How to Download and Use Phantoms

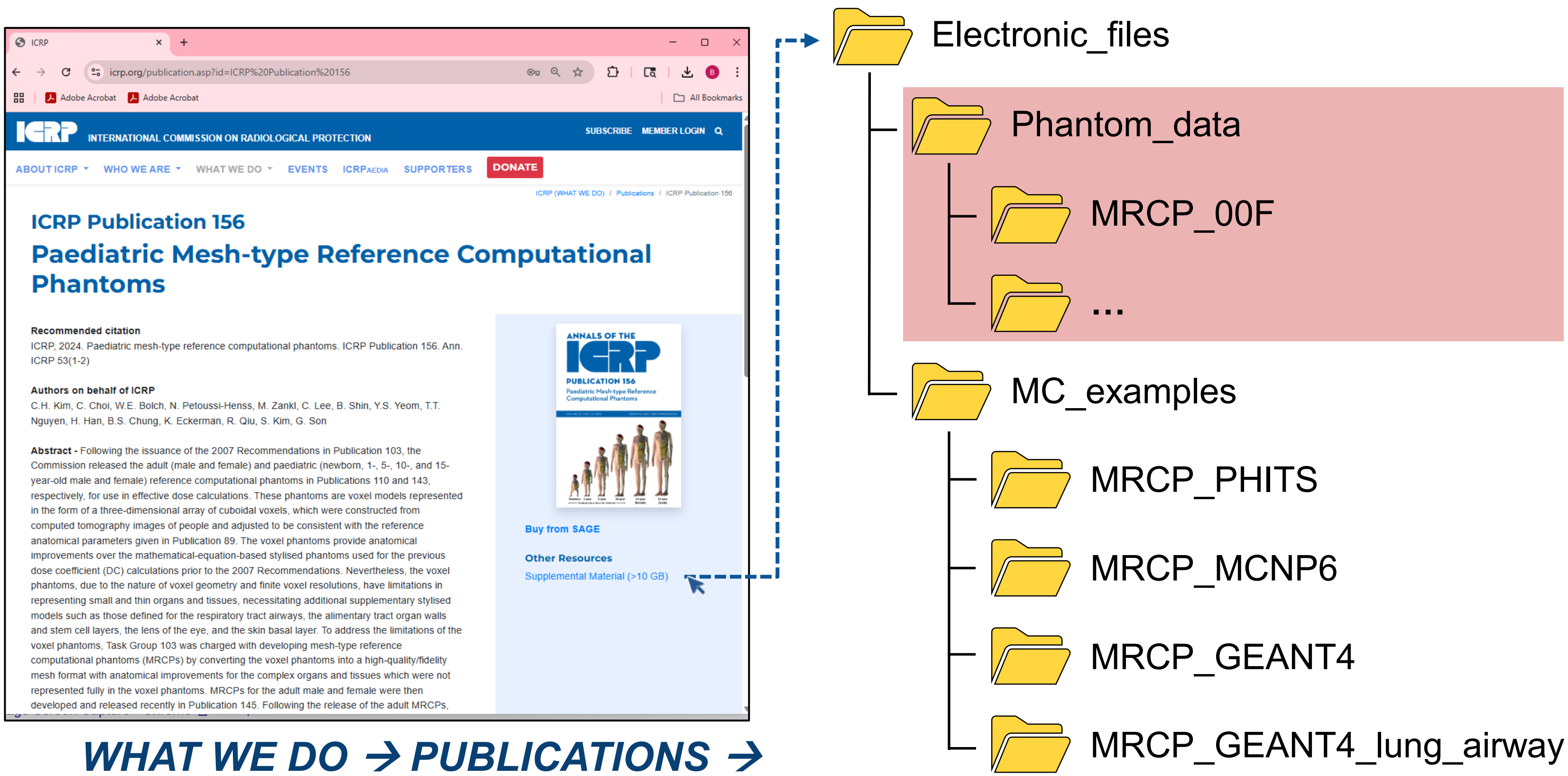
Bangho Shin, Ph.D.

Post-Doctoral Associate, University of Florida
Member, ICRP Task Group 103

ICRP Publication 156 Webinar, 6 August 2025

Objectives

- The Commission provides supplemental electronic files with *Publication 156* to provide users with the phantoms and to help users use the phantoms.
- The presentation will cover **a detailed description on these electronic files** and **useful information for their applications**.



WHAT WE DO → PUBLICATIONS → ICRP Publication 156

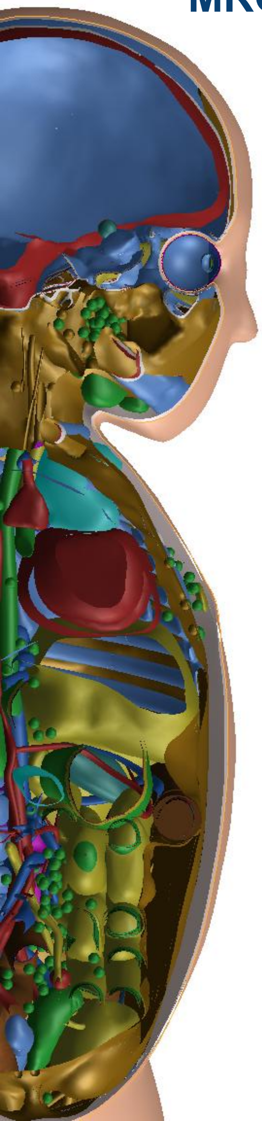
Phantom Data

- Phantom_data
  - MRCP_00F
    - MRCP_00F.ele
    - MRCP_00F.node
    - MRCP_00F.obj
    - MRCP_00F.mtl
    - MRCP_00F_media.dat
    - MRCP_00F_bone.dat
    - MRCP_00F_blood.dat
    - MRCP_00F.pdf
  - … *

** MRCP_00M, 01MF, 05MF, 10MF, and 15MF*
Structure: same as MRCP_00F

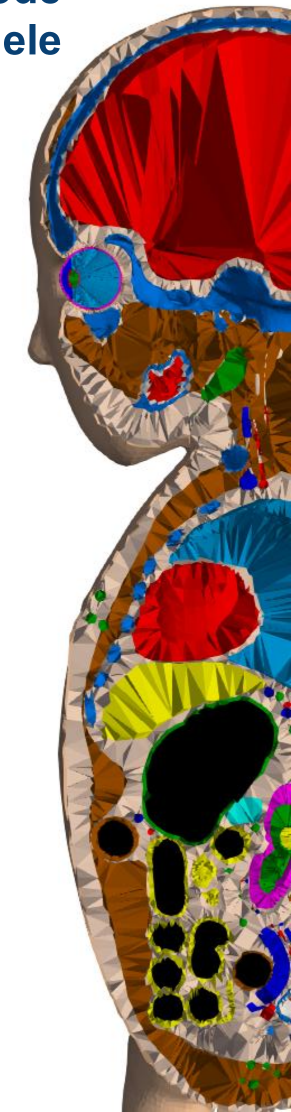
Tetrahedral Mesh Phantom

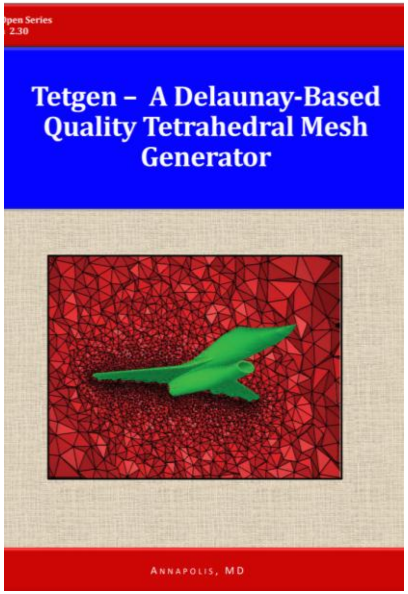
MRCP_00F.obj

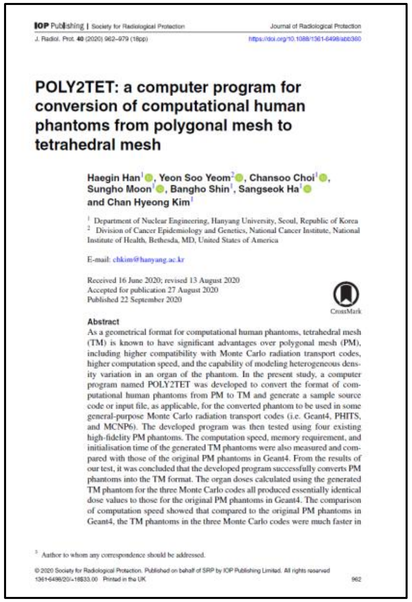


Tetrahedralization

MRCP_00F.node
MRCP_00F.ele







IOP Publishing | Society for Radiological Protection — Journal of Radiological Protection
J. Radiol. Prot. 40 (2020) 962–979 (18pp) https://doi.org/10.1088/1361-6498/abb360

POLY2TET: a computer program for conversion of computational human phantoms from polygonal mesh to tetrahedral mesh

Haegin Han[1], Yeon Soo Yeom[2], Chansoo Choi[1], Sungho Moon[1], Bangho Shin[1], Sangseok Ha[1] and Chan Hyeong Kim[1]

[1] Department of Nuclear Engineering, Hanyang University, Seoul, Republic of Korea
[2] Division of Cancer Epidemiology and Genetics, National Cancer Institute, National Institute of Health, Bethesda, MD, United States of America

E-mail: chkim@hanyang.ac.kr

Received 16 June 2020; revised 13 August 2020
Accepted for publication 27 August 2020
Published 22 September 2020

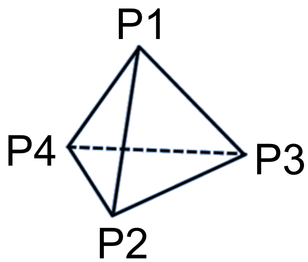
CrossMark

Abstract
As a geometrical format for computational human phantoms, tetrahedral mesh (TM) is known to have significant advantages over polygonal mesh (PM), including higher compatibility with Monte Carlo radiation transport codes, higher computation speed, and the capability of modeling heterogeneous density variation in an organ of the phantom. In the present study, a computer program named POLY2TET was developed to convert the format of computational human phantoms from PM to TM and generate a sample source code or input file, as applicable, for the converted phantom to be used in some general-purpose Monte Carlo radiation transport codes (i.e. Geant4, PHITS, and MCNP6). The developed program was then tested using four existing high-fidelity PM phantoms. The computation speed, memory requirement, and initialisation time of the generated TM phantoms were also measured and compared with those of the original PM phantoms in Geant4. From the results of our test, it was concluded that the developed program successfully converts PM phantoms into the TM format. The organ doses calculated using the generated TM phantom for the three Monte Carlo codes all produced essentially identical dose values to those for the original PM phantoms in Geant4. The comparison of computation speed showed that compared to the original PM phantoms in Geant4, the TM phantoms in the three Monte Carlo codes were much faster in

[3] Author to whom any correspondence should be addressed.

1361-6498/20/+18$33.00 Printed in the UK

MRCP_00F.node and MRCP_00F.ele



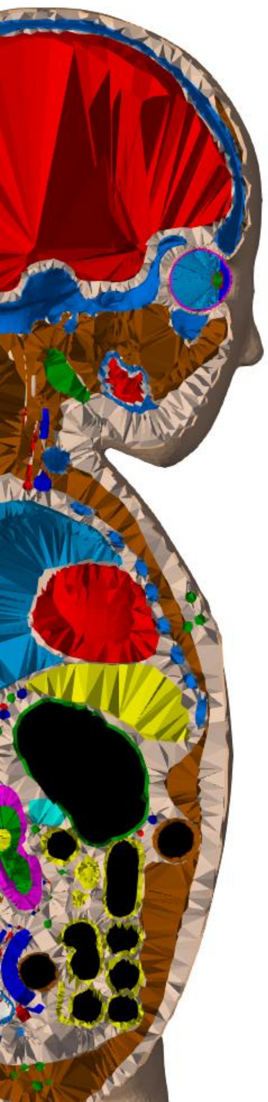
Tetrahedron is defined by …

4 vertices
(X_{P1}, Y_{P1}, Z_{P1}) (X_{P3}, Y_{P3}, Z_{P3})
(X_{P2}, Y_{P2}, Z_{P2}) (X_{P4}, Y_{P4}, Z_{P4})

MRCP_00F.node

4 facets
(P1, P2, P3) (P1, P3, P4)
(P1, P4, P2) (P2, P4, P3)

MRCP_00F.ele



- MRCP_00F.node
  - First line

1156098	3	0	0
<# of nodes>	<dimension>	<n/a>	<n/a>

  - Remaining line list # of points

0	0.44725999999999999	2.9220440000000001	1.088096
1	0.43080000000000002	2.8615390000000001	1.2258279999999999
<node ID>	<x>	<y>	<z>

- MRCP_00F.ele
  - First line

7651951	4	1
<# of tet>	<nodes per tet>	<# of attributes (for organ ID)>

  - Remaining line list # of points

0	257655	248109	248122	248114	3700
1	476202	476203	479479	382323	5800
<tet ID>	<node 1>	<node 2>	<node 3>	<node 4>	<organ ID>

MRCP_00F_media.dat (Copy of Table B.2 in Annex B)

- MRCP_00F_media.dat contains the **physical property information** (elemental compositions and densities) **on each tissue**.

