          = 40. mm
d:Ge/DemoGTwistedTrap/HLX1          = 30. mm
d:Ge/DemoGTwistedTrap/HLX2          = 40. mm
d:Ge/DemoGTwistedTrap/HLY2          = 16. mm
d:Ge/DemoGTwistedTrap/HLX3          = 10. mm
d:Ge/DemoGTwistedTrap/HLX4          = 14. mm
d:Ge/DemoGTwistedTrap/Alpha         = 10 deg
s:Ge/DemoGTwistedTrap/Color         = "grass"
s:Ge/DemoGTwistedTrap/DrawingStyle  = "FullWireFrame"

# TwistedTrd
s:Ge/DemoTwistedTrd/Type             = "G4TwistedTrd"
s:Ge/DemoTwistedTrd/Parent           = "World"
s:Ge/DemoTwistedTrd/Material         = "Air"
d:Ge/DemoTwistedTrd/TransZ           = 336 cm
d:Ge/DemoTwistedTrd/HLX1            = 30. mm
d:Ge/DemoTwistedTrd/HLX2            = 10. mm
d:Ge/DemoTwistedTrd/HLY1            = 40. mm
d:Ge/DemoTwistedTrd/HLY2            = 15. mm
d:Ge/DemoTwistedTrd/HLZ             = 60. mm
d:Ge/DemoTwistedTrd/Twist            = 30 deg
s:Ge/DemoTwistedTrd/Color            = "orange"
s:Ge/DemoTwistedTrd/DrawingStyle     = "FullWireFrame"

# GenericTrap
s:Ge/DemoGenericTrap/Type            = "G4GenericTrap"
s:Ge/DemoGenericTrap/Parent          = "World"
s:Ge/DemoGenericTrap/Material        = "Air"
d:Ge/DemoGenericTrap/TransZ          = 350 cm
d:Ge/DemoGenericTrap/HLZ             = 25 mm
dv:Ge/DemoGenericTrap/Vertices       = 16 -30 -30 30 30 30 30 -30 -5 -20 -20 20_
↪20 20 20 -20 mm
s:Ge/DemoGenericTrap/Color           = "purple"
s:Ge/DemoGenericTrap/DrawingStyle    = "FullWireFrame"

# TwistedTubs
s:Ge/DemoTwistedTubs/Type            = "G4TwistedTubs"
s:Ge/DemoTwistedTubs/Parent          = "World"
s:Ge/DemoTwistedTubs/Material        = "Air"
d:Ge/DemoTwistedTubs/TransZ          = 358 cm
d:Ge/DemoTwistedTubs/Twist           = 60 deg
d:Ge/DemoTwistedTubs/EndInnerRad     = 10 mm
d:Ge/DemoTwistedTubs/EndOuterRad     = 15 mm
d:Ge/DemoTwistedTubs/HLZ             = 20 mm
d:Ge/DemoTwistedTubs/Phi             = 90 deg
s:Ge/DemoTwistedTubs/Color           = "brown"
s:Ge/DemoTwistedTubs/DrawingStyle    = "FullWireFrame"

b:Ge/World/Invisible = "True"

s:Gr/MyViewA/Type = "HepRep"

```

## 19.16 Spectrum.txt

```
# Demonstrates use of EnergySpectrum.
# Resulting particles can be seen in ASCIIOutput.phsp

s:Ge/World/Material = "Vacuum"

s:So/Example/BeamEnergySpectrumType      = "Continuous"
dv:So/Example/BeamEnergySpectrumValues   = 3 50. 100. 150. MeV
uv:So/Example/BeamEnergySpectrumWeights  = 3 .20 .60 .20

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "Vacuum"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm

s:Sc/PhaseSpaceAtVacFilm/Quantity          = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole   = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface           = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType        = "ASCII" # ASCII, Binary or Limited
s:Sc/PhaseSpaceAtVacFilm/OutputFile         = "ASCIIOutput"
s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite"

s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX     = 10. cm
d:So/Example/BeamPositionCutoffY     = 10. cm
d:So/Example/BeamPositionSpreadX     = 0.65 cm
d:So/Example/BeamPositionSpreadY     = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX      = 90. deg
d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 1000

i:Ts/ShowHistoryCountAtInterval = 10
```

## 19.17 TwoBeams.txt

```
# Demonstrates ability to have more than one particle source

s:Ge/World/Material = "Vacuum"

# Second beam component
s:Ge/XRayTube/Parent = "World"
s:Ge/XRayTube/Type   = "Group"
d:Ge/XRayTube/TransX = 200. cm
```

(continues on next page)

(continued from previous page)

```

d:Ge/XRayTube/TransY = 0. m
d:Ge/XRayTube/TransZ = 0. m
d:Ge/XRayTube/RotX   = 0. deg
d:Ge/XRayTube/RotY   = 90. deg
d:Ge/XRayTube/RotZ   = 0. deg

# Second beam source
s:So/Imaging/Type      = "Beam"
s:So/Imaging/Component = "XRayTube"
s:So/Imaging/BeamParticle = "gamma"
d:So/Imaging/BeamEnergy = 100. keV
u:So/Imaging/BeamEnergySpread = 0.
s:So/Imaging/BeamPositionCutoffShape = "Ellipse"
s:So/Imaging/BeamPositionDistribution = "Flat"
d:So/Imaging/BeamPositionCutoffX = 7. cm
d:So/Imaging/BeamPositionCutoffY = 21. cm
s:So/Imaging/BeamAngularDistribution = "None"
i:So/Imaging/NumberOfHistoriesInRun = 5

s:So/Therapy/Type      = "Beam"
s:So/Therapy/Component = "BeamPosition"
s:So/Therapy/BeamParticle = "proton"
d:So/Therapy/BeamEnergy = 169.23 MeV
u:So/Therapy/BeamEnergySpread = 0.757504
s:So/Therapy/BeamPositionDistribution = "Gaussian"
s:So/Therapy/BeamPositionCutoffShape = "Ellipse"
d:So/Therapy/BeamPositionCutoffX = 10. cm
d:So/Therapy/BeamPositionCutoffY = 10. cm
d:So/Therapy/BeamPositionSpreadX = 0.65 cm
d:So/Therapy/BeamPositionSpreadY = 0.65 cm
s:So/Therapy/BeamAngularDistribution = "Gaussian"
d:So/Therapy/BeamAngularCutoffX = 90. deg
d:So/Therapy/BeamAngularCutoffY = 90. deg
d:So/Therapy/BeamAngularSpreadX = 0.0032 rad
d:So/Therapy/BeamAngularSpreadY = 0.0032 rad
i:So/Therapy/NumberOfHistoriesInRun = 10

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 1024
i:Gr/ViewA/WindowSizeY = 768
b:Gr/ViewA/IncludeAxes = "True"
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi        = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 2.

b:Ts/PauseBeforeQuit = "True"

```

## 19.18 VolumetricSource.txt

```

# Demonstrates use of Volumetric Source
# Source is a group component,
# allowing many different shapes to create a single source.

```

(continues on next page)

(continued from previous page)

```

# Shapes do not even have to be contiguous.
# Several offsets are included as a full test of geometry.
# World is set to lead so that particles stop as
# soon as they leave the source.
# This makes it easy to see particle starting positions

s:Ge/World/Material = "G4_Pb"

s:Ge/OffsetGroup/Type      = "Group"
s:Ge/OffsetGroup/Parent    = "World"
dc:Ge/OffsetGroup/RotX     = 0. deg
dc:Ge/OffsetGroup/TransY   = 2. m

s:Ge/ActiveSource/Type     = "Group"
s:Ge/ActiveSource/Parent   = "OffsetGroup"
dc:Ge/OffsetGroup/RotZ     = 0. deg
dc:Ge/ActiveSource/TransZ  = 1. m

s:Ge/SourceCylinder/Type   = "TsCylinder"
s:Ge/SourceCylinder/Parent = "ActiveSource"
sc:Ge/SourceCylinder/Material = "G4_Pb"
dc:Ge/SourceCylinder/RMax  = 50.0 mm
dc:Ge/SourceCylinder/HL    = 200.0 mm

s:Ge/Inner/Type           = "TsCylinder"
s:Ge/Inner/Parent         = "SourceCylinder"
s:Ge/Inner/Material       = "G4_Ir"
d:Ge/Inner/RMax           = 20.0 mm
d:Ge/Inner/HL             = 150.0 mm

s:Ge/SourceCapPlusZ/Type   = "TsCylinder"
s:Ge/SourceCapPlusZ/Parent = "ActiveSource"
sc:Ge/SourceCapPlusZ/Material = "G4_Ir"
dc:Ge/SourceCapPlusZ/RMax   = 50.0 mm
dc:Ge/SourceCapPlusZ/STheta = 0 deg
dc:Ge/SourceCapPlusZ/DTheta = 90 deg
dc:Ge/SourceCapPlusZ/TransZ = 1050. mm
dc:Ge/SourceCapPlusZ/HL    = 50.0 mm

s:Ge/SourceCapMinusZ/Type  = "TsSphere"
s:Ge/SourceCapMinusZ/Parent = "ActiveSource"
sc:Ge/SourceCapMinusZ/Material = "G4_Ir"
dc:Ge/SourceCapMinusZ/RMax   = 50.0 mm
dc:Ge/SourceCapMinusZ/STheta = 90 deg
dc:Ge/SourceCapMinusZ/DTheta = 90 deg
dc:Ge/SourceCapMinusZ/TransZ = -200. mm

s:Ge/SourceExtraBit/Type   = "TsSphere"
s:Ge/SourceExtraBit/Parent = "ActiveSource"
sc:Ge/SourceExtraBit/Material = "G4_Ir"
dc:Ge/SourceExtraBit/RMax   = 50.0 mm
dc:Ge/SourceExtraBit/TransY = -2000. mm
dc:Ge/SourceExtraBit/TransZ = 500. mm

s:So/Example/Type          = "Volumetric"
s:So/Example/Component      = "ActiveSource"
sc:So/Example/ActiveMaterial = "G4_Ir"

```

(continues on next page)

(continued from previous page)

```

s:So/Example/BeamParticle           = "e-"
ic:So/Example/NumberOfHistoriesInRun = 500
b:So/Example/RecursivelyIncludeChildren = "True"
ic:So/Example/MaxNumberOfPointsToSample = 10000000

s:So/Example/BeamEnergySpectrumType = "Continuous"
dv:So/Example/BeamEnergySpectrumValues = 117 0.0614 0.0615 0.0616 0.0629 0.063 0.
→0631 0.065 0.0651 0.0652 0.0667 0.0668 0.0669 0.071 0.0711 0.0712 0.0713 0.0714 0.
→0715 0.0733 0.0734 0.0735 0.0753 0.0754 0.0755 0.0756 0.0757 0.0758 0.0777 0.0778 0.
→0779 0.1103 0.1104 0.1105 0.1362 0.1363 0.1364 0.1769 0.177 0.1771 0.2012 0.2013 0.
→2014 0.2057 0.2058 0.2059 0.2802 0.2803 0.2804 0.2832 0.2833 0.2834 0.2959 0.296 0.
→2961 0.3084 0.3085 0.3086 0.3164 0.3165 0.3166 0.3291 0.3292 0.3293 0.3744 0.3745 0.
→3746 0.4164 0.4165 0.4166 0.4204 0.4205 0.4206 0.468 0.4681 0.4682 0.4845 0.4846 0.
→4847 0.4852 0.4853 0.4854 0.489 0.4891 0.4892 0.5885 0.5886 0.5887 0.5934 0.5935 0.
→5936 0.5993 0.5994 0.5995 0.6043 0.6044 0.6045 0.6124 0.6125 0.6126 0.7038 0.7039 0.
→704 0.7657 0.7658 0.7659 0.8844 0.8845 0.8846 1.0614 1.0615 1.0616 1.0898 1.0899 1.
→089 1.3781 1.3782 1.3783 MeV

uv:So/Example/BeamEnergySpectrumWeightsUnscaled = 117
1e-08 0.412 1e-08 1e-08 0.7039 1e-08 1e-08 1.1309 1e-08 1e-08 1.9178 1e-08 1e-08 0.
→0827 1e-08 1e-08 0.16 1e-08 1e-08 0.056 1e-08 1e-08 0.2292 1e-08 1e-08 0.4408 1e-08
→1e-08 0.157 1e-08 1e-08 0.0042 1e-08 1e-08 0.086 1e-08 1e-08 0.0018 1e-08 1e-08 0.
→1624 1e-08 1e-08 1.1468 1e-08 1e-08 0.0039 1e-08 1e-08 0.0913 1e-08 1e-08 12.3498
→1e-08 1e-08 12.7626 1e-08 1e-08 35.566 1e-08 1e-08 0.006 1e-08 1e-08 0.2493 1e-08
→1e-08 0.2877 1e-08 1e-08 0.0237 1e-08 1e-08 20.5587 1e-08 1e-08 1.0942 1e-08 1e-08
→0.001 1e-08 1e-08 0.1504 1e-08 1e-08 1.9423 1e-08 1e-08 0.0181 1e-08 1e-08 0.0017
→1e-08 1e-08 3.5361 1e-08 1e-08 2.2962 1e-08 1e-08 0.0018 1e-08 1e-08 0.0006 1e-08
→1e-08 0.1251 1e-08 1e-08 0.0228 1e-08 1e-08 0.0005 1e-08 1e-08 0.0005 1e-8

# Scale these values to give total weight of 1.
uv:So/Example/BeamEnergySpectrumWeights = .0102269 * So/Example/
→BeamEnergySpectrumWeightsUnscaled

i:Ts/ShowHistoryCountAtInterval = 100

s:Gr/ViewA/Type = "OpenGL"
b:Gr/ViewA/IncludeStepPoints = "True"

```



## 20.1 ColorByCreatorProcess.txt

```
# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
```

(continues on next page)

(continued from previous page)

```

s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX      = 2. m
d:Ge/Film1/HLY      = 2. m
d:Ge/Film1/HLZ      = .1 m
d:Ge/Film1/TransX   = 0. m
d:Ge/Film1/TransY   = 0. m
d:Ge/Film1/TransZ   = 1. m
d:Ge/Film1/RotX     = 0. deg
d:Ge/Film1/RotY     = 0. deg
d:Ge/Film1/RotZ     = 0. deg
i:Ge/Film1/XBins    = 2
i:Ge/Film1/YBins    = 2
i:Ge/Film1/ZBins    = 2

s:Ge/Film2/Type      = "TsBox"
s:Ge/Film2/Parent    = "Outer"
s:Ge/Film2/Material  = "G4_WATER"
d:Ge/Film2/HLX      = 3. m
d:Ge/Film2/HLY      = 3. m
d:Ge/Film2/HLZ      = .1 m
d:Ge/Film2/TransX   = 0. m
d:Ge/Film2/TransY   = 0. m
d:Ge/Film2/TransZ   = -1. m
d:Ge/Film2/RotX     = 0. deg
d:Ge/Film2/RotY     = 0. deg
d:Ge/Film2/RotZ     = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
s:Gr/MyOGL/ColorBy   = "CreatorProcess"
sv:Gr/MyOGL/ColorByCreatorProcessNames = 10 "primary" "protonInelastic" "phot"
↳ "compt" "eIoni" "eBrem" "hadElastic" "hIoni" "neutronInelastic" "nCapture"
sv:Gr/MyOGL/ColorByCreatorProcessColors = 10 "blue" "purple" "green" "red" "yellow"
↳ "teal" "fuchsia" "olive" "maroon" "aqua"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

(continues on next page)

(continued from previous page)

```
Ts/UseQt = "False"
```

## 20.2 ColorByEnergy.txt

```
# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m
d:Ge/Film1/HLZ     = .1 m
d:Ge/Film1/TransX  = 0. m
d:Ge/Film1/TransY  = 0. m
d:Ge/Film1/TransZ  = 1. m
d:Ge/Film1/RotX    = 0. deg
d:Ge/Film1/RotY    = 0. deg
d:Ge/Film1/RotZ    = 0. deg
i:Ge/Film1/XBins    = 2
i:Ge/Film1/YBins    = 2
i:Ge/Film1/ZBins    = 2

s:Ge/Film2/Type    = "TsBox"
s:Ge/Film2/Parent  = "Outer"
s:Ge/Film2/Material = "G4_WATER"
d:Ge/Film2/HLX     = 3. m
d:Ge/Film2/HLY     = 3. m
```

(continues on next page)

(continued from previous page)

```

d:Ge/Film2/HLZ      = .1 m
d:Ge/Film2/TransX   = 0. m
d:Ge/Film2/TransY   = 0. m
d:Ge/Film2/TransZ   = -1. m
d:Ge/Film2/RotX     = 0. deg
d:Ge/Film2/RotY     = 0. deg
d:Ge/Film2/RotZ     = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
s:Gr/MyOGL/ColorBy   = "Energy"
dv:Gr/MyOGL/ColorByEnergyRanges = 4 1. 10. 250. 398. MeV
sv:Gr/MyOGL/ColorByEnergyColors = 5 "red" "yellow" "green" "blue" "white"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.3 ColorByGeneration.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m
d:Ge/Film1/HLZ     = .1 m
d:Ge/Film1/TransX  = 0. m
d:Ge/Film1/TransY  = 0. m
d:Ge/Film1/TransZ  = 1. m
d:Ge/Film1/RotX    = 0. deg
d:Ge/Film1/RotY    = 0. deg
d:Ge/Film1/RotZ    = 0. deg
i:Ge/Film1/XBins   = 2
i:Ge/Film1/YBins   = 2
i:Ge/Film1/ZBins   = 2

s:Ge/Film2/Type    = "TsBox"
s:Ge/Film2/Parent  = "Outer"
s:Ge/Film2/Material = "G4_WATER"
d:Ge/Film2/HLX     = 3. m
d:Ge/Film2/HLY     = 3. m
d:Ge/Film2/HLZ     = .1 m
d:Ge/Film2/TransX  = 0. m
d:Ge/Film2/TransY  = 0. m
d:Ge/Film2/TransZ  = -1. m
d:Ge/Film2/RotX    = 0. deg
d:Ge/Film2/RotY    = 0. deg
d:Ge/Film2/RotZ    = 0. deg

s:Gr/MyOGL/Type    = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
s:Gr/MyOGL/ColorBy = "Generation"
sv:Gr/MyOGL/ColorByGenerationColors = 2 "green" "red"

s:So/Example/Type    = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504

```

(continues on next page)

(continued from previous page)

```

s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX     = 10. cm
d:So/Example/BeamPositionCutoffY     = 10. cm
d:So/Example/BeamPositionSpreadX     = 0.65 cm
d:So/Example/BeamPositionSpreadY     = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX      = 90. deg
d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.4 ColorByMomentum.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m
d:Ge/Film1/HLZ     = .1 m
d:Ge/Film1/TransX  = 0. m

```

(continues on next page)

(continued from previous page)

```

d:Ge/Film1/TransY    = 0. m
d:Ge/Film1/TransZ    = 1. m
d:Ge/Film1/RotX      = 0. deg
d:Ge/Film1/RotY      = 0. deg
d:Ge/Film1/RotZ      = 0. deg
i:Ge/Film1/XBins     = 2
i:Ge/Film1/YBins     = 2
i:Ge/Film1/ZBins     = 2

s:Ge/Film2/Type       = "TsBox"
s:Ge/Film2/Parent     = "Outer"
s:Ge/Film2/Material   = "G4_WATER"
d:Ge/Film2/HLX        = 3. m
d:Ge/Film2/HLY        = 3. m
d:Ge/Film2/HLZ        = .1 m
d:Ge/Film2/TransX     = 0. m
d:Ge/Film2/TransY     = 0. m
d:Ge/Film2/TransZ     = -1. m
d:Ge/Film2/RotX       = 0. deg
d:Ge/Film2/RotY       = 0. deg
d:Ge/Film2/RotZ       = 0. deg

s:Gr/MyOGL/Type       = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX  = 0
i:Gr/MyOGL/WindowPosY  = 0
s:Gr/MyOGL/ColorBy     = "Momentum"
dv:Gr/MyOGL/ColorByMomentumRanges = 4 1. 10. 450. 1000. MeV
sv:Gr/MyOGL/ColorByMomentumColors   = 5 "red" "yellow" "green" "blue" "white"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.5 ColorByOriginComponent.txt

```
# Demonstrates changing color scheme of trajectories.
# Trajectories that originate in the component named Outer
# or any of its subcomponents will be colored red.
# All others will be colored grey.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m
d:Ge/Film1/HLZ     = .1 m
d:Ge/Film1/TransX  = 0. m
d:Ge/Film1/TransY  = 0. m
d:Ge/Film1/TransZ  = 1. m
d:Ge/Film1/RotX    = 0. deg
d:Ge/Film1/RotY    = 0. deg
d:Ge/Film1/RotZ    = 0. deg
i:Ge/Film1/XBins    = 2
i:Ge/Film1/YBins    = 2
i:Ge/Film1/ZBins    = 2

s:Ge/Film2/Type    = "TsBox"
s:Ge/Film2/Parent  = "Outer"
s:Ge/Film2/Material = "G4_WATER"
d:Ge/Film2/HLX     = 3. m
d:Ge/Film2/HLY     = 3. m
d:Ge/Film2/HLZ     = .1 m
d:Ge/Film2/TransX  = 0. m
d:Ge/Film2/TransY  = 0. m
```

(continues on next page)



(continued from previous page)

```

d:Ge/Film2/TransZ      = -1. m
d:Ge/Film2/RotX        = 0. deg
d:Ge/Film2/RotY        = 0. deg
d:Ge/Film2/RotZ        = 0. deg

s:Gr/MyOGL/Type        = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX  = 0
i:Gr/MyOGL/WindowPosY  = 0
s:Gr/MyOGL/ColorBy     = "OriginComponent"
sv:Gr/MyOGL/ColorByOriginComponentNames = 5 "World" "Outer" "Film1" "Film2" "Box"
sv:Gr/MyOGL/ColorByOriginComponentColors = 5 "Red" "Blue" "Yellow" "Pink" "Green"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy      = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.6 ColorByOriginComponentOrSubComponent.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories that originate in the component named Outer
# or any of its subcomponents will be colored red.
# All others will be colored grey.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/Outer/Type      = "TsBox"
s:Ge/Outer/Parent    = "World"
s:Ge/Outer/Material  = "G4_AIR"
d:Ge/Outer/HLX       = 3.2 m
d:Ge/Outer/HLY       = 3.2 m
d:Ge/Outer/HLZ       = 2. m
d:Ge/Outer/TransX     = 0. m
d:Ge/Outer/TransY     = 0. m
d:Ge/Outer/TransZ     = 1. m
d:Ge/Outer/RotX       = 0. deg
d:Ge/Outer/RotY       = 0. deg
d:Ge/Outer/RotZ       = 0. deg

s:Ge/Film1/Type      = "TsBox"
s:Ge/Film1/Parent    = "Outer"
s:Ge/Film1/Material  = "G4_WATER"
d:Ge/Film1/HLX       = 2. m
d:Ge/Film1/HLY       = 2. m
d:Ge/Film1/HLZ       = .1 m
d:Ge/Film1/TransX     = 0. m
d:Ge/Film1/TransY     = 0. m
d:Ge/Film1/TransZ     = 1. m
d:Ge/Film1/RotX       = 0. deg
d:Ge/Film1/RotY       = 0. deg
d:Ge/Film1/RotZ       = 0. deg
i:Ge/Film1/XBins      = 2
i:Ge/Film1/YBins      = 2
i:Ge/Film1/ZBins      = 2

s:Ge/Film2/Type      = "TsBox"
s:Ge/Film2/Parent    = "Outer"
s:Ge/Film2/Material  = "G4_WATER"
d:Ge/Film2/HLX       = 3. m
d:Ge/Film2/HLY       = 3. m
d:Ge/Film2/HLZ       = .1 m
d:Ge/Film2/TransX     = 0. m
d:Ge/Film2/TransY     = 0. m
d:Ge/Film2/TransZ     = -1. m
d:Ge/Film2/RotX       = 0. deg
d:Ge/Film2/RotY       = 0. deg
d:Ge/Film2/RotZ       = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
s:Gr/MyOGL/ColorBy    = "OriginComponentOrSubComponentOf"
sv:Gr/MyOGL/ColorByOriginComponentNames = 3 "World" "Outer" "Box"
sv:Gr/MyOGL/ColorByOriginComponentColors = 3 "Red" "Blue" "Green"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504

```

(continues on next page)

(continued from previous page)

```

s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX     = 10. cm
d:So/Example/BeamPositionCutoffY     = 10. cm
d:So/Example/BeamPositionSpreadX     = 0.65 cm
d:So/Example/BeamPositionSpreadY     = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX      = 90. deg
d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.7 ColorByOriginVolume.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories that originate in the component named Outer
# or any of its subcomponents will be colored red.
# All others will be colored grey.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m

```

(continues on next page)

(continued from previous page)

```

d:Ge/Film1/HLZ      = .1 m
d:Ge/Film1/TransX   = 0. m
d:Ge/Film1/TransY   = 0. m
d:Ge/Film1/TransZ   = 1. m
d:Ge/Film1/RotX     = 0. deg
d:Ge/Film1/RotY     = 0. deg
d:Ge/Film1/RotZ     = 0. deg
i:Ge/Film1/XBins    = 2
i:Ge/Film1/YBins    = 2
i:Ge/Film1/ZBins    = 2

s:Ge/Film2/Type      = "TsBox"
s:Ge/Film2/Parent    = "Outer"
s:Ge/Film2/Material  = "G4_WATER"
d:Ge/Film2/HLX      = 3. m
d:Ge/Film2/HLY      = 3. m
d:Ge/Film2/HLZ      = .1 m
d:Ge/Film2/TransX   = 0. m
d:Ge/Film2/TransY   = 0. m
d:Ge/Film2/TransZ   = -1. m
d:Ge/Film2/RotX     = 0. deg
d:Ge/Film2/RotY     = 0. deg
d:Ge/Film2/RotZ     = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
s:Gr/MyOGL/ColorBy   = "OriginVolume"
sv:Gr/MyOGL/ColorByOriginVolumeNames = 6 "World" "Outer" "Film1" "Film2" "Film1/
→Film1_Z_Division" "Box"
sv:Gr/MyOGL/ColorByOriginVolumeColors = 6 "Red" "Blue" "Yellow" "Pink" "Magenta"
→"Green"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.8 ColorByParticleType.txt

```
# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m
d:Ge/Film1/HLZ     = .1 m
d:Ge/Film1/TransX  = 0. m
d:Ge/Film1/TransY  = 0. m
d:Ge/Film1/TransZ  = 1. m
d:Ge/Film1/RotX    = 0. deg
d:Ge/Film1/RotY    = 0. deg
d:Ge/Film1/RotZ    = 0. deg
i:Ge/Film1/XBins    = 2
i:Ge/Film1/YBins    = 2
i:Ge/Film1/ZBins    = 2

s:Ge/Film2/Type    = "TsBox"
s:Ge/Film2/Parent  = "Outer"
s:Ge/Film2/Material = "G4_WATER"
d:Ge/Film2/HLX     = 3. m
d:Ge/Film2/HLY     = 3. m
d:Ge/Film2/HLZ     = .1 m
d:Ge/Film2/TransX  = 0. m
d:Ge/Film2/TransY  = 0. m
d:Ge/Film2/TransZ  = -1. m
d:Ge/Film2/RotX    = 0. deg
```

(continues on next page)

(continued from previous page)

```

d:Ge/Film2/RotY      = 0. deg
d:Ge/Film2/RotZ      = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
s:Gr/MyOGL/ColorBy    = "ParticleType"
sv:Gr/MyOGL/ColorByParticleTypeNames = 4 "e-" "gamma" "proton" "neutron"
sv:Gr/MyOGL/ColorByParticleTypeColors = 4 "yellow" "pink" "aqua" "grass"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.9 ColorTest.txt

```

# Demonstrate the 16 standard colors from HTML 4.01 specification

s:Ge/Box01/Type      = "TsBox"
s:Ge/Box01/Parent    = "World"
s:Ge/Box01/Material   = "G4_WATER"
d:Ge/Box01/HLX        = 1. cm
d:Ge/Box01/HLY        = 0.5 cm
d:Ge/Box01/HLZ        = 1. cm
d:Ge/Box01/TransY     = 16. cm
s:Ge/Box01/DrawingStyle = "Solid"
s:Ge/Box01/Color      = "White"

s:Ge/Box02/Type      = "TsBox"
s:Ge/Box02/Parent    = "World"
s:Ge/Box02/Material   = "G4_WATER"
d:Ge/Box02/HLX        = 1. cm
d:Ge/Box02/HLY        = 0.5 cm
d:Ge/Box02/HLZ        = 1. cm
d:Ge/Box02/TransY     = 15. cm

```

(continues on next page)

(continued from previous page)

```

s:Ge/Box02/DrawingStyle = "Solid"
s:Ge/Box02/Color        = "Silver"

s:Ge/Box03/Type         = "TsBox"
s:Ge/Box03/Parent       = "World"
s:Ge/Box03/Material     = "G4_WATER"
d:Ge/Box03/HLX          = 1. cm
d:Ge/Box03/HLY          = 0.5 cm
d:Ge/Box03/HLZ          = 1. cm
d:Ge/Box03/TransY       = 14. cm
s:Ge/Box03/DrawingStyle = "Solid"
s:Ge/Box03/Color        = "Gray"

s:Ge/Box04/Type         = "TsBox"
s:Ge/Box04/Parent       = "World"
s:Ge/Box04/Material     = "G4_WATER"
d:Ge/Box04/HLX          = 1. cm
d:Ge/Box04/HLY          = 0.5 cm
d:Ge/Box04/HLZ          = 1. cm
d:Ge/Box04/TransY       = 13. cm
s:Ge/Box04/DrawingStyle = "Solid"
s:Ge/Box04/Color        = "Black"

s:Ge/Box05/Type         = "TsBox"
s:Ge/Box05/Parent       = "World"
s:Ge/Box05/Material     = "G4_WATER"
d:Ge/Box05/HLX          = 1. cm
d:Ge/Box05/HLY          = 0.5 cm
d:Ge/Box05/HLZ          = 1. cm
d:Ge/Box05/TransY       = 12. cm
s:Ge/Box05/DrawingStyle = "Solid"
s:Ge/Box05/Color        = "Red"

s:Ge/Box06/Type         = "TsBox"
s:Ge/Box06/Parent       = "World"
s:Ge/Box06/Material     = "G4_WATER"
d:Ge/Box06/HLX          = 1. cm
d:Ge/Box06/HLY          = 0.5 cm
d:Ge/Box06/HLZ          = 1. cm
d:Ge/Box06/TransY       = 11. cm
s:Ge/Box06/DrawingStyle = "Solid"
s:Ge/Box06/Color        = "Maroon"

s:Ge/Box07/Type         = "TsBox"
s:Ge/Box07/Parent       = "World"
s:Ge/Box07/Material     = "G4_WATER"
d:Ge/Box07/HLX          = 1. cm
d:Ge/Box07/HLY          = 0.5 cm
d:Ge/Box07/HLZ          = 1. cm
d:Ge/Box07/TransY       = 10. cm
s:Ge/Box07/DrawingStyle = "Solid"
s:Ge/Box07/Color        = "Yellow"

s:Ge/Box08/Type         = "TsBox"
s:Ge/Box08/Parent       = "World"
s:Ge/Box08/Material     = "G4_WATER"
d:Ge/Box08/HLX          = 1. cm

```

(continues on next page)

(continued from previous page)

```
d:Ge/Box08/HLY      = 0.5 cm
d:Ge/Box08/HLZ      = 1. cm
d:Ge/Box08/TransY   = 9. cm
s:Ge/Box08/DrawingStyle = "Solid"
s:Ge/Box08/Color     = "Olive"

s:Ge/Box09/Type      = "TsBox"
s:Ge/Box09/Parent    = "World"
s:Ge/Box09/Material  = "G4_WATER"
d:Ge/Box09/HLX       = 1. cm
d:Ge/Box09/HLY       = 0.5 cm
d:Ge/Box09/HLZ       = 1. cm
d:Ge/Box09/TransY    = 8. cm
s:Ge/Box09/DrawingStyle = "Solid"
s:Ge/Box09/Color     = "Lime"

s:Ge/Box10/Type      = "TsBox"
s:Ge/Box10/Parent    = "World"
s:Ge/Box10/Material  = "G4_WATER"
d:Ge/Box10/HLX       = 1. cm
d:Ge/Box10/HLY       = 0.5 cm
d:Ge/Box10/HLZ       = 1. cm
d:Ge/Box10/TransY    = 7. cm
s:Ge/Box10/DrawingStyle = "Solid"
s:Ge/Box10/Color     = "Green"

s:Ge/Box11/Type      = "TsBox"
s:Ge/Box11/Parent    = "World"
s:Ge/Box11/Material  = "G4_WATER"
d:Ge/Box11/HLX       = 1. cm
d:Ge/Box11/HLY       = 0.5 cm
d:Ge/Box11/HLZ       = 1. cm
d:Ge/Box11/TransY    = 6. cm
s:Ge/Box11/DrawingStyle = "Solid"
s:Ge/Box11/Color     = "Aqua"

s:Ge/Box12/Type      = "TsBox"
s:Ge/Box12/Parent    = "World"
s:Ge/Box12/Material  = "G4_WATER"
d:Ge/Box12/HLX       = 1. cm
d:Ge/Box12/HLY       = 0.5 cm
d:Ge/Box12/HLZ       = 1. cm
d:Ge/Box12/TransY    = 5. cm
s:Ge/Box12/DrawingStyle = "Solid"
s:Ge/Box12/Color     = "Teal"

s:Ge/Box13/Type      = "TsBox"
s:Ge/Box13/Parent    = "World"
s:Ge/Box13/Material  = "G4_WATER"
d:Ge/Box13/HLX       = 1. cm
d:Ge/Box13/HLY       = 0.5 cm
d:Ge/Box13/HLZ       = 1. cm
d:Ge/Box13/TransY    = 4. cm
s:Ge/Box13/DrawingStyle = "Solid"
s:Ge/Box13/Color     = "Blue"

s:Ge/Box14/Type      = "TsBox"
```

(continues on next page)



(continued from previous page)

```

s:Ge/Box14/Parent      = "World"
s:Ge/Box14/Material    = "G4_WATER"
d:Ge/Box14/HLX         = 1. cm
d:Ge/Box14/HLY         = 0.5 cm
d:Ge/Box14/HLZ         = 1. cm
d:Ge/Box14/TransY      = 3. cm
s:Ge/Box14/DrawingStyle = "Solid"
s:Ge/Box14/Color       = "Navy"

s:Ge/Box15/Type        = "TsBox"
s:Ge/Box15/Parent      = "World"
s:Ge/Box15/Material    = "G4_WATER"
d:Ge/Box15/HLX         = 1. cm
d:Ge/Box15/HLY         = 0.5 cm
d:Ge/Box15/HLZ         = 1. cm
d:Ge/Box15/TransY      = 2. cm
s:Ge/Box15/DrawingStyle = "Solid"
s:Ge/Box15/Color       = "Fuchsia"

s:Ge/Box16/Type        = "TsBox"
s:Ge/Box16/Parent      = "World"
s:Ge/Box16/Material    = "G4_WATER"
d:Ge/Box16/HLX         = 1. cm
d:Ge/Box16/HLY         = 0.5 cm
d:Ge/Box16/HLZ         = 1. cm
d:Ge/Box16/TransY      = 1. cm
s:Ge/Box16/DrawingStyle = "Solid"
s:Ge/Box16/Color       = "Purple"

b:Ge/World/Invisible   = "True"

sv:Ph/Default/Modules  = 1 "g4em-standard_opt0"

s:Gr/ViewA/Type        = "OpenGL"

b:Ts/PauseBeforeQuit   = "True"

```

## 20.10 FilterByCreatorProcess.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type          = "TsBox"
s:Ge/Box/Parent        = "World"
s:Ge/Box/Material      = "G4_WATER"
d:Ge/Box/HLX           = 4. m
d:Ge/Box/HLY           = 4. m
d:Ge/Box/HLZ           = .4 m
d:Ge/Box/TransX        = 0. m
d:Ge/Box/TransY        = 0. m
d:Ge/Box/TransZ        = -3. m
d:Ge/Box/RotX          = 0. deg
d:Ge/Box/RotY          = 0. deg
d:Ge/Box/RotZ          = 0. deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/Outer/Type      = "TsBox"
s:Ge/Outer/Parent    = "World"
s:Ge/Outer/Material  = "G4_AIR"
d:Ge/Outer/HLX       = 3.2 m
d:Ge/Outer/HLY       = 3.2 m
d:Ge/Outer/HLZ       = 2. m
d:Ge/Outer/TransX    = 0. m
d:Ge/Outer/TransY    = 0. m
d:Ge/Outer/TransZ    = 1. m
d:Ge/Outer/RotX      = 0. deg
d:Ge/Outer/RotY      = 0. deg
d:Ge/Outer/RotZ      = 0. deg

s:Ge/Film1/Type      = "TsBox"
s:Ge/Film1/Parent    = "Outer"
s:Ge/Film1/Material  = "G4_WATER"
d:Ge/Film1/HLX       = 2. m
d:Ge/Film1/HLY       = 2. m
d:Ge/Film1/HLZ       = .1 m
d:Ge/Film1/TransX    = 0. m
d:Ge/Film1/TransY    = 0. m
d:Ge/Film1/TransZ    = 1. m
d:Ge/Film1/RotX      = 0. deg
d:Ge/Film1/RotY      = 0. deg
d:Ge/Film1/RotZ      = 0. deg
i:Ge/Film1/XBins      = 2
i:Ge/Film1/YBins      = 2
i:Ge/Film1/ZBins      = 2

s:Ge/Film2/Type      = "TsBox"
s:Ge/Film2/Parent    = "Outer"
s:Ge/Film2/Material  = "G4_WATER"
d:Ge/Film2/HLX       = 3. m
d:Ge/Film2/HLY       = 3. m
d:Ge/Film2/HLZ       = .1 m
d:Ge/Film2/TransX    = 0. m
d:Ge/Film2/TransY    = 0. m
d:Ge/Film2/TransZ    = -1. m
d:Ge/Film2/RotX      = 0. deg
d:Ge/Film2/RotY      = 0. deg
d:Ge/Film2/RotZ      = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
sv:Gr/OnlyIncludeParticlesFromProcess = 2 "primary" "compt"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 300 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"

```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffX    = 10. cm
d:So/Example/BeamPositionCutoffY    = 10. cm
d:So/Example/BeamPositionSpreadX    = 0.65 cm
d:So/Example/BeamPositionSpreadY    = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX     = 90. deg
d:So/Example/BeamAngularCutoffY     = 90. deg
d:So/Example/BeamAngularSpreadX     = 0.0032 rad
d:So/Example/BeamAngularSpreadY     = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.11 FilterByEnergy.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type     = "TsBox"
s:Ge/Film1/Parent   = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX      = 2. m
d:Ge/Film1/HLY      = 2. m
d:Ge/Film1/HLZ      = .1 m
d:Ge/Film1/TransX   = 0. m
d:Ge/Film1/TransY   = 0. m
d:Ge/Film1/TransZ   = 1. m

```

(continues on next page)

(continued from previous page)

```

d:Ge/Film1/RotX      = 0. deg
d:Ge/Film1/RotY      = 0. deg
d:Ge/Film1/RotZ      = 0. deg
i:Ge/Film1/XBins     = 2
i:Ge/Film1/YBins     = 2
i:Ge/Film1/ZBins     = 2

s:Ge/Film2/Type       = "TsBox"
s:Ge/Film2/Parent     = "Outer"
s:Ge/Film2/Material   = "G4_WATER"
d:Ge/Film2/HLX        = 3. m
d:Ge/Film2/HLY        = 3. m
d:Ge/Film2/HLZ        = .1 m
d:Ge/Film2/TransX     = 0. m
d:Ge/Film2/TransY     = 0. m
d:Ge/Film2/TransZ     = -1. m
d:Ge/Film2/RotX       = 0. deg
d:Ge/Film2/RotY       = 0. deg
d:Ge/Film2/RotZ       = 0. deg

s:Gr/MyOGL/Type       = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX  = 0
i:Gr/MyOGL/WindowPosY  = 0
d:Gr/OnlyIncludeParticlesWithInitialKEAbove = 200. MeV

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 300 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.12 FilterByMomentum.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"

```

(continues on next page)

(continued from previous page)

```

s:Ge/Box/Parent      = "World"
s:Ge/Box/Material    = "G4_WATER"
d:Ge/Box/HLX         = 4. m
d:Ge/Box/HLY         = 4. m
d:Ge/Box/HLZ         = .4 m
d:Ge/Box/TransX      = 0. m
d:Ge/Box/TransY      = 0. m
d:Ge/Box/TransZ      = -3. m
d:Ge/Box/RotX        = 0. deg
d:Ge/Box/RotY        = 0. deg
d:Ge/Box/RotZ        = 0. deg

s:Ge/Outer/Type      = "TsBox"
s:Ge/Outer/Parent    = "World"
s:Ge/Outer/Material  = "G4_AIR"
d:Ge/Outer/HLX       = 3.2 m
d:Ge/Outer/HLY       = 3.2 m
d:Ge/Outer/HLZ       = 2. m
d:Ge/Outer/TransX    = 0. m
d:Ge/Outer/TransY    = 0. m
d:Ge/Outer/TransZ    = 1. m
d:Ge/Outer/RotX      = 0. deg
d:Ge/Outer/RotY      = 0. deg
d:Ge/Outer/RotZ      = 0. deg

s:Ge/Film1/Type      = "TsBox"
s:Ge/Film1/Parent    = "Outer"
s:Ge/Film1/Material  = "G4_WATER"
d:Ge/Film1/HLX       = 2. m
d:Ge/Film1/HLY       = 2. m
d:Ge/Film1/HLZ       = .1 m
d:Ge/Film1/TransX    = 0. m
d:Ge/Film1/TransY    = 0. m
d:Ge/Film1/TransZ    = 1. m
d:Ge/Film1/RotX      = 0. deg
d:Ge/Film1/RotY      = 0. deg
d:Ge/Film1/RotZ      = 0. deg
i:Ge/Film1/XBins     = 2
i:Ge/Film1/YBins     = 2
i:Ge/Film1/ZBins     = 2

s:Ge/Film2/Type      = "TsBox"
s:Ge/Film2/Parent    = "Outer"
s:Ge/Film2/Material  = "G4_WATER"
d:Ge/Film2/HLX       = 3. m
d:Ge/Film2/HLY       = 3. m
d:Ge/Film2/HLZ       = .1 m
d:Ge/Film2/TransX    = 0. m
d:Ge/Film2/TransY    = 0. m
d:Ge/Film2/TransZ    = -1. m
d:Ge/Film2/RotX      = 0. deg
d:Ge/Film2/RotY      = 0. deg
d:Ge/Film2/RotZ      = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600

```

(continues on next page)

(continued from previous page)

```

i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
d:Gr/OnlyIncludeParticlesWithInitialMomentumAbove = 200. MeV

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 300 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.13 FilterByOriginComponent.txt

```

# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type = "TsBox"
s:Ge/Box/Parent = "World"
s:Ge/Box/Material = "G4_WATER"
d:Ge/Box/HLX = 4. m
d:Ge/Box/HLY = 4. m
d:Ge/Box/HLZ = .4 m
d:Ge/Box/TransX = 0. m
d:Ge/Box/TransY = 0. m
d:Ge/Box/TransZ = -3. m
d:Ge/Box/RotX = 0. deg
d:Ge/Box/RotY = 0. deg
d:Ge/Box/RotZ = 0. deg

s:Ge/Outer/Type = "TsBox"
s:Ge/Outer/Parent = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX = 3.2 m
d:Ge/Outer/HLY = 3.2 m
d:Ge/Outer/HLZ = 2. m
d:Ge/Outer/TransX = 0. m
d:Ge/Outer/TransY = 0. m
d:Ge/Outer/TransZ = 1. m
d:Ge/Outer/RotX = 0. deg
d:Ge/Outer/RotY = 0. deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/Outer/RotZ      = 0. deg

s:Ge/Film1/Type      = "TsBox"
s:Ge/Film1/Parent    = "Outer"
s:Ge/Film1/Material  = "G4_WATER"
d:Ge/Film1/HLX       = 2. m
d:Ge/Film1/HLY       = 2. m
d:Ge/Film1/HLZ       = .1 m
d:Ge/Film1/TransX    = 0. m
d:Ge/Film1/TransY    = 0. m
d:Ge/Film1/TransZ    = 1. m
d:Ge/Film1/RotX      = 0. deg
d:Ge/Film1/RotY      = 0. deg
d:Ge/Film1/RotZ      = 0. deg
i:Ge/Film1/XBins     = 2
i:Ge/Film1/YBins     = 2
i:Ge/Film1/ZBins     = 2

s:Ge/Film2/Type      = "TsBox"
s:Ge/Film2/Parent    = "Outer"
s:Ge/Film2/Material  = "G4_WATER"
d:Ge/Film2/HLX       = 3. m
d:Ge/Film2/HLY       = 3. m
d:Ge/Film2/HLZ       = .1 m
d:Ge/Film2/TransX    = 0. m
d:Ge/Film2/TransY    = 0. m
d:Ge/Film2/TransZ    = -1. m
d:Ge/Film2/RotX      = 0. deg
d:Ge/Film2/RotY      = 0. deg
d:Ge/Film2/RotZ      = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX  = 0
i:Gr/MyOGL/WindowPosY  = 0
sv:Gr/OnlyIncludeParticlesFromComponent = 1 "Film2"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 300 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

(continues on next page)

(continued from previous page)

```
Ts/UseQt = "False"
```

## 20.14 FilterByParticleCharge.txt

```
# Demonstrates changing color scheme of trajectories.
# Trajectories are colored based on the process that created the trajectory.

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 4. m
d:Ge/Box/HLY       = 4. m
d:Ge/Box/HLZ       = .4 m
d:Ge/Box/TransX    = 0. m
d:Ge/Box/TransY    = 0. m
d:Ge/Box/TransZ    = -3. m
d:Ge/Box/RotX      = 0. deg
d:Ge/Box/RotY      = 0. deg
d:Ge/Box/RotZ      = 0. deg

s:Ge/Outer/Type    = "TsBox"
s:Ge/Outer/Parent  = "World"
s:Ge/Outer/Material = "G4_AIR"
d:Ge/Outer/HLX     = 3.2 m
d:Ge/Outer/HLY     = 3.2 m
d:Ge/Outer/HLZ     = 2. m
d:Ge/Outer/TransX  = 0. m
d:Ge/Outer/TransY  = 0. m
d:Ge/Outer/TransZ  = 1. m
d:Ge/Outer/RotX    = 0. deg
d:Ge/Outer/RotY    = 0. deg
d:Ge/Outer/RotZ    = 0. deg

s:Ge/Film1/Type    = "TsBox"
s:Ge/Film1/Parent  = "Outer"
s:Ge/Film1/Material = "G4_WATER"
d:Ge/Film1/HLX     = 2. m
d:Ge/Film1/HLY     = 2. m
d:Ge/Film1/HLZ     = .1 m
d:Ge/Film1/TransX  = 0. m
d:Ge/Film1/TransY  = 0. m
d:Ge/Film1/TransZ  = 1. m
d:Ge/Film1/RotX    = 0. deg
d:Ge/Film1/RotY    = 0. deg
d:Ge/Film1/RotZ    = 0. deg
i:Ge/Film1/XBins    = 2
i:Ge/Film1/YBins    = 2
i:Ge/Film1/ZBins    = 2

s:Ge/Film2/Type    = "TsBox"
s:Ge/Film2/Parent  = "Outer"
s:Ge/Film2/Material = "G4_WATER"
d:Ge/Film2/HLX     = 3. m
d:Ge/Film2/HLY     = 3. m
```

(continues on next page)



(continued from previous page)

```

d:Ge/Film2/HLZ      = .1 m
d:Ge/Film2/TransX   = 0. m
d:Ge/Film2/TransY   = 0. m
d:Ge/Film2/TransZ   = -1. m
d:Ge/Film2/RotX     = 0. deg
d:Ge/Film2/RotY     = 0. deg
d:Ge/Film2/RotZ     = 0. deg

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
sv:Gr/OnlyIncludeParticlesCharged = 2 "Negative" "Positive" # "Neutral"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 300 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
Ts/UseQt = "False"

```

## 20.15 NanoMeterScaleTest.txt

```

# This example demonstrates an issue that occurs when there
# is a very large zoom factor and how to work around that issue.
# You will see that the display is not exactly centered on the
# desired component, and that, if you try to adjust the Theta or Phi,
# the change is not smooth but instead has discontinuities.
# The issue is resolved if one makes the world invisible,
# as this means the initial extent of the displayed scene only
# contains the nm scale components, so little or no zoom is required.

d:Ge/World/HLX      = 20. cm
d:Ge/World/HLY      = 20. cm
d:Ge/World/HLZ      = 20. cm
#b:Ge/World/Invisible = "True" # Uncomment to solve zoom issue

s:Ge/CenterBox/Type      = "TsBox"
s:Ge/CenterBox/Material = "Air"

```

(continues on next page)

(continued from previous page)

```

s:Ge/CenterBox/Parent      = "World"
d:Ge/CenterBox/HLX         = 2.5 nm
d:Ge/CenterBox/HLY         = 2. nm
d:Ge/CenterBox/HLZ         = 1. nm
s:Ge/CenterBox/Color       = "White"

s:Ge/PlusXBox/Type         = "TsBox"
s:Ge/PlusXBox/Material     = "Air"
s:Ge/PlusXBox/Parent       = "World"
d:Ge/PlusXBox/HLX         = 2.5 nm
d:Ge/PlusXBox/HLY         = 2. nm
d:Ge/PlusXBox/HLZ         = 1. nm
d:Ge/PlusXBox/TransX      = 10. nm
s:Ge/PlusXBox/Color       = "Blue"

s:Ge/PlusYBox/Type         = "TsBox"
s:Ge/PlusYBox/Material     = "Air"
s:Ge/PlusYBox/Parent       = "World"
d:Ge/PlusYBox/HLX         = 2.5 nm
d:Ge/PlusYBox/HLY         = 2. nm
d:Ge/PlusYBox/HLZ         = 1. nm
d:Ge/PlusYBox/TransY      = 10. nm
s:Ge/PlusYBox/Color       = "Red"

sv:Ph/Default/Modules     = 1 "g4em-standard_opt0"

s:Gr/ViewA/Type            = "OpenGL"
uv:Gr/ViewA/Scale          = 3 1000 1000 1000 # Comment out to solve zoom issue
u:Gr/ViewA/Zoom            = 40000 # Comment out to solve zoom issue
d:Gr/ViewA/Theta           = 45. deg
d:Gr/ViewA/Phi             = 45. deg

s:Gr/ViewA/CenterOn       = "PlusYBox"

b:Ts/PauseBeforeQuit      = "True"

```

## 20.16 QtTest.txt

```

# Qt example.
# Parameters marked Changeable with the "c:" syntax appear in GUI

s:Ge/MyBox/Type            = "TsBox"
sc:Ge/MyBox/Material       = "G4_WATER"
sc:Ge/MyBox/Parent         = "World"
dc:Ge/MyBox/HLX            = 1.5 m
dc:Ge/MyBox/HLY            = 1. m
dc:Ge/MyBox/HLZ            = 1. m
dc:Ge/MyBox/TransX         = 0. m
dc:Ge/MyBox/TransY         = 0. m
dc:Ge/MyBox/TransZ         = 1.5 m
dc:Ge/MyBox/RotX           = 0. deg
dc:Ge/MyBox/RotY           = 0. deg
dc:Ge/MyBox/RotZ           = 0. deg
ic:Ge/MyBox/XBins          = 1

```

(continues on next page)

(continued from previous page)

```

ic:Ge/MyBox/YBins      = 1
ic:Ge/MyBox/ZBins      = 1
sc:Ge/MyBox/Color      = "blue"
sc:Ge/MyBox/DrawingStyle = "WireFrame"
bc:Ge/MyBox/Invisible  = "False"

s:So/Therapy/Type      = "Beam"
sc:So/Therapy/Component = "BeamPosition"
sc:So/Therapy/BeamParticle = "e-"
dc:So/Therapy/BeamEnergy      = 169.23 MeV
uc:So/Therapy/BeamEnergySpread = 0.757504
sc:So/Therapy/BeamPositionDistribution = "Gaussian"
sc:So/Therapy/BeamPositionCutoffShape = "Ellipse"
dc:So/Therapy/BeamPositionCutoffX = 10. cm
dc:So/Therapy/BeamPositionCutoffY = 10. cm
dc:So/Therapy/BeamPositionSpreadX = 0.65 cm
dc:So/Therapy/BeamPositionSpreadY = 0.65 cm
sc:So/Therapy/BeamAngularDistribution = "Gaussian"
dc:So/Therapy/BeamAngularCutoffX = 90. deg
dc:So/Therapy/BeamAngularCutoffY = 90. deg
dc:So/Therapy/BeamAngularSpreadX = 0.0032 rad
dc:So/Therapy/BeamAngularSpreadY = 0.0032 rad
ic:So/Therapy/NumberOfHistoriesInRun = 10

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:Gr/ViewA/Type      = "OpenGL"

Ts/UseQt = "True"

```

## 20.17 SolidSphere.txt

```

# Demonstrate effect of drawing style and hidden line removal

s:Ge/Sphere/Type      = "TsSphere"
s:Ge/Sphere/Parent     = "World"
s:Ge/Sphere/Material   = "G4_WATER"
d:Ge/Sphere/RMin       = 0. m
d:Ge/Sphere/RMax       = 2. m
d:Ge/Sphere/SPhi        = 0. deg
d:Ge/Sphere/DPhi        = 360. deg
d:Ge/Sphere/STheta      = 0. deg
d:Ge/Sphere/DTheta      = 180. deg
d:Ge/Sphere/TransX      = 0. m
d:Ge/Sphere/TransY      = 0. m
d:Ge/Sphere/TransZ      = 0. m
d:Ge/Sphere/RotX        = 0. deg
d:Ge/Sphere/RotY        = 0. deg
d:Ge/Sphere/RotZ        = 0. deg
s:Ge/Sphere/DrawingStyle = "Solid"

s:Gr/MyOpenGL/Type     = "OpenGL"
i:Gr/MyOpenGL/WindowSizeX = 600
i:Gr/MyOpenGL/WindowSizeY = 600

```

(continues on next page)

(continued from previous page)

```

i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
d:Gr/MyOGL/Theta       = 30. deg
d:Gr/MyOGL/Phi         = 30. deg
u:Gr/MyOGL/Zoom        = 3.
s:Gr/MyOGL/ColorBy     = "particletype"
b:Gr/MyOGL/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

b:Ts/PauseBeforeQuit = "True"

```

## 20.18 TestGraphics.txt

```

# Demonstrates a variety of Graphics options

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 1. m
d:Ge/Box/HLY       = 1. m
d:Ge/Box/HLZ       = .4 m
s:Ge/Box/DrawingStyle = "Solid"

s:Gr/MyOGL/Type      = "OpenGL"
b:Gr/MyOGL/CopyOpenGLToEPS = "True"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0
u:Gr/MyOGL/Zoom      = 3.
d:Gr/MyOGL/Theta     = 30. deg
d:Gr/MyOGL/Phi       = 30. deg
b:Gr/MyOGL/IncludeGeometry = "t"
b:Gr/MyOGL/IncludeTrajectories = "t"
b:Gr/MyOGL/HiddenLineRemovalForGeometry = "f"
b:Gr/MyOGL/HiddenLineRemovalForTrajectories = "f"

```

(continues on next page)

(continued from previous page)

```

# ColorBy options are "Charge", "ParticleType", "OriginComponent", "Energy",
# "Momentum", "Generation" and "CreatorProcess"
s:Gr/MyOGL/ColorBy = "particletype"
sv:Gr/MyOGL/ColorByChargeColors = 3 "red" "red" "red"
sv:Gr/MyOGL/ColorByParticleTypeNames = 2 "proton" "e-"
sv:Gr/MyOGL/ColorByParticleTypeColors = 2 "red" "green"
sv:Gr/MyOGL/ColorByOriginComponentNames = 2 "World" "Box"
sv:Gr/MyOGL/ColorByOriginComponentColors = 2 "yellow" "red"
dv:Gr/MyOGL/ColorByEnergyRanges = 4 .3 2. 100. 200. MeV
sv:Gr/MyOGL/ColorByEnergyColors = 5 "red" "yellow" "green" "blue" "white"
dv:Gr/MyOGL/ColorByMomentumRanges = 4 .3 2. 200. 646. MeV
sv:Gr/MyOGL/ColorByMomentumColors = 5 "red" "yellow" "green" "blue" "white"
sv:Gr/MyOGL/ColorByCreatorProcessNames = 5 "eBrem" "annihil" "Decay" "eIoni" "hIoni"
sv:Gr/MyOGL/ColorByCreatorProcessColors = 5 "red" "green" "blue" "yellow" "magenta"

s:Gr/MyHepRep/Type = "HepRep"
s:Gr/MyHepRep/FileName = "Above200MeV"

#s:Gr/MyVRML/Type = "VRML"

#s:Gr/MyRayTracer/Type = "RayTracer"

#s:Gr/MyRayTracerX/Type = "RayTracerX"
i:Gr/MyRayTracerX/WindowSizeX = 200
i:Gr/MyRayTracerX/WindowSizeY = 800
i:Gr/MyRayTracerX/WindowPosX = 100
i:Gr/MyRayTracerX/WindowPosY = 300

#s:Gr/MyDAWN/Type = "DAWN"

s:Gr/RefreshEvery = "Run"
#sv:Gr/OnlyIncludeParticlesNamed = 1 "proton"
#sv:Gr/OnlyIncludeParticlesCharged = 1 "negative neutral"
d:Gr/OnlyIncludeParticlesWithInitialKEAbove = 199. MeV
d:Gr/OnlyIncludeParticlesWithInitialKEBelow = 202. MeV
#sv:Gr/OnlyIncludeParticlesFromProcess = 1 "eBrem"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 200. MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 20.19 TwoProjections.txt

```
# Demonstrate two different kinds of graphical projections.

s:Gr/MyOGLa/Type           = "OpenGL"
i:Gr/MyOGLa/WindowSizeX    = 400
i:Gr/MyOGLa/WindowSizeY    = 400
i:Gr/MyOGLa/WindowPosX     = 0
i:Gr/MyOGLa/WindowPosY     = 0
s:Gr/MyOGLa/Projection     = "Orthogonal"
d:Gr/MyOGLa/Theta          = 45. deg
d:Gr/MyOGLa/Phi            = 45. deg

s:Gr/MyOGLb/Type           = "OpenGL"
i:Gr/MyOGLb/WindowSizeX    = 400
i:Gr/MyOGLb/WindowSizeY    = 400
i:Gr/MyOGLb/WindowPosX     = 0
i:Gr/MyOGLb/WindowPosY     = 400
s:Gr/MyOGLb/Projection     = "Perspective"
d:Gr/MyOGLb/PerspectiveAngle = 10. deg
d:Gr/MyOGLb/Theta          = 45. deg
d:Gr/MyOGLb/Phi            = 45. deg

s:Ge/Box/Type              = "TsBox"
s:Ge/Box/Parent            = "World"
s:Ge/Box/Material          = "G4_WATER"
d:Ge/Box/HLX               = 1. m
d:Ge/Box/HLY               = 1. m
d:Ge/Box/HLZ               = .4 m
d:Ge/Box/TransX            = 0. m
d:Ge/Box/TransY            = 0. m
d:Ge/Box/TransZ            = 0. m
d:Ge/Box/RotX              = 0. deg
d:Ge/Box/RotY              = 0. deg
d:Ge/Box/RotZ              = 0. deg

s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy     = 200. MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"
```

## 21.1 ConstantsForNozzles.txt

```
# These values are used in all of the nozzle examples.

#All coordination information comes from TsDefaultParameters.
#IEC 61217 compliance

#Fixed Reference system (If)      => World volume
#Gantry coordination system (Ig) => GantryCoordination
#Source coordination system (S)  => BeamPosition rotateX(180), Z position w.r.t.
    ↳GantryCoordination
#Beam source coordinate system rotate 180 and translate along z w.r.t g.
#-> Y is -Y. we have to flip phase space data.
#Isocenter coordination system (Io)

d:Ge/World/HLX = 1. m
d:Ge/World/HLY = 1. m
d:Ge/World/HLZ = 3. m

#S2I
#d:Ge/S2I = 3.0 m

#Gantry coordination system (Ig)
s:Ge/Gantry/Parent = "World"
s:Ge/Gantry/Type   = "Group"
d:Ge/Gantry/TransX = 0. m
d:Ge/Gantry/TransY = 0. m
d:Ge/Gantry/TransZ = 0. m
d:Ge/Gantry/RotX   = 0. deg
d:Ge/Gantry/RotY   = 0. deg
d:Ge/Gantry/RotZ   = 0. deg

# Default Beam position
```

(continues on next page)

(continued from previous page)

```

s:Ge/BeamPosition/Parent = "Gantry"
s:Ge/BeamPosition/Type   = "Group"
d:Ge/BeamPosition/TransX = 0. m
d:Ge/BeamPosition/TransY = 0. m
d:Ge/BeamPosition/TransZ = 3.0 m
d:Ge/BeamPosition/RotX   = 180. deg
d:Ge/BeamPosition/RotY   = 0. deg
d:Ge/BeamPosition/RotZ   = 0. deg

b:Ts/ShowCPUTime          = "true"
i:Ts/ShowHistoryCountAtInterval = 0
b:Ts/ShowHistoryCountOnSingleLine = "False"

#####
#Material
#NitrogenGas for Chamber
#HeliumGas   for Pipe
#####
sv:Ma/NiGas/Components = 1 "Nitrogen"
uv:Ma/NiGas/Fractions  = 1 1.0
d:Ma/NiGas/Density     = 0.001251 g/cm3
#0.001251 g/cm3 (gas, 0 C, 101.325 kPa)

sv:Ma/HeGas/Components = 1 "Helium"
uv:Ma/HeGas/Fractions  = 1 1.0
d:Ma/HeGas/Density     = 0.000166322 g/cm3

sv:Ma/Gold/Components  = 1 "Gold"
uv:Ma/Gold/Fractions   = 1 1.0
d:Ma/Gold/Density      = 19.3 g/cm3
d:Ma/Gold/MeanExcitationEnergy = 790.0 eV
s:Ma/Gold/DefaultColor = "orange"

sv:Ma/CompensatorLucite/Components = 3 "Hydrogen" "Carbon" "Oxygen"
uv:Ma/CompensatorLucite/Fractions  = 3 0.080538 0.599848 0.319614
d:Ma/CompensatorLucite/Density     = 1.181 g/cm3
d:Ma/CompensatorLucite/MeanExcitationEnergy = 74.0 eV
s:Ma/CompensatorLucite/DefaultColor = "grey"

#####
# Dummy beam
#####
s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy    = 160.0 MeV
u:So/Example/BeamEnergySpread = 0.0
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 10

#####
# Physics
#####
sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

```

(continues on next page)



(continued from previous page)

```
i:Tf/Verbosity = 1

#####
# Sequence control
#####
b:Ts/PauseBeforeSequence = "F"
```

## 21.2 RasterScanningPattern.txt

```
# Defines a raster scanning pattern used in some demonstrations.

includeFile = ScanningNozzle.txt

# Time Features for Scanning Magnets
d:Ge/BFieldX_max = 0.15 tesla
d:Ge/BFieldX_min = Ge/BFieldX_max tesla * -1.0
d:Ge/BFieldX_range = Ge/BFieldX_max tesla * 2.0

s:Tf/BField2nd/Function = "Step"
dv:Tf/BField2nd/Times = 2 125.0 250.0 ms
dv:Tf/BField2nd/Values = 2 Tf/BField2ndLeft/Value Tf/BField2ndRight/Value tesla

s:Tf/BField2ndLeft/Function = "Linear tesla"
d:Tf/BField2ndLeft/RepetitionInterval = 125.0 ms
d:Tf/BField2ndLeft/Rate = 0.0024 tesla/ms
d:Tf/BField2ndLeft/StartValue = Ge/BFieldX_min tesla

s:Tf/BField2ndRight/Function = "Linear tesla"
d:Tf/BField2ndRight/RepetitionInterval = 125.0 ms
d:Tf/BField2ndRight/Rate = -0.0024 tesla/ms
d:Tf/BField2ndRight/StartValue = Ge/BFieldX_max tesla

s:Tf/BField1st/Function = "Step"
dv:Tf/BField1st/Times = 4 0.25 0.5 0.75 1.0 s
dv:Tf/BField1st/Values = 4 -0.105 -0.035 0.035 0.105 tesla

d:Ge/Dipole1/MagneticFieldStrength = Tf/BField1st/Value tesla
d:Ge/Dipole2/MagneticFieldStrength = Tf/BField2nd/Value tesla
```

## 21.3 ScanningNozzle.txt

```
# Scanning nozzle used in some demonstrations.

includeFile = ConstantsForNozzles.txt

#Ts/DumpParameters = "T"

#####
#1. Beam Exit Window
#100 um thickness Mylar window
```

(continues on next page)

(continued from previous page)

```
#####
s:Ge/ExitWindow/Parent      = "Gantry"
s:Ge/ExitWindow/Type       = "TsCylinder"
s:Ge/ExitWindow/Material   = "Mylar"
d:Ge/ExitWindow/RMin       = 0.0 mm
d:Ge/ExitWindow/RMax       = 50.0 mm
d:Ge/ExitWindow/HL         = 0.05 mm
d:Ge/ExitWindow/SPhi       = 0.0 deg
d:Ge/ExitWindow/DPhi       = 360.0 deg
d:Ge/ExitWindow/TransX     = 0.0 mm
d:Ge/ExitWindow/TransY     = 0.0 mm
d:Ge/ExitWindow/TransZ     = 299.8 cm
d:Ge/ExitWindow/RotX       = 0.0 deg
d:Ge/ExitWindow/RotY       = 0.0 deg
d:Ge/ExitWindow/RotZ       = 0.0 deg
b:Ge/ExitWindow/Include    = "TRUE"
s:Ge/ExitWindow/DrawingStyle = "Solid"

d:Ge/DistanceOf_CenterExitWindow_To_TopMon1 = -30.0 mm
d:Ge/ZPlusOf_Mon1 = Ge/ExitWindow/TransZ + Ge/DistanceOf_CenterExitWindow_To_TopMon1_
→cm

#####
#2. Monitor chamber 1.
# Filled with Nitrogen Gas, 3 layers
# Thickness, position, Material
#      1.0      Gold
#      0.0      Aluminum
#      -1.0     Gold
#####
s:Ge/MonitorChamber1/Parent = "Gantry"
s:Ge/MonitorChamber1/Type   = "TsCylinder"
s:Ge/MonitorChamber1/Material = "NiGas"
d:Ge/MonitorChamber1/RMin   = 0.0 mm
d:Ge/MonitorChamber1/RMax   = 100.0 mm
d:Ge/MonitorChamber1/HL     = 15.0 mm
d:Ge/MonitorChamber1/SPhi   = 0.0 deg
d:Ge/MonitorChamber1/DPhi   = 360.0 deg
d:Ge/MonitorChamber1/RotX   = 0.0 deg
d:Ge/MonitorChamber1/RotY   = 0.0 deg
d:Ge/MonitorChamber1/RotZ   = 0.0 deg
d:Ge/MonitorChamber1/TransX = 0.0 mm
d:Ge/MonitorChamber1/TransY = 0.0 mm
d:Ge/MonitorChamber1/MinusHL = Ge/MonitorChamber1/HL mm * -1.0
d:Ge/MonitorChamber1/TransZ = Ge/ZPlusOf_Mon1 + Ge/MonitorChamber1/MinusHL mm
b:Ge/MonitorChamber1/Include = "TRUE"

s:Ge/MonitorChamber1/Layer1/Parent = "MonitorChamber1"
s:Ge/MonitorChamber1/Layer1/Type   = "TsCylinder"
s:Ge/MonitorChamber1/Layer1/Material = "Gold"
d:Ge/MonitorChamber1/Layer1/RMin   = 0.0 mm
d:Ge/MonitorChamber1/Layer1/RMax   = 90.0 mm
d:Ge/MonitorChamber1/Layer1/HL     = 0.002 mm
d:Ge/MonitorChamber1/Layer1/SPhi   = 0.0 deg
d:Ge/MonitorChamber1/Layer1/DPhi   = 360.0 deg
d:Ge/MonitorChamber1/Layer1/RotX   = 0.0 deg
d:Ge/MonitorChamber1/Layer1/RotY   = 0.0 deg
```

(continues on next page)

(continued from previous page)

```

d:Ge/MonitorChamber1/Layer1/RotZ      = 0.0 deg
d:Ge/MonitorChamber1/Layer1/TransX    = 0.0 mm
d:Ge/MonitorChamber1/Layer1/TransY    = 0.0 mm
d:Ge/MonitorChamber1/Layer1/TransZ    = 10.0 mm
b:Ge/MonitorChamber1/Layer1/Include   = "TRUE"

s:Ge/MonitorChamber1/Layer2/Parent    = "MonitorChamber1"
s:Ge/MonitorChamber1/Layer2/Type      = "TsCylinder"
s:Ge/MonitorChamber1/Layer2/Material  = "Aluminum"
d:Ge/MonitorChamber1/Layer2/RMin      = 0.0 mm
d:Ge/MonitorChamber1/Layer2/RMax      = 90.0 mm
d:Ge/MonitorChamber1/Layer2/HL        = 0.003 mm
d:Ge/MonitorChamber1/Layer2/SPhi      = 0.0 deg
d:Ge/MonitorChamber1/Layer2/DPhi      = 360.0 deg
d:Ge/MonitorChamber1/Layer2/RotX      = 0.0 deg
d:Ge/MonitorChamber1/Layer2/RotY      = 0.0 deg
d:Ge/MonitorChamber1/Layer2/RotZ      = 0.0 deg
d:Ge/MonitorChamber1/Layer2/TransX    = 0.0 mm
d:Ge/MonitorChamber1/Layer2/TransY    = 0.0 mm
d:Ge/MonitorChamber1/Layer2/TransZ    = 0.0 mm
b:Ge/MonitorChamber1/Layer2/Include   = "TRUE"

s:Ge/MonitorChamber1/Layer3/Parent    = "MonitorChamber1"
s:Ge/MonitorChamber1/Layer3/Type      = "TsCylinder"
s:Ge/MonitorChamber1/Layer3/Material  = "Gold"
d:Ge/MonitorChamber1/Layer3/RMin      = 0.0 mm
d:Ge/MonitorChamber1/Layer3/RMax      = 90.0 mm
d:Ge/MonitorChamber1/Layer3/HL        = 0.003 mm
d:Ge/MonitorChamber1/Layer3/SPhi      = 0.0 deg
d:Ge/MonitorChamber1/Layer3/DPhi      = 360.0 deg
d:Ge/MonitorChamber1/Layer3/RotX      = 0.0 deg
d:Ge/MonitorChamber1/Layer3/RotY      = 0.0 deg
d:Ge/MonitorChamber1/Layer3/RotZ      = 0.0 deg
d:Ge/MonitorChamber1/Layer3/TransX    = 0.0 mm
d:Ge/MonitorChamber1/Layer3/TransY    = 0.0 mm
d:Ge/MonitorChamber1/Layer3/TransZ    = -10.0 mm
b:Ge/MonitorChamber1/Layer3/Include   = "TRUE"

#Distance calculation for Pipe
d:Ge/DistanceOf_BottomMon1_To_TopPipe = -20.0 mm
d:Ge/ZMinusOf_Mon1    = Ge/MonitorChamber1/TransZ + Ge/MonitorChamber1/MinusHL mm
d:Ge/ZPlusOf_Pipe     = Ge/ZMinusOf_Mon1 + Ge/DistanceOf_BottomMon1_To_TopPipe mm

#####
# Quad/Dipole Field specification in length
#####
#Gap      Quad1      Gap      Quad2      Gap
#2.0 cm + 5.0*2 + 4.0 + 5.0*2 + 2.0 = 28.0 cm
d:Ge/HLOf_PipeInQuads = 14.0 cm
d:Ge/HLOf_Quad1       = 5.0 cm
d:Ge/HLOf_Quad2       = Ge/HLOf_Quad1 cm
d:Ge/Gap_Top_Quad1    = 2.0 cm
d:Ge/Gap_Quad1_Quad2  = 4.0 cm
d:Ge/Gap_Quad2_Bottom = 2.0 cm

d:Ge/HLxyOf_Quads      = 47.0 mm
d:Ge/HLxyOf_Dipoles    = 77.0 mm

```

(continues on next page)

(continued from previous page)

```

d:Ge/ThicknessOf_Pipe      = 3.0 mm
s:Ge/MaterialOf_PipeWall   = "Aluminum"
s:Ge/MaterialOf_PipeGas    = "HeGas"
#s:Ge/MaterialOf_PipeGas   = "Vacuum"

d:Ge/FLOf_PipeConnector    = 10.0 cm

d:Ge/HLOf_Dipole1          = 10.5 cm
d:Ge/HLOf_Dipole2          = 13.0 cm
d:Ge/DistanceOf_Coil1_Dipole1 = 6.0 cm
d:Ge/DistanceOf_Coil2_Dipole2 = 8.0 cm
d:Ge/DistanceOf_BottomDipoleCoil1_To_TopDipoleCoil2 = 1.0 cm
d:Ge/DistanceOf_BottomPipe_To_TopBPM1 = -1.0 mm
d:Ge/DistanceOf_BottomBPM1_To_PipeExtension = -0.0 mm
d:Ge/DistanceOf_BottomPipeExtension_To_BPM2 = -0.0 mm

#####
#Calculated parameters from input
#Don't modify following calculations

d:Ge/HLOf_DipoleCoil1      = Ge/HLOf_Dipole1 + Ge/DistanceOf_Coil1_Dipole1 cm
d:Ge/HLOf_DipoleCoil2      = Ge/HLOf_Dipole2 + Ge/DistanceOf_Coil2_Dipole2 cm
d:Ge/MinusHLOf_DipoleCoil1 = Ge/HLOf_DipoleCoil1 cm * -1.0
d:Ge/MinusHLOf_DipoleCoil2 = Ge/HLOf_DipoleCoil2 cm * -1.0

d:Ge/FLOf_DipoleCoil1      = Ge/HLOf_DipoleCoil1 cm * 2.0
d:Ge/FLOf_DipoleCoil2      = Ge/HLOf_DipoleCoil2 cm * 2.0
d:Ge/FLOf_DipoleCoil1_DipoleCoil2 = Ge/FLOf_DipoleCoil1 + Ge/FLOf_DipoleCoil2 cm

d:Ge/FLOf_PipeInQuads      = 2.0 * Ge/HLOf_PipeInQuads cm

d:Ge/HLOf_PipeConnector    = Ge/FLOf_PipeConnector cm * 0.5
d:Ge/MinusHLOf_PipeConnector = Ge/FLOf_PipeConnector cm * -0.5

d:Ge/FLOf_PipeInDipoles    = Ge/FLOf_DipoleCoil1_DipoleCoil2 + Ge/DistanceOf_
↪BottomDipoleCoil1_To_TopDipoleCoil2 cm
d:Ge/HLOf_PipeInDipoles    = Ge/FLOf_PipeInDipoles cm * 0.5
d:Ge/MinusHLOf_PipeInDipoles = Ge/FLOf_PipeInDipoles cm * -0.5

d:Ge/FLOf_QuadsDipoles     = Ge/FLOf_PipeInDipoles + Ge/FLOf_PipeInQuads cm

d:Ge/FLOf_Pipe             = Ge/FLOf_QuadsDipoles + Ge/FLOf_PipeConnector cm
d:Ge/HLOf_Pipe             = Ge/FLOf_Pipe cm * 0.5
#####

#####
# Pipe through Quads/Dipoles/
#####
s:Ge/BeamPipeInQuadsDipoles/Parent = "Gantry"
s:Ge/BeamPipeInQuadsDipoles/Type   = "Group"
d:Ge/BeamPipeInQuadsDipoles/TransX = 0. m
d:Ge/BeamPipeInQuadsDipoles/TransY = 0. m
d:Ge/BeamPipeInQuadsDipoles/TransZ = Ge/ZPlusOf_Pipe - Ge/HLOf_Pipe mm
d:Ge/BeamPipeInQuadsDipoles/RotX   = 0. deg
d:Ge/BeamPipeInQuadsDipoles/RotY   = 0. deg
d:Ge/BeamPipeInQuadsDipoles/RotZ   = 0. deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/PipeInQuads/Parent      = "BeamPipeInQuadsDipoles"
s:Ge/PipeInQuads/Type       = "TsBox"
s:Ge/PipeInQuads/Material   = Ge/MaterialOf_PipeWall
d:Ge/PipeInQuads/HLX        = Ge/HLxyOf_Quads + Ge/ThicknessOf_Pipe mm
d:Ge/PipeInQuads/HLY        = Ge/HLxyOf_Quads + Ge/ThicknessOf_Pipe mm
d:Ge/PipeInQuads/HLZ        = Ge/HLOf_PipeInQuads cm
d:Ge/PipeInQuads/TransX     = 0.0 mm
d:Ge/PipeInQuads/TransY     = 0.0 mm
d:Ge/PipeInQuads/TransZ     = Ge/HLOf_Pipe - Ge/HLOf_PipeInQuads cm
d:Ge/PipeInQuads/RotX       = 0.0 deg
d:Ge/PipeInQuads/RotY       = 0.0 deg
d:Ge/PipeInQuads/RotZ       = 0.0 deg

s:Ge/PipeInQuads/Gas/Parent = "PipeInQuads"
s:Ge/PipeInQuads/Gas/Type   = "TsBox"
s:Ge/PipeInQuads/Gas/Material = Ge/MaterialOf_PipeGas
d:Ge/PipeInQuads/Gas/HLX    = Ge/HLxyOf_Quads mm
d:Ge/PipeInQuads/Gas/HLY    = Ge/HLxyOf_Quads mm
d:Ge/PipeInQuads/Gas/HLZ    = Ge/PipeInQuads/HLZ cm
d:Ge/PipeInQuads/Gas/TransX = 0.0 mm
d:Ge/PipeInQuads/Gas/TransY = 0.0 mm
d:Ge/PipeInQuads/Gas/TransZ = 0.0 mm
d:Ge/PipeInQuads/Gas/RotX   = 0.0 deg
d:Ge/PipeInQuads/Gas/RotY   = 0.0 deg
d:Ge/PipeInQuads/Gas/RotZ   = 0.0 deg

#####
#3. Quadrupole 1 and Quadrupole 2 in pipe
#####
s:Ge/Quad1/Type      = "TsBox"
s:Ge/Quad1/Parent    = "PipeInQuads/Gas"
s:Ge/Quad1/Material  = "parent"
d:Ge/Quad1/HLX       = Ge/PipeInQuads/Gas/HLX cm
d:Ge/Quad1/HLY       = Ge/PipeInQuads/Gas/HLX cm
d:Ge/Quad1/HLZ       = Ge/HLOf_Quad1 cm
d:Ge/Quad1/TransX    = 0.0 cm
d:Ge/Quad1/TransY    = 0.0 cm
d:Ge/Quad1/Upstream  = Ge/PipeInQuads/Gas/HLZ - Ge/Gap_Top_Quad1 cm
d:Ge/Quad1/TransZ    = Ge/Quad1/Upstream - Ge/Quad1/HLZ cm
d:Ge/Quad1/Downstream = Ge/Quad1/TransZ - Ge/Quad1/HLZ cm
d:Ge/Quad1/RotX      = 0.0 deg
d:Ge/Quad1/RotY      = 0.0 deg
d:Ge/Quad1/RotZ      = 0.0 deg
s:Ge/Quad1/Field     = "QuadrupoleMagnet"
u:Ge/Quad1/MagneticFieldDirectionX = -1
u:Ge/Quad1/MagneticFieldDirectionY = 2.0
u:Ge/Quad1/MagneticFieldDirectionZ = 3.0
d:Ge/Quad1/MagneticFieldGradientX  = 0.0 tesla/cm
d:Ge/Quad1/MagneticFieldGradientY  = 0.0 tesla/cm

s:Ge/Quad2/Type      = "TsBox"
s:Ge/Quad2/Parent    = "PipeInQuads/Gas"
s:Ge/Quad2/Material  = "parent"
d:Ge/Quad2/HLX       = Ge/PipeInQuads/Gas/HLX cm
d:Ge/Quad2/HLY       = Ge/PipeInQuads/Gas/HLX cm
d:Ge/Quad2/HLZ       = Ge/HLOf_Quad2 cm
d:Ge/Quad2/TransX    = 0.0 cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/Quad2/TransY      = 0.0 cm
d:Ge/Quad2/Upstream    = Ge/Quad1/Downstream - Ge/Gap_Quad1_Quad2 cm
d:Ge/Quad2/TransZ      = Ge/Quad2/Upstream - Ge/Quad2/HLZ cm
d:Ge/Quad2/RotX        = 0.0 deg
d:Ge/Quad2/RotY        = 0.0 deg
d:Ge/Quad2/RotZ        = 0.0 deg
s:Ge/Quad2/Field = "QuadrupoleMagnet"
u:Ge/Quad2/MagneticFieldDirectionX = -1
u:Ge/Quad2/MagneticFieldDirectionY = 2.0
u:Ge/Quad2/MagneticFieldDirectionZ = 3.0
d:Ge/Quad2/MagneticFieldGradientX = 0.0 tesla/cm
d:Ge/Quad2/MagneticFieldGradientY = 0.0 tesla/cm

d:Ge/ZPlusPipeConnector = Ge/PipeInQuads/TransZ - Ge/HLOf_PipeInQuads cm

s:Ge/PipeConnector/Parent = "BeamPipeInQuadsDipoles"
s:Ge/PipeConnector/Type = "G4Trd"
s:Ge/PipeConnector/Material = Ge/MaterialOf_PipeWall
d:Ge/PipeConnector/HLX2 = Ge/HLxyOf_Quads + Ge/ThicknessOf_Pipe mm
d:Ge/PipeConnector/HLY2 = Ge/HLxyOf_Quads + Ge/ThicknessOf_Pipe mm
d:Ge/PipeConnector/HLX1 = Ge/HLxyOf_Dipoles + Ge/ThicknessOf_Pipe mm
d:Ge/PipeConnector/HLY1 = Ge/HLxyOf_Dipoles + Ge/ThicknessOf_Pipe mm
d:Ge/PipeConnector/HLZ = Ge/HLOf_PipeConnector mm
d:Ge/PipeConnector/TransX = 0.0 mm
d:Ge/PipeConnector/TransY = 0.0 mm
d:Ge/PipeConnector/TransZ = Ge/ZPlusPipeConnector + Ge/MinusHLOf_PipeConnector mm
d:Ge/PipeConnector/RotX = 0.0 deg
d:Ge/PipeConnector/RotY = 0.0 deg
d:Ge/PipeConnector/RotZ = 0.0 deg
b:Ge/PipeConnector/Include = "TRUE"

s:Ge/PipeConnector/Gas/Parent = "PipeConnector"
s:Ge/PipeConnector/Gas/Type = "G4Trd"
s:Ge/PipeConnector/Gas/Material = Ge/MaterialOf_PipeGas
d:Ge/PipeConnector/Gas/HLX2 = Ge/HLxyOf_Quads mm
d:Ge/PipeConnector/Gas/HLY2 = Ge/HLxyOf_Quads mm
d:Ge/PipeConnector/Gas/HLX1 = Ge/HLxyOf_Dipoles mm
d:Ge/PipeConnector/Gas/HLY1 = Ge/HLxyOf_Dipoles mm
d:Ge/PipeConnector/Gas/HLZ = Ge/PipeConnector/HLZ mm
d:Ge/PipeConnector/Gas/TransX = 0.0 mm
d:Ge/PipeConnector/Gas/TransY = 0.0 mm
d:Ge/PipeConnector/Gas/TransZ = 0.0 mm
d:Ge/PipeConnector/Gas/RotX = 0.0 deg
d:Ge/PipeConnector/Gas/RotY = 0.0 deg
d:Ge/PipeConnector/Gas/RotZ = 0.0 deg
b:Ge/PipeConnector/Gas/Include = "TRUE"

d:Ge/ZPlusPipeInDipoles = Ge/PipeConnector/TransZ + Ge/MinusHLOf_PipeConnector cm

s:Ge/PipeInDipoles/Parent = "BeamPipeInQuadsDipoles"
s:Ge/PipeInDipoles/Type = "TsBox"
s:Ge/PipeInDipoles/Material = Ge/MaterialOf_PipeWall
d:Ge/PipeInDipoles/HLX = Ge/HLxyOf_Dipoles + Ge/ThicknessOf_Pipe mm
d:Ge/PipeInDipoles/HLY = Ge/HLxyOf_Dipoles + Ge/ThicknessOf_Pipe mm
d:Ge/PipeInDipoles/HLZ = Ge/HLOf_PipeInDipoles cm
d:Ge/PipeInDipoles/TransX = 0.0 mm
d:Ge/PipeInDipoles/TransY = 0.0 mm