		1	6	7	8	11	12	15	16	17	19	20	26	53	
	Tissues to be used in the Monte Carlo simulation:	H	C	N	O	Na	Mg	P	S	Cl	K	Ca	Fe	I	density
No.							(% by mass)								(g/cm3)
1	Adrenal	10.5	15.8	2.4	71.2	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	1.028
2	ET, Trachea, BB, bb	10.1	13.1	2.3	72.6	0.2	0.0	1.0	0.5	0.2	0.0	0.0	0.0	0.0	1.061
3	Oral mucosa, Tongue	10.4	10.5	2.5	75.9	0.1	0.0	0.1	0.1	0.2	0.2	0.0	0.0	0.0	1.052
4	Blood	10.0	13.1	4.0	72.0	0.1	0.0	0.1	0.2	0.2	0.2	0.0	0.1	0.0	1.070
5	Cortical bone	5.3	15.8	4.2	53.9	0.0	0.3	6.6	0.3	0.0	0.0	13.6	0.0	0.0	1.542
6	Humeri, upper, spongiosa	8.1	24.9	3.8	53.5	0.0	0.1	3.0	0.2	0.1	0.1	6.1	0.1	0.0	1.265
7	Humeri, upper, medullary cavity	10.4	33.3	3.4	52.1	0.0	0.0	0.1	0.2	0.2	0.2	0.0	0.1	0.0	1.037
8	Humeri, lower, spongiosa	8.1	24.9	3.8	53.5	0.0	0.1	3.0	0.2	0.1	0.1	6.1	0.1	0.0	1.265
	...		⋮												
73	Water	11.2	0.0	0.0	88.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

100%

Table A.2. List of organ identification (ID) numbers, medium, and mass of organs/tissues for the polygon mesh version of the newborn, 1-year-old, 5-year-old, 10-year-old, and 15-year-old male and female phantoms.

			Mass (g)									
			Newborn		1-year-old		5-year-old		10-year-old		15-year-old	
Organ ID	Organ/tissue	Medium	Male	Female	Male	Female	Male	Female	Male	Female	Male	Female
100	Adrenal, left	1	3.601	3.601	2.256	2.256	2.974	2.975	4.180	4.180	6.215	5.236
200	Adrenal, right	1	3.601	3.601	2.256	2.256	2.974	2.975	4.180	4.180	6.215	5.236
300	ET_1, 8 µm	2	0.000741	0.000738	0.00124	0.00125	0.00268	0.00259	0.00260	0.00259	0.0106	0.00194
301	ET_1, 40 µm	2	0.00304	0.00303	0.00504	0.00510	0.01088	0.01049	0.0105	0.0105	0.0425	0.00790
302	ET_1, 50 µm	2	0.000976	0.000972	0.00160	0.00163	0.00345	0.00333	0.00334	0.00332	0.0134	0.00251
303	ET_1, surface	2	0.0968	0.0954	0.218	0.212	0.316	0.330	0.667	0.664	4.145	0.729
400	ET_2, 0 µm	73	0.0327	0.0330	0.0692	0.0693	0.127	0.129	0.138	0.136	0.200	0.150

MRCP_00F_blood.dat

- MRCP_00F_blood.dat contains the **information on the fraction of blood content in the media**.

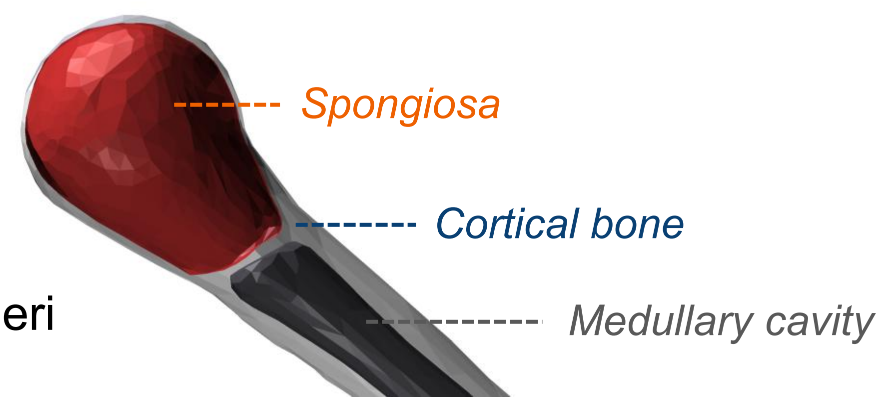
Med	Ratio	
1	0.167	**16.7% of blood content** in *Adrenal*
2	0.078	**7.8% of blood content** in *ET, Trachea, BB, bb*
3	0.078	**7.8% of blood content** in *Oral mucosa, Tongue*
4	1.000	**100.0% of blood content** in *Blood*
5	0.051	**5.1% of blood content** in *Cortical bone*
6	0.142	**14.2% of blood content** in *Humeri, upper, spongiosa*
7	0.212	**21.2% of blood content** in *Humeri, upper, medullary cavity*
8	0.142	**14.2% of blood content** in *Humeri, lower, spongiosa*
	…	⋮
73	0.000	**0.0% of blood content** in *Water*

MRCP_00F_bone.dat

- MRCP_00F_bone.dat contains the **information on the fraction of bone constituents in the bone sites**.

Reb bone marrow (RBM), yellow bone marrow (YBM), trabecular bone (TB), Cortical bone (CB), and miscellaneous skeletal tissue (MST)

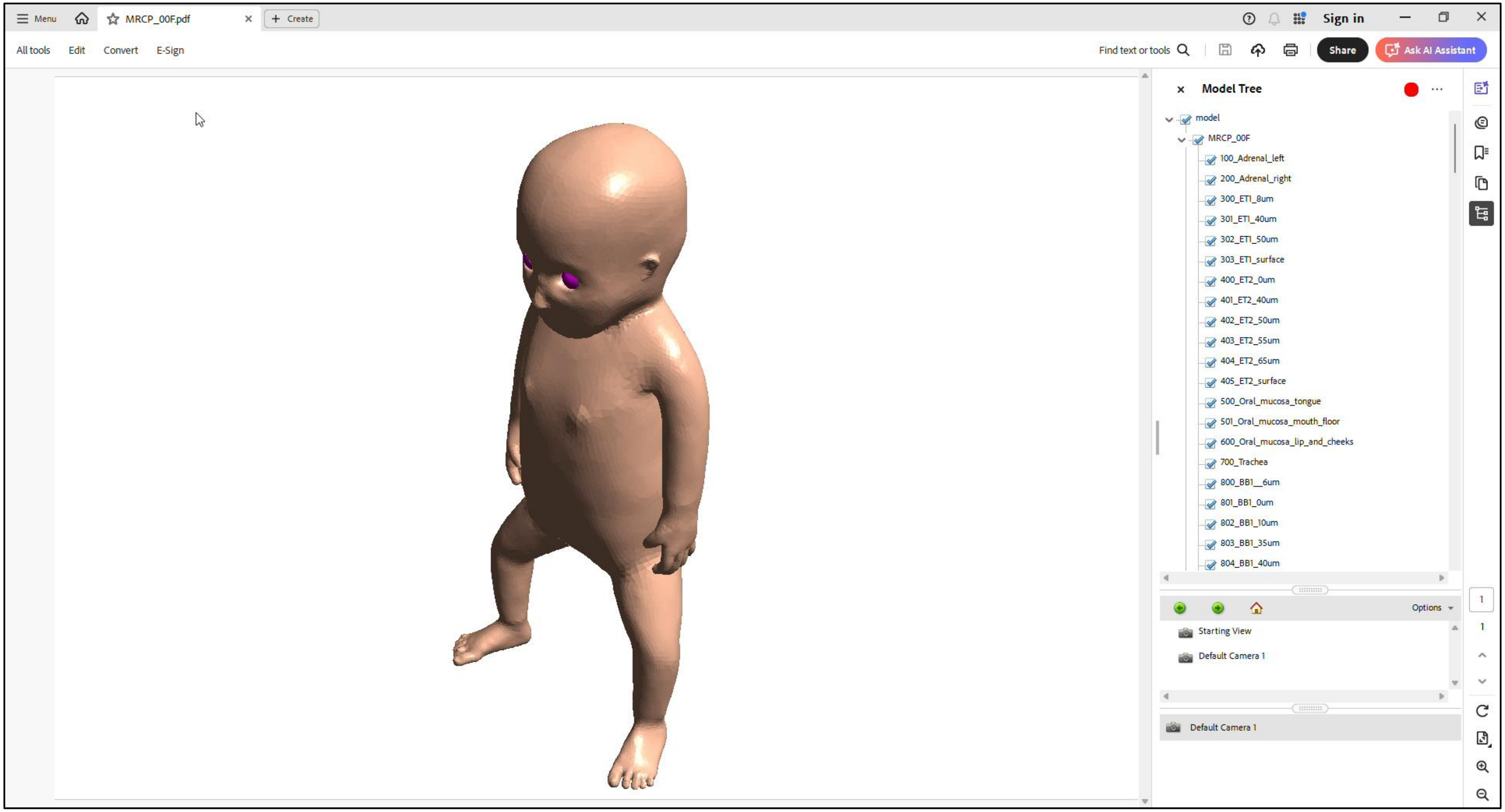
Organ ID	Organ/tissue	Organ/tissue mass exclusive of blood content (ratio)					Organ/tissue mass inclusive of blood content (ratio)				
		RBM	YBM	TB	CB	MST	RBM	YBM	TB	CB	MST
1300	Humeri, upper, cortical	0.000	0.000	0.000	0.926	0.074	0.000	0.000	0.000	0.929	0.071
1400	Humeri, upper, spongiosa	0.348	0.000	0.562	0.000	0.090	0.389	0.000	0.533	0.000	0.078
1500	Humeri, upper, medullary cavity	0.886	0.000	0.000	0.000	0.114	0.909	0.000	0.000	0.000	0.091
1600	Humeri, lower, cortical	0.000	0.000	0.000	0.926	0.074	0.000	0.000	0.000	0.929	0.071
1700	Humeri, lower, spongiosa	0.349	0.000	0.561	0.000	0.090	0.390	0.000	0.532	0.000	0.078
1800	Humeri, lower, medullary cavity	0.886	0.000	0.000	0.000	0.114	0.909	0.000	0.000	0.000	0.091
1900	Radii, cortical	0.000	0.000	0.000	0.926	0.074	0.000	0.000	0.000	0.929	0.071
1910	Ulnae, spongiosa	0.000	0.000	0.000	0.926	0.074	0.000	0.000	0.000	0.929	0.071
	...										
5600	Sternum, spongiosa	0.446	0.000	0.460	0.000	0.094	0.490	0.000	0.429	0.000	0.081
		1.000					1.000				



Upper humeri

MRCP_00F.pdf

- MRCP_00F.pdf visualize the phantoms in a 3D view and **can be opened in Acrobat** (Adobe Systems, San Jose, CA, USA).
- Detailed instruction on these 3D PDF files can be found on the following website: https://helpx.adobe.com/acrobat/using/displaying-3d-models-pdfs.html.



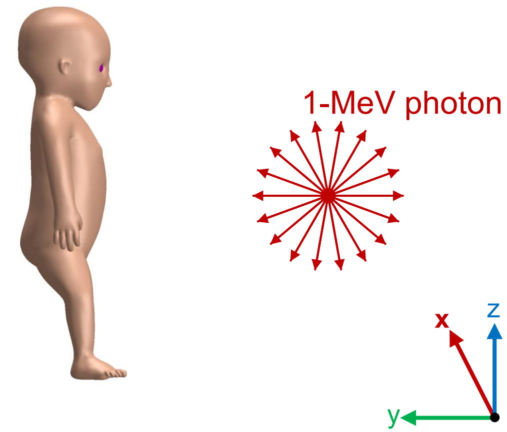
Monte Carlo Code Input Examples

- MC_examples
  - MRCP_PHITS
    - External
    - Internal
    - phantoms
  - MRCP_MCNP6
    - External
    - Internal
    - phantoms
  - MRCP_GEANT4
    - External
    - Internal
    - phantoms
  - MRCP_GEANT4_lung_airway
    - External
    - Internal
    - phantoms

Example Exposure Scenarios

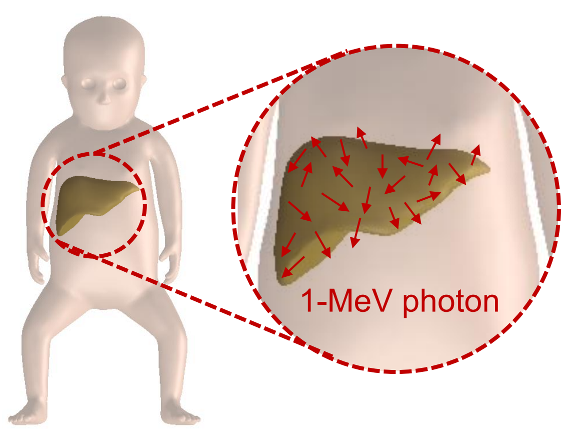
Common **external** exposure scenario

- Particle: photon
- Energy: 1 MeV
- Type and position: point source located 1 m in front of the center of the phantom
- Direction: isotropic
- Result: average absorbed dose for each ID



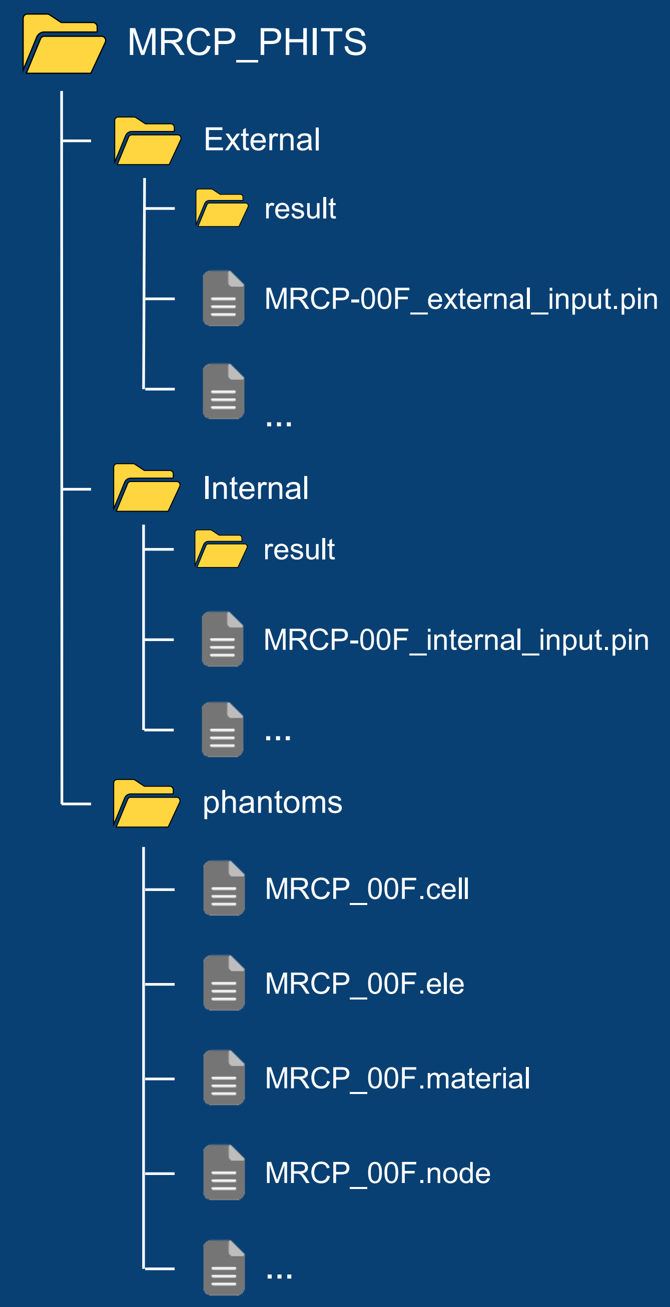
Common **internal** exposure scenario

- Particle: photon
- Energy: 1 MeV
- Type and position: homogeneous liver source
- Direction: isotropic
- Result: average absorbed dose for each ID



Monte Carlo Code Input Examples

PHITS Code



PHITS Code Input File

```
file=MRCP-00F_external_input.pin
# ---------------------------------------------------------------------------
#     name                    : NEWBORN MRCP PHANTOM
#     sex                     : FEMALE
#     author                  : HUREL
# ---------------------------------------------------------------------------

[ Parameters ]
 icntl     =           0     # (D=0) 3:ECH 5:NOR 6:SRC 7,8:GSH 11:DSH 12:DUMP
 maxcas    =     1000000     # (D=10) number of particles per one batch
 maxbch    =          10     # (D=10) number of batches
 itetvol   =           0     # (D=0) =1 Volume calculation for tetrahedrons
 ntetelem =         2000     # Number of tetra element allowed in sub-block
 maxbnk    =     1000000     # (D=10000) maximum bank memory length
 itetra    =           0     # (D=0) tetra data is read(=1)/write(=2) on binary
 file(6)   = MRCP-00F_external.out # (D=phits.out) general output file name
 file(7)   = (path to 'xsdir.jnd' file) # nuclear data input file name
 file(20)  = (path to egs folder)       # Directory library data for EGS5
 negs      =           1     # (D=-1) 1:EGS
 ipegs     =           0     # [EGS](D=0) 0:Full

[ Source ]
  totfact =     1.00         # (D=1.0) global factor
   s-type =     1            # cylindrical source
     proj =    photon        # kind of incident particle
       e0 =     1.00         # energy of beam [MeV/u]
       r0 =     0.00         # radius [cm]
       x0 =     0.00         # (D=0.0) center position of x-axis [cm]
       y0 =  -100.00         # (D=0.0) center position of y-axis [cm]
       z0 =     0.00         # minimum position of z-axis [cm]
       z1 =     0.00         # maximum position of z-axis [cm]
      dir =    all           # z-direction of beam [isotropic]
```

PHITS Code Input File (Cont'd)

```
[ Material ]
$ ------------------------------------------------------------------------------
$ MATERIAL DATA FOR EACH ORGAN/TISSUE
$ ------------------------------------------------------------------------------
infl:{../phantoms/MRCP-00F.material}

[ Surface ]
 10  rpp -21.42163 21.42163 -17.07828 17.07828 -34.41986 34.41986 $ Phantom box
 20  rpp -21.62163 21.62163 -17.27828 17.27828 -34.61986 34.61986
 90  box -1000 -1000 -1000 2000 0 0 0 2000 0 0 0 2000          $ World boundary

[ Cell ]
$ ------------------------------------------------------------------------------
$ CELLS FOR PHANTOM
$ ------------------------------------------------------------------------------
infl:{../phantoms/MRCP-00F.cell}

[ T-deposit ]
# Deposit energy in a certain region
 file    = Result_MRCP-00F_external.out       # file name of output for the axis
 mesh    = reg                  # mesh type is region-wise
 reg     = all                  # all regions become tallying region
 axis    = reg                  # axis of output
 unit    = 0                    # unit is [Gy/source]

[ End ]
```

[Parameters] Section

```
[ Parameters ]
 icntl    =          0      # (D=0) 3:ECH 5:NOR 6:SRC 7,8:GSH 11:DSH 12:DUMP
 maxcas   =    1000000      # (D=10) number of particles per one batch
 maxbch   =         10      # (D=10) number of batches
 itetvol  =          0      # (D=0) =1 Volume calculation for tetrahedrons
 ntetelem =       2000      # Number of tetra element allowed in sub-block
 maxbnk   =    1000000      # (D=10000) maximum bank memory length
 itetra   =          0      # (D=0) tetra data is read(=1)/write(=2) on binary
 file(6)  = MRCP-00F_external.out # (D=phits.out) general output file name
 file(7)  = (path to 'xsdir.jnd' file) # nuclear data input file name
 file(20) = (path to egs folder)       # Directory library data for EGS5
 negs     =          1      # (D=-1) 1:EGS
 ipegs    =          0      # [EGS](D=0) 0:Full
```

- When *icntl* is set to 0, normal PHITS calculation mode is performed.
- *maxcas* is number of histories per batch.
- *maxbch* is number of batches.
- When *negs* is set to 1, electrons, positrons, and photons are transported based on the EGS5 algorithm.

NPS = maxcax * maxbch

[Parameters] Section – Set PHITS Path

- **Before PHITS version 3.00** – set the file(7) and file(20).

```
[ Parameters ]
 icntl    =           0     # (D=0) 3:ECH 5:NOR 6:SRC 7,8:GSH 11:DSH 12:DUMP
 maxcas   =     1000000     # (D=10) number of particles per one batch
 maxbch   =          10     # (D=10) number of batches
 itetvol  =           0     # (D=0) =1 Volume calculation for tetrahedrons
 ntetelem =        2000     # Number of tetra element allowed in sub-block
 maxbnk   =     1000000     # (D=10000) maximum bank memory length
 itetra   =           0     # (D=0) tetra data is read(=1)/write(=2) on binary
 file(6)  = MRCP-00F_external.out # (D=phits.out) general output file name
 file(7)  = (path to 'xsdir.jnd' file) # nuclear data input file name
 file(20) = (path to egs folder)       # Directory library data for EGS5
 negs     =           1     # (D=-1) 1:EGS
 ipegs    =           0     # [EGS](D=0) 0:Full
```

```
RIsource.rad                                  ion_potential.dat            xsdir.adj
Relaxation-NonRadiative.dat                   material.inp                 xsdir.jnd
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341/data$ pwd
/home/shinbangho/PHITS_install/PHITS3341/data
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341/data$
```

Nuclear data input file

Path to “xsdir.jnd” file

ex) file(7) = /home/shinbangho/PHITS_install/PHITS3341/data

```
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341/XS/egs$ pwd
/home/shinbangho/PHITS_install/PHITS3341/XS/egs
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341/XS/egs$ ls
K1.dat       bcomp.dat  density_corrections  int_coherent_cs   pgs5form.dat  shellwise_Compton_profile
aprime.data  dcslib     incoh.dat            int_form_factor   pgs5phtx.dat
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341/XS/egs$
```

Path to egs folder

Files in egs folder

ex) file(7) = /home/shinbangho/PHITS_install/PHITS3341/XS/egs

[Parameters] Section – Set PHITS Path (Cont'd)

- **From PHITS version 3.00** – set the file(1).

```
[ Parameters ]
 icntl    =          0      # (D=0) 3:ECH 5:NOR 6:SRC 7,8:GSH 11:DSH 12:DUMP
 maxcas   =    1000000      # (D=10) number of particles per one batch
 maxbch   =         10      # (D=10) number of batches
 itetvol  =          0      # (D=0) =1 Volume calculation for tetrahedrons
 ntetelem =       2000      # Number of tetra element allowed in sub-block
 maxbnk   =    1000000      # (D=10000) maximum bank memory length
 itetra   =          0      # (D=0) tetra data is read(=1)/write(=2) on binary
 file(6)  = MRCP-00F_external.out # (D=phits.out) general output file name
   file (1) = (PHITS installation folder path)
 negs     =          1      # (D=-1) 1:EGS
 ipegs    =          0      # [EGS](D=0) 0:Full
```

```
XS   data      document  manual  phits_LinGfort_OMP  phitspad        sample  utility
bin  dchain-sp lecture   phig3d  phits_LinIfort_OMP  recommendation  src
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341$ pwd
/home/shinbangho/PHITS_install/PHITS3341
shinbangho@BME-BOL-27X4144:~/PHITS_install/PHITS3341$
```

PHITS installation folder path

ex) file(1) = /home/shinbangho/PHITS_install/PHITS3341

- If the file(1) is set, file(7) and file(20) are automatically determined.
  - ✓ file(7) : "file(1)/data/xsdir.jnd"
  - ✓ File(20) : "file(1)/XS/egs"

[Material] Section

- Define material information from the **"*.material" file in the "phantoms" folder**.

```
[ Material ]
$ ------------------------------------------------------------------------------
$ MATERIAL DATA FOR EACH ORGAN/TISSUE
$ ------------------------------------------------------------------------------
infl:{../phantoms/MRCP-00F.material}
```

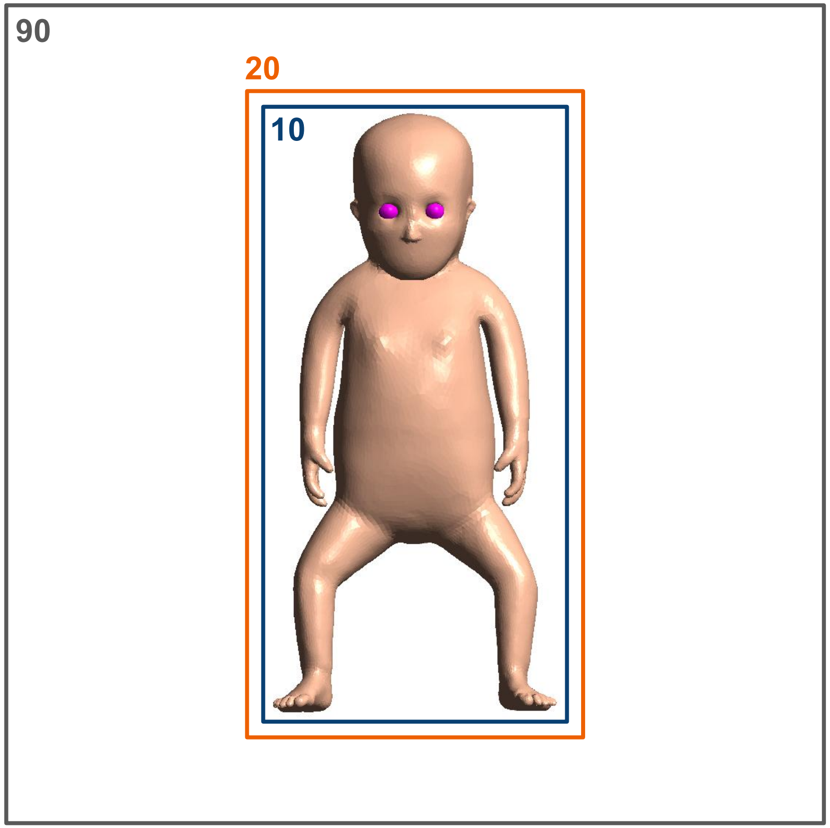
```
$ MATERIALS FOR EACH ORGAN/TISSUE
$ Adrenal_left 1.0280 g/cm3
MAT[100]  <mat. no.>
        1000      -0.1050
        6000      -0.1580
        7000      -0.0240
        8000      -0.7120
        16000     -0.0010
mt100
$ Adrenal_right 1.0280 g/cm3
MAT[200]  <mat. no.>
        1000      -0.1050
        6000      -0.1580
        7000      -0.0240
        8000      -0.7120
        16000     -0.0010
mt200
```

[Surface] Section

- Define **three surfaces** in the [Surface] section.

```
[ Surface ]
10  rpp -21.42163 21.42163 -17.07828 17.07828 -34.41986 34.41986 $ Phantom box
20  rpp -21.62163 21.62163 -17.27828 17.27828 -34.61986 34.61986
90  box -1000 -1000 -1000 2000 0 0 0 2000 0 0 0 2000          $ World boundary
```

- Phantom box (10)
- Box for Universe and Fill (20)
  - ✓ A slightly larger box than the phantom box
- World (90)