```

(continues on next page)

(continued from previous page)

```

d:Ge/PipeInDipoles/TransZ = Ge/ZPlusPipeInDipoles + Ge/MinusHLOf_PipeInDipoles cm
d:Ge/PipeInDipoles/RotX   = 0.0 deg
d:Ge/PipeInDipoles/RotY   = 0.0 deg
d:Ge/PipeInDipoles/RotZ   = 0.0 deg
b:Ge/PipeInDipoles/Include = "TRUE"

s:Ge/PipeInDipoles/Gas/Parent = "PipeInDipoles"
s:Ge/PipeInDipoles/Gas/Type   = "TsBox"
s:Ge/PipeInDipoles/Gas/Material = Ge/MaterialOf_PipeGas
d:Ge/PipeInDipoles/Gas/HLX    = Ge/HLxyOf_Dipoles mm
d:Ge/PipeInDipoles/Gas/HLY    = Ge/HLxyOf_Dipoles mm
d:Ge/PipeInDipoles/Gas/HLZ    = Ge/HLOf_PipeInDipoles cm
d:Ge/PipeInDipoles/Gas/TransX = 0.0 mm
d:Ge/PipeInDipoles/Gas/TransY = 0.0 mm
d:Ge/PipeInDipoles/Gas/TransZ = 0.0 mm
d:Ge/PipeInDipoles/Gas/RotX   = 0.0 deg
d:Ge/PipeInDipoles/Gas/RotY   = 0.0 deg
d:Ge/PipeInDipoles/Gas/RotZ   = 0.0 deg
b:Ge/PipeInDipoles/Gas/Include = "TRUE"

#####
#5. Dipole 1 and 2. in Pipe
#####
s:Ge/Dipole1/Type = "TsBox"
s:Ge/Dipole1/Parent = "PipeInDipoles/Gas"
s:Ge/Dipole1/Material = "parent"
d:Ge/Dipole1/HLX      = Ge/PipeInDipoles/Gas/HLX mm
d:Ge/Dipole1/HLY      = Ge/PipeInDipoles/Gas/HLY mm
d:Ge/Dipole1/HLZ      = Ge/HLOf_Dipole1 mm
d:Ge/Dipole1/TransX    = 0.0 cm
d:Ge/Dipole1/TransY    = 0.0 cm
d:Ge/Dipole1/TransZ    = Ge/HLOf_PipeInDipoles + Ge/MinusHLOf_DipoleCoil1 mm
d:Ge/Dipole1/RotX      = 0.0 deg
d:Ge/Dipole1/RotY      = 0.0 deg
d:Ge/Dipole1/RotZ      = 0.0 deg
s:Ge/Dipole1/Field = "DipoleMagnet"
u:Ge/Dipole1/MagneticFieldDirectionX = 1.0
u:Ge/Dipole1/MagneticFieldDirectionY = 0.0
u:Ge/Dipole1/MagneticFieldDirectionZ = 0.0
d:Ge/Dipole1/MagneticFieldStrength = 0.0 tesla

d:Ge/ZMinusOf_Dipole1 = Ge/Dipole1/TransZ + Ge/MinusHLOf_DipoleCoil1 mm
d:Ge/ZPlusOf_Dipole2  = Ge/ZMinusOf_Dipole1 + Ge/DistanceOf_BottomDipoleCoil1_To_
→TopDipoleCoil2 cm

s:Ge/Dipole2/Type = "TsBox"
s:Ge/Dipole2/Parent = "PipeInDipoles/Gas"
s:Ge/Dipole2/Material = "parent"
d:Ge/Dipole2/HLX      = Ge/PipeInDipoles/Gas/HLX mm
d:Ge/Dipole2/HLY      = Ge/PipeInDipoles/Gas/HLY mm
d:Ge/Dipole2/HLZ      = Ge/HLOf_Dipole2 mm
d:Ge/Dipole2/TransX    = 0.0 cm
d:Ge/Dipole2/TransY    = 0.0 cm
d:Ge/Dipole2/TransZ    = Ge/ZPlusOf_Dipole2 + Ge/MinusHLOf_DipoleCoil2 mm
d:Ge/Dipole2/RotX      = 0.0 deg
d:Ge/Dipole2/RotY      = 0.0 deg
d:Ge/Dipole2/RotZ      = 0.0 deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/Dipole2/Field = "DipoleMagnet"
u:Ge/Dipole2/MagneticFieldDirectionX = 0.0
u:Ge/Dipole2/MagneticFieldDirectionY = 1.0
u:Ge/Dipole2/MagneticFieldDirectionZ = 0.0
d:Ge/Dipole2/MagneticFieldStrength = 0.0 tesla

d:Ge/ZMinusOf_Pipe = Ge/BeamPipeInQuadsDipoles/TransZ - Ge/HLOf_Pipe mm
d:Ge/ZPlusOf_BPM1 = Ge/ZMinusOf_Pipe + Ge/DistanceOf_BottomPipe_To_TopBPM1 mm

#####
#5. Beam profile Monitor chamber at the end of pipe
#HLx, HLy, HLz (9.2, 9.2, 1.8)
# 2 Mylar windows
# 2 sets of Multi-wire chamber
#####
s:Ge/BeamProfileMonitor1/Parent = "Gantry"
s:Ge/BeamProfileMonitor1/Type = "TsBox"
s:Ge/BeamProfileMonitor1/Material = "NiGas"
d:Ge/BeamProfileMonitor1/HLX = 92.0 mm
d:Ge/BeamProfileMonitor1/HLY = 92.0 mm
d:Ge/BeamProfileMonitor1/HLZ = 22.0 mm
d:Ge/BeamProfileMonitor1/MinusHL = Ge/BeamProfileMonitor1/HLZ mm * -1.0
d:Ge/BeamProfileMonitor1/TransX = 0.0 mm
d:Ge/BeamProfileMonitor1/TransY = 0.0 mm
d:Ge/BeamProfileMonitor1/TransZ = Ge/ZPlusOf_BPM1 + Ge/BeamProfileMonitor1/MinusHL_
→cm
d:Ge/BeamProfileMonitor1/RotX = 0.0 deg
d:Ge/BeamProfileMonitor1/RotY = 0.0 deg
d:Ge/BeamProfileMonitor1/RotZ = 0.0 deg
b:Ge/BeamProfileMonitor1/Include = "TRUE"

s:Ge/BeamProfileMonitor1/Window1/Parent = "BeamProfileMonitor1"
s:Ge/BeamProfileMonitor1/Window1/Type = "TsBox"
s:Ge/BeamProfileMonitor1/Window1/Material = "Mylar"
d:Ge/BeamProfileMonitor1/Window1/HLX = 90.0 mm
d:Ge/BeamProfileMonitor1/Window1/HLY = 90.0 mm
d:Ge/BeamProfileMonitor1/Window1/HLZ = 0.02 mm
d:Ge/BeamProfileMonitor1/Window1/TransX = 0.0 mm
d:Ge/BeamProfileMonitor1/Window1/TransY = 0.0 mm
d:Ge/BeamProfileMonitor1/Window1/TransZ = 18.123 mm
d:Ge/BeamProfileMonitor1/Window1/RotX = 0.0 deg
d:Ge/BeamProfileMonitor1/Window1/RotY = 0.0 deg
d:Ge/BeamProfileMonitor1/Window1/RotZ = 0.0 deg

s:Ge/BeamProfileMonitor1/Window2/Parent = "BeamProfileMonitor1"
s:Ge/BeamProfileMonitor1/Window2/Type = "TsBox"
s:Ge/BeamProfileMonitor1/Window2/Material = "Mylar"
d:Ge/BeamProfileMonitor1/Window2/HLX = 90.0 mm
d:Ge/BeamProfileMonitor1/Window2/HLY = 90.0 mm
d:Ge/BeamProfileMonitor1/Window2/HLZ = 0.02 mm
d:Ge/BeamProfileMonitor1/Window2/TransX = 0.0 mm
d:Ge/BeamProfileMonitor1/Window2/TransY = 0.0 mm
d:Ge/BeamProfileMonitor1/Window2/TransZ = -18.123 mm
d:Ge/BeamProfileMonitor1/Window2/RotX = 0.0 deg
d:Ge/BeamProfileMonitor1/Window2/RotY = 0.0 deg
d:Ge/BeamProfileMonitor1/Window2/RotZ = 0.0 deg

```

(continues on next page)



(continued from previous page)

```

s:Ge/MWC_In_BPM1/Parent      = "BeamProfileMonitor1"
s:Ge/MWC_In_BPM1/Type        = "TsMultiWireChamber"
s:Ge/MWC_In_BPM1/Material    = "parent"
d:Ge/MWC_In_BPM1/HLX         = 88.0 mm
d:Ge/MWC_In_BPM1/HLY         = 88.0 mm
d:Ge/MWC_In_BPM1/HLZ         = 13.0 mm
d:Ge/MWC_In_BPM1/TransX      = 0.0 mm
d:Ge/MWC_In_BPM1/TransY      = 0.0 mm
d:Ge/MWC_In_BPM1/TransZ      = 0.0 mm
d:Ge/MWC_In_BPM1/RotX        = 0.0 deg
d:Ge/MWC_In_BPM1/RotY        = 0.0 deg
d:Ge/MWC_In_BPM1/RotZ        = 0.0 deg
i:Ge/MWC_In_BPM1/NbOfLayers  = 2

d:Ge/MWC_In_BPM1/Layer1/RMin  = 0.0 mm
d:Ge/MWC_In_BPM1/Layer1/RMax  = 0.02 mm
s:Ge/MWC_In_BPM1/Layer1/Material = "Brass"
d:Ge/MWC_In_BPM1/Layer1/HL    = 85.0 mm
s:Ge/MWC_In_BPM1/Layer1/Align = "X"
d:Ge/MWC_In_BPM1/Layer1/PosZ  = 5.0 mm
s:Ge/MWC_In_BPM1/Layer1/DrawingStyle = "FullWireFrame"
dv:Ge/MWC_In_BPM1/Layer1/Displacement = 72
-71.9855 -69.9855 -67.9855 -65.9855 -63.9855 -61.9855 -59.9855 -57.9855 -55.9855 -53.
↪9855
-51.9855 -49.9855 -47.9855 -45.9855 -43.9855 -41.9855 -39.9855 -37.9855 -35.9855 -33.
↪9855
-31.9855 -29.9855 -27.9855 -25.9855 -23.9855 -21.9855 -19.9855 -17.9855 -15.9855 -13.
↪9855
-11.9855 -9.9855 -7.9855 -5.9855 -3.9855 -1.9855 0.0145 2.0145 4.0145 6.
↪0145
8.0145 10.0145 12.0145 14.0145 16.0145 18.0145 20.0145 22.0145 24.0145 26.
↪0145
28.0145 30.0145 32.0145 34.0145 36.0145 38.0145 40.0145 42.0145 44.0145 46.
↪0145
48.0145 50.0145 52.0145 54.0145 56.0145 58.0145 60.0145 62.0145 64.0145 66.
↪0145
68.0145 70.0145 mm

d:Ge/MWC_In_BPM1/Layer2/RMin  = 0.0 mm
d:Ge/MWC_In_BPM1/Layer2/RMax  = 0.02 mm
s:Ge/MWC_In_BPM1/Layer2/Material = "Brass"
d:Ge/MWC_In_BPM1/Layer2/HL    = 85.0 mm
s:Ge/MWC_In_BPM1/Layer2/Align = "Y"
d:Ge/MWC_In_BPM1/Layer2/PosZ  = -5.0 mm
s:Ge/MWC_In_BPM1/Layer2/DrawingStyle = "FullWireFrame"
s:Ge/MWC_In_BPM1/Layer2/Color  = "red"
dv:Ge/MWC_In_BPM1/Layer2/Displacement = 72
-71.9855 -69.9855 -67.9855 -65.9855 -63.9855 -61.9855 -59.9855 -57.9855 -55.9855 -53.
↪9855
-51.9855 -49.9855 -47.9855 -45.9855 -43.9855 -41.9855 -39.9855 -37.9855 -35.9855 -33.
↪9855
-31.9855 -29.9855 -27.9855 -25.9855 -23.9855 -21.9855 -19.9855 -17.9855 -15.9855 -13.
↪9855
-11.9855 -9.9855 -7.9855 -5.9855 -3.9855 -1.9855 0.0145 2.0145 4.0145 6.
↪0145
8.0145 10.0145 12.0145 14.0145 16.0145 18.0145 20.0145 22.0145 24.0145 26.
↪0145

```

(continues on next page)

(continued from previous page)

```

28.0145 30.0145 32.0145 34.0145 36.0145 38.0145 40.0145 42.0145 44.0145 46.
↪0145
48.0145 50.0145 52.0145 54.0145 56.0145 58.0145 60.0145 62.0145 64.0145 66.
↪0145
68.0145 70.0145 mm

d:Ge/ZMinusOf_BPM1      = Ge/BeamProfileMonitor1/TransZ + Ge/BeamProfileMonitor1/
↪MinusHL mm
d:Ge/ZPlusOf_PipeExtension = Ge/ZMinusOf_BPM1 + Ge/DistanceOf_BottomBPM1_To_
↪PipeExtension mm

d:Ge/FLOf_PipeExtension  = 80.0 cm
d:Ge/HLOf_PipeExtension  = Ge/FLOf_PipeExtension cm * 0.5
d:Ge/MinusHLOf_PipeExtension = Ge/FLOf_PipeExtension cm * -0.5

#####
#6. Beam pipe extension
#HLx, HLy, (9.2, 9.2)
#HLx, HLy, HLz (1
# 2 Mylar windows
# 2 sets of Multi-wire chamber
#####
s:Ge/PipeExtension/Parent      = "Gantry"
s:Ge/PipeExtension/Type        = "G4Trd"
s:Ge/PipeExtension/Material    = "Aluminum"
d:Ge/PipeExtension/HLX2        = 92.0 mm
d:Ge/PipeExtension/HLY2        = 92.0 mm
d:Ge/PipeExtension/HLX1        = 122.0 mm
d:Ge/PipeExtension/HLY1        = 122.0 mm
d:Ge/PipeExtension/HLZ         = Ge/HLOf_PipeExtension mm
d:Ge/PipeExtension/TransX      = 0.0 mm
d:Ge/PipeExtension/TransY      = 0.0 mm
d:Ge/PipeExtension/TransZ      = Ge/ZPlusOf_PipeExtension + Ge/MinusHLOf_PipeExtension mm
d:Ge/PipeExtension/RotX        = 0.0 deg
d:Ge/PipeExtension/RotY        = 0.0 deg
d:Ge/PipeExtension/RotZ        = 0.0 deg
b:Ge/PipeExtension/Include     = "TRUE"

s:Ge/PipeExtension/Gas/Parent  = "PipeExtension"
s:Ge/PipeExtension/Gas/Type    = "G4Trd"
s:Ge/PipeExtension/Gas/Material = Ge/PipeInQuads/Gas/Material
d:Ge/PipeExtension/Gas/HLX2    = 90.0 mm
d:Ge/PipeExtension/Gas/HLY2    = 90.0 mm
d:Ge/PipeExtension/Gas/HLX1    = 120.0 mm
d:Ge/PipeExtension/Gas/HLY1    = 120.0 mm
d:Ge/PipeExtension/Gas/HLZ     = Ge/PipeExtension/HLZ mm
d:Ge/PipeExtension/Gas/TransX  = 0.0 mm
d:Ge/PipeExtension/Gas/TransY  = 0.0 mm
d:Ge/PipeExtension/Gas/TransZ  = 0.0 mm
d:Ge/PipeExtension/Gas/RotX    = 0.0 deg
d:Ge/PipeExtension/Gas/RotY    = 0.0 deg
d:Ge/PipeExtension/Gas/RotZ    = 0.0 deg
b:Ge/PipeExtension/Gas/Include = "TRUE"

d:Ge/ZMinusOf_PipeExtension = Ge/PipeExtension/TransZ + Ge/MinusHLOf_PipeExtension mm
d:Ge/ZPlusOf_BPM2 = Ge/ZMinusOf_PipeExtension + Ge/DistanceOf_BottomPipeExtension_To_
↪BPM2 mm

```

(continues on next page)

(continued from previous page)

```
#####
#6. Beam profile monitor 2
# 2 Mylar windows
# 2 sets of Multi-wire chamber
#####
s:Ge/BeamProfileMonitor2/Parent      = "Gantry"
s:Ge/BeamProfileMonitor2/Type        = "TsBox"
s:Ge/BeamProfileMonitor2/Material    = "NiGas"
d:Ge/BeamProfileMonitor2/HLX         = 152.0 mm
d:Ge/BeamProfileMonitor2/HLY         = 152.0 mm
d:Ge/BeamProfileMonitor2/HLZ         = 25.0 mm
d:Ge/BeamProfileMonitor2/MinusHL     = Ge/BeamProfileMonitor2/HLZ mm * -1.0
d:Ge/BeamProfileMonitor2/TransX      = 0.0 mm
d:Ge/BeamProfileMonitor2/TransY      = 0.0 mm
d:Ge/BeamProfileMonitor2/TransZ      = Ge/ZPlusOf_BPM2 + Ge/BeamProfileMonitor2/MinusHL_
->mm
d:Ge/BeamProfileMonitor2/RotX        = 0.0 deg
d:Ge/BeamProfileMonitor2/RotY        = 0.0 deg
d:Ge/BeamProfileMonitor2/RotZ        = 0.0 deg
b:Ge/BeamProfileMonitor2/Include     = "TRUE"

s:Ge/BeamProfileMonitor2/Window1/Parent = "BeamProfileMonitor2"
s:Ge/BeamProfileMonitor2/Window1/Type   = "TsBox"
s:Ge/BeamProfileMonitor2/Window1/Material = "Mylar"
d:Ge/BeamProfileMonitor2/Window1/HLX    = 150.0 mm
d:Ge/BeamProfileMonitor2/Window1/HLY    = 150.0 mm
d:Ge/BeamProfileMonitor2/Window1/HLZ    = 0.01 mm
d:Ge/BeamProfileMonitor2/Window1/TransX = 0.0 mm
d:Ge/BeamProfileMonitor2/Window1/TransY = 0.0 mm
d:Ge/BeamProfileMonitor2/Window1/TransZ = 18.123 mm
d:Ge/BeamProfileMonitor2/Window1/RotX   = 0.0 deg
d:Ge/BeamProfileMonitor2/Window1/RotY   = 0.0 deg
d:Ge/BeamProfileMonitor2/Window1/RotZ   = 0.0 deg

s:Ge/BeamProfileMonitor2/Window2/Parent = "BeamProfileMonitor2"
s:Ge/BeamProfileMonitor2/Window2/Type   = "TsBox"
s:Ge/BeamProfileMonitor2/Window2/Material = "Mylar"
d:Ge/BeamProfileMonitor2/Window2/HLX    = 150.0 mm
d:Ge/BeamProfileMonitor2/Window2/HLY    = 150.0 mm
d:Ge/BeamProfileMonitor2/Window2/HLZ    = 0.01 mm
d:Ge/BeamProfileMonitor2/Window2/TransX = 0.0 mm
d:Ge/BeamProfileMonitor2/Window2/TransY = 0.0 mm
d:Ge/BeamProfileMonitor2/Window2/TransZ = -18.123 mm
d:Ge/BeamProfileMonitor2/Window2/RotX   = 0.0 deg
d:Ge/BeamProfileMonitor2/Window2/RotY   = 0.0 deg
d:Ge/BeamProfileMonitor2/Window2/RotZ   = 0.0 deg

s:Ge/MWC_In_BPM2/Parent      = "BeamProfileMonitor2"
s:Ge/MWC_In_BPM2/Type        = "TsMultiWireChamber"
s:Ge/MWC_In_BPM2/Material    = "parent"
d:Ge/MWC_In_BPM2/HLX         = 148.0 mm
d:Ge/MWC_In_BPM2/HLY         = 148.0 mm
d:Ge/MWC_In_BPM2/HLZ         = 13.0 mm
d:Ge/MWC_In_BPM2/TransX      = 0.0 mm
d:Ge/MWC_In_BPM2/TransY      = 0.0 mm
d:Ge/MWC_In_BPM2/TransZ      = 0.0 mm
```

(continues on next page)

(continued from previous page)

```

d:Ge/MWC_In_BPM2/RotX      = 0.0    deg
d:Ge/MWC_In_BPM2/RotY      = 0.0    deg
d:Ge/MWC_In_BPM2/RotZ      = 0.0    deg
i:Ge/MWC_In_BPM2/NbOfLayers = 2

d:Ge/MWC_In_BPM2/Layer1/RMin      = 0.0 mm
d:Ge/MWC_In_BPM2/Layer1/RMax      = 0.02 mm
s:Ge/MWC_In_BPM2/Layer1/Material  = "Brass"
d:Ge/MWC_In_BPM2/Layer1/HL        = 147.0 mm
s:Ge/MWC_In_BPM2/Layer1/Align     = "X"
d:Ge/MWC_In_BPM2/Layer1/PosZ      = 5.0 mm
s:Ge/MWC_In_BPM2/Layer1/DrawingStyle = "Solid"
dv:Ge/MWC_In_BPM2/Layer1/Displacement = 81
-140. -136.5 -133. -129.5 -126. -122.5 -119. -115.5
-112. -108.5 -105. -101.5 -98. -94.5 -91. -87.5
-84. -80.5 -77. -73.5 -70. -66.5 -63. -59.5
-56. -52.5 -49. -45.5 -42. -38.5 -35. -31.5
-28. -24.5 -21. -17.5 -14. -10.5 -7. -3.5
 0. 3.5 7. 10.5 14. 17.5 21. 24.5
28. 31.5 35. 38.5 42. 45.5 49. 52.5
56. 59.5 63. 66.5 70. 73.5 77. 80.5
84. 87.5 91. 94.5 98. 101.5 105. 108.5
112. 115.5 119. 122.5 126. 129.5 133. 136.5
140. mm

d:Ge/MWC_In_BPM2/Layer2/RMin      = 0.0 mm
d:Ge/MWC_In_BPM2/Layer2/RMax      = 0.02 mm
s:Ge/MWC_In_BPM2/Layer2/Material  = "Brass"
d:Ge/MWC_In_BPM2/Layer2/HL        = 147.0 mm
s:Ge/MWC_In_BPM2/Layer2/Align     = "Y"
d:Ge/MWC_In_BPM2/Layer2/PosZ      = -5.0 mm
s:Ge/MWC_In_BPM2/Layer2/DrawingStyle = "Solid"
s:Ge/MWC_In_BPM2/Layer2/Color      = "red"
dv:Ge/MWC_In_BPM2/Layer2/Displacement = 81
-140. -136.5 -133. -129.5 -126. -122.5 -119. -115.5
-112. -108.5 -105. -101.5 -98. -94.5 -91. -87.5
-84. -80.5 -77. -73.5 -70. -66.5 -63. -59.5
-56. -52.5 -49. -45.5 -42. -38.5 -35. -31.5
-28. -24.5 -21. -17.5 -14. -10.5 -7. -3.5
 0. 3.5 7. 10.5 14. 17.5 21. 24.5
28. 31.5 35. 38.5 42. 45.5 49. 52.5
56. 59.5 63. 66.5 70. 73.5 77. 80.5
84. 87.5 91. 94.5 98. 101.5 105. 108.5
112. 115.5 119. 122.5 126. 129.5 133. 136.5
140. mm

```

## 21.4 ScanningStationaryTarget.txt

```

# Use scanning nozzle to paint stationary target

includeFile = RasterScanningPattern.txt TargetInWaterPhantom.txt Viewer.txt

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

```

(continues on next page)

(continued from previous page)

```

b:Ts/PauseBeforeQuit = "False"

i:So/Example/NumberOfHistoriesInRun = 5
d:So/Example/BeamEnergy              = 145.00 MeV

d:Tf/TimelineEnd                    = 1 s
i:Tf/NumberOfSequentialTimes       = 100

```

## 21.5 ScanningTargetMovingHorizontal.txt

```

# Use scanning nozzle to paint target that is moving horizontally

includeFile = RasterScanningPattern.txt TargetInWaterPhantom.txt Viewer.txt

i:Tf/Verbosity = 1

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

d:Ge/X_max   = 1.5 cm
d:Ge/X_min   = Ge/X_max cm * -1.0
d:Ge/X_range = Ge/X_max cm * 2.0

# 1. Motion for Target
s:Tf/HorizontalMoveSine/Function      = "Sine"
d:Tf/HorizontalMoveSine/RepetitionInterval = 1.0 s
d:Tf/HorizontalMoveSine/Rate          = 0.36 deg/ms
d:Tf/HorizontalMoveSine/StartValue    = 0.0 deg

d:Tf/HorizontalMove/Value = 2.0 cm * Tf/HorizontalMoveSine/Value
d:Tf/HorizontalScan/Value = 0.15 tesla * Tf/HorizontalMoveSine/Value

# 2. Compensated BField
d:Tf/BField2ndTrackingField/Value = Tf/HorizontalScan/Value + Tf/BField2nd/Value tesla

i:Tf/NumberOfSequentialTimes = 100
d:Tf/TimelineEnd = 1 s

d:Ge/PlaneAtR80/TransX = Tf/HorizontalMove/Value cm
d:Ge/Dipole2/MagneticFieldStrength = Tf/BField2ndTrackingField/Value tesla

```

## 21.6 ScanningTargetMovingInDepth.txt

```

# Use scanning nozzle to paint target this is moving in depth

includeFile = RasterScanningPattern.txt TargetInWaterPhantom.txt Viewer.txt

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

b:Ts/PauseBeforeQuit = "False"

i:So/Example/NumberOfHistoriesInRun = 10

```

(continues on next page)

(continued from previous page)

```

d:Tf/TimelineEnd           = 1 s
i:Tf/NumberOfSequentialTimes = 100

#1. Target position and Energy varies as a sinusoidal
s:Tf/VerticalMoveCosine/Function = "Cosine"
d:Tf/VerticalMoveCosine/RepetitionInterval = 1.0 s
d:Tf/VerticalMoveCosine/Rate = 0.36 deg/ms
d:Tf/VerticalMoveCosine/StartValue = 0.0 deg

#2. Tracking target
d:Tf/VerticalMoveD_Cosine/Value = 18.6 mm * Tf/VerticalMoveCosine/Value
d:Tf/VerticalMoveR80/Value = -166.142 mm + Tf/VerticalMoveD_Cosine/Value

#3. Beam energy changing for tracking target
d:Tf/VerticalMoveE_Cosine/Value = -10.0 MeV * Tf/VerticalMoveCosine/Value
d:Tf/VerticalMoveEnergy/Value = 155.0 MeV + Tf/VerticalMoveE_Cosine/Value
d:So/Example/BeamEnergy = Tf/VerticalMoveEnergy/Value MeV

d:Ge/R80 = Tf/VerticalMoveR80/Value cm

#Fixed target for visual guidance
d:Ge/R80Fix = -140.50 mm
s:Ge/PlaneAtR80Fix/Parent = "WaterPhantom"
s:Ge/PlaneAtR80Fix/Type = "TsBox"
s:Ge/PlaneAtR80Fix/Material = "G4_WATER"
d:Ge/PlaneAtR80Fix/HLX = 5.0 cm
d:Ge/PlaneAtR80Fix/HLY = 5.0 cm
d:Ge/PlaneAtR80Fix/HLZ = 0.01 cm
d:Ge/PlaneAtR80Fix/TransX = 0. cm
d:Ge/PlaneAtR80Fix/TransY = 0. cm
d:Ge/PlaneAtR80Fix/TransZ = Ge/WaterPhantom/HLZ + Ge/R80Fix cm
d:Ge/PlaneAtR80Fix/RotX = 0.0 deg
d:Ge/PlaneAtR80Fix/RotY = 0.0 deg
d:Ge/PlaneAtR80Fix/RotZ = 0.0 deg
s:Ge/PlaneAtR80Fix/Color = "green"
i:Ge/PlaneAtR80Fix/XBins = 1
i:Ge/PlaneAtR80Fix/YBins = 1
i:Ge/PlaneAtR80Fix/ZBins = 1

d:Gr/ViewA/Theta = 90 deg
d:Gr/ViewA/Phi = 0 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 20 deg
u:Gr/ViewA/Zoom = 10

```

## 21.7 ScatteringNozzle.txt

```

# Scattering nozzle used in some demonstrations.

includeFile = ConstantsForNozzles.txt

#####
#1. Beam Exit Window
#100 um thickness Mylar window

```

(continues on next page)

(continued from previous page)

```
#####
s:Ge/ExitWindow/Parent      = "Gantry"
s:Ge/ExitWindow/Type       = "TsCylinder"
s:Ge/ExitWindow/Material   = "Mylar"
d:Ge/ExitWindow/RMin      = 0.0 mm
d:Ge/ExitWindow/RMax      = 50.0 mm
d:Ge/ExitWindow/HL        = 0.05 mm
d:Ge/ExitWindow/SPhi      = 0.0 deg
d:Ge/ExitWindow/DPhi      = 360.0 deg
d:Ge/ExitWindow/TransX    = 0.0 mm
d:Ge/ExitWindow/TransY    = 0.0 mm
d:Ge/ExitWindow/TransZ    = 299.8 cm
d:Ge/ExitWindow/RotX      = 0.0 deg
d:Ge/ExitWindow/RotY      = 0.0 deg
d:Ge/ExitWindow/RotZ      = 0.0 deg
b:Ge/ExitWindow/Include   = "TRUE"
s:Ge/ExitWindow/DrawingStyle = "Solid"

d:Ge/DistanceOf_CenterExitWindow_To_TopMon1 = -30.0 mm
d:Ge/ZPlusOf_Mon1 = Ge/ExitWindow/TransZ + Ge/DistanceOf_CenterExitWindow_To_TopMon1_
→cm

#####
#2. Monitor chamber 1.
# Filled with Nitrogen Gas, 3 layers
# Thickness, position, Material
#      1.0      Gold
#      0.0      Aluminum
#      -1.0     Gold
#####
s:Ge/MonitorChamber1/Parent = "Gantry"
s:Ge/MonitorChamber1/Type   = "TsCylinder"
s:Ge/MonitorChamber1/Material = "NiGas"
d:Ge/MonitorChamber1/RMin   = 0.0 mm
d:Ge/MonitorChamber1/RMax   = 100.0 mm
d:Ge/MonitorChamber1/HL     = 15.0 mm
d:Ge/MonitorChamber1/SPhi   = 0.0 deg
d:Ge/MonitorChamber1/DPhi   = 360.0 deg
d:Ge/MonitorChamber1/RotX   = 0.0 deg
d:Ge/MonitorChamber1/RotY   = 0.0 deg
d:Ge/MonitorChamber1/RotZ   = 0.0 deg
d:Ge/MonitorChamber1/TransX = 0.0 mm
d:Ge/MonitorChamber1/TransY = 0.0 mm
d:Ge/MonitorChamber1/MinusHL = Ge/MonitorChamber1/HL mm * -1.0
d:Ge/MonitorChamber1/TransZ = Ge/ZPlusOf_Mon1 + Ge/MonitorChamber1/MinusHL mm
b:Ge/MonitorChamber1/Include = "TRUE"

s:Ge/MonitorChamber1/Layer1/Parent = "MonitorChamber1"
s:Ge/MonitorChamber1/Layer1/Type   = "TsCylinder"
s:Ge/MonitorChamber1/Layer1/Material = "Gold"
d:Ge/MonitorChamber1/Layer1/RMin   = 0.0 mm
d:Ge/MonitorChamber1/Layer1/RMax   = 90.0 mm
d:Ge/MonitorChamber1/Layer1/HL     = 0.002 mm
d:Ge/MonitorChamber1/Layer1/SPhi   = 0.0 deg
d:Ge/MonitorChamber1/Layer1/DPhi   = 360.0 deg
d:Ge/MonitorChamber1/Layer1/RotX   = 0.0 deg
d:Ge/MonitorChamber1/Layer1/RotY   = 0.0 deg
```

(continues on next page)

(continued from previous page)

```

d:Ge/MonitorChamber1/Layer1/RotZ      = 0.0 deg
d:Ge/MonitorChamber1/Layer1/TransX    = 0.0 mm
d:Ge/MonitorChamber1/Layer1/TransY    = 0.0 mm
d:Ge/MonitorChamber1/Layer1/TransZ    = 10.0 mm
b:Ge/MonitorChamber1/Layer1/Include   = "TRUE"

s:Ge/MonitorChamber1/Layer2/Parent     = "MonitorChamber1"
s:Ge/MonitorChamber1/Layer2/Type       = "TsCylinder"
s:Ge/MonitorChamber1/Layer2/Material   = "Aluminum"
d:Ge/MonitorChamber1/Layer2/RMin       = 0.0 mm
d:Ge/MonitorChamber1/Layer2/RMax       = 90.0 mm
d:Ge/MonitorChamber1/Layer2/HL         = 0.003 mm
d:Ge/MonitorChamber1/Layer2/SPhi       = 0.0 deg
d:Ge/MonitorChamber1/Layer2/DPhi       = 360.0 deg
d:Ge/MonitorChamber1/Layer2/RotX       = 0.0 deg
d:Ge/MonitorChamber1/Layer2/RotY       = 0.0 deg
d:Ge/MonitorChamber1/Layer2/RotZ       = 0.0 deg
d:Ge/MonitorChamber1/Layer2/TransX     = 0.0 mm
d:Ge/MonitorChamber1/Layer2/TransY     = 0.0 mm
d:Ge/MonitorChamber1/Layer2/TransZ     = 0.0 mm
b:Ge/MonitorChamber1/Layer2/Include    = "TRUE"

s:Ge/MonitorChamber1/Layer3/Parent     = "MonitorChamber1"
s:Ge/MonitorChamber1/Layer3/Type       = "TsCylinder"
s:Ge/MonitorChamber1/Layer3/Material   = "Gold"
d:Ge/MonitorChamber1/Layer3/RMin       = 0.0 mm
d:Ge/MonitorChamber1/Layer3/RMax       = 90.0 mm
d:Ge/MonitorChamber1/Layer3/HL         = 0.003 mm
d:Ge/MonitorChamber1/Layer3/SPhi       = 0.0 deg
d:Ge/MonitorChamber1/Layer3/DPhi       = 360.0 deg
d:Ge/MonitorChamber1/Layer3/RotX       = 0.0 deg
d:Ge/MonitorChamber1/Layer3/RotY       = 0.0 deg
d:Ge/MonitorChamber1/Layer3/RotZ       = 0.0 deg
d:Ge/MonitorChamber1/Layer3/TransX     = 0.0 mm
d:Ge/MonitorChamber1/Layer3/TransY     = 0.0 mm
d:Ge/MonitorChamber1/Layer3/TransZ     = -10.0 mm
b:Ge/MonitorChamber1/Layer3/Include    = "TRUE"

#####
# First Scatterer:
#####
s:Ge/Scatterer1/Type      = "Group"
s:Ge/Scatterer1/Parent    = "Gantry"
d:Ge/Scatterer1/TransX    = 0.0 cm
d:Ge/Scatterer1/TransY    = 0.0 cm
d:Ge/Scatterer1/TransZ    = 280. cm
d:Ge/Scatterer1/RotX      = 0. deg
d:Ge/Scatterer1/RotY      = 0. deg
d:Ge/Scatterer1/RotZ      = 0. deg
s:Ge/Scatterer1/Message   = "Constructing IBA TsScatterer1"

# Center for rotation of scatterers:
d:Ge/Scatterer1/RotationCenterY = 5.0 cm
# Defining the in and out of beam rotation angle
d:Ge/Scatterer1/RotZ_InBeam      = 180.0 deg
d:Ge/Scatterer1/RotZ_OutOfBeam   = 0.0 deg

```

(continues on next page)



(continued from previous page)

```

d:Ge/Scatterer1/Lollipop1 = Ge/Scatterer1/RotZ_OutOfBeam deg
d:Ge/Scatterer1/Lollipop2 = Ge/Scatterer1/RotZ_OutOfBeam deg
d:Ge/Scatterer1/Lollipop3 = Ge/Scatterer1/RotZ_OutOfBeam deg

# Lollipop 1:
s:Ge/Scatterer1/L1/Type = "Group"
s:Ge/Scatterer1/L1/Parent = "Scatterer1"
d:Ge/Scatterer1/L1/TransX = 0.0 cm
d:Ge/Scatterer1/L1/TransY = Ge/Scatterer1/RotationCenterY cm
d:Ge/Scatterer1/L1/TransZ = 0.0 cm
d:Ge/Scatterer1/L1/RotX = 0. deg
d:Ge/Scatterer1/L1/RotY = 0. deg
d:Ge/Scatterer1/L1/RotZ = Ge/Scatterer1/Lollipop1 deg

# L1 Lead Foil:
s:Ge/Scatterer1/L1/LeadFoil/Parent = "Scatterer1/L1"
s:Ge/Scatterer1/L1/LeadFoil/Type = "TsBox"
s:Ge/Scatterer1/L1/LeadFoil/Material = "Lead"
d:Ge/Scatterer1/L1/LeadFoil/HLX = 2.15 cm
d:Ge/Scatterer1/L1/LeadFoil/HLY = 2.15 cm
d:Ge/Scatterer1/L1/LeadFoil/HLZ = 0.01 cm
d:Ge/Scatterer1/L1/LeadFoil/TransX = 0.0 cm
d:Ge/Scatterer1/L1/LeadFoil/TransY = Ge/Scatterer1/RotationCenterY cm
d:Ge/Scatterer1/L1/LeadFoil/TransZ = 1.0 cm
d:Ge/Scatterer1/L1/LeadFoil/RotX = 0. deg
d:Ge/Scatterer1/L1/LeadFoil/RotY = 0. deg
d:Ge/Scatterer1/L1/LeadFoil/RotZ = 0. deg

# Lollipop 2:
s:Ge/Scatterer1/L2/Type = "Group"
s:Ge/Scatterer1/L2/Parent = "Scatterer1"
d:Ge/Scatterer1/L2/TransX = 0.0 cm
d:Ge/Scatterer1/L2/TransY = Ge/Scatterer1/RotationCenterY cm
d:Ge/Scatterer1/L2/TransZ = 0.0 cm
d:Ge/Scatterer1/L2/RotX = 0. deg
d:Ge/Scatterer1/L2/RotY = 0. deg
d:Ge/Scatterer1/L2/RotZ = Ge/Scatterer1/Lollipop2 deg

# L2 Lead Foil:
s:Ge/Scatterer1/L2/LexanFoil/Parent = "Scatterer1/L2"
s:Ge/Scatterer1/L2/LexanFoil/Type = "TsBox"
s:Ge/Scatterer1/L2/LexanFoil/Material = "Lexan"
d:Ge/Scatterer1/L2/LexanFoil/HLX = 2.5 cm
d:Ge/Scatterer1/L2/LexanFoil/HLY = 2.5 cm
d:Ge/Scatterer1/L2/LexanFoil/HLZ = 0.2 cm
d:Ge/Scatterer1/L2/LexanFoil/TransX = 0.0 cm
d:Ge/Scatterer1/L2/LexanFoil/TransY = Ge/Scatterer1/RotationCenterY cm
d:Ge/Scatterer1/L2/LexanFoil/TransZ = 0.0 cm
d:Ge/Scatterer1/L2/LexanFoil/RotX = 0. deg
d:Ge/Scatterer1/L2/LexanFoil/RotY = 0. deg
d:Ge/Scatterer1/L2/LexanFoil/RotZ = 0. deg

# Lollipop 3:
s:Ge/Scatterer1/L3/Type = "Group"
s:Ge/Scatterer1/L3/Parent = "Scatterer1"

```

(continues on next page)

(continued from previous page)

```

d:Ge/Scatterer1/L3/TransX = 0.0 cm
d:Ge/Scatterer1/L3/TransY = Ge/Scatterer1/RotationCenterY cm
d:Ge/Scatterer1/L3/TransZ = 0.0 cm
d:Ge/Scatterer1/L3/RotX = 0. deg
d:Ge/Scatterer1/L3/RotY = 0. deg
d:Ge/Scatterer1/L3/RotZ = Ge/Scatterer1/Lollipop2 deg

# L2 Lead Foil:
s:Ge/Scatterer1/L3/LeadFoil/Parent = "Scatterer1/L3"
s:Ge/Scatterer1/L3/LeadFoil/Type = "TsBox"
s:Ge/Scatterer1/L3/LeadFoil/Material = "Lead"
d:Ge/Scatterer1/L3/LeadFoil/HLX = 2.5 cm
d:Ge/Scatterer1/L3/LeadFoil/HLY = 2.5 cm
d:Ge/Scatterer1/L3/LeadFoil/HLZ = 0.02 cm
d:Ge/Scatterer1/L3/LeadFoil/TransX = 0.0 cm
d:Ge/Scatterer1/L3/LeadFoil/TransY = Ge/Scatterer1/RotationCenterY cm
d:Ge/Scatterer1/L3/LeadFoil/TransZ = -1.0 cm
d:Ge/Scatterer1/L3/LeadFoil/RotX = 0. deg
d:Ge/Scatterer1/L3/LeadFoil/RotY = 0. deg
d:Ge/Scatterer1/L3/LeadFoil/RotZ = 0. deg

#####
#. Range modulator
#(Copied from SpecialComponents/RangeModulator.txt
#####
s:Ge/RMW_1/Type = "TsRangeModulator"
s:Ge/RMW_1/Material = "Parent"
s:Ge/RMW_1/Parent = "Gantry"
d:Ge/RMW_1/TransX = 10.0 cm
d:Ge/RMW_1/TransY = 0.0 cm
d:Ge/RMW_1/TransZ = 250.0 cm
d:Ge/RMW_1/RotX = 0.0 deg
d:Ge/RMW_1/RotY = 0.0 deg
d:Ge/RMW_1/RotZ = 0.0 deg
b:Ge/RMW_1/Invisible = "TRUE"
b:Ge/RMW_1/PrintInformation = "True"

d:Ge/RMW_1/HeightOfUpper = 150 mm
d:Ge/RMW_1/HeightOfMiddle = 1.0 mm
d:Ge/RMW_1/HeightOfLower = 9.0 mm

d:Ge/RMW_1/Shell/Rin = 15.0 cm
d:Ge/RMW_1/Shell/Rout = 15.5 cm
s:Ge/RMW_1/Shell/Material = "Aluminum"
s:Ge/RMW_1/Shell/Color = "grey"
s:Ge/RMW_1/Shell/DrawingStyle = "Solid"
i:Ge/RMW_1/Shell/VisSegsPerCircle = 360

d:Ge/RMW_1/Hub/Rin = 6.0 cm
d:Ge/RMW_1/Hub/Rout = 7.0 cm
s:Ge/RMW_1/Hub/Material = "Aluminum"
s:Ge/RMW_1/Hub/Color = "grey"
s:Ge/RMW_1/Hub/DrawingStyle = "Solid"
i:Ge/RMW_1/Hub/VisSegsPerCircle = 360

#Upper tracks

```

(continues on next page)

(continued from previous page)

```

dv:Ge/RMW_1/Upper/RadialDivisions = 1 11.0 cm
s:Ge/RMW_1/Upper/Track1/Pattern   = "LexanBlockT1"
s:Ge/RMW_1/Upper/Track2/Pattern   = "NULL"
#Middle tracks
dv:Ge/RMW_1/Middle/RadialDivisions = 1 11.0 cm
s:Ge/RMW_1/Middle/Track1/Pattern   = "InterfaceDisk"
s:Ge/RMW_1/Middle/Track2/Pattern   = "HoleTrackDisk"
#Lower tracks
dv:Ge/RMW_1/Lower/RadialDivisions = 1 11.0 cm
s:Ge/RMW_1/Lower/Track1/Pattern   = "LeadBlockT1"
s:Ge/RMW_1/Lower/Track2/Pattern   = "NULL"

#A track pattern: 14 blocks of Lexan
d:Ge/LexanBlockT1/Offset = 0.0 deg
dv:Ge/LexanBlockT1/Angles = 14
    5.00 115.00 146.50 173.2 195.07
216.15 230.14 243.00 255.5 270.60
282.20 294.60 306.20 324.00 deg
dv:Ge/LexanBlockT1/Heights = 14
    77.0 82.0 87.0 92.15 95.0
100.4 106.0 110.2 115.3 119.5
124.0 128.8 132.00 60.0 mm
sv:Ge/LexanBlockT1/Materials = 14
"Lexan" "Lexan" "Lexan" "Lexan" "Lexan"
"Lexan" "Lexan" "Lexan" "Lexan" "Lexan"
"Lexan" "Lexan" "Lexan" "Brass"

#A track pattern of single block
dv:Ge/InterfaceDisk/Angles = 1 0.0 deg
dv:Ge/InterfaceDisk/Heights = 1 1.0 mm
sv:Ge/InterfaceDisk/Materials = 1 "Aluminum"

#A track pattern of two blocks but one block is a hole
dv:Ge/HoleTrackDisk/Angles = 2 90.0 110.0 deg
dv:Ge/HoleTrackDisk/Heights = 2 0.0 1.0 mm
sv:Ge/HoleTrackDisk/Materials = 2 "NULL" "Aluminum"

#A track pattern of 8 Lead blocks
dv:Ge/LeadBlockT1/Angles = 9
    5.00 115.00 146.50 173.2 195.07
216.15 230.14 243.00 255.5 deg
dv:Ge/LeadBlockT1/Heights = 9
    0.890 0.75 0.60 0.52 0.40
    0.30 0.16 0.070 0.0 mm
sv:Ge/LeadBlockT1/Materials = 9
"Lead" "Lead" "Lead" "Lead" "Lead"
"Lead" "Lead" "Lead" "NULL"

#####
# Second Scatterer:
#####
s:Ge/Scatterer2/Type = "Group"
s:Ge/Scatterer2/Parent = "World"
d:Ge/Scatterer2/TransX = 0.0 cm
d:Ge/Scatterer2/TransY = 0.0 cm
d:Ge/Scatterer2/TransZ = 170. cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/Scatterer2/RotX          = 0. deg
d:Ge/Scatterer2/RotY          = 0. deg
d:Ge/Scatterer2/RotZ          = 0. deg
s:Ge/Scatterer2/Message       = "Constructing IBA TsScatterer2"

# Scatterer 2 Brass Box (Mother Volume)
s:Ge/Scatterer2/BrassBox/Parent      = "Scatterer2"
s:Ge/Scatterer2/BrassBox/Type        = "TsBox"
s:Ge/Scatterer2/BrassBox/Material    = "Brass"
d:Ge/Scatterer2/BrassBox/HLX         = 30.0 cm
d:Ge/Scatterer2/BrassBox/HLY         = 30.0 cm
d:Ge/Scatterer2/BrassBox/HLZ         = 3.0 cm
d:Ge/Scatterer2/BrassBox/TransX      = 0.0 cm
d:Ge/Scatterer2/BrassBox/TransY      = 0.0 cm
d:Ge/Scatterer2/BrassBox/TransZ      = 0.0 cm
d:Ge/Scatterer2/BrassBox/RotX        = 0. deg
d:Ge/Scatterer2/BrassBox/RotY        = 0. deg
d:Ge/Scatterer2/BrassBox/RotZ        = 0. deg

# Scatter Holder Group:
s:Ge/Scatterer2/Holder/Type          = "Group"
s:Ge/Scatterer2/Holder/Parent        = "Scatterer2/BrassBox"
d:Ge/Scatterer2/Holder/TransX        = 10.0 cm
d:Ge/Scatterer2/Holder/TransY        = 10.0 cm
d:Ge/Scatterer2/Holder/TransZ        = 0.0 cm
d:Ge/Scatterer2/Holder/RotX          = 0. deg
d:Ge/Scatterer2/Holder/RotY          = 0. deg
d:Ge/Scatterer2/Holder/RotZ          = 0. deg

# Setting up the rotations for selecting a scatterer, this can be made Gantry_
↳specific:
d:Ge/Scatterer2/RotZForS1            = 180. deg
d:Ge/Scatterer2/RotZForS2            = 0. deg

# Scatter Hole 1
s:Ge/Scatterer2/Hole1/Type           = "Group"
s:Ge/Scatterer2/Hole1/Parent         = "Scatterer2/Holder"
d:Ge/Scatterer2/Hole1/TransX         = 10.0 cm
d:Ge/Scatterer2/Hole1/TransY         = 10.0 cm
d:Ge/Scatterer2/Hole1/TransZ         = 0.0 cm
d:Ge/Scatterer2/Hole1/RotX           = 0. deg
d:Ge/Scatterer2/Hole1/RotY           = 0. deg
d:Ge/Scatterer2/Hole1/RotZ           = 0. deg

# Scatter Hole 2
s:Ge/Scatterer2/Hole2/Type           = "Group"
s:Ge/Scatterer2/Hole2/Parent         = "Scatterer2/Holder"
d:Ge/Scatterer2/Hole2/TransX         = -10.0 cm
d:Ge/Scatterer2/Hole2/TransY         = -10.0 cm
d:Ge/Scatterer2/Hole2/TransZ         = 0.0 cm
d:Ge/Scatterer2/Hole2/RotX           = 0. deg
d:Ge/Scatterer2/Hole2/RotY           = 0. deg
d:Ge/Scatterer2/Hole2/RotZ           = 0. deg

# Scatterer 1 (no scattering)
# Air Tube: (no scattering, for pencil beam etc)
s:Ge/Scatterer2/S1/Parent             = "Scatterer2/Hole1"

```

(continues on next page)

(continued from previous page)

```

s:Ge/Scatterer2/S1/Type           = "TsCylinder"
s:Ge/Scatterer2/S1/Material       = "World"
d:Ge/Scatterer2/S1/RMin          = 0.0 cm
d:Ge/Scatterer2/S1/RMax          = 5.0 cm
d:Ge/Scatterer2/S1/HL           = 3.0 cm
d:Ge/Scatterer2/S1/TransX        = 0.0 cm
d:Ge/Scatterer2/S1/TransY        = 0.0 cm
d:Ge/Scatterer2/S1/TransZ        = 0.0 cm
d:Ge/Scatterer2/S1/RotX          = 0.0 deg
d:Ge/Scatterer2/S1/RotY          = 0.0 deg
d:Ge/Scatterer2/S1/RotZ          = 0.0 deg
d:Ge/Scatterer2/S1/SPhi          = 0.0 deg
d:Ge/Scatterer2/S1/DPhi          = 360.0 deg

# Scatterer 2 (example second scatterer)
# Air Hole for Scatterer 2:
s:Ge/Scatterer2/S2/Parent         = "Scatterer2/Hole2"
s:Ge/Scatterer2/S2/Type           = "TsCylinder"
s:Ge/Scatterer2/S2/Material       = "World"
d:Ge/Scatterer2/S2/RMin          = 0.0 cm
d:Ge/Scatterer2/S2/RMax          = 5.0 cm
d:Ge/Scatterer2/S2/HL           = 3.0 cm
d:Ge/Scatterer2/S2/TransX        = 0.0 cm
d:Ge/Scatterer2/S2/TransY        = 0.0 cm
d:Ge/Scatterer2/S2/TransZ        = 0.0 cm
d:Ge/Scatterer2/S2/RotX          = 0.0 deg
d:Ge/Scatterer2/S2/RotY          = 0.0 deg
d:Ge/Scatterer2/S2/RotZ          = 0.0 deg
d:Ge/Scatterer2/S2/SPhi          = 0.0 deg
d:Ge/Scatterer2/S2/DPhi          = 360.0 deg

# Lead Polycone Scatterer2:
s:Ge/Scatterer2/S2/LeadPolycone/Parent = "Scatterer2/S2"
s:Ge/Scatterer2/S2/LeadPolycone/Type   = "G4HPolycone"
s:Ge/Scatterer2/S2/LeadPolycone/Material = "Lead"
dv:Ge/Scatterer2/S2/LeadPolycone/RInner = 31 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
↪0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
↪0.0 0.0 cm
dv:Ge/Scatterer2/S2/LeadPolycone/ROuter = 31 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7
↪0.8 0.9 1.0 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8
↪2.9 3.0 cm
dv:Ge/Scatterer2/S2/LeadPolycone/Z      = 31 -0.1 -0.097 -0.094 -0.09 -0.086 -
↪0.082 -0.078 -0.074 -0.07 -0.065 -0.06 -0.055 -0.05 -0.045 -0.04 -0.036 -0.033 -0.
↪03 -0.027 -0.024 -0.021 -0.018 -0.015 -0.013 -0.011 -0.009 -0.007 -0.006 -0.005 -0.
↪004 -0.003 cm
d:Ge/Scatterer2/S2/LeadPolycone/TransX = 0.0 cm
d:Ge/Scatterer2/S2/LeadPolycone/TransY = 0.0 cm
d:Ge/Scatterer2/S2/LeadPolycone/TransZ = 0.0 cm
d:Ge/Scatterer2/S2/LeadPolycone/RotX   = 0.0 deg
d:Ge/Scatterer2/S2/LeadPolycone/RotY   = 0.0 deg
d:Ge/Scatterer2/S2/LeadPolycone/RotZ   = 0.0 deg
d:Ge/Scatterer2/S2/LeadPolycone/PhiStart = 0.0 deg
d:Ge/Scatterer2/S2/LeadPolycone/PhiTotal = 360.0 deg

# Lexan Polycone Scatterer2:
s:Ge/Scatterer2/S2/LexanPolycone/Parent = "Scatterer2/S2"
s:Ge/Scatterer2/S2/LexanPolycone/Type   = "G4HPolycone"

```

(continues on next page)

(continued from previous page)

```

s:Ge/Scatterer2/S2/LexanPolycone/Material          = "Lexan"
dv:Ge/Scatterer2/S2/LexanPolycone/RInner           = 41 4.0 3.9 3.8 3.7 3.6 3.5 3.4 3.3
→3.2 3.1 3.0 2.9 2.8 2.7 2.6 2.5 2.4 2.3 2.2 2.1 2.0 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2
→1.1 1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0 cm
dv:Ge/Scatterer2/S2/LexanPolycone/ROuter           = 41 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0
→4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0
→4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 4.0 cm
dv:Ge/Scatterer2/S2/LexanPolycone/Z               = 41 1.0 1.0 1.0 1.0 1.0 1.0 1.0 0.98 0.
→96 0.94 0.9 0.86 0.82 0.78 0.75 0.72 0.69 0.66 0.63 0.6 0.57 0.54 0.51 0.48 0.46 0.
→44 0.42 0.39 0.37 0.35 0.33 0.31 0.29 0.27 0.25 0.23 0.21 0.2 0.19 0.18 0.17 0.16 cm
d:Ge/Scatterer2/S2/LexanPolycone/TransX           = 0.0 cm
d:Ge/Scatterer2/S2/LexanPolycone/TransY           = 0.0 cm
d:Ge/Scatterer2/S2/LexanPolycone/TransZ           = 0.0 cm
d:Ge/Scatterer2/S2/LexanPolycone/RotX             = 0.0 deg
d:Ge/Scatterer2/S2/LexanPolycone/RotY             = 0.0 deg
d:Ge/Scatterer2/S2/LexanPolycone/RotZ             = 0.0 deg
d:Ge/Scatterer2/S2/LexanPolycone/PhiStart         = 0.0 deg
d:Ge/Scatterer2/S2/LexanPolycone/PhiTotal         = 360.0 deg

#####
# Snout:
#####
s:Ge/Snout/Type          = "Group"
s:Ge/Snout/Parent        = "Gantry"
d:Ge/Snout/TransX        = 0. m
d:Ge/Snout/TransY        = 0. m
d:Ge/Snout/TransZ        = 50. cm
d:Ge/Snout/RotX          = 0. deg
d:Ge/Snout/RotY          = 0. deg
d:Ge/Snout/RotZ          = 0. deg
s:Ge/Snout/Message       = "Constructing Snout"

# Snout:
s:Ge/Snout/Block/Parent  = "Snout"
s:Ge/Snout/Block/Type    = "TsBox"
s:Ge/Snout/Block/Material = "Brass"
d:Ge/Snout/Block/HLX     = 20.0 cm
d:Ge/Snout/Block/HLY     = 20.0 cm
d:Ge/Snout/Block/HLZ     = 3.5 cm
d:Ge/Snout/Block/TransX  = 0. cm
d:Ge/Snout/Block/TransY  = 0. cm
d:Ge/Snout/Block/TransZ  = 0. cm
d:Ge/Snout/Block/RotX    = 0. deg
d:Ge/Snout/Block/RotY    = 0. deg
d:Ge/Snout/Block/RotZ    = 0. deg

# Snout: Hole in Brass Block
s:Ge/Snout/Hole/Parent   = "Snout/Block"
s:Ge/Snout/Hole/Type     = "TsCylinder"
s:Ge/Snout/Hole/Material = "Brass"
d:Ge/Snout/Hole/RMin     = 0.0 cm
d:Ge/Snout/Hole/RMax     = 4.5 cm
d:Ge/Snout/Hole/HL       = 3.5 cm
d:Ge/Snout/Hole/TransX   = 0.0 cm
d:Ge/Snout/Hole/TransY   = 0.0 cm
d:Ge/Snout/Hole/TransZ   = 0.0 cm
d:Ge/Snout/Hole/RotX     = 0.0 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/Snout/Hole/RotY           = 0.0 deg
d:Ge/Snout/Hole/RotZ           = 0.0 deg
d:Ge/Snout/Hole/SPhi           = 0.0 deg
d:Ge/Snout/Hole/DPhi           = 360.0 deg

# Snout: Brass Cone
s:Ge/Snout/BrassCone/Parent     = "Snout"
s:Ge/Snout/BrassCone/Type       = "G4Cons"
s:Ge/Snout/BrassCone/Material   = "Brass"
d:Ge/Snout/BrassCone/RMin1      = 3.5 cm
d:Ge/Snout/BrassCone/RMax1      = 6.0 cm
d:Ge/Snout/BrassCone/RMin2      = 3.8 cm
d:Ge/Snout/BrassCone/RMax2      = 6.0 cm
d:Ge/Snout/BrassCone/HL         = 3.5 cm
d:Ge/Snout/BrassCone/TransX     = 0.0 cm
d:Ge/Snout/BrassCone/TransY     = 0.0 cm
d:Ge/Snout/BrassCone/TransZ     = -7.0 cm
d:Ge/Snout/BrassCone/RotX       = 0.0 deg
d:Ge/Snout/BrassCone/RotY       = 0.0 deg
d:Ge/Snout/BrassCone/RotZ       = 0.0 deg
d:Ge/Snout/BrassCone/SPhi       = 0.0 deg
d:Ge/Snout/BrassCone/DPhi       = 360.0 deg

#####
# Aperture
#####
s:Ge/Aperture/Type              = "TsAperture"
s:Ge/Aperture/Parent            = "Snout"
s:Ge/Aperture/InputFile         = "ApertureFileIn.ap"
s:Ge/Aperture/FileFormat        = "XYCoordinates"
b:Ge/Aperture/PrintPoints       = "True"
s:Ge/Aperture/Material          = "Brass"
d:Ge/Aperture/RMax               = 4.5 cm
d:Ge/Aperture/HL                 = 2.5 cm
d:Ge/Aperture/TransX             = 0.0 cm
d:Ge/Aperture/TransY             = 0.0 cm
d:Ge/Aperture/TransZ             = -13.0 cm
d:Ge/Aperture/RotX               = 0.0 deg
d:Ge/Aperture/RotY               = 0.0 deg
d:Ge/Aperture/RotZ               = 0.0 deg
d:Ge/Aperture/LowerEdge          = Ge/Aperture/TransZ - Ge/Aperture/HL cm
s:Ge/Aperture/Message            = "Constructing Aperture"

#####
# Compensator
#####
s:Ge/Compensator/Type            = "TsCompensator"
s:Ge/Compensator/Parent          = "Snout"
s:Ge/Compensator/Material        = "CompensatorLucite"
d:Ge/Compensator/RMax             = 5.5 cm
d:Ge/Compensator/TransX           = 0. cm
d:Ge/Compensator/TransY           = 0. cm
dc:Ge/Compensator/Thickness       = 0. cm # will be reset to actual thickness when
→compensator is read in
d:Ge/Compensator/HL               = 0.5 * Ge/Compensator/Thickness cm
d:Ge/Compensator/TransZ           = Ge/Aperture/LowerEdge - Ge/Compensator/HL cm
d:Ge/Compensator/LowerEdge        = Ge/Compensator/TransZ - Ge/Compensator/HL cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/Compensator/RotX      = 0. deg
d:Ge/Compensator/RotY      = 0. deg
d:Ge/Compensator/RotZ      = 0. deg
s:Ge/Compensator/InputFile  = "CompensatorFileInRowsDepths.rc"
s:Ge/Compensator/FileFormat = "RowsAndDepths"
s:Ge/Compensator/Method     = "ExtrudedSolid" # Polyhedra, ExtrudedSolid,
↳SubtractionCylinders or UnionCylinders
d:Ge/Compensator/XTolerance = 1. mm
d:Ge/Compensator/YTolerance = 1. mm
b:Ge/Compensator/PrintPoints = "True"
s:Ge/Compensator/Message    = "Constructing Compensator"

#####
# Phase space volume
#####
s:Ge/ZPhaseSpaceVol/Type      = "TsBox"
s:Ge/ZPhaseSpaceVol/Parent    = "Snout"
s:Ge/ZPhaseSpaceVol/Material  = "Parent"
d:Ge/ZPhaseSpaceVol/HLX       = 20. cm
d:Ge/ZPhaseSpaceVol/HLY       = 20. cm
d:Ge/ZPhaseSpaceVol/HLZ       = 0.05 mm
d:Ge/ZPhaseSpaceVol/TransX     = 0. m
d:Ge/ZPhaseSpaceVol/TransY     = 0. m
d:Ge/ZPhaseSpaceVol/TransZ     = -0.015 cm + Ge/Compensator/LowerEdge
d:Ge/ZPhaseSpaceVol/RotX       = 0. deg
d:Ge/ZPhaseSpaceVol/RotY       = 0. deg
d:Ge/ZPhaseSpaceVol/RotZ       = 0. deg

#####
# Graphics:
#####
s:Gr/ViewA/Type              = "OpenGL"
i:Gr/ViewA/WindowSizeX       = 1024
i:Gr/ViewA/WindowSizeY       = 768
d:Gr/ViewA/Theta              = 30 deg
d:Gr/ViewA/Phi                = 20 deg
s:Gr/ViewA/Projection         = "Perspective"
d:Gr/ViewA/PerspectiveAngle   = 30 deg
u:Gr/ViewA/Zoom               = 4.
u:Gr/ViewA/TransX             = -.8
u:Gr/ViewA/TransY             = -.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
b:Gr/ViewA/IncludeAxes        = "true"
s:Gr/ViewA/AxesComponent      = "Gantry"
d:Gr/ViewA/AxesSize           = 0.5 m

i:Ts/ShowHistoryCountAtInterval = 0

```



## 21.8 ScanningNozzle\_QtTest.txt

## 21.9 ScatteringNozzle\_run.txt

```
# Use scattering nozzle to generate spread out Bragg peak

includeFile = ScatteringNozzle.txt TsAnalysisWaterPhantom.txt

d:Ge/RMW_1/RotZ = Tf/RMW_Rotation/Value deg

i:So/Example/NumberOfHistoriesInRun = 3

b:Ge/CheckForOverlaps = "t"

d:Tf/TimelineEnd = 100.0 ms
i:Tf/NumberOfSequentialTimes = 100

s:Tf/RMW_Rotation/Function = "Linear deg"
d:Tf/RMW_Rotation/Rate = 3.6 deg/ms
d:Tf/RMW_Rotation/StartValue = 0.0 deg
d:Tf/RMW_Rotation/RepetitionInterval = 100. ms

b:Ts/PauseBeforeQuit = "False"
i:Ts/ShowHistoryCountAtInterval = 0

# Setting first scatterers:
s:Tf/Scatterer1/L1/Function = "Step"
dv:Tf/Scatterer1/L1/Times = 5 20 40 60 80 100 ms
dv:Tf/Scatterer1/L1/Values = 5 Ge/Scatterer1/RotZ_OutOfBeam Ge/Scatterer1/RotZ_
↪InBeam Ge/Scatterer1/RotZ_OutOfBeam Ge/Scatterer1/RotZ_InBeam Ge/Scatterer1/RotZ_
↪OutOfBeam deg

s:Tf/Scatterer1/L2/Function = "Step"
dv:Tf/Scatterer1/L2/Times = 2 30 100 ms
dv:Tf/Scatterer1/L2/Values = 2 Ge/Scatterer1/RotZ_InBeam Ge/Scatterer1/RotZ_
↪OutOfBeam deg

s:Tf/Scatterer1/L3/Function = "Step"
dv:Tf/Scatterer1/L3/Times = 4 20 40 80 100 ms
dv:Tf/Scatterer1/L3/Values = 4 Ge/Scatterer1/RotZ_InBeam Ge/Scatterer1/RotZ_
↪OutOfBeam Ge/Scatterer1/RotZ_InBeam Ge/Scatterer1/RotZ_OutOfBeam deg

Ge/Scatterer1/Lollipop1 = Tf/Scatterer1/L1/Value deg
Ge/Scatterer1/Lollipop2 = Tf/Scatterer1/L2/Value deg
Ge/Scatterer1/Lollipop3 = Tf/Scatterer1/L3/Value deg

# Setting second scatterer:
s:Tf/Scatterer2/Function = "Step"
dv:Tf/Scatterer2/Times = 2 50 100 ms
dv:Tf/Scatterer2/Values = 2 Ge/Scatterer2/RotZForS2 Ge/Scatterer2/RotZForS1 deg

d:Ge/Scatterer2/Holder/RotZ = Tf/Scatterer2/Value deg

# Show time feature steps on console
i:Tf/Verbosity = 1
```

## 21.10 ScanningNozzle\_run\_QtTest.txt

## 21.11 TargetInWaterPhantom.txt

```
# Target used in some demonstrations.

includeFile = ConstantsForNozzles.txt

s:Ge/WaterPhantom/Type      = "TsBox"
s:Ge/WaterPhantom/Parent    = "World"
s:Ge/WaterPhantom/Material   = "G4_WATER"
d:Ge/WaterPhantom/HLX       = 8.0 cm
d:Ge/WaterPhantom/HLY       = 8.0 cm
d:Ge/WaterPhantom/HLZ       = 10.0 cm
d:Ge/WaterPhantom/MinusHLZ  = Ge/WaterPhantom/HLZ cm * -1.0
d:Ge/WaterPhantom/TransX    = 0. cm
d:Ge/WaterPhantom/TransY    = 0. cm
d:Ge/WaterPhantom/TransZ    = Ge/WaterPhantom/MinusHLZ cm
d:Ge/WaterPhantom/RotX      = 0. deg
d:Ge/WaterPhantom/RotY      = 0. deg
d:Ge/WaterPhantom/RotZ      = 0. deg
s:Ge/WaterPhantom/Color      = "white"

#Value for placing a target.
#146.25 mm is the R80 for about 141 MeV proton
d:Ge/R80 = -146.25 mm

s:Ge/PlaneAtR80/Type        = "TsBox"
s:Ge/PlaneAtR80/Parent      = "WaterPhantom"
s:Ge/PlaneAtR80/Material     = "G4_WATER"
d:Ge/PlaneAtR80/HLX         = 5.0 cm
d:Ge/PlaneAtR80/HLY         = 5.0 cm
d:Ge/PlaneAtR80/HLZ         = 0.5 cm
d:Ge/PlaneAtR80/TransX      = 0. cm
d:Ge/PlaneAtR80/TransY      = 0. cm
d:Ge/PlaneAtR80/TransZ      = Ge/WaterPhantom/HLZ + Ge/R80 cm
d:Ge/PlaneAtR80/RotX        = 0.0 deg
d:Ge/PlaneAtR80/RotY        = 0.0 deg
d:Ge/PlaneAtR80/RotZ        = 0.0 deg
s:Ge/PlaneAtR80/Color       = "red"
i:Ge/PlaneAtR80/XBins       = 1
i:Ge/PlaneAtR80/YBins       = 1
i:Ge/PlaneAtR80/ZBins       = 1
```

## 21.12 TsAnalysisWaterPhantom.txt

```
# WaterPhantom used in some demonstrations.
# 40cm long, 20 cm in x, 20 cm in y

includeFile = ConstantsForNozzles.txt

# Water Tank / Phantom
s:Ge/WaterPhantom/Parent    = "Gantry"
```

(continues on next page)

(continued from previous page)

```

s:Ge/WaterPhantom/Type           = "TsBox"
s:Ge/WaterPhantom/Material       = "G4_WATER"
d:Ge/WaterPhantom/HLX           = 10.0 cm
d:Ge/WaterPhantom/HLY           = 10.0 cm
d:Ge/WaterPhantom/HLZ           = 20.0 cm
d:Ge/WaterPhantom/TransX        = 0.0 cm
d:Ge/WaterPhantom/TransY        = 0.0 cm
d:Ge/WaterPhantom/TransZ        = 0.0 cm
d:Ge/WaterPhantom/RotX          = 0. deg
d:Ge/WaterPhantom/RotY          = 0. deg
d:Ge/WaterPhantom/RotZ          = 0. deg
d:Ge/WaterPhantom/MaxStepSize   = 0.5 mm

s:Ge/ScoringCylinder3cm/Parent   = "WaterPhantom"
s:Ge/ScoringCylinder3cm/Type     = "TsCylinder"
s:Ge/ScoringCylinder3cm/Material = "G4_WATER"
d:Ge/ScoringCylinder3cm/RMin     = 0.0 cm
d:Ge/ScoringCylinder3cm/RMax     = 3.0 cm
d:Ge/ScoringCylinder3cm/HL       = Ge/WaterPhantom/HLZ cm
d:Ge/ScoringCylinder3cm/SPhi     = 0. deg
d:Ge/ScoringCylinder3cm/DPhi     = 360. deg
d:Ge/ScoringCylinder3cm/TransX   = 0. cm
d:Ge/ScoringCylinder3cm/TransY   = 0. cm
d:Ge/ScoringCylinder3cm/TransZ   = 0. cm
d:Ge/ScoringCylinder3cm/RotX     = 0. deg
d:Ge/ScoringCylinder3cm/RotY     = 0.0 deg
d:Ge/ScoringCylinder3cm/RotZ     = 180. deg
s:Ge/ScoringCylinder3cm/Color    = "blue"
i:Ge/ScoringCylinder3cm/RBins     = 1
i:Ge/ScoringCylinder3cm/PhiBins   = 1
i:Ge/ScoringCylinder3cm/ZBins     = 200

s:Sc/ScoringCylinder3cm/Quantity = "DoseToMedium"
s:Sc/ScoringCylinder3cm/Component = "ScoringCylinder3cm"
b:Sc/ScoringCylinder3cm/OutputToConsole = "False"
s:Sc/ScoringCylinder3cm/IfOutputFileAlreadyExists = "Overwrite"

```

## 21.13 Viewer.txt

```

# Graphics settings used in some demonstrations.

includeFile = ConstantsForNozzles.txt

s:Gr/ViewA/Type           = "OpenGL"
i:Gr/ViewA/WindowSizeX    = 768
i:Gr/ViewA/WindowSizeY    = 768
d:Gr/ViewA/Theta          = 25 deg
d:Gr/ViewA/Phi            = 14 deg
s:Gr/ViewA/Projection     = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 20 deg
u:Gr/ViewA/Zoom           = 10
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "False"
b:Gr/ViewA/IncludeAxes    = "True"
s:Gr/ViewA/AxesComponent  = "World"

```

(continues on next page)

(continued from previous page)

```
d:Gr/ViewA/AxesSize           = 0.1 m
#b:Gr/ViewA/CopyOpenGLToEPS    = "True"
b:Gr/ViewA/IncludeTrajectories = "True"

s:Gr/ViewA/ColorBy             = "Charge"
s:Gr/ViewA/ChargeColors        = "red green cyan"
```

## 22.1 OpticalMaterialSample.txt

```
# Materials used by Optical examples

# Materials
dv:Ma/Air/RefractiveIndex/Energies = 3 2.0 3.5 7.14 eV
uv:Ma/Air/RefractiveIndex/Values = 3 1.0 1.0 1.0

#
sv:Ma/PMMA/Components = 3 "Carbon" "Hydrogen" "Oxygen"
uv:Ma/PMMA/Fractions = 3 0.599848 0.080538 0.319614
d:Ma/PMMA/Density = 1.190 g/cm3
d:Ma/PMMA/MeanExcitationEnergy=85.7 eV
s:Ma/PMMA/DefaultColor = "lightblue"
dv:Ma/PMMA/RefractiveIndex/Energies = 2 2.0 3.35 eV
uv:Ma/PMMA/RefractiveIndex/Values = 2 1.49 1.49

#
sv:Ma/Buapfcm/Components = 2 "Hydrogen" "Carbon"
uv:Ma/Buapfcm/Fractions = 2 0.5 0.5
d:Ma/Buapfcm/Density = 1.032 g/cm3
s:Ma/Buapfcm/DefaultColor= "yellow"
u:Ma/Buapfcm/ScintillationYield = 50
u:Ma/Buapfcm/ResolutionScale = 1.0
d:Ma/Buapfcm/FastTimeConstant = 2.1 ns
u:Ma/Buapfcm/YieldRatio = 1.0
dv:Ma/Buapfcm/RefractiveIndex/Energies = 4 2.11501 2.3509 2.85803 3.10177 eV
uv:Ma/Buapfcm/RefractiveIndex/Values = 4 1.58 1.58 1.58 1.58
dv:Ma/Buapfcm/AbsLength/Energies = 4 2.11501 2.3509 2.85803 3.10177 eV
dv:Ma/Buapfcm/AbsLength/Values = 4 50 50 50 50 cm
dv:Ma/Buapfcm/FastComponent/Energies = 300
2.06785 2.07016 2.07247 2.07479 2.07711 2.07944 2.08177 2.08411 2.08646 2.
→08881
```

(continues on next page)

(continued from previous page)

```

2.09116 2.09352 2.09589 2.09826 2.10064 2.10302 2.1054 2.1078 2.11019 2.1126
↳
2.11501 2.11742 2.11984 2.12227 2.1247 2.12713 2.12958 2.13202 2.13448 2.
↳13694
2.1394 2.14187 2.14435 2.14683 2.14932 2.15181 2.15431 2.15682 2.15933 2.
↳16184
2.16437 2.1669 2.16943 2.17197 2.17451 2.17707 2.17962 2.18219 2.18476 2.
↳18734
2.18992 2.19251 2.1951 2.19771 2.20031 2.20292 2.20554 2.20817 2.2108 2.
↳21344
2.21608 2.21873 2.22139 2.22405 2.22672 2.2294 2.23208 2.23477 2.23747 2.
↳24017
2.24288 2.2456 2.24832 2.25104 2.25378 2.25652 2.25927 2.26203 2.26479 2.
↳26756
2.27033 2.27312 2.2759 2.2787 2.2815 2.28431 2.28713 2.28995 2.29278 2.29562
↳
2.29847 2.30132 2.30417 2.30704 2.30991 2.31279 2.31568 2.31858 2.32148 2.
↳32439
2.32731 2.33023 2.33316 2.3361 2.33904 2.342 2.34496 2.34793 2.3509 2.35389
2.35688 2.35987 2.36288 2.36589 2.36892 2.37195 2.37498 2.37803 2.38108 2.
↳38414
2.38721 2.39028 2.39337 2.39646 2.39956 2.40267 2.40579 2.40891 2.41204 2.
↳41519
2.41834 2.42149 2.42465 2.42783 2.43101 2.4342 2.4374 2.44061 2.44382 2.
↳44705
2.45028 2.45352 2.45677 2.46003 2.4633 2.46657 2.46986 2.47315 2.47645 2.
↳47976
2.48308 2.48641 2.48975 2.49309 2.49645 2.49981 2.50319 2.50657 2.50996 2.
↳51337
2.51677 2.52019 2.52362 2.52706 2.53051 2.53396 2.53743 2.54091 2.54439 2.
↳54788
2.55139 2.5549 2.55843 2.56196 2.56551 2.56906 2.57262 2.5762 2.57978 2.
↳58337
2.58697 2.59059 2.59421 2.59785 2.60149 2.60514 2.60881 2.61248 2.61617 2.
↳61986
2.62356 2.62728 2.63101 2.63475 2.63849 2.64225 2.64602 2.6498 2.65359 2.
↳65739
2.66121 2.66503 2.66887 2.67271 2.67657 2.68044 2.68432 2.68821 2.6921 2.
↳69602
2.69994 2.70388 2.70783 2.71179 2.71576 2.71974 2.72373 2.72774 2.73175 2.
↳73578
2.73982 2.74388 2.74794 2.75202 2.75611 2.76021 2.76433 2.76845 2.77259 2.
↳77674
2.7809 2.78508 2.78927 2.79347 2.79768 2.80191 2.80615 2.81039 2.81466 2.
↳81894
2.82323 2.82753 2.83185 2.83618 2.84052 2.84488 2.84925 2.85363 2.85803 2.
↳86244
2.86686 2.8713 2.87576 2.88022 2.8847 2.8892 2.8937 2.89822 2.90276 2.90731
2.91187 2.91645 2.92104 2.92565 2.93027 2.93491 2.93956 2.94422 2.9489 2.
↳9536
2.95831 2.96304 2.96778 2.97254 2.97731 2.98209 2.98689 2.99171 2.99654 3.
↳00139
3.00626 3.01114 3.01604 3.02095 3.02588 3.03082 3.03578 3.04075 3.04575 3.
↳05076
3.05579 3.06083 3.06589 3.07097 3.07606 3.08116 3.08629 3.09143 3.0966 3.
↳10177
eV
uv:Ma/Buapfcfm/FastComponent/Values = 300