[Cell] Section

- Define cell information from the **"*.cell" file in the "phantoms" folder.**

```
[ Cell ]
$ ------------------------------------------------------------------------------
$ CELLS FOR PHANTOM
$ ------------------------------------------------------------------------------
infl:{../phantoms/MRCP-00F.cell}
```

Cell no.1 is a universe (U=15000)
Cell no.2 is a component of the main space
Fill the component in main space by using the universe (FILL=15000)

```
$ CELLS FOR PHANTOM
 00001          0         -20              U=15000 LAT=3 tfile=../phantoms/MRCP-00F
 00002          0         -10              FILL=15000
 00003          0         -20      10
 00004          0         -90      20
 00005         -1          90
   100    100  -1.0280  -90  u=100         VOL=3.5033032874
   200    200  -1.0280  -90  u=200         VOL=3.5033046105
   300    300  -1.0610  -90  u=300         VOL=0.0006959394
   301    301  -1.0610  -90  u=301         VOL=0.0028555911
   302    302  -1.0610  -90  u=302         VOL=0.0009161311
   303    303  -1.0610  -90  u=303         VOL=0.0898854358
   400    400  -1.0000  -90  u=400         VOL=0.0330015026
   401    401  -1.0610  -90  u=401         VOL=0.0890227302
   402    402  -1.0610  -90  u=402         VOL=0.0224873722
   403    403  -1.0610  -90  u=403         VOL=0.0112789081
   404    404  -1.0610  -90  u=404         VOL=0.0226266147
   405    405  -1.0610  -90  u=405         VOL=1.5878587176
   500    500  -1.0520  -90  u=500         VOL=0.0077053053
   501    501  -1.0520  -90  u=501         VOL=0.0093137322
```

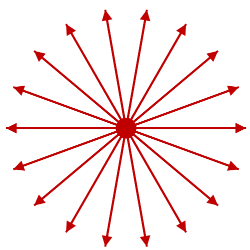
<cell no.><mat. no.> <density> <univ. no.>

'LAT=3' means that the geometry by the node and element files with names specified by TFILE is used.

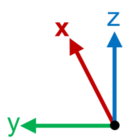
[Source] Section – External Exposure

- Set the external source.

```
[ Source ]
  totfact =      1.00          # (D=1.0) global factor
   s-type =      1             # cylindrical source
     proj =    photon          # kind of incident particle
       e0 =      1.00          # energy of beam [MeV/u]
       r0 =      0.00          # radius [cm]
       x0 =      0.00          # (D=0.0) center position of x-axis [cm]
       y0 =   -100.00          # (D=0.0) center position of y-axis [cm]
       z0 =      0.00          # minimum position of z-axis [cm]
       z1 =      0.00          # maximum position of z-axis [cm]
      dir =    all             # z-direction of beam [isotropic]
```



- **Type: point source**
- **Particle: photon**
- **Energy: 1 MeV**
- **Position: (0, -100, 0)**
- **Direction: isotropic**



[Source] Section – Internal Exposure

- **Before PHITS version 3.02** – use the **"s-type = 2" (rectangular solid)** to generate the source particles using the **rejection method**.

```
[ Source ]
  totfact =    1.00          # (D=1.0) global factor
   s-type =    2             # rectangular-solid source
     proj =   photon         # kind of incident particle
       e0 =    1.00          # energy of beam [MeV/u]
       x0 =   -5.70          # minimum position of x-axis [cm]
       x1 =    4.30          # maximum position of x-axis [cm]
       y0 =   -3.30          # minimum position of y-axis [cm]
       y1 =    4.30          # maximum position of y-axis [cm]
       z0 =   -1.50          # minimum position of z-axis [cm]
       z1 =    5.70          # maximum position of z-axis [cm]
      dir =   all            # z-direction of beam [isotropic]
 # region selection of source
      reg =    9500
    ntmax =    1000          # (D=1000) maximum trial number of region selection
```

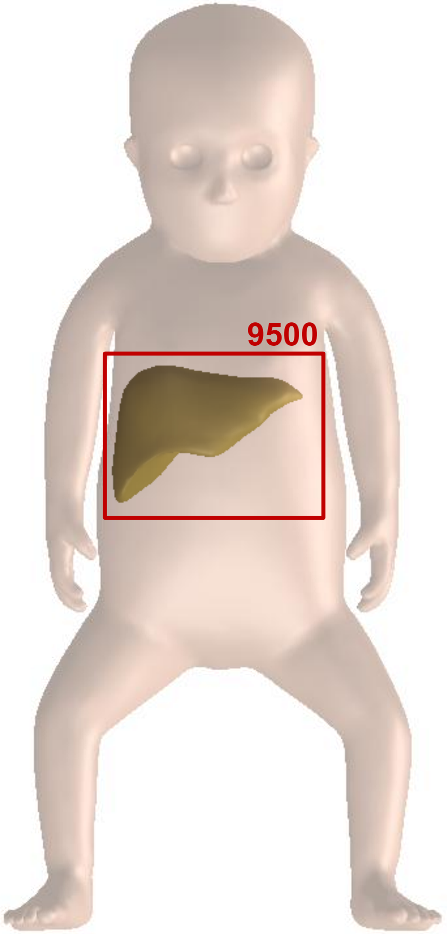
<box size information>

<organ/cell number>

- **From PHITS version 3.02** – use the **"s-type = 24" (tetra-mesh source)** to generate the source particles **directly from the tetrahedrons**.

```
[ Source ]
  totfact =    1.00          # (D=1.0) global factor
   s-type =    24            # rectangular-solid source
     proj =   photon         # kind of incident particle
       e0 =    1.00          # energy of beam [MeV/u]
      dir =    all          # (D=1000) maximum trial number of region selection
   tetreg =    9500
```

<organ/cell number>



[T-deposit] Section

- Use the [T-deposit] section to calculate the **energy deposition (heat) of organs**.

```
[ T-deposit ]
# Deposit energy in a certain region
 file    = Result_MRCP-00F_external.out       # file name of output for the axis
 mesh    = reg                   # mesh type is region-wise
 reg     = all                   # all regions become tallying region
 axis    = reg                   # axis of output
 unit    = 0                     # unit is [Gy/source]
```

- Set the geometry option by the "mesh" parameters.
  - ✓ reg: region mesh (cell number)
- Set the specific region for calculating energy deposition by "reg" parameter.
  - ✓ all: all cell regions (calculation of the energy deposition for all the organs separately)
- If the user wants to calculate the energy deposition for the **specific organ**, **enter the cell number**.

  ex) reg = 100

- If the user wants to calculate the energy deposition for the **several organs together**, **enter their cell number in parentheses.**

  ex) reg = (9700 9900)

Tip for Reducing Implementation Time

- Phantom implementation time can be reduced by **generating and reading the tetrahedron geometry data file in binary format (“Tetra.bin”) file**.

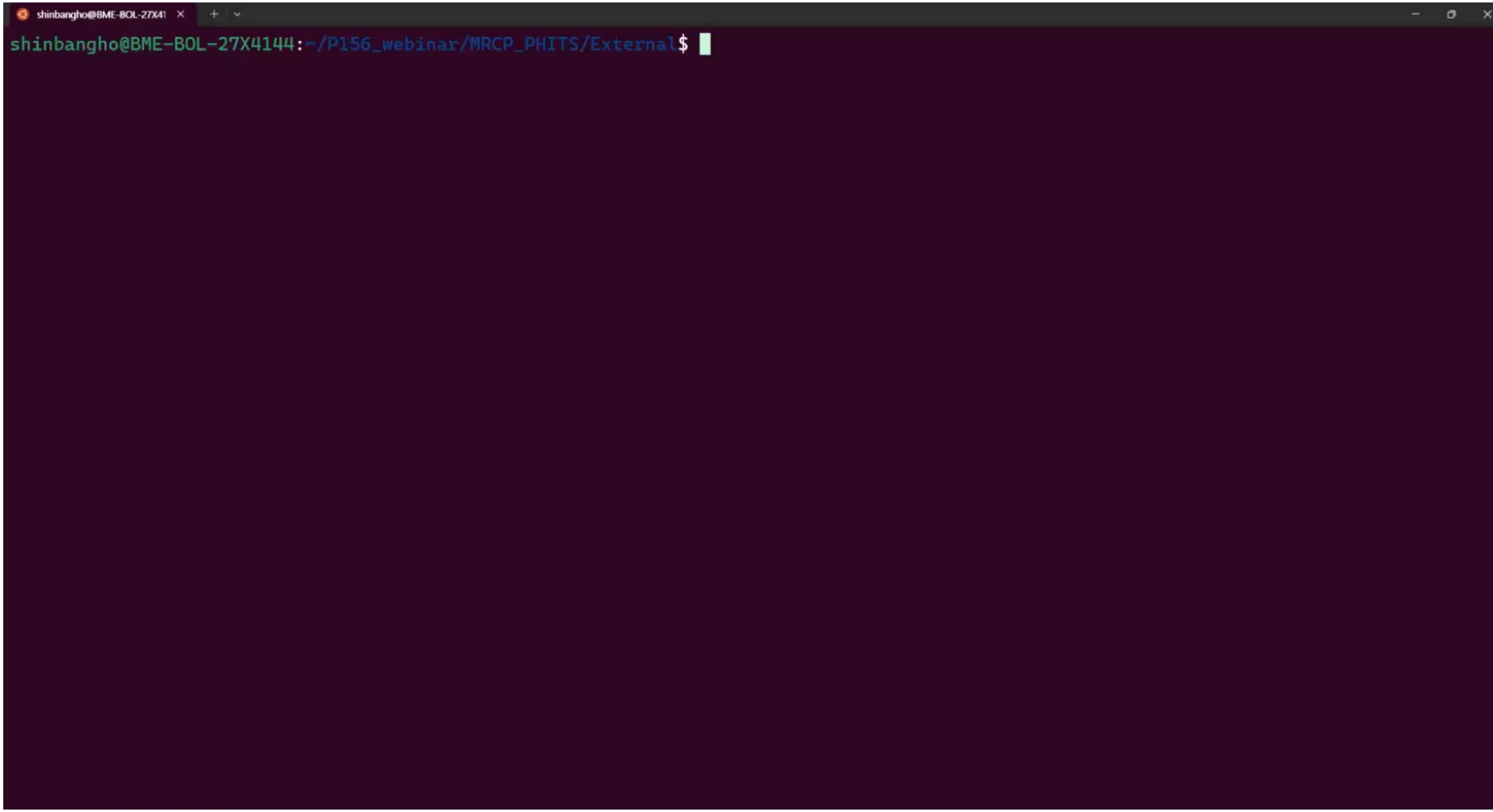
```
[ Parameters ]
 icntl     =           0     # (D=0) 3:ECH 5:NOR 6:SRC 7,8:GSH 11:DSH 12:DUMP
 maxcas    =     1000000     # (D=10) number of particles per one batch
 maxbch    =          10     # (D=10) number of batches
 itetvol   =           0     # (D=0) =1 Volume calculation for tetrahedrons
 ntetelem =         2000     # Number of tetra element allowed in sub-block
 maxbnk    =     1000000     # (D=10000) maximum bank memory length
 itetra    =           0     # (D=0) tetra data is read(=1)/write(=2) on binary
 file(6)   = MRCP-00F_external.out # (D=phits.out) general output file name
 file(7)   = (path to 'xsdir.jnd' file) # nuclear data input file name
 file(20)  = (path to egs folder)       # Directory library data for EGS5
 negs      =           1     # (D=-1) 1:EGS
 ipegs     =           0     # [EGS](D=0) 0:Full
```

- Itetra = 0 : read the ELE/NODE files; **1: read the binary file; 2: write the binary file**
- The default binary file name is “Tetra.bin”.
- From version 3.35, the user can specify the binary file name and path via “file(30)” parameter.

Phantom file	Implementation time
ELE/NODE file	~4 minutes
Binary file (“Tetra.bin”)	Less than a minute

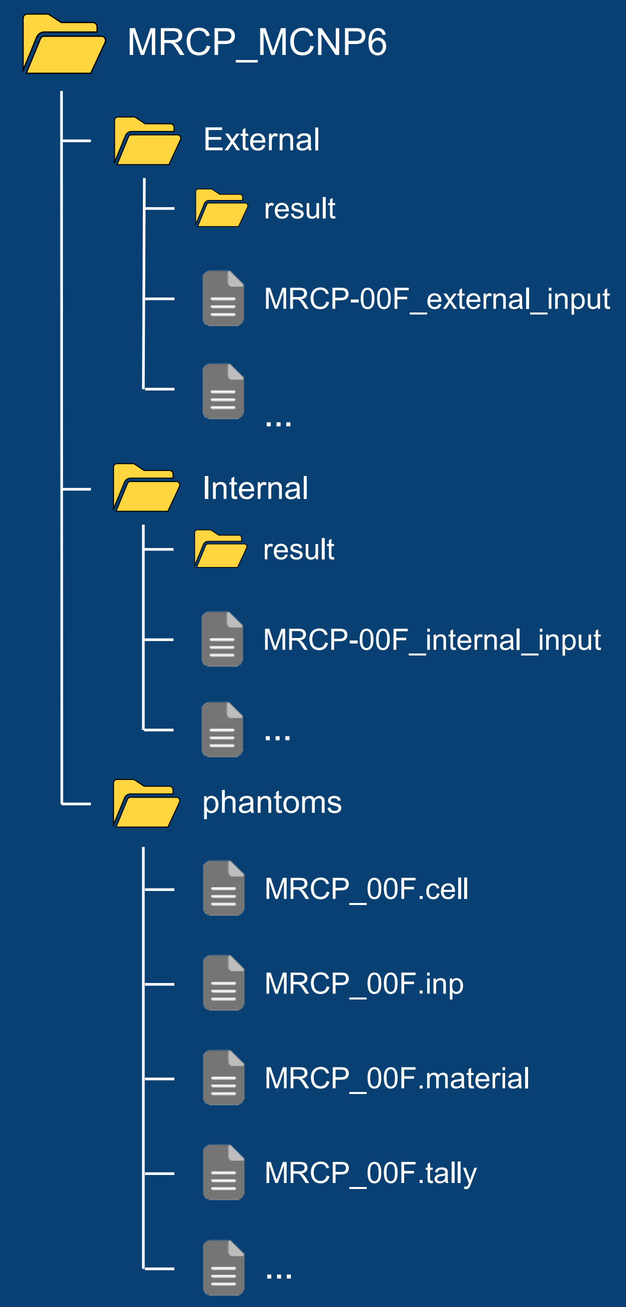
** Time was measured in “Intel® Core(TM) i7-14700K CPU @ 3.40GHz and PHITS ver.3.34 was used.*

Execution of PHITS – Demo Video



Monte Carlo Code Input Examples

MCNP6 Code



MCNP6 Code Input File

```
MRCP FEMALE PHANTOM - EXTERNAL
C ---------------------------------------------------------------------------
C     name                  : NEWBORN MRCP PHANTOM
C     sex                   : FEMALE
C     author                : HUREL
C ---------------------------------------------------------------------------
C ***************************************************************************
C                                CELL CARDS
C ***************************************************************************
C ---------------------------------------------------------------------------
C PSUEDO CELLS FOR ABAQUS
C ---------------------------------------------------------------------------
read file=../phantoms/mrcp-00f.cell
C ---------------------------------------------------------------------------
C LEGACY CELLS
C ---------------------------------------------------------------------------
666  0   -10  fill=2              $ Phantom box
777  0   -99  10                  $ Void in the world
999  0        99                  $ World

C ***************************************************************************
C                             SURFACE CARDS
C ***************************************************************************
10  rpp -21.42163 21.42163 -17.07828 17.07828 -34.41986 34.41986 $ Phantom box
99  box -1000 -1000 -1000 2000 0 0 0 2000 0 0 0 2000             $ World boundary
```

MCNP6 Code Input File (Cont'd)

```
C ***************************************************************************
C                               DATA CARDS
C ***************************************************************************
dbcn 48j 1                        $ Debug information
mode p e                          $ Track photons and electrons
mphys on                          $ Turn on model physics
imp:p,e 1 204r 0                  $ Importance
prdmp 10000000 10000000  -1       $ Print and dump cycle
rand  seed=RNSEED                 $ Random seed
nps 10000000                      $ Number of particles
sdef par=p erg=1 pos=0 -100 0     $ General source definition
C ---------------------------------------------------------------------------
C TALLIES FOR EACH ORGAN/TISSUE
C ---------------------------------------------------------------------------
read file=../phantoms/mrcp-00f.tally
C ---------------------------------------------------------------------------
C MATERIAL DATA FOR EACH ORGAN/TISSUE
C ---------------------------------------------------------------------------
read file=../phantoms/mrcp-00f.material
C ---------------------------------------------------------------------------
C EMBED is required for embedding a mesh geometry into MCNP6 input
embed2 meshgeo=abaqus                         $ Format specification
       mgeoin=../phantoms/mrcp-00f.inp        $ Name of the input file
       background=15000                       $ Cell number of the background
       matcell=     1   100     2   200     3   300     4   301     5   302
                    6   303     7   400     8   401     9   402    10   403
                   11   404    12   405    13   500    14   501    15   600
                   16   700    17   800    18   801    19   802    20   803
                   21   804    22   805    23   806    24   807    25   808
                   26   900    27   910    28  1000    29  1010    30  1100
```

Data Card – Material

- Define material information from the **"*.material" file in the "phantoms" folder**.

```
[ Material ]
$ ---------------------------------------------------------------------------
$ MATERIAL DATA FOR EACH ORGAN/TISSUE
$ ---------------------------------------------------------------------------
infl:{../phantoms/MRCP-00F.material}
```

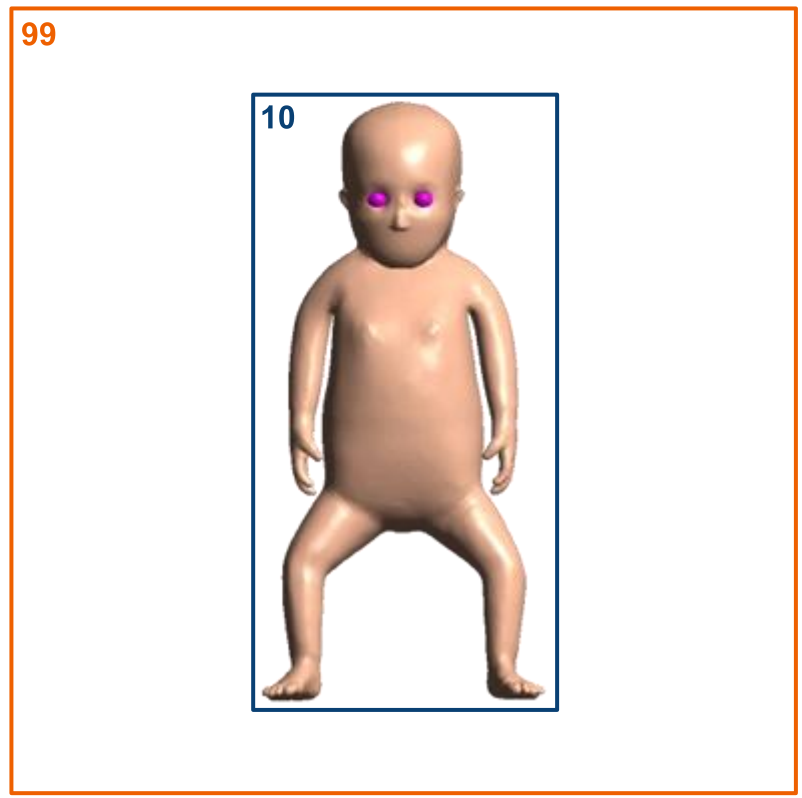
```
C MATERIALS FOR EACH ORGAN/TISSUE
C Adrenal_left 1.0280 g/cm3
m100 <mat. no.>
       1000      -0.1050
       6000      -0.1580
       7000      -0.0240
       8000      -0.7120
       16000     -0.0010
C
C Adrenal_right 1.0280 g/cm3
m200 <mat. no.>
       1000      -0.1050
       6000      -0.1580
       7000      -0.0240
       8000      -0.7120
       16000     -0.0010
C
```

Surface Card

- Define **two surfaces** in the surface card.

```
C ****************************************************************************
C                              SURFACE CARDS
C ****************************************************************************
10   rpp -21.42163 21.42163 -17.07828 17.07828 -34.41986 34.41986 $ Phantom box
99   box -1000 -1000 -1000 2000 0 0 0 2000 0 0 0 2000             $ World boundary
```

- Phantom box (10)
- World (99)



Cell Card

- Define cell information from the **“.cell” file in the “phantoms” folder** and **fill the phantom box with the universe**.

```
C ****************************************************************************
C                                CELL CARDS
C ****************************************************************************
C ----------------------------------------------------------------------------
C PSUEDO CELLS FOR ABAQUS
C ----------------------------------------------------------------------------
read file=../phantoms/mrcp-00f.cell
C ----------------------------------------------------------------------------
C LEGACY CELLS
C ----------------------------------------------------------------------------
666   0    -10   fill=2           $ Phantom box
777   0    -99   10               $ Void in the world
999   0          99               $ World
```

```
C PSUEDO CELLS FOR ABAQUS
     100    100   -1.0280   0   u=2  vol=3.5033032874
     200    200   -1.0280   0   u=2  vol=3.5033046105
     300    300   -1.0610   0   u=2  vol=0.0006959394
     301    301   -1.0610   0   u=2  vol=0.0028555911
     302    302   -1.0610   0   u=2  vol=0.0009161311
     303    303   -1.0610   0   u=2  vol=0.0898854358
```

<pseudo cell no.> <mat. no.> <density> <0 for pseudo cell> <univ. no.>

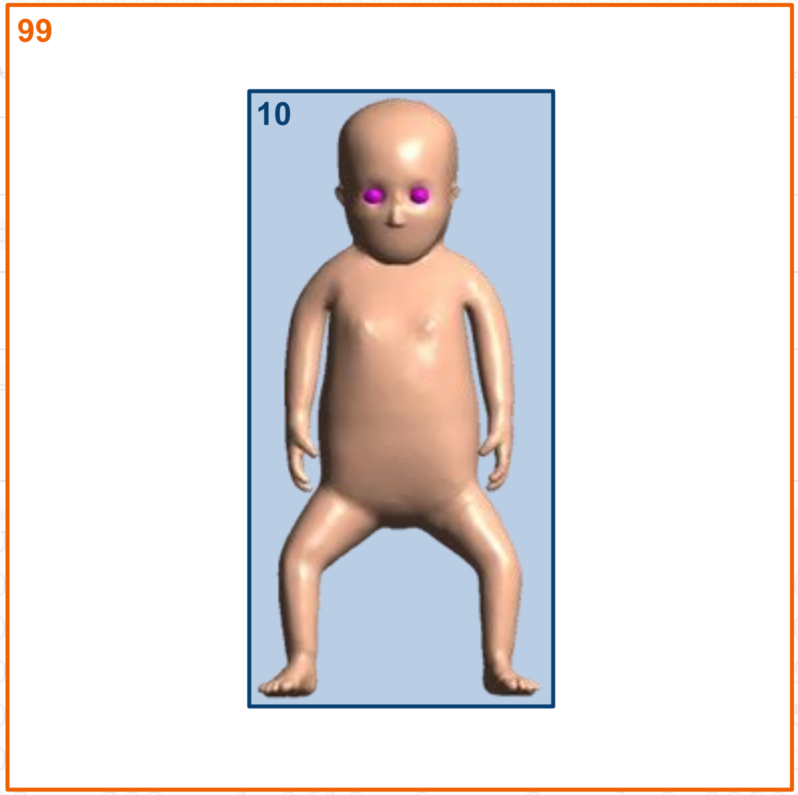
```
   14000  14000   -0.0010   0   u=2  vol=1.9187220487
   15000      0             0   u=2
```

<background>

Cell Card

- Define cell information from the **".cell" file in the "phantoms" folder** and **fill the phantom box with the universe**.

```
C ****************                                        *****************
C
C ****************                                        *****************
C ----------------                                        -----------------
C PSUEDO CELLS FOR
C ----------------                                        -----------------
read file=../phanto
C ----------------                                        -----------------
C LEGACY CELLS
C ----------------                                        -----------------
666   0    -10   fill=
777   0    -99   10
999   0          99
```




```
C PSU
    10                                  032874
    20                                  046105
    30                                  959394
    30                                  555911
    30                                  161311
    303    303   -1.0610   0   u=2 vol=0.0898854358
```

<pseudo cell no.> <mat. no.> <density> <0 for pseudo cell> <univ. no.>

```
 14000 14000   -0.0010   0   u=2 vol=1.9187220487
 15000     0             0   u=2
```

<background>

Cell Card

- Define cell information from the **“.cell” file in the “phantoms” folder** and **fill the phantom box with the universe**.

```
C *****************************************************************************
C                                CELL CARDS
C *****************************************************************************
C -----------------------------------------------------------------------------
C PSUEDO CELLS FOR ABAQUS
C -----------------------------------------------------------------------------
read file=../phantoms/mrcp-00f.cell
C -----------------------------------------------------------------------------
C LEGACY CELLS
C -----------------------------------------------------------------------------
666   0    -10   fill=2            $ Phantom box
777   0    -99   10                $ Void in the world
999   0          99                $ World
```

```
C PSUEDO CELLS FOR ABAQUS
    100    100   -1.0280   0   u=2  vol=3.5033032874
    200    200   -1.0280   0   u=2  vol=3.5033046105
    300    300   -1.0610   0   u=2  vol=0.0006959394
    301    301   -1.0610   0   u=2  vol=0.0028555911
    302    302   -1.0610   0   u=2  vol=0.0009161311
    303    303   -1.0610   0   u=2  vol=0.0898854358
  14000  14000   -0.0010   0   u=2  vol=1.9187220487
  15000      0             0   u=2
```

<pseudo cell no.> <mat. no.> <density> <0 for pseudo cell> <univ. no.> <background>

Data Card – EMBED

- **Implement the phantom in “ABAQUS” file format** using the EMBED.

```
C EMBED is required for embedding a mesh geometry into MCNP6 input
embed2 meshgeo=abaqus                         $ Format specification
       mgeoin=../phantoms/mrcp-00f.inp        $ Name of the input file
       background=15000                       $ Cell number of the background
       matcell=     1   100     2   200     3   300     4   301     5   302
                    6   303     7   400     8   401     9   402    10   403
                   11   404    12   405    13   500    14   501    15   600
                   16   700    17   800    18   801    19   802    20   803
                   21   804    22   805    23   806    24   807    25   808
                   26   900    27   910    28  1000    29  1010    30  1100
                   31  1110    32  1200    33  1210    34  1300    35  1400
                   36  1500    37  1600    38  1700    39  1800    40  1900
```

- EMBED should be written in the data card.
- Number behind “embed” should be same as the universe number of the phantom.
- Set the mesh geometry type using “meshgeo” (ABAQUS, HDF5, or MCNPUM).
- Set the phantom file path using “mgeoin”.