```

(continues on next page)

(continued from previous page)

```

2.53707e-09 0.000969057 0.00219543 0.00343101 0.00448992 0.00406717 0.00212089
0.00189828 0.0034739 0.00463901 0.00349636 0.00220565 0.00175737 0.00296129
0.00380065 0.00368119 0.00409474 0.00472785 0.0048657 0.00466862 0.00530172
0.005315 0.00629325 0.00749818 0.00654444 0.00603592 0.00653117 0.00754618
0.008069 0.00776368 0.00707135 0.00696311 0.00715202 0.00650564 0.00602775
0.00659958 0.00726128 0.00799548 0.00871537 0.00873375 0.00831917 0.00788008
0.00862041 0.00989887 0.0104472 0.0106055 0.0109608 0.0116215 0.0108495
0.00959865 0.00941076 0.0104288 0.0124027 0.012899 0.0126437 0.0145103
0.0155641 0.0146206 0.0129929 0.0128979 0.0138261 0.0160961 0.0175686
0.0179913 0.0182762 0.0192045 0.0196721 0.0188471 0.0178269 0.0188705
0.0201327 0.0206014 0.0217052 0.0230357 0.0232155 0.0231389 0.0232696
0.0244664 0.0250239 0.0264045 0.0274848 0.027927 0.0291013 0.0298528
0.0303409 0.0309005 0.0306902 0.0303001 0.0314928 0.0332348 0.035756
0.0381802 0.0389613 0.0390594 0.0394403 0.0405829 0.0417868 0.0431204
0.044548 0.0462808 0.0479912 0.0492891 0.0500682 0.0519726 0.0544111
0.0558611 0.0579667 0.0595688 0.0598027 0.0606798 0.063392 0.0665656
0.0670456 0.0669813 0.0689469 0.0711097 0.072769 0.0742732 0.0746877
0.0753168 0.077166 0.0802386 0.0824984 0.0829599 0.0822237 0.0833326
0.0862817 0.0884291 0.09171 0.0954229 0.0989172 0.100978 0.103933
0.108923 0.113152 0.117175 0.120361 0.124456 0.129869 0.135552
0.141064 0.145967 0.151178 0.15674 0.164827 0.172542 0.177745
0.182116 0.187929 0.194925 0.201241 0.208795 0.216055 0.222254
0.228368 0.235244 0.242736 0.249196 0.255093 0.261023 0.26667
0.273616 0.280392 0.286072 0.290522 0.294205 0.298384 0.30374
0.30793 0.313362 0.319508 0.324279 0.328948 0.335092 0.341312
0.345841 0.350326 0.355925 0.362182 0.369145 0.377476 0.385885
0.394869 0.404849 0.415726 0.427795 0.440715 0.453566 0.468143
0.48397 0.499906 0.516324 0.535005 0.553831 0.571448 0.589444
0.607162 0.62493 0.643692 0.661074 0.678412 0.694858 0.711383
0.72796 0.742975 0.754724 0.76482 0.774458 0.784008 0.791667
0.797605 0.802749 0.804437 0.804963 0.806107 0.806766 0.804071
0.799845 0.798582 0.799343 0.79763 0.793356 0.788948 0.784289
0.782514 0.783498 0.785583 0.791562 0.799319 0.807515 0.817578
0.829744 0.843536 0.85967 0.879576 0.901532 0.921301 0.938109
0.954733 0.971245 0.985684 0.9937 0.998387 1 0.994588 0.982845
0.965506 0.943082 0.915695 0.883038 0.844113 0.798443 0.749678
0.69914 0.645699 0.592066 0.540742 0.493092 0.444413 0.397385
0.353534 0.309104 0.266364 0.229146 0.195239 0.166194 0.139671
0.115177 0.0956812 0.0815875 0.0709453 0.0619685 0.0553199 0.0496077
0.0453853 0.0420237 0.0373051 0.0330806 0.0319349 0.0313999 0.0296261
0.0282793 0.0273684 0.0262166 0.0257111 0.0254037 0.0241733 0.0225476
0.0212947 0.0200285 0.0194209 0.0188899 0.0198386 0.0202113 0.019085
0.0186091 0.0188777 0.0201643 0.0233094 0.026576

```

#

sv:Ma/Polystyrene/Components = 2 "Carbon" "Hydrogen"

uv:Ma/Polystyrene/Fractions = 2 0.9226 0.0774

d:Ma/Polystyrene/Density = 1.06 g/cm3

d:Ma/Polystyrene/MeanExcitationEnergy = 68 eV

s:Ma/Polystyrene/DefaultColor = "blue"

d:Ma/Polystyrene/WLSTimeConstant = 2.7 ns

dv:Ma/Polystyrene/RefractiveIndex/Energies = 67

```

2.08871 2.09688 2.1067 2.11901 2.13227 2.14775 2.16097 2.17225 2.18364 2.
↪19859

```

```

2.21026 2.22648 2.23891 2.25375 2.26738 2.27888 2.29049 2.30364 2.3179 2.
↪33282

```

```

2.34843 2.36028 2.37327 2.39096 2.40429 2.41311 2.42304 2.43043 2.43947 2.
↪44483

```

(continues on next page)

(continued from previous page)

```

2.45021 2.4551 2.46054 2.46818 2.47584 2.48463 2.49182 2.50069 2.50792 2.
→51687
2.52494 2.52869 2.53548 2.54 2.5476 2.55294 2.55983 2.56443 2.57218 2.57919
2.58782 2.59887 2.60524 2.61001 2.61641 2.61961 2.62441 2.6317 2.63573 2.
→64143
2.648 2.65211 2.6554 2.66204 2.66787 2.67712 2.68643 eV
uv:Ma/Polystyrene/RefractiveIndex/Values = 67
1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6
1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6
1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6
1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6
1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6
1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6
1.6 1.6 1.6 1.6 1.6 1.6 1.6
dv:Ma/Polystyrene/WLSAbsLength/Energies = 117
2.62543 2.63692 2.65308 2.66523 2.67411 2.68221 2.68692
2.69251 2.69684 2.70162 2.70685 2.7108 2.71696 2.72228
2.72455 2.72906 2.73495 2.73962 2.74511 2.74695 2.74879
2.75152 2.75336 2.75521 2.75615 2.75934 2.76207 2.76843
2.77616 2.78071 2.78573 2.79076 2.79767 2.80321 2.81018
2.81718 2.82422 2.83177 2.84031 2.84554 2.84982 2.85702
2.86575 2.87161 2.87457 2.87851 2.88297 2.88596 2.88847
2.89199 2.89452 2.89854 2.90109 2.90463 2.90768 2.91073
2.91481 2.91737 2.92147 2.92302 2.92609 2.93373 2.94037
2.94495 2.94954 2.95467 2.9598 2.96392 2.96804 2.97585
2.98159 2.98682 2.9926 2.99681 3.00102 3.00792 3.01605
3.02206 3.02972 3.03703 3.04481 3.05846 3.06638 3.07268
3.07858 3.08366 3.09129 3.0968 3.10104 3.10785 3.11469
3.12328 3.13192 3.14061 3.14846 3.15458 3.16339 3.1678
3.17489 3.18651 3.20186 3.21461 3.23025 3.24885 3.26483
3.28095 3.29435 3.30881 3.32635 3.34805 3.37005 3.40055
3.42847 3.45475 3.48037 3.50747 3.54278 eV
dv:Ma/Polystyrene/WLSAbsLength/Values =117
0.298184 0.298199 0.297833 0.297076 0.294001 0.28996
0.287072 0.283028 0.278404 0.27378 0.268193 0.261253
0.251423 0.241592 0.229636 0.211509 0.188755 0.149411
0.126657 0.1174 0.109108 0.101974 0.0950325 0.0873191
0.0805691 0.0742074 0.0701598 0.0668879 0.0690183
0.0721096 0.0771303 0.0817652 0.086595 0.0916162
0.0960602 0.10089 0.106106 0.109972 0.116346
0.121945 0.127929 0.132759 0.133154 0.130846
0.126991 0.123909 0.119284 0.113694 0.108488
0.100005 0.0924847 0.085352 0.0766746 0.0705058
0.0629863 0.0560454 0.0487198 0.0415855 0.0334883
0.0284748 0.0234628 0.0213487 0.0228984 0.0271465
0.0329377 0.0383437 0.0449069 0.0501189 0.0559096
0.061511 0.067689 0.0732878 0.0800445 0.0856423
0.0918188 0.0991551 0.107602 0.116287 0.125987
0.131491 0.135115 0.135417 0.134412 0.132248
0.130083 0.126905 0.126478 0.128363 0.131839
0.136764 0.142267 0.148062 0.155013 0.161097
0.167758 0.174997 0.181949 0.18745 0.19411
0.201643 0.209179 0.215844 0.221065 0.226577
0.23093 0.237309 0.242238 0.249772 0.25673
0.264269 0.269783 0.275014 0.279375 0.282287
0.284331 0.286086 0.287847 m
dv:Ma/Polystyrene/WLSComponent/Energies = 67

```

(continues on next page)



(continued from previous page)

```

2.08871 2.09688 2.1067 2.11901 2.13227 2.14775 2.16097 2.17225
2.18364 2.19859 2.21026 2.22648 2.23891 2.25375 2.26738 2.27888
2.29049 2.30364 2.3179 2.33282 2.34843 2.36028 2.37327 2.39096
2.40429 2.41311 2.42304 2.43043 2.43947 2.44483 2.45021 2.4551
2.46054 2.46818 2.47584 2.48463 2.49182 2.50069 2.50792 2.51687
2.52494 2.52869 2.53548 2.54 2.5476 2.55294 2.55983 2.56443 2.57218
2.57919 2.58782 2.59887 2.60524 2.61001 2.61641 2.61961 2.62441
2.6317 2.63573 2.64143 2.648 2.65211 2.6554 2.66204 2.66787 2.67712 2.68643 ↵
↪ eV
uv:Ma/Polystyrene/WLSComponent/Values = 67
0.00571321 0.00979897 0.0200973 0.0283023 0.0406511 0.0581719
0.0767485 0.092228 0.108745 0.121088 0.136567 0.165503 0.191354
0.226526 0.23681 0.252295 0.265706 0.288444 0.304952 0.323532
0.354558 0.377308 0.406278 0.43937 0.477676 0.514972 0.555375
0.604093 0.659026 0.699456 0.739886 0.791731 0.839424 0.892291
0.932709 0.963784 0.984493 0.995856 0.989592 0.972944 0.955609
0.936223 0.908521 0.873915 0.835143 0.800533 0.754848 0.718859
0.675937 0.641318 0.601159 0.541621 0.51254 0.475167 0.432254
0.404572 0.356133 0.315982 0.267547 0.224637 0.191406 0.166487
0.138805 0.106957 0.0737305 0.0404873 0.0100106

sv:Ma/Borosilicate/Components = 6 "Boron" "Oxygen" "Sodium" "Aluminum" "Silicon"
↪ "Potassium"
uv:Ma/Borosilicate/Fractions = 6 0.040064 0.539562 0.028191 0.011644 0.377220 0.
↪ 003321
d:Ma/Borosilicate/Density = 2.23 g/cm3
s:Ma/Borosilicate/DefaultColor = "red"
dv:Ma/Borosilicate/RefractiveIndex/Energies = 2 2.297 2.298 eV
uv:Ma/Borosilicate/RefractiveIndex/Values = 2 1.5 1.5

```

## 22.2 OpticalPhotonCount.txt

```

# Demonstrates optical behavior in a plastic scintillator

includeFile = OpticalMaterialSample.txt

i:Ts/SequenceVerbosity = 1

# SEQUENCE
b:Ts/PauseBeforeQuit = "True"
b:Ts/ShowCPUtime = "True"
i:Ts/ShowHistoryCountAtInterval = 1

# SCORER
s:Sc/PhSp/Quantity = "PhaseSpace"
s:Sc/PhSp/Surface = "Plastic/OuterCurvedSurface"
s:Sc/PhSp/OutputType = "ASCII"
s:Sc/PhSp/OutputFile = "PHSP"
s:Sc/PhSp/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/PhSp/IncludeTimeOfFlight = "true"
b:Sc/PhSp/UsePDGEncoding = "True"

# VISUALIZATION
s:Gr/view/Type = "OpenGL"

```

(continues on next page)

(continued from previous page)

```

i:Gr/view/WindowSizeX = 600
i:Gr/view/WindowSizeY = 600
d:Gr/view/Theta       = 45. deg
d:Gr/view/Phi         = 45. deg

# PHYSICS LIST
s:Ph/ListName          = "Optical"
s:Ph/Optical/Type      = "Geant4_Modular"
sv:Ph/Optical/Modules = 2 "g4em-standard_opt3" "g4optical"

# SOURCE
s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 100 MeV
u:So/Example/BeamEnergySpread = 0.5
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 0.5 mm
d:So/Example/BeamPositionCutoffY = 0.5 mm
s:So/Example/BeamAngularDistribution = "Flat"
d:So/Example/BeamAngularCutoffX = 0.2 rad
d:So/Example/BeamAngularCutoffY = 0.2 rad
i:So/Example/NumberOfHistoriesInRun = 100

# GEOMETRY
d:Ge/World/HLX = 1 cm
d:Ge/World/HLY = 1 cm
d:Ge/World/HLZ = 1 cm
s:Ge/World/Material = "Air"

# Plastic
s:Ge/Plastic/Parent = "World"
s:Ge/Plastic/Type = "TsCylinder"
s:Ge/Plastic/Material = "Buapfcfm"
d:Ge/Plastic/HL = 0.5 cm
d:Ge/Plastic/RMin = 0.0 cm
d:Ge/Plastic/RMax = 0.5 cm
d:Ge/Plastic/SPhi = 0 deg
d:Ge/Plastic/DPhi = 360 deg
d:Ge/Plastic/TransX = 0 mm
d:Ge/Plastic/TransY = 0 mm
d:Ge/Plastic/TransZ = 0 mm
d:Ge/Plastic/RotX = 0 deg
d:Ge/Plastic/RotY = 0 deg
d:Ge/Plastic/RotZ = 0 deg

s:Ge/PMT/Parent = "World"
s:Ge/PMT/Type = "TsBox"
s:Ge/PMT/Material = "Buapfcfm"
d:Ge/PMT/HLX = 0.5 cm
d:Ge/PMT/HLY = 0.5 cm
d:Ge/PMT/HLZ = 0.1 mm
d:Ge/PMT/TransZ1 = Ge/Plastic/TransZ - Ge/Plastic/HL cm
d:Ge/PMT/TransZ = Ge/PMT/TransZ1 - Ge/PMT/HLZ cm

s:Ge/FOT/Parent = "PMT"

```

(continues on next page)

(continued from previous page)

```

s:Ge/FOT/Type = "TsBox"
s:Ge/FOT/Material = "Buapfcfm"
d:Ge/FOT/HLX = 0.45 cm
d:Ge/FOT/HLY = 0.45 cm
d:Ge/FOT/HLZ = 0.05 mm
s:Ge/FOT/OpticalBehavior = "SurfaceDetector"

# Surface used to detect optical photons
s:Su/SurfaceDetector/Type = "dielectric_metal"
s:Su/SurfaceDetector/Finish = "polished"
s:Su/SurfaceDetector/Model = "unified"
dv:Su/SurfaceDetector/Efficiency/Energies = 2 1.6 7.2 eV
uv:Su/SurfaceDetector/Efficiency/Values = 2 1.0 1.0
dv:Su/SurfaceDetector/Reflectivity/Energies = 2 1.6 4.2 eV
uv:Su/SurfaceDetector/Reflectivity/Values = 2 0.0 0.0

# The scorer is attached to the PMT, but only those
# photons that arrives to the surface of the photocathode
# are collected (with 20% probability, see surface setup)
s:Sc/Scorer/Quantity = "OpticalPhotonCount"
s:Sc/Scorer/Component = "PMT"
s:Sc/Scorer/OutputFile = "PhotocathodeSurface"
s:Sc/Scorer/OutputType = "root"
s:Sc/Scorer/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Scorer/BounceLimit = 1000000

```

## 22.3 PlasticScintillator.txt

```

# Demonstrates optical behavior in a plastic scintillator

includeFile = OpticalMaterialSample.txt

i:Ts/SequenceVerbosity = 1

# SEQUENCE
b:Ts/PauseBeforeQuit = "True"
b:Ts/ShowCPUTime = "True"
i:Ts/ShowHistoryCountAtInterval = 1

# SCORER
s:Sc/PhSp/Quantity = "PhaseSpace"
s:Sc/PhSp/Surface = "Plastic/OuterCurvedSurface"
s:Sc/PhSp/OutputType = "ASCII"
s:Sc/PhSp/OutputFile = "PHSP"
s:Sc/PhSp/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/PhSp/IncludeTimeOfFlight = "true"
b:Sc/PhSp/UsePDGEncoding = "True"

# VISUALIZATION
s:Gr/view/Type = "OpenGL"
i:Gr/view/WindowSizeX = 600
i:Gr/view/WindowSizeY = 600
d:Gr/view/Theta = 45. deg
d:Gr/view/Phi = 45. deg

```

(continues on next page)

(continued from previous page)

```

# PHYSICS LIST
s:Ph/ListOfNames      = "Optical"
s:Ph/Optical/Type      = "Geant4_Modular"
sv:Ph/Optical/Modules = 2 "g4em-standard_opt3" "g4optical"

# SOURCE
s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 100 MeV
u:So/Example/BeamEnergySpread = 0.5
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 0.5 mm
d:So/Example/BeamPositionCutoffY = 0.5 mm
s:So/Example/BeamAngularDistribution = "Flat"
d:So/Example/BeamAngularCutoffX = 0.2 rad
d:So/Example/BeamAngularCutoffY = 0.2 rad
i:So/Example/NumberOfHistoriesInRun = 1

# GEOMETRY
d:Ge/World/HLX = 1 cm
d:Ge/World/HLY = 1 cm
d:Ge/World/HLZ = 1 cm
s:Ge/World/Material = "Air"

# Plastic
s:Ge/Plastic/Parent = "World"
s:Ge/Plastic/Type = "TsCylinder"
s:Ge/Plastic/Material = "Buapfcfm"
d:Ge/Plastic/HL = 0.5 cm
d:Ge/Plastic/RMin = 0.0 cm
d:Ge/Plastic/RMax = 0.5 cm
d:Ge/Plastic/SPhi = 0 deg
d:Ge/Plastic/DPhi = 360 deg
d:Ge/Plastic/TransX = 0 mm
d:Ge/Plastic/TransY = 0 mm
d:Ge/Plastic/TransZ = 0 mm
d:Ge/Plastic/RotX = 0 deg
d:Ge/Plastic/RotY = 0 deg
d:Ge/Plastic/RotZ = 0 deg

```

## 22.4 Rotating\_Surfaces.txt

```

# Demonstrates optical behavior on rotating surfaces

includeFile = Surfaces.txt

i:Ts/SequenceVerbosity = 0

i:Tf/Verbosity = 0
d:Tf/TimelineEnd = 60.0 ms
i:Tf/NumberOfSequentialTimes = 40

```

(continues on next page)

(continued from previous page)

```

i:So/Example/NumberOfHistoriesInRun = 20

d:Ge/Splitter/RotX = Tf/SplitterRot/Value deg

s:Tf/SplitterRot/Function      = "Linear deg"
d:Tf/SplitterRot/Rate         = 1. deg/ms
d:Tf/SplitterRot/StartValue    = -57.0 deg
d:Tf/SplitterRot/RepetitionInterval = 60. ms

d:Ge/Mirror/RotX = Tf/MirrorRot/Value deg

s:Tf/MirrorRot/Function      = "Linear deg"
d:Tf/MirrorRot/Rate         = -1. deg/ms
d:Tf/MirrorRot/StartValue    = 100.0 deg
d:Tf/MirrorRot/RepetitionInterval = 60. ms

```

## 22.5 Surfaces.txt

```

# Demonstrate use of optical surfaces

includeFile = OpticalMaterialSample.txt

i:Ts/SequenceVerbosity = 1

# SEQUENCE
b:Ts/PauseBeforeQuit      = "True"
b:Ts/ShowCPUTime          = "True"
i:Ts/ShowHistoryCountAtInterval = 10

# VISUALIZATION
s:Gr/view/Type            = "OpenGL"
#b:Gr/view/IncludeAxes    = "true"
d:Gr/view/AxesSize        = 10 cm

# PHYSICS LIST
s:Ph/ListName             = "Optical"
s:Ph/Optical/Type         = "Geant4_Modular"
sv:Ph/Optical/Modules     = 2 "g4optical" "g4em-standard_opt3"

# SCORER
s:Sc/PhSp/Quantity        = "PhaseSpace"
s:Sc/PhSp/Surface         = "Screen/YMinusSurface"
s:Sc/PhSp/OutputType      = "ASCII"
s:Sc/PhSp/OutputFile      = "PhSp"
s:Sc/PhSp/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/PhSp/UsePDGEncoding  = "True"
b:Sc/PhSp/KillAfterPhaseSpace = "True"

# SOURCE
s:So/Example/Type         = "Beam"
s:So/Example/Component    = "BeamPosition"
s:So/Example/BeamParticle = "opticalphoton"
u:So/Example/BeamPolarizationX = 1

```

(continues on next page)

(continued from previous page)

```

u:So/Example/BeamPolarizationY      = 0
u:So/Example/BeamPolarizationZ      = 0
d:So/Example/BeamEnergy              = 1.9593 eV
u:So/Example/BeamEnergySpread        = 0.04898
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX     = 0.405 mm
d:So/Example/BeamPositionCutoffY     = 0.405 mm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX      = 90. deg
d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 1. mrad
d:So/Example/BeamAngularSpreadY      = 1. mrad
i:So/Example/NumberOfHistoriesInRun  = 100
d:Ge/BeamPosition/TransZ             = -1.0 * Ge/World/HLZ cm
d:Ge/BeamPosition/RotY               = 180 deg

d:Ge/World/HLX = 15 cm
d:Ge/World/HLY = 15 cm
d:Ge/World/HLZ = 15 cm

# Splitter
s:Ge/Splitter/Parent = "World"
s:Ge/Splitter/Type   = "Group"
d:Ge/Splitter/TransX = 0 mm
d:Ge/Splitter/TransY = 0 mm
d:Ge/Splitter/TransZ = 0 mm
d:Ge/Splitter/RotX   = -45 deg
d:Ge/Splitter/RotY   = 0 deg
d:Ge/Splitter/RotZ   = 0 deg

s:Ge/Split/Parent    = "Splitter"
s:Ge/Split/Type      = "TsBox"
s:Ge/Split/Material   = "Borosilicate"
d:Ge/Split/HLX        = 2 cm
d:Ge/Split/HLY        = 2 cm
d:Ge/Split/HLZ        = 0.75 mm
d:Ge/Split/TransX     = 0 mm
d:Ge/Split/TransY     = 0 mm
d:Ge/Split/TransZ     = 0 mm
d:Ge/Split/RotX       = 0 deg
d:Ge/Split/RotY       = 0 deg
d:Ge/Split/RotZ       = 0 deg
s:Ge/Split/OpticalBehaviorTo/SilverSurface = "SurfaceSplitter"

s:Ge/SilverSurface/Parent = "Splitter"
s:Ge/SilverSurface/Type   = "TsBox"
s:Ge/SilverSurface/Material = "Borosilicate"
d:Ge/SilverSurface/HLX    = 1.9 cm
d:Ge/SilverSurface/HLY    = 1.9 cm
d:Ge/SilverSurface/HLZ    = 0.5 nm
d:Ge/SilverSurface/TransX = 0 mm
d:Ge/SilverSurface/TransY = 0 mm
d:Ge/SilverSurface/TransZ = Ge/Split/HLZ + Ge/SilverSurface/HLZ cm
d:Ge/SilverSurface/RotX   = 0 deg
d:Ge/SilverSurface/RotY   = 0 deg
d:Ge/SilverSurface/RotZ   = 0 deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/SilverSurface/OpticalBehaviorTo/Split = "SurfaceSplitter"

# Mirrors
s:Ge/Mirror/Parent = "World"
s:Ge/Mirror/Type = "TsBox"
s:Ge/Mirror/Material = "Borosilicate"
d:Ge/Mirror/HLX = 2 cm
d:Ge/Mirror/HLY = 2 cm
d:Ge/Mirror/HLZ = 0.75 mm
d:Ge/Mirror/TransX = 0 mm
d:Ge/Mirror/TransY = 10 cm
d:Ge/Mirror/TransZ = 0 mm
d:Ge/Mirror/RotX = 89.7 deg
d:Ge/Mirror/RotY = 0 deg
d:Ge/Mirror/RotZ = 0 deg
s:Ge/Mirror/OpticalBehavior = "surfaceMirror"

s:Ge/screen/Parent = "World"
s:Ge/screen/Type = "TsBox"
s:Ge/screen/Material = "Air"
d:Ge/screen/HLX = 2 cm
d:Ge/screen/HLY = 1 mm
d:Ge/screen/HLZ = 2 cm
d:Ge/screen/TransX = 0 mm
d:Ge/screen/TransY = -10 cm
d:Ge/screen/TransZ = 0 cm
d:Ge/screen/RotX = 0 deg
d:Ge/screen/RotY = 0 deg
d:Ge/screen/RotZ = 0 deg

# Optical properties of the surfaces needed
s:Su/SurfaceSplitter/Type = "dielectric_dielectric"
s:Su/SurfaceSplitter/Finish = "polished"
s:Su/SurfaceSplitter/Model = "unified"
dv:Su/SurfaceSplitter/Efficiency/Energies = 2 1.6 4.2 eV
uv:Su/SurfaceSplitter/Efficiency/Values = 2 1.0 1.0
dv:Su/SurfaceSplitter/Reflectivity/Energies = 2 1.6 4.2 eV
uv:Su/SurfaceSplitter/Reflectivity/Values = 2 0.5 0.5

s:Su/SurfaceMirror/Type = "dielectric_metal"
s:Su/SurfaceMirror/Finish = "polished"
s:Su/SurfaceMirror/Model = "unified"
dv:Su/SurfaceMirror/Reflectivity/Energies = 2 1.6 4.2 eV
uv:Su/SurfaceMirror/Reflectivity/Values = 2 1.0 1.0

```

## 22.6 WavelengthShifter.txt

```

# Demonstrates optical behavior in a wavelength shifter

includeFile = OpticalMaterialSample.txt

i:Ts/SequenceVerbosity = 0

# SEQUENCE

```

(continues on next page)

(continued from previous page)

```

b:Ts/PauseBeforeQuit = "False"
b:Ts/ShowCPUTime = "True"
i:Ts/ShowHistoryCountAtInterval= 10000

# SCORER
s:Sc/PhSp1/Quantity = "PhaseSpace"
s:Sc/PhSp1/Surface = "WLS/OuterCurvedSurface"
s:Sc/PhSp1/OutputType = "ASCII"
s:Sc/PhSp1/OutputFile = "In"
s:Sc/PhSp1/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/PhSp1/IncludeTimeOfFlight = "true"
b:Sc/PhSp1/UsePDGEncoding = "True"
sv:Sc/PhSp1/OnlyIncludeParticlesFromVolume = 1 "Plastic"
s:Sc/PhSp1/OnlyIncludeParticlesGoing = "in"

s:Sc/PhSp2/Quantity = "PhaseSpace"
s:Sc/PhSp2/Surface = "WLS/Core/ZPlusSurface"
s:Sc/PhSp2/OutputType = "ASCII"
s:Sc/PhSp2/OutputFile = "Out"
s:Sc/PhSp2/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/PhSp2/IncludeTimeOfFlight = "true"
b:Sc/PhSp2/UsePDGEncoding = "True"

# VISUALIZATION
#s:Gr/view/Type = "OpenGL"
b:Gr/view/IncludeAxes = "True"
i:Gr/view/WindowSizeX = 600
i:Gr/view/WindowSizeY = 600
d:Gr/view/Theta = 45. deg
d:Gr/view/Phi = 45. deg

# PHYSICS LIST
s:Ph/ListName = "Optical"
s:Ph/Optical/Type = "Geant4_Modular"
sv:Ph/Optical/Modules = 7 "g4optical" "g4em-standard_opt3" "g4h-phy_QGSP_BIC_HP"
↪ "g4decay"
" g4ion-binarycascade" "g4h-elastic_HP" "g4stopping"

# SOURCE
s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 200 MeV
u:So/Example/BeamEnergySpread = 0.5
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 0.5 mm
d:So/Example/BeamPositionCutoffY = 0.5 mm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.2 rad
d:So/Example/BeamAngularSpreadY = 0.2 rad
i:So/Example/NumberOfHistoriesInRun = 100000
d:Ge/BeamPosition/TransY = -3.0 cm

# GEOMETRY

```

(continues on next page)



(continued from previous page)

```

d:Ge/World/HLX      =6 cm
d:Ge/World/HLY      =6 cm
d:Ge/World/HLZ      =20 cm
s:Ge/World/Material = "Air"

# Plastic
s:Ge/WaterBox/Parent = "World"
s:Ge/WaterBox/Type   = "TsBox"
s:Ge/WaterBox/Material = "G4_WATER"
d:Ge/WaterBox/HLX    = 2.5 cm
d:Ge/WaterBox/HLY    = 2.5 cm
d:Ge/WaterBox/HLZ    = 5.0 cm
d:Ge/WaterBox/TransX = 0 mm
d:Ge/WaterBox/TransY = -3.0 cm
d:Ge/WaterBox/TransZ = 0 mm
d:Ge/WaterBox/RotX   = 0 deg
d:Ge/WaterBox/RotY   = 0 deg
d:Ge/WaterBox/RotZ   = 0 deg

# Plastic
s:Ge/Plastic/Parent = "World"
s:Ge/Plastic/Type   = "TsBox"
s:Ge/Plastic/Material = "BUAPFCFM"
d:Ge/Plastic/HLX    = 0.5 cm
d:Ge/Plastic/HLY    = 0.5 cm
d:Ge/Plastic/HLZ    = 2.5 cm
d:Ge/Plastic/TransX = 0 mm
d:Ge/Plastic/TransY = 0 mm
d:Ge/Plastic/TransZ = 0 mm
d:Ge/Plastic/RotX   = 0 deg
d:Ge/Plastic/RotY   = 0 deg
d:Ge/Plastic/RotZ   = 0 deg

# WLS
d:Ge/WLS/Diameter = 1.03 mm
s:Ge/WLS/Parent   = "World"
s:Ge/WLS/Type     = "TsCylinder"
s:Ge/WLS/Material = "PMMA"
d:Ge/WLS/HL       = 10 cm
d:Ge/WLS/RMin     = 0.50 mm
d:Ge/WLS/RMax     = 0.515 mm
d:Ge/WLS/SPhi     = 0 deg
d:Ge/WLS/DPhi     = 360 deg
d:Ge/WLS/TransX   = 0 mm
d:Ge/WLS/TransY   = Ge/Plastic/HLY + Ge/WLS/RMax mm
d:Ge/WLS/TransZ   = Ge/WLS/HL - Ge/Plastic/HLZ cm
d:Ge/WLS/RotX     = 0 deg
d:Ge/WLS/RotY     = 0 deg
d:Ge/WLS/RotZ     = 0 deg

s:Ge/WLS/Core/Parent = "World"
s:Ge/WLS/Core/Type   = "TsCylinder"
s:Ge/WLS/Core/Material = "Polystyrene"
d:Ge/WLS/Core/HL     = Ge/WLS/HL cm
d:Ge/WLS/Core/RMin   = 0.0 mm
d:Ge/WLS/Core/RMax   = 0.5 mm
d:Ge/WLS/Core/SPhi   = 0 deg

```

(continues on next page)

(continued from previous page)

```
d:Ge/WLS/Core/DPhi      = 360 deg
d:Ge/WLS/Core/TransX    = 0 mm
d:Ge/WLS/Core/TransY    = Ge/Plastic/HLY + Ge/WLS/RMax mm
d:Ge/WLS/Core/TransZ    = Ge/WLS/HL - Ge/Plastic/HLZ cm
d:Ge/WLS/Core/RotX      = 0 deg
d:Ge/WLS/Core/RotY      = 0 deg
d:Ge/WLS/Core/RotZ      = 0 deg
```

## 23.1 CEModelBurman.txt

## 23.2 CVModelStavrev.txt

```
#Parameter obtained from Stavrev et. al. Physica Medica, Vol 17(2) (2001).
# Table 1.
#          Organ
#FSU std      Gamma50_FSU std      mu_cr std      sigma std      D50_
uv:Sc/CriticalVolumeValues/Lung      = 8      0.08      0.02      0.
  02      0.01      220      51      0.3      0.02      # Pneumonitis
uv:Sc/CriticalVolumeValues/Liver      = 8      0.04      0.01      0.
  01      0.003      181      20      0.5      0.02      # Liver failure
uv:Sc/CriticalVolumeValues/Brain      = 8      0.08      0.02      0.
  05      0.02      104      7.9      1      0.1      # Necrosis
uv:Sc/CriticalVolumeValues/Heart      = 8      0.14      0.07      0.
  05      0.02      83      22      0.8      0.2      # Pericarditis
uv:Sc/CriticalVolumeValues/Kidney      = 8      0.22      0.06      0.
  05      0.01      59      14      0.4      0.06      # Clinical_
  nephritis
uv:Sc/CriticalVolumeValues/Esophagus      = 8      0.01      0.02      0.
  02      0.03      97      14      2.4      0.8      # Clinicial_
  stricture, perforation
uv:Sc/CriticalVolumeValues/Stomach      = 8      0.02      0.01      0.
  03      0.01      94      7.1      2.2      0.5      # Ulceration,_
  perforation
uv:Sc/CriticalVolumeValues/BrachialPlexus = 8      0.01      0.03      0.
  05      0.1      86      9.9      6      4      # Clinically apparent_
  nerve damage
uv:Sc/CriticalVolumeValues/Bladder      = 8      0.26      0.11      0.
  11      0.02      108      24      0.8      0.2      # Symptomatic_
  bladder contraction and volume loss
uv:Sc/CriticalVolumeValues/TMJointMandible= 8      0.07      0.06      0.
  09      0.07      86      7.4      3.1      1      # Marked
  of joint function
```

(continues on next page)

(continued from previous page)

```

uv:Sc/CriticalVolumeValues/BrainStem      = 8      0.12      0.06      0.
→08      0.04      97      12      1.2      0.2      # Necrosis
uv:Sc/CriticalVolumeValues/Larynx          = 8      0.07      0.08      0.
→08      0.05      108      22      1.9      0.5      # Cartilage,
→necrosis perforation
uv:Sc/CriticalVolumeValues/SmallIntestine = 8      0.04      0.05      0.
→04      0.04      97      28      1.2      0.3      # Obstruction,
→perforation
uv:Sc/CriticalVolumeValues/Colon           = 8      0.03      0.02      0.
→02      0.01      108      16      1.1      0.1      # Obstruction,
→perforation
uv:Sc/CriticalVolumeValues/SpinalCord      = 8      0.08      0.09      0.
→16      0.15      81      12      2.6      1      # Myelitis necrosis
uv:Sc/CriticalVolumeValues/Skin            = 8      0.11      0.05      0.
→09      0.05      93      8.5      1.7      0.3      # Necrosis,
→ulceration

```

## 23.3 Phantom.txt

```

# The body
s:Ge/WaterPhantom/Parent = "World"
s:Ge/WaterPhantom/Type   = "TsBox"
s:Ge/WaterPhantom/Material = "G4_WATER"
d:Ge/WaterPhantom/HLX = 40 mm
d:Ge/WaterPhantom/HLY = 40 mm
d:Ge/WaterPhantom/HLZ = 100 mm
d:Ge/WaterPhantom/TransX = 0.0 cm
d:Ge/WaterPhantom/TransY = 0.0 cm
d:Ge/WaterPhantom/TransZ = 0.0 cm
d:Ge/WaterPhantom/RotX= 0. deg
d:Ge/WaterPhantom/RotY= 0. deg
d:Ge/WaterPhantom/RotZ= 0. deg

# The target/tumor
d:Ge/InitPosZ = -1.0 * Ge/WaterPhantom/HLZ mm
s:Ge/WaterPhantom/Target/Parent = "WaterPhantom"
s:Ge/WaterPhantom/Target/Type   = "TsBox"
s:Ge/WaterPhantom/Target/Material = "G4_WATER"
d:Ge/WaterPhantom/Target/HLX = 10 mm
d:Ge/WaterPhantom/Target/HLY = 10 mm
d:Ge/WaterPhantom/Target/HLZ = 22 mm
d:Ge/WaterPhantom/Target/TransX = 0.0 cm
d:Ge/WaterPhantom/Target/TransY = 0.0 cm
d:Ge/WaterPhantom/Target/TransZTmp = 24 mm + Ge/InitPosZ
d:Ge/WaterPhantom/Target/TransZ = Ge/WaterPhantom/Target/TransZTmp + Ge/WaterPhantom/
→Target/HLZ mm
d:Ge/WaterPhantom/Target/RotX= 0. deg
d:Ge/WaterPhantom/Target/RotY= 0. deg
d:Ge/WaterPhantom/Target/RotZ= 0. deg
i:Ge/WaterPhantom/Target/XBins = 20
i:Ge/WaterPhantom/Target/YBins = 20
i:Ge/WaterPhantom/Target/ZBins = 44

# An organ at risk at the entrance of the beam

```

(continues on next page)

(continued from previous page)

```

s:Ge/WaterPhantom/OAR1/Parent = "WaterPhantom"
s:Ge/WaterPhantom/OAR1/Type   = "TsBox"
s:Ge/WaterPhantom/OAR1/Material = "G4_WATER"
d:Ge/WaterPhantom/OAR1/HLX = 10 mm
d:Ge/WaterPhantom/OAR1/HLY = 10 mm
d:Ge/WaterPhantom/OAR1/HLZ = 22 mm
d:Ge/WaterPhantom/OAR1/TransX = 0.0 cm
d:Ge/WaterPhantom/OAR1/TransY = 0.0 cm
d:Ge/WaterPhantom/OAR1/TransZTmp = Ge/WaterPhantom/Target/TransZ + Ge/WaterPhantom/
↳Target/HLZ mm
d:Ge/WaterPhantom/OAR1/TransZ = Ge/WaterPhantom/OAR1/TransZTmp + Ge/WaterPhantom/OAR1/
↳HLZ mm
d:Ge/WaterPhantom/OAR1/RotX= 0. deg
d:Ge/WaterPhantom/OAR1/RotY= 0. deg
d:Ge/WaterPhantom/OAR1/RotZ= 0. deg
i:Ge/WaterPhantom/OAR1/XBins = 20
i:Ge/WaterPhantom/OAR1/YBins = 20
i:Ge/WaterPhantom/OAR1/ZBins = 44

# An organ at risk at distal position
s:Ge/WaterPhantom/OAR2/Parent = "WaterPhantom"
s:Ge/WaterPhantom/OAR2/Type   = "TsBox"
s:Ge/WaterPhantom/OAR2/Material = "G4_WATER"
d:Ge/WaterPhantom/OAR2/HLX = 12 mm
d:Ge/WaterPhantom/OAR2/HLY = 12 mm
d:Ge/WaterPhantom/OAR2/HLZ = 12 mm
d:Ge/WaterPhantom/OAR2/TransX = 0.0 cm
d:Ge/WaterPhantom/OAR2/TransY = 0.0 cm
d:Ge/WaterPhantom/OAR2/TransZ = Ge/InitPosZ + Ge/WaterPhantom/OAR2/HLZ mm
d:Ge/WaterPhantom/OAR2/RotX= 0. deg
d:Ge/WaterPhantom/OAR2/RotY= 0. deg
d:Ge/WaterPhantom/OAR2/RotZ= 0. deg
i:Ge/WaterPhantom/OAR2/XBins = 24
i:Ge/WaterPhantom/OAR2/YBins = 24
i:Ge/WaterPhantom/OAR2/ZBins = 24

```

## 23.4 PoissonModelOkuneff.txt

```

#Parameter obtained from Okunieff et. al. Int. J. Radiation Oncology Biol. Phys. Vol.
↳32 pp 1227-1237. (1995)
# Table No. 2. Single institution macroscopic tumor
#      Tumor site      TD50      Gamma50      Slope50      Comments
uv:Sc/PoissonValues/MFungoides      = 3      10.49      0.33      3.
↳1
uv:Sc/PoissonValues/MFungoides      = 3      10.53      0.31      2.
↳97      # Recurrent
uv:Sc/PoissonValues/Hodgkin      = 3      14.65      0.41      2.
↳8      # Stage I + II
uv:Sc/PoissonValues/Hodgkin      = 3      15.3      1.39      9.
↳09      # > 3.0 cm
uv:Sc/PoissonValues/Hodgkin      = 3      15.31      1.51      9.
↳84      # 0.5-3.0 cm
uv:Sc/PoissonValues/Hodgkin      = 3      21.42      1.62      7.
↳58      # All

```

(continues on next page)

(continued from previous page)

```

uv:Sc/PoissonValues/Breast          = 3      21.71      0.04      0.
↳17      # 4-6 cm
uv:Sc/PoissonValues/UterineCervix   = 3      24.27      0.67      2.77
↳# Stage IA
uv:Sc/PoissonValues/UterineCervix   = 3      25.51      0.64      2.52
↳# Stage IIA
uv:Sc/PoissonValues/UterineCervix   = 3      26.35      0.71      2.7      #L
↳All
uv:Sc/PoissonValues/UterineCervix   = 3      34.15      20.93      61.
↳27      # Stage IIB
uv:Sc/PoissonValues/Breast          = 3      36.24      0.88      2.
↳42      # Recurrent
uv:Sc/PoissonValues/UterineCervix   = 3      45.45      0.69      1.52
↳# Stage IIB
uv:Sc/PoissonValues/Breast          = 3      46.94      0.28      0.6
↳# All T3 + 4
uv:Sc/PoissonValues/NSCLC          = 3      48.51      0.57      1.18
↳# CR + PR
uv:Sc/PoissonValues/Lung            = 3      49.12      1.25      2.54
↳# Large Cell
uv:Sc/PoissonValues/Melanoma        = 3      49.84      0.99      1.
↳98      # --
uv:Sc/PoissonValues/Nasopharynx     = 3      50.01      17.54      35.
↳06      # Node < 3 cm
uv:Sc/PoissonValues/Nasopharynx     = 3      50.27      32.22      64.
↳1      # All Node +
uv:Sc/PoissonValues/Supraglottic    = 3      50.44      1.83      3.62
↳# T1
uv:Sc/PoissonValues/Lung            = 3      51.87      2.17      4.18
↳# Adenocarcinoma
uv:Sc/PoissonValues/Lung            = 3      51.97      1.81      3.48
↳# All NSCLC
uv:Sc/PoissonValues/Tonsil          = 3      54.02      1.03      1.
↳92      # N1-3
uv:Sc/PoissonValues/Lung            = 3      54.92      2.04      3.72
↳# Squamous Cell
uv:Sc/PoissonValues/Nasopharynx     = 3      55.12      1.82      3.31
↳# Node 3-6 cm
uv:Sc/PoissonValues/Tonsil          = 3      57.27      1.29      3.
↳25      # T1-4
uv:Sc/PoissonValues/Breast          = 3      57.37      0.93      1.
↳61      # Stage III
uv:Sc/PoissonValues/Nasopharynx     = 3      58.79      47.26      80.
↳39      # T1 +T2
uv:Sc/PoissonValues/Breast          = 3      59.08      1.33      2.
↳25      # Stage III
uv:Sc/PoissonValues/Nasopharynx     = 3      60.15      4.41      7.33
↳# T3
uv:Sc/PoissonValues/PyriiformSinus  = 3      60.76      3.8       6.25      #L
↳Nodes<3cm
uv:Sc/PoissonValues/Nasopharynx     = 3      61.59      3.38      5.49
↳# T1-4
uv:Sc/PoissonValues/Supraglottic    = 3      61.97      2.69      4.33
↳# Squamous Cell
uv:Sc/PoissonValues/Breast          = 3      62.59      1.24      1.
↳98      # > 6 cm
uv:Sc/PoissonValues/Breast          = 3      62.83      1.55      2.
↳46      # Stage II + III

```

(continues on next page)

(continued from previous page)

```

uv:Sc/PoissonValues/Supraglottic = 3      62.92      0.63      1      # T4
uv:Sc/PoissonValues/Supraglottic = 3      63.28      4.01      6.33
↳# T2+T3
uv:Sc/PoissonValues/PyriformSinus = 3      63.4      2.85      4.49      #
↳All Nodes
uv:Sc/PoissonValues/Supraglottic = 3      63.43      2.66      4.19
↳# Stage I, II, IIIA
uv:Sc/PoissonValues/PharyngealWall = 3      63.66      1.67      2.62
↳# T1-4
uv:Sc/PoissonValues/UterineCervix = 3      64.38      0.69      1.08
↳# Stage III
uv:Sc/PoissonValues/PyriformSinus = 3      65.89      1.61      2.45
↳# T1+2
uv:Sc/PoissonValues/Nasopharynx = 3      67.23      3.25      4.84
↳# T4
uv:Sc/PoissonValues/Nasopharynx = 3      68.81      1.67      2.43
↳# Node>6cm
uv:Sc/PoissonValues/PyriformSinus = 3      69.72      2.81      4.03
↳# Nodes>3cm
uv:Sc/PoissonValues/PharyngealWall = 3      72.19      0.8      1.11      #
↳T3 + 4
uv:Sc/PoissonValues/PyriformSinus = 3      77.15      0.75      0.97
↳# T1+2+3
uv:Sc/PoissonValues/PyriformSinus = 3      81.75      0.69      0.85
↳# T3
uv:Sc/PoissonValues/Supraglottic = 3      83.35      0.97      1.16
↳# Stage IIIB, IV
uv:Sc/PoissonValues/Breast = 3      90.31      0.41      0.
↳45      # T4

```

## 23.5 TestOutcomeModel.txt

```

# The geometry
includeFile = LKMMModelBurman.txt Phantom.txt

Ge/World/HLX = 101 mm
Ge/World/HLY = 101 mm
Ge/World/HLZ = 101 mm

# To reach ~60 Gy at maximum dose in the target
u:Sc/ScaleFactor = 5779493.074240799

# Calculates outcome from DVH
s:Sc/OAR1/Quantity = "DoseToMedium"
s:Sc/OAR1/Component = "WaterPhantom/OAR1"
s:Sc/OAR1/OutputFile = "DoseAtOAR1"
s:Sc/OAR1/OutputType = "Binary"
s:Sc/OAR1/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OAR1/Report = 2 "sum" "cumulativevolumehistogram"
i:Sc/OAR1/HistogramBins = 100
d:Sc/OAR1/HistogramMin = 0 Gy
d:Sc/OAR1/HistogramMax = 1.42E-5 Gy

# Here cDVH is activated, then the outcome will be calculate from DVH.

```

(continues on next page)

(continued from previous page)

```

# For that, cDVH is internally converted to dDVH
# Models to be calculated
sv:Sc/OAR1/OutcomeModelName = 3 "lkb" "CriticalElement" "CriticalVolume"
# User may wants to scale the dose distribution
u:Sc/OAR1/OutcomeOutputScaleFactor = Sc/ScaleFactor
# Correct for dose fractionation. Set number of fractions:
i:Sc/OAR1/NumberOfFractions = 20
# Optionally set the dose per fraction and alpha/beta. Otherwise
# The values 2.0 Gy and 3.0 Gy are set by default respectively
d:Sc/OAR1/DosePerFraction = 2.0 Gy
d:Sc/OAR1/AlphaOverBeta = 3.0 Gy

# There is an internal data base for LKB only.
# If the organ name is not found in data base, TOPAS is stopped and
# a list with the organs is displayed. Otherwise user can set
# the parameters by hand: n, m and TD50
s:Sc/OAR1/lkb/UsePresetParametersFromOrganNamed = "Kidney"

# Parameters for the rest of the models are set by hand
u:Sc/OAR1/CriticalElement/Gamma = 2.8
u:Sc/OAR1/CriticalElement/td50 = 28.0
u:Sc/OAR1/CriticalVolume/mu_cr = 0.22
u:Sc/OAR1/CriticalVolume/sigma_mu_cr = 0.05
u:Sc/OAR1/CriticalVolume/d50fsu = 59
u:Sc/OAR1/CriticalVolume/gamma50fsu = 0.4

# From full dose distribution
s:Sc/OAR2/Quantity = "DoseToMedium"
s:Sc/OAR2/Component = "WaterPhantom/OAR2"
s:Sc/OAR2/OutputFile = "DoseAtOAR2"
s:Sc/OAR2/OutputType = "Binary"
s:Sc/OAR2/IfOutputFileAlreadyExists = "Overwrite"

# If DVH is not reported, then the outcome is calculated
# from full dose distribution. In that case, all voxels
# must to have the same volume.
sv:Sc/OAR2/OutcomeModelName = 1 "lkb"
u:Sc/OAR2/OutcomeOutputScaleFactor = Sc/ScaleFactor
s:Sc/OAR2/lkb/UsePresetParametersFromOrganNamed = "colon"
# Correct for dose fractionation
i:Sc/OAR2/NumberOfFractions = 20

s:Sc/Target/Quantity = "DoseToMedium"
s:Sc/Target/Component = "WaterPhantom/Target"
s:Sc/Target/OutputFile = "DoseAtTarget"
s:Sc/Target/OutputType = "Binary"
s:Sc/Target/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/Target/Report = 2 "sum" "cumulativevolumehistogram"
i:Sc/Target/HistogramBins = 100
d:Sc/Target/HistogramMin = 0 Gy
d:Sc/Target/HistogramMax = 1.56e-5 Gy
sv:Sc/Target/OutcomeModelName = 1 "poisson"
u:Sc/Target/OutcomeOutputScaleFactor = Sc/ScaleFactor
u:Sc/Target/Poisson/Gamma50 = 0.60
u:Sc/Target/Poisson/TCD50 = 41.78
# Correct for dose fractionation
i:Sc/Target/NumberOfFractions = 20

```

(continues on next page)



(continued from previous page)

```

# Set a raw SOBP
s:Tf/Energy/Function = "Step"
dv:Tf/Energy/Times = 5 20 40 60 80 100 ms
dv:Tf/Energy/Values = 5 160 155 150 145 140 MeV
s:Tf/Histories/Function = "Step"
dv:Tf/Histories/Times = 5 20 40 60 80 100 ms
iv:Tf/Histories/Values = 5 6600 2500 1750 1400 1200

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = Tf/Energy/Value MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. mm
d:So/Example/BeamPositionCutoffY = 10. mm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 10 * Tf/Histories/Value

# Start
Ts/PauseBeforeQuit = "False"
i:Tf/NumberOfSequentialTimes = 5
d:Tf/TimelineEnd = 100 ms
Ts/ShowHistoryCountAtInterval = 10000
Tf/Verbosity = 1
b:Ts/ShowCPUTime = "True"

```

## 23.6 TestRestoreModel.txt

```

# The geometry
includeFile = LKMMModelBurman.txt Phantom.txt

Ge/World/HLX = 101 mm
Ge/World/HLY = 101 mm
Ge/World/HLZ = 101 mm

# To reach ~60 Gy at maximum dose in the target
u:Sc/ScaleFactor = 5779493.074240799

# Calculates outcome from DVH
s:Sc/OAR1/Quantity = "DoseToMedium"
s:Sc/OAR1/Component = "WaterPhantom/OAR1"
s:Sc/OAR1/OutputFile = "DoseAtOAR1"
s:Sc/OAR1/OutputType = "Binary"
s:Sc/OAR1/InputFile = "DoseAtOAR1Saved"
s:Sc/OAR1/InputType = "Binary"
s:Sc/OAR1/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OAR1/Report = 2 "sum" "cumulativevolumehistogram"
i:Sc/OAR1/HistogramBins = 100
d:Sc/OAR1/HistogramMin = 0 Gy
d:Sc/OAR1/HistogramMax = 1.42E-5 Gy

```

(continues on next page)

(continued from previous page)

```

# Here cDVH is activated, then the outcome will be calculate from DVH.
# For that, cDVH is internally converted to dDVH
# Models to be calculated
sv:Sc/OAR1/OutcomeModelName = 3 "lkb" "CriticalElement" "CriticalVolume"
# User may wants to scale the dose distribution
u:Sc/OAR1/OutcomeOutputScaleFactor = Sc/ScaleFactor
# Correct for dose fractionation. Set number of fractions:
i:Sc/OAR1/NumberOfFractions = 20
# Optionally set the dose per fraction and alpha/beta. Otherwise
# The values 2.0 Gy and 3.0 Gy are set by default respectively
d:Sc/OAR1/DosePerFraction = 2.0 Gy
d:Sc/OAR1/AlphaOverBeta = 3.0 Gy

# There is an internal data base for LKB only.
# If the organ name is not found in data base, TOPAS is stopped and
# a list with the organs is displayed. Otherwise user can set
# the parameters by hand: n, m and TD50
s:Sc/OAR1/lkb/UsePresetParametersFromOrganNamed = "Kidney"

# Parameters for the rest of the models are set by hand
u:Sc/OAR1/CriticalElement/Gamma = 2.8
u:Sc/OAR1/CriticalElement/td50 = 28.0
u:Sc/OAR1/CriticalVolume/mu_cr = 0.22
u:Sc/OAR1/CriticalVolume/sigma_mu_cr = 0.05
u:Sc/OAR1/CriticalVolume/d50fsu = 59
u:Sc/OAR1/CriticalVolume/gamma50fsu = 0.4

# From full dose distribution
s:Sc/OAR2/Quantity = "DoseToMedium"
s:Sc/OAR2/Component = "WaterPhantom/OAR2"
s:Sc/OAR2/OutputFile = "DoseAtOAR2"
s:Sc/OAR2/OutputType = "Binary"
s:Sc/OAR2/InputFile = "DoseAtOAR2Saved"
s:Sc/OAR2/InputType = "Binary"
s:Sc/OAR2/IfOutputFileAlreadyExists = "Overwrite"

# If DVH is not reported, then the outcome is calculated
# from full dose distribution. In that case, all voxels
# must to have the same volume.
sv:Sc/OAR2/OutcomeModelName = 1 "lkb"
u:Sc/OAR2/OutcomeOutputScaleFactor = Sc/ScaleFactor
s:Sc/OAR2/lkb/UsePresetParametersFromOrganNamed = "colon"
# Correct for dose fractionation
i:Sc/OAR2/NumberOfFractions = 20

s:Sc/Target/Quantity = "DoseToMedium"
s:Sc/Target/Component = "WaterPhantom/Target"
s:Sc/Target/OutputFile = "DoseAtTarget"
s:Sc/Target/OutputType = "Binary"
s:Sc/Target/InputFile = "DoseAtTargetSaved"
s:Sc/Target/InputType = "Binary"
s:Sc/Target/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/Target/Report = 2 "sum" "cumulativevolumehistogram"
i:Sc/Target/HistogramBins = 100
d:Sc/Target/HistogramMin = 0 Gy
d:Sc/Target/HistogramMax = 1.56e-5 Gy
sv:Sc/Target/OutcomeModelName = 1 "poisson"

```

(continues on next page)

(continued from previous page)

```

u:Sc/Target/OutcomeOutputScaleFactor = Sc/ScaleFactor
u:Sc/Target/Poisson/Gamma50 = 0.40
u:Sc/Target/Poisson/TCD50 = 11.78
# Correct for dose fractionation
i:Sc/Target/NumberOfFractions = 20

# Set a raw SOBP
s:Tf/Energy/Function = "Step"
dv:Tf/Energy/Times = 5 20 40 60 80 100 ms
dv:Tf/Energy/Values = 5 160 155 150 145 140 MeV
s:Tf/Histories/Function = "Step"
dv:Tf/Histories/Times = 5 20 40 60 80 100 ms
iv:Tf/Histories/Values = 5 6600 2500 1750 1400 1200

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = Tf/Energy/Value MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10 mm
d:So/Example/BeamPositionCutoffY = 10 mm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 10 * Tf/Histories/Value

# Start
Ts/PauseBeforeQuit = "False"
i:Tf/NumberOfSequentialTimes = 5
d:Tf/TimelineEnd = 100 ms
Ts/ShowHistoryCountAtInterval = 10000
Tf/Verbosity = 1
b:Ts/ShowCPUTime = "True"

Ts/RestoreResultsFromFile = "True"

```



## 24.1 Applicator.txt

```
# Set world material to lead so that test particles
# stop right away, making it easy to just see all the
# starting positions of the particles.
```

```
Ge/World/HLX = 50. cm
Ge/World/HLY = 50. cm
Ge/World/HLZ = 50. cm
Ge/World/Material = "G4_WATER"
b:Ge/World/Invisible = "True"
```

```
# If using water phantom
s:Ge/Patient/Type = "TsBox"
s:Ge/Patient/Parent = "World"
s:Ge/Patient/Material = "G4_WATER"
d:Ge/Patient/HLX = 4. cm
d:Ge/Patient/HLY = 4. cm
d:Ge/Patient/HLZ = 4. cm
#i:Ge/Patient/XBins = 10
#i:Ge/Patient/YBins = 10
#i:Ge/Patient/ZBins = 5
```

```
# If using real DICOM
#s:Ge/Patient/Type = "TsDicomPatient"
#s:Ge/Patient/Parent = "World"
d:Ge/Patient/TransX = 0. cm
d:Ge/Patient/TransY = 0. cm
d:Ge/Patient/TransZ = 0. cm
d:Ge/Patient/RotX = 0. deg
d:Ge/Patient/RotY = 0. deg
d:Ge/Patient/RotZ = 0. deg
s:Ge/Patient/DicomDirectory = "Abdomen"
```

(continues on next page)

(continued from previous page)

```

sv:Ge/Patient/DicomModalityTags = 1 "CT"
includeFile = HUtoMaterialSchneider.txt

# Scoring
s:Sc/Dose/Quantity = "DoseToMedium"
s:Sc/Dose/Component = "Patient"
s:Sc/Dose/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/Dose/OutputType = "DICOM"
b:Sc/Dose/DICOMOutput32BitsPerPixel = "True"

# Applicator Geometry
s:Ge/Applicator/Type = "TsBrachyApplicator"
s:Ge/Applicator/Parent = "Patient"
b:Ge/Applicator/IsParallel = "True"
s:Ge/Applicator/Material = "G4_WATER"
d:Ge/Applicator/CylinderLength = 40. mm
d:Ge/Applicator/Radius = 12.5 mm
i:Ge/Applicator/NumberOfRadialHoles = 6
d:Ge/Applicator/HoleOffset = 6 mm
d:Ge/Applicator/HoleRadius = 1.2 mm
s:Ge/Applicator/DrawingStyle = "Solid"
s:Ge/Applicator/Color = "transparentgrey"
iv:Gr/Color/transparentgrey = 4 255 255 255 90

# The following parameters are updated automatically
# by the applicator component to show true hole centers.
# They need to be defined here, but these initial values
# are not important
dc:Ge/Applicator/Hole0/TransX = 0. mm
dc:Ge/Applicator/Hole0/TransY = 0. mm
dc:Ge/Applicator/Hole1/TransX = 0. mm
dc:Ge/Applicator/Hole1/TransY = 0. mm
dc:Ge/Applicator/Hole2/TransX = 0. mm
dc:Ge/Applicator/Hole2/TransY = 0. mm
dc:Ge/Applicator/Hole3/TransX = 0. mm
dc:Ge/Applicator/Hole3/TransY = 0. mm
dc:Ge/Applicator/Hole4/TransX = 0. mm
dc:Ge/Applicator/Hole4/TransY = 0. mm
dc:Ge/Applicator/Hole5/TransX = 0. mm
dc:Ge/Applicator/Hole5/TransY = 0. mm
dc:Ge/Applicator/Hole6/TransX = 0. mm
dc:Ge/Applicator/Hole6/TransY = 0. mm

# Varian Source wire model VS2000 Geometry
s:Ge/Wire/Type = "TsCylinder"
s:Ge/Wire/Parent = "Patient"
b:Ge/Wire/IsParallel = Ge/Applicator/IsParallel
s:Ge/Wire/ParallelWorldName = "Applicator"

# These for fixed position
#d:Ge/Wire/TransX = Ge/Applicator/Hole1/TransX mm
#d:Ge/Wire/TransY = Ge/Applicator/Hole1/TransY mm
#d:Ge/Wire/TransZ = 0. mm - Ge/SourcePellets/TransZ # Offset wire so
→SourcePellets is centered in Patient

# These to have wire move under time feature

```

(continues on next page)

(continued from previous page)

```

d:Ge/Wire/TransX      = Tf/WireX/Value mm
d:Ge/Wire/TransY      = Tf/WireY/Value mm
d:Ge/Wire/TransZ      = Tf/WireZ/Value - Ge/SourcePellets/TransZ mm # Offset
→wire so SourcePellets is centered in Patient

s:Ge/Wire/Material     = "G4_AIR"
d:Ge/Wire/RMax         = .295 mm
d:Ge/Wire/HalfRMax     = 0.5 * Ge/Wire/RMax mm
d:Ge/Wire/HL          = Ge/WireCylinder/HL + Ge/Wire/HalfRMax mm
b:Ge/Wire/Invisible = "True"

s:Ge/WireCylinder/Type = "TsCylinder"
s:Ge/WireCylinder/Parent = "Wire"
b:Ge/WireCylinder/IsParallel = Ge/Applicator/IsParallel
s:Ge/WireCylinder/ParallelWorldName = "Applicator"
s:Ge/WireCylinder/Material = "TitaniumNickel"
d:Ge/WireCylinder/RMax = Ge/Wire/RMax mm
d:Ge/WireCylinder/HL = 10.0 mm
d:Ge/WireCylinder/TransZ = Ge/Wire/HalfRMax mm
d:Ge/WireCylinder/DistanceFromEndOfActiveSourceToEndOfCylinder = 1. mm - Ge/Wire/RMax
s:Ge/WireCylinder/Color = "blue"

s:Ge/WireTip/Type      = "TsSphere"
s:Ge/WireTip/Parent    = "Wire"
b:Ge/WireTip/IsParallel = Ge/Applicator/IsParallel
s:Ge/WireTip/ParallelWorldName = "Applicator"
s:Ge/WireTip/Material  = "TitaniumNickel"
d:Ge/WireTip/RMax      = Ge/Wire/RMax mm
d:Ge/WireTip/STheta    = 90 deg
d:Ge/WireTip/DTheta    = 90 deg
d:Ge/WireTip/NegTransZ = Ge/WireCylinder/HL - Ge/Wire/HalfRMax mm
d:Ge/WireTip/TransZ    = -1. * Ge/WireTip/NegTransZ mm
s:Ge/WireTip/Color     = "blue"

s:Ge/SourcePellets/Type = "TsCylinder"
s:Ge/SourcePellets/Parent = "WireCylinder"
b:Ge/SourcePellets/IsParallel = Ge/Applicator/IsParallel
s:Ge/SourcePellets/ParallelWorldName = "Applicator"
s:Ge/SourcePellets/Material = Ge/Wire/Material
d:Ge/SourcePellets/RMax = .17 mm
d:Ge/SourcePellets/HL = 2.5 mm
d:Ge/SourcePellets/OnePelletHL = 0.5 * Ge/SourcePellets/HL mm
d:Ge/SourcePellets/CylinderHL = Ge/SourcePellets/OnePelletHL - Ge/SourcePellets/
→RMax mm
d:Ge/SourcePellets/Temp1 = Ge/WireCylinder/HL - Ge/SourcePellets/HL mm
d:Ge/SourcePellets/Temp2 = Ge/SourcePellets/Temp1 - Ge/WireCylinder/
→DistanceFromEndOfActiveSourceToEndOfCylinder mm
d:Ge/SourcePellets/TransZ = -1. * Ge/SourcePellets/Temp2 mm
b:Ge/SourcePellets/Invisible = "True"
s:Ge/SourcePellets/Color = "white"

s:Ge/SourcePellet/Type = "TsCylinder"
s:Ge/SourcePellet/Parent = "SourcePellets"
b:Ge/SourcePellet/IsParallel = Ge/Applicator/IsParallel
s:Ge/SourcePellet/ParallelWorldName = "Applicator"
s:Ge/SourcePellet/Material = Ge/Wire/Material
d:Ge/SourcePellet/RMax = Ge/SourcePellets/RMax mm

```

(continues on next page)

(continued from previous page)

```

d:Ge/Source1Pellet/HL           = Ge/SourcePellets/OnePelletHL mm
d:Ge/Source1Pellet/TransZ       = Ge/SourcePellets/OnePelletHL mm
b:Ge/Source1Pellet/Invisible = "True"

s:Ge/Source1Cylinder/Type       = "TsCylinder"
s:Ge/Source1Cylinder/Parent     = "Source1Pellet"
b:Ge/Source1Cylinder/IsParallel = Ge/Applicator/IsParallel
s:Ge/Source1Cylinder/ParallelWorldName = "Applicator"
s:Ge/Source1Cylinder/Material   = "G4_Ir"
d:Ge/Source1Cylinder/RMax       = Ge/SourcePellets/RMax mm
d:Ge/Source1Cylinder/HL        = Ge/SourcePellets/CylinderHL mm

s:Ge/Source1CapPlusZ/Type       = "TsSphere"
s:Ge/Source1CapPlusZ/Parent     = "Source1Pellet"
b:Ge/Source1CapPlusZ/IsParallel = Ge/Applicator/IsParallel
s:Ge/Source1CapPlusZ/ParallelWorldName = "Applicator"
s:Ge/Source1CapPlusZ/Material   = "G4_Ir"
d:Ge/Source1CapPlusZ/RMax       = Ge/SourcePellets/RMax mm
d:Ge/Source1CapPlusZ/STheta     = 0 deg
d:Ge/Source1CapPlusZ/DTheta     = 90 deg
d:Ge/Source1CapPlusZ/TransZ     = Ge/SourcePellets/CylinderHL mm

s:Ge/Source1CapMinusZ/Type      = "TsSphere"
s:Ge/Source1CapMinusZ/Parent    = "Source1Pellet"
b:Ge/Source1CapMinusZ/IsParallel = Ge/Applicator/IsParallel
s:Ge/Source1CapMinusZ/ParallelWorldName = "Applicator"
s:Ge/Source1CapMinusZ/Material  = "G4_Ir"
d:Ge/Source1CapMinusZ/RMax      = Ge/SourcePellets/RMax mm
d:Ge/Source1CapMinusZ/STheta    = 90 deg
d:Ge/Source1CapMinusZ/DTheta    = 90 deg
d:Ge/Source1CapMinusZ/TransZ    = -1. * Ge/SourcePellets/CylinderHL mm

s:Ge/Source2Pellet/Type         = "TsCylinder"
s:Ge/Source2Pellet/Parent       = "SourcePellets"
b:Ge/Source2Pellet/IsParallel   = Ge/Applicator/IsParallel
s:Ge/Source2Pellet/ParallelWorldName = "Applicator"
s:Ge/Source2Pellet/Material     = Ge/Wire/Material
d:Ge/Source2Pellet/RMax         = Ge/SourcePellets/RMax mm
d:Ge/Source2Pellet/HL          = Ge/SourcePellets/OnePelletHL mm
d:Ge/Source2Pellet/TransZ      = -1. * Ge/SourcePellets/OnePelletHL mm
b:Ge/Source2Pellet/Invisible = "True"

s:Ge/Source2Cylinder/Type       = "TsCylinder"
s:Ge/Source2Cylinder/Parent     = "Source2Pellet"
b:Ge/Source2Cylinder/IsParallel = Ge/Applicator/IsParallel
s:Ge/Source2Cylinder/ParallelWorldName = "Applicator"
s:Ge/Source2Cylinder/Material   = "G4_Ir"
d:Ge/Source2Cylinder/RMax       = Ge/SourcePellets/RMax mm
d:Ge/Source2Cylinder/HL        = Ge/SourcePellets/CylinderHL mm

s:Ge/Source2CapPlusZ/Type       = "TsSphere"
s:Ge/Source2CapPlusZ/Parent     = "Source2Pellet"
b:Ge/Source2CapPlusZ/IsParallel = Ge/Applicator/IsParallel
s:Ge/Source2CapPlusZ/ParallelWorldName = "Applicator"
s:Ge/Source2CapPlusZ/Material   = "G4_Ir"
d:Ge/Source2CapPlusZ/RMax       = Ge/SourcePellets/RMax mm
d:Ge/Source2CapPlusZ/STheta     = 0 deg

```

(continues on next page)



(continued from previous page)

```

d:Ge/Source2CapPlusZ/DTheta      = 90 deg
d:Ge/Source2CapPlusZ/TransZ      = Ge/SourcePellets/CylinderHL mm

s:Ge/Source2CapMinusZ/Type       = "TsSphere"
s:Ge/Source2CapMinusZ/Parent     = "Source2Pellet"
b:Ge/Source2CapMinusZ/IsParallel = Ge/Applicator/IsParallel
s:Ge/Source2CapMinusZ/ParallelWorldName = "Applicator"
s:Ge/Source2CapMinusZ/Material   = "G4_Ir"
d:Ge/Source2CapMinusZ/RMax       = Ge/SourcePellets/RMax mm
d:Ge/Source2CapMinusZ/STheta     = 90 deg
d:Ge/Source2CapMinusZ/DTheta     = 90 deg
d:Ge/Source2CapMinusZ/TransZ     = -1. * Ge/SourcePellets/CylinderHL mm

sv:Ma/TitaniumNickel/Components = 2 "Titanium" "Nickel"
uv:Ma/TitaniumNickel/Fractions  = 2 0.444 0.556
d:Ma/TitaniumNickel/Density     = 6.5 g/cm3
s:Ma/TitaniumNickel/DefaultColor = "grass"

# Radioactivity
s:So/Example/Type               = "Volumetric"
s:So/Example/Component          = "SourcePellets"
s:So/Example/ActiveMaterial     = "G4_Ir"
s:So/Example/BeamParticle       = "gamma"
i:So/Example/NumberOfHistoriesInRun = Tf/NumberOfHistoriesInRun/Value
b:So/Example/RecursivelyIncludeChildren = "True"
i:So/Example/MaxNumberOfPointsToSample = 1000000000

s:So/Example/BeamEnergySpectrumType = "Continuous"
dv:So/Example/BeamEnergySpectrumValues = 117 0.0614 0.0615 0.0616 0.0629 0.063 0.
→0631 0.065 0.0651 0.0652 0.0667 0.0668 0.0669 0.071 0.0711 0.0712 0.0713 0.0714 0.
→0715 0.0733 0.0734 0.0735 0.0753 0.0754 0.0755 0.0756 0.0757 0.0758 0.0777 0.0778 0.
→0779 0.1103 0.1104 0.1105 0.1362 0.1363 0.1364 0.1769 0.177 0.1771 0.2012 0.2013 0.
→2014 0.2057 0.2058 0.2059 0.2802 0.2803 0.2804 0.2832 0.2833 0.2834 0.2959 0.296 0.
→2961 0.3084 0.3085 0.3086 0.3164 0.3165 0.3166 0.3291 0.3292 0.3293 0.3744 0.3745 0.
→3746 0.4164 0.4165 0.4166 0.4204 0.4205 0.4206 0.468 0.4681 0.4682 0.4845 0.4846 0.
→4847 0.4852 0.4853 0.4854 0.489 0.4891 0.4892 0.5885 0.5886 0.5887 0.5934 0.5935 0.
→5936 0.5993 0.5994 0.5995 0.6043 0.6044 0.6045 0.6124 0.6125 0.6126 0.7038 0.7039 0.
→704 0.7657 0.7658 0.7659 0.8844 0.8845 0.8846 1.0614 1.0615 1.0616 1.0898 1.0899 1.
→089 1.3781 1.3782 1.3783 MeV

uv:So/Example/BeamEnergySpectrumWeightsUnscaled = 117
1e-08 0.412 1e-08 1e-08 0.7039 1e-08 1e-08 1.1309 1e-08 1e-08 1.9178 1e-08 1e-08 0.
→0827 1e-08 1e-08 0.16 1e-08 1e-08 0.056 1e-08 1e-08 0.2292 1e-08 1e-08 0.4408 1e-08
→1e-08 0.157 1e-08 1e-08 0.0042 1e-08 1e-08 0.086 1e-08 1e-08 0.0018 1e-08 1e-08 0.
→1624 1e-08 1e-08 1.1468 1e-08 1e-08 0.0039 1e-08 1e-08 0.0913 1e-08 1e-08 12.3498
→1e-08 1e-08 12.7626 1e-08 1e-08 35.566 1e-08 1e-08 0.006 1e-08 1e-08 0.2493 1e-08
→1e-08 0.2877 1e-08 1e-08 0.0237 1e-08 1e-08 20.5587 1e-08 1e-08 1.0942 1e-08 1e-08
→0.001 1e-08 1e-08 0.1504 1e-08 1e-08 1.9423 1e-08 1e-08 0.0181 1e-08 1e-08 0.0017
→1e-08 1e-08 3.5361 1e-08 1e-08 2.2962 1e-08 1e-08 0.0018 1e-08 1e-08 0.0006 1e-08
→1e-08 0.1251 1e-08 1e-08 0.0228 1e-08 1e-08 0.0005 1e-08 1e-08 0.0005 1e-8

# Scale these values to give total weight of 1.
uv:So/Example/BeamEnergySpectrumWeights = .0102269 * So/Example/
→BeamEnergySpectrumWeightsUnscaled

```

(continues on next page)

(continued from previous page)

```

# Graphics
s:Gr/ViewA/Type           = "OpenGL"
b:Gr/ViewA/IncludeStepPoints = "True"
d:Gr/ViewA/Phi = 0 deg
d:Gr/ViewA/Theta = 65 deg
u:Gr/ViewA/Zoom = 2.

# Simplify physics to only require electromagnetics
# (don't need hadronic physics which is otherwise in default setup)
sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
sv:Ph/Default/LayeredMassGeometryWorlds = 1 "Applicator"

# Overall run parameters
i:Ts/ShowHistoryCountAtInterval = 100
Ts/NumberOfThreads = 1

i:Tf/Verbosity           = 2
d:Tf/TimelineEnd         = 70.0 s
i:Tf/NumberOfSequentialTimes = 70

s:Tf/WireX/Function       = "Step"
dv:Tf/WireX/Times         = 7 10. 20. 30. 40. 50. 60. 70. s
dv:Tf/WireX/Values        = 7 Ge/Applicator/Hole0/TransX Ge/Applicator/Hole1/
→TransX Ge/Applicator/Hole2/TransX Ge/Applicator/Hole3/TransX Ge/Applicator/Hole4/
→TransX Ge/Applicator/Hole5/TransX Ge/Applicator/Hole6/TransX mm
d:Tf/WireX/RepetitionInterval = 70. s

s:Tf/WireY/Function       = "Step"
dv:Tf/WireY/Times         = 7 10. 20. 30. 40. 50. 60. 70. s
dv:Tf/WireY/Values        = 7 Ge/Applicator/Hole0/TransY Ge/Applicator/Hole1/
→TransY Ge/Applicator/Hole2/TransY Ge/Applicator/Hole3/TransY Ge/Applicator/Hole4/
→TransY Ge/Applicator/Hole5/TransY Ge/Applicator/Hole6/TransY mm
d:Tf/WireY/RepetitionInterval = 70. s

s:Tf/WireZ/Function       = "Step"
dv:Tf/WireZ/Times         = 10 1. 2. 3. 4. 5. 6. 7. 8. 9. 10. s
dv:Tf/WireZ/Values        = 10 -18. -14. -10. -6. -2. 2. 6. 10. 14. 18. mm
d:Tf/WireZ/RepetitionInterval = 10. s

s:Tf/NumberOfHistoriesInRun/Function = "Step"
dv:Tf/NumberOfHistoriesInRun/Times   = 7 10. 20. 30. 40. 50. 60. 70. s
iv:Tf/NumberOfHistoriesInRun/Values  = 7 70 30 10 10 40 60 70
d:Tf/NumberOfHistoriesInRun/RepetitionInterval = 70. s

```

## 24.2 DoseTo4DCT.txt

```

# Demonstrates use of a DICOM file.
# You must unzip synthetic_lung.tar.bz2 before you run this example.
# Reads in and displays a simple test DICOM that represents
# three different time phases of a simplified lung phantom.
# This phantom consists of a tumor in a cylinder of water in a tube of air.

```

(continues on next page)

(continued from previous page)

```

includeFile = HUtoMaterialSchneider.txt

# Geometry
s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/Patient/Parent      = "World"
s:Ge/Patient/Material    = "G4_WATER"
d:Ge/Patient/TransX      = 0. cm # -8. cm
d:Ge/Patient/TransY      = 0. cm # -8. cm
d:Ge/Patient/TransZ      = 0. m
d:Ge/Patient/RotX        = 90. deg
d:Ge/Patient/RotY        = 0. deg
d:Ge/Patient/RotZ        = 0. deg
s:Ge/Patient/Type        = "TsDicomPatient"
s:Ge/Patient/DicomDirectory = Tf/ImageName/Value
#b:Ge/Patient/PreLoadAllMaterials = "True"

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
#iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
iv:Ge/Patient/ShowSpecificSlicesY = 1 -1
#iv:Ge/Patient/ShowSpecificSlicesZ = 1 -2

# Time feature, load different CTs at different times
s:Tf/ImageName/Function = "Step"
dv:Tf/ImageName/Times   = 3 10 20 30 ms
sv:Tf/ImageName/Values  = 3 "synthetic_lung/lung-1" "synthetic_lung/lung-2"
↪ "synthetic_lung/lung-3"

d:Tf/TimelineEnd = 120.0 ms
i:Tf/NumberOfSequentialTimes = 12
i:Tf/Verbosity = 1

s:Tf/Zoom/Function      = "Step"
dv:Tf/Zoom/Times        = 3 30 60 180 ms
uv:Tf/Zoom/Values       = 3 1.4 3. 9.
d:Tf/Zoom/RepetitionInterval = 360. ms

s:Tf/Pan/Function        = "Step"
dv:Tf/Pan/Times          = 3 30 60 180 ms
uv:Tf/Pan/Values         = 3 0. .1 .1
d:Tf/Pan/RepetitionInterval = 360. ms

s:Tf/PanY/Function       = "Step"
dv:Tf/PanY/Times         = 3 30 60 180 ms
uv:Tf/PanY/Values        = 3 0. -.05 -.05
d:Tf/PanY/RepetitionInterval = 360. ms

# Uncomment to enable scoring
s:Sc/DoseAtPhantom/Quantity = "DoseToMedium"

```

(continues on next page)

(continued from previous page)

```

s:Sc/DoseAtPhantom/Component           = "Patient"
b:Sc/DoseAtPhantom/OutputToConsole      = "False"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom/OutputType           = "DICOM"
b:Sc/DoseAtPhantom/DICOMOutput32BitsPerPixel = "False"

# Uncomment to split scoring by ImageName
#s:Sc/DoseAtPhantom/SplitByTimeFeature = "ImageName"

s:So/Example/Type                       = "Beam"
s:So/Example/Component                  = "BeamPosition"
s:So/Example/BeamParticle               = "proton"
d:So/Example/BeamEnergy                 = 120. MeV
u:So/Example/BeamEnergySpread           = 0.757504
s:So/Example/BeamPositionDistribution    = "Flat"
s:So/Example/BeamPositionCutoffShape    = "Ellipse"
d:So/Example/BeamPositionCutoffX        = 0.5 cm
d:So/Example/BeamPositionCutoffY        = 0.5 cm
s:So/Example/BeamAngularDistribution    = "None"
i:So/Example/NumberOfHistoriesInRun     = 100

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

# Graphics
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 45 deg
d:Gr/ViewA/Phi = 45 deg
u:Gr/ViewA/Zoom = Tf/Zoom/Value
u:Gr/ViewA/TransX = Tf/Pan/Value
u:Gr/ViewA/TransY = Tf/PanY/Value

# Extra graphics options to handle complex voxel geometries
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 2100000

i:Ts/ShowHistoryCountAtInterval = 5000

b:Ts/PauseBeforeQuit = "False"

```

## 24.3 DoseToCT.txt

```

# Demonstrates scoring in a DICOM file.
# You must unzip DICOM_Box.zip before you run this example.
# Reads in and displays a simple test DICOM that represents
# a box of water in air.
# Can optionally display patient from XiO input file.
# If Graphics is enabled, the display writes very slowly to the screen.
# This will be improved in an upcoming new release.
# Until then, one useful trick is that if you iconize the graphics window,
# then un-iconize it when drawing is done, the drawing will go much faster.

includeFile = HUtoMaterialSchneider.txt

```

(continues on next page)

(continued from previous page)

```

s:Ge/World/Material      = "Vacuum"
d:Ge/World/HLX           = 2.0 m
d:Ge/World/HLY           = 2.0 m
d:Ge/World/HLZ           = 2.0 m
b:Ge/World/Invisible     = "TRUE"

s:Ge/Patient/Parent      = "World"
s:Ge/Patient/Material    = "G4_WATER"
d:Ge/Patient/TransX      = 0. m
d:Ge/Patient/TransY      = 0. m
d:Ge/Patient/TransZ      = 0. m
d:Ge/Patient/RotX        = 0. deg
d:Ge/Patient/RotY        = 0. deg
d:Ge/Patient/RotZ        = 0. deg

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
#iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
#iv:Ge/Patient/ShowSpecificSlicesY = 1 -2
iv:Ge/Patient/ShowSpecificSlicesZ = 1 -2

# Can read either DICOM or XiO input files.
# To change input format, change which
# Ge/Patient/Type file is commented out below.

# Additional needed for DICOM input
s:Ge/Patient/Type        = "TsDicomPatient"
s:Ge/Patient/DicomDirectory = "DICOM_Box"

# Additional needed for XiO input
#s:Ge/Patient/Type        = "TsXiOPatient"
s:Ge/Patient/InputDirectory = "./"
s:Ge/Patient/InputFile      = "XIO_Example.dat"
i:Ge/Patient/NumberOfVoxelsX = 25
i:Ge/Patient/NumberOfVoxelsY = 25
iv:Ge/Patient/NumberOfVoxelsZ = 2 3 7
d:Ge/Patient/VoxelSizeX     = 2.0 mm
d:Ge/Patient/VoxelSizeY     = 2.0 mm
dv:Ge/Patient/VoxelSizeZ    = 2 5. 2.5 mm

s:Sc/DoseOnCTGrid/Quantity      = "DoseToMedium"
s:Sc/DoseOnCTGrid/Component     = "Patient"
b:Sc/DoseOnCTGrid/OutputToConsole = "True"
s:Sc/DoseOnCTGrid/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseOnCTGrid/OutputType    = "DICOM"
b:Sc/DoseOnCTGrid/DICOMOutput32BitsPerPixel = "True"

s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "proton"
d:So/Example/BeamEnergy         = 30. MeV
u:So/Example/BeamEnergySpread   = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"

```

(continues on next page)

(continued from previous page)

```
d:So/Example/BeamPositionCutoffX      = 5. cm
d:So/Example/BeamPositionCutoffY      = 20. cm
s:So/Example/BeamAngularDistribution  = "None"
i:So/Example/NumberOfHistoriesInRun   = 500

# Uncomment to activate graphics
s:Gr/ViewA/Type                       = "OpenGL"
sv:Gr/ViewA/VisibleWorlds             = 1 "All"
i:Gr/ViewA/WindowSizeX                = 900
i:Gr/ViewA/WindowSizeY                = 900
d:Gr/ViewA/Theta                      = 70 deg
d:Gr/ViewA/Phi                       = 10 deg

# Extra graphics options to handle complex voxel geometries
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 210000000

i:Ts/ShowHistoryCountAtInterval       = 100

b:Ts/PauseBeforeQuit                  = "True"
```

## 24.4 HUtoMaterialSchneiber.txt

```
# Topas parameter settings for conversion of Hounsfield Units to materials using the
↳Schneider paper

s:Ge/Patient/ImagingToMaterialConverter = "Schneider"

# Correction Factor for the relative stopping power of Geant4 and the XiO planning
↳system:

dv:Ge/Patient/DensityCorrection = 3996 9.35212 5.55269 4.14652 3.41395 2.9645 2.66061
↳2.44144 2.27588 2.1464 2.04239 1.95698 1.8856 1.82506 1.77307 1.72791 1.68835 1.
↳6534 1.62229 1.59442 1.56932 1.54659 1.52591 1.50701 1.48968 1.47373 1.45899 1.
↳44534 1.43265 1.42084 1.40981 1.39949 1.3898 1.38071 1.37214 1.36406 1.35643 1.
↳34921 1.34237 1.33587 1.3297 1.32383 1.31824 1.31291 1.30781 1.30295 1.29829 1.
↳29383 1.28956 1.28546 1.28152 1.1284 1.12519 1.12209 1.11912 1.11625 1.11348 1.
↳11081 1.10823 1.10574 1.10333 1.101 1.09875 1.09657 1.09445 1.0924 1.09041 1.08848
↳1.08661 1.08479 1.08303 1.08131 1.07964 1.07801 1.07643 1.0749 1.0734 1.07194 1.
↳07052 1.06913 1.06778 1.06646 1.06518 1.06392 1.0627 1.0615 1.06033 1.05919 1.05808
↳1.05698 1.05592 1.05487 1.05386 1.05286 1.05188 1.05092 1.04999 1.04907 1.04817 1.
↳04728 1.04643 1.04558 1.04475 1.04394 1.04314 1.04235 1.04158 1.04084 1.04009 1.
↳03936 1.03866 1.03796 1.03726 1.0366 1.03593 1.03527 1.03463 1.03401 1.03339 1.
↳03277 1.03218 1.03159 1.03101 1.03044 1.02988 1.02933 1.02878 1.02825 1.02772 1.
↳0272 1.0267 1.02619 1.0257 1.02522 1.02473 1.02426 1.02379 1.02334 1.02288 1.02243
↳1.022 1.02156 1.02113 1.02072 1.02029 1.01988 1.01948 1.01909 1.01869 1.0183 1.
↳01793 1.01754 1.01717 1.0168 1.01644 1.01608 1.01572 1.01538 1.01503 1.01469 1.
↳01436 1.01403 1.01369 1.01337 1.01306 1.01273 1.01242 1.01212 1.01181 1.0115 1.
↳01121 1.01092 1.01062 1.01033 1.01006 1.00977 1.00949 1.00922 1.00895 1.00867 1.
↳00842 1.00815 1.00789 1.00763 1.00738 1.00713 1.00687 1.00663 1.00639 1.00614 1.
↳00591 1.00568 1.00544 1.0052 1.00498 1.00476 1.00453 1.00431 1.00409 1.00387 1.
↳00366 1.00345 1.00323 1.00302 1.00282 1.00261 1.00241 1.00221 1.00201 1.00181 1.
↳00162 1.00143 1.00123 1.00104 1.00086 1.00067 1.00049 1.0003 1.00012 0.99994 0.
↳99977 0.99959 0.99941 0.99924 0.99907 0.9989 0.99873 0.99856 0.9984 0.99824 0.99807
↳0.99791 0.99775 0.99759 0.99744 0.99728 0.99713 0.99697 0.99682 0.99667 0.99652 0.
↳99637 0.99623 0.99608 0.99594 0.99579 0.99565 0.99551 0.99537 0.99523 0.99509 0.
↳99496 0.99482 0.99469 0.99455 0.99442 0.99429 0.99416 0.99403 0.9939 0.99376 0.
↳99365 0.99352 0.9934 0.99328 0.99315 0.99303 0.99292 0.99279 0.99267 0.99256 0.
↳99244 0.99232 0.99221 0.99209 0.99198 0.99187 0.99176 0.99164 0.99153 0.99143 0.
↳99132 0.99121 0.9911 0.991 0.99089 0.99078 0.99067 0.99056 0.99045 0.99034 0.99023 0.99012 0.99001 0.9899 0.98979 0.98968 0.98957 0.98946 0.98935 0.98924 0.98913 0.98902 0.98891 0.9888 0.98869 0.98858 0.98847 0.98836 0.98825 0.98814 0.98803 0.98792 0.98781 0.9877 0.98759 0.98748 0.98737 0.98726 0.98715 0.98704 0.98693 0.98682 0.98671 0.9866 0.98649 0.98638 0.98627 0.98616 0.98605 0.98594 0.98583 0.98572 0.98561 0.9855 0.98539 0.98528 0.98517 0.98506 0.98495 0.98484 0.98473 0.98462 0.98451 0.9844 0.98429 0.98418 0.98407 0.98396 0.98385 0.98374 0.98363 0.98352 0.98341 0.9833 0.98319 0.98308 0.98297 0.98286 0.98275 0.98264 0.98253 0.98242 0.98231 0.9822 0.98209 0.98198 0.98187 0.98176 0.98165 0.98154 0.98143 0.98132 0.98121 0.9811 0.98099 0.98088 0.98077 0.98066 0.98055 0.98044 0.98033 0.98022 0.98011 0.98001 0.9799 0.97979 0.97968 0.97957 0.97946 0.97935 0.97924 0.97913 0.97902 0.97891 0.9788 0.97869 0.97858 0.97847 0.97836 0.97825 0.97814 0.97803 0.97792 0.97781 0.9777 0.97759 0.97748 0.97737 0.97726 0.97715 0.97704 0.97693 0.97682 0.97671 0.9766 0.97649 0.97638 0.97627 0.97616 0.97605 0.97594 0.97583 0.97572 0.97561 0.9755 0.97539 0.97528 0.97517 0.97506 0.97495 0.97484 0.97473 0.97462 0.97451 0.9744 0.97429 0.97418 0.97407 0.97396 0.97385 0.97374 0.97363 0.97352 0.97341 0.9733 0.97319 0.97308 0.97297 0.97286 0.97275 0.97264 0.97253 0.97242 0.97231 0.9722 0.97209 0.97198 0.97187 0.97176 0.97165 0.97154 0.97143 0.97132 0.97121 0.9711 0.97099 0.97088 0.97077 0.97066 0.97055 0.97044 0.97033 0.97022 0.97011 0.97001 0.9699 0.96979 0.96968 0.96957 0.96946 0.96935 0.96924 0.96913 0.96902 0.96891 0.9688 0.96869 0.96858 0.96847 0.96836 0.96825 0.96814 0.96803 0.96792 0.96781 0.9677 0.96759 0.96748 0.96737 0.96726 0.96715 0.96704 0.966
```

(continues on next page)

(continued from previous page)

```

# Formula: Density = (Offset + (Factor*(FactorOffset + HU[-1000,2995] ))) *
→DensityCorrection
iv:Ge/Patient/SchneiderHounsfieldUnitSections = 8 -1000 -98 15 23 101 2001 2995 2996
uv:Ge/Patient/SchneiderDensityOffset = 7 0.00121 1.018 1.03 1.003 1.017 2.201 4.54
uv:Ge/Patient/SchneiderDensityFactor = 7 0.001029700665188 0.000893 0.0 0.001169 0.
→000592 0.0005 0.0
uv:Ge/Patient/SchneiderDensityFactorOffset = 7 1000. 0. 1000. 0. 0. -2000. 0.

# Define Materials used for HU
sv:Ge/Patient/SchneiderElements = 13 "Hydrogen" "Carbon" "Nitrogen" "Oxygen"
→"Magnesium" "Phosphorus" "Sulfur" "Chlorine" "Argon" "Calcium" "Sodium" "Potassium"
→"Titanium"
iv:Ge/Patient/SchneiderHUToMaterialSections = 26 -1000 -950 -120 -83 -53 -23 7 18 80
→120 200 300 400 500 600 700 800 900 1000 1100 1200 1300 1400 1500 2995 2996
uv:Ge/Patient/SchneiderMaterialsWeight1 = 13 0.0 0.0 0.755 0.232 0.0
→0.0 0.0 0.0 0.013 0.0 0.0 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight2 = 13 0.103 0.105 0.031 0.749 0.0
→0.002 0.003 0.003 0.0 0.0 0.002 0.002 0.0
uv:Ge/Patient/SchneiderMaterialsWeight3 = 13 0.116 0.681 0.002 0.198 0.0
→0.0 0.001 0.001 0.0 0.0 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight4 = 13 0.113 0.567 0.009 0.308 0.0
→0.0 0.001 0.001 0.0 0.0 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight5 = 13 0.110 0.458 0.015 0.411 0.0
→0.001 0.002 0.002 0.0 0.0 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight6 = 13 0.108 0.356 0.022 0.509 0.0
→0.001 0.002 0.002 0.0 0.0 0.0 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight7 = 13 0.106 0.284 0.026 0.578 0.0
→0.001 0.002 0.002 0.0 0.0 0.0 0.001 0.0
uv:Ge/Patient/SchneiderMaterialsWeight8 = 13 0.103 0.134 0.030 0.723 0.0
→0.002 0.002 0.002 0.0 0.0 0.002 0.002 0.0
uv:Ge/Patient/SchneiderMaterialsWeight9 = 13 0.094 0.207 0.062 0.622 0.0
→0.0 0.006 0.003 0.0 0.0 0.006 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight10 = 13 0.095 0.455 0.025 0.355 0.0
→0.021 0.001 0.001 0.0 0.045 0.001 0.001 0.0
uv:Ge/Patient/SchneiderMaterialsWeight11 = 13 0.089 0.423 0.027 0.363 0.0
→0.030 0.001 0.001 0.0 0.064 0.001 0.001 0.0
uv:Ge/Patient/SchneiderMaterialsWeight12 = 13 0.082 0.391 0.029 0.372 0.0
→0.039 0.001 0.001 0.0 0.083 0.001 0.001 0.0
uv:Ge/Patient/SchneiderMaterialsWeight13 = 13 0.076 0.361 0.030 0.380 0.001
→0.047 0.002 0.001 0.0 0.101 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight14 = 13 0.071 0.335 0.032 0.387 0.001
→0.054 0.002 0.0 0.0 0.117 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight15 = 13 0.066 0.310 0.033 0.394 0.001
→0.061 0.002 0.0 0.0 0.132 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight16 = 13 0.061 0.287 0.035 0.400 0.001
→0.067 0.002 0.0 0.0 0.146 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight17 = 13 0.056 0.265 0.036 0.405 0.002
→0.073 0.003 0.0 0.0 0.159 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight18 = 13 0.052 0.246 0.037 0.411 0.002
→0.078 0.003 0.0 0.0 0.170 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight19 = 13 0.049 0.227 0.038 0.416 0.002
→0.083 0.003 0.0 0.0 0.181 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight20 = 13 0.045 0.210 0.039 0.420 0.002
→0.088 0.003 0.0 0.0 0.192 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight21 = 13 0.042 0.194 0.040 0.425 0.002
→0.092 0.003 0.0 0.0 0.201 0.001 0.0 0.0

```

(continues on next page)

(continued from previous page)

```

uv:Ge/Patient/SchneiderMaterialsWeight22 = 13 0.039 0.179 0.041 0.429 0.002
→ 0.096 0.003 0.0 0.0 0.210 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight23 = 13 0.036 0.165 0.042 0.432 0.002
→ 0.100 0.003 0.0 0.0 0.219 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight24 = 13 0.034 0.155 0.042 0.435 0.002
→ 0.103 0.003 0.0 0.0 0.225 0.001 0.0 0.0
uv:Ge/Patient/SchneiderMaterialsWeight25 = 13 0.0 0.0 0.0 0.0 0.0 0.0
→ 0.0 0.0 0.0 0.0 0.0 0.0 1.0

iv:Gr/Color/PatientTissue1 = 3 63 63 63
iv:Gr/Color/PatientTissue2 = 3 100 0 0
iv:Gr/Color/PatientTissue3 = 3 100 30 0
iv:Gr/Color/PatientTissue4 = 3 100 60 0
iv:Gr/Color/PatientTissue5 = 3 100 90 0
iv:Gr/Color/PatientTissue6 = 3 100 120 0
iv:Gr/Color/PatientTissue7 = 3 100 150 0
iv:Gr/Color/PatientTissue8 = 3 100 180 0
iv:Gr/Color/PatientTissue9 = 3 100 210 0
iv:Gr/Color/PatientTissue10 = 3 100 240 0
iv:Gr/Color/PatientTissue11 = 3 200 200 0
iv:Gr/Color/PatientTissue12 = 3 200 200 30
iv:Gr/Color/PatientTissue13 = 3 200 200 60
iv:Gr/Color/PatientTissue14 = 3 200 200 90
iv:Gr/Color/PatientTissue15 = 3 200 200 120
iv:Gr/Color/PatientTissue16 = 3 200 200 150
iv:Gr/Color/PatientTissue17 = 3 200 200 180
iv:Gr/Color/PatientTissue18 = 3 200 200 210
iv:Gr/Color/PatientTissue19 = 3 200 200 240
iv:Gr/Color/PatientTissue20 = 3 230 200 240
iv:Gr/Color/PatientTissue21 = 3 230 230 240
iv:Gr/Color/PatientTissue22 = 3 230 230 240
iv:Gr/Color/PatientTissue23 = 3 240 240 240
iv:Gr/Color/PatientTissue24 = 3 255 255 255
iv:Gr/Color/PatientTissue25 = 3 100 100 100

```

## 24.5 HUtoMaterialSchneider2.txt

```

# Topas parameter settings for conversion of Hounsfield Units to materials using the
→ Schneider paper

s:Ge/Patient2/ImagingToMaterialConverter = "Schneider"

# Correction Factor for the relative stopping power of Geant4 and the XiO planning
→ system:
dv:Ge/Patient2/DensityCorrection = 3996 9.35212 5.55269 4.14652 3.41395 2.9645 2.
→ 66061 2.44144 2.27588 2.1464 2.04239 1.95698 1.8856 1.82506 1.77307 1.72791 1.68835
→ 1.6534 1.62229 1.59442 1.56932 1.54659 1.52591 1.50701 1.48968 1.47373 1.45899 1.
→ 44534 1.43265 1.42084 1.40981 1.39949 1.3898 1.38071 1.37214 1.36406 1.35643 1.
→ 34921 1.34237 1.33587 1.3297 1.32383 1.31824 1.31291 1.30781 1.30295 1.29829 1.
→ 29383 1.28956 1.28546 1.28152 1.1284 1.12519 1.12209 1.11912 1.11625 1.11348 1.
→ 11081 1.10823 1.10574 1.10333 1.101 1.09875 1.09657 1.09445 1.0924 1.09041 1.08848
→ 1.08661 1.08479 1.08303 1.08131 1.07964 1.07801 1.07643 1.0749 1.0734 1.07194 1.
→ 07052 1.06913 1.06778 1.06646 1.06518 1.06392 1.0627 1.0615 1.06033 1.05919 1.05808
→ 1.05698 1.05592 1.05487 1.05386 1.05286 1.05188 1.05092 1.04999 1.04907 1.04817 1.
→ 04728 1.04643 1.04558 1.04475 1.04394 1.04314 1.04235 1.04158 1.04084 1.03999 1.
→ 03936 1.03866 1.03796 1.03726 1.0366 1.03593 1.03527 1.03463 1.03401 1.03339 1.
→ 03277 1.03218 1.03159 1.03101 1.03044 1.02988 1.02933 1.02878 1.02825 1.02772 1.
→ 0272 1.0267 1.02619 1.0257 1.02522 1.02473 1.02426 1.02379 1.02334 1.02289 1.
→ 0224 1.0219 1.02143 1.02097 1.02052 1.02007 1.01962 1.01917 1.01872 1.01827 1.
→ 01782 1.01737 1.01692 1.01647 1.01602 1.01557 1.01512 1.01467 1.01422 1.01377 1.
→ 01332 1.01287 1.01242 1.01197 1.01152 1.01107 1.01062 1.01017 1.00972 1.00927 1.
→ 00882 1.00837 1.00792 1.00747 1.00702 1.00657 1.00612 1.00567 1.00522 1.00477 1.
→ 00432 1.00387 1.00342 1.00297 1.00252 1.00207 1.00162 1.00117 1.00072 1.00027 1.
→ 00082 1.00037 1.00027 1.00017 1.00007 1.00002 1.00002 1.00002 1.00002 1.00002 1.
→ 00002 1.00002 1.00002 1.00002 1.00002 1.00002 1.00002 1.00002 1.00002 1.00002

```

(continues on next page)



(continued from previous page)

```

# Formula: Density = (Offset + (Factor*(FactorOffset + HU[-1000,2995] ))) *
→DensityCorrection
iv:Ge/Patient2/SchneiderHounsfieldUnitSections = 8 -1000 -98 15 23 101 2001 2995 2996
uv:Ge/Patient2/SchneiderDensityOffset = 7 0.00121 1.018 1.03 1.003 1.017 2.201 4.54
uv:Ge/Patient2/SchneiderDensityFactor = 7 0.001029700665188 0.000893 0.0 0.001169 0.
→000592 0.0005 0.0
uv:Ge/Patient2/SchneiderDensityFactorOffset = 7 1000. 0. 1000. 0. 0. -2000. 0.

# Define Materials used for HU
sv:Ge/Patient2/SchneiderElements = 13 "Hydrogen" "Carbon" "Nitrogen" "Oxygen"
→"Magnesium" "Phosphorus" "Sulfur" "Chlorine" "Argon" "Calcium" "Sodium" "Potassium"
→"Titanium"
iv:Ge/Patient2/SchneiderHUToMaterialSections = 26 -1000 -950 -120 -83 -53 -23 7 18 80
→120 200 300 400 500 600 700 800 900 1000 1100 1200 1300 1400 1500 2995 2996
uv:Ge/Patient2/SchneiderMaterialsWeight1 = 13 0.0 0.0 0.755 0.232 0.0
→0.0 0.0 0.0 0.013 0.0 0.0 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight2 = 13 0.103 0.105 0.031 0.749 0.0
→0.002 0.003 0.003 0.0 0.0 0.002 0.002 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight3 = 13 0.116 0.681 0.002 0.198 0.0
→0.0 0.001 0.001 0.0 0.0 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight4 = 13 0.113 0.567 0.009 0.308 0.0
→0.0 0.001 0.001 0.0 0.0 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight5 = 13 0.110 0.458 0.015 0.411 0.0
→0.001 0.002 0.002 0.0 0.0 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight6 = 13 0.108 0.356 0.022 0.509 0.0
→0.001 0.002 0.002 0.0 0.0 0.0 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight7 = 13 0.106 0.284 0.026 0.578 0.0
→0.001 0.002 0.002 0.0 0.0 0.0 0.001 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight8 = 13 0.103 0.134 0.030 0.723 0.0
→0.002 0.002 0.002 0.0 0.0 0.002 0.002 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight9 = 13 0.094 0.207 0.062 0.622 0.0
→0.0 0.006 0.003 0.0 0.0 0.006 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight10 = 13 0.095 0.455 0.025 0.355 0.0
→0.021 0.001 0.001 0.0 0.045 0.001 0.001 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight11 = 13 0.089 0.423 0.027 0.363 0.0
→0.030 0.001 0.001 0.0 0.064 0.001 0.001 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight12 = 13 0.082 0.391 0.029 0.372 0.0
→0.039 0.001 0.001 0.0 0.083 0.001 0.001 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight13 = 13 0.076 0.361 0.030 0.380 0.001
→0.047 0.002 0.001 0.0 0.101 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight14 = 13 0.071 0.335 0.032 0.387 0.001
→0.054 0.002 0.0 0.0 0.117 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight15 = 13 0.066 0.310 0.033 0.394 0.001
→0.061 0.002 0.0 0.0 0.132 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight16 = 13 0.061 0.287 0.035 0.400 0.001
→0.067 0.002 0.0 0.0 0.146 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight17 = 13 0.056 0.265 0.036 0.405 0.002
→0.073 0.003 0.0 0.0 0.159 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight18 = 13 0.052 0.246 0.037 0.411 0.002
→0.078 0.003 0.0 0.0 0.170 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight19 = 13 0.049 0.227 0.038 0.416 0.002
→0.083 0.003 0.0 0.0 0.181 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight20 = 13 0.045 0.210 0.039 0.420 0.002
→0.088 0.003 0.0 0.0 0.192 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight21 = 13 0.042 0.194 0.040 0.425 0.002
→0.092 0.003 0.0 0.0 0.201 0.001 0.0 0.0

```

(continues on next page)

(continued from previous page)

```

uv:Ge/Patient2/SchneiderMaterialsWeight22 = 13 0.039 0.179 0.041 0.429 0.002
→ 0.096 0.003 0.0 0.0 0.210 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight23 = 13 0.036 0.165 0.042 0.432 0.002
→ 0.100 0.003 0.0 0.0 0.219 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight24 = 13 0.034 0.155 0.042 0.435 0.002
→ 0.103 0.003 0.0 0.0 0.225 0.001 0.0 0.0
uv:Ge/Patient2/SchneiderMaterialsWeight25 = 13 0.0 0.0 0.0 0.0 0.0
→ 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0

iv:Gr/Color/Patient2Tissue1 = 3 63 63 63
iv:Gr/Color/Patient2Tissue2 = 3 100 0 0
iv:Gr/Color/Patient2Tissue3 = 3 100 30 0
iv:Gr/Color/Patient2Tissue4 = 3 100 60 0
iv:Gr/Color/Patient2Tissue5 = 3 100 90 0
iv:Gr/Color/Patient2Tissue6 = 3 100 120 0
iv:Gr/Color/Patient2Tissue7 = 3 100 150 0
iv:Gr/Color/Patient2Tissue8 = 3 100 180 0
iv:Gr/Color/Patient2Tissue9 = 3 100 210 0
iv:Gr/Color/Patient2Tissue10 = 3 100 240 0
iv:Gr/Color/Patient2Tissue11 = 3 200 200 0
iv:Gr/Color/Patient2Tissue12 = 3 200 200 30
iv:Gr/Color/Patient2Tissue13 = 3 200 200 60
iv:Gr/Color/Patient2Tissue14 = 3 200 200 90
iv:Gr/Color/Patient2Tissue15 = 3 200 200 120
iv:Gr/Color/Patient2Tissue16 = 3 200 200 150
iv:Gr/Color/Patient2Tissue17 = 3 200 200 180
iv:Gr/Color/Patient2Tissue18 = 3 200 200 210
iv:Gr/Color/Patient2Tissue19 = 3 200 200 240
iv:Gr/Color/Patient2Tissue20 = 3 230 200 240
iv:Gr/Color/Patient2Tissue21 = 3 230 230 240
iv:Gr/Color/Patient2Tissue22 = 3 230 230 240
iv:Gr/Color/Patient2Tissue23 = 3 240 240 240
iv:Gr/Color/Patient2Tissue24 = 3 255 255 255
iv:Gr/Color/Patient2Tissue25 = 3 100 100 100

```

## 24.6 Implant.txt

```

# Demonstrates use of a DICOM file.
# You must unzip DICOM_Box.zip before you run this example.
# Reads in and displays a simple test DICOM that represents
# a box of water in air
# and uses Layered Mass Geometry to overlay a titanium sphere "implant"
# onto this DICOM geomtry.
# The display writes very slowly to the screen.
# This will be improved in an upcoming new release.
# Until then, one useful trick is that if you iconize the graphics window,
# then un-inconize it when drawing is done, the drawing will go much faster.

includeFile = HUtoMaterialSchneider.txt

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX = 2.0 m
d:Ge/World/HLY = 2.0 m
d:Ge/World/HLZ = 2.0 m

```

(continues on next page)

(continued from previous page)

```

b:Ge/World/Invisible = "TRUE"

s:Ge/Patient/Parent      = "World"
s:Ge/Patient/Material    = "G4_WATER"
d:Ge/Patient/TransX      = 0. m
d:Ge/Patient/TransY      = 0. m
d:Ge/Patient/TransZ      = 0. m
d:Ge/Patient/RotX        = 0. deg
d:Ge/Patient/RotY        = 0. deg
d:Ge/Patient/RotZ        = 0. deg
s:Ge/Patient/Type        = "TsDicomPatient"
s:Ge/Patient/DicomDirectory = "DICOM_Box"

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
iv:Ge/Patient/ShowSpecificSlicesY = 1 -2
iv:Ge/Patient/ShowSpecificSlicesZ = 1 -2

# Overlay implant in a parallel world using Layered Mass Geometry.
s:Ge/Implant/Type        = "TsSphere"
s:Ge/Implant/Parent      = "Patient"
b:Ge/Implant/IsParallel  = "True"
s:Ge/Implant/Material    = "Titanium"
d:Ge/Implant/RMin        = 0. cm
d:Ge/Implant/RMax        = 5.0 cm
d:Ge/Implant/SPhi        = 0.0 deg
d:Ge/Implant/DPhi        = 360. deg
d:Ge/Implant/STheta      = 0 deg
d:Ge/Implant/DTheta      = 180 deg
d:Ge/Implant/TransX      = 0. cm
d:Ge/Implant/TransY      = 0. cm
d:Ge/Implant/TransZ      = 15. cm
d:Ge/Implant/RotX        = 0. deg
d:Ge/Implant/RotY        = 0.0 deg
d:Ge/Implant/RotZ        = 0. deg
s:Ge/Implant/Color       = "yellow"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
sv:Ph/Default/LayeredMassGeometryWorlds = 1 "Implant"

s:So/Example/Type        = "Beam"
s:So/Example/Component    = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy   = 30. MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 5 cm
d:So/Example/BeamPositionCutoffY = 20 cm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = Tf/Num/Value

# Graphics
s:Gr/ViewA/Type          = "OpenGL"

```

(continues on next page)

(continued from previous page)

```

sv:Gr/ViewA/VisibleWorlds = 1 "All"
i:Gr/ViewA/WindowSizeX    = 900
i:Gr/ViewA/WindowSizeY    = 900
d:Gr/ViewA/Theta          = 90 deg
d:Gr/ViewA/Phi            = 0 deg
u:Gr/ViewA/Zoom           = Tf/Zoom/Value
u:Gr/ViewA/TransX         = Tf/Pan/Value

# Graphics options to handle complex voxel geometries
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 2100000
i:Gr/SwitchOGLtoOGLIIfVoxelCountExceeds = 210

# Add zoom in over time
s:Tf/Zoom/Function         = "Step"
dv:Tf/Zoom/Times           = 2 1 2 ms
uv:Tf/Zoom/Values          = 2 1. 2.
d:Tf/Zoom/RepetitionInterval = 360. ms

s:Tf/Num/Function          = "Step"
dv:Tf/Num/Times            = 2 1 2 ms
iv:Tf/Num/Values           = 2 250 1000
d:Tf/Num/RepetitionInterval = 360. ms

s:Tf/Pan/Function          = "Step"
dv:Tf/Pan/Times            = 2 1 2 ms
uv:Tf/Pan/Values           = 2 0. -.1
d:Tf/Pan/RepetitionInterval = 360. ms

d:Tf/TimelineEnd = 2.0 ms
i:Tf/NumberOfSequentialTimes = 2
i:Tf/Verbosity = 1

i:Ts/ShowHistoryCountAtInterval = 5000

b:Ts/PauseBeforeQuit = "False"

```

## 24.7 TwoDicoms.txt

```

# Demonstrates use of a DICOM file.
# You must unzip Abdomen.zip before you run this example.
# Reads in and displays a 3 slice scan of an abdomen.
# The display writes very slowly to the screen.
# This will be improved in an upcoming new release.
# Until then, one useful trick is that if you iconize the graphics window,
# then un-iconize it when drawing is done, the drawing will go much faster.

includeFile = HUtoMaterialSchneider.txt
includeFile = HUtoMaterialSchneider2.txt

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

```

(continues on next page)

(continued from previous page)

```

s:Ge/Patient/Parent           = "World"
s:Ge/Patient/Material         = "G4_WATER"
s:Ge/Patient/Type             = "TsDicomPatient"
d:Ge/Patient/TransX           = 0. m
d:Ge/Patient/TransY           = 0. m
d:Ge/Patient/TransZ           = 0. m
d:Ge/Patient/RotX             = 0. deg
d:Ge/Patient/RotY             = 0. deg
d:Ge/Patient/RotZ             = 0. deg
s:Ge/Patient/DicomDirectory   = "Abdomen"
sv:Ge/Patient/DicomModalityTags = 1 "CT"
b:Ge/Patient/SchneiderUseVariableDensityMaterials = "True"

s:Ge/Patient2/Parent          = "World"
s:Ge/Patient2/Material        = "G4_WATER"
s:Ge/Patient2/Type            = "TsDicomPatient"
d:Ge/Patient2/TransX          = 0. m
d:Ge/Patient2/TransY          = 0. m
d:Ge/Patient2/TransZ          = 1. m
d:Ge/Patient2/RotX            = 0. deg
d:Ge/Patient2/RotY            = 0. deg
d:Ge/Patient2/RotZ            = 0. deg
s:Ge/Patient2/DicomDirectory   = "DICOM_Box"
sv:Ge/Patient2/DicomModalityTags = 1 "CT"
b:Ge/Patient2/SchneiderUseVariableDensityMaterials = "True"

Ts/DumpNonDefaultParameters = "True"

# Restrict which voxels are loaded.
# This will speed things up for purposes of this demonstration,
# but also changes simulation results, as only part of the DICOM
# will be loaded. Take care when using this feature in your research.
#i:Ge/Patient/RestrictVoxelsXMin = 200
#i:Ge/Patient/RestrictVoxelsYMin = 200
#i:Ge/Patient/RestrictVoxelsXMax = 400
#i:Ge/Patient/RestrictVoxelsYMax = 400

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
#iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
#iv:Ge/Patient/ShowSpecificSlicesY = 1 -2
iv:Ge/Patient/ShowSpecificSlicesZ = 1 -1

# Simplify physics for faster start of this demonstration
#sv:Ph/Default/Modules = 1 "transportation_only"
sv:Ph/Default/Modules = 1 "g4em-standard_opt4"

s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "proton"
d:So/Example/BeamEnergy        = 169.23 MeV
u:So/Example/BeamEnergySpread  = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"

```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 0
Ts/ShowHistoryCountAtInterval = 100
#Ts/NumberOfThreads = 4

#s:Sc/DoseAtPhantom/Quantity          = "DoseToMedium"
s:Sc/DoseAtPhantom/Component          = "Patient"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/DoseAtPhantom/Report = 8 "Sum" "Mean" "Count_in_bin" "Second_Moment" "Variance"
↪ "Standard_Deviation" "Min" "Max"

s:Gr/ViewA/Type                      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta                = 70 deg
d:Gr/ViewA/Phi                   = 10 deg
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 100000000

Ts/PauseBeforeInit = "F"
Ts/PauseBeforeSequence = "F"
Ts/PauseBeforeQuit = "True"

```

## 24.8 ViewAbdomen\_rtdose.txt

```

# Demonstrates use of a DICOM file.
# You must unzip Abdomen.zip before you run this example.
# Reads in and displays a 3 slice scan of an abdomen.
# The display writes very slowly to the screen.
# This will be improved in an upcoming new release.
# Until then, one useful trick is that if you iconize the graphics window,
# then un-iconize it when drawing is done, the drawing will go much faster.

# This example has three blocks of component definition to mimic beam delivery to a
↪ patient in a treatment room.

# 1. Coordination definition
# 2. Plan information: Gantry angle, couch angle, Plan Isocenter
# 2. DICOM Patient:
# 3. Beam setup
# 4. Dosegrid setup

includeFile = HUtoMaterialSchneider.txt

#####
# Coordination

```

(continues on next page)

(continued from previous page)

```
#####
# World: Fixed coordination in IEC
s:Ge/World/Material = "G4_AIR"
d:Ge/World/HLX      = 0.6 m
d:Ge/World/HLY      = 0.6 m
d:Ge/World/HLZ      = 0.6 m
b:Ge/World/Invisible = "T"
# IEC_G: Gantry
s:Ge/IEC_G/Parent   = "World"
s:Ge/IEC_G/Type     = "Group"
d:Ge/IEC_G/RotX     = 0. deg
d:Ge/IEC_G/RotY     = -1.0 * Ge/Gantry_Angle deg
d:Ge/IEC_G/RotZ     = 0. deg
d:Ge/IEC_G/TransX   = 0. m
d:Ge/IEC_G/TransY   = 0. m
d:Ge/IEC_G/TransZ   = 0. m

# IEC_S: Patient support
s:Ge/IEC_S/Parent   = "World"
s:Ge/IEC_S/Type     = "Group"
d:Ge/IEC_S/RotX     = 0. deg
d:Ge/IEC_S/RotY     = 0. deg
d:Ge/IEC_S/RotZ     = Ge/Couch_Angle deg
d:Ge/IEC_S/TransX   = 0. m
d:Ge/IEC_S/TransY   = 0. m
d:Ge/IEC_S/TransZ   = 0. m

# DICOM_to_IEC
s:Ge/DICOM_to_IEC/Type = "Group"
s:Ge/DICOM_to_IEC/Parent = "IEC_S"
d:Ge/DICOM_to_IEC/TransX = 0 mm
d:Ge/DICOM_to_IEC/TransY = 0 mm
d:Ge/DICOM_to_IEC/TransZ = 0 mm
d:Ge/DICOM_to_IEC/RotX   = 90.0 deg
d:Ge/DICOM_to_IEC/RotY   = 0.0 deg
d:Ge/DICOM_to_IEC/RotZ   = 0.0 deg

#####
# Geometry parameters from plan
#####
#for plan
d:Ge/Gantry_Angle = 170.0 deg
d:Ge/Couch_Angle  = 0.0 deg

#Geant4: Abdomen CT's isocenter
d:Ge/IsoCenterX   = 15.3125 mm
d:Ge/IsoCenterY   = 206.7656 mm
d:Ge/IsoCenterZ   = -92.5 mm

#Beam is on Gantry coordination
Ge/BeamPosition/Parent = "IEC_G"
Ge/BeamPosition/TransZ = 25.0 cm

#####
# Patient in DICOM
#####
```

(continues on next page)

(continued from previous page)

```

s:Ge/Patient/Parent      = "DICOM_to_IEC"
s:Ge/Patient/Type        = "TsDicomPatient"
s:Ge/Patient/Material     = "G4_WATER"
d:Ge/Patient/RotX        = 0.0 deg
d:Ge/Patient/RotY        = 0.0 deg
d:Ge/Patient/RotZ        = 0.0 deg
s:Ge/Patient/HUtoMaterialConversionMethod = "Schneider"

# Transient parameters
# DicomOrigin: a vector to CT cube's center
# note: patient positioning is properly done when the simulation starts
dc:Ge/Patient/DicomOriginX = 0.0 mm
dc:Ge/Patient/DicomOriginY = 0.0 mm
dc:Ge/Patient/DicomOriginZ = 0.0 mm

d:Ge/Patient/TransX      = Ge/Patient/DicomOriginX - Ge/IsoCenterX mm
d:Ge/Patient/TransY      = Ge/Patient/DicomOriginY - Ge/IsoCenterY mm
d:Ge/Patient/TransZ      = Ge/Patient/DicomOriginZ - Ge/IsoCenterZ mm

s:Ge/Patient/DicomDirectory = "Abdomen"

# Create a parallel grid for scoring. "Patient/RTDoseGrid"
# you have three options to create RTDoseGrid
# option 1 : You can create same size from DICOM-RT Dose
# s:Ge/Patient/CloneRTDoseGridFrom = Ge/Patient/DicomDirectory + "/rtdose.dcm"

# option 2 : You can redefine size from DICOM-RT Dose but keep same position
# s:Ge/Patient/CloneRTDoseGridFrom = Ge/Patient/DicomDirectory + "/rtdose.dcm"
# dv:Ge/Patient/CloneRTDoseGridSize = 3 5 5 8 mm

# option 3 : If you don't have RT-Dose, simply specify the pixel size.
dv:Ge/Patient/CloneRTDoseGridSize = 3 5 5 5 mm

b:Ge/Patient/IgnoreInconsistentFrameOfReferenceUID = "True"

sv:Ge/Patient/DicomModalityTags = 1 "CT"

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
#iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
#iv:Ge/Patient/ShowSpecificSlicesY = 1 -2
iv:Ge/Patient/ShowSpecificSlicesZ = 1 -1

#####
##---- Beam setup ----#
#####

s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy    = 60.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"

```

(continues on next page)



(continued from previous page)

```

d:So/Example/BeamPositionCutoffX      = 1. cm
d:So/Example/BeamPositionCutoffY      = 1. cm
d:So/Example/BeamPositionSpreadX      = 0.2 cm
d:So/Example/BeamPositionSpreadY      = 0.2 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 10

#####
# Dose calculation
#####
# scoring dose on DoseGrid from RTDOSE
# "Patient/RTDose"

s:Sc/DoseOnRTGrid/Quantity             = "DoseToMedium"
s:Sc/DoseOnRTGrid/Component            = "Patient/RTDoseGrid"
b:Sc/DoseOnRTGrid/OutputToConsole      = "F"
s:Sc/DoseOnRTGrid/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseOnRTGrid/OutputType           = "DICOM"
b:Sc/DoseOnRTGrid/DICOMOutput32BitsPerPixel = "True"

#####
# Visualization
#####

s:Gr/ViewA/Type                       = "OpenGL"
i:Gr/ViewA/WindowSizeX                = 900
i:Gr/ViewA/WindowSizeY                = 900
d:Gr/ViewA/Theta                      = 265 deg
d:Gr/ViewA/Phi                       = -90 deg
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 1000000

s:Gr/ViewA/AxesComponent              = "World"
b:Gr/ViewA/IncludeAxes                = "true"
d:Gr/ViewA/AxesSize                   = 30.0 cm
u:Gr/ViewA/Zoom                       = 1.5

s:Gr/ViewA/Projection                 = "Perspective"
d:Gr/ViewA/PerspectiveAngle           = 10 deg

i:Ts/NumberOfThreads                  = 1

```

## 24.9 ViewAbdomen.txt

```

# Demonstrates use of a DICOM file.
# You must unzip Abdomen.zip before you run this example.
# Reads in and displays a 3 slice scan of an abdomen.
# The display writes very slowly to the screen.
# This will be improved in an upcoming new release.
# Until then, one useful trick is that if you iconize the graphics window,
# then un-iconize it when drawing is done, the drawing will go much faster.

```

(continues on next page)

(continued from previous page)

```

includeFile = HUtoMaterialSchneider.txt

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/Patient/Parent      = "World"
s:Ge/Patient/Material    = "G4_WATER"
s:Ge/Patient/Type        = "TsDicomPatient"
d:Ge/Patient/TransX      = 0. m
d:Ge/Patient/TransY      = 0. m
d:Ge/Patient/TransZ      = 0. m
d:Ge/Patient/RotX        = 0. deg
d:Ge/Patient/RotY        = 0. deg
d:Ge/Patient/RotZ        = 0. deg
s:Ge/Patient/DicomDirectory = "Abdomen"
sv:Ge/Patient/DicomModalityTags = 1 "CT"

# Restrict which voxels are loaded.
# This will speed things up for purposes of this demonstration,
# but also changes simulation results, as only part of the DICOM
# will be loaded. Take care when using this feature in your research.
i:Ge/Patient/RestrictVoxelsXMin = 200
i:Ge/Patient/RestrictVoxelsYMin = 200
i:Ge/Patient/RestrictVoxelsXMax = 400
i:Ge/Patient/RestrictVoxelsYMax = 400

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
#iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
#iv:Ge/Patient/ShowSpecificSlicesY = 1 -2
iv:Ge/Patient/ShowSpecificSlicesZ = 1 -1

# Simplify physics for faster start of this demonstration
sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy    = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad

```

(continues on next page)

(continued from previous page)

```
i:So/Example/NumberOfHistoriesInRun = 10

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 70 deg
d:Gr/ViewA/Phi = 10 deg
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 1000000
```

## 24.10 XCAT\_Materials.txt

```
# XCAT Material Definitions
# (any spaces in material name from XCAT log file are replaced with underscores)

# Temporarily, all are actually water, just with different colors

sv:Ma/XCAT_Air/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Air/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Air/Density = 1.0 g/cm3
s:Ma/XCAT_Air/DefaultColor = "TransparentBlack"
iv:Gr/Color/TransparentBlack = 4 0 0 0 0

sv:Ma/XCAT_Body_(water)/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Body_(water)/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Body_(water)/Density = 1.0 g/cm3
d:Ma/XCAT_Body_(water)/MeanExcitationEnergy = 75.0 eV
s:Ma/XCAT_Body_(water)/DefaultColor = "blue"

sv:Ma/XCAT_Muscle/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Muscle/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Muscle/Density = 1.0 g/cm3
s:Ma/XCAT_Muscle/DefaultColor = "orange"

sv:Ma/XCAT_Adipose_(fat)/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Adipose_(fat)/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Adipose_(fat)/Density = 1.0 g/cm3
s:Ma/XCAT_Adipose_(fat)/DefaultColor = "yellow"

sv:Ma/XCAT_Lung/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Lung/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Lung/Density = 1.0 g/cm3
s:Ma/XCAT_Lung/DefaultColor = "skyblue"

sv:Ma/XCAT_Spine_Bone/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Spine_Bone/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Spine_Bone/Density = 1.0 g/cm3
s:Ma/XCAT_Spine_Bone/DefaultColor = "white"

sv:Ma/XCAT_Rib_Bone/Components = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Rib_Bone/Fractions = 2 0.111894 0.888106
d:Ma/XCAT_Rib_Bone/Density = 1.0 g/cm3
s:Ma/XCAT_Rib_Bone/DefaultColor = "grey"

sv:Ma/XCAT_Blood/Components = 2 "Hydrogen" "Oxygen"
```

(continues on next page)

(continued from previous page)

```

uv:Ma/XCAT_Blood/Fractions           = 2 0.111894 0.888106
d:Ma/XCAT_Blood/Density              = 1.0 g/cm3
s:Ma/XCAT_Blood/DefaultColor         = "blue"

sv:Ma/XCAT_Heart/Components          = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Heart/Fractions           = 2 0.111894 0.888106
d:Ma/XCAT_Heart/Density              = 1.0 g/cm3
s:Ma/XCAT_Heart/DefaultColor         = "red"

sv:Ma/XCAT_Kidney/Components          = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Kidney/Fractions           = 2 0.111894 0.888106
d:Ma/XCAT_Kidney/Density              = 1.0 g/cm3
s:Ma/XCAT_Kidney/DefaultColor         = "red"

sv:Ma/XCAT_Liver/Components          = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Liver/Fractions           = 2 0.111894 0.888106
d:Ma/XCAT_Liver/Density              = 1.0 g/cm3
s:Ma/XCAT_Liver/DefaultColor         = "red"

sv:Ma/XCAT_Lymph/Components          = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Lymph/Fractions           = 2 0.111894 0.888106
d:Ma/XCAT_Lymph/Density              = 1.0 g/cm3
s:Ma/XCAT_Lymph/DefaultColor         = "green"

sv:Ma/XCAT_Pancreas/Components        = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Pancreas/Fractions         = 2 0.111894 0.888106
d:Ma/XCAT_Pancreas/Density           = 1.0 g/cm3
s:Ma/XCAT_Pancreas/DefaultColor      = "red"

sv:Ma/XCAT_Spleen/Components          = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Spleen/Fractions           = 2 0.111894 0.888106
d:Ma/XCAT_Spleen/Density             = 1.0 g/cm3
s:Ma/XCAT_Spleen/DefaultColor        = "red"

sv:Ma/XCAT_Intestine/Components       = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Intestine/Fractions        = 2 0.111894 0.888106
d:Ma/XCAT_Intestine/Density           = 1.0 g/cm3
s:Ma/XCAT_Intestine/DefaultColor     = "yellow"

sv:Ma/XCAT_Skull/Components           = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Skull/Fractions            = 2 0.111894 0.888106
d:Ma/XCAT_Skull/Density              = 1.0 g/cm3
s:Ma/XCAT_Skull/DefaultColor         = "grey"

sv:Ma/XCAT_Cartilage/Components       = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Cartilage/Fractions        = 2 0.111894 0.888106
d:Ma/XCAT_Cartilage/Density           = 1.0 g/cm3
s:Ma/XCAT_Cartilage/DefaultColor     = "grey"

sv:Ma/XCAT_Brain/Components           = 2 "Hydrogen" "Oxygen"
uv:Ma/XCAT_Brain/Fractions            = 2 0.111894 0.888106
d:Ma/XCAT_Brain/Density              = 1.0 g/cm3
s:Ma/XCAT_Brain/DefaultColor         = "grey"

```

## 24.11 XCAT.txt

```
# Demonstrates scoring in an XCAT file.

s:Ge/Patient/Type = "TsImageCube"
s:Ge/Patient/InputDirectory = "./"
s:Ge/Patient/InputFile = "XCAT_FullMouse_86x86x161_atn_1.bin"
s:Ge/Patient/MetaDataFile = "XCAT_FullMouse_86x86x161_atn_1.log"
s:Ge/Patient/ImagingToMaterialConverter = "XCAT_Attenuation" # "XCAT_Activity"
includeFile = XCAT_Materials.txt

# These parameters are only used if no MetaDataFile is provided.
i:Ge/Patient/NumberOfVoxelsX = 86
i:Ge/Patient/NumberOfVoxelsY = 86
i:Ge/Patient/NumberOfVoxelsZ = 161
d:Ge/Patient/VoxelSizeX = .5 mm
d:Ge/Patient/VoxelSizeY = .5 mm
dv:Ge/Patient/VoxelSizeZ = 1 .5 mm

s:Ge/Patient/DataType = "FLOAT"
u:Ge/Patient/AttenuationForMaterial_XCAT_Air = 0.
u:Ge/Patient/AttenuationForMaterial_XCAT_Body_(water) = 203.8293
u:Ge/Patient/AttenuationForMaterial_XCAT_Muscle = 195.2515
u:Ge/Patient/AttenuationForMaterial_XCAT_Adipose_(fat) = 114.9825
u:Ge/Patient/AttenuationForMaterial_XCAT_Lung = 57.5347
u:Ge/Patient/AttenuationForMaterial_XCAT_Spine_Bone = 225.7159
u:Ge/Patient/AttenuationForMaterial_XCAT_Rib_Bone = 283.3633
u:Ge/Patient/AttenuationForMaterial_XCAT_Blood = 201.7046
u:Ge/Patient/AttenuationForMaterial_XCAT_Heart = 195.6546
u:Ge/Patient/AttenuationForMaterial_XCAT_Kidney = 196.5382
u:Ge/Patient/AttenuationForMaterial_XCAT_Liver = 197.4710
u:Ge/Patient/AttenuationForMaterial_XCAT_Lymph = 204.0590
u:Ge/Patient/AttenuationForMaterial_XCAT_Pancreas = 190.0039
u:Ge/Patient/AttenuationForMaterial_XCAT_Spleen = 200.6941
u:Ge/Patient/AttenuationForMaterial_XCAT_Intestine = 195.1612
u:Ge/Patient/AttenuationForMaterial_XCAT_Skull = 247.5866
u:Ge/Patient/AttenuationForMaterial_XCAT_Cartilage = 208.0629
u:Ge/Patient/AttenuationForMaterial_XCAT_Brain = 192.2248

s:Ge/Patient/Parent = "World"
d:Ge/Patient/TransX = 0. m
d:Ge/Patient/TransY = 0. m
d:Ge/Patient/TransZ = 0. m
d:Ge/Patient/RotX = 0. deg
d:Ge/Patient/RotY = 0. deg
d:Ge/Patient/RotZ = 0. deg

# Specify which slices to show.
# Comment this out or set to zero to show all slices.
# Set to -1 to show only center slice.
# Set to -2 to show first, center and last slice.
#iv:Ge/Patient/ShowSpecificSlicesX = 1 -2
#iv:Ge/Patient/ShowSpecificSlicesY = 1 -2
iv:Ge/Patient/ShowSpecificSlicesZ = 9 1 21 41 61 81 101 121 141 161

s:Ge/World/Material = "G4_AIR"
d:Ge/World/HLX = .5 m
```

(continues on next page)

(continued from previous page)

```
d:Ge/World/HLY           = .5 m
d:Ge/World/HLZ           = .5 m
b:Ge/World/Invisible     = "TRUE"

s:So/Example/Type        = "Beam"
s:So/Example/Component   = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy   = 70. MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 5. mm
d:So/Example/BeamPositionCutoffY = 5. mm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 10

s:Gr/ViewA/Type          = "OpenGL"
i:Gr/ViewA/WindowSizeX   = 900
i:Gr/ViewA/WindowSizeY   = 900
d:Gr/ViewA/Theta         = 60 deg
d:Gr/ViewA/Phi           = 10 deg

# Extra graphics options to handle complex voxel geometries
i:Gr/ShowOnlyOutlineIfVoxelCountExceeds = 210000000

i:Ts/ShowHistoryCountAtInterval = 100

b:Ts/PauseBeforeQuit = "True"
```

## 25.1 MultiRun\_Write\_ASCII.txt

```
# Writes phase space separately for several time intervals

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent   = "World"
s:Ge/VacFilm/Material = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg
d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:Sc/PhaseSpaceAtVacFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface        = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType     = "ASCII" # ASCII or Binary
s:Sc/PhaseSpaceAtVacFilm/OutputFile     = "ASCIIOutput"
i:Sc/PhaseSpaceAtVacFilm/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesGoing = "In"
b:Sc/PhaseSpaceAtVacFilm/IncludeRunID    = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeEventID  = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTrackID  = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTime     = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeSeed     = "True"
#sv:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesNamed = 1 "Proton"
```

(continues on next page)

(continued from previous page)

```

s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite" # Exit, Overwrite,
↳or Increment
b:Sc/PhaseSpaceAtVacFilm/OutputAfterRun             = "True"

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

d:Tf/TimelineEnd = 100.0 ms
i:Tf/NumberOfSequentialTimes = 4

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 25.2 ReadASCII.txt

```

# Read phase space in TOPAS ASCII form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type = "TsBox"
s:Ge/VacFilm/Parent = "World"
s:Ge/VacFilm/Material = "G4_WATER"
d:Ge/VacFilm/HLX = 50.0 cm
d:Ge/VacFilm/HLY = 50.0 cm
d:Ge/VacFilm/HLZ = 1.0 cm
d:Ge/VacFilm/TransX = 0. cm
d:Ge/VacFilm/TransY = 0. cm
d:Ge/VacFilm/TransZ = 0. cm
d:Ge/VacFilm/RotX = 0. deg
d:Ge/VacFilm/RotY = 0. deg

```

(continues on next page)



(continued from previous page)

```

d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:So/Example/Type      = "PhaseSpace"
s:So/Example/PhaseSpaceFileName = "ASCIIOutput"
s:So/Example/Component = "World"
#i:So/Example/PhaseSpaceMultipleUse = 2
#i:So/Example/PhaseSpaceBufferSize = 100
b:So/Example/PhaseSpaceIncludeEmptyHistories = "True"

# Graphics
s:Gr/ViewA/Type        = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta       = 55 deg
d:Gr/ViewA/Phi         = 20 deg
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.3 ReadBinary.txt

```

# Read phase space in TOPAS Binary form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg
d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:So/Example/Type      = "PhaseSpace"
s:So/Example/PhaseSpaceFileName = "BinaryOutput"
s:So/Example/Component = "World"
#i:So/Example/PhaseSpaceMultipleUse = 2
#i:So/Example/PhaseSpaceBufferSize = 100
b:So/Example/PhaseSpaceIncludeEmptyHistories = "True"

# Graphics
s:Gr/ViewA/Type        = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900

```

(continues on next page)

(continued from previous page)

```

i:Gr/ViewA/WindowSizeY      = 900
d:Gr/ViewA/Theta            = 55 deg
d:Gr/ViewA/Phi              = 20 deg
s:Gr/ViewA/Projection        = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom              = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.4 ReadLimited.txt

```

# Write phase space in Limited form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg
d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:So/Example/Type      = "PhaseSpace"
s:So/Example/PhaseSpaceFileName = "LimitedOutput"
s:So/Example/Component = "World"
#i:So/Example/PhaseSpaceMultipleUse = 2
#i:So/Example/PhaseSpaceBufferSize = 100
b:So/Example/PhaseSpaceIncludeEmptyHistories = "True"

# Graphics
s:Gr/ViewA/Type        = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta       = 55 deg
d:Gr/ViewA/Phi         = 20 deg
s:Gr/ViewA/Projection   = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom         = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.5 WriteASCII.txt

```
# Write phase space in TOPAS ASCII form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg
d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:Sc/PhaseSpaceAtVacFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface        = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType     = "ASCII" # ASCII, Binary,
↳ Limited or ROOT
s:Sc/PhaseSpaceAtVacFilm/OutputFile      = "ASCIIOutput"
i:Sc/PhaseSpaceAtVacFilm/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesGoing = "In"
b:Sc/PhaseSpaceAtVacFilm/IncludeTOPASTime = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTimeOfFlight = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeRunID      = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeEventID    = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTrackID    = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeParentID   = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeCreatorProcess = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeVertexInfo = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeSeed       = "True"
#sv:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesNamed = 1 "Proton"
s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi         = 20 deg
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 100

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.6 WriteBinary.txt

```

# Write phase space in TOPAS Binary form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg
d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:Sc/PhaseSpaceAtVacFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface        = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType     = "Binary" # ASCII, Binary, ↵
↳ Limited or ROOT
s:Sc/PhaseSpaceAtVacFilm/OutputFile      = "BinaryOutput"
i:Sc/PhaseSpaceAtVacFilm/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesGoing = "In"
b:Sc/PhaseSpaceAtVacFilm/IncludeTOPASTime = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTimeOfFlight = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeRunID      = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeEventID    = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTrackID    = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeParentID   = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeVertexInfo = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeSeed       = "True"
#sv:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesNamed = 1 "Proton"
s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type = "OpenGL"

```

(continues on next page)

(continued from previous page)

```

i:Gr/ViewA/WindowSizeX      = 900
i:Gr/ViewA/WindowSizeY      = 900
d:Gr/ViewA/Theta            = 55 deg
d:Gr/ViewA/Phi              = 20 deg
s:Gr/ViewA/Projection        = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom              = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type            = "Beam"
s:So/Example/Component       = "BeamPosition"
s:So/Example/BeamParticle    = "proton"
d:So/Example/BeamEnergy      = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.7 WritelonsASCII.txt

```

# Write ASCII phase space with ion beam

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "Vacuum"
d:Ge/VacFilm/HLX        = 50.0 cm
d:Ge/VacFilm/HLY        = 50.0 cm
d:Ge/VacFilm/HLZ        = 1.0 cm
d:Ge/VacFilm/TransX     = 0. cm
d:Ge/VacFilm/TransY     = 0. cm
d:Ge/VacFilm/TransZ     = 0. cm
d:Ge/VacFilm/RotX       = 0. deg
d:Ge/VacFilm/RotY       = 0. deg
d:Ge/VacFilm/RotZ       = 0. deg
s:Ge/VacFilm/Color      = "skyblue"

s:Sc/PhaseSpaceAtVacFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface       = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType    = "ASCII" # ASCII, Binary,
↳ Limited or ROOT

```

(continues on next page)

(continued from previous page)

```

s:Sc/PhaseSpaceAtVacFilm/OutputFile           = "ASCIIOutput"
i:Sc/PhaseSpaceAtVacFilm/OutputBufferSize      = 1000
sv:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesNamed = 1 "GenericIon(3,7)"
s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type                                = "OpenGL"
i:Gr/ViewA/WindowSizeX                         = 900
i:Gr/ViewA/WindowSizeY                         = 900
d:Gr/ViewA/Theta                              = 55 deg
d:Gr/ViewA/Phi                                = 20 deg
s:Gr/ViewA/Projection                          = "Perspective"
d:Gr/ViewA/PerspectiveAngle                   = 30 deg
u:Gr/ViewA/Zoom                               = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories    = "True"

s:Ph/Default/Type = "QGSP_BIC_HP"

s:So/Example/Type                                = "Beam"
s:So/Example/Component                          = "BeamPosition"
s:So/Example/BeamParticle                      = "GenericIon(3,7)"
d:So/Example/BeamEnergy                       = 400 MeV
u:So/Example/BeamEnergySpread                  = 0.757504
s:So/Example/BeamPositionDistribution           = "Gaussian"
s:So/Example/BeamPositionCutoffShape           = "Ellipse"
d:So/Example/BeamPositionCutoffX               = 10. cm
d:So/Example/BeamPositionCutoffY              = 10. cm
d:So/Example/BeamPositionSpreadX               = 0.65 cm
d:So/Example/BeamPositionSpreadY              = 0.65 cm
s:So/Example/BeamAngularDistribution           = "Gaussian"
d:So/Example/BeamAngularCutoffX               = 90. deg
d:So/Example/BeamAngularCutoffY              = 90. deg
d:So/Example/BeamAngularSpreadX               = 0.0032 rad
d:So/Example/BeamAngularSpreadY              = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun            = 4

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.8 WriteLimited.txt

```

# Write phase space in Limited form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color      = "skyblue"

s:Sc/PhaseSpaceAtVacFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface        = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType     = "Limited" # ASCII, Binary,
↳ Limited or ROOT
s:Sc/PhaseSpaceAtVacFilm/OutputFile      = "LimitedOutput"
i:Sc/PhaseSpaceAtVacFilm/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesGoing = "In"
b:Sc/PhaseSpaceAtVacFilm/IncludeTOPASTime = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTimeOfFlight = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeRunID = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeEventID = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTrackID = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeParentID = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeVertexInfo = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeSeed = "True"
#sv:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesNamed = 1 "Proton"
s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi         = 20 deg
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom         = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 25.9 WriteROOT.txt

```
# Write phase space in TOPAS ASCII form

b:Ge/World/Invisible = "TRUE"

s:Ge/VacFilm/Type      = "TsBox"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/HLX       = 50.0 cm
d:Ge/VacFilm/HLY       = 50.0 cm
d:Ge/VacFilm/HLZ       = 1.0 cm
d:Ge/VacFilm/TransX    = 0. cm
d:Ge/VacFilm/TransY    = 0. cm
d:Ge/VacFilm/TransZ    = 0. cm
d:Ge/VacFilm/RotX      = 0. deg
d:Ge/VacFilm/RotY      = 0. deg
d:Ge/VacFilm/RotZ      = 0. deg
s:Ge/VacFilm/Color     = "skyblue"

s:Sc/PhaseSpaceAtVacFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtVacFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtVacFilm/Surface        = "VacFilm/ZMinusSurface"
s:Sc/PhaseSpaceAtVacFilm/OutputType     = "ROOT" # ASCII, Binary, Limited or ROOT
s:Sc/PhaseSpaceAtVacFilm/OutputFile      = "ROOTOutput"
i:Sc/PhaseSpaceAtVacFilm/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesGoing = "In"
b:Sc/PhaseSpaceAtVacFilm/IncludeTOPASTime = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTimeOfFlight = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeRunID       = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeEventID     = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeTrackID     = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeParentID    = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeVertexInfo  = "True"
b:Sc/PhaseSpaceAtVacFilm/IncludeSeed        = "True"
#sv:Sc/PhaseSpaceAtVacFilm/OnlyIncludeParticlesNamed = 1 "Proton"
s:Sc/PhaseSpaceAtVacFilm/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi        = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
```

(continues on next page)



(continued from previous page)

```
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 100

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"
```



## 26.1 ChargeInSphere.txt

```
# Score charge in a sphere

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/Sphere/Type      = "TsSphere"
s:Ge/Sphere/Parent    = "World"
s:Ge/Sphere/Material  = "Air"
d:Ge/Sphere/RMax      = 55.0 cm
s:Ge/Sphere/DrawingStyle = "FullWireFrame"

s:Sc/Charge/Quantity      = "Charge"
s:Sc/Charge/Component     = "Sphere"
b:Sc/Charge/OutputToConsole = "TRUE"
s:Sc/Charge/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/Charge/OnlyIncludeParticlesNamed = 1 "proton"
#s:Sc/Charge/OnlyIncludeParticlesOfGeneration = "secondary"

s:Gr/ViewA/Type = "OpenGL"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 500

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 26.2 Complex.txt

```

# A complex example showing lots of different scorers and options.

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Film/Type       = "TsBox"
s:Ge/Film/Parent     = "World"
s:Ge/Film/Material   = "G4_WATER"
d:Ge/Film/HLX        = 50.0 cm
d:Ge/Film/HLY        = 50.0 cm
d:Ge/Film/HLZ        = 1.0 cm
d:Ge/Film/TransX     = 0. cm
d:Ge/Film/TransY     = 0. cm
d:Ge/Film/TransZ     = 0. cm
d:Ge/Film/RotX       = 0. deg
d:Ge/Film/RotY       = 0. deg
d:Ge/Film/RotZ       = 0. deg
s:Ge/Film/Color      = "skyblue"

s:Ge/Phantom1/Type   = "TsBox"
s:Ge/Phantom1/Parent = "World"
s:Ge/Phantom1/Material = "G4_WATER"
d:Ge/Phantom1/HLX    = 30.0 cm
d:Ge/Phantom1/HLY    = 30.0 cm
d:Ge/Phantom1/HLZ    = 1.1 cm
d:Ge/Phantom1/TransZ = -30. cm

s:Ge/Phantom2/Type   = "TsBox"
s:Ge/Phantom2/Parent = "World"
s:Ge/Phantom2/Material = "G4_WATER"
d:Ge/Phantom2/HLX    = 10.0 cm
d:Ge/Phantom2/HLY    = 10.0 cm
d:Ge/Phantom2/HLZ    = 10.0 cm
d:Ge/Phantom2/TransZ = -50. cm
s:Ge/Phantom2/Color  = "red"
i:Ge/Phantom2/XBins  = 2
i:Ge/Phantom2/YBins  = 2

```

(continues on next page)

(continued from previous page)

```

i:Ge/Phantom2/ZBins      = 1

s:Sc/PhaseSpaceAtFilm/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtFilm/OutputToConsole = "True"
s:Sc/PhaseSpaceAtFilm/Surface        = "Film/ZMinusSurface"
s:Sc/PhaseSpaceAtFilm/OutputType     = "ASCII" # ASCII or Binary
i:Sc/PhaseSpaceAtFilm/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtFilm/OnlyIncludeParticlesGoing = "In"
b:Sc/PhaseSpaceAtFilm/IncludeRunID    = "True"
b:Sc/PhaseSpaceAtFilm/IncludeEventID  = "True"
b:Sc/PhaseSpaceAtFilm/IncludeTrackID  = "True"
b:Sc/PhaseSpaceAtFilm/IncludeTime     = "True"
#sv:Sc/PhaseSpaceAtFilm/OnlyIncludeParticlesNamed = 1 "Proton"
s:Sc/PhaseSpaceAtFilm/IfOutputFileAlreadyExists = "Increment" # Exit, Overwrite or
↳Increment

s:Sc/PhaseSpaceAtFilmb/Quantity      = "PhaseSpace"
b:Sc/PhaseSpaceAtFilmb/OutputToConsole = "True"
s:Sc/PhaseSpaceAtFilmb/Surface        = "Film/ZMinusSurface"
s:Sc/PhaseSpaceAtFilmb/OutputType     = "binary" # ASCII or Binary
i:Sc/PhaseSpaceAtFilmb/OutputBufferSize = 1000
#s:Sc/PhaseSpaceAtFilmb/OnlyIncludeParticlesGoing = "In"
#sv:Sc/PhaseSpaceAtFilmb/OnlyIncludeParticlesNamed = 1 "Proton"

s:Sc/DoseAtPhantom1/Quantity      = "DoseToMedium"
s:Sc/DoseAtPhantom1/Component     = "Phantom1"
s:Sc/DoseAtPhantom1/OutputFile    = "DoseAtPhantom1"
s:Sc/DoseAtPhantom1/OutputType    = "csv"
b:Sc/DoseAtPhantom1/OutputToConsole = "TRUE"
b:Sc/DoseAtPhantom1/Visualize     = "TRUE"
s:Sc/DoseAtPhantom1/IfOutputFileAlreadyExists = "Increment" # Exit, Overwrite or
↳Increment
sv:Sc/DoseAtPhantom1/Report = 8 "Sum" "Mean" "Count_in_bin" "Second_Moment" "Variance
↳" "Standard_Deviation" "Min" "Max"
i:Sc/DoseAtPhantom1/ZBins = 10

s:Sc/DoseAtPhantom1b/Quantity      = "DoseToMedium"
s:Sc/DoseAtPhantom1b/Component     = "Phantom1"
s:Sc/DoseAtPhantom1b/OutputFile    = "DoseAtPhantom1b"
s:Sc/DoseAtPhantom1b/OutputType    = "binary"
b:Sc/DoseAtPhantom1b/OutputToConsole = "TRUE"
b:Sc/DoseAtPhantom1b/Visualize     = "TRUE"
s:Sc/DoseAtPhantom1b/IfOutputFileAlreadyExists = "Increment" # Exit, Overwrite or
↳Increment
sv:Sc/DoseAtPhantom1b/Report = 8 "Sum" "Mean" "Count_in_bin" "Second_Moment" "Variance
↳" "Standard_Deviation" "Min" "Max"

s:Sc/EnergyDepAtPhantom1/Quantity      = "EnergyDeposit"
s:Sc/EnergyDepAtPhantom1/Component     = "Phantom1"
s:Sc/EnergyDepAtPhantom1/OutputType    = "csv"
b:Sc/EnergyDepAtPhantom1/OutputToConsole = "TRUE"
b:Sc/EnergyDepAtPhantom1/Visualize     = "TRUE"
s:Sc/EnergyDepAtPhantom1/IfOutputFileAlreadyExists = "Increment" # Exit, Overwrite or
↳Increment

s:Sc/FluenceAtPhantom1/Quantity      = "Fluence"
s:Sc/FluenceAtPhantom1/Component     = "Phantom1"

```

(continues on next page)

(continued from previous page)

```

s:Sc/FluenceAtPhantom1/OutputFile      = "FluenceAtPhantom1"
s:Sc/FluenceAtPhantom1/OutputType      = "csv"
b:Sc/FluenceAtPhantom1/OutputToConsole = "TRUE"
b:Sc/FluenceAtPhantom1/Visualize       = "TRUE"

s:Sc/EnergyFluenceAtPhantom1/Quantity  = "EnergyFluence"
s:Sc/EnergyFluenceAtPhantom1/Component = "Phantom1"
s:Sc/EnergyFluenceAtPhantom1/OutputFile = "EnergyFluenceAtPhantom1"
s:Sc/EnergyFluenceAtPhantom1/OutputType = "csv"
b:Sc/EnergyFluenceAtPhantom1/OutputToConsole = "TRUE"
b:Sc/EnergyFluenceAtPhantom1/Visualize     = "TRUE"
sv:Sc/EnergyFluenceAtPhantom1/OnlyIncludeParticlesNamed = 1 "proton"

s:Sc/eDoseAtPhantom1/Quantity          = "DoseToMedium"
s:Sc/eDoseAtPhantom1/Component         = "Phantom1"
sv:Sc/eDoseAtPhantom1/OnlyIncludeParticlesNamed = 1 "e-"
#sv:Sc/eDoseAtPhantom1/OnlyIncludeParticlesCharged = 1 "negative" # zero or
↳more of positive, negative or neutral
#d:Sc/eDoseAtPhantom1/OnlyIncludeIfIncidentParticleKEAbove = 200. keV # minimum energy
#d:Sc/eDoseAtPhantom1/OnlyIncludeIfIncidentParticleKEBelow = 250. keV # minimum energy
#sv:Sc/eDoseAtPhantom1/OnlyIncludeParticlesFromComponent = 1 "Phantom1"
#sv:Sc/eDoseAtPhantom1/OnlyIncludeParticlesFromProcess = 1 "hIoni"
#b:Sc/eDoseAtPhantom1/InvertFilter = "TRUE"
s:Sc/eDoseAtPhantom1/OutputFile      = "eDoseAtPhantom1"
s:Sc/eDoseAtPhantom1/OutputType      = "csv"
b:Sc/eDoseAtPhantom1/OutputToConsole = "FALSE"
b:Sc/eDoseAtPhantom1/Visualize       = "TRUE"

s:Sc/EDepAtPhantom1/Quantity          = "DoseToMedium"
s:Sc/EDepAtPhantom1/Component         = "Phantom2"
b:Sc/EDepAtPhantom1/OutputToConsole = "TRUE"
sv:Sc/EDepAtPhantom1/OnlyIncludeParticlesNamed = 1 "proton"

s:Sc/DoseAtPhantom2/Quantity          = "DoseToMedium"
s:Sc/DoseAtPhantom2/Component         = "Phantom2"
s:Sc/DoseAtPhantom2/OutputFile        = "DoseAtPhantom2"
s:Sc/DoseAtPhantom2/OutputType        = "csv"
b:Sc/DoseAtPhantom2/OutputToConsole = "TRUE"
b:Sc/DoseAtPhantom2/Visualize         = "TRUE"

s:Sc/EDepAtPhantom2/Quantity = "EnergyDeposit"
s:Sc/EDepAtPhantom2/Component = "Phantom2"

s:Gr/ViewA/Type              = "OpenGL"
i:Gr/ViewA/WindowSizeX       = 900
i:Gr/ViewA/WindowSizeY       = 900
d:Gr/ViewA/Theta              = 55 deg
d:Gr/ViewA/Phi                = 20 deg
s:Gr/ViewA/Projection         = "Perspective"
d:Gr/ViewA/PerspectiveAngle   = 30 deg
u:Gr/ViewA/Zoom               = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type            = "Beam"
s:So/Example/Component       = "BeamPosition"
s:So/Example/BeamParticle    = "proton"
d:So/Example/BeamEnergy      = 169.23 MeV

```

(continues on next page)

(continued from previous page)

```

u:So/Example/BeamEnergySpread      = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX    = 10. cm
d:So/Example/BeamPositionCutoffY    = 10. cm
d:So/Example/BeamPositionSpreadX    = 0.65 cm
d:So/Example/BeamPositionSpreadY    = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX     = 90. deg
d:So/Example/BeamAngularCutoffY     = 90. deg
d:So/Example/BeamAngularSpreadX     = 0.0032 rad
d:So/Example/BeamAngularSpreadY     = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 26.3 Dose.txt

```

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Phantom/Type    = "TsBox"
s:Ge/Phantom/Parent  = "World"
s:Ge/Phantom/Material = "Air"
d:Ge/Phantom/HLX     = 30.0 cm
d:Ge/Phantom/HLY     = 30.0 cm
d:Ge/Phantom/HLZ     = 10.0 cm
d:Ge/Phantom/TransZ   = -30. cm
s:Ge/Phantom/Color    = "blue"

s:Sc/DoseAtPhantom/Quantity      = "DoseToMedium"
s:Sc/DoseAtPhantom/Component     = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"

#s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi        = 20 deg
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV

```

(continues on next page)

(continued from previous page)

```

u:So/Example/BeamEnergySpread      = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX    = 10. cm
d:So/Example/BeamPositionCutoffY    = 10. cm
d:So/Example/BeamPositionSpreadX    = 0.65 cm
d:So/Example/BeamPositionSpreadY    = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX     = 90. deg
d:So/Example/BeamAngularCutoffY     = 90. deg
d:So/Example/BeamAngularSpreadX     = 0.0032 rad
d:So/Example/BeamAngularSpreadY     = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100000

i:Ts/ShowHistoryCountAtInterval = 10000
i:Ts/NumberOfThreads = 7

#b:Ts/PauseBeforeQuit = "True"

```

## 26.4 DoseInVoxelMaterials.txt

```

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Phantom/Type    = "TsBox"
s:Ge/Phantom/Parent  = "World"
s:Ge/Phantom/Material = "G4_WATER"
d:Ge/Phantom/HLX     = 30.0 cm
d:Ge/Phantom/HLY     = 30.0 cm
d:Ge/Phantom/HLZ     = 30.0 cm
d:Ge/Phantom/TransZ  = -30. cm
s:Ge/Phantom/Color   = "blue"
i:Ge/Phantom/XBins   = 10
i:Ge/Phantom/YBins   = 10
i:Ge/Phantom/ZBins   = 10
includeFile          = VoxelMaterials.txt
d:Ge/Phantom/RotY    = 270 deg

s:Sc/DoseAtPhantom/Quantity      = "DoseToMedium"
s:Sc/DoseAtPhantom/Component     = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
#d:Gr/ViewA/Theta     = 55 deg
#d:Gr/ViewA/Phi       = 20 deg
#s:Gr/ViewA/Projection = "Perspective"
#d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom       = 1.3

```

(continues on next page)



(continued from previous page)

```

b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

i:Ts/ShowHistoryCountAtInterval = 10000
i:Ts/NumberOfThreads = 7

b:Ts/PauseBeforeQuit = "True"

```

## 26.5 DoseToMediumVsWater.txt

```

# Demonstrates dose to medium versus dose to water.

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX = 2.0 m
d:Ge/World/HLY = 2.0 m
d:Ge/World/HLZ = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/WaterBox/Type = "TsBox"
s:Ge/WaterBox/Parent = "World"
s:Ge/WaterBox/Material = "G4_WATER"
d:Ge/WaterBox/HLX = 10.0 cm
d:Ge/WaterBox/HLY = 10.0 cm
d:Ge/WaterBox/HLZ = 2.0 cm
d:Ge/WaterBox/TransX = 0. cm
d:Ge/WaterBox/TransY = 0. cm
d:Ge/WaterBox/TransZ = 10. cm
d:Ge/WaterBox/RotX = 0. deg
d:Ge/WaterBox/RotY = 0. deg
d:Ge/WaterBox/RotZ = 0. deg
s:Ge/WaterBox/Color = "blue"

s:Ge/LeadBox/Type = "TsBox"
s:Ge/LeadBox/Parent = "World"
s:Ge/LeadBox/Material = "Lead"

```

(continues on next page)

(continued from previous page)

```

d:Ge/LeadBox/HLX      = 10.0 cm
d:Ge/LeadBox/HLY      = 10.0 cm
d:Ge/LeadBox/HLZ      = 5.0 cm
d:Ge/LeadBox/TransX   = 0. cm
d:Ge/LeadBox/TransY   = 0. cm
d:Ge/LeadBox/TransZ   = 0. cm
d:Ge/LeadBox/RotX     = 0. deg
d:Ge/LeadBox/RotY     = 0. deg
d:Ge/LeadBox/RotZ     = 0. deg
s:Ge/LeadBox/Color    = "blue"

s:Sc/1WaterBoxDoseToMedium/Quantity      = "DoseToMedium"
s:Sc/1WaterBoxDoseToMedium/Component     = "WaterBox"
b:Sc/1WaterBoxDoseToMedium/OutputToConsole = "TRUE"
s:Sc/1WaterBoxDoseToMedium/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/2WaterBoxDoseToWater/Quantity      = "DoseToWater"
s:Sc/2WaterBoxDoseToWater/Component     = "WaterBox"
b:Sc/2WaterBoxDoseToWater/OutputToConsole = "TRUE"
s:Sc/2WaterBoxDoseToWater/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/3WaterBoxDoseToMaterialWater/Quantity      = "DoseToMaterial"
s:Sc/3WaterBoxDoseToMaterialWater/Material     = "G4_WATER"
s:Sc/3WaterBoxDoseToMaterialWater/Component     = "WaterBox"
b:Sc/3WaterBoxDoseToMaterialWater/OutputToConsole = "TRUE"
s:Sc/3WaterBoxDoseToMaterialWater/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/4WaterBoxDoseToMaterialLead/Quantity      = "DoseToMaterial"
s:Sc/4WaterBoxDoseToMaterialLead/Material     = "Lead"
s:Sc/4WaterBoxDoseToMaterialLead/Component     = "WaterBox"
b:Sc/4WaterBoxDoseToMaterialLead/OutputToConsole = "TRUE"
s:Sc/4WaterBoxDoseToMaterialLead/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/5LeadBoxDoseToMedium/Quantity      = "DoseToMedium"
s:Sc/5LeadBoxDoseToMedium/Component     = "LeadBox"
b:Sc/5LeadBoxDoseToMedium/OutputToConsole = "TRUE"
s:Sc/5LeadBoxDoseToMedium/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/6LeadBoxBoxDoseToWater/Quantity      = "DoseToWater"
s:Sc/6LeadBoxBoxDoseToWater/Component     = "LeadBox"
b:Sc/6LeadBoxBoxDoseToWater/OutputToConsole = "TRUE"
s:Sc/6LeadBoxBoxDoseToWater/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/7LeadBoxDoseToMaterialWater/Quantity      = "DoseToMaterial"
s:Sc/7LeadBoxDoseToMaterialWater/Material     = "G4_WATER"
s:Sc/7LeadBoxDoseToMaterialWater/Component     = "LeadBox"
b:Sc/7LeadBoxDoseToMaterialWater/OutputToConsole = "TRUE"
s:Sc/7LeadBoxDoseToMaterialWater/IfOutputFileAlreadyExists = "Overwrite"

s:Sc/8LeadBoxDoseToMaterialLead/Quantity      = "DoseToMaterial"
s:Sc/8LeadBoxDoseToMaterialLead/Material     = "Lead"
s:Sc/8LeadBoxDoseToMaterialLead/Component     = "LeadBox"
b:Sc/8LeadBoxDoseToMaterialLead/OutputToConsole = "TRUE"
s:Sc/8LeadBoxDoseToMaterialLead/IfOutputFileAlreadyExists = "Overwrite"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"

```

(continues on next page)

(continued from previous page)

```

s:So/Example/BeamParticle           = "gamma"
d:So/Example/BeamEnergy             = 169.23 MeV
u:So/Example/BeamEnergySpread       = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX    = 10. cm
d:So/Example/BeamPositionCutoffY    = 10. cm
d:So/Example/BeamPositionSpreadX    = 0.65 cm
d:So/Example/BeamPositionSpreadY    = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX     = 90. deg
d:So/Example/BeamAngularCutoffY     = 90. deg
d:So/Example/BeamAngularSpreadX     = 0.0032 rad
d:So/Example/BeamAngularSpreadY     = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

s:Gr/ViewA/Type = "OpenGL"

b:Ts/PauseBeforeQuit = "True"

```

## 26.6 DoseVolumeHistogram.txt

```

# Score dose to a cumulative volume histogram

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m

s:Ge/Phantom/Type   = "TsBox"
s:Ge/Phantom/Parent = "World"
s:Ge/Phantom/Material = "Lead"
d:Ge/Phantom/HLX    = 30.0 cm
d:Ge/Phantom/HLY    = 30.0 cm
d:Ge/Phantom/HLZ    = 10.0 cm
d:Ge/Phantom/TransX = 0. cm
d:Ge/Phantom/TransY = 0. cm
d:Ge/Phantom/TransZ = -30. cm
d:Ge/Phantom/RotX   = 0. deg
d:Ge/Phantom/RotY   = 0. deg
d:Ge/Phantom/RotZ   = 0. deg
s:Ge/Phantom/Color  = "blue"
i:Ge/Phantom/XBins  = 10
i:Ge/Phantom/YBins  = 10
i:Ge/Phantom/ZBins  = 10

s:Sc/DoseAtPhantom/Quantity      = "DoseToMedium"
s:Sc/DoseAtPhantom/Component     = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/DoseAtPhantom/Report       = 2 "Sum" "CumulativeVolumeHistogram"
i:Sc/DoseAtPhantom/HistogramBins = 100
d:Sc/DoseAtPhantom/HistogramMin  = 0. Gy
d:Sc/DoseAtPhantom/HistogramMax = .0001 Gy

```

(continues on next page)

(continued from previous page)

```

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

b:Ts/PauseBeforeQuit = "True"

```

## 26.7 EnergyAndDose.txt

```

# Demonstrate scoring energy and dose in a simple water phantom

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX = 2.0 m
d:Ge/World/HLY = 2.0 m
d:Ge/World/HLZ = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Phantom/Type = "TsBox"
s:Ge/Phantom/Parent = "World"
s:Ge/Phantom/Material = "Lead"
d:Ge/Phantom/HLX = 30.0 cm
d:Ge/Phantom/HLY = 30.0 cm
d:Ge/Phantom/HLZ = 10.0 cm
d:Ge/Phantom/TransZ = -30. cm
s:Ge/Phantom/Color = "blue"

s:Sc/EnergyDepAtPhantom/Quantity = "EnergyDeposit"
s:Sc/EnergyDepAtPhantom/Component = "Phantom"
b:Sc/EnergyDepAtPhantom/OutputToConsole = "TRUE"
s:Sc/EnergyDepAtPhantom/IfOutputFileAlreadyExists = "Overwrite"

```

(continues on next page)

(continued from previous page)

```

s:Sc/DoseAtPhantom/Quantity           = "DoseToMedium"
s:Sc/DoseAtPhantom/Component          = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole    = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type                       = "OpenGL"
i:Gr/ViewA/WindowSizeX                = 900
i:Gr/ViewA/WindowSizeY                = 900
d:Gr/ViewA/Theta                      = 55 deg
d:Gr/ViewA/Phi                       = 20 deg
s:Gr/ViewA/Projection                 = "Perspective"
d:Gr/ViewA/PerspectiveAngle           = 30 deg
u:Gr/ViewA/Zoom                       = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type                     = "Beam"
s:So/Example/Component                 = "BeamPosition"
s:So/Example/BeamParticle              = "proton"
d:So/Example/BeamEnergy                = 169.23 MeV
u:So/Example/BeamEnergySpread          = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape  = "Ellipse"
d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 10

b:Ts/PauseBeforeQuit = "True"

```

## 26.8 EnergyDepositBinnedByEnergy.txt

```

# This example demonstrates the three different ways one can bin results by energy.
↪Adjust the value of EBinEnergy to see the effect.
#
# IncidentTrack means we bin based on the energy of the track that was initially
# incident on the scoring volume, even though the track that finally delivers the
# scored energy is likely to be not this track but one of its resulting delta rays.
# This is the default setting.
# It is set this way because the original use of energy binning was to study how
# proton therapy dose due to various energy protons incident on the patient.
#
# PreStep means we bin based on the kinetic energy of the track at the time
# that it deposited the dose.
# This is generally used when one wants to score a spectrum.
# So, for example, if you adjust Quantity below to Fluence, then the scored
# value is what is referred to in the literature as a Spectral Fluence.
#

```

(continues on next page)

(continued from previous page)

```

# DepositedInStep means we bin based on the energy that was deposited in the step.
# You can use this setting to see that almost all actual energy deposits occur
# in small amounts spread along the track.