Data Card – EMBED

Journal of
Radiation Protection and Research 2016;41(4):389-394
https://doi.org/10.14407/jrpr.2016.41.4.389

pISSN 2508-1888 | eISSN 2466-2461

TET2MCNP: A Conversion Program to Implement Tetrahedral-mesh Models in MCNP

Min Cheol Han, Yeon Soo Yeom, Thang Tat Nguyen, Chansoo Choi, Hyun Su Lee, Chan Hyeong Kim*

Department of Nuclear Engineering, Hanyang University, Seoul, Korea

ABSTRACT

Background: Tetrahedral-mesh geometries can be used in the MCNP code, but the MCNP code accepts only the geometry in the Abaqus input file format; hence, the existing tetrahedral-mesh models first need to be converted to the Abacus input file format to be used in the MCNP code. In the present study, we developed a simple but useful computer program, TET2MCNP, for converting TetGen-generated tetrahedral-mesh models to the Abacus input file format.

Materials and Methods: TET2MCNP is written in C++ and contains two components: one for converting a TetGen output file to the Abacus input file and the other for the reverse conversion process. The TET2MCP program also produces an MCNP input file. Further, the program provides some MCNP-specific functions: the maximum number of elements (i.e., tetrahedrons) per part can be limited, and the material density of each element can be transferred to the MCNP input file.

Results and Discussion: To test the developed program, two tetrahedral-mesh models were generated using TetGen and converted to the Abaqus input file format using TET2MCNP. Subsequently, the converted files were used in the MCNP code to calculate the object- and organ-averaged absorbed dose in the sphere and phantom, respectively. The results show that the converted models provide, within statistical uncertainties, identical dose values to those obtained using the PHITS code, which uses the original tetrahedral-mesh models produced by the TetGen program. The results show that the developed program can successfully convert TetGen tetrahedral-mesh models to Abacus input files.

Conclusion: In the present study, we have developed a computer program, TET2MCNP, which can be used to convert TetGen-generated tetrahedral-mesh models to the Abaqus input file format for use in the MCNP code. We believe this program will be used by many MCNP users for implementing complex tetrahedral-mesh models, including computational human phantoms, in the MCNP code.

Keywords: Tetrahedral mesh, Unstructured mesh, TetGen, MCNP, Abaqus

Technical Paper

Received August 22, 2016
Accepted September 19, 2016

Corresponding author:
Chan Hyeong Kim
Department of Nuclear Engineering, Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul 04763, Korea
Tel: +82-2-2220-0513,
Fax: +82-2-2220-4054,
E-mail: chkim@hanyang.ac.kr

JRPR

Introduction

Most general-purpose Monte Carlo radiation transport codes now support the use of tetrahedral mesh in geometry modeling [1-5]. The tetrahedral mesh is a powerful geometry for modeling complex and arbitrarily curved-surface models such as computational human phantoms [6-9]. The use of tetrahedral mesh overcomes the limitations of voxel geometry in computational human phantoms and precisely preserves the original structure of the polygonal-mesh geometry. Moreover, the use of tetrahedral mesh makes it possible to directly implement Computer-Aided Design (CAD)-based models in Monte Carlo codes.

Data Card – EMBED

- **Implement the phantom in “ABAQUS” file format** using the EMBED.

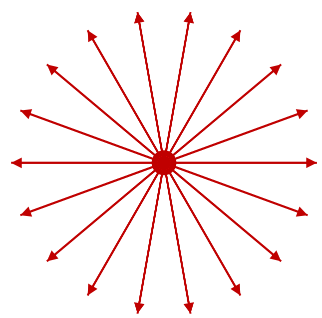
```
C EMBED is required for embedding a mesh geometry into MCNP6 input
embed2 meshgeo=abaqus                       $ Format specification
       mgeoin=../phantoms/mrcp-00f.inp      $ Name of the input file
       background=15000                     $ Cell number of the background
       matcell=     1   100     2   200     3   300     4   301     5   302
                    6   303     7   400     8   401     9   402    10   403
                   11   404    12   405    13   500    14   501    15   600
                   16   700    17   800    18   801    19   802    20   803
                   21   804    22   805    23   806    24   807    25   808
                   26   900    27   910    28  1000    29  1010    30  1100
                   31  1110    32  1200    33  1210    34  1300    35  1400
                   36  1500    37  1600    38  1700    39  1800    40  1900
```

- EMBED should be written in the data card.
- Number behind “embed” should be same as the universe number of the phantom.
- Set the mesh geometry type using “meshgeo” (ABAQUS or MCNPUM).
- Set the phantom file path using “mgeoin”.
- Set the background number using “background” defined in the cell file.
- Set the material for the pseudo cell using “matcell” involving the integer pairs.
  - ✓ First number: embedded pseudo-cell number (number must be sequential starting at 1)
  - ✓ Second number: pseudo-cell number

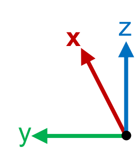
Data Card – Source for External Exposure

- Define the **external source** using the **SDEF.**

```
sdef par=p erg=1 pos=0 -100 0   $ General source definition
```



- **Type: point source**
- **Particle: photon**
- **Energy: 1 MeV**
- **Position: (0, -100, 0)**
- **Direction: isotropic**



Data Card – Source for Internal Exposure

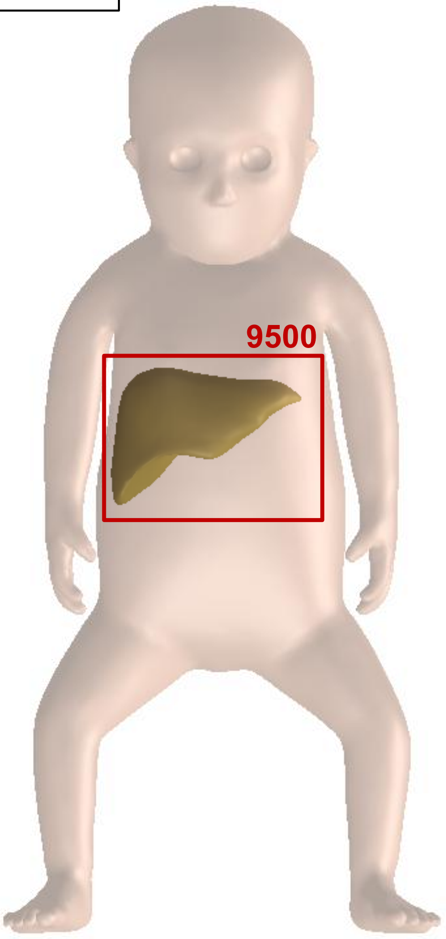
- Define the **internal source** using the **SDEF** and **“ABAQUS” file**.
  - Set **“pos=volumer”** in the **SDEF**.

```
sdef par=p erg=1 pos=volumer   $ General source definition
```

  - Add the **source line** in the **“ABAQUS” file**.

```
4820,  1247,  1346,  1432,  1379
4821,  1432,  1384,  1247,  1425
4822,  794,  917,  792,  1158
4823,  1430,  917,  1433,  1137
4824,  1432,  1346,  1424,  1379
4825,  1360,  1423,  1426,  1429
4826,  1407,  1360,  1373,  1426
4827,  1137,  917,  1158,  1154
4828,  1360,  1407,  1429,  1426
4829,  1431,  1420,  1417,  1426
4830,  1423,  1432,  1424,  1414
4831,  1346,  1247,  1432,  1424
4832,  1346,  1407,  1424,  914
4833,  1424,  1429,  1423,  1407
*Nset, nset=Set-material_9500, generate
1,   1434,   1
*Elset, elset=Set-material_9500, generate
1,   4833,   1
*Nset, nset=Set-statistic_9500, generate
1,   1434,   1
*Elset, elset=Set-statistic_9500, generate
1,   4833,   1
*Elset, elset=Set-source_9500, generate
1,   4833,   1
*End Part
**
```

<# of tetrahedrons for the source organ>



Data Card – Tally

- Define tally information from the **"*.tally" file in the "phantoms" folder**.
- Use the **+F6 tally** to calculate the **energy deposition of organs**.

```
C -------------------------------------------------------------------------
C TALLIES FOR EACH ORGAN/TISSUE
C -------------------------------------------------------------------------
read file=../phantoms/mrcp-00f.tally
C -------------------------------------------------------------------------
```

```
C TALLIES FOR EACH ORGAN/TISSUE
+f1006    ( 100 ) $Adrenal_left
+f2006    ( 200 ) $Adrenal_right
+f3006    ( 300 ) $ET1(0-8)
+f3016    ( 301 ) $ET1(8-40)
+f3026    ( 302 ) $ET1(40-50)
+f3036    ( 303 ) $ET1(50-Surface)
+f4006    ( 400 ) $ET2(-15-0)
+f4016    ( 401 ) $ET2(0-40)
+f4026    ( 402 ) $ET2(40-50)
```

<+F6 tally> <pseudo-cell no.>

- If the user wants to calculate the **energy deposition for several organs together**, enter their pseudo-cell number **in parentheses**.

ex) +f16 (9700 9900)

+f26 (400 401 402 403 404 405)

Tip for Unstructured Mesh File Format

- From MCNP 6.2, phantom implementation time can be reduced by **generating and reading the “MCNPUM” file,** which can be converted from the “ABAQUS” file using the “**um_convert**” program.

ABAQUS file

```
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_MCNP6/phantoms$ ls
mrcp-00f.cell  mrcp-00f.inp  mrcp-00f.material  mrcp-00f.tally
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_MCNP6/phantoms$ um_convert -a mrcp-00f.inp -um mrcp-00f.um

 UM_CONVERT input processing begins.        8- 4-2025 @  9:37:01

   Max threads available:     28
```

✓ -a: ABAQUS input file

✓ -um: MCNPUM output file

- From MCNP 6.3, “**HDF5” format can be implemented, which can be easily visualized** via visualization package such as ParaView.

```
embed2 meshgeo=abaqus
       mgeoin=../phantoms/mrcp-00f.inp
       hdf5file=../phantoms/mrcp-00f.hdf5
```

✓ -hdf5file: HDF5 output file

- Some parts of the EMBED should also be modified to read the converted files.

```
embed2 meshgeo=abaqus
       mgeoin=../phantoms/mrcp-00f.inp
```

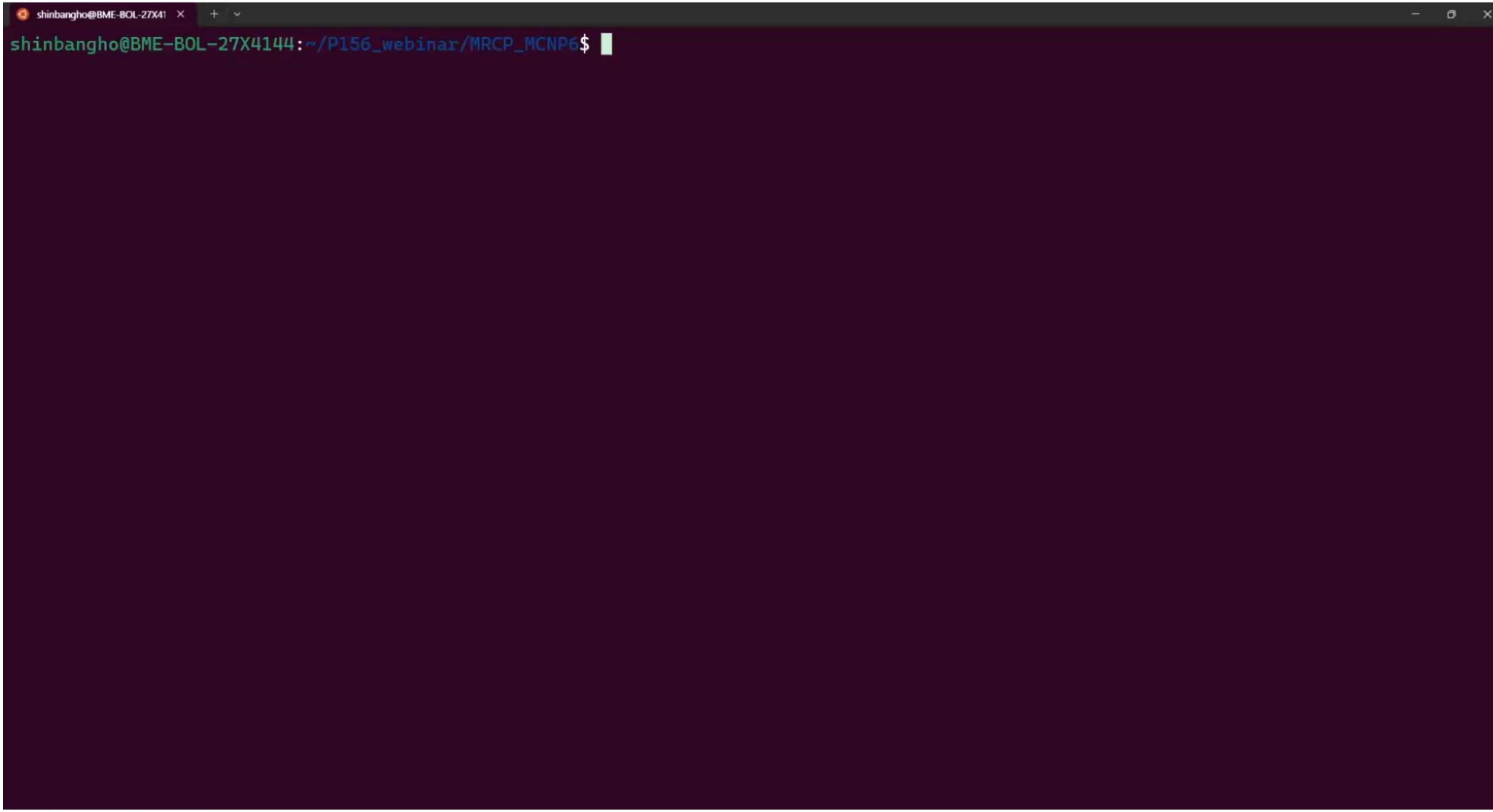
✓ Meshgeo: abaqus → mcnpum / hdf5

✓ mgeoin: **.inp → **.um / **.hdf5

Phantom file	Version	Implementation time
ABAQUS file	MCNP 6.2	About half a day
	MCNP 6.3	~30 minutes
HDF5 file	MCNP 6.3	~30 minutes
MCNPUM file	MCNP 6.2 & 6.3	Less than a minute

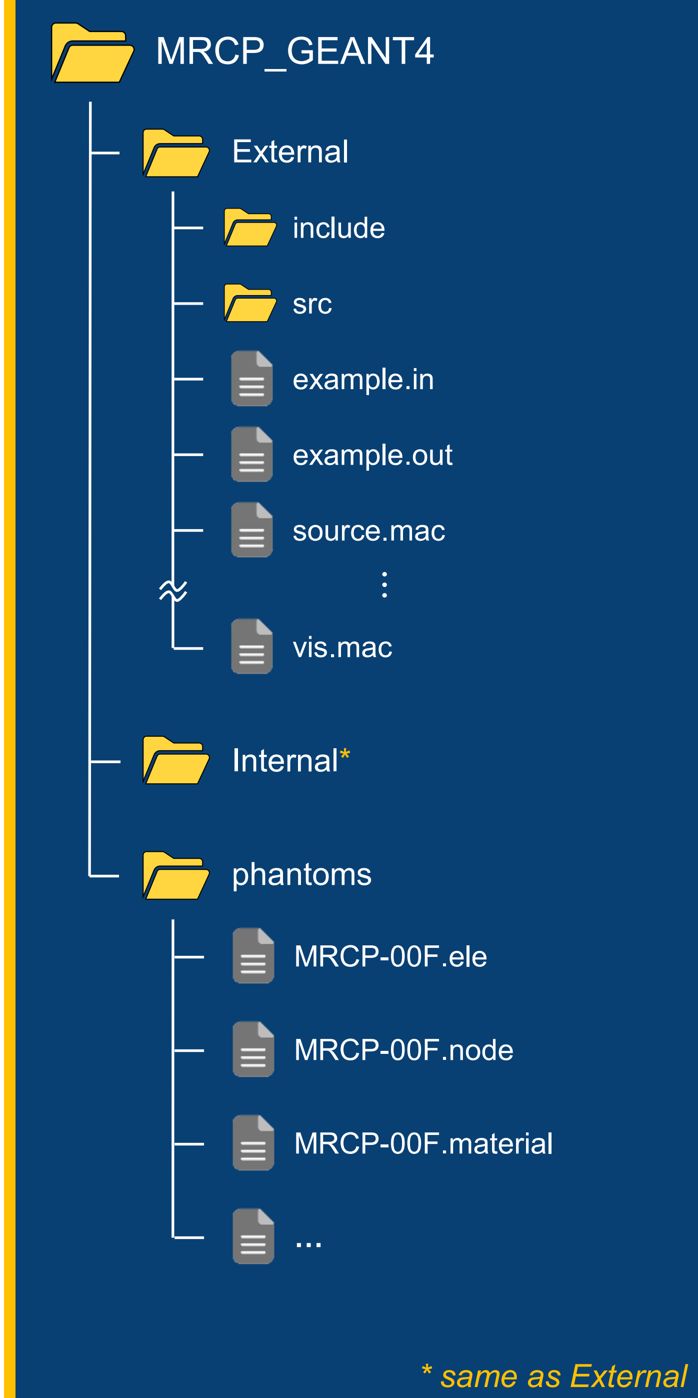
** Time was measured in “Intel® Core(TM) i7-14700K CPU @ 3.40GHz.*

Execution of MCNP6 – Demo Video



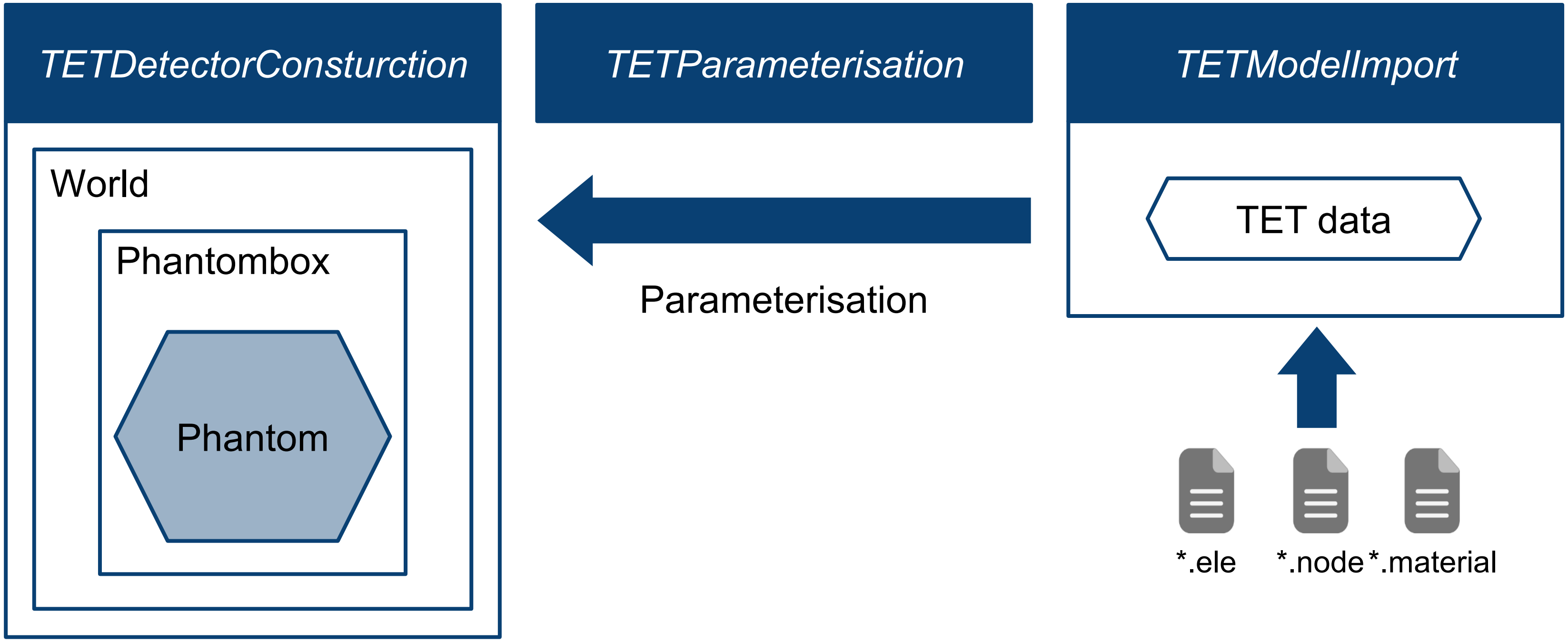
Monte Carlo Code Input Examples

GEANT4 Code



Geant4 Classes – Geometries

- *TETModelImport* is to **import the phantom data** (“*.ele”, “*.node”, “*.material” files).
- *TETDetectorConstruction* is to **construct the phantom and other geometries** (phantombox, world, scorer).
- *TETParameterisation* is to **define the tetrahedral mesh phantom** by parameterisation using the phantom data which is imported in *TETModelImport*.



Geant4 Classes – Geometries (Cont'd)

- *TETModelImport* is to **import the phantom data** ("*.ele", "*.node", "*.material" files).

```cpp
#include "TETModelImport.hh"

TETModelImport::TETModelImport(G4bool isAF, G4UIExecutive* ui)
{
    // set path for phantom data
    char* pPATH = getenv("PHANTOM_PATH");
    if( pPATH == 0 ){
        // exception for the case when PHANTOM_PATH environment variable was not set
        G4Exception("TETModelImport::TETModelImport","",JustWarning,
                G4String("PHANTOM_PATH environment variable was not set.").c_str());
        // default path for phantom data
        phantomDataPath = "../../phantoms";
    }
    else {
        // set path for phantom data as PHANTOM_PATH
        phantomDataPath = pPATH;
    }

    // set phantom name
    if(!isAF) phantomName = "MRCP_AM";
    else      phantomName = "MRCP_AF";

    G4cout << "================================================================================"<<G4endl;
    G4cout << "\t" << phantomName << " was implemented in this CODE!!   "<< G4endl;
    G4cout << "================================================================================"<<G4endl;

    G4String eleFile      =  phantomName + ".ele";
    G4String nodeFile     =  phantomName + ".node";
    G4String materialFile =  phantomName + ".material";

    // read phantom data files (*. ele, *.node)
    DataRead(eleFile, nodeFile);
    // read material file (*.material)
    MaterialRead(materialFile);
    // read colour data file (colour.dat) if this is interactive mode
    if(ui) ColourRead();
    // print the summary of phantom information
    PrintMaterialInfomation();
}
```

Read ELE and NODE tetrahedral-mesh phantom files

Read Material file

Geant4 Classes – Geometries (Cont'd)

- *TETDetectorConsturction* is to **construct the phantom and other geometries** (phantombox, world, scorer).

```cpp
void TETDetectorConstruction::SetupWorldGeometry()
{
    // Define the world box (size: 10*10*10 m3)
    //
    G4double worldXYZ = 10. * m;
    G4Material* vacuum = G4NistManager::Instance()->FindOrBuildMaterial("G4_Galactic");

    G4VSolid* worldSolid
      = new G4Box("worldSolid", worldXYZ/2, worldXYZ/2, worldXYZ/2);

    G4LogicalVolume* worldLogical
      = new G4LogicalVolume(worldSolid,vacuum,"worldLogical");

    worldPhysical
      = new G4PVPlacement(0,G4ThreeVector(), worldLogical,"worldPhysical", 0, false,0,false);

    // Define the phantom container (10-cm margins from the bounding box of phantom)
    //
    G4Box* containerSolid = new G4Box("phantomBox", phantomSize.x()/2 + 10.*cm,
                                                    phantomSize.y()/2 + 10.*cm,
                                                    phantomSize.z()/2 + 10.*cm);

    container_logic = new G4LogicalVolume(containerSolid, vacuum, "phantomLogical");

    new G4PVPlacement(0, G4ThreeVector(), container_logic, "PhantomPhysical",
                      worldLogical, false, 0);
    container_logic->SetOptimisation(TRUE);
    container_logic->SetSmartless( 0.5 ); // for optimization (default=2)
}
```

Define world

Define phantom box

Geant4 Classes – Geometries (Cont'd)

- *TETDetectorConsturction* is to **construct the phantom and other geometries** (phantombox, world, scorer).

```
void TETDetectorConstruction::ConstructPhantom()
{
    // Define the tetrahedral mesh phantom as a parameterised geometry
    //
    // solid and logical volume to be used for parameterised geometry
    G4VSolid* tetraSolid = new G4Tet("TetSolid",
                                G4ThreeVector(),
                                G4ThreeVector(1.*cm,0,0),
                                G4ThreeVector(0,1.*cm,0),
                                G4ThreeVector(0,0,1.*cm));

    G4Material* vacuum = G4NistManager::Instance()->FindOrBuildMaterial("G4_Galactic");
    tetLogic = new G4LogicalVolume(tetraSolid, vacuum, "TetLogic");

    // physical volume (phantom) constructed as parameterised geometry
    new G4PVParameterised("wholePhantom",tetLogic,container_logic,
                          kUndefined,tetData->GetNumTetrahedron(),
                          new TETParameterisation(tetData));
}
```

Implementation of tetrahedral-mesh phantoms using parameterization class

Geant4 Classes – Geometries (Cont'd)

- *TETParameterisation* is to **define the tetrahedral mesh phantom** by parameterisation using the phantom data which is imported in *TETModelImport*.

```
TETParameterisation::~TETParameterisation()
{}

G4VSolid* TETParameterisation::ComputeSolid(
                                   const G4int copyNo, G4VPhysicalVolume* )
{
    // return G4Tet*
    return tetData->GetTetrahedron(copyNo);
}

void TETParameterisation::ComputeTransformation(
                                   const G4int,G4VPhysicalVolume*) const
{}

G4Material* TETParameterisation::ComputeMaterial(const G4int copyNo,
                                                 G4VPhysicalVolume* phy,
                                                 const G4VTouchable* )
{
   // set the colour for each organ if visualization is required
    if(isforVis){
        G4int idx = tetData->GetMaterialIndex(copyNo);
        phy->GetLogicalVolume()->SetVisAttributes(visAttMap[idx]);
    }

    // return the material data for each material index
    return tetData->GetMaterial(tetData->GetMaterialIndex(copyNo));
}
```