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Phantom/Type    = "TsBox"
s:Ge/Phantom/Parent  = "World"
s:Ge/Phantom/Material = "G4_WATER"
d:Ge/Phantom/HLX     = 20.0 cm
d:Ge/Phantom/HLY     = 20.0 cm
d:Ge/Phantom/HLZ     = 20.0 cm
d:Ge/Phantom/TransZ  = -30. cm

s:Sc/Case1BinnedByIncidentTrackEnergy/Quantity = "EnergyDeposit"
s:Sc/Case1BinnedByIncidentTrackEnergy/Component = "Phantom"
b:Sc/Case1BinnedByIncidentTrackEnergy/OutputToConsole = "TRUE"
s:Sc/Case1BinnedByIncidentTrackEnergy/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Case1BinnedByIncidentTrackEnergy/EBins = 4
d:Sc/Case1BinnedByIncidentTrackEnergy/EBinMin = 0. MeV
d:Sc/Case1BinnedByIncidentTrackEnergy/EBinMax = 200. MeV
sc:Sc/Case1BinnedByIncidentTrackEnergy/EBinEnergy = "IncidentTrack"

s:Sc/Case2BinnedByIncidentPreStepEnergy/Quantity = "EnergyDeposit"
s:Sc/Case2BinnedByIncidentPreStepEnergy/Component = "Phantom"
b:Sc/Case2BinnedByIncidentPreStepEnergy/OutputToConsole = "TRUE"
s:Sc/Case2BinnedByIncidentPreStepEnergy/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Case2BinnedByIncidentPreStepEnergy/EBins = 4
d:Sc/Case2BinnedByIncidentPreStepEnergy/EBinMin = 0. MeV
d:Sc/Case2BinnedByIncidentPreStepEnergy/EBinMax = 200. MeV
sc:Sc/Case2BinnedByIncidentPreStepEnergy/EBinEnergy = "PreStep"

s:Sc/Case3BinnedByEnergyDepositedInStep/Quantity = "EnergyDeposit"
s:Sc/Case3BinnedByEnergyDepositedInStep/Component = "Phantom"
b:Sc/Case3BinnedByEnergyDepositedInStep/OutputToConsole = "TRUE"
s:Sc/Case3BinnedByEnergyDepositedInStep/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Case3BinnedByEnergyDepositedInStep/EBins = 4
d:Sc/Case3BinnedByEnergyDepositedInStep/EBinMin = 0. MeV
d:Sc/Case3BinnedByEnergyDepositedInStep/EBinMax = 200. MeV
sc:Sc/Case3BinnedByEnergyDepositedInStep/EBinEnergy = "DepositedInStep"

#s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

```

(continues on next page)

(continued from previous page)

```

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "proton"
d:So/Example/BeamEnergy     = 175. MeV
u:So/Example/BeamEnergySpread = 0.
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1

#b:Ts/PauseBeforeQuit = "True"

```

## 26.9 EnergyDivisions.txt

```

# Score surface track count of primary protons
# in energy bins in two phantoms

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/WaterBox/Type   = "TsBox"
s:Ge/WaterBox/Parent = "World"
s:Ge/WaterBox/Material = "G4_WATER"
d:Ge/WaterBox/HLX     = 30.0 cm
d:Ge/WaterBox/HLY     = 30.0 cm
d:Ge/WaterBox/HLZ     = 2.0 cm
d:Ge/WaterBox/TransZ  = -20. cm

s:Ge/LeadBox/Type    = "TsBox"
s:Ge/LeadBox/Parent  = "World"
s:Ge/LeadBox/Material = "Lead"
d:Ge/LeadBox/HLX     = 30.0 cm
d:Ge/LeadBox/HLY     = 30.0 cm
d:Ge/LeadBox/HLZ     = 10.0 cm
d:Ge/LeadBox/TransZ  = -40. cm
d:Ge/LeadBox/RotX    = 0. deg
d:Ge/LeadBox/RotY    = 0. deg
d:Ge/LeadBox/RotZ    = 0. deg

s:Sc/EnergyDepAtPhantom/Quantity = "SurfaceTrackCount"
s:Sc/EnergyDepAtPhantom/Surface  = "WaterBox/ZPlusSurface"
b:Sc/EnergyDepAtPhantom/OutputToConsole = "True"
s:Sc/EnergyDepAtPhantom/OutputType = "csv"

```

(continues on next page)

(continued from previous page)

```

s:Sc/EnergyDepAtPhantom/IfOutputFileAlreadyExists      = "Overwrite"
i:Sc/EnergyDepAtPhantom/EBins                          = 10
sv:Sc/EnergyDepAtPhantom/OnlyIncludeParticlesNamed     = 1 "proton"
s:Sc/EnergyDepAtPhantom/OnlyIncludeParticlesOfGeneration = "primary"
d:Sc/EnergyDepAtPhantom/EBinMin                       = 0. MeV
d:Sc/EnergyDepAtPhantom/EBinMax                       = 100. MeV

s:Sc/EnergyDepAtPhantom2/Quantity                     = "SurfaceTrackCount"
s:Sc/EnergyDepAtPhantom2/Surface                     = "LeadBox/ZPlusSurface"
b:Sc/EnergyDepAtPhantom2/OutputToConsole              = "True"
s:Sc/EnergyDepAtPhantom2/OutputType                   = "csv"
s:Sc/EnergyDepAtPhantom2/IfOutputFileAlreadyExists     = "Overwrite"
i:Sc/EnergyDepAtPhantom2/EBins                        = 10
sv:Sc/EnergyDepAtPhantom2/OnlyIncludeParticlesNamed    = 1 "proton"
s:Sc/EnergyDepAtPhantom2/OnlyIncludeParticlesOfGeneration = "primary"
d:Sc/EnergyDepAtPhantom2/EBinMin                     = 0. MeV
d:Sc/EnergyDepAtPhantom2/EBinMax                     = 100. MeV

s:Gr/ViewA/Type                                       = "OpenGL"
i:Gr/ViewA/WindowSizeX                               = 900
i:Gr/ViewA/WindowSizeY                               = 900
d:Gr/ViewA/Theta                                      = 55 deg
d:Gr/ViewA/Phi                                        = 20 deg
s:Gr/ViewA/Projection                                = "Perspective"
d:Gr/ViewA/PerspectiveAngle                           = 30 deg
u:Gr/ViewA/Zoom                                       = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories           = "True"

s:So/Example/Type                                     = "Beam"
s:So/Example/Component                               = "BeamPosition"
s:So/Example/BeamParticle                             = "proton"
d:So/Example/BeamEnergy                               = 169.23 MeV
u:So/Example/BeamEnergySpread                         = 0.757504
s:So/Example/BeamPositionDistribution                 = "Gaussian"
s:So/Example/BeamPositionCutoffShape                 = "Ellipse"
d:So/Example/BeamPositionCutoffX                     = 10. cm
d:So/Example/BeamPositionCutoffY                     = 10. cm
d:So/Example/BeamPositionSpreadX                     = 0.65 cm
d:So/Example/BeamPositionSpreadY                     = 0.65 cm
s:So/Example/BeamAngularDistribution                 = "Gaussian"
d:So/Example/BeamAngularCutoffX                     = 90. deg
d:So/Example/BeamAngularCutoffY                     = 90. deg
d:So/Example/BeamAngularSpreadX                     = 0.0032 rad
d:So/Example/BeamAngularSpreadY                     = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun                  = 1000

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 26.10 EnergyInBinnedCylinder.txt

```

# Demonstrates binning by R, Phi and Z in a cylinder

s:Ge/World/Material      = "Vacuum"

```

(continues on next page)



(continued from previous page)

```

d:Ge/World/HLX           = 1.0 m
d:Ge/World/HLY           = 1.0 m
d:Ge/World/HLZ           = 1.0 m
d:Ge/World/MaxStepSize   = 1.0 m

s:Ge/ScoringCylinder/Type = "TsCylinder"
s:Ge/ScoringCylinder/Parent = "World"
s:Ge/ScoringCylinder/Material = "G4_WATER"
d:Ge/ScoringCylinder/RMin   = 2. cm
d:Ge/ScoringCylinder/RMax   = 40. cm
d:Ge/ScoringCylinder/HL     = 40. cm
d:Ge/ScoringCylinder/SPhi   = 0. deg
d:Ge/ScoringCylinder/DPhi   = 360. deg
i:Ge/ScoringCylinder/RBins   = 6
i:Ge/ScoringCylinder/PhiBins = 6
i:Ge/ScoringCylinder/ZBins   = 6

s:Sc/MyScorer/Quantity      = "EnergyDeposit"
s:Sc/MyScorer/Component     = "ScoringCylinder"
s:Sc/MyScorer/OutputFile    = "Energy_Test"
s:Sc/MyScorer/OutputType    = "csv"
b:Sc/MyScorer/OutputToConsole = "False"
s:Sc/MyScorer/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type             = "OpenGL"
i:Gr/ViewA/WindowSizeX      = 1024
i:Gr/ViewA/WindowSizeY      = 768
d:Gr/ViewA/Theta             = 55 deg
d:Gr/ViewA/Phi               = 20 deg
s:Gr/ViewA/Projection        = "Perspective"
d:Gr/ViewA/PerspectiveAngle  = 30 deg
u:Gr/ViewA/Zoom              = 2.
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle    = "proton"
d:So/Example/BeamEnergy      = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1000

i:Ts/ShowHistoryCountAtInterval = 100
b:Ts/PauseBeforeQuit = "True"

```

## 26.11 Filters.txt

```
# Test every kind of scoring filter

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/Foil/Type      = "TsBox"
s:Ge/Foil/Parent    = "World"
s:Ge/Foil/Material   = "Lead"
d:Ge/Foil/HLX       = 30.0 cm
d:Ge/Foil/HLY       = 30.0 cm
d:Ge/Foil/HLZ       = .5 cm
d:Ge/Foil/TransX    = 0. cm
d:Ge/Foil/TransY    = 0. cm
d:Ge/Foil/TransZ    = 0. cm
d:Ge/Foil/RotX      = 0. deg
d:Ge/Foil/RotY      = 0. deg
d:Ge/Foil/RotZ      = 0. deg

s:Ge/Box/Type       = "TsBox"
s:Ge/Box/Parent     = "World"
s:Ge/Box/Material    = "G4_WATER"
d:Ge/Box/HLX        = 20.0 cm
d:Ge/Box/HLY        = 20.0 cm
d:Ge/Box/HLZ        = 20.0 cm
d:Ge/Box/TransX     = 0. cm
d:Ge/Box/TransY     = 0. cm
d:Ge/Box/TransZ     = -30. cm
d:Ge/Box/RotX       = 0. deg
d:Ge/Box/RotY       = 0. deg
d:Ge/Box/RotZ       = 0. deg

s:Sc/OnlyIncludeParticlesOfGeneration/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesOfGeneration/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesOfGeneration/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesOfGeneration/OutputType = "csv"
s:Sc/OnlyIncludeParticlesOfGeneration/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/OnlyIncludeParticlesOfGeneration/OnlyIncludeParticlesOfGeneration = "Primary"

s:Sc/OnlyIncludeParticlesCharged/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesCharged/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesCharged/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesCharged/OutputType = "csv"
s:Sc/OnlyIncludeParticlesCharged/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesCharged/OnlyIncludeParticlesCharged = 1 "Positive"

s:Sc/OnlyIncludeParticlesNotCharged/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotCharged/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotCharged/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotCharged/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotCharged/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesNotCharged/OnlyIncludeParticlesNotCharged = 1 "Positive"
```

(continues on next page)

(continued from previous page)

```

s:Sc/OnlyIncludeParticlesOfAtomicMass/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesOfAtomicMass/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesOfAtomicMass/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesOfAtomicMass/OutputType = "csv"
s:Sc/OnlyIncludeParticlesOfAtomicMass/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/OnlyIncludeParticlesOfAtomicMass/OnlyIncludeParticlesOfAtomicMass = 1

s:Sc/OnlyIncludeParticlesNotOfAtomicMass/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotOfAtomicMass/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotOfAtomicMass/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotOfAtomicMass/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotOfAtomicMass/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/OnlyIncludeParticlesNotOfAtomicMass/OnlyIncludeParticlesNotOfAtomicMass = 1

s:Sc/OnlyIncludeParticlesOfAtomicNumber/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesOfAtomicNumber/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesOfAtomicNumber/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesOfAtomicNumber/OutputType = "csv"
s:Sc/OnlyIncludeParticlesOfAtomicNumber/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/OnlyIncludeParticlesOfAtomicNumber/OnlyIncludeParticlesOfAtomicNumber = 1

s:Sc/OnlyIncludeParticlesNotOfAtomicNumber/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotOfAtomicNumber/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotOfAtomicNumber/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotOfAtomicNumber/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotOfAtomicNumber/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/OnlyIncludeParticlesNotOfAtomicNumber/OnlyIncludeParticlesNotOfAtomicNumber = 1

s:Sc/OnlyIncludeParticlesWithInitialKEBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialKEBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialKEBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialKEBelow/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialKEBelow/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialKEBelow/OnlyIncludeParticlesWithInitialKEBelow = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialKENotBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialKENotBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialKENotBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialKENotBelow/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialKENotBelow/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialKENotBelow/OnlyIncludeParticlesWithInitialKENotBelow = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialKE/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialKE/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialKE/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialKE/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialKE/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialKE/OnlyIncludeParticlesWithInitialKE = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialKENot/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialKENot/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialKENot/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialKENot/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialKENot/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialKENot/OnlyIncludeParticlesWithInitialKENot = 100. MeV

```

↪ MeV

(continues on next page)

(continued from previous page)

```

s:Sc/OnlyIncludeParticlesWithInitialKEAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialKEAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialKEAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialKEAbove/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialKEAbove/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialKEAbove/OnlyIncludeParticlesWithInitialKEAbove = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialKENotAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialKENotAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialKENotAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialKENotAbove/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialKENotAbove/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialKENotAbove/
OnlyIncludeParticlesWithInitialKENotAbove = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialMomentumBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialMomentumBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialMomentumBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialMomentumBelow/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialMomentumBelow/IfOutputFileAlreadyExists =
Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialMomentumBelow/
OnlyIncludeParticlesWithInitialMomentumBelow = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialMomentumNotBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNotBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialMomentumNotBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNotBelow/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNotBelow/IfOutputFileAlreadyExists =
Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialMomentumNotBelow/
OnlyIncludeParticlesWithInitialMomentumNotBelow = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialMomentum/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialMomentum/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialMomentum/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialMomentum/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialMomentum/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialMomentum/OnlyIncludeParticlesWithInitialMomentum = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialMomentumNot/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNot/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialMomentumNot/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNot/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNot/IfOutputFileAlreadyExists = "Overwrite"
Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialMomentumNot/
OnlyIncludeParticlesWithInitialMomentumNot = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialMomentumAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialMomentumAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialMomentumAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialMomentumAbove/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialMomentumAbove/IfOutputFileAlreadyExists =
Overwrite"

```

(continues on next page)

(continued from previous page)

```

d:Sc/OnlyIncludeParticlesWithInitialMomentumAbove/
→OnlyIncludeParticlesWithInitialMomentumAbove = 100. MeV

s:Sc/OnlyIncludeParticlesWithInitialMomentumNotAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNotAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesWithInitialMomentumNotAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNotAbove/OutputType = "csv"
s:Sc/OnlyIncludeParticlesWithInitialMomentumNotAbove/IfOutputFileAlreadyExists =
→"Overwrite"
d:Sc/OnlyIncludeParticlesWithInitialMomentumNotAbove/
→OnlyIncludeParticlesWithInitialMomentumNotAbove = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleKEBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleKEBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleKEBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleKEBelow/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleKEBelow/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleKEBelow/OnlyIncludeIfIncidentParticleKEBelow = 100.
→MeV

s:Sc/OnlyIncludeIfIncidentParticleKENotBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleKENotBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleKENotBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleKENotBelow/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleKENotBelow/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleKENotBelow/OnlyIncludeIfIncidentParticleKENotBelow
→= 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleKE/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleKE/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleKE/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleKE/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleKE/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleKE/OnlyIncludeIfIncidentParticleKE = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleKENot/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleKENot/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleKENot/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleKENot/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleKENot/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleKENot/OnlyIncludeIfIncidentParticleKENot = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleKEAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleKEAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleKEAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleKEAbove/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleKEAbove/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleKEAbove/OnlyIncludeIfIncidentParticleKEAbove = 100.
→MeV

s:Sc/OnlyIncludeIfIncidentParticleKENotAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleKENotAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleKENotAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleKENotAbove/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleKENotAbove/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleKENotAbove/OnlyIncludeIfIncidentParticleKENotAbove
→= 100. MeV

```

(continues on next page)

(continued from previous page)

```

s:Sc/OnlyIncludeIfIncidentParticleMomentumBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleMomentumBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleMomentumBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleMomentumBelow/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleMomentumBelow/IfOutputFileAlreadyExists = "Overwrite"
↪"
d:Sc/OnlyIncludeIfIncidentParticleMomentumBelow/
↪OnlyIncludeIfIncidentParticleMomentumBelow = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleMomentumNotBelow/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNotBelow/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleMomentumNotBelow/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNotBelow/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNotBelow/IfOutputFileAlreadyExists =
↪"Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleMomentumNotBelow/
↪OnlyIncludeIfIncidentParticleMomentumNotBelow = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleMomentum/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleMomentum/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleMomentum/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleMomentum/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleMomentum/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleMomentum/OnlyIncludeIfIncidentParticleMomentum = ↪
↪100. MeV

s:Sc/OnlyIncludeIfIncidentParticleMomentumNot/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNot/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleMomentumNot/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNot/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNot/IfOutputFileAlreadyExists = "Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleMomentumNot/
↪OnlyIncludeIfIncidentParticleMomentumNot = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleMomentumAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleMomentumAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleMomentumAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleMomentumAbove/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleMomentumAbove/IfOutputFileAlreadyExists = "Overwrite"
↪"
d:Sc/OnlyIncludeIfIncidentParticleMomentumAbove/
↪OnlyIncludeIfIncidentParticleMomentumAbove = 100. MeV

s:Sc/OnlyIncludeIfIncidentParticleMomentumNotAbove/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNotAbove/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfIncidentParticleMomentumNotAbove/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNotAbove/OutputType = "csv"
s:Sc/OnlyIncludeIfIncidentParticleMomentumNotAbove/IfOutputFileAlreadyExists =
↪"Overwrite"
d:Sc/OnlyIncludeIfIncidentParticleMomentumNotAbove/
↪OnlyIncludeIfIncidentParticleMomentumNotAbove = 100. MeV

s:Sc/OnlyIncludeParticlesFromProcess/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesFromProcess/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesFromProcess/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesFromProcess/OutputType = "csv"

```

(continues on next page)

(continued from previous page)

```

s:Sc/OnlyIncludeParticlesFromProcess/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesFromProcess/OnlyIncludeParticlesFromProcess = 1 "hIoni"

s:Sc/OnlyIncludeParticlesNotFromProcess/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotFromProcess/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotFromProcess/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotFromProcess/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotFromProcess/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesNotFromProcess/OnlyIncludeParticlesNotFromProcess = 1 "hIoni"
↪ "

s:Sc/OnlyIncludeIfParticleOrAncestorFromProcess/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorFromProcess/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorFromProcess/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorFromProcess/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorFromProcess/IfOutputFileAlreadyExists = "Overwrite"
↪ "
sv:Sc/OnlyIncludeIfParticleOrAncestorFromProcess/
↪ OnlyIncludeIfParticleOrAncestorFromProcess = 1 "hIoni"

s:Sc/OnlyIncludeIfParticleOrAncestorNotFromProcess/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromProcess/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotFromProcess/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromProcess/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromProcess/IfOutputFileAlreadyExists =
↪ "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotFromProcess/
↪ OnlyIncludeIfParticleOrAncestorNotFromProcess = 1 "hIoni"

s:Sc/OnlyIncludeParticlesNamed/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNamed/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNamed/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNamed/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNamed/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesNamed/OnlyIncludeParticlesNamed = 1 "proton"

s:Sc/OnlyIncludeParticlesNotNamed/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotNamed/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotNamed/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotNamed/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotNamed/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesNotNamed/OnlyIncludeParticlesNotNamed = 1 "proton"

s:Sc/OnlyIncludeIfParticleOrAncestorNamed/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNamed/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNamed/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNamed/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNamed/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNamed/OnlyIncludeIfParticleOrAncestorNamed = 1
↪ "proton"

s:Sc/OnlyIncludeIfParticleOrAncestorNotNamed/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotNamed/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotNamed/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotNamed/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotNamed/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotNamed/OnlyIncludeIfParticleOrAncestorNotNamed,
↪ = 1 "proton"

```

(continues on next page)



(continued from previous page)

```

s:Sc/OnlyIncludeParticlesFromVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesFromVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesFromVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesFromVolume/OutputType = "csv"
s:Sc/OnlyIncludeParticlesFromVolume/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesFromVolume/OnlyIncludeParticlesFromVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeParticlesNotFromVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotFromVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotFromVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotFromVolume/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotFromVolume/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesNotFromVolume/OnlyIncludeParticlesNotFromVolume = 2 "World"
↪ "Foil"

s:Sc/OnlyIncludeParticlesFromComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesFromComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesFromComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesFromComponent/OutputType = "csv"
s:Sc/OnlyIncludeParticlesFromComponent/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesFromComponent/OnlyIncludeParticlesFromComponent = 2 "World"
↪ "Foil"

s:Sc/OnlyIncludeParticlesNotFromComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotFromComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesNotFromComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotFromComponent/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotFromComponent/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeParticlesNotFromComponent/OnlyIncludeParticlesNotFromComponent = 2
↪ "World" "Foil"

s:Sc/OnlyIncludeParticlesFromComponentOrSubComponentsOf/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesFromComponentOrSubComponentsOf/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeParticlesFromComponentOrSubComponentsOf/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesFromComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeParticlesFromComponentOrSubComponentsOf/IfOutputFileAlreadyExists =
↪ "Overwrite"
sv:Sc/OnlyIncludeParticlesFromComponentOrSubComponentsOf/
↪ OnlyIncludeParticlesFromComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf/Quantity =
↪ "SurfaceTrackCount"
s:Sc/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf/Surface = "Box/ZPlusSurface"
↪ "
b:Sc/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf/IfOutputFileAlreadyExists ↪
↪ "Overwrite"
sv:Sc/OnlyIncludeParticlesNotFromComponentOrSubComponentsOf/
↪ OnlyIncludeParticlesNotFromComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorFromVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorFromVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorFromVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorFromVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorFromVolume/IfOutputFileAlreadyExists = "Overwrite"

```

(continues on next page)



(continued from previous page)

```

sv:Sc/OnlyIncludeIfParticleOrAncestorFromVolume/
↳OnlyIncludeIfParticleOrAncestorFromVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotFromVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotFromVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromVolume/IfOutputFileAlreadyExists =
↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotFromVolume/
↳OnlyIncludeIfParticleOrAncestorNotFromVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorFromComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorFromComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorFromComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorFromComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorFromComponent/IfOutputFileAlreadyExists =
↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorFromComponent/
↳OnlyIncludeIfParticleOrAncestorFromComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponent/IfOutputFileAlreadyExists =
↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponent/
↳OnlyIncludeIfParticleOrAncestorNotFromComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf/Quantity =
↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf/Surface = "Box/
↳ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf/OutputToConsole =
↳"TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf/
↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf/
↳OnlyIncludeIfParticleOrAncestorFromComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf/Quantity =
↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf/Surface = "Box/
↳ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf/OutputToConsole_
↳= "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf/OutputType =
↳"csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf/
↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf/
↳OnlyIncludeIfParticleOrAncestorNotFromComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleInteractedInVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleInteractedInVolume/Surface = "Box/ZPlusSurface"

```

(continues on next page)

(continued from previous page)

```

b:Sc/OnlyIncludeIfParticleInteractedInVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleInteractedInVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleInteractedInVolume/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleInteractedInVolume/OnlyIncludeIfParticleInteractedInVolume_
↳= 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleNotInteractedInVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleNotInteractedInVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleNotInteractedInVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleNotInteractedInVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleNotInteractedInVolume/IfOutputFileAlreadyExists = "Overwrite"
↳"
sv:Sc/OnlyIncludeIfParticleNotInteractedInVolume/
↳OnlyIncludeIfParticleNotInteractedInVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleInteractedInComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleInteractedInComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleInteractedInComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleInteractedInComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleInteractedInComponent/IfOutputFileAlreadyExists = "Overwrite"
↳"
sv:Sc/OnlyIncludeIfParticleInteractedInComponent/
↳OnlyIncludeIfParticleInteractedInComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleNotInteractedInComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleNotInteractedInComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleNotInteractedInComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleNotInteractedInComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleNotInteractedInComponent/IfOutputFileAlreadyExists =
↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleNotInteractedInComponent/
↳OnlyIncludeIfParticleNotInteractedInComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf/Quantity =
↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf/Surface = "Box/
↳ZPlusSurface"
b:Sc/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf/OutputToConsole =
↳"TRUE"
s:Sc/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf/
↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf/
↳OnlyIncludeIfParticleInteractedInComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf/Quantity =
↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf/Surface = "Box/
↳ZPlusSurface"
b:Sc/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf/OutputToConsole =
↳"TRUE"
s:Sc/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf/
↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf/
↳OnlyIncludeIfParticleNotInteractedInComponentOrSubComponentsOf = 2 "World" "Foil"

```

(continues on next page)

(continued from previous page)

```

s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorInteractedInVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInVolume/IfOutputFileAlreadyExists =
↳ "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorInteractedInVolume/
↳ OnlyIncludeIfParticleOrAncestorInteractedInVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume/Quantity =
↳ "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume/IfOutputFileAlreadyExists =
↳ "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInVolume/
↳ OnlyIncludeIfParticleOrAncestorNotInteractedInVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponent/Quantity =
↳ "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponent/IfOutputFileAlreadyExists =
↳ "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponent/
↳ OnlyIncludeIfParticleOrAncestorInteractedInComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent/Quantity =
↳ "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent/Surface = "Box/
↳ ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent/
↳ IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponent/
↳ OnlyIncludeIfParticleOrAncestorNotInteractedInComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf/Quantity =
↳ "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf/Surface =
↳ "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf/
↳ OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf/OutputType,
↳ = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf/
↳ IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf/
↳ OnlyIncludeIfParticleOrAncestorInteractedInComponentOrSubComponentsOf = 2 "World"
↳ "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf/
↳ Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf/Surface,
↳ = "Box/ZPlusSurface"

```

(continues on next page)

(continued from previous page)

```

b:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf/
↳OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf/
↳OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf/
↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf/
↳OnlyIncludeIfParticleOrAncestorNotInteractedInComponentOrSubComponentsOf = 2 "World
↳" "Foil"

s:Sc/OnlyIncludeIfParticleTraversedVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleTraversedVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleTraversedVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleTraversedVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleTraversedVolume/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleTraversedVolume/OnlyIncludeIfParticleTraversedVolume = 2
↳"World" "Foil"

s:Sc/OnlyIncludeIfParticleNotTraversedVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleNotTraversedVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleNotTraversedVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleNotTraversedVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleNotTraversedVolume/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleNotTraversedVolume/OnlyIncludeIfParticleNotTraversedVolume_
↳= 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleTraversedComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleTraversedComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleTraversedComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleTraversedComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleTraversedComponent/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleTraversedComponent/OnlyIncludeIfParticleTraversedComponent_
↳= 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleNotTraversedComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleNotTraversedComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleNotTraversedComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleNotTraversedComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleNotTraversedComponent/IfOutputFileAlreadyExists = "Overwrite
↳"
sv:Sc/OnlyIncludeIfParticleNotTraversedComponent/
↳OnlyIncludeIfParticleNotTraversedComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf/Quantity =
↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf/Surface = "Box/
↳ZPlusSurface"
b:Sc/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf/
↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf/
↳OnlyIncludeIfParticleTraversedComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf/Quantity =
↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf/Surface = "Box/
↳ZPlusSurface"

```

(continues on next page)

(continued from previous page)

```

b:Sc/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf/OutputToConsole =
  ↳"TRUE"
s:Sc/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf/
  ↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf/
  ↳OnlyIncludeIfParticleNotTraversedComponentOrSubComponentsOf = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorTraversedVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorTraversedVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedVolume/IfOutputFileAlreadyExists =
  ↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorTraversedVolume/
  ↳OnlyIncludeIfParticleOrAncestorTraversedVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedVolume/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedVolume/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedVolume/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedVolume/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedVolume/IfOutputFileAlreadyExists =
  ↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedVolume/
  ↳OnlyIncludeIfParticleOrAncestorNotTraversedVolume = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponent/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponent/IfOutputFileAlreadyExists =
  ↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponent/
  ↳OnlyIncludeIfParticleOrAncestorTraversedComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponent/Quantity =
  ↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponent/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponent/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponent/OutputType = "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponent/IfOutputFileAlreadyExists =
  ↳"Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponent/
  ↳OnlyIncludeIfParticleOrAncestorNotTraversedComponent = 2 "World" "Foil"

s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf/Quantity =
  ↳"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf/Surface =
  ↳"Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf/
  ↳OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf/OutputType =
  ↳"csv"
s:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf/
  ↳IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf/
  ↳OnlyIncludeIfParticleOrAncestorTraversedComponentOrSubComponentsOf = 2 "World" "Foil"
  ↳"

```

(continues on next page)

(continued from previous page)

```

s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf/Quantity =
→"SurfaceTrackCount"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf/Surface =
→"Box/ZPlusSurface"
b:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf/
→OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf/OutputType
→= "csv"
s:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf/
→IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf/
→OnlyIncludeIfParticleOrAncestorNotTraversedComponentOrSubComponentsOf = 2 "World"
→"Foil"

s:Sc/OnlyIncludeIfInMaterial/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfInMaterial/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfInMaterial/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfInMaterial/OutputType = "csv"
s:Sc/OnlyIncludeIfInMaterial/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfInMaterial/OnlyIncludeIfInMaterial = 1 "Lead"

s:Sc/OnlyIncludeIfNotInMaterial/Quantity = "SurfaceTrackCount"
s:Sc/OnlyIncludeIfNotInMaterial/Surface = "Box/ZPlusSurface"
b:Sc/OnlyIncludeIfNotInMaterial/OutputToConsole = "TRUE"
s:Sc/OnlyIncludeIfNotInMaterial/OutputType = "csv"
s:Sc/OnlyIncludeIfNotInMaterial/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/OnlyIncludeIfNotInMaterial/OnlyIncludeIfNotInMaterial = 1 "Lead"

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1

b:Ts/PauseBeforeQuit = "True"

```

## 26.12 FoilToBox.txt

```
# Test scoring in geometry of particle passing through a foil to a box

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Foil/Type       = "TsBox"
s:Ge/Foil/Parent     = "World"
s:Ge/Foil/Material   = "Lead"
d:Ge/Foil/HLX        = 30.0 cm
d:Ge/Foil/HLY        = 30.0 cm
d:Ge/Foil/HLZ        = .5 cm
d:Ge/Foil/TransX     = 0. cm
d:Ge/Foil/TransY     = 0. cm
d:Ge/Foil/TransZ     = 0. cm
d:Ge/Foil/RotX       = 0. deg
d:Ge/Foil/RotY       = 0. deg
d:Ge/Foil/RotZ       = 0. deg
i:Ge/Foil/XBins      = 1
i:Ge/Foil/YBins      = 1
i:Ge/Foil/ZBins      = 1

s:Ge/Box/Type        = "TsBox"
s:Ge/Box/Parent      = "World"
s:Ge/Box/Material     = "G4_WATER"
d:Ge/Box/HLX         = 20.0 cm
d:Ge/Box/HLY         = 20.0 cm
d:Ge/Box/HLZ         = 20.0 cm
d:Ge/Box/TransX      = 0. cm
d:Ge/Box/TransY      = 0. cm
d:Ge/Box/TransZ      = -30. cm
d:Ge/Box/RotX        = 0. deg
d:Ge/Box/RotY        = 0. deg
d:Ge/Box/RotZ        = 0. deg
s:Ge/Box/Color       = "blue"
i:Ge/Box/ZBins       = 2

s:Sc/MyScorer/Quantity = "SurfaceTrackCount"
s:Sc/MyScorer/Surface  = "Box/ZPlusSurface"
b:Sc/MyScorer/OutputToConsole = "True"
s:Sc/MyScorer/OutputType = "csv"
s:Sc/MyScorer/IfOutputFileAlreadyExists = "Overwrite"
#i:Sc/MyScorer/EBins = 10
s:Sc/MyScorer/OnlyIncludeParticlesGoing = "Out"
#sv:Sc/MyScorer/OnlyIncludeParticlesNamed = 1 "gamma"
s:Sc/MyScorer/OnlyIncludeParticlesOfGeneration = "secondary"
#d:Sc/MyScorer/EBinMin = 0. MeV
d:Sc/MyScorer/EBinMax = 110. MeV
#d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKEAbove = 0 MeV

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
```

(continues on next page)

(continued from previous page)

```

d:Gr/ViewA/Theta           = 55 deg
d:Gr/ViewA/Phi             = 20 deg
s:Gr/ViewA/Projection       = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom             = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "proton"
d:So/Example/BeamEnergy     = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 26.13 FoilToCylinder.txt

```

# Test scoring in geometry of particle passing through a foil to a cylinder

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Foil/Type       = "TsBox"
s:Ge/Foil/Parent     = "World"
s:Ge/Foil/Material   = "Lead"
d:Ge/Foil/HLX        = 30.0 cm
d:Ge/Foil/HLY        = 30.0 cm
d:Ge/Foil/HLZ        = .5 cm
d:Ge/Foil/TransX     = 0. cm
d:Ge/Foil/TransY     = 0. cm
d:Ge/Foil/TransZ     = -10. cm
d:Ge/Foil/RotX       = 0. deg
d:Ge/Foil/RotY       = 0. deg
d:Ge/Foil/RotZ       = 0. deg
s:Ge/Foil/Color       = "blue"
i:Ge/Foil/XBins       = 1
i:Ge/Foil/YBins       = 1
i:Ge/Foil/ZBins       = 1

```

(continues on next page)



(continued from previous page)

```

s:Ge/Cylinder/Type      = "TsCylinder"
s:Ge/Cylinder/Parent    = "World"
s:Ge/Cylinder/Material  = "Vacuum"
d:Ge/Cylinder/RMin      = 50.0 cm
d:Ge/Cylinder/RMax      = 55.0 cm
d:Ge/Cylinder/SPhi      = 20.0 deg
d:Ge/Cylinder/DPhi      = 100. deg
d:Ge/Cylinder/HL        = 50. cm
d:Ge/Cylinder/TransX    = 0. cm
d:Ge/Cylinder/TransY    = 0. cm
d:Ge/Cylinder/TransZ    = -30. cm
d:Ge/Cylinder/RotX      = 0. deg
d:Ge/Cylinder/RotY      = 0. deg
d:Ge/Cylinder/RotZ      = 0. deg
s:Ge/Cylinder/Color     = "blue"
i:Ge/Cylinder/RBins     = 2

s:Sc/MyScorer/Quantity      = "SurfaceTrackCount"
s:Sc/MyScorer/Surface       = "Cylinder/OuterCurvedSurface"
b:Sc/MyScorer/OutputToConsole = "True"
s:Sc/MyScorer/OutputType    = "csv"
s:Sc/MyScorer/IfOutputFileAlreadyExists = "Overwrite"
#i:Sc/MyScorer/EBins        = 10
s:Sc/MyScorer/OnlyIncludeParticlesGoing = "Out"
#sv:Sc/MyScorer/OnlyIncludeParticlesNamed = 1 "gamma"
s:Sc/MyScorer/OnlyIncludeParticlesOfGeneration = "secondary"
#d:Sc/MyScorer/EBinMin      = 0. MeV
d:Sc/MyScorer/EBinMax      = 110. MeV
#d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKEAbove = 0 MeV

# Graphics
s:Gr/ViewA/Type            = "OpenGL"
i:Gr/ViewA/WindowSizeX     = 900
i:Gr/ViewA/WindowSizeY     = 900
d:Gr/ViewA/Theta           = 55 deg
d:Gr/ViewA/Phi             = 20 deg
s:Gr/ViewA/Projection      = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom            = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy    = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad

```

(continues on next page)

(continued from previous page)

```
i:So/Example/NumberOfHistoriesInRun = 10
b:Ts/PauseBeforeQuit = "True"
```

## 26.14 FoilToSphere.txt

```
# Test scoring in geometry of particle passing through a foil to a sphere

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Foil/Type      = "TsBox"
s:Ge/Foil/Parent    = "World"
s:Ge/Foil/Material  = "Lead"
d:Ge/Foil/HLX       = 30.0 cm
d:Ge/Foil/HLY       = 30.0 cm
d:Ge/Foil/HLZ       = .5 cm
d:Ge/Foil/TransX     = 0. cm
d:Ge/Foil/TransY     = 0. cm
d:Ge/Foil/TransZ     = -30. cm
d:Ge/Foil/RotX       = 0. deg
d:Ge/Foil/RotY       = 0. deg
d:Ge/Foil/RotZ       = 0. deg
s:Ge/Foil/Color      = "blue"
i:Ge/Foil/XBins       = 1
i:Ge/Foil/YBins       = 1
i:Ge/Foil/ZBins       = 1

s:Ge/Sphere/Type     = "TsSphere"
s:Ge/Sphere/Parent   = "World"
s:Ge/Sphere/Material = "Vacuum"
d:Ge/Sphere/RMin     = 50.0 cm
d:Ge/Sphere/RMax     = 55.0 cm
d:Ge/Sphere/SPhi     = 0.0 deg
d:Ge/Sphere/DPhi     = 120. deg
d:Ge/Sphere/STheta   = 0 deg
d:Ge/Sphere/DTheta   = 180 deg
d:Ge/Sphere/TransX    = 0. cm
d:Ge/Sphere/TransY    = 0. cm
d:Ge/Sphere/TransZ    = -30. cm
d:Ge/Sphere/RotX     = 0. deg
d:Ge/Sphere/RotY     = 0. deg
d:Ge/Sphere/RotZ     = 0. deg
s:Ge/Sphere/Color    = "blue"
i:Ge/Sphere/RBins    = 2

s:Sc/MyScorer/Quantity = "SurfaceTrackCount"
s:Sc/MyScorer/Surface  = "Sphere/InnerCurvedSurface"
b:Sc/MyScorer/OutputToConsole = "True"
s:Sc/MyScorer/OutputType = "csv"
s:Sc/MyScorer/IfOutputFileAlreadyExists = "Overwrite"
```

(continues on next page)

(continued from previous page)

```

#i:Sc/MyScorer/EBins = 10
s:Sc/MyScorer/OnlyIncludeParticlesGoing = "Out"
#sv:Sc/MyScorer/OnlyIncludeParticlesNamed = 1 "gamma"
s:Sc/MyScorer/OnlyIncludeParticlesOfGeneration = "secondary"
#d:Sc/MyScorer/EBinMin = 0. MeV
d:Sc/MyScorer/EBinMax = 110. MeV
#d:Sc/MyScorer/OnlyIncludeIfIncidentParticleKEAbove = 0 MeV

# Graphics
s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 26.15 Gated.txt

```

# Demonstrates Gated scoring.
# First scorer is gated, sometimes active, sometimes inactive,
# under the control of a time feature and outputs after each run.
# Second scorer is gated, sometimes active, sometimes inactive,
# under the control of a time feature and outputs only at end of session.
# Third scorer is always active and outputs only at end of session.

s:Ge/Box/Type = "TsBox"
s:Ge/Box/Parent = "World"
s:Ge/Box/Material = "G4_WATER"
d:Ge/Box/HLX = 1. m
d:Ge/Box/HLY = 1. m
d:Ge/Box/HLZ = .4 m

```

(continues on next page)

(continued from previous page)

```

s:Sc/GatedAndOutputPerRun/Quantity          = "Fluence"
s:Sc/GatedAndOutputPerRun/Component          = "Box"
s:Sc/GatedAndOutputPerRun/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/GatedAndOutputPerRun/OutputToConsole    = "True"
b:Sc/GatedAndOutputPerRun/Active              = Tf/ScoringOnOff/Value
b:Sc/GatedAndOutputPerRun/OutputAfterRun     = "True"

s:Sc/GatedAndOutputPerSession/Quantity       = "Fluence"
s:Sc/GatedAndOutputPerSession/Component      = "Box"
s:Sc/GatedAndOutputPerSession/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/GatedAndOutputPerSession/OutputToConsole = "True"
b:Sc/GatedAndOutputPerSession/Active         = Tf/ScoringOnOff/Value

s:Sc/NotGated/Quantity                      = "Fluence"
s:Sc/NotGated/Component                    = "Box"
s:Sc/NotGated/IfOutputFileAlreadyExists    = "Overwrite"
b:Sc/NotGated/OutputToConsole              = "True"

s:Gr/MyOGL/Type                            = "OpenGL"
i:Gr/MyOGL/WindowSizeX                     = 600
i:Gr/MyOGL/WindowSizeY                     = 600
i:Gr/MyOGL/WindowPosX                      = 0
i:Gr/MyOGL/WindowPosY                      = 0

s:Tf/ScoringOnOff/Function                  = "Step"
dv:Tf/ScoringOnOff/Times                    = 10 10 20 30 40 50 60 70 80 90 100 ms
bv:Tf/ScoringOnOff/Values                   = 10 "true" "false" "true" "false" "false" "true" "true"
↪ "true" "false" "true"

i:Tf/Verbosity                             = 1
d:Tf/TimelineEnd                           = 100.0 ms
i:Tf/NumberOfSequentialTimes               = 10

s:So/Example/Type                          = "Beam"
s:So/Example/Component                     = "BeamPosition"
s:So/Example/BeamParticle                  = "proton"
d:So/Example/BeamEnergy                    = 169.23 MeV
u:So/Example/BeamEnergySpread              = 0.757504
s:So/Example/BeamPositionDistribution       = "Gaussian"
s:So/Example/BeamPositionCutoffShape       = "Ellipse"
d:So/Example/BeamPositionCutoffX           = 10. cm
d:So/Example/BeamPositionCutoffY           = 10. cm
d:So/Example/BeamPositionSpreadX           = 0.65 cm
d:So/Example/BeamPositionSpreadY           = 0.65 cm
s:So/Example/BeamAngularDistribution        = "Gaussian"
d:So/Example/BeamAngularCutoffX            = 90. deg
d:So/Example/BeamAngularCutoffY            = 90. deg
d:So/Example/BeamAngularSpreadX            = 0.0032 rad
d:So/Example/BeamAngularSpreadY            = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun        = 10

b:Ts/PauseBeforeQuit                       = "True"

```

## 26.16 GeometryDivisions.txt

```
# Demonstrates scoring by various geometry divisions.
# First scorer uses geometry of the regular world.
# Second scorer uses a parallel geometry with 2 X bins.
$ Third scorer uses a parallel geoemtry with 2 Y bins.

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Phantom/Type    = "TsBox"
s:Ge/Phantom/Parent  = "World"
s:Ge/Phantom/Material = "G4_WATER"
d:Ge/Phantom/HLX     = 30.0 cm
d:Ge/Phantom/HLY     = 30.0 cm
d:Ge/Phantom/HLZ     = 1.1 cm
d:Ge/Phantom/TransX   = 0. cm
d:Ge/Phantom/TransY   = 0. cm
d:Ge/Phantom/TransZ   = -30. cm
d:Ge/Phantom/RotX     = 0. deg
d:Ge/Phantom/RotY     = 0. deg
d:Ge/Phantom/RotZ     = 0. deg
s:Ge/Phantom/Color    = "blue"

# Scores in regular world
s:Sc/DoseAtPhantom/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantom/Component     = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"

# X binning causes creation of a parallel world for scoring
s:Sc/DoseAtPhantomParallel/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantomParallel/Component     = "Phantom"
b:Sc/DoseAtPhantomParallel/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantomParallel/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/DoseAtPhantomParallel/XBins         = 2

# Y binning causes creation of a parallel world for scoring
s:Sc/DoseAtPhantomParallel2/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantomParallel2/Component     = "Phantom"
b:Sc/DoseAtPhantomParallel2/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantomParallel2/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/DoseAtPhantomParallel2/YBins         = 2

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi         = 20 deg
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom        = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
sv:Gr/ViewA/VisibleWorlds = 1 "All"
```

(continues on next page)

(continued from previous page)

```

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 20

b:Ts/PauseBeforeQuit = "True"

```

## 26.17 Histograms.txt

```

# Create a variety of energy histograms

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX = 2.0 m
d:Ge/World/HLY = 2.0 m
d:Ge/World/HLZ = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/Phantom/Type = "TsBox"
s:Ge/Phantom/Parent = "World"
s:Ge/Phantom/Material = "Lead"
d:Ge/Phantom/HLX = 30.0 cm
d:Ge/Phantom/HLY = 30.0 cm
d:Ge/Phantom/HLZ = 10.0 cm
d:Ge/Phantom/TransX = 0. cm
d:Ge/Phantom/TransY = 0. cm
d:Ge/Phantom/TransZ = -30. cm
d:Ge/Phantom/RotX = 0. deg
d:Ge/Phantom/RotY = 0. deg
d:Ge/Phantom/RotZ = 0. deg
s:Ge/Phantom/Color = "blue"

s:Sc/DoseAtPhantom/Quantity = "EnergyDeposit"
s:Sc/DoseAtPhantom/Component = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom/OutputType = "Root"
i:Sc/DoseAtPhantom/HistogramBins = 10
d:Sc/DoseAtPhantom/HistogramMin = 0. MeV
d:Sc/DoseAtPhantom/HistogramMax = 200. MeV

s:Sc/DoseAtPhantomByX/Quantity = "EnergyDeposit"

```

(continues on next page)

(continued from previous page)

```

s:Sc/DoseAtPhantomByX/Component           = "Phantom"
b:Sc/DoseAtPhantomByX/OutputToConsole      = "TRUE"
s:Sc/DoseAtPhantomByX/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantomByX/OutputType           = "Root"
i:Sc/DoseAtPhantomByX/XBins                = 10
i:Sc/DoseAtPhantomByX/HistogramBins        = 10
d:Sc/DoseAtPhantomByX/HistogramMin         = 0. MeV
d:Sc/DoseAtPhantomByX/HistogramMax        = 200. MeV

s:Sc/DoseAtPhantomByY/Quantity             = "EnergyDeposit"
s:Sc/DoseAtPhantomByY/Component            = "Phantom"
b:Sc/DoseAtPhantomByY/OutputToConsole      = "TRUE"
s:Sc/DoseAtPhantomByY/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantomByY/OutputType           = "Root"
i:Sc/DoseAtPhantomByY/YBins                = 10
i:Sc/DoseAtPhantomByY/HistogramBins        = 10
d:Sc/DoseAtPhantomByY/HistogramMin         = 0. MeV
d:Sc/DoseAtPhantomByY/HistogramMax        = 200. MeV

s:Sc/DoseAtPhantomByZ/Quantity             = "EnergyDeposit"
s:Sc/DoseAtPhantomByZ/Component            = "Phantom"
b:Sc/DoseAtPhantomByZ/OutputToConsole      = "TRUE"
s:Sc/DoseAtPhantomByZ/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantomByZ/OutputType           = "Root"
i:Sc/DoseAtPhantomByZ/ZBins                = 10
i:Sc/DoseAtPhantomByZ/HistogramBins        = 10
d:Sc/DoseAtPhantomByZ/HistogramMin         = 0. MeV
d:Sc/DoseAtPhantomByZ/HistogramMax        = 200. MeV

s:Sc/DoseAtPhantomByE/Quantity             = "EnergyDeposit"
s:Sc/DoseAtPhantomByE/Component            = "Phantom"
b:Sc/DoseAtPhantomByE/OutputToConsole      = "TRUE"
s:Sc/DoseAtPhantomByE/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantomByE/OutputType           = "Root"
i:Sc/DoseAtPhantomByE/EBins                = 10
d:Sc/DoseAtPhantomByE/EBinMax              = 200. MeV
i:Sc/DoseAtPhantomByE/HistogramBins        = 10
d:Sc/DoseAtPhantomByE/HistogramMin         = 0. MeV
d:Sc/DoseAtPhantomByE/HistogramMax        = 200. MeV

#s:Gr/ViewA/Type                          = "OpenGL"
i:Gr/ViewA/WindowSizeX                    = 900
i:Gr/ViewA/WindowSizeY                    = 900
d:Gr/ViewA/Theta                          = 55 deg
d:Gr/ViewA/Phi                            = 20 deg
s:Gr/ViewA/Projection                     = "Perspective"
d:Gr/ViewA/PerspectiveAngle               = 30 deg
u:Gr/ViewA/Zoom                           = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type                        = "Beam"
s:So/Example/Component                   = "BeamPosition"
s:So/Example/BeamParticle                = "proton"
d:So/Example/BeamEnergy                   = 169.23 MeV
u:So/Example/BeamEnergySpread             = 0.757504
s:So/Example/BeamPositionDistribution     = "Gaussian"
s:So/Example/BeamPositionCutoffShape      = "Ellipse"

```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 10000

i:Ts/ShowHistoryCountAtInterval = 100

```

## 26.18 Inactive.txt

```

# Demonstrate effect of setting a scorer inactive

s:Ge/Box/Type      = "TsBox"
s:Ge/Box/Parent    = "World"
s:Ge/Box/Material  = "G4_WATER"
d:Ge/Box/HLX       = 1. m
d:Ge/Box/HLY       = 1. m
d:Ge/Box/HLZ       = .4 m

s:Sc/WasActive/Quantity      = "Fluence"
s:Sc/WasActive/Component     = "Box"
s:Sc/WasActive/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/WasActive/OutputToConsole = "True"

s:Sc/WasInactive/Quantity    = "Fluence"
s:Sc/WasInactive/Component   = "Box"
s:Sc/WasInactive/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/WasInactive/OutputToConsole = "True"
b:Sc/WasInactive/Active      = "False"

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowPosX = 600
i:Gr/MyOGL/WindowPosY = 600
i:Gr/MyOGL/WindowPosZ = 0
i:Gr/MyOGL/WindowPosY = 0

s:So/Example/Type      = "Beam"
s:So/Example/Component  = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy  = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg

```

(continues on next page)



(continued from previous page)

```

d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 100

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 26.19 Ion.txt

```

# Test filtering on ions

s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/BeamPosition/Parent = "World"
s:Ge/BeamPosition/Type   = "Group"
d:Ge/BeamPosition/TransX = 0.0 cm
d:Ge/BeamPosition/TransY = 0.0 cm
d:Ge/BeamPosition/TransZ = Ge/World/HLZ m
d:Ge/BeamPosition/RotX   = 180. deg
d:Ge/BeamPosition/RotY   = 0. deg
d:Ge/BeamPosition/RotZ   = 0. deg

s:Ge/Phantom/Type       = "TsBox"
s:Ge/Phantom/Parent     = "World"
s:Ge/Phantom/Material   = "G4_WATER"
d:Ge/Phantom/HLX        = 5.0 cm
d:Ge/Phantom/HLY        = 5.0 cm
d:Ge/Phantom/HLZ        = 5.0 cm
d:Ge/Phantom/TransX     = 0. cm
d:Ge/Phantom/TransY     = 0. cm
d:Ge/Phantom/TransZ     = 0. cm
d:Ge/Phantom/RotX       = 0. deg
d:Ge/Phantom/RotY       = 0.0 deg
d:Ge/Phantom/RotZ       = 0. deg
s:Ge/Phantom/Color      = "blue"
i:Ge/Phantom/XBins      = 1
i:Ge/Phantom/YBins      = 1
i:Ge/Phantom/ZBins      = 1

s:Sc/AllOfAtomicNumber6/Quantity      = "EnergyDeposit"
s:Sc/AllOfAtomicNumber6/Component     = "Phantom"
sv:Sc/AllOfAtomicNumber6/Report       = 3 "Mean" "Count_In_Bin"
→ "Standard_Deviation"
b:Sc/AllOfAtomicNumber6/OutputToConsole = "True"
s:Sc/AllOfAtomicNumber6/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/AllOfAtomicNumber6/OnlyIncludeParticlesOfAtomicNumber = 6

s:Sc/IonsWithWildCards/Quantity      = "EnergyDeposit"
s:Sc/IonsWithWildCards/Component     = "Phantom"

```

(continues on next page)

(continued from previous page)

```

sv:Sc/IonsWithWildCards/Report                = 3 "Mean" "Count_In_Bin"
  ↳"Standard_Deviation"
b:Sc/IonsWithWildCards/OutputToConsole         = "True"
s:Sc/IonsWithWildCards/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/IonsWithWildCards/OnlyIncludeParticlesNamed = 1 "GenericIon(6,*,*)"

s:Sc/IonsWithCharge3to5/Quantity               = "EnergyDeposit"
s:Sc/IonsWithCharge3to5/Component              = "Phantom"
sv:Sc/IonsWithCharge3to5/Report                = 3 "Mean" "Count_In_Bin"
  ↳"Standard_Deviation"
b:Sc/IonsWithCharge3to5/OutputToConsole         = "True"
s:Sc/IonsWithCharge3to5/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/IonsWithCharge3to5/OnlyIncludeParticlesNamed = 3 "GenericIon(6,12,3)"
  ↳"GenericIon(6,12,4)" "GenericIon(6,12,5)"

s:So/Example/Type                             = "Beam"
s:So/Example/Component                        = "BeamPosition"
s:So/Example/BeamParticle                    = "GenericIon(6,12)"
d:So/Example/BeamEnergy                      = 100.0 MeV
u:So/Example/BeamEnergySpread                 = 0.0
s:So/Example/BeamPositionDistribution         = "None"
s:So/Example/BeamAngularDistribution          = "None"
i:So/Example/NumberOfHistoriesInRun           = 10000

i:Ts/ShowHistoryCountAtInterval = 1000

```

## 26.20 Origin.txt

```

# Particles pass through Film1Container which contains a divided box, Film1.
# We score fluence in a phantom, filtering several different ways based on
# origin of the scored particle.

s:Ge/Film1Container/Type      = "TsBox"
s:Ge/Film1Container/Parent    = "World"
s:Ge/Film1Container/Material  = "G4_WATER"
d:Ge/Film1Container/HLX       = 2.1 m
d:Ge/Film1Container/HLY       = 2.1 m
d:Ge/Film1Container/HLZ       = .15 m
d:Ge/Film1Container/TransZ    = 1. m

s:Ge/Film1/Type              = "TsBox"
s:Ge/Film1/Parent            = "Film1Container"
s:Ge/Film1/Material           = "G4_WATER"
d:Ge/Film1/HLX               = 2. m
d:Ge/Film1/HLY               = 2. m
d:Ge/Film1/HLZ               = .1 m
i:Ge/Film1/XBins              = 2
i:Ge/Film1/YBins              = 2
i:Ge/Film1/ZBins              = 2

s:Ge/Phantom/Type            = "TsBox"
s:Ge/Phantom/Parent          = "World"
s:Ge/Phantom/Material         = "G4_WATER"
d:Ge/Phantom/HLX              = 3. m

```

(continues on next page)

(continued from previous page)

```

d:Ge/Phantom/HLY      = 3. m
d:Ge/Phantom/HLZ      = 0.5 m
d:Ge/Phantom/TransZ   = -1. m
i:Ge/Phantom/XBins    = 1
i:Ge/Phantom/YBins    = 1
i:Ge/Phantom/ZBins    = 1

s:Sc/ParticlesFromFilm1ContainerAndSubcomponents/Quantity      = "Fluence"
s:Sc/ParticlesFromFilm1ContainerAndSubcomponents/Component     = "Phantom"
s:Sc/ParticlesFromFilm1ContainerAndSubcomponents/IfOutputFileAlreadyExists =
  ↳ "Overwrite"
b:Sc/ParticlesFromFilm1ContainerAndSubcomponents/OutputToConsole = "True"
sv:Sc/ParticlesFromFilm1ContainerAndSubcomponents/
  ↳ OnlyIncludeParticlesFromComponentOrSubComponentsOf = 1 "Film1Container"

s:Sc/ParticlesFromFilm1ContainerButNotSubcomponents/Quantity   =
  ↳ "Fluence"
s:Sc/ParticlesFromFilm1ContainerButNotSubcomponents/Component   =
  ↳ "Phantom"
s:Sc/ParticlesFromFilm1ContainerButNotSubcomponents/IfOutputFileAlreadyExists =
  ↳ "Overwrite"
b:Sc/ParticlesFromFilm1ContainerButNotSubcomponents/OutputToConsole = "True"
sv:Sc/ParticlesFromFilm1ContainerButNotSubcomponents/
  ↳ OnlyIncludeParticlesFromComponent = 1 "Film1Container"

s:Sc/ParticlesFromFilm1/Quantity      = "Fluence"
s:Sc/ParticlesFromFilm1/Component     = "Phantom"
s:Sc/ParticlesFromFilm1/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/ParticlesFromFilm1/OutputToConsole = "True"
sv:Sc/ParticlesFromFilm1/OnlyIncludeParticlesFromComponentOrSubComponentsOf = 1 "Film1
  ↳ "

s:Gr/MyOGL/Type      = "OpenGL"
i:Gr/MyOGL/WindowSizeX = 600
i:Gr/MyOGL/WindowSizeY = 600
i:Gr/MyOGL/WindowPosX = 0
i:Gr/MyOGL/WindowPosY = 0

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 400 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

b:Ts/PauseBeforeQuit = "True"

```

## 26.21 ParallelBoxRebinned.txt

```
# Score in a variety of x, y and z divisions.
# Topas creates parallel worlds as needed to accomplish this.

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"

s:Ge/MassWorldBox/Parent      = "World"
s:Ge/MassWorldBox/Type        = "Group"
s:Ge/MassWorldBox/Material    = "G4_WATER"
b:Ge/MassWorldBox/IsParallel   = "FALSE"
d:Ge/MassWorldBox/HLX         = 9.0 cm
d:Ge/MassWorldBox/HLY         = 9.0 cm
d:Ge/MassWorldBox/HLZ         = 9.0 cm
s:Ge/MassWorldBox/Color       = "blue"

s:Ge/ParallelWorldBox/Parent   = "MassWorldBox"
s:Ge/ParallelWorldBox/Type      = "TsBox"
b:Ge/ParallelWorldBox/IsParallel = "TRUE"
d:Ge/ParallelWorldBox/HLX       = 6.0 cm
d:Ge/ParallelWorldBox/HLY       = 5.0 cm
d:Ge/ParallelWorldBox/HLZ       = 4.0 cm
s:Ge/ParallelWorldBox/Color     = "green"
i:Ge/ParallelWorldBox/XBins     = 1
i:Ge/ParallelWorldBox/YBins     = 2
i:Ge/ParallelWorldBox/ZBins     = 3

s:Sc/Score2by3by4/Quantity      = "DoseToMedium"
sv:Sc/Score2by3by4/OnlyIncludeParticlesFromProcess = 1 "eBrem"
s:Sc/Score2by3by4/Component      = "ParallelWorldBox"
b:Sc/Score2by3by4/OutputToConsole = "TRUE"
s:Sc/Score2by3by4/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Score2by3by4/XBins          = 2
i:Sc/Score2by3by4/YBins          = 3
i:Sc/Score2by3by4/ZBins          = 4

s:Sc/Score3by4by5/Quantity      = "DoseToMedium"
sv:Sc/Score3by4by5/OnlyIncludeParticlesFromProcess = 1 "hIoni"
s:Sc/Score3by4by5/Component      = "ParallelWorldBox"
b:Sc/Score3by4by5/OutputToConsole = "TRUE"
s:Sc/Score3by4by5/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Score3by4by5/XBins          = 3
i:Sc/Score3by4by5/YBins          = 4
i:Sc/Score3by4by5/ZBins          = 5

s:Sc/Score4by5by6/Quantity      = "DoseToMedium"
sv:Sc/Score4by5by6/OnlyIncludeParticlesFromProcess = 1 "hIoni"
s:Sc/Score4by5by6/Component      = "ParallelWorldBox"
b:Sc/Score4by5by6/OutputToConsole = "TRUE"
s:Sc/Score4by5by6/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/Score4by5by6/XBins          = 4
i:Sc/Score4by5by6/YBins          = 5
i:Sc/Score4by5by6/ZBins          = 6
```

(continues on next page)

(continued from previous page)

```

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 1024
i:Gr/ViewA/WindowSizeY = 768
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 50.
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 200

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 26.22 SplitByTimeFeature.txt

```

# Demonstrates how scoring can be split by a time feature.
# In this example, though the time feature ImageName doesn't really get # used to_
↳load any images,
# the scoring is still split by this time feature.
# This means that separate scorers are created for each of the
# three values of ImageName, and each scorer is active only when
# the ImageName value matches the that scorer's required value.

s:Ge/Phantom/Type = "TsBox"
s:Ge/Phantom/Parent = "World"
s:Ge/Phantom/Material = "G4_WATER"
d:Ge/Phantom/HLX = 80.0 cm
d:Ge/Phantom/HLY = 80.0 cm
d:Ge/Phantom/HLZ = 80.0 cm
d:Ge/Phantom/TransX = 0. cm
d:Ge/Phantom/TransY = 0. cm
d:Ge/Phantom/TransZ = -30. cm
d:Ge/Phantom/RotX = 0. deg
d:Ge/Phantom/RotY = 0. deg
d:Ge/Phantom/RotZ = 0. deg

```

(continues on next page)

(continued from previous page)

```

s:Sc/DoseAtPhantom/Quantity           = "DoseToMedium"
s:Sc/DoseAtPhantom/Component          = "Phantom"
b:Sc/DoseAtPhantom/OutputToConsole    = "TRUE"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom/SplitByTimeFeature  = "ImageName"

# This time feature will take one of three string values.
s:Tf/ImageName/Function               = "Step"
dv:Tf/ImageName/Times                 = 3 10 20 30 ms
sv:Tf/ImageName/Values                = 3 "lung1" "lung2" "lung3"
d:Tf/TimelineEnd                      = 40.0 ms
i:Tf/NumberOfSequentialTimes          = 4
i:Tf/Verbosity                        = 1

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type                     = "Beam"
s:So/Example/Component                = "BeamPosition"
s:So/Example/BeamParticle             = "proton"
d:So/Example/BeamEnergy               = 169.23 MeV
u:So/Example/BeamEnergySpread         = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape  = "Ellipse"
d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 100

s:Gr/View/Type = "OpenGL"

```

## 26.23 Surfaces.txt

```

# Test of scoring on various surfaces

d:Ge/World/HLX           = 2.0 m
d:Ge/World/HLY           = 2.0 m
d:Ge/World/HLZ           = 2.0 m
b:Ge/World/Invisible     = "TRUE"

s:Ge/TestBox/Type        = "TsBox"
s:Ge/TestBox/Parent      = "World"
s:Ge/TestBox/Material    = "G4_WATER"
d:Ge/TestBox/HLX         = 20.0 cm
d:Ge/TestBox/HLY         = 15.0 cm
d:Ge/TestBox/HLZ         = 10.0 cm
s:Ge/TestBox/Color       = "blue"

s:Ge/TestSphere/Type     = "TsSphere"

```

(continues on next page)

(continued from previous page)

```

s:Ge/TestSphere/Parent      = "World"
s:Ge/TestSphere/Material    = "G4_WATER"
d:Ge/TestSphere/RMin       = 5.0 cm
d:Ge/TestSphere/RMax       = 7.0 cm
d:Ge/TestSphere/SPhi       = 0. deg
d:Ge/TestSphere/DPhi       = 360. deg
d:Ge/TestSphere/STheta     = 0. deg
d:Ge/TestSphere/DTheta     = 180. deg
d:Ge/TestSphere/TransZ     = -20. cm
s:Ge/TestSphere/Color      = "blue"

s:Ge/TestCylinder/Type      = "TsCylinder"
s:Ge/TestCylinder/Parent    = "World"
s:Ge/TestCylinder/Material  = "G4_WATER"
d:Ge/TestCylinder/RMin     = 0.0 cm
d:Ge/TestCylinder/RMax     = 8.0 cm
d:Ge/TestCylinder/HL       = 10.0 cm
d:Ge/TestCylinder/SPhi     = 0. deg
d:Ge/TestCylinder/DPhi     = 360. deg
d:Ge/TestCylinder/TransZ   = -40. cm
s:Ge/TestCylinder/Color    = "blue"

s:Sc/BoxXPlus/Quantity      = "SurfaceTrackCount"
s:Sc/BoxXPlus/Surface       = "TestBox/XPlusSurface"
b:Sc/BoxXPlus/OutputToConsole = "TRUE"

s:Sc/BoxXMinus/Quantity     = "SurfaceTrackCount"
s:Sc/BoxXMinus/Surface      = "TestBox/XMinusSurface"
b:Sc/BoxXMinus/OutputToConsole = "TRUE"

s:Sc/BoxYPlus/Quantity      = "SurfaceTrackCount"
s:Sc/BoxYPlus/Surface       = "TestBox/YPlusSurface"
b:Sc/BoxYPlus/OutputToConsole = "TRUE"

s:Sc/BoxYMinus/Quantity     = "SurfaceTrackCount"
s:Sc/BoxYMinus/Surface      = "TestBox/YMinusSurface"
b:Sc/BoxYMinus/OutputToConsole = "TRUE"

s:Sc/BoxZPlus/Quantity      = "SurfaceTrackCount"
s:Sc/BoxZPlus/Surface       = "TestBox/ZPlusSurface"
b:Sc/BoxZPlus/OutputToConsole = "TRUE"

s:Sc/BoxZMinus/Quantity     = "SurfaceTrackCount"
s:Sc/BoxZMinus/Surface      = "TestBox/ZMinusSurface"
b:Sc/BoxZMinus/OutputToConsole = "TRUE"

s:Sc/SphereInner/Quantity   = "SurfaceTrackCount"
s:Sc/SphereInner/Surface    = "TestSphere/InnerCurvedSurface"
b:Sc/SphereInner/OutputToConsole = "TRUE"

s:Sc/SphereOuter/Quantity   = "SurfaceTrackCount"
s:Sc/SphereOuter/Surface    = "TestSphere/OuterCurvedSurface"
b:Sc/SphereOuter/OutputToConsole = "TRUE"

s:Sc/CylinderZPlus/Quantity  = "SurfaceTrackCount"
s:Sc/CylinderZPlus/Surface   = "TestCylinder/ZPlusSurface"

```

(continues on next page)

(continued from previous page)

```

b:Sc/CylinderZPlus/OutputToConsole = "TRUE"

s:Sc/CylinderZMinus/Quantity      = "SurfaceTrackCount"
s:Sc/CylinderZMinus/Surface       = "TestCylinder/ZMinusSurface"
b:Sc/CylinderZMinus/OutputToConsole = "TRUE"

s:Sc/CylinderInner/Quantity       = "SurfaceTrackCount"
s:Sc/CylinderInner/Surface        = "TestCylinder/InnerCurvedSurface"
b:Sc/CylinderInner/OutputToConsole = "TRUE"

s:Sc/CylinderOuter/Quantity       = "SurfaceTrackCount"
s:Sc/CylinderOuter/Surface        = "TestCylinder/OuterCurvedSurface"
b:Sc/CylinderOuter/OutputToConsole = "TRUE"

s:Gr/ViewA/Type = "OpenGL"

s:So/Example/Type                = "Beam"
s:So/Example/Component            = "BeamPosition"
s:So/Example/BeamParticle         = "proton"
d:So/Example/BeamEnergy           = 169.23 MeV
u:So/Example/BeamEnergySpread     = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX  = 10. cm
d:So/Example/BeamPositionCutoffY  = 10. cm
d:So/Example/BeamPositionSpreadX  = 0.65 cm
d:So/Example/BeamPositionSpreadY  = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX   = 90. deg
d:So/Example/BeamAngularCutoffY   = 90. deg
d:So/Example/BeamAngularSpreadX   = 0.0032 rad
d:So/Example/BeamAngularSpreadY   = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1000

i:Ts/ShowHistoryCountAtInterval = 100
b:Ts/PauseBeforeQuit = "True"

```

## 26.24 VoxelMaterials.txt

```

sv:Ge/Phantom/VoxelMaterials = 1000
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"

```

(continues on next page)



[illegible]

(continued from previous page)

[illegible]

(continues on next page)

(continued from previous page)

```
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
"G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER" "G4_WATER"
↳ "G4_WATER" "G4_WATER"
```



## 27.1 CAD.txt

```
# Builds components on-the-fly from information in CAD files.  
# The first component is built from the PLY format.  
# The second component is built from the STL format.
```

```
d:Ge/World/HLX = 10 cm  
d:Ge/World/HLY = 10 cm  
d:Ge/World/HLZ = 10 cm  
  
d:Ge/BeamPosition/TransZ = -7 cm  
d:Ge/BeamPosition/RotX   = 0 deg  
  
s:Ge/CADVolumePly/Parent      = "World"  
s:Ge/CADVolumePly/Material    = "Brass"  
s:Ge/CADVolumePly/Type        = "TsCAD"  
s:Ge/CADVolumePly/InputFile   = "Aperture_cm"  
s:Ge/CADVolumePly/FileFormat  = "ply"  
d:Ge/CADVolumePly/Units       = 0.5 cm  
d:Ge/CADVolumePly/TransX      = -4 cm  
d:Ge/CADVolumePly/TransY      = -4 cm  
d:Ge/CADVolumePly/TransZ      = -6.5 cm  
d:Ge/CADVolumePly/RotX        = 0 deg  
d:Ge/CADVolumePly/RotY        = 0 deg  
d:Ge/CADVolumePly/RotZ        = 0 deg  
  
s:Ge/CADVolumeStl/Parent      = "World"  
s:Ge/CADVolumeStl/Material    = "Lead"  
s:Ge/CADVolumeStl/Type        = "TsCAD"  
s:Ge/CADVolumeStl/InputFile   = "Aperture_cm"  
s:Ge/CADVolumeStl/FileFormat  = "stl"  
d:Ge/CADVolumeStl/Units       = 0.5 cm  
d:Ge/CADVolumeStl/TransX      = -4 cm
```

(continues on next page)

(continued from previous page)

```

d:Ge/CADVolumeStl/TransY      = -4 cm
d:Ge/CADVolumeStl/TransZ      = 0 cm
d:Ge/CADVolumeStl/RotX        = 0 deg
d:Ge/CADVolumeStl/RotY        = 0 deg
d:Ge/CADVolumeStl/RotZ        = 0 deg

s:Gr/view/Type                 = "OpenGL"
d:Gr/view/Theta                 = 45 deg
d:Gr/view/Phi                   = 45 deg
u:Gr/view/Zoom                  = 1.5
b:Gr/view/IncludeAxes           = "True"
d:Gr/view/AxesSize              = 8 cm

s:So/Example/Type               = "Beam"
s:So/Example/Component          = "BeamPosition"
s:So/Example/BeamParticle       = "proton"
d:So/Example/BeamEnergy         = 150 MeV
u:So/Example/BeamEnergySpread   = 0.757504
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 3 cm
d:So/Example/BeamPositionCutoffY = 3 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 30

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/PauseBeforeQuit = "True"

```

## 27.2 DipoleMagnet.txt

```

# Dipole magnet

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Nozzle/Type     = "Group"
s:Ge/Nozzle/Parent   = "World"

s:Ge/Dipole/Type      = "TsBox"
s:Ge/Dipole/Parent    = "Nozzle"
s:Ge/Dipole/Material   = "G4_AIR"
d:Ge/Dipole/HLX        = 10 cm
d:Ge/Dipole/HLY        = 10 cm
d:Ge/Dipole/HLZ        = 20 cm
s:Ge/Dipole/Field      = "DipoleMagnet"
u:Ge/Dipole/MagneticFieldDirectionX = 0.0

```

(continues on next page)

(continued from previous page)

```

u:Ge/Dipole/MagneticFieldDirectionY = 1.0
u:Ge/Dipole/MagneticFieldDirectionZ = 0.0
d:Ge/Dipole/MagneticFieldStrength   = 3.0 tesla

s:Ge/InnerBox/Type      = "TsBox"
s:Ge/InnerBox/Parent    = "Dipole"
s:Ge/InnerBox/Material  = "G4_AIR"
d:Ge/InnerBox/HLX       = 9 cm
d:Ge/InnerBox/HLY       = 9 cm
d:Ge/InnerBox/HLZ       = 18 cm

d:Ge/BeamPosition/TransZ = Ge/World/HLZ cm
d:Ge/BeamPosition/RotX   = 180. deg

s:So/Example/Type        = "Beam"
s:So/Example/Component   = "BeamPosition"
s:So/Example/BeamParticle = "chargedgeantino"
d:So/Example/BeamEnergy   = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.0
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 5.0 cm
d:So/Example/BeamPositionCutoffY = 5.0 cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 1

s:Gr/ViewA/Type          = "OpenGL"
i:Gr/ViewA/WindowSizeX   = 1024
i:Gr/ViewA/WindowSizeY   = 768
d:Gr/ViewA/Theta         = 55 deg
d:Gr/ViewA/Phi           = 20 deg
s:Gr/ViewA/Projection     = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom           = 1.
b:Gr/ViewA/IncludeStepPoints = "True"

b:Ts/PauseBeforeQuit = "True"

```

## 27.3 MultiLeafCollimator.txt

```

# Multileaf Collimator

d:Ge/World/HLX      = 2. m
d:Ge/World/HLY      = 2. m
d:Ge/World/HLZ      = 2. m
s:Ge/World/Material  = "G4_Galactic"
b:Ge/World/Invisible = "TRUE"

#####
# Multi-leaf Collimator
#####

```

(continues on next page)

(continued from previous page)

```

s:Ge/MultiLeafCollimatorA/Type           = "TsMultiLeafCollimator"
s:Ge/MultiLeafCollimatorA/Parent         = "World"
s:Ge/MultiLeafCollimatorA/Material       = "Aluminum"
d:Ge/MultiLeafCollimatorA/TransX         = 0.0 cm
d:Ge/MultiLeafCollimatorA/TransY         = 0.0 cm
d:Ge/MultiLeafCollimatorA/TransZ         = 0.0 cm
d:Ge/MultiLeafCollimatorA/RotX           = 0.0 deg
d:Ge/MultiLeafCollimatorA/RotY           = 0.0 deg
d:Ge/MultiLeafCollimatorA/RotZ           = 0.0 deg
s:Ge/MultiLeafCollimatorA/DrawingStyle   = "Solid"
b:Ge/MultiLeafCollimatorA/PrintInformation = "True"

d:Ge/MultiLeafCollimatorA/MaximumLeafOpen = 5.0 cm
d:Ge/MultiLeafCollimatorA/Thickness       = 5.0 cm
d:Ge/MultiLeafCollimatorA/Length         = 6.0 cm
dv:Ge/MultiLeafCollimatorA/Widths        = 5 1.5 0.5 0.5 0.5 1.5 cm
dv:Ge/MultiLeafCollimatorA/XMinusLeavesOpen = 5 0.0 -0.3 -0.2 -0.5 0.0 cm
dv:Ge/MultiLeafCollimatorA/XPlusLeavesOpen = 5 0.0 0.3 0.2 0.5 0.0 cm

#####
# Physics and Beam Section
#####
s:So/Example/Type           = "Beam"
s:So/Example/Component       = "BeamPosition"
s:So/Example/BeamParticle    = "proton"
d:So/Example/BeamEnergy      = 230.0 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 1.0 cm
d:So/Example/BeamPositionSpreadY = 1.0 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 5

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
b:Ts/PauseBeforeQuit = "True"

s:Gr/ViewA/Type           = "OpenGL"
i:Gr/ViewA/WindowSizeX    = 600
i:Gr/ViewA/WindowSizeY    = 600
d:Gr/ViewA/Theta           = 55 deg
d:Gr/ViewA/Phi             = 20 deg
s:Gr/ViewA/Projection      = "Orthogonal"
d:Gr/ViewA/PerspectiveAngle = 20 deg
u:Gr/ViewA/Zoom            = 1.6
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

b:Gr/ViewA/IncludeAxes     = "true"
s:Gr/ViewA/AxesComponent    = "MultiLeafCollimatorA"
d:Gr/ViewA/AxesSize        = 0.1 m

```



## 27.4 MultiLeafCollimator\_sequence.txt

```
# Multileaf collimator moving in a time sequence

includeFile = MultiLeafCollimator.txt

dv:Ge/MultiLeafCollimatorA/XPlusLeavesOpen = 5 0.0 Tf/LeafXPlus1/Value Tf/LeafXPlus0/
↳Value Tf/LeafXPlus2/Value 0.0 cm
dv:Ge/MultiLeafCollimatorA/XMinusLeavesOpen = 5 0.0 Tf/LeafXMinus1/Value Tf/
↳LeafXMinus0/Value Tf/LeafXMinus2/Value 0.0 cm

s:Tf/LeafXPlus0/Function = "Linear cm"
d:Tf/LeafXPlus0/Rate = 0.02 cm/ms
d:Tf/LeafXPlus0/StartValue = 0.0 cm
d:Tf/LeafXPlus0/RepetitionInterval = 100.0 ms

s:Tf/LeafXMinus0/Function = "Linear cm"
d:Tf/LeafXMinus0/Rate = 0.01 cm/ms
d:Tf/LeafXMinus0/StartValue = -0.5 cm
d:Tf/LeafXMinus0/RepetitionInterval = 100.0 ms

s:Tf/LeafXPlus1/Function = "Linear cm"
d:Tf/LeafXPlus1/Rate = 0.04 cm/ms
d:Tf/LeafXPlus1/StartValue = 0.5 cm
d:Tf/LeafXPlus1/RepetitionInterval = 100.0 ms

s:Tf/LeafXMinus1/Function = "Linear cm"
d:Tf/LeafXMinus1/Rate = 0.01 cm/ms
d:Tf/LeafXMinus1/StartValue = 0.0 cm
d:Tf/LeafXMinus1/RepetitionInterval = 100.0 ms

s:Tf/LeafXPlus2/Function = "Linear cm"
d:Tf/LeafXPlus2/Rate = -0.02 cm/ms
d:Tf/LeafXPlus2/StartValue = 0.5 cm
d:Tf/LeafXPlus2/RepetitionInterval = 100.0 ms

s:Tf/LeafXMinus2/Function = "Linear cm"
d:Tf/LeafXMinus2/Rate = -0.03 cm/ms
d:Tf/LeafXMinus2/StartValue = 0.0 cm
d:Tf/LeafXMinus2/RepetitionInterval = 100.0 ms

d:Tf/TimelineEnd = 100.0 ms
i:Tf/NumberOfSequentialTimes = 60
i:Tf/Verbosity = 1
b:Ts/PauseBeforeQuit = "False"

i:Gr/ViewA/WindowSizeX = 1024
i:Gr/ViewA/WindowSizeY = 768
u:Gr/ViewA/Zoom = 2.
d:Gr/ViewA/Theta = Tf/Rot1/Value deg
d:Gr/ViewA/Phi = 90 deg + Tf/Rot1/Value
b:Gr/ViewA/CopyOpenGLToEPS = "False"

s:Tf/Rotate/Function = "Linear deg"
d:Tf/Rotate/Rate = 1. deg/ms
d:Tf/Rotate/StartValue = -40.0 deg
d:Tf/Rotate/RepetitionInterval = 100. ms
```

(continues on next page)

(continued from previous page)

```

s:Tf/Rot1/Function      = "Step"
dv:Tf/Rot1/Times        = 2 40 100 ms
dv:Tf/Rot1/Values       = 2 0. Tf/Rotate/Value deg
d:Tf/Rot1/RepetitionInterval = 100. ms

```

## 27.5 MultiWire\_Chamber.txt

```

# Multiwire chamber

d:Ge/World/HLX = 50. cm
d:Ge/World/HLY = 60. cm
d:Ge/World/HLZ = 20. cm

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
b:Ts/PauseBeforeQuit = "True"

s:Ge/WireChamberA/Parent      = "World"
s:Ge/WireChamberA/Type        = "TsMultiWireChamber"
d:Ge/WireChamberA/HLX         = 30.0 cm
d:Ge/WireChamberA/HLY         = 30.0 cm
d:Ge/WireChamberA/HLZ         = 10.0 cm
s:Ge/WireChamberA/Material     = "Air"
d:Ge/WireChamberA/TransX       = 0.0 cm
d:Ge/WireChamberA/TransY       = 0.0 cm
d:Ge/WireChamberA/TransZ       = 0.0 cm
d:Ge/WireChamberA/RotX         = 0.0 deg
d:Ge/WireChamberA/RotY         = 0.0 deg
d:Ge/WireChamberA/RotZ         = 0.0 deg
b:Ge/WireChamberA/PrintInformation = "True"
i:Ge/WireChamberA/NbOfLayers    = 2

d:Ge/WireChamberA/Layer1/RMin   = 0.0 cm
d:Ge/WireChamberA/Layer1/RMax   = 1.0 cm
s:Ge/WireChamberA/Layer1/Material = "Brass"
d:Ge/WireChamberA/Layer1/HL     = 20.0 cm
s:Ge/WireChamberA/Layer1/Align  = "X"
dv:Ge/WireChamberA/Layer1/Displacement = 3 -10 0 10 cm
d:Ge/WireChamberA/Layer1/PosZ   = 5.0 cm
s:Ge/WireChamberA/Layer1/DrawingStyle = "FullWireFrame"
s:Ge/WireChamberA/Layer1/Color  = "red"

d:Ge/WireChamberA/Layer2/RMin   = 0.5 cm
d:Ge/WireChamberA/Layer2/RMax   = 1.0 cm
s:Ge/WireChamberA/Layer2/Material = "Lexan"
d:Ge/WireChamberA/Layer2/HL     = 20.0 cm
s:Ge/WireChamberA/Layer2/Align  = "Y"
dv:Ge/WireChamberA/Layer2/Displacement = 5 -20 -10 0 10 20 cm
d:Ge/WireChamberA/Layer2/PosZ   = -5.0 cm
s:Ge/WireChamberA/Layer2/DrawingStyle = "Solid"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 600
i:Gr/ViewA/WindowSizeY = 600

```

(continues on next page)

(continued from previous page)

```

d:Gr/ViewA/Theta          = 55 deg
d:Gr/ViewA/Phi            = 20 deg
s:Gr/ViewA/Projection      = "Orthogonal"
#s:Gr/ViewA/Projection     = "Perspective"
#d:Gr/ViewA/PerspectiveAngle = 20 deg
u:Gr/ViewA/Zoom            = 2.
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

b:Gr/ViewA/IncludeAxes    = "true"
s:Gr/ViewA/AxesComponent  = "WireChamberA"
d:Gr/ViewA/AxesSize       = 0.1 m

```

## 27.6 Propeller\_ContinuousRotation.txt

```

# Propeller with continuous rotation

includeFile = Propeller.txt

b:Ts/PauseBeforeQuit = "False"

d:Ge/PropellerA/RotZ = Tf/PropellerRot/Value deg

s:Tf/PropellerRot/Function      = "Linear deg"
d:Tf/PropellerRot/Rate          = 3.6 deg/ms
d:Tf/PropellerRot/StartValue    = 0.0 deg
d:Tf/PropellerRot/RepetitionInterval = 100.0 ms

d:Tf/TimelineEnd                = 100.0 ms
i:Tf/NumberOfSequentialTimes    = 100
i:Tf/Verbosity                  = 1

```

## 27.7 Propeller\_StepRotation.txt

```

# Propeller with step rotation

includeFile = Propeller.txt

b:Ts/PauseBeforeQuit = "False"

d:Ge/PropellerA/RotZ      = Tf/PropellerRot/Value deg

s:Tf/PropellerRot/Function = "Step"
dv:Tf/PropellerRot/Times   = 10 10 20 30 40 50 60 70 80 90 100 ms
dv:Tf/PropellerRot/Values  = 10 0 36 72 108 144 180 216 252 288 324 deg

d:Tf/TimelineEnd          = 100.0 ms
i:Tf/NumberOfSequentialTimes = 100
i:So/Example/NumberOfHistoriesInRun = 10
i:Tf/Verbosity             = 1

```

## 27.8 Propeller.txt

```
# Propeller

d:Ge/World/HLX = 30. cm
d:Ge/World/HLY = 30. cm
d:Ge/World/HLZ = 10. cm

s:Ge/PropellerA/Type           = "TsPropeller"
s:Ge/PropellerA/Parent         = "World"
d:Ge/PropellerA/TransX         = 0.0 cm
d:Ge/PropellerA/TransY         = 0.0 cm
d:Ge/PropellerA/TransZ         = 0.0 cm
d:Ge/PropellerA/RotX           = 0.0 deg
d:Ge/PropellerA/RotY           = 0.0 deg
d:Ge/PropellerA/RotZ           = 0.0 deg
b:Ge/PropellerA/Invisible       = "TRUE"
b:Ge/PropellerA/PrintInformation = "True"

i:Ge/PropellerA/NbOfBlades     = 4
d:Ge/PropellerA/Rin            = 10.0 mm
d:Ge/PropellerA/Rout           = 127.5 mm

s:Ge/PropellerA/Blade/Material  = "Parent" #Set to Parent
s:Ge/PropellerA/Blade/Color     = "skyblue"
s:Ge/PropellerA/Blade/DrawingStyle = "Solid"
i:Ge/PropellerA/Blade/VisSegsPerCircle = 360
d:Ge/PropellerA/Blade/MaxStepSize = 10 mm

dv:Ge/PropellerA/Thickness = 10
0.5      0.6      0.7      0.8      0.9      1.0      1.1      1.
↪2      1.3      1.4      mm

dv:Ge/PropellerA/Angles = 10
63.15  54.15  46.450  40.65  35.85  31.8  28.1  24.725  21.8  19.1  deg

sv:Ge/PropellerA/Materials = 10
"Lexan" "G4_WATER" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE"
↪"
"G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_ACETATE" "G4_POLYVINYL_
↪ACETATE" "G4_POLYVINYL_ACETATE"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type           = "Beam"
s:So/Example/Component       = "BeamPosition"
s:So/Example/BeamParticle    = "proton"
d:So/Example/BeamEnergy      = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 2

b:Ts/PauseBeforeQuit = "True"

s:Gr/ViewA/Type      = "OpenGL"
i:Gr/ViewA/WindowSizeX = 600
i:Gr/ViewA/WindowSizeY = 600
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi         = 20 deg
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 20 deg
u:Gr/ViewA/Zoom         = 2.
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

b:Gr/ViewA/IncludeAxes = "true"
s:Gr/ViewA/AxesComponent = "PropellerA"
d:Gr/ViewA/AxesSize     = 0.1 m

```

## 27.9 PurgingMagnet\_move.txt

```

# A purging magnet, with a non-uniform field,
# defined by the Opera3D format file, PurgMag3D.TABLE
# is moved through a beam, such that the beam encounters
# different areas of the field map at different times.
# The effect is seen as a change over time of beam deflection.

d:Ge/World/HLX      = 1.5 m
d:Ge/World/HLY      = 1.5 m
d:Ge/World/HLZ      = 1.5 m
s:Ge/World/Material = "Vacuum"
b:Ge/World/Invisible = "True"

sv:Ma/Iron/Components = 1 "Iron"
uv:Ma/Iron/Fractions  = 1 1.0
d:Ma/Iron/Density      = 7.87 g/cm3
s:Ma/Iron/DefaultColor = "skyblue"

# Nozzle (a group of components)
s:Ge/Nozzle/Type      = "Group"
s:Ge/Nozzle/Parent    = "World"
d:Ge/Nozzle/TransX     = 0. m
d:Ge/Nozzle/TransY     = 0. m
d:Ge/Nozzle/TransZ     = 0. m
d:Ge/Nozzle/RotX       = 0. deg
d:Ge/Nozzle/RotY       = 0. deg
d:Ge/Nozzle/RotZ       = 0. deg

d:Ge/BeamPosition/TransZ = 15. cm
d:Ge/BeamPosition/RotX   = 180. deg

s:So/Example/Type      = "Beam"

```

(continues on next page)

(continued from previous page)

```

s:So/Example/Component           = "BeamPosition"
s:So/Example/BeamParticle        = "e-"
d:So/Example/BeamEnergy          = 50.0 MeV
u:So/Example/BeamEnergySpread    = 0.0
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "None"

s:Ge/MagGroup/Type               = "Group"
s:Ge/MagGroup/Parent             = "Nozzle"
d:Ge/MagGroup/TransX             = 0. m
d:Ge/MagGroup/TransY             = 0. m
d:Ge/MagGroup/TransZ             = 0. m
#Topas will crash, if the following is used
#d:Ge/MagGroup/TransZ = Tf/BackStep/Value mm
d:Ge/MagGroup/RotX               = 0. deg
d:Ge/MagGroup/RotY               = 0. deg
d:Ge/MagGroup/RotZ               = 0. deg

s:Ge/V3DBox/Field3D/Type         = "TsBox"
s:Ge/V3DBox/Field3D/Field        = "MappedMagnet"
s:Ge/V3DBox/Field3D/MagneticField3DTable = "PurgMag3D.TABLE"
# ---> ... done reading
# ---> assumed the order:  x, y, z, Bx, By, Bz
# ---> Min values x,y,z: -5 -5 -26 cm
# ---> Max values x,y,z: 5 17 10 cm
#
#After reordering if necessary
# ---> Min values x,y,z: -5 -5 -26 cm
# ---> Max values x,y,z: 5 17 10 cm
# ---> Dif values x,y,z (range): 10 22 36 cm in z

s:Ge/V3DBox/Field3D/Parent       = "MagGroup"
s:Ge/V3DBox/Field3D/Material      = "Vacuum"
d:Ge/V3DBox/Field3D/HLX           = 5 cm
d:Ge/V3DBox/Field3D/HLY           = 17 cm
d:Ge/V3DBox/Field3D/HLZ           = 26 cm
d:Ge/V3DBox/Field3D/TransX        = 0.0 mm
d:Ge/V3DBox/Field3D/TransY        = 0.0 mm
d:Ge/V3DBox/Field3D/TransZ        = Tf/BackForward/Value mm
#d:Ge/V3DBox/Field3D/TransZ       = Tf/BackStep/Value mm
#d:Ge/V3DBox/Field3D/TransZ       = 4.4 mm
d:Ge/V3DBox/Field3D/RotX          = 0.0 deg
d:Ge/V3DBox/Field3D/RotY          = 0.0 deg
d:Ge/V3DBox/Field3D/RotZ          = 0.0 deg
b:Ge/V3DBox/Field3D/Include       = "TRUE"

d:Ge/Constants/SSD = -50.0 cm

s:Ge/MeasureVolume/Type          = "TsBox"
s:Ge/MeasureVolume/Parent        = "World"
s:Ge/MeasureVolume/Material      = "World"
d:Ge/MeasureVolume/HLX           = 140.0 cm
d:Ge/MeasureVolume/HLY           = 140.0 cm
d:Ge/MeasureVolume/HLZ           = 0.5 cm
d:Ge/MeasureVolume/TransX        = 0.0 cm
d:Ge/MeasureVolume/TransY        = 0.0 cm
#-1*(SSD + HLZ)

```

(continues on next page)

(continued from previous page)

```

d:Ge/MeasureVolume/TransZ = -50.5 cm
d:Ge/MeasureVolume/RotX   = 0.0 deg
d:Ge/MeasureVolume/RotY   = 0.0 deg
d:Ge/MeasureVolume/RotZ   = 0.0 deg

s:Ge/Gap1/Type             = "G4Trd"
s:Ge/Gap1/Parent           = "Nozzle"
#s:Ge/Gap1/Parent          = V3DBox
s:Ge/Gap1/Material         = "Iron"
d:Ge/Gap1/HLX1             = 5.0 cm
d:Ge/Gap1/HLX2             = 9.185 cm
d:Ge/Gap1/HLY1             = 5.0 cm
d:Ge/Gap1/HLY2             = 5.0 cm
d:Ge/Gap1/HLZ              = 5.725 cm
d:Ge/Gap1/TransX           = -9.55 cm
d:Ge/Gap1/TransY           = 0.0 cm
d:Ge/Gap1/TransZ           = -6.89 cm
d:Ge/Gap1/RotX             = 0.0 deg
d:Ge/Gap1/RotY             = 0.0 deg
d:Ge/Gap1/RotZ             = 0.0 deg
s:Ge/Gap1/DrawingStyle     = "FullWireFrame"
b:Ge/Gap1/Include          = "FALSE"

s:Ge/Gap2/Type             = "G4Trd"
s:Ge/Gap2/Parent           = "Nozzle"
#s:Ge/Gap2/Parent          = "V3DBox"
s:Ge/Gap2/Material         = "Iron"
d:Ge/Gap2/HLX1             = 5.0 cm
d:Ge/Gap2/HLX2             = 9.185 cm
d:Ge/Gap2/HLY1             = 5.0 cm
d:Ge/Gap2/HLY2             = 5.0 cm
d:Ge/Gap2/HLZ              = 5.725 cm
d:Ge/Gap2/TransX           = 9.55 cm
d:Ge/Gap2/TransY           = 0.0 cm
d:Ge/Gap2/TransZ           = -6.89 cm
d:Ge/Gap2/RotX             = 0.0 deg
d:Ge/Gap2/RotY             = 0.0 deg
d:Ge/Gap2/RotZ             = 0.0 deg
b:Ge/Gap2/Include          = "FALSE"
s:Ge/Gap2/DrawingStyle     = "FullWireFrame"

#1. Available steppers:
s:Ge/V3DBox/Field3D/Stepper = "ExplicitEuler"
#ExplicitEuler (default)
#ImplicitEuler
#SimpleRunge
#SimpleHeum
#HelixExplicitEuler
#HelixImplicitEuler
#HelixSimpleRunge
#CashKarpRKF45
#RKG3_Stepper

#2. StepMinimum (1.0 mm by default)
d:Ge/V3DBox/Field3D/StepMinimum = 1.0 mm

#3. DeltaChord (0.1 mm by default)

```

(continues on next page)

(continued from previous page)

```

d:Ge/V3DBox/Field3D/DeltaChord = 0.1 mm

Ph/Default/Modules = 1 "g4em-standard_opt0"

s:Tf/BackStep/Function          = "Linear mm"
d:Tf/BackStep/RepetitionInterval = 100.0 ms
d:Tf/BackStep/Rate              = 3 mm/ms
d:Tf/BackStep/StartValue        = 0.0 mm

s:Tf/ForwardStep/Function       = "Linear mm"
d:Tf/ForwardStep/RepetitionInterval = 100.0 ms
d:Tf/ForwardStep/Rate           = -3 mm/ms
d:Tf/ForwardStep/StartValue     = 300.0 mm

s:Tf/BackForward/Function = "step"
dv:Tf/BackForward/Times    = 2 100.0 200.0 ms
dv:Tf/BackForward/Values  = 2 Tf/BackStep/Value Tf/ForwardStep/Value mm

s:Gr/ViewA/Type            = "OpenGL"
i:Gr/ViewA/WindowSizeX     = 1024
i:Gr/ViewA/WindowSizeY     = 768
d:Gr/ViewA/Theta           = 55 deg
d:Gr/ViewA/Phi             = 20 deg
s:Gr/ViewA/Projection       = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom             = 2.
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

i:So/Example/NumberOfHistoriesInRun = 4
d:Tf/TimelineEnd              = 200.0 ms
i:Tf/NumberOfSequentialTimes = 200

```

## 27.10 QuadAndDipoleMagnets.txt

```

# Beam passes through a quadrupole magnet followed by
# a time-varying dipole magnet.

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "TRUE"
s:Ge/World/Material  = "Vacuum"

# Nozzle (a group of components)
s:Ge/Nozzle/Type     = "Group"
s:Ge/Nozzle/Parent   = "World"
d:Ge/Nozzle/TransX   = 0. m
d:Ge/Nozzle/TransY   = 0. m
d:Ge/Nozzle/TransZ   = 0. m
d:Ge/Nozzle/RotX     = 0. deg
d:Ge/Nozzle/RotY     = 0. deg
d:Ge/Nozzle/RotZ     = 0. deg

```

(continues on next page)



(continued from previous page)

```
#####
# Quadrupole pair for focusing beam spot
#####
s:Ge/QBox1/Quad/Type           = "TsBox"
s:Ge/QBox1/Quad/Parent         = "Nozzle"
s:Ge/QBox1/Quad/Material       = "G4_AIR"
d:Ge/QBox1/Quad/HLX            = 15 cm
d:Ge/QBox1/Quad/HLY            = 15 cm
d:Ge/QBox1/Quad/HLZ            = 10 cm
d:Ge/QBox1/Quad/TransX         = 0.0 cm
d:Ge/QBox1/Quad/TransY         = 0.0 cm
d:Ge/QBox1/Quad/TransZ         = 160 cm + Ge/QBox1/Quad/HLZ
d:Ge/QBox1/Quad/RotX           = 0.0 deg
d:Ge/QBox1/Quad/RotY           = 0.0 deg
d:Ge/QBox1/Quad/RotZ           = 0.0 deg
s:Ge/QBox1/Quad/Field = "QuadrupoleMagnet"
d:Ge/QBox1/Quad/MagneticFieldGradientX = .1 tesla/cm
d:Ge/QBox1/Quad/MagneticFieldGradientY = .1 tesla/cm

s:Ge/QBox2/Quad/Type           = "TsBox"
s:Ge/QBox2/Quad/Parent         = "Nozzle"
s:Ge/QBox2/Quad/Material       = "G4_AIR"
d:Ge/QBox2/Quad/HLX            = 15 cm
d:Ge/QBox2/Quad/HLY            = 15 cm
d:Ge/QBox2/Quad/HLZ            = 10 cm
d:Ge/QBox2/Quad/TransX         = 0.0 cm
d:Ge/QBox2/Quad/TransY         = 0.0 cm
d:Ge/QBox2/Quad/TransZ         = 140 cm + Ge/QBox2/Quad/HLZ
d:Ge/QBox2/Quad/RotX           = 0.0 deg
d:Ge/QBox2/Quad/RotY           = 0.0 deg
d:Ge/QBox2/Quad/RotZ           = 0.0 deg
s:Ge/QBox2/Quad/Field = "QuadrupoleMagnet"
d:Ge/QBox2/Quad/MagneticFieldGradientX = .1 tesla/cm
d:Ge/QBox2/Quad/MagneticFieldGradientY = .1 tesla/cm

#####
# Dipole for stiring beam direction!
#####
s:Ge/VBox1/Dipole/Type         = "TsBox"
s:Ge/VBox1/Dipole/Parent       = "Nozzle"
s:Ge/VBox1/Dipole/Material     = "G4_AIR"
d:Ge/VBox1/Dipole/HLX          = 10 cm
d:Ge/VBox1/Dipole/HLY          = 10 cm
d:Ge/VBox1/Dipole/HLZ          = 20 cm
d:Ge/VBox1/Dipole/TransX       = 0.0 cm
d:Ge/VBox1/Dipole/TransY       = 0.0 cm
d:Ge/VBox1/Dipole/TransZ       = 50.0 cm + Ge/VBox1/Dipole/HLZ
d:Ge/VBox1/Dipole/RotX         = 0.0 deg
d:Ge/VBox1/Dipole/RotY         = 0.0 deg
d:Ge/VBox1/Dipole/RotZ         = 0.0 deg
s:Ge/VBox1/Dipole/Field = "DipoleMagnet"
u:Ge/VBox1/Dipole/MagneticFieldDirectionX = 0.0
u:Ge/VBox1/Dipole/MagneticFieldDirectionY = 1.0
u:Ge/VBox1/Dipole/MagneticFieldDirectionZ = 0.0
d:Ge/VBox1/Dipole/MagneticFieldStrength = 0.0 tesla

s:Ge/VBox2/Dipole/Type         = "TsBox"
```

(continues on next page)

(continued from previous page)

```

s:Ge/VBox2/Dipole/Parent           = "Nozzle"
s:Ge/VBox2/Dipole/Material         = "G4_AIR"
d:Ge/VBox2/Dipole/HLX              = 10 cm
d:Ge/VBox2/Dipole/HLY              = 10 cm
d:Ge/VBox2/Dipole/HLZ              = 20 cm
d:Ge/VBox2/Dipole/TransX           = 0.0 cm
d:Ge/VBox2/Dipole/TransY           = 0.0 cm
d:Ge/VBox2/Dipole/TransZ           = 90.0 cm + Ge/VBox2/Dipole/HLZ
d:Ge/VBox2/Dipole/RotX             = 0.0 deg
d:Ge/VBox2/Dipole/RotY             = 0.0 deg
d:Ge/VBox2/Dipole/RotZ             = 0.0 deg
s:Ge/VBox2/Dipole/Field            = "DipoleMagnet"
u:Ge/VBox2/Dipole/MagneticFieldDirectionX = 1.0
u:Ge/VBox2/Dipole/MagneticFieldDirectionY = 0.0
u:Ge/VBox2/Dipole/MagneticFieldDirectionZ = 0.0
d:Ge/VBox2/Dipole/MagneticFieldStrength = Tf/BField1st/Value tesla

#####
# Target window
#####
# Vacuum Window: Target !
s:Ge/VacFilm/Type                  = "TsBox"
s:Ge/VacFilm/Parent                = "World"
s:Ge/VacFilm/Material              = "Lead"
d:Ge/VacFilm/HLX                   = 50.0 cm
d:Ge/VacFilm/HLY                   = 50.0 cm
d:Ge/VacFilm/HLZ                   = 1.0 cm
d:Ge/VacFilm/TransX                = 0. cm
d:Ge/VacFilm/TransY                = 0. cm
d:Ge/VacFilm/TransZ                = -60. cm
d:Ge/VacFilm/RotX                  = 0. deg
d:Ge/VacFilm/RotY                  = 0.0 deg
d:Ge/VacFilm/RotZ                  = 0. deg
s:Ge/VacFilm/Color                  = "brown"
s:Ge/VacFilm/DrawingStyle           = "solid"

#####
# Ps source
#####
d:Ge/BeamPosition/TransZ           = Ge/World/HLZ cm
d:Ge/BeamPosition/RotX             = 180. deg

s:So/Example/Type                   = "Beam"
s:So/Example/Component              = "BeamPosition"
s:So/Example/BeamParticle            = "proton"
d:So/Example/BeamEnergy              = 40.0 MeV
u:So/Example/BeamEnergySpread        = 0.0
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX     = 5.0 cm
d:So/Example/BeamPositionCutoffY     = 5.0 cm
d:So/Example/BeamPositionSpreadX     = 0.65 cm
d:So/Example/BeamPositionSpreadY     = 0.65 cm
s:So/Example/BeamAngularDistribution = "None"

s:Tf/BField1st/Function              = "Step"
dv:Tf/BField1st/Times                = 2 10.0 20.0 ms

```

(continues on next page)

(continued from previous page)

```

dv:Tf/BField1st/Values = 2 Tf/BField1stLeft/Value Tf/BField1stRight/Value tesla

s:Tf/BField1stLeft/Function = "Linear tesla"
d:Tf/BField1stLeft/Rate = 0.08 tesla/ms
d:Tf/BField1stLeft/StartValue = -0.32 tesla
d:Tf/BField1stLeft/RepetitionInterval = 10.0 ms

s:Tf/BField1stRight/Function = "Linear tesla"
d:Tf/BField1stRight/Rate = -0.08 tesla/ms
d:Tf/BField1stRight/StartValue = 0.32 tesla
d:Tf/BField1stRight/RepetitionInterval = 10.0 ms

s:Tf/SmallBox/Function = "Linear tesla"
d:Tf/SmallBox/Rate = 0.1 tesla/ms
d:Tf/SmallBox/StartValue = 0.0 tesla
d:Tf/SmallBox/RepetitionInterval = 70. ms

#Ph/Default/Modules = 1 "g4em-standard_opt0"

i:So/Example/NumberOfHistoriesInRun = 10
d:Tf/TimelineEnd = 40.0 ms
i:Tf/NumberOfSequentialTimes = 120

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 1024
i:Gr/ViewA/WindowSizeY = 768
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
u:Gr/ViewA/TransX = -.2
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 2.0
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
b:Gr/ViewA/CopyOpenGLToEPS = "False"

```

## 27.11 QuadInMovingNozzle.txt

```

# Magnet mounted in a rotating nozzle

s:So/Example/Type = "Beam"
s:So/Example/Component = "ExitWindow"
s:So/Example/BeamParticle = "chargedgeantino"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.0
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 3.0 cm
d:So/Example/BeamPositionCutoffY = 3.0 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

```

(continues on next page)

(continued from previous page)

```

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

b:Ts/PauseBeforeQuit = "True"

s:Tf/NozzleRotation/Function      = "Linear deg"
d:Tf/NozzleRotation/Rate         = 15. deg/ms
d:Tf/NozzleRotation/StartValue   = .0 deg
d:Tf/NozzleRotation/RepetitionInterval = 360. ms
d:Tf/TimelineEnd                 = 20.0 ms
i:Tf/NumberOfSequentialTimes     = 20

d:Ge/World/HLX                   = 2.0 m
d:Ge/World/HLY                   = 2.0 m
d:Ge/World/HLZ                   = 2.0 m
b:Ge/World/Invisible             = "True"

s:Ge/Nozzle/Type                 = "Group"
s:Ge/Nozzle/Parent               = "World"
d:Ge/Nozzle/RotX                 = Tf/NozzleRotation/value deg

s:Ge/ExitWindow/Type             = "TsCylinder"
s:Ge/ExitWindow/Parent           = "Nozzle"
s:Ge/ExitWindow/Material         = "Mylar"
d:Ge/ExitWindow/RMax             = 50.0 mm
d:Ge/ExitWindow/HL              = 0.05 mm
d:Ge/ExitWindow/TransZ          = 40. cm
d:Ge/ExitWindow/RotY            = 180.0 deg
s:Ge/ExitWindow/DrawingStyle     = "Solid"

s:Ge/MagnetBox/Type              = "TsBox"
s:Ge/MagnetBox/Parent            = "Nozzle"
s:Ge/MagnetBox/Material          = "G4_AIR"
d:Ge/MagnetBox/HLX               = 18 cm
d:Ge/MagnetBox/HLY               = 10 cm
d:Ge/MagnetBox/HLZ               = 20 cm

#s:Ge/MagnetBox/Field = "DipoleMagnet"
#u:Ge/MagnetBox/MagneticFieldDirectionX = 1.0
#u:Ge/MagnetBox/MagneticFieldDirectionY = 1.0
#u:Ge/MagnetBox/MagneticFieldDirectionZ = 0.0
#d:Ge/MagnetBox/MagneticFieldStrength = 1.0 tesla

s:Ge/MagnetBox/Field = "QuadrupoleMagnet"
d:Ge/MagnetBox/MagneticFieldGradientX = .1 tesla/cm
d:Ge/MagnetBox/MagneticFieldGradientY = .1 tesla/cm

s:Gr/ViewA/Type                  = "OpenGL"
i:Gr/ViewA/WindowSizeX           = 1024
i:Gr/ViewA/WindowSizeY           = 768
d:Gr/ViewA/Theta                 = 55 deg
d:Gr/ViewA/Phi                   = 20 deg
s:Gr/ViewA/Projection            = "Perspective"
d:Gr/ViewA/PerspectiveAngle      = 30 deg
u:Gr/ViewA/Zoom                  = 1.
b:Gr/ViewA/IncludeStepPoints     = "True"
#i:Gr/ViewA/MagneticFieldArrowDensity = 10

```

## 27.12 QuadrupoleMagnet.txt

```
# MagnetBox magnet

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"

s:Ge/Nozzle/Type     = "Group"
s:Ge/Nozzle/Parent   = "World"

s:Ge/MagnetBox/Type   = "TsBox"
s:Ge/MagnetBox/Parent = "Nozzle"
s:Ge/MagnetBox/Material = "G4_AIR"
d:Ge/MagnetBox/HLX    = 10 cm
d:Ge/MagnetBox/HLY    = 10 cm
d:Ge/MagnetBox/HLZ    = 20 cm
d:Ge/MagnetBox/RotY   = 0 deg
d:Ge/MagnetBox/RotX   = 0 deg

#s:Ge/MagnetBox/Field = "DipoleMagnet"
#u:Ge/MagnetBox/MagneticFieldDirectionX = 1.0
#u:Ge/MagnetBox/MagneticFieldDirectionY = 1.0
#u:Ge/MagnetBox/MagneticFieldDirectionZ = 0.0
#d:Ge/MagnetBox/MagneticFieldStrength   = 1.0 tesla

s:Ge/MagnetBox/Field = "QuadrupoleMagnet"
d:Ge/MagnetBox/MagneticFieldGradientX = .1 tesla/cm
d:Ge/MagnetBox/MagneticFieldGradientY = .1 tesla/cm
d:Ge/MagnetBox/TransX                  = 0 cm

d:Ge/BeamPosition/TransZ = Ge/World/HLZ cm
d:Ge/BeamPosition/RotX   = 180. deg

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "chargedgeantino"
d:So/Example/BeamEnergy     = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.0
s:So/Example/BeamPositionDistribution = "Flat"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 3.0 cm
d:So/Example/BeamPositionCutoffY = 3.0 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 100

s:Gr/ViewA/Type             = "OpenGL"
i:Gr/ViewA/WindowSizeX     = 1024
i:Gr/ViewA/WindowSizeY     = 768
d:Gr/ViewA/Theta            = 55 deg
d:Gr/ViewA/Phi              = 20 deg
s:Gr/ViewA/Projection       = "Perspective"
```

(continues on next page)

(continued from previous page)

```

d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom              = 1.
b:Gr/ViewA/IncludeStepPoints = "True"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

b:Ts/PauseBeforeQuit = "True"

```

## 27.13 RangeModulator\_ConstantBeam.txt

```

# Range modulator wheel in constant current beam

includeFile = RangeModulator.txt

d:Ge/RangeModulatorA/Rotz = Tf/RMW_Rotation/Value deg

i:So/Example/NumberOfHistoriesInRun = 3

d:Tf/TimelineEnd           = 12.0 ms
i:Tf/NumberOfSequentialTimes = 20

s:Tf/RMW_Rotation/Function      = "Linear deg"
d:Tf/RMW_Rotation/Rate          = 3.6 deg/ms
d:Tf/RMW_Rotation/StartValue    = 0.0 deg
d:Tf/RMW_Rotation/RepetitionInterval = 100. ms

b:Ts/PauseBeforeQuit = "False"
i:Ts/ShowHistoryCountAtInterval = 0

# Show time feature steps on console
i:Tf/Verbosity = 1

b:Gr/ViewA/CopyOpenGLToEPS = "False"

```

## 27.14 RangeModulator\_CurrentModulatedBeam.txt

```

# Range modulator wheel in modulated current beam

includeFile = RangeModulator_ConstantBeam.txt

s:Tf/BeamCurrent/Function = "Step"
dv:Tf/BeamCurrent/Times   = 1 10 ms
iv:Tf/BeamCurrent/Values  = 1 10

s:Tf/BeamWeight/Function = "Step"
dv:Tf/BeamWeight/Times   = 10  1 2 3 4 5 6 7 8 9 10 ms
iv:Tf/BeamWeight/Values  = 10  1 4 1 1 2 5 2 3 2 1

i:So/Example/NumberOfHistoriesInRun = Tf/BCM_1/Value

i:Tf/BCM_1/Value = Tf/BeamWeight/Value * Tf/BeamCurrent/Value

```

## 27.15 RangeModulator.txt

```
# Range modulator wheel

b:Ge/World/Invisible = "TRUE"

s:Ge/RangeModulatorA/Type           = "TsRangeModulator"
s:Ge/RangeModulatorA/Material       = "Parent"
s:Ge/RangeModulatorA/Parent         = "World"
d:Ge/RangeModulatorA/TransX         = 10.0 cm
d:Ge/RangeModulatorA/TransY         = 0.0 cm
d:Ge/RangeModulatorA/TransZ         = 0.0 cm
d:Ge/RangeModulatorA/RotX           = 0.0 deg
d:Ge/RangeModulatorA/RotY           = 0.0 deg
d:Ge/RangeModulatorA/RotZ           = 0.0 deg
b:Ge/RangeModulatorA/Invisible      = "TRUE"
b:Ge/RangeModulatorA/PrintInformation = "True"

d:Ge/RangeModulatorA/HeightOfUpper  = 150 mm
d:Ge/RangeModulatorA/HeightOfMiddle = 1.0 mm
d:Ge/RangeModulatorA/HeightOfLower  = 9.0 mm

d:Ge/RangeModulatorA/Shell/Rin       = 15.0 cm
d:Ge/RangeModulatorA/Shell/Rout      = 15.5 cm
s:Ge/RangeModulatorA/Shell/Material  = "Aluminum"
s:Ge/RangeModulatorA/Shell/Color     = "grey"
s:Ge/RangeModulatorA/Shell/DrawingStyle = "Solid"
i:Ge/RangeModulatorA/Shell/VisSegsPerCircle = 360

d:Ge/RangeModulatorA/Hub/Rin         = 6.0 cm
d:Ge/RangeModulatorA/Hub/Rout        = 7.0 cm
s:Ge/RangeModulatorA/Hub/Material    = "Aluminum"
s:Ge/RangeModulatorA/Hub/Color       = "grey"
s:Ge/RangeModulatorA/Hub/DrawingStyle = "Solid"
i:Ge/RangeModulatorA/Hub/VisSegsPerCircle = 360

#Upper tracks
dv:Ge/RangeModulatorA/Upper/RadialDivisions = 1 11.0 cm
s:Ge/RangeModulatorA/Upper/Track1/Pattern   = "LexanBlockT1"
s:Ge/RangeModulatorA/Upper/Track2/Pattern   = "NULL"

#Middle tracks
dv:Ge/RangeModulatorA/Middle/RadialDivisions = 1 11.0 cm
s:Ge/RangeModulatorA/Middle/Track1/Pattern   = "InterfaceDisk"
s:Ge/RangeModulatorA/Middle/Track2/Pattern   = "HoleTrackDisk"

#Lower tracks
dv:Ge/RangeModulatorA/Lower/RadialDivisions = 1 11.0 cm
s:Ge/RangeModulatorA/Lower/Track1/Pattern   = "LeadBlockT1"
s:Ge/RangeModulatorA/Lower/Track2/Pattern   = "NULL"

#A track pattern: 14 blocks of Lexan
d:Ge/LexanBlockT1/Offset = 0.0 deg
dv:Ge/LexanBlockT1/Angles = 14
    5.00 115.00 146.50 173.2 195.07
216.15 230.14 243.00 255.5 270.60
282.20 294.60 306.20 324.00 deg
dv:Ge/LexanBlockT1/Heights = 14
    77.0 82.0 87.0 92.15 95.0
```

(continues on next page)

(continued from previous page)

```

100.4 106.0 110.2 115.3 119.5
124.0 128.8 132.00 60.0 mm
sv:Ge/LexanBlockTl/Materials = 14
"Lexan" "Lexan" "Lexan" "Lexan" "Lexan"
"Lexan" "Lexan" "Lexan" "Lexan" "Lexan"
"Lexan" "Lexan" "Lexan" "Brass"

#A track pattern of single block
dv:Ge/InterfaceDisk/Angles = 1 0.0 deg
dv:Ge/InterfaceDisk/Heights = 1 1.0 mm
sv:Ge/InterfaceDisk/Materials = 1 "Aluminum"

#A track pattern of two blocks but one block is a hole
dv:Ge/HoleTrackDisk/Angles = 2 90.0 110.0 deg
dv:Ge/HoleTrackDisk/Heights = 2 0.0 1.0 mm
sv:Ge/HoleTrackDisk/Materials = 2 "NULL" "Aluminum"

#A track pattern of 8 Lead blocks
dv:Ge/LeadBlockTl/Angles = 9
5.00 115.00 146.50 173.2 195.07
216.15 230.14 243.00 255.5 deg
dv:Ge/LeadBlockTl/Heights = 9
0.890 0.75 0.60 0.52 0.40
0.30 0.16 0.070 0.0 mm
sv:Ge/LeadBlockTl/Materials = 9
"Lead" "Lead" "Lead" "Lead" "Lead"
"Lead" "Lead" "Lead" "NULL"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 1024
i:Gr/ViewA/WindowSizeY = 768
d:Gr/ViewA/Theta = 30 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 1.5
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
b:Gr/ViewA/IncludeAxes = "true"
s:Gr/ViewA/AxesComponent = "RangeModulatorA"

```

(continues on next page)



(continued from previous page)

```

d:Gr/ViewA/AxesSize           = 0.1 m
sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
b:Ts/PauseBeforeQuit = "True"
i:Ts/ShowHistoryCountAtInterval = 0

```

## 27.16 RidgeFilter.txt

```

# Ridge filter

d:Ge/World/HLX           = 0.15 m
d:Ge/World/HLY           = 0.15 m
d:Ge/World/HLZ           = 0.05 m
b:Ge/World/Invisible = "True"

#####
# RidgeFilter Group
#####

s:Ge/RidgeGroup/Parent = "World"
s:Ge/RidgeGroup/Type   = "Group"
d:Ge/RidgeGroup/RotX   = 0.0 deg
d:Ge/RidgeGroup/RotY   = 0.0 deg
d:Ge/RidgeGroup/RotZ   = 0.0 deg
d:Ge/RidgeGroup/TransX = 0.0 cm
d:Ge/RidgeGroup/TransY = 0.0 cm
d:Ge/RidgeGroup/TransZ = 0.0 cm

#####
# Ridge Filter
#####
s:Ge/RidgeFilterA/Type   = "TsRidgeFilter"
s:Ge/RidgeFilterA/Parent = "RidgeGroup"
s:Ge/RidgeFilterA/Material = "Aluminum"
d:Ge/RidgeFilterA/TransX = 0.0 cm
d:Ge/RidgeFilterA/TransY = 0.0 cm
d:Ge/RidgeFilterA/TransZ = 0.0 cm
d:Ge/RidgeFilterA/RotX   = 0.0 deg
d:Ge/RidgeFilterA/RotY   = 0.0 deg
d:Ge/RidgeFilterA/RotZ   = 0.0 deg
s:Ge/RidgeFilterA/DrawingStyle = "Solid"
b:Ge/RidgeFilterA/PrintInformation = "True"

# Ridge Geometry: TOPAS will connect points of x and z to build a ridge.
# Then copy a ridge and place it using Displacement parameter
dv:Ge/RidgeFilterA/XPoints = 8
0.0 0.8 1.3 1.8 2.2 2.7 3.2 4.0 mm
dv:Ge/RidgeFilterA/ZPoints = 8
2.4 4.0 9.1 14.0 14.0 9.1 4.0 2.4 mm
d:Ge/RidgeFilterA/Width = 4.0 mm
d:Ge/RidgeFilterA/Length = 1.0 cm
dv:Ge/RidgeFilterA/Displacement = 3 -5.0 0.0 5.0 mm

```

(continues on next page)

(continued from previous page)

```
#####
# Physics, Beam, and Graphic setting
#####
s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 230.0 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 1.0 cm
d:So/Example/BeamPositionSpreadY = 1.0 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
b:Ts/PauseBeforeQuit = "True"

s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 600
i:Gr/ViewA/WindowSizeY = 600
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Orthogonal"
d:Gr/ViewA/PerspectiveAngle = 20 deg
u:Gr/ViewA/Zoom = 1.6
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"

b:Gr/ViewA/IncludeAxes = "true"
s:Gr/ViewA/AxesComponent = "RidgeGroup"
d:Gr/ViewA/AxesSize = 0.1 m
```

## 27.17 RotatingMagnet.txt

```
# A dipole magnet has field strength changing over time
# such that magnet is sweeping back and forth.
# At the same time, the nozzle that contains this magnet is rotating.

d:Ge/World/HLX = 2.0 m
d:Ge/World/HLY = 2.0 m
d:Ge/World/HLZ = 2.0 m
b:Ge/World/Invisible = "TRUE"

# Nozzle (a group of components)
s:Ge/Nozzle/Type = "Group"
s:Ge/Nozzle/Parent = "World"
d:Ge/Nozzle/TransX = 0. m
d:Ge/Nozzle/TransY = 0. m
```

(continues on next page)

(continued from previous page)

```

d:Ge/Nozzle/TransZ = 0. m
d:Ge/Nozzle/RotX   = 0. deg
d:Ge/Nozzle/RotY   = 0. deg
d:Ge/Nozzle/RotZ   = Tf/MagnetRotation/Value deg

s:Tf/MagnetRotation/Function = "Linear deg"
d:Tf/MagnetRotation/Rate    = 2. deg/ms
d:Tf/MagnetRotation/StartValue = -0. deg
d:Tf/MagnetRotation/RepetitionInterval = 80.0 ms

#####
# Dipole for stiring beam direction!
#####

s:Ge/VBox2/Dipole/Type      = "TsBox"
s:Ge/VBox2/Dipole/Parent    = "Nozzle"
s:Ge/VBox2/Dipole/Material  = "G4_AIR"
d:Ge/VBox2/Dipole/HLX      = 10 cm
d:Ge/VBox2/Dipole/HLY      = 10 cm
d:Ge/VBox2/Dipole/HLZ      = 20 cm
d:Ge/VBox2/Dipole/TransX    = 0.0 cm
d:Ge/VBox2/Dipole/TransY    = 0.0 cm
d:Ge/VBox2/Dipole/TransZ    = 90.0 cm + Ge/VBox2/Dipole/HLZ
d:Ge/VBox2/Dipole/RotX      = 0.0 deg
d:Ge/VBox2/Dipole/RotY      = 0.0 deg
d:Ge/VBox2/Dipole/RotZ      = 0.0 deg
s:Ge/VBox2/Dipole/Field     = "DipoleMagnet"
u:Ge/VBox2/Dipole/MagneticFieldDirectionX = 1.0
u:Ge/VBox2/Dipole/MagneticFieldDirectionY = 0.0
u:Ge/VBox2/Dipole/MagneticFieldDirectionZ = 0.0
d:Ge/VBox2/Dipole/MagneticFieldStrength   = Tf/BField1st/Value tesla

#####
# Vacuum window for phase space!
#####
# Vacuum Window: Target !
s:Ge/VacFilm/Type      = "TsCylinder"
s:Ge/VacFilm/Parent    = "World"
s:Ge/VacFilm/Material  = "G4_WATER"
d:Ge/VacFilm/RMin      = 0.0 cm
d:Ge/VacFilm/RMax      = 100.0 cm
d:Ge/VacFilm/HL        = 1.0 cm
d:Ge/VacFilm/SPhi      = 0. deg
d:Ge/VacFilm/DPhi      = 360. deg
d:Ge/VacFilm/TransX     = 0. cm
d:Ge/VacFilm/TransY     = 0. cm
d:Ge/VacFilm/TransZ     = 0. cm
d:Ge/VacFilm/RotX       = 0. deg
d:Ge/VacFilm/RotY       = 0.0 deg
d:Ge/VacFilm/RotZ       = 0. deg
s:Ge/VacFilm/Color      = "skyblue"

#s:Sc/Plane1/Quantity    = "PhaseSpace"
#s:Sc/Plane1/Surface     = "VacFilm/ZMinusSurface"
#s:Sc/Plane1/OutputType  = "ASCII"
#i:Sc/Plane1/OutputBufferSize = 1000

```

(continues on next page)

(continued from previous page)

```
#####
# Ps source
#####
d:Ge/BeamPosition/TransZ = Ge/World/HLZ cm
d:Ge/BeamPosition/RotX   = 180. deg

s:Ge/World/Material = "Vacuum"

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "e+"
d:So/Example/BeamEnergy     = 40.0 MeV
u:So/Example/BeamEnergySpread = 5.
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX   = 5.0 cm
d:So/Example/BeamPositionCutoffY   = 5.0 cm
d:So/Example/BeamPositionSpreadX   = 0.65 cm
d:So/Example/BeamPositionSpreadY   = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX    = 90. deg
d:So/Example/BeamAngularCutoffY    = 90. deg
d:So/Example/BeamAngularSpreadX    = 0.005 rad
d:So/Example/BeamAngularSpreadY    = 0.005 rad
i:So/Example/NumberOfHistoriesInRun = 500

s:Tf/BField1st/Function = "Step"
dv:Tf/BField1st/Times   = 2 10.0 20.0 ms
dv:Tf/BField1st/Values  = 2 Tf/BField1stLeft/Value Tf/BField1stRight/Value tesla

s:Tf/BField1stLeft/Function      = "Linear tesla"
d:Tf/BField1stLeft/Rate          = 0.064 tesla/ms
d:Tf/BField1stLeft/StartValue    = -0.32 tesla
d:Tf/BField1stLeft/RepetitionInterval = 10.0 ms

s:Tf/BField1stRight/Function     = "Linear tesla"
d:Tf/BField1stRight/Rate         = -0.064 tesla/ms
d:Tf/BField1stRight/StartValue   = 0.32 tesla
d:Tf/BField1stRight/RepetitionInterval = 10.0 ms

s:Tf/SmallBox/Function          = "Linear tesla"
d:Tf/SmallBox/Rate              = 0.1 tesla/ms
d:Tf/SmallBox/StartValue        = 0.0 tesla
d:Tf/SmallBox/RepetitionInterval = 70. ms

d:Tf/TimelineEnd                = 40.0 ms
i:Tf/NumberOfSequentialTimes    = 10

s:Gr/ViewA/Type                 = "OpenGL"
i:Gr/ViewA/WindowSizeX          = 1024
i:Gr/ViewA/WindowSizeY          = 768
d:Gr/ViewA/Theta                 = 55 deg
d:Gr/ViewA/Phi                   = 20 deg
s:Gr/ViewA/Projection            = "Perspective"
d:Gr/ViewA/PerspectiveAngle      = 30 deg
u:Gr/ViewA/Zoom                  = 1.4
u:Gr/ViewA/TransX                = -.2
```

(continues on next page)

(continued from previous page)

```
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
```

## 27.18 UniformElectroMagneticField.txt

```
# Uniform electric field

d:Ge/World/HLX      = 2.0 m
d:Ge/World/HLY      = 2.0 m
d:Ge/World/HLZ      = 2.0 m
b:Ge/World/Invisible = "True"
s:Ge/World/Material  = "Vacuum"

s:Ge/DriftBox/Type   = "TsBox"
s:Ge/DriftBox/Parent = "World"
s:Ge/DriftBox/Material = "Vacuum"
d:Ge/DriftBox/HLX    = 80 cm
d:Ge/DriftBox/HLY    = 80 cm
d:Ge/DriftBox/HLZ    = 100 cm
s:Ge/DriftBox/Field  = "UniformElectroMagnetic"
u:Ge/DriftBox/ElectricFieldDirectionX = 1.0
u:Ge/DriftBox/ElectricFieldDirectionY = 1.0
u:Ge/DriftBox/ElectricFieldDirectionZ = 0.0
d:Ge/DriftBox/ElectricFieldStrength   = 5000 kV/cm
u:Ge/DriftBox/MagneticFieldDirectionX = 0.0
u:Ge/DriftBox/MagneticFieldDirectionY = 1.0
u:Ge/DriftBox/MagneticFieldDirectionZ = 0.0
d:Ge/DriftBox/MagneticFieldStrength   = 5.0 tesla
d:Ge/DriftBox/RotZ                    = Tf/RotationStep/Value deg

d:Ge/BeamPosition/TransZ = Ge/World/HLZ cm
d:Ge/BeamPosition/RotX   = 180. deg

i:Tf/Verbosity           = 1
d:Tf/TimelineEnd         = 30.0 ms
i:Tf/NumberOfSequentialTimes = 40

s:Tf/RotationStep/Function      = "Linear deg"
d:Tf/RotationStep/Rate         = 5. deg/ms
d:Tf/RotationStep/StartValue    = -28.0 deg
d:Tf/RotationStep/RepetitionInterval = 360. ms

s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "chargedgeantino"
d:So/Example/BeamEnergy        = 169.23 MeV
u:So/Example/BeamEnergySpread   = 0.0
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 5.0 cm
d:So/Example/BeamPositionCutoffY = 5.0 cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "None"
i:So/Example/NumberOfHistoriesInRun = 10
```

(continues on next page)

(continued from previous page)

```
s:Gr/ViewA/Type           = "OpenGL"
i:Gr/ViewA/WindowSizeX    = 1024
i:Gr/ViewA/WindowSizeY    = 768
d:Gr/ViewA/Theta          = 55 deg
d:Gr/ViewA/Phi            = 20 deg
s:Gr/ViewA/Projection     = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom           = 1.
b:Gr/ViewA/IncludeStepPoints = "True"

b:Ts/PauseBeforeQuit = "True"
```

## 28.1 BoxWithinBox.txt

```
# A box undergoes rotation relative to a parent box
# which is itself undergoing a different rotation.

d:Ge/World/HLX      = 10. m
d:Ge/World/HLY      = 10. m
d:Ge/World/HLZ      = 10. m
s:Ge/World/Material  = "Vacuum"
b:Ge/World/Invisible = "True"

d:Ge/BeamPosition/TransX = 3.9 m
d:Ge/BeamPosition/TransY = 3.9 m

s:Ge/OuterBox/Parent   = "World"
s:Ge/OuterBox/Type     = "TsBox"
s:Ge/OuterBox/Material = "Air"
d:Ge/OuterBox/HLX      = 4. m
d:Ge/OuterBox/HLY      = 4. m
d:Ge/OuterBox/HLZ      = 4. m
d:Ge/OuterBox/TransX   = 2. m
d:Ge/OuterBox/TransY   = 2. m
d:Ge/OuterBox/TransZ   = 0. m
d:Ge/OuterBox/RotX     = 0. deg
d:Ge/OuterBox/RotY     = 0. deg
d:Ge/OuterBox/RotZ     = Tf/OuterBoxStep/Value deg

s:Ge/InnerBox/Parent   = "OuterBox"
s:Ge/InnerBox/Type     = "TsBox"
s:Ge/InnerBox/Material = "Lead"
d:Ge/InnerBox/HLX      = 1. m
d:Ge/InnerBox/HLY      = 1. m
d:Ge/InnerBox/HLZ      = 2. m
```

(continues on next page)

(continued from previous page)

```

d:Ge/InnerBox/TransX      = 1. m
d:Ge/InnerBox/TransY      = 1. m
d:Ge/InnerBox/TransZ      = 0. m
d:Ge/InnerBox/RotX        = 0. deg
d:Ge/InnerBox/RotY        = Tf/InnerBoxStep/Value deg
d:Ge/InnerBox/RotZ        = Tf/InnerBoxStep/Value deg
s:Ge/InnerBox/DrawingStyle = "Solid"
s:Ge/InnerBox/Color        = "red"

s:Tf/OuterBoxRot/Function  = "Linear deg"
d:Tf/OuterBoxRot/Rate      = 2. deg/ms
d:Tf/OuterBoxRot/StartValue = -28.0 deg
d:Tf/OuterBoxRot/RepetitionInterval = 360. ms

s:Tf/InnerBoxRot/Function  = "Linear deg"
d:Tf/InnerBoxRot/Rate      = -2. deg/ms
d:Tf/InnerBoxRot/StartValue = 0.0 deg
d:Tf/InnerBoxRot/RepetitionInterval = 360. ms

s:Tf/InnerBoxStep/Function = "Step"
dv:Tf/InnerBoxStep/Times   = 2 6 60 ms
dv:Tf/InnerBoxStep/Values  = 2 0. Tf/InnerBoxRot/Value deg
d:Tf/InnerBoxStep/RepetitionInterval = 360. ms

s:Tf/OuterBoxStep/Function = "Step"
dv:Tf/OuterBoxStep/Times   = 2 14 60 ms
dv:Tf/OuterBoxStep/Values  = 2 0. Tf/OuterBoxRot/Value deg
d:Tf/OuterBoxStep/RepetitionInterval = 360. ms

i:Tf/Verbosity             = 1
d:Tf/TimelineEnd           = 30.0 ms
i:Tf/NumberOfSequentialTimes = 40

s:Gr/ViewA/Type            = "OpenGL"
i:Gr/ViewA/WindowSizeX     = 1024
i:Gr/ViewA/WindowSizeY     = 768
d:Gr/ViewA/Theta           = 55 deg
d:Gr/ViewA/Phi             = 20 deg
u:Gr/ViewA/Zoom            = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
s:Gr/ViewA/Projection       = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
b:Gr/ViewA/CopyOpenGLToEPS  = "False"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy     = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm

```

(continues on next page)



(continued from previous page)

```

s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX     = 90. deg
d:So/Example/BeamAngularCutoffY     = 90. deg
d:So/Example/BeamAngularSpreadX     = 0.0032 rad
d:So/Example/BeamAngularSpreadY     = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

```

## 28.2 CameraRotateAndZoom.txt

```

# Graphics views zoom and rotate under control of time features

s:Ge/Box/Type           = "TsBox"
s:Ge/Box/Parent         = "World"
s:Ge/Box/Material       = "G4_WATER"
d:Ge/Box/HLX            = 1. m
d:Ge/Box/HLY            = 1. m
d:Ge/Box/HLZ            = .4 m
d:Ge/Box/TransX         = 0. m
d:Ge/Box/TransY         = 0. m
d:Ge/Box/TransZ         = 0. m
d:Ge/Box/RotX           = 0. deg
d:Ge/Box/RotY           = 0. deg
d:Ge/Box/RotZ           = 0. deg
s:Ge/Box/DrawingStyle   = "Solid"

s:Gr/ViewA/Type         = "OpenGL"
i:Gr/ViewA/WindowSizeX  = 400
i:Gr/ViewA/WindowSizeY  = 400
i:Gr/ViewA/WindowPosX   = 0
i:Gr/ViewA/WindowPosY   = 0
s:Gr/ViewA/ColorBy      = "particletype"
sv:Gr/ViewA/ColorByParticleTypeNames = 2 "proton" "e-"
sv:Gr/ViewA/ColorByParticleTypeColors = 2 "red" "green"
d:Gr/ViewA/Theta        = Tf/Rotate/Value deg
d:Gr/ViewA/Phi          = Tf/Rotate/Value deg

s:Gr/ViewB/Type         = "OpenGL"
i:Gr/ViewB/WindowSizeX  = 400
i:Gr/ViewB/WindowSizeY  = 400
i:Gr/ViewB/WindowPosX   = 0
i:Gr/ViewB/WindowPosY   = 400
u:Gr/ViewB/Zoom          = Tf/Zoom/Value
b:Gr/ViewB/HiddenLineRemovalForTrajectories = "t"
s:Gr/ViewB/ColorBy      = "origincomponent"
sv:Gr/ViewB/ColorByOriginComponentNames = 2 "World" "Box"
sv:Gr/ViewB/ColorByOriginComponentColors = 2 "yellow" "red"

s:Tf/Rotate/Function     = "Linear deg"
d:Tf/Rotate/Rate         = 2. deg/ms
d:Tf/Rotate/StartValue   = 0.0 deg
d:Tf/Rotate/RepetitionInterval = 360. ms

s:Tf/Zoom/Function       = "Linear"
d:Tf/Zoom/Rate           = 0.1 1/ms

```

(continues on next page)

(continued from previous page)

```

u:Tf/Zoom/StartValue          = 1
d:Tf/Zoom/RepetitionInterval = 360. ms

d:Tf/TimelineEnd              = 90.0 ms
i:Tf/NumberOfSequentialTimes = 90

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "proton"
d:So/Example/BeamEnergy        = 169.23 MeV
u:So/Example/BeamEnergySpread  = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX  = 10. cm
d:So/Example/BeamPositionCutoffY  = 10. cm
d:So/Example/BeamPositionSpreadX  = 0.65 cm
d:So/Example/BeamPositionSpreadY  = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX   = 90. deg
d:So/Example/BeamAngularCutoffY   = 90. deg
d:So/Example/BeamAngularSpreadX   = 0.0032 rad
d:So/Example/BeamAngularSpreadY   = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1

b:Ts/PauseBeforeQuit = "False"

```

## 28.3 ChangingKEFilterByTimeFeature.txt

```

# Kinetic Energy filter cutoff varies by time feature

b:Ts/PauseBeforeQuit          = "True"
b:Ge/CheckForUnusedComponents = "False"

s:Ge/Box/Type                 = "TsBox"
s:Ge/Box/Parent               = "World"
s:Ge/Box/Material              = "G4_WATER"
d:Ge/Box/HLX                  = 1. m
d:Ge/Box/HLY                  = 1. m
d:Ge/Box/HLZ                  = .4 m
d:Ge/Box/TransX               = 0. m
d:Ge/Box/TransY               = 0. m
d:Ge/Box/TransZ               = 0. m
d:Ge/Box/RotX                 = 0. deg
d:Ge/Box/RotY                 = 0. deg
d:Ge/Box/RotZ                 = 0. deg

s:Sc/Box/Quantity              = "Fluence"
s:Sc/Box/Component             = "Box"
s:Sc/Box/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/Box/OutputToConsole       = "True"

s:Sc/Boxb/Quantity             = "Fluence"

```

(continues on next page)

(continued from previous page)

```

s:Sc/Boxb/Component           = "Box"
s:Sc/Boxb/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/Boxb/OutputToConsole      = "True"
d:Sc/Boxb/OnlyIncludeIfIncidentParticleKEAbove = 160. MeV

s:Sc/Boxc/Quantity            = "Fluence"
s:Sc/Boxc/Component           = "Box"
s:Sc/Boxc/IfOutputFileAlreadyExists = "Overwrite"
b:Sc/Boxc/OutputToConsole      = "True"
d:Sc/Boxc/OnlyIncludeIfIncidentParticleKEAbove = Tf/Zoom/Value MeV

s:Gr/MyOpenGL/Type            = "OpenGL"
i:Gr/MyOpenGL/WindowSizeX     = 600
i:Gr/MyOpenGL/WindowSizeY     = 600
i:Gr/MyOpenGL/WindowPosX      = 0
i:Gr/MyOpenGL/WindowPosY      = 0

s:Tf/Zoom/Function             = "Linear MeV"
d:Tf/Zoom/Rate                 = 1. MeV/ms
d:Tf/Zoom/StartValue           = 160.0 MeV
d:Tf/Zoom/RepetitionInterval   = 360. ms

d:Tf/TimelineEnd               = 720.0 ms
i:Tf/NumberOfSequentialTimes   = 720

s:So/Example/Type              = "Beam"
s:So/Example/Component         = "BeamPosition"
s:So/Example/BeamParticle      = "proton"
d:So/Example/BeamEnergy        = 169.23 MeV
u:So/Example/BeamEnergySpread   = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1

```

## 28.4 ColorChange.txt

```

# Demonstrates step time feature by changing a box's color over time

d:Ge/World/HLX                = 10. m
d:Ge/World/HLY                = 10. m
d:Ge/World/HLZ                = 10. m
b:Ge/World/Invisible          = "TRUE"

s:Ge/BigBox/Parent             = "World"
s:Ge/BigBox/Type               = "TsBox"

```

(continues on next page)

(continued from previous page)

```

s:Ge/BigBox/Material = "Air"
d:Ge/BigBox/HLX      = 2. m
d:Ge/BigBox/HLY      = 3. m
d:Ge/BigBox/HLZ      = 4. m
d:Ge/BigBox/TransX    = 0. m
d:Ge/BigBox/TransY    = 0. m
d:Ge/BigBox/TransZ    = 0. m
d:Ge/BigBox/RotX      = 0. deg
d:Ge/BigBox/RotY      = 0. deg
d:Ge/BigBox/RotZ      = 0. deg
s:Ge/BigBox/Color     = Tf/Color/Value

s:Tf/Color/Function = "Step"
sv:Tf/Color/Values   = 4 "white" "blue" "green" "red"
dv:Tf/Color/Times    = 4 1. 2. 3. 4. ms

i:Tf/Verbosity        = 1
d:Tf/TimelineEnd      = 60.0 ms
i:Tf/NumberOfSequentialTimes = 60

s:Gr/ViewA/Type       = "OpenGL"
i:Gr/ViewA/WindowSizeX = 1024
i:Gr/ViewA/WindowSizeY = 768
d:Gr/ViewA/Theta      = 55 deg
d:Gr/ViewA/Phi        = 20 deg
u:Gr/ViewA/Zoom        = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
s:Gr/ViewA/Projection  = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 28.5 CylinderGrowingInPhi.txt

```
# Demonstrate time features by changing a cylinder's phi extent
# and color over time.
```

```
d:Ge/World/HLX      = 10. m
d:Ge/World/HLY      = 10. m
d:Ge/World/HLZ      = 10. m
b:Ge/World/Invisible = "True"

s:Ge/Outer/Type      = "TsCylinder"
s:Ge/Outer/Parent    = "World"
s:Ge/Outer/Material  = "Vacuum"
d:Ge/Outer/RMin      = 0.0 cm
d:Ge/Outer/RMax      = 60.0 cm
d:Ge/Outer/SPhi      = 0.0 deg
d:Ge/Outer/DPhi      = 360. deg
d:Ge/Outer/HL        = 3. cm
d:Ge/Outer/TransX    = 0. cm
d:Ge/Outer/TransY    = 0. cm
d:Ge/Outer/TransZ    = -53. cm
d:Ge/Outer/RotX      = 0. deg
d:Ge/Outer/RotY      = 0. deg
d:Ge/Outer/RotZ      = 0. deg

s:Ge/Outer2/Type     = "TsCylinder"
s:Ge/Outer2/Parent   = "World"
s:Ge/Outer2/Material = "Vacuum"
d:Ge/Outer2/RMin     = 0.0 cm
d:Ge/Outer2/RMax     = 60.0 cm
d:Ge/Outer2/SPhi     = 0.0 deg
d:Ge/Outer2/DPhi     = 360. deg
d:Ge/Outer2/HL       = 3. cm
d:Ge/Outer2/TransX   = 0. cm
d:Ge/Outer2/TransY   = 0. cm
d:Ge/Outer2/TransZ   = 53. cm
d:Ge/Outer2/RotX     = 0. deg
d:Ge/Outer2/RotY     = 0. deg
d:Ge/Outer2/RotZ     = 0. deg

s:Ge/Cylinder/Type   = "TsCylinder"
s:Ge/Cylinder/Parent = "World"
s:Ge/Cylinder/Material = "Vacuum"
d:Ge/Cylinder/RMin   = 50.0 cm
d:Ge/Cylinder/RMax   = 55.0 cm
d:Ge/Cylinder/SPhi   = 20.0 deg
d:Ge/Cylinder/DPhi   = Tf/Phi/Value deg
d:Ge/Cylinder/HL     = 50. cm
d:Ge/Cylinder/TransX = 0. cm
d:Ge/Cylinder/TransY = 0. cm
d:Ge/Cylinder/TransZ = 0. cm
d:Ge/Cylinder/RotX   = 0. deg
d:Ge/Cylinder/RotY   = 0. deg
d:Ge/Cylinder/RotZ   = 0. deg
s:Ge/Cylinder/Color   = Tf/Color/Value
i:Ge/Cylinder/VisSegsPerCircle = 360
s:Ge/Cylinder/DrawingStyle = "FullWireFrame"
```

(continues on next page)

(continued from previous page)

```

s:Tf/Color/Function = "Step"
sv:Tf/Color/Values  = 4 "white" "Blue" "grEEn" "red"
dv:Tf/Color/Times   = 4 1. 2. 3. 4. ms

s:Tf/PosPhi/Function      = "Linear deg"
d:Tf/PosPhi/Rate          = 12. deg/ms
d:Tf/PosPhi/StartValue    = 10.0 deg
d:Tf/PosPhi/RepetitionInterval = 29. ms

s:Tf/NegPhi/Function      = "Linear deg"
d:Tf/NegPhi/Rate          = -12. deg/ms
d:Tf/NegPhi/StartValue    = 358.0 deg
d:Tf/NegPhi/RepetitionInterval = 29. ms

s:Tf/Phi/Function        = "Step"
dv:Tf/Phi/Times          = 2 29 58 ms
dv:Tf/Phi/Values         = 2 Tf/PosPhi/Value Tf/NegPhi/value deg
d:Tf/Phi/RepetitionInterval = 100. ms

i:Tf/Verbosity           = 1
d:Tf/TimelineEnd         = 58.0 ms
i:Tf/NumberOfSequentialTimes = 58

s:Gr/ViewA/Type          = "OpenGL"
i:Gr/ViewA/WindowSizeX   = 1024
i:Gr/ViewA/WindowSizeY   = 768
d:Gr/ViewA/Theta         = 55 deg
d:Gr/ViewA/Phi           = 20 deg
u:Gr/ViewA/Zoom           = 2.
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
s:Gr/ViewA/Projection     = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
b:Gr/ViewA/CopyOpenGLToEPS = "False"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type        = "Beam"
s:So/Example/Component   = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy   = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

```

## 28.6 Darkening.txt

```
# Boxes are set up such that they darken as they receive radiation.
# They therefore represent pieces of radiosensitive film.
# One of these films starts outside of the beam
# and moves into the beam over the course of the session.
```

```
s:Ge/World/Material = "Vacuum"
d:Ge/World/HLX      = 10. m
d:Ge/World/HLY      = 10. m
d:Ge/World/HLZ      = 10. m

d:Ge/Phantom/TransY = Tf/BoxMove/Value cm

b:Ts/PauseBeforeQuit = "False"

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "proton"
d:So/Example/BeamEnergy     = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.1 cm
d:So/Example/BeamPositionSpreadY = 0.1 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.3 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 4

s:Tf/BoxMoveDown/Function = "Linear m"
d:Tf/BoxMoveDown/Rate     = -5. cm/ms
d:Tf/BoxMoveDown/StartValue = 1. m
d:Tf/BoxMoveDown/RepetitionInterval = 250. ms

s:Tf/BoxMove/Function = "Step"
dv:Tf/BoxMove/Times = 2 20. 250 ms
dv:Tf/BoxMove/Values = 2 Tf/BoxMoveDown/Value 0. cm

d:Tf/TimelineEnd = 75.0 ms
i:Tf/NumberOfSequentialTimes = 200

# Graphics
s:Gr/ViewA/Type = "OpenGL"
i:Gr/ViewA/WindowSizeX = 900
i:Gr/ViewA/WindowSizeY = 900
d:Gr/ViewA/Theta = 55 deg
d:Gr/ViewA/Phi = 20 deg
s:Gr/ViewA/Projection = "Perspective"
d:Gr/ViewA/PerspectiveAngle = 30 deg
u:Gr/ViewA/Zoom = 15.
u:Gr/ViewA/TransX = .0
u:Gr/ViewA/TransY = .2
```

(continues on next page)

(continued from previous page)

```

b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
#b:Gr/ViewA/CopyOpenGLToEPS = "True"

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:Ge/Phantom/Type = "TsBox"
s:Ge/Phantom/Parent = "World"
s:Ge/Phantom/Material = "G4_WATER"
d:Ge/Phantom/HLX = 30.0 cm
d:Ge/Phantom/HLY = 30.0 cm
d:Ge/Phantom/HLZ = 9. cm
d:Ge/Phantom/TransX = 0. cm
d:Ge/Phantom/TransZ = 20. cm
d:Ge/Phantom/RotX = 0. deg
d:Ge/Phantom/RotY = 0. deg
d:Ge/Phantom/RotZ = 0. deg
s:Ge/Phantom/Color = "white"
s:Ge/Phantom/DrawingStyle = "solid"

s:Ge/Phantom1/Type = "TsBox"
s:Ge/Phantom1/Parent = "World"
s:Ge/Phantom1/Material = "G4_WATER"
d:Ge/Phantom1/HLX = 30.0 cm
d:Ge/Phantom1/HLY = 30.0 cm
d:Ge/Phantom1/HLZ = 10.0 cm
d:Ge/Phantom1/TransX = 0. cm
d:Ge/Phantom1/TransY = 0. cm
d:Ge/Phantom1/TransZ = -30. cm
d:Ge/Phantom1/RotX = 0. deg
d:Ge/Phantom1/RotY = 0. deg
d:Ge/Phantom1/RotZ = 0. deg
s:Ge/Phantom1/Color = "white"
s:Ge/Phantom1/DrawingStyle = "solid"

s:Ge/Phantom2/Type = "TsBox"
s:Ge/Phantom2/Parent = "World"
s:Ge/Phantom2/Material = "G4_WATER"
d:Ge/Phantom2/HLX = 30.0 cm
d:Ge/Phantom2/HLY = 30.0 cm
d:Ge/Phantom2/HLZ = 10.0 cm
d:Ge/Phantom2/TransX = -70. cm
d:Ge/Phantom2/TransY = 0. cm
d:Ge/Phantom2/TransZ = -30. cm
d:Ge/Phantom2/RotX = 0. deg
d:Ge/Phantom2/RotY = 0. deg
d:Ge/Phantom2/RotZ = 0. deg
s:Ge/Phantom2/Color = "white"
s:Ge/Phantom2/DrawingStyle = "solid"

s:Ge/Phantom3/Type = "TsBox"
s:Ge/Phantom3/Parent = "World"
s:Ge/Phantom3/Material = "G4_WATER"
d:Ge/Phantom3/HLX = 30.0 cm
d:Ge/Phantom3/HLY = 30.0 cm
d:Ge/Phantom3/HLZ = 10.0 cm
d:Ge/Phantom3/TransX = 70. cm
d:Ge/Phantom3/TransY = 0. cm

```

(continues on next page)



(continued from previous page)

```

d:Ge/Phantom3/TransZ      = -30. cm
d:Ge/Phantom3/RotX        = 0. deg
d:Ge/Phantom3/RotY        = 0. deg
d:Ge/Phantom3/RotZ        = 0. deg
s:Ge/Phantom3/Color       = "white"
s:Ge/Phantom3/DrawingStyle = "solid"

s:Ge/Phantom4/Type        = "TsBox"
s:Ge/Phantom4/Parent      = "World"
s:Ge/Phantom4/Material    = "G4_WATER"
d:Ge/Phantom4/HLX         = 30.0 cm
d:Ge/Phantom4/HLY         = 30.0 cm
d:Ge/Phantom4/HLZ         = 10.0 cm
d:Ge/Phantom4/TransX      = -140. cm
d:Ge/Phantom4/TransY      = 0. cm
d:Ge/Phantom4/TransZ      = -30. cm
d:Ge/Phantom4/RotX        = 0. deg
d:Ge/Phantom4/RotY        = 0. deg
d:Ge/Phantom4/RotZ        = 0. deg
s:Ge/Phantom4/Color       = "white"
s:Ge/Phantom4/DrawingStyle = "solid"

s:Ge/Phantom5/Type        = "TsBox"
s:Ge/Phantom5/Parent      = "World"
s:Ge/Phantom5/Material    = "G4_WATER"
d:Ge/Phantom5/HLX         = 30.0 cm
d:Ge/Phantom5/HLY         = 30.0 cm
d:Ge/Phantom5/HLZ         = 10.0 cm
d:Ge/Phantom5/TransX      = 140. cm
d:Ge/Phantom5/TransY      = 0. cm
d:Ge/Phantom5/TransZ      = -30. cm
d:Ge/Phantom5/RotX        = 0. deg
d:Ge/Phantom5/RotY        = 0. deg
d:Ge/Phantom5/RotZ        = 0. deg
s:Ge/Phantom5/Color       = "white"
s:Ge/Phantom5/DrawingStyle = "solid"

s:Sc/DoseAtPhantom/Quantity = "EnergyDeposit"
s:Sc/DoseAtPhantom/Component = "Phantom"
s:Sc/DoseAtPhantom/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom/ColorBy   = "Sum"
sv:Sc/DoseAtPhantom/ColorNames = 14
"white"
"grey240"
"grey220"
"grey200"
"grey180"
"grey160"
"grey140"
"grey120"
"grey100"
"grey080"
"grey060"
"grey040"
"grey020"
"black"

```

(continues on next page)

(continued from previous page)

```
dv:Sc/DoseAtPhantom/ColorValues = 13
0.
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
MeV

b:Sc/DoseAtPhantom/OutputAfterRun = "True"

s:Sc/DoseAtPhantom1/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantom1/Component     = "Phantom1"
s:Sc/DoseAtPhantom1/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom1/ColorBy       = "Sum"
sv:Sc/DoseAtPhantom1/ColorNames = 14
"white"
"grey240"
"grey220"
"grey200"
"grey180"
"grey160"
"grey140"
"grey120"
"grey100"
"grey080"
"grey060"
"grey040"
"grey020"
"black"

dv:Sc/DoseAtPhantom1/ColorValues = 13
0.
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
MeV

b:Sc/DoseAtPhantom1/OutputAfterRun = "True"
```

(continues on next page)

(continued from previous page)

```

s:Sc/DoseAtPhantom2/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantom2/Component     = "Phantom2"
s:Sc/DoseAtPhantom2/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom2/ColorBy       = "Sum"
sv:Sc/DoseAtPhantom2/ColorNames  = 14
"white"
"grey240"
"grey220"
"grey200"
"grey180"
"grey160"
"grey140"
"grey120"
"grey100"
"grey080"
"grey060"
"grey040"
"grey020"
"black"

dv:Sc/DoseAtPhantom2/ColorValues = 13
0.
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
MeV

b:Sc/DoseAtPhantom2/OutputAfterRun = "True"

s:Sc/DoseAtPhantom3/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantom3/Component     = "Phantom3"
s:Sc/DoseAtPhantom3/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom3/ColorBy       = "Sum"
sv:Sc/DoseAtPhantom3/ColorNames  = 14
"white"
"grey240"
"grey220"
"grey200"
"grey180"
"grey160"
"grey140"
"grey120"
"grey100"
"grey080"
"grey060"
"grey040"

```

(continues on next page)

(continued from previous page)

```
"grey020"
"black"

dv:Sc/DoseAtPhantom3/ColorValues = 13
0.
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
MeV

b:Sc/DoseAtPhantom3/OutputAfterRun = "True"

s:Sc/DoseAtPhantom4/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantom4/Component    = "Phantom4"
s:Sc/DoseAtPhantom4/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom4/ColorBy      = "Sum"
sv:Sc/DoseAtPhantom4/ColorNames = 14
"white"
"grey240"
"grey220"
"grey200"
"grey180"
"grey160"
"grey140"
"grey120"
"grey100"
"grey080"
"grey060"
"grey040"
"grey020"
"black"

dv:Sc/DoseAtPhantom4/ColorValues = 13
0.
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
```

(continues on next page)

(continued from previous page)

```

MeV

b:Sc/DoseAtPhantom4/OutputAfterRun = "True"

s:Sc/DoseAtPhantom5/Quantity      = "EnergyDeposit"
s:Sc/DoseAtPhantom5/Component     = "Phantom5"
s:Sc/DoseAtPhantom5/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/DoseAtPhantom5/ColorBy       = "Sum"
sv:Sc/DoseAtPhantom5/ColorNames  = 14
"white"
"grey240"
"grey220"
"grey200"
"grey180"
"grey160"
"grey140"
"grey120"
"grey100"
"grey080"
"grey060"
"grey040"
"grey020"
"black"

dv:Sc/DoseAtPhantom5/ColorValues = 13
0.
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
MeV

b:Sc/DoseAtPhantom5/OutputAfterRun = "True"

```

## 28.7 Logo.txt

```

# Constructs the name TOPAS out of a variety of geometry components
# and sends particles through this geometry.

b:Ge/World/Invisible = "True"

#----- Letter T -----
s:Ge/Letter_T/Parent  = "World"
s:Ge/Letter_T/Type    = "Group"

```

(continues on next page)

(continued from previous page)

```

d:Ge/Letter_T/TransX = 0. m
d:Ge/Letter_T/TransY = 0.5 m
d:Ge/Letter_T/TransZ = 0. m
d:Ge/Letter_T/RotX = 0. deg
d:Ge/Letter_T/RotY = 0. deg
d:Ge/Letter_T/RotZ = 0. deg
#b:Ge/Letter_T/Include = "False"

s:Ge/T_1st/Type = "TsBox"
s:Ge/T_1st/Material = "Vacuum"
s:Ge/T_1st/Parent = "Letter_T"
d:Ge/T_1st/HLX = 50.0 cm
d:Ge/T_1st/HLY = 10.0 cm
d:Ge/T_1st/HLZ = 10.0 cm
d:Ge/T_1st/TransX = 0. m
d:Ge/T_1st/TransY = 0. m
d:Ge/T_1st/TransZ = 0. m
d:Ge/T_1st/RotX = 0. deg
d:Ge/T_1st/RotY = 0. deg
d:Ge/T_1st/RotZ = 0. deg

s:Ge/T_2nd/Type = "TsCylinder"
#G4_Water doesn't work, CASE sensitive
s:Ge/T_2nd/Material = "G4_WATER"
s:Ge/T_2nd/Parent = "Letter_T"
d:Ge/T_2nd/RMin = 6.0 cm
d:Ge/T_2nd/RMax = 10.0 cm
d:Ge/T_2nd/HL = 50.0 cm
d:Ge/T_2nd/SPhi = 0.0 deg
d:Ge/T_2nd/DPhi = 360.0 deg
d:Ge/T_2nd/TransX = 0.0 cm
d:Ge/T_2nd/TransY = -1.25 * Ge/T_2nd/HL cm
d:Ge/T_2nd/TransZ = 0.0 cm
d:Ge/T_2nd/RotX = 90.0 deg
d:Ge/T_2nd/RotY = 0.0 deg
d:Ge/T_2nd/RotZ = 0.0 deg

#----- Letter O -----
s:Ge/Letter_O/Parent = "World"
s:Ge/Letter_O/Type = "Group"
d:Ge/Letter_O/TransX = 1.0 m
d:Ge/Letter_O/TransY = 0. m
d:Ge/Letter_O/TransZ = 0. m
d:Ge/Letter_O/RotX = 0. deg
d:Ge/Letter_O/RotY = 0. deg
d:Ge/Letter_O/RotZ = 0. deg
#b:Ge/Letter_O/Include = "False"

s:Ge/O_Left/Type = "G4Cons"
s:Ge/O_Left/Material = "Aluminum"
s:Ge/O_Left/Parent = "Letter_O"
#RM[in,max]1 : -Z surface
#RM[in,max]2 : +Z surface
d:Ge/O_Left/RMin1 = 38.0 cm
d:Ge/O_Left/RMax1 = 45.0 cm
d:Ge/O_Left/RMin2 = 45.0 cm
d:Ge/O_Left/RMax2 = 50.0 cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/O_Left/HL      = 5.0 cm
#Sphi : angle from x axis
d:Ge/O_Left/SPhi    = 90.0 deg
d:Ge/O_Left/DPhi    = 180.0 deg #- doesn't work
d:Ge/O_Left/TransX  = 0. m
d:Ge/O_Left/TransY  = 0. m
d:Ge/O_Left/TransZ  = 0. m
d:Ge/O_Left/RotX    = 0. deg
d:Ge/O_Left/RotY    = 0. deg
d:Ge/O_Left/RotZ    = 0. deg

s:Ge/O_Right/Type   = "TsSphere"
s:Ge/O_Right/Material = "Carbon"
s:Ge/O_Right/Parent  = "Letter_O"
#RM[in,max]1 : -Z surface
#RM[in,max]2 : +Z surface
d:Ge/O_Right/RMin   = 45.0 cm
d:Ge/O_Right/RMax   = 50.0 cm
#Sphi : angle from x axis
d:Ge/O_Right/SPhi   = 271.0 deg
d:Ge/O_Right/DPhi   = 180.0 deg
#STheta : angle from Y rotation
d:Ge/O_Right/STheta = 90.0 deg
d:Ge/O_Right/DTheta = 45.0 deg
#When 180, it should cover half but actually not.
#need for more checks!
d:Ge/O_Right/TransX  = 0. m
d:Ge/O_Right/TransY  = 0. m
d:Ge/O_Right/TransZ  = 0. m
d:Ge/O_Right/RotX    = 0. deg
d:Ge/O_Right/RotY    = 0. deg
d:Ge/O_Right/RotZ    = 0. deg

#----- Letter P -----
s:Ge/Letter_P/Parent = "World"
s:Ge/Letter_P/Type   = "Group"
d:Ge/Letter_P/TransX  = 1.7 m
d:Ge/Letter_P/TransY  = 0. m
d:Ge/Letter_P/TransZ  = 0. m
d:Ge/Letter_P/RotX    = 0. deg
d:Ge/Letter_P/RotY    = 0. deg
d:Ge/Letter_P/RotZ    = 0. deg
#b:Ge/Letter_P/Include = "False"

s:Ge/P_1st/Type      = "G4EllipticalTube"
s:Ge/P_1st/Material  = "Lexan"
s:Ge/P_1st/Parent    = "Letter_P"
d:Ge/P_1st/HLX       = 9.0 cm
d:Ge/P_1st/HLY       = 7.0 cm
d:Ge/P_1st/HLZ       = 25.0 cm
d:Ge/P_1st/TransX    = 0.0 cm
d:Ge/P_1st/TransY    = Ge/P_1st/HLZ cm
d:Ge/P_1st/TransZ    = 0.0 cm
d:Ge/P_1st/RotX      = 90.0 deg
d:Ge/P_1st/RotY      = 0.0 deg
d:Ge/P_1st/RotZ      = 0.0 deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/P_2nd/Type      = "G4Trd"
s:Ge/P_2nd/Material  = "Kapton"
s:Ge/P_2nd/Parent    = "Letter_P"
d:Ge/P_2nd/HLX1      = 10.0 cm
d:Ge/P_2nd/HLY1      = 8.0 cm
d:Ge/P_2nd/HLX2      = 9.5 cm
d:Ge/P_2nd/HLY2      = 7.5 cm
d:Ge/P_2nd/HLZ       = 25.0 cm
d:Ge/P_2nd/TransX    = 0.0 cm
d:Ge/P_2nd/TransY    = -1.0 * Ge/P_2nd/HLZ cm
d:Ge/P_2nd/TransZ    = 0.0 cm
d:Ge/P_2nd/RotX      = 90.0 deg
d:Ge/P_2nd/RotY      = 0.0 deg
d:Ge/P_2nd/RotZ      = 0.0 deg

s:Ge/P_3rd/Type      = "G4Torus"
s:Ge/P_3rd/Material  = "Lucite"
s:Ge/P_3rd/Parent    = "Letter_P"
d:Ge/P_3rd/RMin      = 5.0 cm
d:Ge/P_3rd/RMax      = 6.0 cm
d:Ge/P_3rd/RTor      = 25.0 cm
d:Ge/P_3rd/SPhi      = 265.0 deg
d:Ge/P_3rd/DPhi      = 190.0 deg
d:Ge/P_3rd/TransX    = -8.0 cm + Ge/P_3rd/RTor
d:Ge/P_3rd/TransY    = Ge/P_3rd/RTor cm
d:Ge/P_3rd/TransZ    = 0.0 cm
d:Ge/P_3rd/RotX      = 0.0 deg
d:Ge/P_3rd/RotY      = 0.0 deg
d:Ge/P_3rd/RotZ      = 0.0 deg

#----- Letter A -----
s:Ge/Letter_A/Parent = "World"
s:Ge/Letter_A/Type   = "Group"
d:Ge/Letter_A/TransX = 2.7 m
d:Ge/Letter_A/TransY = 0. m
d:Ge/Letter_A/TransZ = 0. m
d:Ge/Letter_A/RotX   = 0. deg
d:Ge/Letter_A/RotY   = 0. deg
d:Ge/Letter_A/RotZ   = 0. deg
#b:Ge/Letter_A/Include = "False"

#HLZ, Theta, Phi, HLY1,
#HLX1, HLX2, HLX3, HLX4,
#Alp2, Alp1, HLY2,
s:Ge/A_1st/Type      = "G4Para"
s:Ge/A_1st/Material  = "Brass"
s:Ge/A_1st/Parent    = "Letter_A"
d:Ge/A_1st/HLX       = 9.0 cm
d:Ge/A_1st/HLY       = 50.0 cm
d:Ge/A_1st/HLZ       = 7.0 cm
d:Ge/A_1st/Alpha     = 15.0 deg
d:Ge/A_1st/Theta     = 0.0 deg
d:Ge/A_1st/Phi       = 0.0 deg
d:Ge/A_1st/TransX    = -2.7 * Ge/A_1st/HLX cm
d:Ge/A_1st/TransY    = 0.0 cm
d:Ge/A_1st/TransZ    = 0.0 cm
d:Ge/A_1st/RotX      = 0.0 deg

```

(continues on next page)



(continued from previous page)

```

d:Ge/A_1st/RotY      = 0.0 deg
d:Ge/A_1st/RotZ      = 0.0 deg

#Will Try to build with G4GTrap or G4RTrap
s:Ge/A_2nd/Type      = "G4Para"
s:Ge/A_2nd/Material  = "Mylar"
s:Ge/A_2nd/Parent    = "Letter_A"
d:Ge/A_2nd/HLX       = 9.0 cm
d:Ge/A_2nd/HLY       = 50.0 cm
d:Ge/A_2nd/HLZ       = 7.0 cm
d:Ge/A_2nd/Alpha     = -15.0 deg
d:Ge/A_2nd/Theta     = 0.0 deg
d:Ge/A_2nd/Phi       = 0.0 deg
d:Ge/A_2nd/TransX    = -1.0 * Ge/A_1st/TransX  cm
d:Ge/A_2nd/TransY    = 0.0 cm
d:Ge/A_2nd/TransZ    = 0.0 cm
d:Ge/A_2nd/RotX      = 0.0 deg
d:Ge/A_2nd/RotY      = 0.0 deg
d:Ge/A_2nd/RotZ      = 0.0 deg

s:Ge/A_3rd/Type      = "G4Orb"
s:Ge/A_3rd/Material  = "Titanium"
s:Ge/A_3rd/Parent    = "Letter_A"
d:Ge/A_3rd/R         = 8.0 cm
d:Ge/A_3rd/TransX    = 0.0 cm
d:Ge/A_3rd/TransY    = 0.0 cm
d:Ge/A_3rd/TransZ    = 0.0 cm
d:Ge/A_3rd/RotX      = 0.0 deg
d:Ge/A_3rd/RotY      = 0.0 deg
d:Ge/A_3rd/RotZ      = 0.0 deg

#----- Letter S -----
s:Ge/Letter_S/Parent = "World"
s:Ge/Letter_S/Type   = "Group"
#d:Ge/Letter_S/TransX = 0.0 m
d:Ge/Letter_S/TransX = 3.7 m
d:Ge/Letter_S/TransY = 0. m
d:Ge/Letter_S/TransZ = 0. m
d:Ge/Letter_S/RotX   = 0. deg
d:Ge/Letter_S/RotY   = 0. deg
d:Ge/Letter_S/RotZ   = -20. deg
#b:Ge/Letter_S/Include = "False"

s:Ge/S_1st/Type      = "G4HPolycone"
s:Ge/S_1st/Material  = "Copper"
s:Ge/S_1st/Parent    = "Letter_S"
d:Ge/S_1st/PhiStart  = 350.0 deg
d:Ge/S_1st/PhiTotal  = 225.0 deg
dv:Ge/S_1st/Z        = 6   8.0  4.0  4.0 -4.0 -4.0 -8.0 cm
dv:Ge/S_1st/RInner   = 6   10.0 10.0 10.0 10.0 10.0 10.0 cm
dv:Ge/S_1st/ROuter   = 6   18.0 18.0 30.0 30.0 18.0 18.0 cm
d:Ge/S_1st/TransX    = 10.0 cm
d:Ge/S_1st/TransY    = 22.0 cm
d:Ge/S_1st/TransZ    = 0.0 cm
d:Ge/S_1st/RotX      = 0.0 deg
d:Ge/S_1st/RotY      = 0.0 deg
d:Ge/S_1st/RotZ      = 0.0 deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/S_2nd/Type      = "G4Hype"
s:Ge/S_2nd/Material  = "Brass"
s:Ge/S_2nd/Parent    = "Letter_S"
d:Ge/S_2nd/IR        = 7.0 cm
d:Ge/S_2nd/OR        = 10.0 cm
d:Ge/S_2nd/IS        = 10.0 deg
d:Ge/S_2nd/OS        = 20.0 deg
d:Ge/S_2nd/HLZ       = 10.0 cm
d:Ge/S_2nd/TransX    = 0.0 cm
d:Ge/S_2nd/TransY    = 0.0 cm
d:Ge/S_2nd/TransZ    = 0.0 cm
d:Ge/S_2nd/RotX      = 90.0 deg
d:Ge/S_2nd/RotY      = 35.0 deg
d:Ge/S_2nd/RotZ      = 0.0 deg
#b:Ge/S_2nd/Include  = "false"

s:Ge/S_3rd/Type      = "G4HPolyhedra"
s:Ge/S_3rd/Material  = "Kapton"
s:Ge/S_3rd/Parent    = "Letter_S"
d:Ge/S_3rd/PhiStart  = 170.0 deg
d:Ge/S_3rd/PhiTotal  = 225.0 deg
i:Ge/S_3rd/NSides    = 8
dv:Ge/S_3rd/Z        = 6   8.0  4.0  4.0 -4.0 -4.0 -8.0 cm
dv:Ge/S_3rd/RInner   = 6  10.0 10.0 10.0 10.0 10.0 10.0 cm
dv:Ge/S_3rd/ROuter   = 6  18.0 18.0 30.0 30.0 18.0 18.0 cm
d:Ge/S_3rd/TransX    = -10.0 cm
d:Ge/S_3rd/TransY    = -23.0 cm
d:Ge/S_3rd/TransZ    = 0.0 cm
d:Ge/S_3rd/RotX      = 0.0 deg
d:Ge/S_3rd/RotY      = 0.0 deg
d:Ge/S_3rd/RotZ      = 0.0 deg

# Default Beam position (S)
s:Ge/BeamPosition/Parent = "World"
s:Ge/BeamPosition/Type   = "Group"
d:Ge/BeamPosition/TransX = 0.15 m
d:Ge/BeamPosition/TransY = 0. m
d:Ge/BeamPosition/TransZ = 0 m
#flipped cause the beam flies 0 to +z.
d:Ge/BeamPosition/RotX   = 0. deg
d:Ge/BeamPosition/RotY   = 270. deg
d:Ge/BeamPosition/RotZ   = 0. deg

b:Ge/CheckForOverlaps = "False"

s:Tf/BeamCurrent/Function = "Step"
dv:Tf/BeamCurrent/Times   = 2  18 20 s
iv:Tf/BeamCurrent/Values  = 2  10 0

s:So/Example/Type      = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"

```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = Tf/BeamCurrent/Value

d:Tf/TimeLineEnd                      = 20 s
i:Tf/NumberOfSequentialTimes          = 20

s:Gr/ViewA/Type                      = "OpenGL"
i:Gr/ViewA/WindowSizeX                = 800
i:Gr/ViewA/WindowSizeY                = 800
s:Gr/ViewA/ColorBy                    = "particletype"
s:Gr/ViewA/AxesComponent              = "Letter_O"
d:Gr/ViewA/AxesSize                   = 20.0 cm
u:Gr/ViewA/Zoom                       = 1
s:Gr/ViewA/Projection                 = "Orthogonal"
#s:Gr/ViewA/Projection                = "Perspective"
d:Gr/ViewA/Theta                      = 0.0 deg
d:Gr/ViewA/Phi                       = 90.0 deg

b:Ts/PauseBeforeQuit                  = "True"

```

## 28.8 RotateResizeRecolor.txt

```

# A box is set to rotate, grow and change color
# as a test of the time feature system.

d:Ge/World/HLX = 10. m
d:Ge/World/HLY = 10. m
d:Ge/World/HLZ = 10. m

s:Ge/Box/Parent      = "World"
s:Ge/Box/Type        = "TsBox"
s:Ge/Box/Material     = "Air"
d:Ge/Box/HLX         = Tf/Size/Value m
d:Ge/Box/HLY         = Tf/Size/Value m
d:Ge/Box/HLZ         = 4. m
d:Ge/Box/TransX      = 0. m
d:Ge/Box/TransY      = 0. m
d:Ge/Box/TransZ      = 0. m
d:Ge/Box/RotX        = 0. deg
d:Ge/Box/RotY        = 0. deg
d:Ge/Box/RotZ        = Tf/Box/Value deg
s:Ge/Box/Color       = Tf/Color/Value

s:Tf/Box/Function     = "Linear deg"
d:Tf/Box/Rate         = 1. deg/ms
d:Tf/Box/StartValue   = 0.0 deg

```

(continues on next page)

(continued from previous page)

```

d:Tf/Box/RepetitionInterval = 60. ms

s:Tf/Size/Function           = "Linear cm"
d:Tf/Size/Rate               = 0.1 m/ms
d:Tf/Size/StartValue         = 1.0 m
d:Tf/Size/RepetitionInterval = 60. ms

s:Tf/Color/Function = "Step"
sv:Tf/Color/Values  = 4 "white" "blue" "green" "red"
dv:Tf/Color/Times    = 4 1. 2. 3. 4. ms

i:Tf/Verbosity           = 1
d:Tf/TimelineEnd         = 60.0 ms
i:Tf/NumberOfSequentialTimes = 60

s:Sc/MyScorer/Quantity      = "EnergyDeposit"
s:Sc/MyScorer/Component     = "Box"
b:Sc/MyScorer/OutputToConsole = "1"
s:Sc/MyScorer/IfOutputFileAlreadyExists = "Overwrite"

s:Gr/ViewA/Type             = "OpenGL"
i:Gr/ViewA/WindowSizeX      = 1024
i:Gr/ViewA/WindowSizeY      = 768
d:Gr/ViewA/Theta             = 55 deg
d:Gr/ViewA/Phi               = 20 deg
u:Gr/ViewA/Zoom              = 1.3
b:Gr/ViewA/HiddenLineRemovalForTrajectories = "True"
s:Gr/ViewA/Projection        = "Perspective"
d:Gr/ViewA/PerspectiveAngle  = 30 deg

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"

s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "proton"
d:So/Example/BeamEnergy      = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

b:Ts/PauseBeforeQuit = "True"

```

## 28.9 Rotation.txt

```
# Demonstrates direction of rotations in TOPAS placements
# by rotating a simple structure three ways.
# First, it rotates in the X direction.
# Then, after resetting, it rotates in the Y direction.
# Then, after resetting, it rotates in the Z direction.

d:Ge/World/HLX = 1.1 cm
d:Ge/World/HLY = 1.1 cm
d:Ge/World/HLZ = 1.1 cm
s:Ge/World/Color = "black"

s:Ge/system/Parent = "World"
s:Ge/system/Type = "Group"
d:Ge/system/RotX = Tf/rotationX/Value deg
d:Ge/system/RotY = Tf/rotationY/Value deg
d:Ge/system/RotZ = Tf/rotationZ/Value deg

s:Ge/tet/Parent = "system"
s:Ge/tet/Material = "Air"
s:Ge/tet/Type = "G4Tet"
dv:Ge/tet/Anchor = 3 0 0 1 cm
dv:Ge/tet/P2 = 3 1 0 0 cm
dv:Ge/tet/P3 = 3 0 1 0 cm
dv:Ge/tet/P4 = 3 0 0 0 cm
s:Ge/tet/Color = "yellow"
#d:Ge/tet/RotX = Tf/rotationX/Value deg
#d:Ge/tet/RotY = Tf/rotationY/Value deg
#d:Ge/tet/RotZ = Tf/rotationZ/Value deg

s:Ge/pointz/Parent = "system"
s:Ge/pointz/Material = "Air"
s:Ge/pointz/Type = "G4Orb"
d:Ge/pointz/R = 0.2 mm
s:Ge/pointz/Color = "blue"
s:Ge/pointz/DrawingStyle = "solid"
d:Ge/pointz/TransZ = 1 cm

s:Ge/pointy/Parent = "system"
s:Ge/pointy/Material = "Air"
s:Ge/pointy/Type = "G4Orb"
d:Ge/pointy/R = 0.2 mm
s:Ge/pointy/Color = "grass"
s:Ge/pointy/DrawingStyle = "solid"
d:Ge/pointy/TransY = 1 cm

s:Ge/pointx/Parent = "system"
s:Ge/pointx/Material = "Air"
s:Ge/pointx/Type = "G4Orb"
d:Ge/pointx/R = 0.2 mm
s:Ge/pointx/Color = "red"
s:Ge/pointx/DrawingStyle = "solid"
d:Ge/pointx/TransX = 1 cm

s:Tf/rotationX/Function = "Step"
dv:Tf/rotationX/Times = 288
```

(continues on next page)

(continued from previous page)

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
→ 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
→ 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
→ 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87
→ 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105
→ 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122
→ 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138
→ 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155
→ 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171
→ 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188
→ 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204
→ 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221
→ 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237
→ 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254
→ 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270
→ 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287
→ 288 ms
dv:Tf/rotationX/Values = 288
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
→ 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
→ 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
→ 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87
→ 88 89 90 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 deg
s:Tf/rotationY/Function = "Step"
dv:Tf/rotationY/Times = 288
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
→ 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
→ 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
→ 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87
→ 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106
→ 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123
→ 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139
→ 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156
→ 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172
→ 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189
→ 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205
→ 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222
→ 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238
→ 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255
→ 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271
→ 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288
→ ms
dv:Tf/rotationY/Values = 288
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
→ 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54
→ 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 (continues on next page)
→ 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 deg
```

(continued from previous page)

```

s:Tf/rotationZ/Function = "Step"
dv:Tf/rotationZ/Times = 288
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
→ 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
→ 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66
→ 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87
→ 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106
→ 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123
→ 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139
→ 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156
→ 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172
→ 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189
→ 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205
→ 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222
→ 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238
→ 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255
→ 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271
→ 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288
→ ms
dv:Tf/rotationZ/Values = 288
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
→ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
→ 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44
→ 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65
→ 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86
→ 87 88 89 90 deg

d:Tf/TimelineEnd = 288 ms
i:Tf/NumberOfSequentialTimes = 288
b:Ts/ShowCPUTime = "True"

s:Gr/view/Type = "OpenGL"
b:Gr/view/IncludeAxes = "True"
d:Gr/view/AxesSize = 1 cm
d:Gr/view/Theta = 45 deg
d:Gr/view/Phi = 45 deg
u:Gr/view/Zoom = 1.8

sv:Ph/Default/Modules = 1 "g4em-standard_opt0"
b:Ge/CheckForOverlaps = "False"

b:Ts/PauseBeforeQuit = "True"

s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "proton"
d:So/Example/BeamEnergy = 169.23 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"

```

(continues on next page)

(continued from previous page)

```

d:So/Example/BeamPositionCutoffX      = 10. cm
d:So/Example/BeamPositionCutoffY      = 10. cm
d:So/Example/BeamPositionSpreadX      = 0.65 cm
d:So/Example/BeamPositionSpreadY      = 0.65 cm
s:So/Example/BeamAngularDistribution  = "Gaussian"
d:So/Example/BeamAngularCutoffX       = 90. deg
d:So/Example/BeamAngularCutoffY       = 90. deg
d:So/Example/BeamAngularSpreadX       = 0.0032 rad
d:So/Example/BeamAngularSpreadY       = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun   = 1

```

## 28.10 RunRandom\_Mode.txt

```

includeFile = RunSequential_Mode.txt

#####
#--- Timeline setting
#####
b:Tf/RandomizeTimeDistribution = "True"

#####
#--- Time Feature for Probability
#####
s:Tf/Weights/Function = "Step"
dv:Tf/Weights/Times   = 5   1   2   3   4   5   s
uv:Tf/Weights/Values  = 5 1.0 0.8 0.6 0.4 0.2

#####
#--- Assign particles
#####
#Same number of total particles in RunSequential_Mode.txt
i:So/Example/NumberOfHistoriesInRandomJob = 3000

u:So/Example/ProbabilityOfUsingAGivenRandomTime = Tf/Weights/Value
i:Ts/ShowHistoryCountAtInterval = 0

#####
#--- Dose for DoseGrid
#####
s:Sc/dose/OutputFile = "Dose_Rand"

```

## 28.11 RunSequential\_Mode.txt

```

# Demonstration of TOPAS time modes: Sequential Time Mode

#####
#--- Time Feature for Energy and Number of particles
#####
s:Tf/Energies/Function = "Step"
dv:Tf/Energies/Times   = 5 1 2 3 4 5   s

```

(continues on next page)



(continued from previous page)

```

dv:Tf/Energies/Values      = 5 110.0 100.0 90.0 80.0 70.0 MeV

s:Tf/Particles/Function = "Step"
dv:Tf/Particles/Times    = 5 1 2 3 4 5 s
iv:Tf/Particles/Values   = 5 1000 800 600 400 200

#-----#
#--- Timeline setting
#-----#
d:Tf/TimelineStart = 0.0 s
d:Tf/TimelineEnd   = 5.0 s

#For Sequential Time Mode
i:Tf/NumberOfSequentialTimes = 5

#-----#
#--- Assign particles
#-----#
s:So/Example/Type           = "Beam"
s:So/Example/Component      = "BeamPosition"
s:So/Example/BeamParticle   = "proton"
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad

#Change number of particles per time stamps
#total particles are 1000 + 800 + 600 + 400 + 200 = 3000
i:So/Example/NumberOfHistoriesInRun = Tf/Particles/Value

#Energy changes
d:So/Example/BeamEnergy = Tf/Energies/Value MeV
#From 10.0 cm
d:Ge/BeamPosition/TransZ = 10.0 cm

#-----#
#--- Water phantom target
#-----#
s:Ge/waterphantom/Type      = "TsBox"
s:Ge/waterphantom/Parent    = "World"
s:Ge/waterphantom/Material  = "G4_WATER"
d:Ge/waterphantom/HLX       = 5 cm
d:Ge/waterphantom/HLY       = 5 cm
d:Ge/waterphantom/HLZ       = 6 cm
d:Ge/waterphantom/TransX     = 0. cm
d:Ge/waterphantom/TransY     = 0. cm
d:Ge/waterphantom/Upstream   = 0.0 cm
d:Ge/waterphantom/TransZ     = -1.0 * Ge/waterphantom/HLZ cm
d:Ge/waterphantom/RotX       = 0. deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/waterphantom/RotY      = 0. deg
d:Ge/waterphantom/RotZ      = 0. deg
i:Ge/waterphantom/XBins     = 1
i:Ge/waterphantom/YBins     = 1
i:Ge/waterphantom/ZBins     = 1
s:Ge/waterphantom/Color     = "red"

#=====
#--- Dose grid for 1D PDD
#=====
s:Ge/dosegrid/Type          = "TsCylinder"
s:Ge/dosegrid/Parent        = "waterphantom"
d:Ge/dosegrid/RMin          = 0.0 cm
d:Ge/dosegrid/RMax          = 5 cm
d:Ge/dosegrid/SPhi          = 0.0 deg
d:Ge/dosegrid/DPhi          = 360.0 deg
d:Ge/dosegrid/HL            = Ge/waterphantom/HLZ cm
d:Ge/dosegrid/TransX        = 0. cm
d:Ge/dosegrid/TransY        = 0. cm
d:Ge/dosegrid/TransZ        = 0. cm
d:Ge/dosegrid/RotX          = 0. deg
d:Ge/dosegrid/RotY          = 0. deg
d:Ge/dosegrid/RotZ          = 0. deg
i:Ge/dosegrid/RBins         = 1
i:Ge/dosegrid/PhiBins       = 1
i:Ge/dosegrid/ZBins         = 120
b:Ge/dosegrid/IsParallel    = "T"

#=====
#--- Dose for DoseGrid
#=====
s:Sc/dose/Quantity          = "DoseToMedium"
s:Sc/dose/Component         = "dosegrid"
s:Sc/dose/OutputFile        = "Dose_Seq"
s:Sc/dose/OutputType        = "binary"
b:Sc/dose/OutputToConsole   = "F"
b:Sc/dose/Visualize         = "F"
s:Sc/dose/IfOutputFileAlreadyExists = "Overwrite"

#=====
#--- Summary
#=====
b:Ts/ShowCPUTime            = "tRuE" #For checking case sensitivity
i:Ts/ShowHistoryCountAtInterval = 10000

```



## CHAPTER 29

---

### UCSFETF

---

**29.1 UCSF\_Beamline\_Base.txt**

**29.2 UCSF\_Beamline\_Propeller10.txt**

**29.3 UCSF\_Beamline\_Propeller15.txt**

**29.4 UCSF\_Beamline\_Propeller20.txt**

**29.5 UCSF\_Beamline\_Propeller24.txt**

**29.6 UCSF\_DefaultParameters.txt**

**29.7 UCSF\_WaterPhantom.txt**

**29.8 User\_Beamline\_WC\_Viewer.txt**

**29.9 User\_BP\_R28\_WaterPhantom.txt**

**29.10 User\_SOBP24\_R28\_WaterPhantom.txt**

**29.11 User\_SOBP24\_Viewer.txt**

**29.12 Viewer.txt**

### 30.1 AmbientDoseEquivalent.txt

```
#####
# Simple shielding example
#####
d:Ge/World/HLX = 2 m
d:Ge/World/HLY = 2 m
d:Ge/World/HLZ = 2.1 m

#####
# Gemetry for vrt
#####
d:Ge/ScorerHLX = 1 m
d:Ge/ScorerHLY = 1 m
d:Ge/ScorerHLZ = 1 m

d:Ge/subComponent/Thickness      = 20 cm
d:Ge/subComponent/StartPosition = -1.0 * Ge/ScorerHLZ cm

s:Ge/subComponent1/Material      = "G4_CONCRETE"
s:Ge/subComponent2/Material      = "G4_CONCRETE"
s:Ge/subComponent3/Material      = "G4_CONCRETE"
s:Ge/subComponent4/Material      = "G4_CONCRETE"
s:Ge/subComponent5/Material      = "G4_CONCRETE"
s:Ge/subComponent6/Material      = "G4_CONCRETE"
s:Ge/subComponent7/Material      = "G4_CONCRETE"
s:Ge/subComponent8/Material      = "G4_CONCRETE"
s:Ge/subComponent9/Material      = "G4_CONCRETE"
s:Ge/subComponent10/Material     = "G4_CONCRETE"
s:Ge/subComponent11/Material     = "G4_CONCRETE"

s:Ge/subComponent1/Type          = "TsBox"
s:Ge/subComponent1/Parent        = "World"
```

(continues on next page)

(continued from previous page)

```

d:Ge/subComponent1/HLX      = Ge/ScorerHLX m
d:Ge/subComponent1/HLY      = Ge/ScorerHLY m
d:Ge/subComponent1/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent1/TransX    = 0 cm
d:Ge/subComponent1/TransY    = 0 cm
d:Ge/subComponent1/TransZ    = Ge/subComponent/StartPosition + Ge/subComponent1/HLZ cm
d:Ge/subComponent1/RotX      = 0 deg
d:Ge/subComponent1/RotY      = 0 deg
d:Ge/subComponent1/RotZ      = 0 deg

s:Ge/subComponent2/Type      = "TsBox"
s:Ge/subComponent2/Parent    = "World"
d:Ge/subComponent2/HLX      = Ge/ScorerHLX m
d:Ge/subComponent2/HLY      = Ge/ScorerHLY m
d:Ge/subComponent2/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent2/TransX    = 0 cm
d:Ge/subComponent2/TransY    = 0 cm
d:Ge/subComponent2/TransZTmp = Ge/subComponent1/TransZ + Ge/subComponent1/HLZ cm
d:Ge/subComponent2/TransZ    = Ge/subComponent2/TransZTmp + Ge/subComponent2/HLZ cm
d:Ge/subComponent2/RotX      = 0 deg
d:Ge/subComponent2/RotY      = 0 deg
d:Ge/subComponent2/RotZ      = 0 deg

s:Ge/subComponent3/Type      = "TsBox"
s:Ge/subComponent3/Parent    = "World"
d:Ge/subComponent3/HLX      = Ge/ScorerHLX m
d:Ge/subComponent3/HLY      = Ge/ScorerHLY m
d:Ge/subComponent3/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent3/TransX    = 0 cm
d:Ge/subComponent3/TransY    = 0 cm
d:Ge/subComponent3/TransZTmp = Ge/subComponent2/TransZ + Ge/subComponent2/HLZ cm
d:Ge/subComponent3/TransZ    = Ge/subComponent3/TransZTmp + Ge/subComponent3/HLZ cm
d:Ge/subComponent3/RotX      = 0 deg
d:Ge/subComponent3/RotY      = 0 deg
d:Ge/subComponent3/RotZ      = 0 deg

s:Ge/subComponent4/Type      = "TsBox"
s:Ge/subComponent4/Parent    = "World"
d:Ge/subComponent4/HLX      = Ge/ScorerHLX m
d:Ge/subComponent4/HLY      = Ge/ScorerHLY m
d:Ge/subComponent4/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent4/TransX    = 0 cm
d:Ge/subComponent4/TransY    = 0 cm
d:Ge/subComponent4/TransZTmp = Ge/subComponent3/TransZ + Ge/subComponent3/HLZ cm
d:Ge/subComponent4/TransZ    = Ge/subComponent4/TransZTmp + Ge/subComponent4/HLZ cm
d:Ge/subComponent4/RotX      = 0 deg
d:Ge/subComponent4/RotY      = 0 deg
d:Ge/subComponent4/RotZ      = 0 deg

s:Ge/subComponent5/Type      = "TsBox"
s:Ge/subComponent5/Parent    = "World"
d:Ge/subComponent5/HLX      = Ge/ScorerHLX m
d:Ge/subComponent5/HLY      = Ge/ScorerHLY m
d:Ge/subComponent5/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent5/TransX    = 0 cm
d:Ge/subComponent5/TransY    = 0 cm
d:Ge/subComponent5/TransZTmp = Ge/subComponent4/TransZ + Ge/subComponent4/HLZ cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/subComponent5/TransZ      = Ge/subComponent5/TransZTmp + Ge/subComponent5/HLZ cm
d:Ge/subComponent5/RotX        = 0 deg
d:Ge/subComponent5/RotY        = 0 deg
d:Ge/subComponent5/RotZ        = 0 deg

s:Ge/subComponent6/Type        = "TsBox"
s:Ge/subComponent6/Parent      = "World"
d:Ge/subComponent6/HLX         = Ge/ScorerHLX m
d:Ge/subComponent6/HLY         = Ge/ScorerHLY m
d:Ge/subComponent6/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent6/TransX      = 0 cm
d:Ge/subComponent6/TransY      = 0 cm
d:Ge/subComponent6/TransZTmp   = Ge/subComponent5/TransZ + Ge/subComponent5/HLZ cm
d:Ge/subComponent6/TransZ      = Ge/subComponent6/TransZTmp + Ge/subComponent6/HLZ cm
d:Ge/subComponent6/RotX        = 0 deg
d:Ge/subComponent6/RotY        = 0 deg
d:Ge/subComponent6/RotZ        = 0 deg

s:Ge/subComponent7/Type        = "TsBox"
s:Ge/subComponent7/Parent      = "World"
d:Ge/subComponent7/HLX         = Ge/ScorerHLX m
d:Ge/subComponent7/HLY         = Ge/ScorerHLY m
d:Ge/subComponent7/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent7/TransX      = 0 cm
d:Ge/subComponent7/TransY      = 0 cm
d:Ge/subComponent7/TransZTmp   = Ge/subComponent6/TransZ + Ge/subComponent6/HLZ cm
d:Ge/subComponent7/TransZ      = Ge/subComponent7/TransZTmp + Ge/subComponent7/HLZ cm
d:Ge/subComponent7/RotX        = 0 deg
d:Ge/subComponent7/RotY        = 0 deg
d:Ge/subComponent7/RotZ        = 0 deg

s:Ge/subComponent8/Type        = "TsBox"
s:Ge/subComponent8/Parent      = "World"
d:Ge/subComponent8/HLX         = Ge/ScorerHLX m
d:Ge/subComponent8/HLY         = Ge/ScorerHLY m
d:Ge/subComponent8/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent8/TransX      = 0 cm
d:Ge/subComponent8/TransY      = 0 cm
d:Ge/subComponent8/TransZTmp   = Ge/subComponent7/TransZ + Ge/subComponent7/HLZ cm
d:Ge/subComponent8/TransZ      = Ge/subComponent8/TransZTmp + Ge/subComponent8/HLZ cm
d:Ge/subComponent8/RotX        = 0 deg
d:Ge/subComponent8/RotY        = 0 deg
d:Ge/subComponent8/RotZ        = 0 deg

s:Ge/subComponent9/Type        = "TsBox"
s:Ge/subComponent9/Parent      = "World"
d:Ge/subComponent9/HLX         = Ge/ScorerHLX m
d:Ge/subComponent9/HLY         = Ge/ScorerHLY m
d:Ge/subComponent9/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent9/TransX      = 0 cm
d:Ge/subComponent9/TransY      = 0 cm
d:Ge/subComponent9/TransZTmp   = Ge/subComponent8/TransZ + Ge/subComponent8/HLZ cm
d:Ge/subComponent9/TransZ      = Ge/subComponent9/TransZTmp + Ge/subComponent9/HLZ cm
d:Ge/subComponent9/RotX        = 0 deg
d:Ge/subComponent9/RotY        = 0 deg
d:Ge/subComponent9/RotZ        = 0 deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/subComponent10/Type      = "TsBox"
s:Ge/subComponent10/Parent    = "World"
d:Ge/subComponent10/HLX      = Ge/ScorerHLX m
d:Ge/subComponent10/HLY      = Ge/ScorerHLY m
d:Ge/subComponent10/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent10/TransX    = 0 cm
d:Ge/subComponent10/TransY    = 0 cm
d:Ge/subComponent10/TransZTmp = Ge/subComponent9/TransZ + Ge/subComponent9/HLZ cm
d:Ge/subComponent10/TransZ    = Ge/subComponent10/TransZTmp + Ge/subComponent10/HLZ cm
d:Ge/subComponent10/RotX      = 0 deg
d:Ge/subComponent10/RotY      = 0 deg
d:Ge/subComponent10/RotZ      = 0 deg

s:Ge/subComponent11/Type      = "TsBox"
s:Ge/subComponent11/Parent    = "World"
d:Ge/subComponent11/HLX      = Ge/ScorerHLX m
d:Ge/subComponent11/HLY      = Ge/ScorerHLY m
d:Ge/subComponent11/HLZ      = 0.05 mm
d:Ge/subComponent11/TransX    = 0 cm
d:Ge/subComponent11/TransY    = 0 cm
d:Ge/subComponent11/TransZTmp = Ge/subComponent10/TransZ + Ge/subComponent10/HLZ cm
d:Ge/subComponent11/TransZ    = Ge/subComponent11/TransZTmp + Ge/subComponent11/HLZ cm
d:Ge/subComponent11/RotX      = 0 deg
d:Ge/subComponent11/RotY      = 0 deg
d:Ge/subComponent11/RotZ      = 0 deg

#####
# VRT
#####
b:Vr/UseVarianceReduction      = "true"
b:Vr/ParticleSplit/Active      = "true"
sv:Vr/ParticleSplit/ParticleName = 1 "neutron"
s:Vr/ParticleSplit/Component    = "subComponent1"
sv:Vr/ParticleSplit/SubComponents = 10
"subComponent2"
"subComponent3" "subComponent4"
"subComponent5" "subComponent6"
"subComponent7" "subComponent8"
"subComponent9" "subComponent10" "subComponent11"
s:Vr/ParticleSplit/Type = "ImportanceSampling"
uv:Vr/ParticleSplit/ImportanceValues = 10 2 4 8 16 32 64 128 256 512 512

#####
# Scorer
#####
# Fix binwidth
sv:Sc/scorerFix/OnlyIncludeParticlesNamed = 1 "neutron"
s:Sc/scorerFix/Quantity                    = "Fluence"
s:Sc/scorerFix/Component                    = "subComponent11"
s:Sc/scorerFix/OutputFile                   = "FluenceSpectra_FixedBin"
s:Sc/scorerFix/OutputType                   = "Binary"
s:Sc/scorerFix/IfOutputFileAlreadyExists    = "Overwrite"
i:Sc/scorerFix/EBins                       = 100
d:Sc/scorerFix/EBinMin                     = 0.01 MeV
d:Sc/scorerFix/EBinMax                     = 110.1 MeV

# Logarithmic binwidth

```

(continues on next page)



(continued from previous page)

```

sv:Sc/scorerLog/OnlyIncludeParticlesNamed = 1 "neutron"
s:Sc/scorerLog/Quantity                  = "Fluence"
s:Sc/scorerLog/Component                  = "subComponent11"
s:Sc/scorerLog/OutputFile                 = "FluenceSpectra_LogBin"
s:Sc/scorerLog/OutputType                 = "Binary"
s:Sc/scorerLog/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/scorerLog/EBins                     = 100
d:Sc/scorerLog/EBinMin                   = 0.01 MeV
d:Sc/scorerLog/EBinMax                   = 110.1 MeV
b:Sc/scorerLog/EBinLog                    = "True"

# Score dose to medium
sv:Sc/scorerDosePerSource/OnlyIncludeParticlesNamed = 1 "neutron"
s:Sc/scorerDosePerSource/Quantity                  = "DoseToMedium"
s:Sc/scorerDosePerSource/Component                  = "subComponent11"
s:Sc/scorerDosePerSource/OutputFile                 = "DoseNeutronPerSourceNeutron"
b:Sc/scorerDosePerSource/OutputToConsole            = "True"
s:Sc/scorerDosePerSource/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/scorerDosePerSource/Report                    = 1 "Mean"

# Score ambient dose equivalent for neutrons
sv:Sc/scorerAmbDosePerSource/OnlyIncludeParticlesNamed = 1 "neutron"
s:Sc/scorerAmbDosePerSource/Quantity                  = "AmbientDoseEquivalent"
s:Sc/scorerAmbDosePerSource/Component                  = "subComponent11"
s:Sc/scorerAmbDosePerSource/OutputFile                 =
→ "AmbientDoseNeutronPerSourceNeutron"
b:Sc/scorerAmbDosePerSource/OutputToConsole            = "True"
s:Sc/scorerAmbDosePerSource/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/scorerAmbDosePerSource/Report                    = 1 "Mean"

# Set the fluence-to-dose conversion factors.
s:Sc/scorerAmbDosePerSource/GetAmbientDoseEquivalentForParticleNamed = "neutron"
dv:Sc/scorerAmbDosePerSource/FluenceToDoseConversionEnergies = 58
2.5314e-08 7.71605e-08 2.35195e-07 6.33404e-07 1.70582e-06 4.05885e-06 1.02746e-05 2.
→ 44475e-05 6.18866e-05 0.000142765
0.000309568 0.000611723 0.00100388 0.00150131 0.00217678 0.00305995 0.00430144 0.
→ 00604662 0.00849986 0.0119484
0.0157877 0.0221931 0.0293242 0.0399651 0.0511969 0.0676476 0.0866593 0.
→ 101168 0.1296 0.171243
0.233382 0.289858 0.37132 0.490632 0.590784 0.711379 1 1.
→ 85741 2.95521 4.95083
9.98711 14.7825 18.937 20.1466 48.9368 98.7183 195.073 504.
→ 105 996.138 2182.41
5086.78 9846.4 29400 99357.1 302853 982103 3.05600e+06 9.
→ 91011e+06 MeV
dv:Sc/scorerAmbDosePerSource/FluenceToDoseConversionValues = 58
1.04694e-09 1.0279e-09 1.00922e-09 9.90868e-10 9.72854e-10 9.55168e-10 9.37803e-10 9.
→ 29239e-10 9.12346e-10 8.95759e-10
8.79474e-10 8.71443e-10 8.63485e-10 1.07615e-09 1.31681e-09 1.59657e-09 1.91809e-09 2.
→ 32559e-09 2.79393e-09 3.35658e-09
3.99571e-09 4.7131e-09 5.50853e-09 6.55742e-09 7.52478e-09 8.71443e-09 9.8182e-09 1.
→ 09608e-08 1.29287e-08 1.56754e-08
1.86602e-08 2.18095e-08 2.59623e-08 3.06236e-08 3.51412e-08 3.95922e-08 4.93432e-08 4.
→ 67006e-08 4.46069e-08 4.26071e-08
4.69871e-08 5.15014e-08 5.64495e-08 5.24551e-08 3.61218e-08 2.6443e-08 2.2418e-08 2.
→ 95202e-08 3.7933e-08 4.31315e-08
4.93432e-08 5.27769e-08 5.15014e-08 5.05651e-08 5.96437e-08 7.20944e-08 9.0956e-08 1.
→ 17594e-07 Sv*mm2

```

(continues on next page)

(continued from previous page)

```
#####
# Beam setting
#####
s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "neutron"
d:So/Example/BeamEnergy = 100.0 MeV
u:So/Example/BeamEnergySpread = 0.5
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1000

d:Ge/BeamPosition/TransZ = -1.0 * Ge/World/HLZ m
d:Ge/BeamPosition/RotX = 0 deg

i:Ts/ShowHistoryCountAtInterval = 100
```

## 30.2 CrossSectionEnhancement.txt

```
#
i:Ts/NumberOfThreads = 1

# Secondary biasing occurs in the region
# of this component
s:Ge/target/Parent = "World"
s:Ge/target/Material = "Lead"
s:Ge/target/Type = "TsBox"
d:Ge/target/HLX = 1.5 cm
d:Ge/target/HLY = 1.5 cm
d:Ge/target/HLZ = 0.5 mm
s:Ge/target/Color = "yellow"
# Set the region
s:Ge/target/AssignToRegionNaMed = "targetregion"

# Component for scoring
s:Ge/scorer/Parent = "World"
s:Ge/scorer/Material = "G4_WATER"
s:Ge/scorer/Type = "TsSphere"
d:Ge/scorer/RMin = 5 cm
d:Ge/scorer/RMax = 5.01 cm
s:Ge/scorer/Color = "yellow"
s:Ge/scorer/AssignToRegionNamed = "scorerregion"

s:Sc/scorer/Quantity = "EnergyFluence"
```

(continues on next page)

(continued from previous page)

```

s:Sc/scorer/Component = "scorer"
s:Sc/scorer/OutputFile = "secondaryBiasing"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"
i:Sc/scorer/EBins = 50
d:Sc/scorer/EBinMin = 0.01 MeV
d:Sc/scorer/EBinMax = 6 MeV
b:Sc/scorer/EBinLog = "True"

# VISUALIZATION
s:Gr/view/Type = "OpenGL"
i:Gr/view/WindowSizeX = 600
i:Gr/view/WindowSizeY = 600
d:Gr/view/Phi = 0 deg
d:Gr/view/Theta = 270 deg
u:Gr/view/Zoom = 5

b:Ts/PauseBeforeQuit = "True"

# SOURCE
s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 6 MeV
u:So/Example/BeamEnergySpread = 0.1
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad

Ge/BeamPosition/Parent = "World"
Ge/BeamPosition/TransX = 0.0 cm
Ge/BeamPosition/TransZ = -1.0 * Ge/target/HLZ cm
Ge/BeamPosition/TransY = 0.0 mm
Ge/BeamPosition/RotX = 0 deg

# GEOMETRY
d:Ge/World/HLX = 16.1 cm
d:Ge/World/HLY = 16.1 cm
d:Ge/World/HLZ = 1.5 m
s:Ge/World/Material = "Air"

sv:Ph/Default/Modules = 1 "g4em-standard_opt3"

i:So/Example/NumberOfHistoriesInRun = 100
i:Ts/ShowHistoryCountAtInterval = 10
b:Ts/ShowCPUtime = "True"

#Cross section enhancement
#This occurs everywhere
b:Vr/UseVarianceReduction = "true"
s:Vr/CrosssectionEnhancement/Type = "CrosssectionEnhancement"
sv:Vr/CrosssectionEnhancement/processesNamed = 1 "eBrem"
uv:Vr/CrosssectionEnhancement/EnhancementFactor = 1 400

#Brem Uniform splitting in the target region

```

(continues on next page)

(continued from previous page)

```

b:Vr/UniformSplitting/Active = "True"
s:Vr/UniformSplitting/Type   = "UniformSplitting"
sv:Vr/UniformSplitting/forregion/targetregion/processesNamed = 1 "eBrem"
uv:Vr/UniformSplitting/ForRegion/targetregion/SplitNumber   = 1 10
dv:Vr/UniformSplitting/ForRegion/targetregion/MaximumEnergies = 1 6 MeV

#Forced Interaction in the sphere shell
s:Vr/ForcedInteraction/Type = "ForcedInteraction"
sv:Vr/ForcedInteraction/forregion/scorerregion/processesNamed = 1 "compt"
dv:Vr/ForcedInteraction/ForRegion/scorerregion/ForcedDistances = 1 0.001 cm
b:Vr/ForcedInteraction/ForRegion/scorerregion/CorrectByWeight = "True"

```

### 30.3 ForcedInteraction.txt

```

#
i:Ts/NumberOfThreads = 1

# Secondary biasing occurs in the region
# of this component
s:Ge/target/Parent = "World"
s:Ge/target/Material = "G4_WATER"
s:Ge/target/Type = "TsBox"
d:Ge/target/HLX = 1.5 cm
d:Ge/target/HLY = 1.5 cm
d:Ge/target/HLZ = 0.5 mm
s:Ge/target/Color = "yellow"
# Set the region
s:Ge/target/AssignToRegionName = "targetregion"

# Component for scoring
s:Ge/scorer/Parent = "World"
s:Ge/scorer/Material = "G4_WATER"
s:Ge/scorer/Type = "TsCylinder"
d:Ge/scorer/RMax = 5 cm
d:Ge/scorer/HL = 0.05 mm
d:Ge/scorer/SPhi = 0 deg
d:Ge/scorer/DPhi = 360 deg
d:Ge/scorer/TransZ = 1 m + Ge/scorer/HL
s:Ge/scorer/Color = "yellow"
i:Ge/scorer/RBins = 20
i:Ge/scorer/PhiBins = 1
i:Ge/scorer/ZBins = 1

s:Sc/scorer/Quantity = "EnergyFluence"
s:Sc/scorer/Component = "scorer"
s:Sc/scorer/OutputFile = "secondaryBiasing"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"

# VISUALIZATION
s:Gr/view/Type = "OpenGL"
i:Gr/view/WindowSizeX = 600
i:Gr/view/WindowSizeY = 600
d:Gr/view/Phi = 0 deg
d:Gr/view/Theta = 270 deg

```

(continues on next page)

(continued from previous page)

```

u:Gr/view/Zoom          = 5

b:Ts/PauseBeforeQuit = "True"

# SOURCE
s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "gamma"
d:So/Example/BeamEnergy    = 10 MeV
u:So/Example/BeamEnergySpread = 0.1
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad

Ge/BeamPosition/Parent = "World"
Ge/BeamPosition/TransX = 0.0 cm
Ge/BeamPosition/TransZ = -1.0 * Ge/target/HLZ cm
Ge/BeamPosition/TransY = 0.0 mm
Ge/BeamPosition/RotX   = 0 deg

# GEOMETRY
d:Ge/World/HLX         = 16.1 cm
d:Ge/World/HLY         = 16.1 cm
d:Ge/World/HLZ         = 1.5 m
s:Ge/World/Material    = "Air"

sv:Ph/Default/Modules = 1 "g4em-standard_opt3"

i:So/Example/NumberOfHistoriesInRun = 1000
i:Ts/ShowHistoryCountAtInterval = 100
b:Ts/ShowCPUTime = "True"

#Forced interaction in a region
#A distance value shorter than the thickness of the volume
#of interest must be set.
#By setting that distance of interaction large enough,
#then the true physic length is taken into account. This
#will mimic activation deactivation of this technique.
s:Tf/ActiveVRT/Function = "step"
dv:Tf/ActiveVRT/Times   = 5 1 2 3 4 5 ms
dv:Tf/ActiveVRT/Values  = 5 0.1 Ge/World/HLZ 0.1 Ge/World/HLZ 0.1 mm

Tf/TimeLineEnd = 5 ms
Tf/NumberOfSequentialTimes = 5

b:Vr/UseVarianceReduction = "true"
s:Vr/ForcedInteraction/Type = "ForcedInteraction"
sv:Vr/ForcedInteraction/forregion/targetregion/processesNamed = 2 "compt" "conv"
dv:Vr/ForcedInteraction/ForRegion/targetregion/ForcedDistances = 2 Tf/ActiveVRT/
→Value Tf/ActiveVRT/Value mm
b:Vr/ForcedInteraction/ForRegion/targetregion/CorrectByWeight = "True"

```

## 30.4 GeometricalParticleSplit.txt

```
#####
# Time features
#####
s:Tf/Phi/Function      = "Linear deg"
d:Tf/Phi/Rate          = 90. deg/ms
d:Tf/Phi/StartValue    = 45 deg
d:Tf/Phi/RepetitionInterval = 4. ms
d:Tf/TimelineEnd       = 4. ms
i:Tf/NumberOfSequentialTimes = 4

#####
# Geometry
#####
d:Ge/World/HLX        = 1.0 m
d:Ge/World/HLY        = 1.0 m
d:Ge/World/HLZ        = 1.0 m
b:Ge/World/Invisible = "TRUE"

#####
# Phantom
#####
s:Ge/Phantom/Type      = "TsSphere"
s:Ge/Phantom/Parent    = "World"
s:Ge/Phantom/Material  = "G4_WATER"
s:Ge/Phantom/Color     = "blue"
d:Ge/Phantom/RMin      = 0.0 cm
d:Ge/Phantom/RMax      = 10.0 cm
d:Ge/Phantom/SPhi      = 0.0 deg
d:Ge/Phantom/DPhi      = 360 deg
d:Ge/Phantom/STheta    = 0 deg
d:Ge/Phantom/DTheta    = 180 deg
d:Ge/Phantom/TransX    = 0 cm
d:Ge/Phantom/TransY    = 0 cm
d:Ge/Phantom/TransZ    = 0 cm
d:Ge/Phantom/RotX      = 0 deg
d:Ge/Phantom/RotY      = 0 deg
d:Ge/Phantom/RotZ      = 0 deg

#####
# Nozzle
#####
s:Ge/Nozzle/Type       = "Group"
s:Ge/Nozzle/Parent     = "World"
d:Ge/Nozzle/TransX     = 0. cm
d:Ge/Nozzle/TransY     = 0. cm
d:Ge/Nozzle/TransZ     = 0. cm
d:Ge/Nozzle/RotX       = Tf/Phi/Value deg
d:Ge/Nozzle/RotY       = 0 deg
d:Ge/Nozzle/RotZ       = 0 deg
s:Ge/Nozzle/Color      = "blue"

#####
# Scatterers
#####
s:Ge/Scatterer1/Type   = "TsBox"
```

(continues on next page)

(continued from previous page)

```

s:Ge/Scatterer1/Parent      = "Nozzle"
s:Ge/Scatterer1/Material    = "Lead"
d:Ge/Scatterer1/HLX         = 2.0 cm
d:Ge/Scatterer1/HLY         = 2.0 cm
d:Ge/Scatterer1/HLZ         = 0.25 cm
d:Ge/Scatterer1/TransX      = 0. cm
d:Ge/Scatterer1/TransY      = 0. cm
d:Ge/Scatterer1/TransZ      = -50 cm + Ge/Scatterer1/HLZ
d:Ge/Scatterer1/RotX        = 0 deg
d:Ge/Scatterer1/RotY        = 0 deg
d:Ge/Scatterer1/RotZ        = 0 deg
s:Ge/Scatterer1/Color       = "red"

s:Ge/Scatterer2/Type        = "TsBox"
s:Ge/Scatterer2/Parent      = "Nozzle"
s:Ge/Scatterer2/Material    = "Lead"
d:Ge/Scatterer2/HLX         = 4.0 cm
d:Ge/Scatterer2/HLY         = 4.0 cm
d:Ge/Scatterer2/HLZ         = 0.25 cm
d:Ge/Scatterer2/TransX      = 0. cm
d:Ge/Scatterer2/TransY      = 0. cm
d:Ge/Scatterer2/TransZtmp   = -1.0 * Ge/Scatterer2/HLZ cm
d:Ge/Scatterer2/TransZ      = -40 cm + Ge/Scatterer2/TransZtmp
d:Ge/Scatterer2/RotX        = 0 deg
d:Ge/Scatterer2/RotY        = 0 deg
d:Ge/Scatterer2/RotZ        = 0 deg
s:Ge/Scatterer2/Color       = "red"

#####
# Gemetry for vrt
#####
s:Ge/VrtParallelWorld/Type  = "TsBox"
s:Ge/VrtParallelWorld/Parent = "Nozzle"
d:Ge/VrtParallelWorld/HLX   = 15 cm
d:Ge/VrtParallelWorld/HLY   = 15 cm
d:Ge/VrtParallelWorld/HLZ   = 80 cm
d:Ge/VrtParallelWorld/TransX = 0 cm
d:Ge/VrtParallelWorld/TransY = 0 cm
d:Ge/VrtParallelWorld/TransZ = -40 cm
d:Ge/VrtParallelWorld/RotX   = 0 deg
d:Ge/VrtParallelWorld/RotY   = 0 deg
d:Ge/VrtParallelWorld/RotZ   = 0 deg
b:Ge/VrtParallelWorld/IsParallel = "true"

s:Ge/subComponent1/Type      = "TsBox"
s:Ge/subComponent1/Parent    = "VrtParallelWorld"
d:Ge/subComponent1/HLX       = 0.9 * Ge/VrtParallelWorld/HLX cm
d:Ge/subComponent1/HLY       = 0.9 * Ge/VrtParallelWorld/HLY cm
d:Ge/subComponent1/HLZ       = 5.0 cm
d:Ge/subComponent1/TransX     = 0 cm
d:Ge/subComponent1/TransY     = 0 cm
d:Ge/subComponent1/TransZ     = -5.0 cm
d:Ge/subComponent1/RotX       = 0 deg
d:Ge/subComponent1/RotY       = 0 deg
d:Ge/subComponent1/RotZ       = 0 deg
b:Ge/subComponent1/IsParallel = "true"

```

(continues on next page)

(continued from previous page)

```

s:Ge/subComponent2/Type          = "TsBox"
s:Ge/subComponent2/Parent        = "VrtParallelWorld"
d:Ge/subComponent2/HLX           = 0.9 * Ge/VrtParallelWorld/HLX cm
d:Ge/subComponent2/HLY           = 0.9 * Ge/VrtParallelWorld/HLX cm
d:Ge/subComponent2/HLZ           = 35.0 cm
d:Ge/subComponent2/TransX        = 0 cm
d:Ge/subComponent2/TransY        = 0 cm
d:Ge/subComponent2/TransZTmp     = Ge/subComponent1/TransZ + Ge/subComponent1/HLZ cm
d:Ge/subComponent2/TransZ        = Ge/subComponent2/TransZTmp + Ge/subComponent2/HLZ cm
d:Ge/subComponent2/RotX          = 0 deg
d:Ge/subComponent2/RotY          = 0 deg
d:Ge/subComponent2/RotZ          = 0 deg
b:Ge/subComponent2/IsParallel    = "true"

#####
# Variance reduction
#####
b:Vr/UseVarianceReduction        = "true"
b:Vr/ParticleSplit/Active        = "true"
sv:Vr/ParticleSplit/ParticleName = 1 "proton"
s:Vr/ParticleSplit/Component      = "VrtParallelWorld"
sv:Vr/ParticleSplit/SubComponents = 2 "subComponent1" "subComponent2"
s:Vr/ParticleSplit/Type          = "GeometricalParticleSplit"
iv:Vr/ParticleSplit/SplitNumber   = 2 4 4
bv:Vr/ParticleSplit/Symmetric     = 2 "false" "true"
d:Vr/ParticleSplit/RussianRoulette/ROIRadius = 2.5 cm
d:Vr/ParticleSplit/RussianRoulette/ROITrans  = 20 cm
bv:Vr/ParticleSplit/RussianRoulette         = 2 "true" "true"
s:Vr/ParticleSplit/SplitAxis               = "zaxis"

#####
# Beam settings
#####
s:So/Example/Type          = "Beam"
s:So/Example/Component     = "BeamPosition"
s:So/Example/BeamParticle  = "proton"
d:So/Example/BeamEnergy    = 170 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX    = 0.15 cm
d:So/Example/BeamPositionCutoffY    = 0.15 cm
d:So/Example/BeamPositionSpreadX    = 0.65 cm
d:So/Example/BeamPositionSpreadY    = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX     = 90. deg
d:So/Example/BeamAngularCutoffY     = 90. deg
d:So/Example/BeamAngularSpreadX     = 0.0032 rad
d:So/Example/BeamAngularSpreadY     = 0.0032 rad

s:Ge/BeamPosition/Parent = "Nozzle"
d:Ge/BeamPosition/RotX   = 0 deg
d:Ge/BeamPosition/TransZ = -60. cm

#####
# Scorer
#####

```

(continues on next page)



(continued from previous page)

```

s:Sc/scorer/Quantity           = "DoseToMedium"
s:Sc/scorer/Component          = "Phantom"
sv:Sc/scorer/Report            = 3 "Sum" "Mean" "Variance"
s:Sc/scorer/OutputType         = "CSV"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"

#####
# Histories
#####
i:So/Example/NumberOfHistoriesInRun = 100
i:Ts/ShowHistoryCountAtInterval     = 100
b:Ts/PauseBeforeQuit                = "False"

#####
# Visualization
#####
s:Gr/viewer/Type                = "OpenGL"
u:Gr/viewer/Zoom                 = 0.5
i:Gr/viewer/WindowSizeX         = 900
i:Gr/viewer/WindowSizeY         = 900
i:Gr/viewer/WindowPosX          = 0
i:Gr/viewer/WindowPosY          = 0
s:Gr/viewer/Projection           = "Orthogonal"
d:Gr/viewer/Theta                = -90. deg
d:Gr/viewer/Phi                  = 0. deg
b:Gr/viewer/IncludeAxes          = "true"
s:Gr/viewer/AxesComponent        = "World"
sv:Gr/OnlyIncludeParticlesNamed = 1 "proton"
d:Gr/viewer/AxesSize             = 20 cm

```

## 30.5 ImportanceSampling.txt

```

#####
# Simple shielding example
#####
d:Ge/World/HLX = 2 m
d:Ge/World/HLY = 2 m
d:Ge/World/HLZ = 2.1 m

s:Ge/wall/Type      = "TsBox"
s:Ge/wall/Parent    = "World"
s:Ge/wall/Material  = "G4_CONCRETE"
d:Ge/wall/HLX       = 1 m
d:Ge/wall/HLY       = 1 m
d:Ge/wall/HLZ       = 1 m
d:Ge/wall/TransX    = 0 m
d:Ge/wall/TransY    = 0 m
d:Ge/wall/TransZ    = 0 m
d:Ge/wall/RotX      = 0 deg
d:Ge/wall/RotY      = 0 deg
d:Ge/wall/RotZ      = 0 deg
s:Ge/wall/Color      = "red"
i:Ge/wall/XBins      = 1
i:Ge/wall/YBins      = 1

```

(continues on next page)

(continued from previous page)

```

i:Ge/wall/ZBins      = 10

#####
# Gemetry for vrt
#####
d:Ge/subComponent/Thickness      = 20 cm
d:Ge/subComponent/StartPosition = -1.0 * Ge/wall/HLZ cm

s:Ge/VrtParallelWorld/Type      = "TsBox"
s:Ge/VrtParallelWorld/Parent    = "World"
d:Ge/VrtParallelWorld/HLX       = 1.1 m
d:Ge/VrtParallelWorld/HLY       = 1.1 m
d:Ge/VrtParallelWorld/HLZ       = 1.1 m
d:Ge/VrtParallelWorld/TransX    = 0 cm
d:Ge/VrtParallelWorld/TransY    = 0 cm
d:Ge/VrtParallelWorld/TransZ    = 0 cm
d:Ge/VrtParallelWorld/RotX      = 0 deg
d:Ge/VrtParallelWorld/RotY      = 0 deg
d:Ge/VrtParallelWorld/RotZ      = 0 deg
b:Ge/VrtParallelWorld/IsParallel = "true"

s:Ge/subComponent1/Type      = "TsBox"
s:Ge/subComponent1/Parent    = "VrtParallelWorld"
d:Ge/subComponent1/HLX       = Ge/Wall/HLX m
d:Ge/subComponent1/HLY       = Ge/Wall/HLY m
d:Ge/subComponent1/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent1/TransX    = 0 cm
d:Ge/subComponent1/TransY    = 0 cm
d:Ge/subComponent1/TransZ    = Ge/subComponent/StartPosition + Ge/subComponent1/HLZ_
→cm
d:Ge/subComponent1/RotX      = 0 deg
d:Ge/subComponent1/RotY      = 0 deg
d:Ge/subComponent1/RotZ      = 0 deg
b:Ge/subComponent1/IsParallel = "true"

s:Ge/subComponent2/Type      = "TsBox"
s:Ge/subComponent2/Parent    = "VrtParallelWorld"
d:Ge/subComponent2/HLX       = Ge/Wall/HLX m
d:Ge/subComponent2/HLY       = Ge/Wall/HLY m
d:Ge/subComponent2/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent2/TransX    = 0 cm
d:Ge/subComponent2/TransY    = 0 cm
d:Ge/subComponent2/TransZTmp = Ge/subComponent1/TransZ + Ge/subComponent1/HLZ cm
d:Ge/subComponent2/TransZ    = Ge/subComponent2/TransZTmp + Ge/subComponent2/HLZ cm
d:Ge/subComponent2/RotX      = 0 deg
d:Ge/subComponent2/RotY      = 0 deg
d:Ge/subComponent2/RotZ      = 0 deg
b:Ge/subComponent2/IsParallel = "true"

s:Ge/subComponent3/Type      = "TsBox"
s:Ge/subComponent3/Parent    = "VrtParallelWorld"
d:Ge/subComponent3/HLX       = Ge/Wall/HLX m
d:Ge/subComponent3/HLY       = Ge/Wall/HLY m
d:Ge/subComponent3/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent3/TransX    = 0 cm
d:Ge/subComponent3/TransY    = 0 cm
d:Ge/subComponent3/TransZTmp = Ge/subComponent2/TransZ + Ge/subComponent2/HLZ cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/subComponent3/TransZ      = Ge/subComponent3/TransZTmp + Ge/subComponent3/HLZ cm
d:Ge/subComponent3/RotX       = 0 deg
d:Ge/subComponent3/RotY       = 0 deg
d:Ge/subComponent3/RotZ       = 0 deg
b:Ge/subComponent3/IsParallel  = "true"

s:Ge/subComponent4/Type       = "TsBox"
s:Ge/subComponent4/Parent     = "VrtParallelWorld"
d:Ge/subComponent4/HLX       = Ge/Wall/HLX m
d:Ge/subComponent4/HLY       = Ge/Wall/HLY m
d:Ge/subComponent4/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent4/TransX     = 0 cm
d:Ge/subComponent4/TransY     = 0 cm
d:Ge/subComponent4/TransZTmp  = Ge/subComponent3/TransZ + Ge/subComponent3/HLZ cm
d:Ge/subComponent4/TransZ     = Ge/subComponent4/TransZTmp + Ge/subComponent4/HLZ cm
d:Ge/subComponent4/RotX       = 0 deg
d:Ge/subComponent4/RotY       = 0 deg
d:Ge/subComponent4/RotZ       = 0 deg
b:Ge/subComponent4/IsParallel  = "true"

s:Ge/subComponent5/Type       = "TsBox"
s:Ge/subComponent5/Parent     = "VrtParallelWorld"
d:Ge/subComponent5/HLX       = Ge/Wall/HLX m
d:Ge/subComponent5/HLY       = Ge/Wall/HLY m
d:Ge/subComponent5/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent5/TransX     = 0 cm
d:Ge/subComponent5/TransY     = 0 cm
d:Ge/subComponent5/TransZTmp  = Ge/subComponent4/TransZ + Ge/subComponent4/HLZ cm
d:Ge/subComponent5/TransZ     = Ge/subComponent5/TransZTmp + Ge/subComponent5/HLZ cm
d:Ge/subComponent5/RotX       = 0 deg
d:Ge/subComponent5/RotY       = 0 deg
d:Ge/subComponent5/RotZ       = 0 deg
b:Ge/subComponent5/IsParallel  = "true"

s:Ge/subComponent6/Type       = "TsBox"
s:Ge/subComponent6/Parent     = "VrtParallelWorld"
d:Ge/subComponent6/HLX       = Ge/Wall/HLX m
d:Ge/subComponent6/HLY       = Ge/Wall/HLY m
d:Ge/subComponent6/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent6/TransX     = 0 cm
d:Ge/subComponent6/TransY     = 0 cm
d:Ge/subComponent6/TransZTmp  = Ge/subComponent5/TransZ + Ge/subComponent5/HLZ cm
d:Ge/subComponent6/TransZ     = Ge/subComponent6/TransZTmp + Ge/subComponent6/HLZ cm
d:Ge/subComponent6/RotX       = 0 deg
d:Ge/subComponent6/RotY       = 0 deg
d:Ge/subComponent6/RotZ       = 0 deg
b:Ge/subComponent6/IsParallel  = "true"

s:Ge/subComponent7/Type       = "TsBox"
s:Ge/subComponent7/Parent     = "VrtParallelWorld"
d:Ge/subComponent7/HLX       = Ge/Wall/HLX m
d:Ge/subComponent7/HLY       = Ge/Wall/HLY m
d:Ge/subComponent7/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent7/TransX     = 0 cm
d:Ge/subComponent7/TransY     = 0 cm
d:Ge/subComponent7/TransZTmp  = Ge/subComponent6/TransZ + Ge/subComponent6/HLZ cm
d:Ge/subComponent7/TransZ     = Ge/subComponent7/TransZTmp + Ge/subComponent7/HLZ cm

```

(continues on next page)

(continued from previous page)

```

d:Ge/subComponent7/RotX      = 0 deg
d:Ge/subComponent7/RotY      = 0 deg
d:Ge/subComponent7/RotZ      = 0 deg
b:Ge/subComponent7/IsParallel = "true"

s:Ge/subComponent8/Type      = "TsBox"
s:Ge/subComponent8/Parent    = "VrtParallelWorld"
d:Ge/subComponent8/HLX       = Ge/Wall/HLX m
d:Ge/subComponent8/HLY       = Ge/Wall/HLY m
d:Ge/subComponent8/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent8/TransX    = 0 cm
d:Ge/subComponent8/TransY    = 0 cm
d:Ge/subComponent8/TransZTmp = Ge/subComponent7/TransZ + Ge/subComponent7/HLZ cm
d:Ge/subComponent8/TransZ    = Ge/subComponent8/TransZTmp + Ge/subComponent8/HLZ cm
d:Ge/subComponent8/RotX      = 0 deg
d:Ge/subComponent8/RotY      = 0 deg
d:Ge/subComponent8/RotZ      = 0 deg
b:Ge/subComponent8/IsParallel = "true"

s:Ge/subComponent9/Type      = "TsBox"
s:Ge/subComponent9/Parent    = "VrtParallelWorld"
d:Ge/subComponent9/HLX       = Ge/Wall/HLX m
d:Ge/subComponent9/HLY       = Ge/Wall/HLY m
d:Ge/subComponent9/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent9/TransX    = 0 cm
d:Ge/subComponent9/TransY    = 0 cm
d:Ge/subComponent9/TransZTmp = Ge/subComponent8/TransZ + Ge/subComponent8/HLZ cm
d:Ge/subComponent9/TransZ    = Ge/subComponent9/TransZTmp + Ge/subComponent9/HLZ cm
d:Ge/subComponent9/RotX      = 0 deg
d:Ge/subComponent9/RotY      = 0 deg
d:Ge/subComponent9/RotZ      = 0 deg
b:Ge/subComponent9/IsParallel = "true"

s:Ge/subComponent10/Type     = "TsBox"
s:Ge/subComponent10/Parent   = "VrtParallelWorld"
d:Ge/subComponent10/HLX      = Ge/Wall/HLX m
d:Ge/subComponent10/HLY      = Ge/Wall/HLY m
d:Ge/subComponent10/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent10/TransX    = 0 cm
d:Ge/subComponent10/TransY    = 0 cm
d:Ge/subComponent10/TransZTmp = Ge/subComponent9/TransZ + Ge/subComponent9/HLZ cm
d:Ge/subComponent10/TransZ    = Ge/subComponent10/TransZTmp + Ge/subComponent10/HLZ_
→cm
d:Ge/subComponent10/RotX      = 0 deg
d:Ge/subComponent10/RotY      = 0 deg
d:Ge/subComponent10/RotZ      = 0 deg
b:Ge/subComponent10/IsParallel = "true"

#####
# VRT
#####
b:Vr/UseVarianceReduction    = "true"
b:Vr/ParticleSplit/Active    = "true"
sv:Vr/ParticleSplit/ParticleName = 1 "neutron"
s:Vr/ParticleSplit/Component  = "VrtParallelWorld"
sv:Vr/ParticleSplit/SubComponents = 10
"subComponent1" "subComponent2"

```

(continues on next page)

(continued from previous page)

```

"subComponent3" "subComponent4"
"subComponent5" "subComponent6"
"subComponent7" "subComponent8"
"subComponent9" "subComponent10"
s:Vr/ParticleSplit/Type = "ImportanceSampling"
uv:Vr/ParticleSplit/ImportanceValues = 10 2 4 8 16 32 64 128 256 512 1024

#####
# Scorer
#####
s:Sc/scorer/Quantity = "DoseToMedium"
s:Sc/scorer/Component = "Wall"
s:Sc/scorer/OutputFile = "ImportanceSampling"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/scorer/Report = 3 "Sum" "Mean" "Variance"

#####
# Beam setting
#####
s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "neutron"
d:So/Example/BeamEnergy = 10 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

d:Ge/BeamPosition/TransZ = -1.0 * Ge/World/HLZ m
d:Ge/BeamPosition/RotX = 0 deg

i:Ts/ShowHistoryCountAtInterval = 10

b:Ts/PauseBeforeQuit = "True"
s:Gr/view/Type = "OpenGL"
b:Gr/view/IncludeAxes = "True"

```

## 30.6 ImportanceSamplingMassGeometry.txt

```

#####
# Simple shielding example
#####
d:Ge/World/HLX = 2 m
d:Ge/World/HLY = 2 m
d:Ge/World/HLZ = 2.1 m

```

(continues on next page)

(continued from previous page)

```
#####
# Scoring region
#####
s:Ge/Scorer/Type      = "TsBox"
s:Ge/Scorer/Parent    = "World"
b:Ge/Scorer/IsParallel = "True"
d:Ge/Scorer/HLX       = 1 m
d:Ge/Scorer/HLY       = 1 m
d:Ge/Scorer/HLZ       = 1 m
d:Ge/Scorer/TransX    = 0 m
d:Ge/Scorer/TransY    = 0 m
d:Ge/Scorer/TransZ    = 0 m
d:Ge/Scorer/RotX      = 0 deg
d:Ge/Scorer/RotY      = 0 deg
d:Ge/Scorer/RotZ      = 0 deg
s:Ge/Scorer/Color     = "red"
i:Ge/Scorer/XBins     = 1
i:Ge/Scorer/YBins     = 1
i:Ge/Scorer/ZBins     = 10

#####
# Gemetry for vrt
#####
d:Ge/subComponent/Thickness = 20 cm
d:Ge/subComponent/StartPosition = -1.0 * Ge/Scorer/HLZ cm

s:Ge/subComponent1/Material = "G4_CONCRETE"
s:Ge/subComponent2/Material = "G4_CONCRETE"
s:Ge/subComponent3/Material = "G4_CONCRETE"
s:Ge/subComponent4/Material = "G4_CONCRETE"
s:Ge/subComponent5/Material = "G4_CONCRETE"
s:Ge/subComponent6/Material = "G4_CONCRETE"
s:Ge/subComponent7/Material = "G4_CONCRETE"
s:Ge/subComponent8/Material = "G4_CONCRETE"
s:Ge/subComponent9/Material = "G4_CONCRETE"
s:Ge/subComponent10/Material = "G4_CONCRETE"

s:Ge/subComponent1/Type      = "TsBox"
s:Ge/subComponent1/Parent    = "World"
d:Ge/subComponent1/HLX      = Ge/Scorer/HLX m
d:Ge/subComponent1/HLY      = Ge/Scorer/HLY m
d:Ge/subComponent1/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent1/TransX    = 0 cm
d:Ge/subComponent1/TransY    = 0 cm
d:Ge/subComponent1/TransZ    = Ge/subComponent/StartPosition + Ge/subComponent1/HLZ cm
d:Ge/subComponent1/RotX     = 0 deg
d:Ge/subComponent1/RotY     = 0 deg
d:Ge/subComponent1/RotZ     = 0 deg

s:Ge/subComponent2/Type      = "TsBox"
s:Ge/subComponent2/Parent    = "World"
d:Ge/subComponent2/HLX      = Ge/Scorer/HLX m
d:Ge/subComponent2/HLY      = Ge/Scorer/HLY m
d:Ge/subComponent2/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent2/TransX    = 0 cm
d:Ge/subComponent2/TransY    = 0 cm
```

(continues on next page)

(continued from previous page)

```

d:Ge/subComponent2/TransZTmp = Ge/subComponent1/TransZ + Ge/subComponent1/HLZ cm
d:Ge/subComponent2/TransZ    = Ge/subComponent2/TransZTmp + Ge/subComponent2/HLZ cm
d:Ge/subComponent2/RotX      = 0 deg
d:Ge/subComponent2/RotY      = 0 deg
d:Ge/subComponent2/RotZ      = 0 deg

s:Ge/subComponent3/Type      = "TsBox"
s:Ge/subComponent3/Parent    = "World"
d:Ge/subComponent3/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent3/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent3/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent3/TransX     = 0 cm
d:Ge/subComponent3/TransY     = 0 cm
d:Ge/subComponent3/TransZTmp = Ge/subComponent2/TransZ + Ge/subComponent2/HLZ cm
d:Ge/subComponent3/TransZ     = Ge/subComponent3/TransZTmp + Ge/subComponent3/HLZ cm
d:Ge/subComponent3/RotX      = 0 deg
d:Ge/subComponent3/RotY      = 0 deg
d:Ge/subComponent3/RotZ      = 0 deg

s:Ge/subComponent4/Type      = "TsBox"
s:Ge/subComponent4/Parent    = "World"
d:Ge/subComponent4/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent4/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent4/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent4/TransX     = 0 cm
d:Ge/subComponent4/TransY     = 0 cm
d:Ge/subComponent4/TransZTmp = Ge/subComponent3/TransZ + Ge/subComponent3/HLZ cm
d:Ge/subComponent4/TransZ     = Ge/subComponent4/TransZTmp + Ge/subComponent4/HLZ cm
d:Ge/subComponent4/RotX      = 0 deg
d:Ge/subComponent4/RotY      = 0 deg
d:Ge/subComponent4/RotZ      = 0 deg

s:Ge/subComponent5/Type      = "TsBox"
s:Ge/subComponent5/Parent    = "World"
d:Ge/subComponent5/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent5/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent5/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent5/TransX     = 0 cm
d:Ge/subComponent5/TransY     = 0 cm
d:Ge/subComponent5/TransZTmp = Ge/subComponent4/TransZ + Ge/subComponent4/HLZ cm
d:Ge/subComponent5/TransZ     = Ge/subComponent5/TransZTmp + Ge/subComponent5/HLZ cm
d:Ge/subComponent5/RotX      = 0 deg
d:Ge/subComponent5/RotY      = 0 deg
d:Ge/subComponent5/RotZ      = 0 deg

s:Ge/subComponent6/Type      = "TsBox"
s:Ge/subComponent6/Parent    = "World"
d:Ge/subComponent6/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent6/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent6/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent6/TransX     = 0 cm
d:Ge/subComponent6/TransY     = 0 cm
d:Ge/subComponent6/TransZTmp = Ge/subComponent5/TransZ + Ge/subComponent5/HLZ cm
d:Ge/subComponent6/TransZ     = Ge/subComponent6/TransZTmp + Ge/subComponent6/HLZ cm
d:Ge/subComponent6/RotX      = 0 deg
d:Ge/subComponent6/RotY      = 0 deg
d:Ge/subComponent6/RotZ      = 0 deg

```

(continues on next page)

(continued from previous page)

```

s:Ge/subComponent7/Type      = "TsBox"
s:Ge/subComponent7/Parent    = "World"
d:Ge/subComponent7/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent7/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent7/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent7/TransX     = 0 cm
d:Ge/subComponent7/TransY     = 0 cm
d:Ge/subComponent7/TransZTmp  = Ge/subComponent6/TransZ + Ge/subComponent6/HLZ cm
d:Ge/subComponent7/TransZ     = Ge/subComponent7/TransZTmp + Ge/subComponent7/HLZ cm
d:Ge/subComponent7/RotX       = 0 deg
d:Ge/subComponent7/RotY       = 0 deg
d:Ge/subComponent7/RotZ       = 0 deg

s:Ge/subComponent8/Type      = "TsBox"
s:Ge/subComponent8/Parent    = "World"
d:Ge/subComponent8/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent8/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent8/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent8/TransX     = 0 cm
d:Ge/subComponent8/TransY     = 0 cm
d:Ge/subComponent8/TransZTmp  = Ge/subComponent7/TransZ + Ge/subComponent7/HLZ cm
d:Ge/subComponent8/TransZ     = Ge/subComponent8/TransZTmp + Ge/subComponent8/HLZ cm
d:Ge/subComponent8/RotX       = 0 deg
d:Ge/subComponent8/RotY       = 0 deg
d:Ge/subComponent8/RotZ       = 0 deg

s:Ge/subComponent9/Type      = "TsBox"
s:Ge/subComponent9/Parent    = "World"
d:Ge/subComponent9/HLX       = Ge/Scorer/HLX m
d:Ge/subComponent9/HLY       = Ge/Scorer/HLY m
d:Ge/subComponent9/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent9/TransX     = 0 cm
d:Ge/subComponent9/TransY     = 0 cm
d:Ge/subComponent9/TransZTmp  = Ge/subComponent8/TransZ + Ge/subComponent8/HLZ cm
d:Ge/subComponent9/TransZ     = Ge/subComponent9/TransZTmp + Ge/subComponent9/HLZ cm
d:Ge/subComponent9/RotX       = 0 deg
d:Ge/subComponent9/RotY       = 0 deg
d:Ge/subComponent9/RotZ       = 0 deg

s:Ge/subComponent10/Type     = "TsBox"
s:Ge/subComponent10/Parent   = "World"
d:Ge/subComponent10/HLX      = Ge/Scorer/HLX m
d:Ge/subComponent10/HLY      = Ge/Scorer/HLY m
d:Ge/subComponent10/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent10/TransX    = 0 cm
d:Ge/subComponent10/TransY    = 0 cm
d:Ge/subComponent10/TransZTmp = Ge/subComponent9/TransZ + Ge/subComponent9/HLZ cm
d:Ge/subComponent10/TransZ    = Ge/subComponent10/TransZTmp + Ge/subComponent10/HLZ cm
d:Ge/subComponent10/RotX      = 0 deg
d:Ge/subComponent10/RotY      = 0 deg
d:Ge/subComponent10/RotZ      = 0 deg

#####
# VRT
#####
b:Vr/UseVarianceReduction    = "true"

```

(continues on next page)



(continued from previous page)

```

b:Vr/ParticleSplit/Active           = "true"
sv:Vr/ParticleSplit/ParticleName    = 4 "neutron" "gamma" "proton" "e-"
s:Vr/ParticleSplit/Component         = "subComponent1"
sv:Vr/ParticleSplit/SubComponents   = 9
  "subComponent2"
  "subComponent3" "subComponent4"
  "subComponent5" "subComponent6"
  "subComponent7" "subComponent8"
  "subComponent9" "subComponent10"
s:Vr/ParticleSplit/Type = "ImportanceSampling"
uv:Vr/ParticleSplit/ImportanceValues = 9 2 4 8 16 32 64 128 256 512

#####
# Scorer
#####
s:Sc/scorer/Quantity                = "DoseToMedium"
s:Sc/scorer/Component               = "Scorer"
s:Sc/scorer/OutputFile              = "ImportanceSamplingAllMassNew"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/scorer/Report                 = 2 "Mean" "Standard_Deviation"

#####
# Beam setting
#####
s:So/Example/Type                   = "Beam"
s:So/Example/Component              = "BeamPosition"
s:So/Example/BeamParticle           = "neutron"
d:So/Example/BeamEnergy              = 10 MeV
u:So/Example/BeamEnergySpread        = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX     = 10. cm
d:So/Example/BeamPositionCutoffY     = 10. cm
d:So/Example/BeamPositionSpreadX     = 0.65 cm
d:So/Example/BeamPositionSpreadY     = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX      = 90. deg
d:So/Example/BeamAngularCutoffY      = 90. deg
d:So/Example/BeamAngularSpreadX      = 0.0032 rad
d:So/Example/BeamAngularSpreadY      = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun  = 100

d:Ge/BeamPosition/TransZ = -1.0 * Ge/World/HLZ m
d:Ge/BeamPosition/RotX   = 0 deg

i:Ts/ShowHistoryCountAtInterval = 10

b:Ts/PauseBeforeQuit = "True"
s:Gr/view/Type        = "OpenGL"
b:Gr/view/IncludeAxes = "True"

```

## 30.7 ImportanceSamplingSecondaryBiasing.txt

```
#
# This example demonstrate the use of two variance reduction
# techniques in the same simulation: uniform bremsstrahlung splitting
# and importance sampling.
# Bremsstrahlung x-rays produced by e- hitting a lead target are split
# and further split in their propagation through a parallel geometry
# of stacked slabs.
# Due to the high number of tracks, only the e- shower is shown in the
# OpenGL viewer.
#
#####
# Simple shielding example
#####
d:Ge/World/HLX = 4 m
d:Ge/World/HLY = 4 m
d:Ge/World/HLZ = 4.1 m
b:Ge/World/Invisible = "True"
s:Ge/World/Material = "G4_WATER"

s:Ge/group/Parent = "World"
s:Ge/group/Type = "Group"

s:Ge/target/Type = "TsBox"
s:Ge/target/Parent = "group"
s:Ge/target/Material = "G4_Pb"
s:Ge/target/Color = "magenta"
s:Ge/target/DrawingStyle = "Solid"
d:Ge/target/HLX = 1 cm
d:Ge/target/HLY = 1 cm
d:Ge/target/HLZ = 2 mm
d:Ge/target/TransX = 0 m
d:Ge/target/TransY = 0 m
d:Ge/target/TransZ = 0 m
d:Ge/target/RotX = 0 deg
d:Ge/target/RotY = 0 deg
d:Ge/target/RotZ = 0 deg

s:Ge/target/AssignToRegionNamed = "targetregion"

#####
# Gemetry for vrt
#####
d:Ge/subComponent/Thickness = 2 mm
d:Ge/subComponent/StartPosition = -1.0 * Ge/VrtParallelWorld/HLZ cm

s:Ge/VrtParallelWorld/Type = "TsBox"
s:Ge/VrtParallelWorld/Parent = "group"
d:Ge/VrtParallelWorld/HLX = 21 cm
d:Ge/VrtParallelWorld/HLY = 21 cm
d:Ge/VrtParallelWorld/HLZ = 6 cm
d:Ge/VrtParallelWorld/TransX = 0 cm
d:Ge/VrtParallelWorld/TransY = 0 cm
d:Ge/VrtParallelWorld/TransZ = Ge/target/HLZ + Ge/VrtParallelWorld/HLZ cm
d:Ge/VrtParallelWorld/RotX = 0 deg
d:Ge/VrtParallelWorld/RotY = 0 deg
```

(continues on next page)

(continued from previous page)

```

d:Ge/VrtParallelWorld/RotZ      = 0 deg
b:Ge/VrtParallelWorld/IsParallel = "true"

s:Ge/subComponent1/Type        = "TsBox"
s:Ge/subComponent1/Parent      = "VrtParallelWorld"
d:Ge/subComponent1/HLX         = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent1/HLY         = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent1/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent1/TransX      = 0 cm
d:Ge/subComponent1/TransY      = 0 cm
d:Ge/subComponent1/TransZ      = Ge/subComponent/StartPosition + Ge/subComponent1/HLZ_
↪cm
d:Ge/subComponent1/RotX        = 0 deg
d:Ge/subComponent1/RotY        = 0 deg
d:Ge/subComponent1/RotZ        = 0 deg
b:Ge/subComponent1/IsParallel  = "true"

s:Ge/subComponent2/Type        = "TsBox"
s:Ge/subComponent2/Parent      = "VrtParallelWorld"
d:Ge/subComponent2/HLX         = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent2/HLY         = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent2/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent2/TransX      = 0 cm
d:Ge/subComponent2/TransY      = 0 cm
d:Ge/subComponent2/TransZTmp   = Ge/subComponent1/TransZ + Ge/subComponent1/HLZ cm
d:Ge/subComponent2/TransZ      = Ge/subComponent2/TransZTmp + Ge/subComponent2/HLZ cm
d:Ge/subComponent2/RotX        = 0 deg
d:Ge/subComponent2/RotY        = 0 deg
d:Ge/subComponent2/RotZ        = 0 deg
b:Ge/subComponent2/IsParallel  = "true"

s:Ge/subComponent3/Type        = "TsBox"
s:Ge/subComponent3/Parent      = "VrtParallelWorld"
d:Ge/subComponent3/HLX         = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent3/HLY         = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent3/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent3/TransX      = 0 cm
d:Ge/subComponent3/TransY      = 0 cm
d:Ge/subComponent3/TransZTmp   = Ge/subComponent2/TransZ + Ge/subComponent2/HLZ cm
d:Ge/subComponent3/TransZ      = Ge/subComponent3/TransZTmp + Ge/subComponent3/HLZ cm
d:Ge/subComponent3/RotX        = 0 deg
d:Ge/subComponent3/RotY        = 0 deg
d:Ge/subComponent3/RotZ        = 0 deg
b:Ge/subComponent3/IsParallel  = "true"

s:Ge/subComponent4/Type        = "TsBox"
s:Ge/subComponent4/Parent      = "VrtParallelWorld"
d:Ge/subComponent4/HLX         = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent4/HLY         = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent4/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent4/TransX      = 0 cm
d:Ge/subComponent4/TransY      = 0 cm
d:Ge/subComponent4/TransZTmp   = Ge/subComponent3/TransZ + Ge/subComponent3/HLZ cm
d:Ge/subComponent4/TransZ      = Ge/subComponent4/TransZTmp + Ge/subComponent4/HLZ cm
d:Ge/subComponent4/RotX        = 0 deg
d:Ge/subComponent4/RotY        = 0 deg
d:Ge/subComponent4/RotZ        = 0 deg

```

(continues on next page)

(continued from previous page)

```

b:Ge/subComponent4/IsParallel = "true"

s:Ge/subComponent5/Type      = "TsBox"
s:Ge/subComponent5/Parent    = "VrtParallelWorld"
d:Ge/subComponent5/HLX       = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent5/HLY       = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent5/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent5/TransX    = 0 cm
d:Ge/subComponent5/TransY    = 0 cm
d:Ge/subComponent5/TransZTmp = Ge/subComponent4/TransZ + Ge/subComponent4/HLZ cm
d:Ge/subComponent5/TransZ    = Ge/subComponent5/TransZTmp + Ge/subComponent5/HLZ cm
d:Ge/subComponent5/RotX      = 0 deg
d:Ge/subComponent5/RotY      = 0 deg
d:Ge/subComponent5/RotZ      = 0 deg
b:Ge/subComponent5/IsParallel = "true"

s:Ge/subComponent6/Type      = "TsBox"
s:Ge/subComponent6/Parent    = "VrtParallelWorld"
d:Ge/subComponent6/HLX       = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent6/HLY       = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent6/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent6/TransX    = 0 cm
d:Ge/subComponent6/TransY    = 0 cm
d:Ge/subComponent6/TransZTmp = Ge/subComponent5/TransZ + Ge/subComponent5/HLZ cm
d:Ge/subComponent6/TransZ    = Ge/subComponent6/TransZTmp + Ge/subComponent6/HLZ cm
d:Ge/subComponent6/RotX      = 0 deg
d:Ge/subComponent6/RotY      = 0 deg
d:Ge/subComponent6/RotZ      = 0 deg
b:Ge/subComponent6/IsParallel = "true"

s:Ge/subComponent7/Type      = "TsBox"
s:Ge/subComponent7/Parent    = "VrtParallelWorld"
d:Ge/subComponent7/HLX       = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent7/HLY       = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent7/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent7/TransX    = 0 cm
d:Ge/subComponent7/TransY    = 0 cm
d:Ge/subComponent7/TransZTmp = Ge/subComponent6/TransZ + Ge/subComponent6/HLZ cm
d:Ge/subComponent7/TransZ    = Ge/subComponent7/TransZTmp + Ge/subComponent7/HLZ cm
d:Ge/subComponent7/RotX      = 0 deg
d:Ge/subComponent7/RotY      = 0 deg
d:Ge/subComponent7/RotZ      = 0 deg
b:Ge/subComponent7/IsParallel = "true"

s:Ge/subComponent8/Type      = "TsBox"
s:Ge/subComponent8/Parent    = "VrtParallelWorld"
d:Ge/subComponent8/HLX       = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent8/HLY       = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent8/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent8/TransX    = 0 cm
d:Ge/subComponent8/TransY    = 0 cm
d:Ge/subComponent8/TransZTmp = Ge/subComponent7/TransZ + Ge/subComponent7/HLZ cm
d:Ge/subComponent8/TransZ    = Ge/subComponent8/TransZTmp + Ge/subComponent8/HLZ cm
d:Ge/subComponent8/RotX      = 0 deg
d:Ge/subComponent8/RotY      = 0 deg
d:Ge/subComponent8/RotZ      = 0 deg
b:Ge/subComponent8/IsParallel = "true"

```

(continues on next page)

(continued from previous page)

```

s:Ge/subComponent9/Type           = "TsBox"
s:Ge/subComponent9/Parent         = "VrtParallelWorld"
d:Ge/subComponent9/HLX            = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent9/HLY            = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent9/HLZ            = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent9/TransX         = 0 cm
d:Ge/subComponent9/TransY         = 0 cm
d:Ge/subComponent9/TransZTmp      = Ge/subComponent8/TransZ + Ge/subComponent8/HLZ cm
d:Ge/subComponent9/TransZ         = Ge/subComponent9/TransZTmp + Ge/subComponent9/HLZ cm
d:Ge/subComponent9/RotX           = 0 deg
d:Ge/subComponent9/RotY           = 0 deg
d:Ge/subComponent9/RotZ           = 0 deg
b:Ge/subComponent9/IsParallel     = "true"

s:Ge/subComponent10/Type          = "TsBox"
s:Ge/subComponent10/Parent        = "VrtParallelWorld"
d:Ge/subComponent10/HLX           = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent10/HLY           = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent10/HLZ           = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent10/TransX        = 0 cm
d:Ge/subComponent10/TransY        = 0 cm
d:Ge/subComponent10/TransZTmp     = Ge/subComponent9/TransZ + Ge/subComponent9/HLZ cm
d:Ge/subComponent10/TransZ        = Ge/subComponent10/TransZTmp + Ge/subComponent10/HLZ_
→cm
d:Ge/subComponent10/RotX          = 0 deg
d:Ge/subComponent10/RotY          = 0 deg
d:Ge/subComponent10/RotZ          = 0 deg
b:Ge/subComponent10/IsParallel    = "true"

s:Ge/subComponent11/Type          = "TsBox"
s:Ge/subComponent11/Parent        = "VrtParallelWorld"
d:Ge/subComponent11/HLX           = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent11/HLY           = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent11/HLZ           = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent11/TransX        = 0 cm
d:Ge/subComponent11/TransY        = 0 cm
d:Ge/subComponent11/TransZTmp     = Ge/subComponent10/TransZ + Ge/subComponent10/HLZ cm
d:Ge/subComponent11/TransZ        = Ge/subComponent11/TransZTmp + Ge/subComponent11/HLZ_
→cm
d:Ge/subComponent11/RotX          = 0 deg
d:Ge/subComponent11/RotY          = 0 deg
d:Ge/subComponent11/RotZ          = 0 deg
b:Ge/subComponent11/IsParallel    = "true"

s:Ge/subComponent12/Type          = "TsBox"
s:Ge/subComponent12/Parent        = "VrtParallelWorld"
d:Ge/subComponent12/HLX           = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent12/HLY           = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent12/HLZ           = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent12/TransX        = 0 cm
d:Ge/subComponent12/TransY        = 0 cm
d:Ge/subComponent12/TransZTmp     = Ge/subComponent11/TransZ + Ge/subComponent11/HLZ cm
d:Ge/subComponent12/TransZ        = Ge/subComponent12/TransZTmp + Ge/subComponent12/HLZ_
→cm
d:Ge/subComponent12/RotX          = 0 deg
d:Ge/subComponent12/RotY          = 0 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/subComponent12/RotZ      = 0 deg
b:Ge/subComponent12/IsParallel = "true"

s:Ge/subComponent13/Type      = "TsBox"
s:Ge/subComponent13/Parent    = "VrtParallelWorld"
d:Ge/subComponent13/HLX       = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent13/HLY       = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent13/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent13/TransX     = 0 cm
d:Ge/subComponent13/TransY     = 0 cm
d:Ge/subComponent13/TransZTmp  = Ge/subComponent12/TransZ + Ge/subComponent12/HLZ cm
d:Ge/subComponent13/TransZ     = Ge/subComponent13/TransZTmp + Ge/subComponent13/HLZ_
→cm
d:Ge/subComponent13/RotX       = 0 deg
d:Ge/subComponent13/RotY       = 0 deg
d:Ge/subComponent13/RotZ       = 0 deg
b:Ge/subComponent13/IsParallel = "true"

s:Ge/subComponent14/Type      = "TsBox"
s:Ge/subComponent14/Parent    = "VrtParallelWorld"
d:Ge/subComponent14/HLX       = Ge/VrtParallelWorld/HLX m
d:Ge/subComponent14/HLY       = Ge/VrtParallelWorld/HLY m
d:Ge/subComponent14/HLZ       = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent14/TransX     = 0 cm
d:Ge/subComponent14/TransY     = 0 cm
d:Ge/subComponent14/TransZTmp  = Ge/subComponent13/TransZ + Ge/subComponent13/HLZ cm
d:Ge/subComponent14/TransZ     = Ge/subComponent14/TransZTmp + Ge/subComponent14/HLZ_
→cm
d:Ge/subComponent14/RotX       = 0 deg
d:Ge/subComponent14/RotY       = 0 deg
d:Ge/subComponent14/RotZ       = 0 deg
b:Ge/subComponent14/IsParallel = "true"

#####
# VRT
#####
b:Vr/UseVarianceReduction      = "true"
# Importance sampling
b:Vr/ImportanceSampling/Active = "true"
sv:Vr/ImportanceSampling/ParticleName = 1 "gamma"
s:Vr/ImportanceSampling/Component = "VrtParallelWorld"
sv:Vr/ImportanceSampling/SubComponents = 14
"subComponent1" "subComponent2"
"subComponent3" "subComponent4"
"subComponent5" "subComponent6"
"subComponent7" "subComponent8"
"subComponent9" "subComponent10"
"subComponent11" "subComponent12"
"subComponent13" "subComponent14"

s:Vr/ImportanceSampling/Type = "ImportanceSampling"
uv:Vr/ImportanceSampling/ImportanceValues = 14 2 4 8 16 32 64 128 256 512 1024 2048_
→2048 2048 2048

# Uniform split in the lead target
s:Vr/Uniform/Type = "SecondaryBiasing"
sv:Vr/Uniform/forregion/targetregion/processesNamed = 1 "eBrem"

```

(continues on next page)

(continued from previous page)

```

uv:Vr/Uniform/ForRegion/targetregion/SplitNumber      = 1    100.
dv:Vr/Uniform/ForRegion/targetregion/MaximumEnergies  = 1    6    MeV

#####
# Beam setting
#####
s:So/Example/Type = "Beam"
s:So/Example/Component = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy = 6 MeV
u:So/Example/BeamEnergySpread = 0.757504
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 1

Ge/BeamPosition/Parent = "group"
d:Ge/BeamPosition/TransZ = -1.0 * Ge/target/HLZ m
d:Ge/BeamPosition/RotX = 0 deg

i:Ts/ShowHistoryCountAtInterval = 1000

b:Ts/PauseBeforeQuit = "True"
s:Gr/view/Type = "OpenGL"
u:Gr/view/Zoom = 4
sv:Gr/OnlyIncludeParticlesNamed = 1 "e-"

```

## 30.8 SecondaryBiasing.txt

```

#
i:Ts/NumberOfThreads = 2

# Secondary biasing occurs in the region
# of this component
s:Ge/target/Parent = "World"
s:Ge/target/Material = "Lead"
s:Ge/target/Type = "TsBox"
d:Ge/target/HLX = 1.5 cm
d:Ge/target/HLY = 1.5 cm
d:Ge/target/HLZ = 0.5 cm
s:Ge/target/Color = "yellow"
# Set the region
s:Ge/target/AssignToRegionName = "targetregion"

# Component for scoring

```

(continues on next page)

(continued from previous page)

```

s:Ge/scorer/Parent      = "World"
s:Ge/scorer/Material    = "G4_WATER"
s:Ge/scorer/Type        = "TsCylinder"
d:Ge/scorer/RMax        = 5 cm
d:Ge/scorer/HL          = 0.05 mm
d:Ge/scorer/SPhi        = 0 deg
d:Ge/scorer/DPhi        = 360 deg
d:Ge/scorer/TransZ      = 1 m + Ge/scorer/HL
s:Ge/scorer/Color       = "yellow"
i:Ge/scorer/RBins       = 20
i:Ge/scorer/PhiBins     = 1
i:Ge/scorer/ZBins       = 1

s:Sc/scorer/Quantity    = "EnergyFluence"
s:Sc/scorer/Component    = "scorer"
s:Sc/scorer/OutputFile  = "secondaryBiasing"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"

# VISUALIZATION
s:Gr/view/Type          = "OpenGL"
i:Gr/view/WindowSizeX   = 600
i:Gr/view/WindowSizeY   = 600
d:Gr/view/Phi           = 0 deg
d:Gr/view/Theta         = 270 deg
u:Gr/view/Zoom          = 5

b:Ts/PauseBeforeQuit    = "True"

# SOURCE
s:So/Example/Type        = "Beam"
s:So/Example/Component    = "BeamPosition"
s:So/Example/BeamParticle = "e-"
d:So/Example/BeamEnergy   = 6 MeV
u:So/Example/BeamEnergySpread = 0.1
s:So/Example/BeamPositionDistribution = "None"
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad

Ge/BeamPosition/Parent = "World"
Ge/BeamPosition/TransX = 0.0 cm
Ge/BeamPosition/TransZ = -1.0 * Ge/target/HLZ cm
Ge/BeamPosition/TransY = 0.0 mm
Ge/BeamPosition/RotX   = 0 deg

# GEOMETRY
d:Ge/World/HLX          = 16.1 cm
d:Ge/World/HLY          = 16.1 cm
d:Ge/World/HLZ          = 1.5 m
s:Ge/World/Material     = "Air"

sv:Ph/Default/Modules   = 1 "g4em-standard_opt3"

i:So/Example/NumberOfHistoriesInRun = 1000
i:Ts/ShowHistoryCountAtInterval = 100

```

(continues on next page)



(continued from previous page)

```

b:Ts/ShowCPUTime = "True"

#Secondary biasing
b:Vr/UseVarianceReduction = "True"
b:Vr/BremsstrahlungSplit/Active = "True"
s:Vr/BremsstrahlungSplit/Type = "secondarybiasing"
# The secondary particles produced in this process will
# be split 100 times if their energies are below of 6 MeV
sv:Vr/ParTicleSplit1/forregion/targetregion/processesNamed = 1 "eBrem"
uv:Vr/BremsstrahlungSplit/ForRegion/targetregion/SplitNumber = 1 100.
dv:Vr/BremsstrahlungSplit/ForRegion/targetregion/MaximumEnergies = 1 6 MeV

```

## 30.9 WeightWindow.txt

```

#####
# Simple shielding example
#####
d:Ge/World/HLX = 2 m
d:Ge/World/HLY = 2 m
d:Ge/World/HLZ = 2.1 m

s:Ge/wall/Type = "TsBox"
s:Ge/wall/Parent = "World"
s:Ge/wall/Material = "G4_CONCRETE"
d:Ge/wall/HLX = 1 m
d:Ge/wall/HLY = 1 m
d:Ge/wall/HLZ = 1 m
d:Ge/wall/TransX = 0 m
d:Ge/wall/TransY = 0 m
d:Ge/wall/TransZ = 0 m
d:Ge/wall/RotX = 0 deg
d:Ge/wall/RotY = 0 deg
d:Ge/wall/RotZ = 0 deg
s:Ge/wall/Color = "red"
i:Ge/wall/XBins = 1
i:Ge/wall/YBins = 1
i:Ge/wall/ZBins = 10

#####
# Gemetry for vrt
#####
d:Ge/subComponent/Thickness = 20 cm
d:Ge/subComponent/StartPosition = -1.0 * Ge/wall/HLZ cm

s:Ge/VrtParallelWorld/Type = "TsBox"
s:Ge/VrtParallelWorld/Parent = "World"
d:Ge/VrtParallelWorld/HLX = 1.1 m
d:Ge/VrtParallelWorld/HLY = 1.1 m
d:Ge/VrtParallelWorld/HLZ = 1.1 m
d:Ge/VrtParallelWorld/TransX = 0 cm
d:Ge/VrtParallelWorld/TransY = 0 cm
d:Ge/VrtParallelWorld/TransZ = 0 cm
d:Ge/VrtParallelWorld/RotX = 0 deg
d:Ge/VrtParallelWorld/RotY = 0 deg

```

(continues on next page)

(continued from previous page)

```

d:Ge/VrtParallelWorld/RotZ      = 0 deg
b:Ge/VrtParallelWorld/IsParallel = "true"

s:Ge/subComponent1/Type        = "TsBox"
s:Ge/subComponent1/Parent      = "VrtParallelWorld"
d:Ge/subComponent1/HLX         = Ge/Wall/HLX m
d:Ge/subComponent1/HLY         = Ge/Wall/HLY m
d:Ge/subComponent1/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent1/TransX      = 0 cm
d:Ge/subComponent1/TransY      = 0 cm
d:Ge/subComponent1/TransZ      = Ge/subComponent/StartPosition + Ge/subComponent1/HLZ_
↪cm
d:Ge/subComponent1/RotX        = 0 deg
d:Ge/subComponent1/RotY        = 0 deg
d:Ge/subComponent1/RotZ        = 0 deg
b:Ge/subComponent1/IsParallel   = "true"

s:Ge/subComponent2/Type        = "TsBox"
s:Ge/subComponent2/Parent      = "VrtParallelWorld"
d:Ge/subComponent2/HLX         = Ge/Wall/HLX m
d:Ge/subComponent2/HLY         = Ge/Wall/HLY m
d:Ge/subComponent2/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent2/TransX      = 0 cm
d:Ge/subComponent2/TransY      = 0 cm
d:Ge/subComponent2/TransZTmp   = Ge/subComponent1/TransZ + Ge/subComponent1/HLZ cm
d:Ge/subComponent2/TransZ      = Ge/subComponent2/TransZTmp + Ge/subComponent2/HLZ cm
d:Ge/subComponent2/RotX        = 0 deg
d:Ge/subComponent2/RotY        = 0 deg
d:Ge/subComponent2/RotZ        = 0 deg
b:Ge/subComponent2/IsParallel   = "true"

s:Ge/subComponent3/Type        = "TsBox"
s:Ge/subComponent3/Parent      = "VrtParallelWorld"
d:Ge/subComponent3/HLX         = Ge/Wall/HLX m
d:Ge/subComponent3/HLY         = Ge/Wall/HLY m
d:Ge/subComponent3/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent3/TransX      = 0 cm
d:Ge/subComponent3/TransY      = 0 cm
d:Ge/subComponent3/TransZTmp   = Ge/subComponent2/TransZ + Ge/subComponent2/HLZ cm
d:Ge/subComponent3/TransZ      = Ge/subComponent3/TransZTmp + Ge/subComponent3/HLZ cm
d:Ge/subComponent3/RotX        = 0 deg
d:Ge/subComponent3/RotY        = 0 deg
d:Ge/subComponent3/RotZ        = 0 deg
b:Ge/subComponent3/IsParallel   = "true"

s:Ge/subComponent4/Type        = "TsBox"
s:Ge/subComponent4/Parent      = "VrtParallelWorld"
d:Ge/subComponent4/HLX         = Ge/Wall/HLX m
d:Ge/subComponent4/HLY         = Ge/Wall/HLY m
d:Ge/subComponent4/HLZ         = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent4/TransX      = 0 cm
d:Ge/subComponent4/TransY      = 0 cm
d:Ge/subComponent4/TransZTmp   = Ge/subComponent3/TransZ + Ge/subComponent3/HLZ cm
d:Ge/subComponent4/TransZ      = Ge/subComponent4/TransZTmp + Ge/subComponent4/HLZ cm
d:Ge/subComponent4/RotX        = 0 deg
d:Ge/subComponent4/RotY        = 0 deg
d:Ge/subComponent4/RotZ        = 0 deg

```

(continues on next page)

(continued from previous page)

```

b:Ge/subComponent4/IsParallel = "true"

s:Ge/subComponent5/Type      = "TsBox"
s:Ge/subComponent5/Parent    = "VrtParallelWorld"
d:Ge/subComponent5/HLX      = Ge/Wall/HLX m
d:Ge/subComponent5/HLY      = Ge/Wall/HLY m
d:Ge/subComponent5/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent5/TransX    = 0 cm
d:Ge/subComponent5/TransY    = 0 cm
d:Ge/subComponent5/TransZTmp = Ge/subComponent4/TransZ + Ge/subComponent4/HLZ cm
d:Ge/subComponent5/TransZ    = Ge/subComponent5/TransZTmp + Ge/subComponent5/HLZ cm
d:Ge/subComponent5/RotX      = 0 deg
d:Ge/subComponent5/RotY      = 0 deg
d:Ge/subComponent5/RotZ      = 0 deg
b:Ge/subComponent5/IsParallel = "true"

s:Ge/subComponent6/Type      = "TsBox"
s:Ge/subComponent6/Parent    = "VrtParallelWorld"
d:Ge/subComponent6/HLX      = Ge/Wall/HLX m
d:Ge/subComponent6/HLY      = Ge/Wall/HLY m
d:Ge/subComponent6/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent6/TransX    = 0 cm
d:Ge/subComponent6/TransY    = 0 cm
d:Ge/subComponent6/TransZTmp = Ge/subComponent5/TransZ + Ge/subComponent5/HLZ cm
d:Ge/subComponent6/TransZ    = Ge/subComponent6/TransZTmp + Ge/subComponent6/HLZ cm
d:Ge/subComponent6/RotX      = 0 deg
d:Ge/subComponent6/RotY      = 0 deg
d:Ge/subComponent6/RotZ      = 0 deg
b:Ge/subComponent6/IsParallel = "true"

s:Ge/subComponent7/Type      = "TsBox"
s:Ge/subComponent7/Parent    = "VrtParallelWorld"
d:Ge/subComponent7/HLX      = Ge/Wall/HLX m
d:Ge/subComponent7/HLY      = Ge/Wall/HLY m
d:Ge/subComponent7/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent7/TransX    = 0 cm
d:Ge/subComponent7/TransY    = 0 cm
d:Ge/subComponent7/TransZTmp = Ge/subComponent6/TransZ + Ge/subComponent6/HLZ cm
d:Ge/subComponent7/TransZ    = Ge/subComponent7/TransZTmp + Ge/subComponent7/HLZ cm
d:Ge/subComponent7/RotX      = 0 deg
d:Ge/subComponent7/RotY      = 0 deg
d:Ge/subComponent7/RotZ      = 0 deg
b:Ge/subComponent7/IsParallel = "true"

s:Ge/subComponent8/Type      = "TsBox"
s:Ge/subComponent8/Parent    = "VrtParallelWorld"
d:Ge/subComponent8/HLX      = Ge/Wall/HLX m
d:Ge/subComponent8/HLY      = Ge/Wall/HLY m
d:Ge/subComponent8/HLZ      = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent8/TransX    = 0 cm
d:Ge/subComponent8/TransY    = 0 cm
d:Ge/subComponent8/TransZTmp = Ge/subComponent7/TransZ + Ge/subComponent7/HLZ cm
d:Ge/subComponent8/TransZ    = Ge/subComponent8/TransZTmp + Ge/subComponent8/HLZ cm
d:Ge/subComponent8/RotX      = 0 deg
d:Ge/subComponent8/RotY      = 0 deg
d:Ge/subComponent8/RotZ      = 0 deg
b:Ge/subComponent8/IsParallel = "true"

```

(continues on next page)

(continued from previous page)

```

s:Ge/subComponent9/Type           = "TsBox"
s:Ge/subComponent9/Parent         = "VrtParallelWorld"
d:Ge/subComponent9/HLX           = Ge/Wall/HLX m
d:Ge/subComponent9/HLY           = Ge/Wall/HLY m
d:Ge/subComponent9/HLZ           = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent9/TransX        = 0 cm
d:Ge/subComponent9/TransY        = 0 cm
d:Ge/subComponent9/TransZTmp     = Ge/subComponent8/TransZ + Ge/subComponent8/HLZ cm
d:Ge/subComponent9/TransZ        = Ge/subComponent9/TransZTmp + Ge/subComponent9/HLZ cm
d:Ge/subComponent9/RotX          = 0 deg
d:Ge/subComponent9/RotY          = 0 deg
d:Ge/subComponent9/RotZ          = 0 deg
b:Ge/subComponent9/IsParallel    = "true"

s:Ge/subComponent10/Type          = "TsBox"
s:Ge/subComponent10/Parent        = "VrtParallelWorld"
d:Ge/subComponent10/HLX          = Ge/Wall/HLX m
d:Ge/subComponent10/HLY          = Ge/Wall/HLY m
d:Ge/subComponent10/HLZ          = 0.5 * Ge/subComponent/Thickness cm
d:Ge/subComponent10/TransX        = 0 cm
d:Ge/subComponent10/TransY        = 0 cm
d:Ge/subComponent10/TransZTmp    = Ge/subComponent9/TransZ + Ge/subComponent9/HLZ cm
d:Ge/subComponent10/TransZ        = Ge/subComponent10/TransZTmp + Ge/subComponent10/HLZ_
→cm
d:Ge/subComponent10/RotX          = 0 deg
d:Ge/subComponent10/RotY          = 0 deg
d:Ge/subComponent10/RotZ          = 0 deg
b:Ge/subComponent10/IsParallel    = "true"

#####
# VRT
#####
b:Vr/UseVarianceReduction         = "true"
b:Vr/ParticleSplit/Active         = "true"
sv:Vr/ParticleSplit/ParticleName = 1 "neutron"
s:Vr/ParticleSplit/Component      = "VrtParallelWorld"
sv:Vr/ParticleSplit/SubComponents = 10
"subComponent1" "subComponent2"
"subComponent3" "subComponent4"
"subComponent5" "subComponent6"
"subComponent7" "subComponent8"
"subComponent9" "subComponent10"
s:Vr/ParticleSplit/Type = "WeightWindow"
uv:Vr/ParticleSplit/WeightMap = 10 1.0 0.5 0.25 0.125 0.0625 0.03125 0.015625 0.
→0078125 0.00390625 0.001953125
dv:Vr/ParticleSplit/EnergyMap = 10 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 GeV
u:Vr/ParticleSplit/UpperLimitFactor = 1
u:Vr/ParticleSplit/SurvivalFactor   = 1
i:Vr/ParticleSplit/MaximumSplitNumber = 100
s:Vr/ParticleSplit/PlaceOfAction    = "OnBoundary"

#####
# Scorer
#####
s:Sc/scorer/Quantity               = "DoseToMedium"
s:Sc/scorer/Component              = "Wall"

```

(continues on next page)

(continued from previous page)

```

s:Sc/scorer/OutputFile           = "WeightWindow"
s:Sc/scorer/IfOutputFileAlreadyExists = "Overwrite"
sv:Sc/scorer/Report              = 3 "Sum" "Mean" "Variance"

#####
# Beam setting
#####
s:So/Example/Type                = "Beam"
s:So/Example/Component            = "BeamPosition"
s:So/Example/BeamParticle        = "neutron"
d:So/Example/BeamEnergy          = 10 MeV
u:So/Example/BeamEnergySpread    = 0.757504
s:So/Example/BeamPositionDistribution = "Gaussian"
s:So/Example/BeamPositionCutoffShape = "Ellipse"
d:So/Example/BeamPositionCutoffX = 10. cm
d:So/Example/BeamPositionCutoffY = 10. cm
d:So/Example/BeamPositionSpreadX = 0.65 cm
d:So/Example/BeamPositionSpreadY = 0.65 cm
s:So/Example/BeamAngularDistribution = "Gaussian"
d:So/Example/BeamAngularCutoffX = 90. deg
d:So/Example/BeamAngularCutoffY = 90. deg
d:So/Example/BeamAngularSpreadX = 0.0032 rad
d:So/Example/BeamAngularSpreadY = 0.0032 rad
i:So/Example/NumberOfHistoriesInRun = 10

d:Ge/BeamPosition/TransZ = -1.0 * Ge/World/HLZ m
d:Ge/BeamPosition/RotX   = 0 deg
i:Ts/ShowHistoryCountAtInterval = 1

b:Ts/PauseBeforeQuit = "True"
s:Gr/view/Type       = "OpenGL"
b:Gr/view/IncludeAxes = "True"

```



---

## Introduction to Extensions

---

While most TOPAS users will find that they can implement everything they want from parameter files, those who require additional functionality are free to write their own C++ code to extend TOPAS. Your code can take advantage of the full syntax richness of C++. You may use almost any Geant4 class in your work.

These new classes can be:

- Geometry Components
- Scorers
- Outcome Models
- Filters
- Physics Lists and Physics Modules
- Particle Sources
- Magnetic Field Descriptions
- ElectroMagnetic Field Descriptions
- Imaging To Material Converters

And you can also provide classes that will be called to do whatever you want at:

- Start or End of Session
- Start or End of Run
- Start or End of History

As the first line of each of your classes, you will provide very specific comment lines that tells us how to weave your class into the rest of TOPAS. For example, to define your own Geometry Component, your class will start with something like:

```
// Component for MyComponent1
```

This tells TOPAS that your class defines a Geometry Component, and that this component should be used if a parameter file has the matching component type:

```
s:Ge/something/Type = "MyComponent1"
```

C++ does not require that a give file, such as `MyComponent1.cc`, contain a class of the same name. However the Topas make system DOES require that this file name and class name match. So, for example, a file named `MyComponent1.cc` and is corresponding `MyComponent1.hh` must contain a class named `MyComponent1`.

You can find a set of example extensions on the [topasmc.org](http://topasmc.org) code repository page. You can see there what the special comment string is for each type of class (Geometry Component, Scorer, Filter, etc.).

To build your new TOPAS executable that incorporates all of your extensions, you run CMake with an argument that tells it the location of your extensions. Your extensions then coexist with the rest of the TOPAS code.

You can even have subdirectories within your extensions directory, so that you might for example have different subdirectories with extensions from different collaborators:

```
extensions/my_extensions_from_university_a
extensions/my_extensions_from_company_b
extensions/some_other_extensions
```

Our CMake script will recursively search your extensions directory to take all of your extensions.

Details are in the README in TOPAS.

Even when you have to write your own C++ code, TOPAS work is still easier than plain Geant4. You write your extensions as concrete implementations of TOPAS base classes which provide a wealth of helper functions to simplify your work. You may use the TOPAS parameter system to provide parameters to your classes, and those parameters can vary in time, like any other TOPAS parameters. All user extensions have a pointer to the parameter manager in their constructor. Thus, to access TOPAS parameters, call one of the following methods: `fPm->someMethod`

In all of the following forms, the `parameterName` argument can be either a `G4String` or a `char*`.

```
// See if parameter exists
G4bool ParameterExists(parameterName);
// Get number of values in a vector parameter
G4int GetVectorLength(parameterName);
// Get dimensioned double value of parameter in Geant4's internal units
G4double GetDoubleParameter(parameterName, const char* unitCategory);
// Get double value of a unitless parameter
G4double GetUnitlessParameter(parameterName);
// Get integer value of parameter
G4int GetIntegerParameter(parameterName);
// Get Boolean value of parameter
G4bool GetBooleanParameter(parameterName);
// Get string value of parameter (whether it is actually a string parameter or not)
G4String GetStringParameter(parameterName);
// Get vector of dimensioned double values of parameter in Geant4's internal units
G4double* GetDoubleVector(parameterName, const char* unitCategory);
// Get vector of double values of a unitless parameter
G4double* GetUnitlessVector(parameterName);
// Get vector of integer values of parameter
G4int* GetIntegerVector(parameterName);
// Get vector of Boolean values of parameter
G4bool* GetBooleanVector(parameterName);
// Get vector of string values of parameter
G4String* GetStringVector(parameterName);
// Get TwoVector of double values of parameter in Geant4's internal units
G4TwoVector GetTwoVectorParameter(parameterName, const char* unitCategory);
// Get ThreeVector of double values of parameter in Geant4's internal units
G4ThreeVector GetThreeVectorParameter(parameterName, const char* unitCategory);
```



Stubs of extension classes are included in the `topas/extensions` directory in your TOPAS release. A set of additional example components, scorers and filters are distributed as a zip file on the TOPAS web site (see the file called `extension_examples...`). To create your own extension, start with the example that is the closest to what you want, then change the file name (and the class name throughout the file), then adjust the code as you wish.

We believe this extensions mechanism should allow you to do almost anything you like from within TOPAS. If you find any significant limitations, please reach out to us. We want to enable your unique research.

## 31.1 Extra Classes

First line of the `cc` file must be of the form:

```
// Extra Class for use by TsMyBeginHistory
```

Any of your extension classes are welcome to themselves instantiate other classes. You just need to advise us to link in these classes by providing the above special line.

## 31.2 Changeable Parameters

In general, parameters cannot change once the TOPAS session has begun. Changes due to *Time Features* are fine (since the time feature's behavior itself is well defined), but any other change violates basic principles of repeatability.

C++ code that changes a parameter during the session, aside from time features, is allowed only for a special case in which a specialized geometry component needs to set a parameter value on the fly. An example is when `TsCompensator` reads in the compensator definition from a special file format. The resulting compensator thickness updates a parameter that affects positioning of other components.

Such a special case is allowed if the relevant parameter is defined from the start to be “Changeable”. This is done by adding a `c` at the end of the parameter type, for example:

```
dc:Ge/Compensator/TransZ = 2. cm # the initial dc indicates that this is a double_
    ↳that is changeable
```

For vector parameters, the `c` still comes just before the colon, for example:

```
svc:...
```

In a complex parameter file chain, if any level of the chain redefines this as just a `d` rather than a `dc`, other parameter files will see this as a non-changeable parameter. Thus one parameter file may lock out others from making such changes.

TOPAS makes note of which parts of the system uses this changeable parameter (either directly or through a chain of parameters depending on other parameters) and takes care to explicitly update those parts of the system if this parameter ever changes.

Of course any parameter value can override the same parameter's value from a parent parameter file. This override at initial parameter read-in time is not what we mean by changeable. By Changeable we mean a value that changes during the TOPAS session.

The `c` syntax is not required when you are simply setting a parameter's value to a time feature. We allow:

```
d:Ge/Propeller/RotZ = Tf/PropellerRot/Value
```

It is true that this `Tf/PropellerRot/Value` is changeable, but that is handled internally by TOPAS.

## 31.3 Transient Parameters

When a parameter is changed during the session, either because it is a time feature value, or because some piece of C++ code changes the parameter, TOPAS does not actually overwrite the original parameter in memory, but instead adds it to a “Transient Parameter List”. The Transient Parameter list always takes precedence over any other parameters file.

Transient parameters may be the first occurrence of a given parameter, as for the materials for a patient that are only instantiated as the patient is read in from DICOM, or transient parameters may override previously-defined parameters.

---

## Custom Geometry Components

---

First line of the cc file must be of the form:

```
// Component for MyComponent1
```

TOPAS geometry components are like small pieces of what Geant4 users call their “detector construction” class. The Geometry sections of the Geant4 Application Developers Guide provide details on the full geometrical functionality of Geant4. In this section, we explain some details about how to write TOPAS components, but we assume that you are already comfortable with basic concepts of C++ and Geant4 geometry. The notes below are intended to discuss only those parts which may not be obvious.

Your geometry component class will be a concrete implementation of the base class: `TsVGeometryComponent`.

You can get any parameter name of the current component by using the `GetFullParmName` method. For example, if your parameter file specifies:

```
d:Ge/MyComponent/Blatz = 42. mm
```

`GetFullParmName("Blatz")` will return `Ge/MyComponent/Blatz`. You can then feed this resulting string into the parameter access methods such as:

```
G4Double blatzLength = fPm->GetDoubleParameter(GetFullParmName("Blatz"), "length");
```

Your component may contain any of the following methods. Careful attention to what goes where will insure that your classes are robust under 4D and the base classes will do much of the work for you.

**Constructor:** Must exist and may be empty. This method will only be called at the very beginning of the simulation. It will not be called after changes in 4D. Only put things here if you are absolutely certain you will not need to recompute them during the simulation.

**Destructor:** Must exist and may be empty. Destroy any special objects you created with “new” statements. You may not destroy solids, logical volumes or physical volumes. These destructions are handled for you by the base class.

You do not need to do anything to handle the basic parameters, Parent, TransX, TransY, TransZ, RotX, RotY, RotZ, Material and Color. These are handled for you by the base class, including 4D capabilities.

If there are any other parameters that you may want to vary in 4D, provide a method `UpdateForSpecificParameterChange(G4String parameter)`.

- If the parameter name is one that you want to handle, do so. Be sure to use `GetFullParmNameLower` rather than `GetFullParmName` in your check.
- If your handling of this parameter moves a volume relative to its mother volume, advise Geant4's smart voxel system that it needs to re-optimize the mother volume by calling `AddToReoptimizeList`. The argument should be the mother's logical volume.
- If the parameter name is not one that you want to handle, pass it on to the base class handler, `TsVGeometryComponent::UpdateForSpecificParameterChange`. This is essential to enable basic 4D behaviors such as overall component motion.

`TsMultiLeafCollimator.cc` is a good example of this kind of behavior. It allows leaf position to change over time.

For the rest of your work, provide a method `Construct`.

The first line of `Construct` MUST be:

```
BeginConstruction();
```

The rest of `Construct` is whatever you want to do to create Geant4 Solids, Logical Volumes and Physical Volumes. But you must follow some rules to insure that TOPAS will be able to properly manage your volumes in 4D.

- You create Geant4 Solids just as you would in any Geant4 geometry.
- You DO NOT create Geant4 Logical Volumes or Physical Volumes directly, but instead use helper methods from the base class. This allows TOPAS to manage your solids and volumes efficiently, even if they are moving.

To create the overall logical volume for your component, use:

```
fEnvelopeLog = CreateLogicalVolume(G4VSolid* solid);
```

The logical volume will automatically get the material and visualization properties specified in your parameter file for this component, such as `Ge/MyComponent/Material` and `Ge/MyComponent/Color`. Be sure that the value on the left side of the above is exactly "`fEnvelopeLog`". This insures that TOPAS knows the overall logical volume's name and is essential for TOPAS to support your component in 4D.

If a component is made up of more than one volume, these additional volumes are called "SubComponents." A component may have more zero, one or more SubComponents. An example of SubComponents is the Blades in a Propeller, such as:

```
s:Ge/Propeller/Type = "TsPropeller"
...
i:Ge/Propeller/NbOfBlades = Ge/PropellerConstant/NbBlades
s:Ge/Propeller/Blade/Material = "World"
s:Ge/Propeller/Blade/Color = "skyblue"
```

In all of the following forms, the `subComponentName` argument can be either a `G4String` or a `char*`.

To create a logical volume for a subcomponent, use:

```
G4LogicalVolume* CreateLogicalVolume(subComponentName, G4VSolid* solid);
```

TOPAS will look for material and visualization parameters such as:

```
Ge/ComponentName/SubComponentName/Material = ...
```

To hard-code the material, rather than having it come from this parameter, use:

```
G4LogicalVolume* CreateLogicalVolume( subComponentName, G4String& materialName,
    ↪G4VSolid* solid);
```

This is particularly useful in cases where you want the material to be the same as the component's mother, that is, the material surrounding your component. We do this, for example, when we want to make a void in a collimator. To get that mother volume's material name, use:

```
G4String envelopeMaterialName = fParentComponent->GetResolvedMaterialName();
```

The base class will take care of automatically setting your component's visualization attributes based on the component's parameters. But you can set different attributes for subComponents with code such as:

```
G4VisAttributes* yokeColor = new G4VisAttributes(G4Colour(0.2, 1.0, 0.2)); // Sets
    ↪RGB color
RegisterVisAtt(yokeColor); // Necessary so that TOPAS can delete the attribute if the
    ↪component is rebuild during 4D behavior
yokeLogicalVolumePointer->SetVisAttributes(yokeColor);
```

To create the overall physical volume for your component, use:

```
fEnvelopePhys = CreatePhysicalVolume(fEnvelopeLog);
```

Be sure that the value on the left side of the above is exactly `fEnvelopePhys`. This insures that TOPAS knows the overall physical volume's name and is essential for TOPAS to support your component in 4D.

Additional forms of `CreatePhysicalVolume` allow you to place subcomponents within your component.

To place a subcomponent in the center of your logical volume `lVol`:

```
G4VPhysicalVolume* CreatePhysicalVolume(subComponentName, G4LogicalVolume* lVol,
    ↪G4VPhysicalVolume* parent);
```

To place a subcomponent into your logical volume `lVol`, with an offset or rotation:

```
G4VPhysicalVolume* CreatePhysicalVolume(subComponentName, G4LogicalVolume* lVol,
    ↪G4RotationMatrix* rot, G4ThreeVector* trans, G4VPhysicalVolume* parent);
```

To place multiple copies of the same subcomponent name into your logical volume, call:

```
G4VPhysicalVolume* CreatePhysicalVolume(subComponentName, G4int copy, G4bool
    ↪reuseLogical, G4LogicalVolume* lVol, G4RotationMatrix* rot, G4ThreeVector* trans,
    ↪G4VPhysicalVolume* parent);
```

- `copy` should be a unique integer to differentiate the different copies of your subcomponent. This copy number is useful in some of the visualization commands when you want to control just one copy or another.
- Set `reuseLogical` true if you are using the same logical volume in all of these placements. This is efficient if all of the copies of the subcomponent are identical except for their placement.
- Set `reuseLogical` false if you are using different logical volumes in each of these placements. This allows you to make each copy of the subcomponent different (different material, different shape, different size, etc.).

To place multiple copies of the same subcomponent using a Geant4 parameterization (creating Geant4 parameterized volumes), call:

```
G4VPhysicalVolume* CreatePhysicalVolume(const char* subComponentName,
    ↪G4LogicalVolume* lVol, G4VPhysicalVolume* parent, const EAxis pAxis, const G4int
    ↪nReplicas, G4VPVParameterisation* pParam);
```

To place multiple copies of the same subcomponent using a Geant4 replica volume, call:

```
G4VPhysicalVolume* CreatePhysicalVolume(const char* subComponentName,
↳G4LogicalVolume* lVol, G4VPhysicalVolume* parent, const EAxis pAxis, const G4int
↳nReplicas, G4double width);
```

The last line of Construct MUST be:

```
return fEnvelopePhys;
```

Some helper functions you may want to use from the TsParameterManager:

```
G4VisAttributes* GetColor(G4String name);
G4VisAttributes* GetColor(const char* name);
G4VisAttributes* GetInvisible();
```

Some helper functions you may want to use from the TsVGeometryComponent:

```
SetTooComplexForOGLES()
```

Call this to tell Graphics that this component has become too complex to efficiently render in OpenGL's Stored Mode. It will instead be rendered in OpenGL's Immediate Mode (can be less quick to update, but uses less memory)

```
GetMaterial
```

By default, the logical volumes you create will get their material from the material parameter you specified for this component. But you can use GetMaterial to obtain any other named material.

---

Custom Particle Sources

---

First line of the cc file must be of the form:

```
// Particle Source for MyParticleSource1  
or  
// Particle Generator for MyParticleGenerator1
```

Your particle source defines the initial particles that are then transported by the simulation. Because Geant4's multi-threaded capability keeps part of this functionality in the master thread and other parts in the worker threads, you actually create two separate classes to create a particle source.

For the part of the source that controls overall behavior (usually just setting the number of histories, but optionally also things like reading in some kind of phase space file), you write a class derived from `TsSource`. TOPAS instantiates this in the Geant4 Master thread. If you really just need this class to set the number of histories, you may just use our existing `TsSource` (that is, you don't have to write your own class at all for this part).

For the part of the source that generates the individual events (setting the primary particle positions and momenta), you write a class derived from `TsVGenerator`. TOPAS instantiates this in the Geant4 worker thread.

In both cases, parameter lookups should be done in `ResolveParameters`. Call `ResolveParameters` directly from your constructor, and then you can also rely on TOPAS to re-call this method any time one of this particle source's parameters is changed.

TOPAS will call your `GeneratePrimaries` method once per history. You should always start this method with this test:

```
if (CurrentSourceHasGeneratedEnough()) return;
```

This allows your source to properly coexist with other sources that may have other numbers of histories.

The body of your `GeneratePrimaries` method should create and fill some number of `TsPrimaryParticles` (a single history may contain zero, one or more primary particles).

The `TsPrimaryParticle` structure is defined in the header file `TsVParticleSource.hh`. For each `TsPrimaryParticle` that you define, call `GenerateOnePrimary`.

Once you have finished creating all of the `TsPrimaryParticles` for this history, call `AddPrimariesToEvent`.





---

### Custom Physics Lists and Physics Modules

---

First line of the cc file must be of the form:

```
// Physics List for MyPhysicsList1  
or  
// Physics Module for MyPhysicsModule1
```

You can supply your own physics list or physics module. Note however that this option is not recommended unless you have significant Geant4 expertise. Even most long-time Geant4 users get into difficulty writing their own physics lists and physics modules. Wherever possible, you should try to use one of the Reference physics list or the Modular physics list with pre-written Geant4 physics modules.

The example physics list and physics module provided in `topas/extensions/MyPhysicsList1` and `MyPhysicsModule` include pointers to the TOPAS parameter manager as their arguments. This is not required, but allows you to use TOPAS parameters to adjust options within your list or modules.



---

Custom Scorers

---

First line of the cc file must be of the form:

```
// Scorer for MyScorer1
```

Your custom scorer can either accumulate binned data (like our built-in dose scorer), or n-tuple data (like our built-in phase space scorer).

- For binned scorers, your scorer should inherit from `TsVBinnedScorer`.
- For n-tuple scorers, your scorer should inherit from `TsVNtupleScorer`.

At a minimum, your scorer should provide a constructor, a destructor and a `ProcessHits` method. The base class will take care of all the details of filtering, accumulating and outputting results.

For binned scorers, your scorer's constructor must contain a call to:

```
SetUnits
```

For n-tuple scorers, your scorer's constructor defines each column and its data type by calls to:

```
RegisterColumnD  
RegisterColumnF  
RegisterColumnI  
RegisterColumnI8  
RegisterColumnB  
RegisterColumnS
```

`RegisterColumnD` and `RegisterColumnF` also take a unit string.

If your scorer is a Surface Scorer, the constructor must also contain the line:

```
SetSurfaceScorer();
```

Otherwise, your scorer is assumed to be a Volume Scorer.

The scorer's `ProcessHits` method must be written carefully to avoid slowing down the simulation since this method is called for every hit in the scoring component. Slow operations such as string comparisons should be avoided here.

Try to write your code so that you perform these sorts of slow operations only during construction, save values and pointers in class variables and then use these pre-calculated values in the ProcessHits method. Once you have your value computed:

- For binned scorers, accumulate data by calling `AccumulateHit`
- For n-tuple scorers, accumulate data by calling `fNtuple->Fill`

If you want to take more complete control of the scoring process, you can provide optional methods:

```
// called after the last hit of a given track
void UserHookForEndOfTrack()
// called after the last hit of all tracks resulting from a given particle incident_
→on the scoring component
void UserHookForEndOfIncidentParticle()
// called at the end of the event
void UserHookForEndOfEvent()
// called at the end of the run
void UserHookForEndOfRun()
```

Between the ProcessHits method and these other four methods, you have complete control over how you will accumulate and handle your scored values. Accumulate values in your own data structures that you provide in your scorer's header file or in other classes that your scorer calls. Manipulate and output these values as you wish. It is all up to you. You can still choose to fill the fEvtMap just like a regular scorer, in which case TOPAS will accumulate and output those values, or you can fill nothing into that fEvtMap, in which case TOPAS will not take any further action for this scorer.

Some helper functions you may want to use from the TsParameterManager:

```
// Activates creation of the TsTrackInformation object
SetNeedsTrackingAction
// Activates creation of the extra part of the TsTrackInformation object that_
→contains information on what volumes were traversed
SetNeedsSteppingAction
```

Some helper functions you may want to use from the TsVScorer:

```
// Get pointer to a material
GetMaterial
// Tell whether a given material is used in the geometry
UsedMaterial
// Get the voxel index from hits in divided or parameterized components
GetIndex
// Get the current TOPAS time (for the time of flight, use fTimeOfFlight)
GetTime
GetRunID
GetEventID
GetRandomNumberStatusForThisEvent
// Disable the automatic creation and filling of output, leaving this work entirely_
→to your scorer
SuppressStandardOutputHandling
```

For divided components, the combined index one finds in scorers is formed from three bin indices (x,y,z or r, phi, z or r, phi, theta for TsBox, TsCylinder and TsSphere respectively). A helper function is now provided to return the individual bin indices given the combined index:

```
GetBin(index, iBin) // where iBin is 0, 1 or 2
```

A scorer can itself instantiate additional scorers. We refer to these as “SubScorers”. The main scorer can then perform

calculations using results of one or more subscorers to obtain a final value. A good example of this is in `ExtensionExamplesMore/MyScoreProtonLET`. At the end of the constructor, it contains the following:

```
InstantiateSubScorer("ProtonLET_Denominator", outFileName, "Denominator");
```

And later there is a method that combines the scorer and the subscorer on a bin-by-bin basis to obtain a final quantity per bin:

```
G4int MyScoreProtonLET::CombineSubScorers()
...
TsVBinnedScorer* denomScorer = dynamic_cast<TsVBinnedScorer*>(GetSubScorer(
    ↪ "Denominator"));
    fFirstMomentMap[index] = fFirstMomentMap[index] / denomScorer->
    ↪ fFirstMomentMap[index];
```



---

### Custom Outcome Models

---

First line of the cc file must be of the form:

```
// Outcome Model for MyOutcomeModel1
```

Your custom outcome model can perform whatever analysis you wish from a TOPAS DVH. The work is all in your `Initialize` method.

See `ExtensionsExamples/MyOutcomeModel1`





## CHAPTER 37

---

### Custom Filters

---

First line of the cc file must be of the form:

```
// Filter for OnlyIncludeParticlesOfTwiceAtomicNumber,  
->OnlyIncludeParticlesNotOfTwiceAtomicNumber
```

Note that a single filter can be used for more than one filter condition, hence comma separated list.

Filters must be written carefully to avoid slowing down the simulation. The filter's Accept method is called for every hit in the scoring component. Slow operations such as string comparisons should be avoided during this method. Try to write your code so that you perform these sorts of slow operations only during the constructor, ResolveParameters method or CacheGeometryPointers method, save values and pointers in class variables and then use these pre-calculated values in the ProcessHits method.

Parameter lookups should be done in ResolveParameters. Call ResolveParameters directly from your constructor, and then you can also rely on TOPAS to re-call this method any time one of this filter's parameters is changed.

4D behaviors may require TOPAS to destroy and rebuild components during the simulation. Accordingly, you can not rely on the pointer to a given component remaining the same throughout the simulation. Any lookup of a component pointer should be done in the filter's CacheGeometryPointers method. TOPAS will re-call this method any time relevant components are rebuilt.

Some helper functions you may want to use from the TsParameterManager:

```
// Activates creation of the TsTrackInformation object  
SetNeedsTrackingAction  
// Activates creation of the extra part of the TsTrackInformation object that_  
->contains information on what volumes were traversed  
SetNeedsSteppingAction
```

Some helper functions you may want to use from the TsVFilter:

```
// Get pointer to a material  
GetMaterial  
// Get pointer to a named physics volume  
GetPhysicalVolume
```

(continues on next page)

(continued from previous page)

```
// Get pointer to a named component
GetComponent
// Get pointers to all children of a named component
GetChildComponentsOf
```

## CHAPTER 38

---

### Custom Fields

---

While an ElectroMagnetic Field can have just an electric field, just a magnetic field, or a combination of the two, Geant4's architecture does not provide any base class for a purely electric field, but instead provides one base class for purely Magnetic fields and another for combined ElectroMagnetic fields. To allow you to use all features of both Geant4 classes, TOPAS emulates this curious aspect of Geant4's design.

If you want to create a purely magnetic field, the first line of the cc file should be of the form:

```
// Magnetic Field for MyField1
```

If you want to create a purely electric field, or a combined electromagnetic field, the first line of the cc file should be of the form:

```
// ElectroMagnetic Field for MyField1
```

and then if what you really wanted was just an electric field, you implement the magnetic field strength as just zero.

Geant4 will call your GetFieldValue every time it needs to query the field. For reasons that are not clear to this author, Geant4 will sometimes query your field for points outside of your intended geometry component, so make sure to return at least some value (at least a zero) for every possible point.

Parameter lookups should be done in ResolveParameters. Call ResolveParameters directly from your constructor, and then you can also rely on TOPAS to re-call this method any time one of this field class's parameters is changed.



---

## Custom Imaging to Material Conversion

---

First line of the cc file must be of the form:

```
// Imaging to Material Converter for MyImagingToMaterialConverter1
```

You can supply your own class to assign imaging values to materials.

To use your Imaging to Material converter, reference its name in the parameter:

```
s:Ge/Patient/ImagingToMaterialConverter = "MyImagingToMaterialConverter1"
```

The number of image files read by Topas is determined by the parameter:

```
i:Ge/Patient/NumberOfEnergies = 1 # defaults to 1
```

If this value is just 1, Topas will look for imaging files directly in your DicomDirectory. If this value is larger, Topas will expect your DicomDirectory to contain numbered subdirectories:

```
YourDicomDirectory/1  
YourDicomDirectory/2  
etc.
```

The allowed modalities of the imaging files is determined by the parameter:

```
sv:Ge/Patient/DicomModalityTags = 1 "CT" # defaults to just CT
```

Other modality tags are, for example, MR for Magnetic Resonance and US for Ultrasound. A complete list can be found [here](#).

You apply whatever algorithm you like in your class's AssignMaterial method.

This will be called once for each voxel. TOPAS will pass you a vector of imagingValues, with each value representing this voxel's value from one of your image values:

```
std::vector< signed short >* imagingValues
```

For example, if you are doing Dual Energy CT, you will get two values in this vector, the HU values from each of the two CT files.

The materials you use can be defined in your parameter file, in your `ImagingToMaterial` class's constructor or in your `ImagingToMaterial` class's `AssignMaterial` method. Either way, by the time you are finished assigning all of your materials, you will have built up a vector of pointers to materials in `fMaterialList` and, for each voxel, your `AssignMaterials` method will have returned an appropriate index into this vector.

`AssignMaterials` is also passed a `timeSliceIndex`, which is useful if your imaging is time-dependent (that is, 4D imaging). In this case you can use the `timeSliceIndex` however you wish in your `AssignMaterials` algorithm.

To avoid spending CPU time on repeated parameter lookups, it is best to do them in `ResolveParameters`. Call `ResolveParameters` directly from your constructor, and then you can also rely on TOPAS to re-call this method any time one of this class's parameters are changed.

---

### Additional User Hooks

---

Six additional hooks are provided in `TsExtensionManager` for you to attach your own code.

- `BeginSession`
- `BeginRun`
- `BeginHistory`
- `EndHistory`
- `EndRun`
- `EndSession`

First line of the cc file must be of the form:

```
// BeginSession for TOPAS
// BeginRun for TOPAS
// BeginHistory for TOPAS
// EndHistory for TOPAS
// EndRun for TOPAS
// EndSession for TOPAS
```

There are no particular constraints on what you can do in these methods. They are provided simply to give you more flexibility in the design of your extensions.





---

Proton RBE Scorers

---

TOPAS has included RBE scorers for several RBE models. These extensions are available from Github at: <https://github.com/topasmc/extensions>

The implementation follows in principle the methods described in [Polster2015]. The models can be separated into two categories, scorers that depend on dose, LET and alpha/beta ratios, and scorers that are not directly a function of LET. For a detailed description of each model please refer to the references provided.

The following models have been implemented for proton RBE calculations:

LET-based scorers:

1. Carabe [Carabe2012]; [Carabe2007]
2. Chen [Chen2012]
3. McNamara [McNamara2015]
4. Microdosimetric Kinetic Model (MKM), LET based version [Hawkins1998]; [Kase2007]
5. Min-Max Model (a generic class, including McNamara and Carabe models)
6. Wedenberg [Wedenberg2013]
7. Wilkens [Wilkens2004]

Non-LET based scorers

8. Monte Carlo Damage Simulation (MCDS) for DSB Induction [Semenenko2004]; [Semenenko2006]; [Stewart2011]; [Stewart2015]
9. Repair Misrepair Fixation (RMF) [Carlson2008]; [Frese2012]
10. Tabulated RBE tables (e.g. using the PIDE data base) [Friedrich2012]

The basic concept behind the RBE scorers is to first calculate separately the dose and LET or other relevant quantities, and then, after the simulation is complete, to call the function `CombineSubScorers` to combine these separate quantities on a voxel by voxel bases to calculate one of the following quantities:

- RBE (“rbe”)
- Alpha (“alpha”)

- Beta (“beta”)
- Survival fraction (“survivalfraction” or “SF”)
- RBE x Dose (“rbe\_x\_dose” or “RWD”, short for RBE weighted dose)

The desired quantity is requested by specifying the `OutputQuantity` parameter, for example:

```
s:Sc/MyScorer/OutputQuantity = "RBE"
```

using the options indicated in brackets above (case insensitive).

## 41.1 Important notes

1. RBE-related calculations have to consider the entire radiation field. For single field irradiations, the RBE scorers provided here can be used to directly calculate RBE. However, for patient simulations that try to estimate RBE for multiple field treatments, one has to consider the total dose each voxel receives to obtain the correct RBE. Thus one would have to follow a dose-averaged summation of alpha and beta (see also the description in [Polster2015]).
2. For most LET-based scorers, it may be easier to simply score dose and LET, sum the dose and LET distributions across fields in a post-processing step, and then calculate RBE or RBE-weighted dose. TOPAS currently only provides an LET scorer for protons (*ProtonLET Scorer*).
3. Proton LET only considers energy depositions from protons and secondary electrons. Non-LET based models can also include contributions from other secondaries ( $Z>1$ ) and can be used for other ion irradiation modalities. This may also result in differences in RBE for some scenarios.

## 41.2 Normalizing simulations to prescriptions

Since the simulated number of histories is typically much smaller than the number of protons delivered, it is important to correctly normalize accumulated quantities (e.g. dose, fluence) to the prescribed dose, before using them to compute the RBE. We provide two normalization schemes which are appropriate for different types of simulation.

1. Simultaneous Exposure (default). This is appropriate for patient simulations, where the scorer bins measure RBE for a single irradiation. That is, a single beam passes through the scoring volume, and the RBE found in each bin is reported. In this case, the accumulated quantities are normalized by a scaling factor, whose numerator (i.e. the delivered dose) is set by the `PrescribedDose` parameter (e.g. `Sc/MyScorer/PrescribedDose = 60 Gy`) and whose denominator (i.e. the simulated dose) is chosen by the `PrescribedDoseMetric` and `PrescribedDoseStructure` parameters. The `PrescribedDoseMetric` can be chosen from “Max” (default), “Mean” and “D90”. The `PrescribedDoseStructure` parameter identifies an `RTSTRUCT` structure (e.g. “CTV”) for which the `PrescribedDoseMetric` is calculated. If `PrescribedDoseStructure` is not set, then the metric is evaluated for the entire scoring volume.

Note that, to use `RTStructures` in an RBE scorer, it is currently necessary to include the `RTStructure` in the `Ge/Patient/ColorByRTStructNames` parameter. Also, `RTStructures` can only be used when scoring RBE upon the CT grid (i.e. the scorer Component is a `TsDicomPatient`, not a `TsBox`).

2. Repeat exposures (chosen with `Sc/MyScorer/SimultaneousExposure = "False"`). This is appropriate for simulations of cell experiments, where each scorer bin measures RBE for a separate irradiation. That is, cell experiments are repeated in each scoring bin (e.g. depth) using the same prescribed dose (e.g. 2 Gy). TOPAS can simulate all these experiments in a single run, by normalizing accumulated quantities appropriately. In this case, the prescribed dose delivered to each scorer bin is set by the `PrescribedDose` parameter (e.g. `Sc/MyScorer/PrescribedDose = 2 Gy`).

## 41.3 Using tissue-specific RBE model parameters

When performing patient simulations, it is useful to ascribe different RBE model parameters to different tissues. For example, it is well known that the tumor can have a dramatically different alpha/beta ratio to the surrounding normal tissue. TOPAS supports this feature by enabling the user to assign cell lines (e.g. `Sc/CellLineV79`) to structures in the RTSTRUCT DICOM file. When scoring RBE in a specific voxel, the RBE model parameters are then retrieved from the corresponding cell line.

This is specified using the following parameters:

```
sv:Sc/MyScorer/RTStructures = 2 "CTV" "Brain"
sv:Sc/MyScorer/CellLines = 3 "Tumor" "BrainTissue" "OtherTissue"
```

The order of `RTStructures` is important because earlier structures take precedence over structures listed later (e.g. if a voxel is in CTV and Brain, then it uses model parameters for CTV). The number of strings in `CellLines` (which is used to lookup model parameters) is one greater than the number of structures. This allows a default `CellLine` to be defined (at the end). If no structures are listed (i.e. `RTStructures` parameter not defined), then `CellLines` has only one string, which defines the model parameters to use everywhere.

Note that, to use `RTStructures` in an RBE scorer, it is currently necessary to include the `RTStructure` in the `Ge/Patient/ColorByRTStructNames` parameter. Also, `RTStructures` can only be used when scoring RBE upon the CT grid (i.e. the scorer Component is a `TsDicomPatient`, not a `TsBox`).

## 41.4 Reusing sub-scorers to reduce simulation memory requirements

Each RBE scorer creates an additional scoring grid for each variable that is being used (known as sub-scorers). For example, the McNamara RBE scorer creates dose and LET sub-scorers. This significantly increases the memory footprint of the simulation. In particular, if one wants to use multiple RBE scorers, each has their own sub-scorers, which can result in huge requirements in RAM and CPU time. To avoid duplicating sub-scorers unnecessarily, we added the option for scorers to share sub-scorers, for example, when a dose scorer is already defined, let's say it's called "PhysDose", and an LET scorer called "ProtonLET", then multiple RBE scorers can use these scorers by defining:

```
s:Sc/McNamara/ReferencedSubScorer_Dose = "PhysDose"
s:Sc/McNamara/ReferencedSubScorer_LET = "ProtonLET"
```

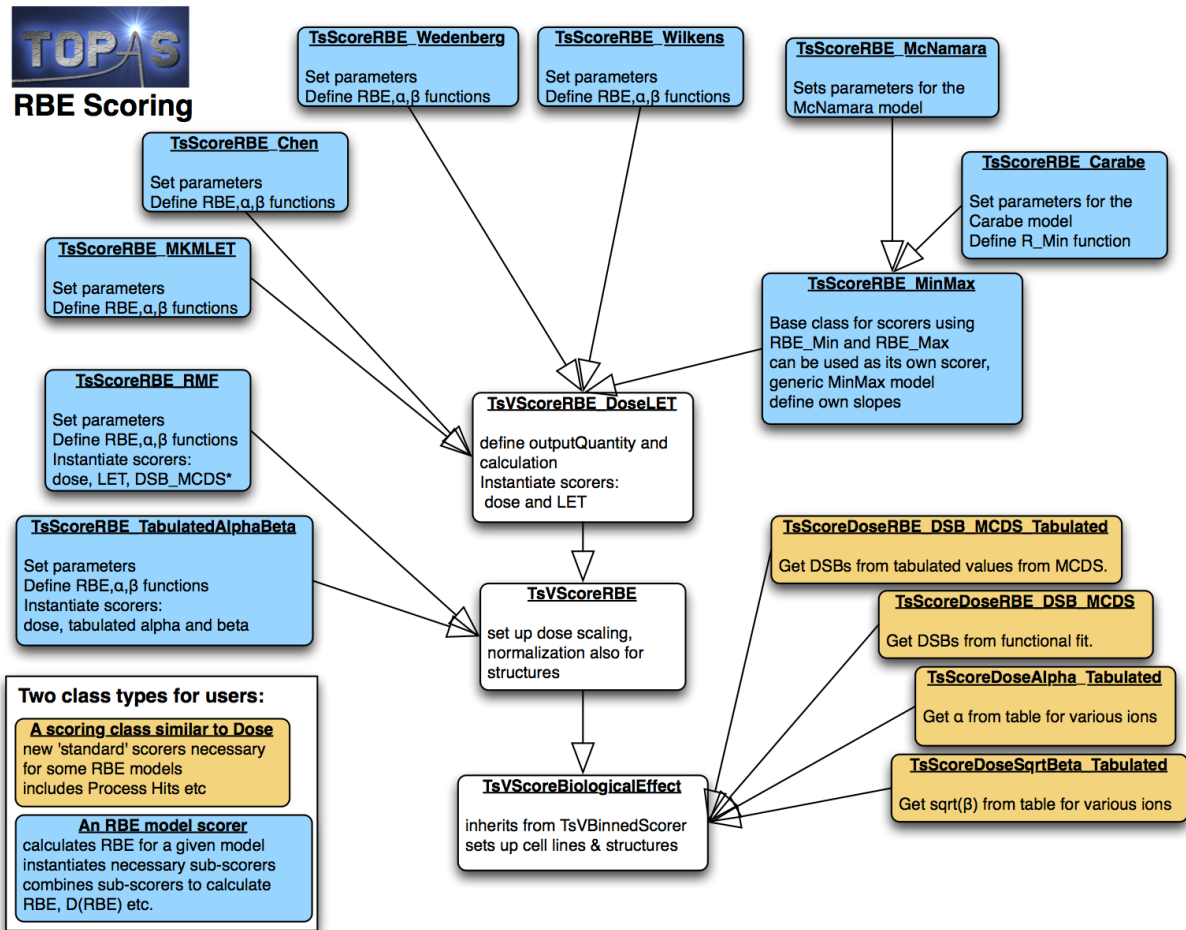
## 41.5 Content

The RBE scorers are contained in a folder that includes an example folder. The directory consists of the following file types:

Scorers:

- `TsScoreDose*`: These score quantities in the `ProcessHits` function like normal scorers.
- `RBE Scorers`: These do not have a `ProcessHits` function and instead combine scored properties (dose, LET, etc) to RBE or biological dose, etc.
- `TsV*`: Base classes for the scorers

A schematic view of the class hierarchy is shown below.



Additionally, the example directory contains an example experiment irradiation (experiment.txt) scoring each of the available RBE scorers (rbe\_scorers.txt) for V79 cells (CellLineV79.txt). V79 cells are used because they are one of the most studied cells and biological parameters for all models were available.

The simulations can be run with `topas run.txt` and analyzed with the provided python script.

In order to change the experimental setup, edit experiment.txt.

In order to change the cell line, provide a new cell line file and change the following line in run.txt:

```
sv:Sc/CellLines = 1 "CellLineV79"
```

run.txt also controls the `PrescribedDose` used to calculate RBE and the `OutputQuantity`. The output quantities available depend on the RBE model.

RBE scorers are defined in rbe\_scorers.txt and can be edited there. Typically, we recommend not to run too many scorers at once as that increases memory use. In particular, the two parameters `ReferencedSubScorer_Dose` and `ReferencedSubScorer_LET` should be set if a dose and LET scorer already exists, otherwise each RBE scorer will create sub-scorers for all properties it needs, potentially resulting in duplicated scorers.

## 41.6 Scoring

A typical scorer for using the RBE model looks like this (from examples/rbe\_scorers.txt):

```

s:Sc/McNamara/Quantity           = "RBE_McNamara"
s:Sc/McNamara/Component          = "Sample"
s:Sc/McNamara/OutputQuantity     = "RBE"
d:Sc/McNamara/PrescribedDose     = 2 Gy
sv:Sc/McNamara/CellLines         = 1 "CellLineV79"
b:Sc/McNamara/SimultaneousExposure = "True"
s:Sc/McNamara/OutputType         = "csv"
s:Sc/McNamara/OutputFile         = "OutputFileName"
s:Sc/McNamara/IfOutputFileAlreadyExists = "Overwrite"
s:Sc/McNamara/ReferencedSubScorer_Dose = "PhysDose" # these two scorers need
s:Sc/McNamara/ReferencedSubScorer_LET = "ProtonLET" # to be defined as well.

```

Each RBE scorer needs a cell line and selected variables to be defined. The file CellLineV79.txt includes all necessary information for all RBE scorers available. The cell lines can easily be changed by changing the CellLines parameter. You must provide the relevant parameters (e.g. alpha/beta ratio) for new cell lines. Note: We recommend to not use the provided CellLineV79.txt parameter file as it includes parameters for all models. Using a smaller file with only the parameters necessary for your simulations guarantees that you will use the correct values, not duplicate ones available in the sample file.

For a list of parameters necessary for each scorer please refer to the text files. The parameters necessary also depend on the selected OutputQuantity, for example in the Carabe model, to get RBE, one only has to define AlphaBetaRatio<sub>x</sub>, i.e. the ratio of  $\alpha/\beta$ , as:

```

d:Sc/CellLineV79/AlphaBetaRatiox = 1.412 Gy

```

If the requested output quantity is alpha, one also needs to define:

```

d:Sc/CellLineV79/Alphax = 0.0722 /Gy

```

For beta:

```

d:Sc/CellLineV79/Betax = 0.0502 /Gy2

```

and for the survival fraction all three parameters need to be set. Please refer to the example files and the scorers for details.

## 41.7 References



---

## Bibliography

---

- [Carabe2012] Carabe, A. et al., 2012. Range uncertainty in proton therapy due to variable biological effectiveness. *Physics in Medicine and Biology*, 57(5), pp.1159–1172.
- [Carabe2007] Carabe-Fernandez, A., Dale, R.G. & Jones, B., 2007. The incorporation of the concept of minimum RBE (RBEmin) into the linear-quadratic model and the potential for improved radiobiological analysis of high-LET treatments. *International Journal of Radiation Biology*, 83(1), pp.27–39.
- [Carlson2008] Carlson, D.J. et al., 2008. Combined use of Monte Carlo DNA damage simulations and deterministic repair models to examine putative mechanisms of cell killing. *Radiation Research*, 169(4), pp.447–459.
- [Chen2012] Chen, Y. & Ahmad, S., 2012. Empirical model estimation of relative biological effectiveness for proton beam therapy. *Radiat.Prot.Dosim.*, 149(2), pp.116–123.
- [Frese2012] Frese, M.C. et al., 2012. A Mechanism-Based Approach to Predict the Relative Biological Effectiveness of Protons and Carbon Ions in Radiation Therapy. *Int J Radiat Oncol*, 83(1), pp.442–450.
- [Friedrich2012] Friedrich, T. et al., 2012. Systematic analysis of RBE and related quantities using a database of cell survival experiments with ion beam irradiation. *Journal of Radiation Research*, 54(3), pp.rrs114–514.
- [Hawkins1998] Hawkins, R.B., 1998. A microdosimetric-kinetic theory of the dependence of the RBE for cell death on LET. *Medical Physics*, 25(7), pp.1157–1170.
- [Kase2007] Kase, Y. et al., 2007. Biophysical calculation of cell survival probabilities using amorphous track structure models for heavy-ion irradiation. *Physics in Medicine and Biology*, 53(1), pp.37–59.
- [McNamara2015] McNamara, A.L., Schuemann, J. & Paganetti, H., 2015. A phenomenological relative biological effectiveness (RBE) model for proton therapy based on all published in vitro cell survival data. *Physics in Medicine and Biology*, 60(21), pp.8399–8416.
- [Polster2015] Polster, L. et al., 2015. Extension of TOPAS for the simulation of proton radiation effects considering molecular and cellular endpoints. *Physics in Medicine and Biology*, 60(13), pp.5053–5070.
- [Semenenko2004] Semenenko, V.A. & Stewart, R.D., 2004. A fast Monte Carlo algorithm to simulate the spectrum of DNA damages formed by ionizing radiation. *Radiation Research*, 161(4), pp.451–457.
- [Semenenko2006] Semenenko, V.A. & Stewart, R.D., 2006. Fast Monte Carlo simulation of DNA damage formed by electrons and light ions. *Physics in Medicine and Biology*, 51(7), pp.1693–1706.
- [Stewart2011] Stewart, R.D. et al., 2011. Effects of Radiation Quality and Oxygen on Clustered DNA Lesions and Cell Death. *Radiation Research*, 176(5), pp.587–602.

- [Stewart2015] Stewart, R.D. et al., 2015. Rapid MCNP simulation of DNA double strand break (DSB) relative biological effectiveness (RBE) for photons, neutrons, and light ions. *Physics in Medicine and Biology*, 60(21), pp.8249–8274.
- [Wedenberg2013] Wedenberg, M., Lind, B.K. & Hårdemark, B., 2013. A model for the relative biological effectiveness of protons: the tissue specific parameter  $\alpha/\beta$  of photons is a predictor for the sensitivity to LET changes. *Acta Oncologica*, 52(3), pp.580–588.
- [Wilkens2004] Wilkens, J.J. & Oelfke, U., 2004. A phenomenological model for the relative biological effectiveness in therapeutic proton beams. *Physics in Medicine and Biology*, 49(13), pp.2811–2825.