Install tetrahedrons in function of the copy number

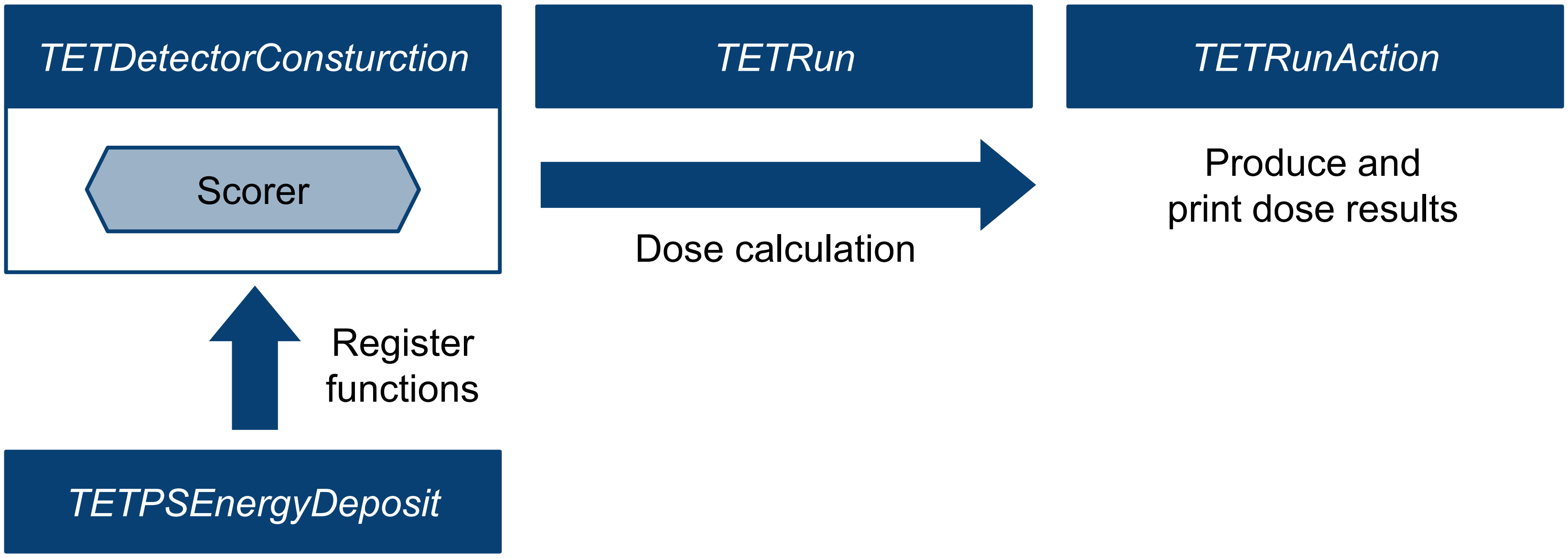
→ In this example, the tetrahedrons are saved in *TETModelImport*

Change the material of the volume in function of the copy number

→ In this example, the material information are saved in *TETModelImport*

Geant4 Classes – Organ Dose Calculation

- *TETDetectorConstruction* is to **construct scorer** in the phantom to get doses in organs.
- *TETPSEnergyDeposit* is to **specify scorer functions** for energy deposition.
- *TETRun* is to **calculate organ doses**.
- *TETRunAction* is to **produce and print dose results**.



Geant4 Classes – Organ Dose Calculation (Cont'd)

- *TETDetectorConstruction* is to **construct scorer** in the phantom to get doses in organs.

```
void TETDetectorConstruction::ConstructSDandField()
{
    // Define detector (Phantom SD) and scorer (eDep)
    //
    G4SDManager* pSDman = G4SDManager::GetSDMpointer();
    G4String phantomSDname = "PhantomSD";

    // MultiFunctional detector
    G4MultiFunctionalDetector* MFDet = new G4MultiFunctionalDetector(phantomSDname);
    pSDman->AddNewDetector( MFDet );

    // scorer for energy depositon in each organ
    MFDet->RegisterPrimitive(new TETPSEnergyDeposit("eDep", tetData));

    // attach the detector to logical volume for parameterised geometry (phantom geometry)
    SetSensitiveDetector(tetLogic, MFDet);
}
```

Compute energy deposition in each organ using G4*PSEnergyDeposit* class

Geant4 Classes – Organ Dose Calculation (Cont'd)

- *TETPSEnergyDeposit* is to **specify scorer functions** for energy deposition.

```
G4int TETPSEnergyDeposit::GetIndex(G4Step* aStep)
{
    // return the organ ID (= material index)
    G4int copyNo = aStep->GetPreStepPoint()->GetTouchable()->GetCopyNumber();
    return tetData->GetMaterialIndex(copyNo);
}
```

Energy deposition is stored in the organs where the *step* interacted

- *TETRun* is to **calculate organ doses**.

```
void TETRun::RecordEvent(const G4Event* event)
{
    auto  fCollID
    = G4SDManager::GetSDMpointer()->GetCollectionID("PhantomSD/eDep");

    // Hits collections
    //
    G4HCofThisEvent* HCE = event->GetHCofThisEvent();
    if(!HCE) return;

    G4THitsMap<G4double>* evtMap =
            static_cast<G4THitsMap<G4double>*>(HCE->GetHC(fCollID));

    // sum up the energy deposition and the square of it
    for (auto itr : *evtMap->GetMap()) {
        edepMap[itr.first].first  += *itr.second;                        //sum
        edepMap[itr.first].second += (*itr.second) * (*itr.second); //sum square
    }
}
```

Compute organ doses from the stored data in *TETPSEnergyDeposit* class

Geant4 Classes – Organan Dose Calculation (Cont’d)

- *TETRunAction* is to **produce and print dose results**.

Write the result file at the *EndOfRunAction!*

```cpp
void TETRunAction::PrintResult(std::ostream &out)
{
    // Print run result
    //
    using namespace std;
    EDEPMAP edepMap = *fRun->GetEdepMap();

    out << G4endl
        << "=====================================================================" << G4endl
        << " Run #" << runID << " / Number of event processed : "<< numOfEvent     << G4endl
        << "=====================================================================" << G4endl
        << "organ ID| "
        << setw(19) << "Organ Mass (g)"
        << setw(19) << "Dose (Gy/source)"
        << setw(19) << "Relative Error" << G4endl;

    out.precision(3);
    auto massMap = tetData->GetMassMap();
    for(auto itr : massMap){
        G4double meanDose    = edepMap[itr.first].first  / itr.second / numOfEvent;
        G4double squareDoese = edepMap[itr.first].second / (itr.second*itr.second);
        G4double variance    = ((squareDoese/numOfEvent) - (meanDose*meanDose))/numOfEvent;
        G4double relativeE   = sqrt(variance)/meanDose;

        out << setw(8)  << itr.first << "| "
            << setw(19) << fixed      << itr.second/g;
        out << setw(19) << scientific << meanDose/(joule/kg);
        out << setw(19) << fixed      << relativeE << G4endl;
    }
    out << "=====================================================================" << G4endl << G4endl;
}
```

Geant4 Classes – Organ Dose Calculation (Cont'd)

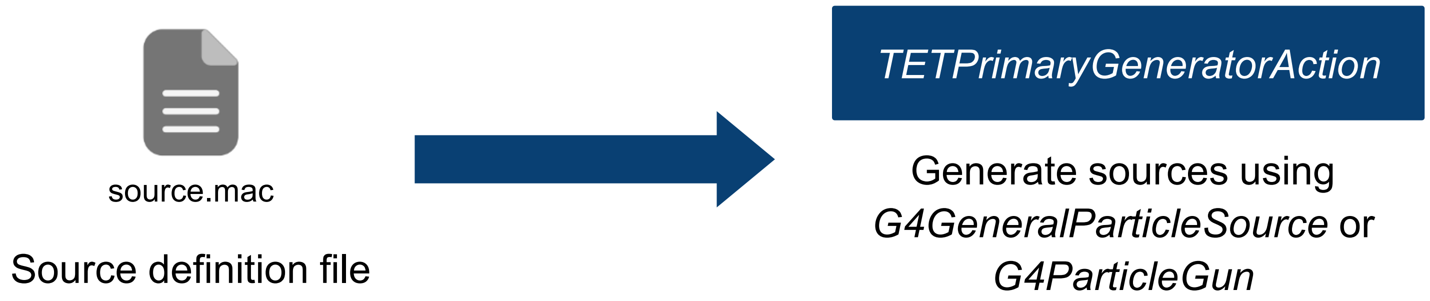
- *TETRunAction* is to **produce and print dose results**.

```

=============================================================================
 Run #0 / Number of event processed : 10000000
=============================================================================
organ ID|      Organ Mass (g)   Dose (Gy/source)     Relative Error
     100|               8.683          1.823e-17              0.214
     200|               8.683          2.226e-17              0.184
     300|               0.022          1.212e-17              0.623
     301|               0.090          1.298e-17              0.594
     302|               0.028          1.613e-17              0.553
     303|              11.291          2.867e-17              0.163
     400|               0.141          3.189e-17              0.582
     401|               0.390          1.916e-17              0.400
     402|               0.098          1.795e-17              0.367
     403|               0.049          2.251e-17              0.450
     404|               0.098          2.337e-17              0.340
     405|              28.808          2.232e-17              0.105
     500|               0.086          9.438e-18              0.511
     501|               0.024          4.264e-17              0.521
     600|               0.023          3.064e-17              0.470
     700|              10.364          2.425e-17              0.165
     800|               0.025          7.156e-18              0.530
     801|               0.031          7.956e-18              0.529
     802|               0.052          1.636e-17              0.735
     803|               0.130          1.912e-17              0.470
     804|               0.026          4.134e-17              0.673
     805|               0.052          4.320e-17              0.589
     806|               0.052          3.789e-17              0.545
```

Geant4 Classes – Set Source

- *TETPrimaryGeneratorAction* is to **generate sources**.
  - External exposure
    - ✓ Sources are generated through *G4GeneralParticleSource.*
  - Internal exposure
    - ✓ Sources are generated through *G4ParticleGun*.
- **Source information** (e.g., particle type, energy, ...) is specified in **source definition file** (“*.mac”).



Source Definition File Example

- Source definition file (“Source.mac”) for **external exposure**

```
# GeneralParticleSource :
# isotropic 1 MeV-gamma point source at (0, -1 m, 0)
/gps/ang/type iso
/gps/energy 1 MeV
/gps/particle gamma
/gps/pos/type Point
/gps/pos/centre 0 -1. 0 m
```

**** For more details about G4GeneralParicleSource (“/gps/”), see Geant4 GPS manual.***

- Source can be specified by using macro commands for *G4GeneralParticleSource*.
- **Particle direction, energy, type, geometry, position** is set in sequence in the “source.mac” file.

- Source definition file (“Source.mac”) for **internal exposure**

```
# ParticleGun macro:
# 1 MeV-gamma source
/gun/particle gamma
/gun/energy 1. MeV
```

**** For more details about G4ParticleGun (“/gun/”), see Geant4 Users Guide.***

- Source can be specified by using macro commands for *G4ParticleGun*.
- **Particle type and energy** is set in sequence in the “source.mac” file.
- **Source position** (i.e., source organ) is defined in the **command line for execution.**

Input Macro File Example

- Input macro file ("sample.in")

```
# Example macro file to run in the batch mode

# Set the verbose
/run/verbose 2

# Set the number of threads for multi-threading mode
#/run/numberOfThreads 1

# Initialize
/run/initialize

# source setting
/control/execute source.mac

# Set the nps
/run/beamOn 10000000
```

- Set the **number of threads** for Geant4 simulation using "/run/numberOfThreads" (Geant4 should be compiled in multi-threaded mode).
- Set the **source definition file path** using "/control/execute".
- Set the **number of particles (NPS)** for each run using "/run/beamOn"

How to Compile Geant4

- Geant4 compile (with CMake)

    1. cd PATH_to_example (either "External" or "Internal" folder)
    2. mkdir example_build
    3. cd example_build
    4. cmake ..
    5. make

Compile example

```
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_GEANT4$ cd External/
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_GEANT4/External$ mkdir example_build
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_GEANT4/External$ cd example_build
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_GEANT4/External/example_build$ cmake ..
```

⋮

```
-- Build files have been written to: /home/shinbangho/P156_webinar/MRCP_GEANT4/External/example_build
shinbangho@BME-BOL-27X4144:~/P156_webinar/MRCP_GEANT4/External/example_build$ make
[  8%] Building CXX object CMakeFiles/External.dir/External.cc.o
[ 16%] Building CXX object CMakeFiles/External.dir/src/TETActionInitialization.cc.o
[ 25%] Building CXX object CMakeFiles/External.dir/src/TETDetectorConstruction.cc.o
[ 33%] Building CXX object CMakeFiles/External.dir/src/TETModelImport.cc.o
[ 41%] Building CXX object CMakeFiles/External.dir/src/TETPSEnergyDeposit.cc.o
[ 50%] Building CXX object CMakeFiles/External.dir/src/TETParameterisation.cc.o
[ 58%] Building CXX object CMakeFiles/External.dir/src/TETPhysicsList.cc.o
[ 66%] Building CXX object CMakeFiles/External.dir/src/TETPrimaryGeneratorAction.cc.o
[ 75%] Building CXX object CMakeFiles/External.dir/src/TETRun.cc.o
[ 83%] Building CXX object CMakeFiles/External.dir/src/TETRunAction.cc.o
[ 91%] Building CXX object CMakeFiles/External.dir/src/TETSteppingAction.cc.o
[100%] Linking CXX executable External
[100%] Built target External
```

Created!

External	8/4/2025 10:15 AM	File	439 KB
Internal	8/4/2025 10:20 AM	File	446 KB

How to Execute Geant4

- Geant4 execution
  1. ./*: Executable "External" or "Internal" file
  2. –m: input macro file (not provided->interactive mode)
  3. –o: output file (default: [Input macro file name].out)
  4. –p: option to change the phantom (default: MRCP-00M)
  5. –i: ID of source organ (mandatory)

Execution example for external exposure

```
example_build$ ./External -m example.in -o example.out -p MRCP-00F
```

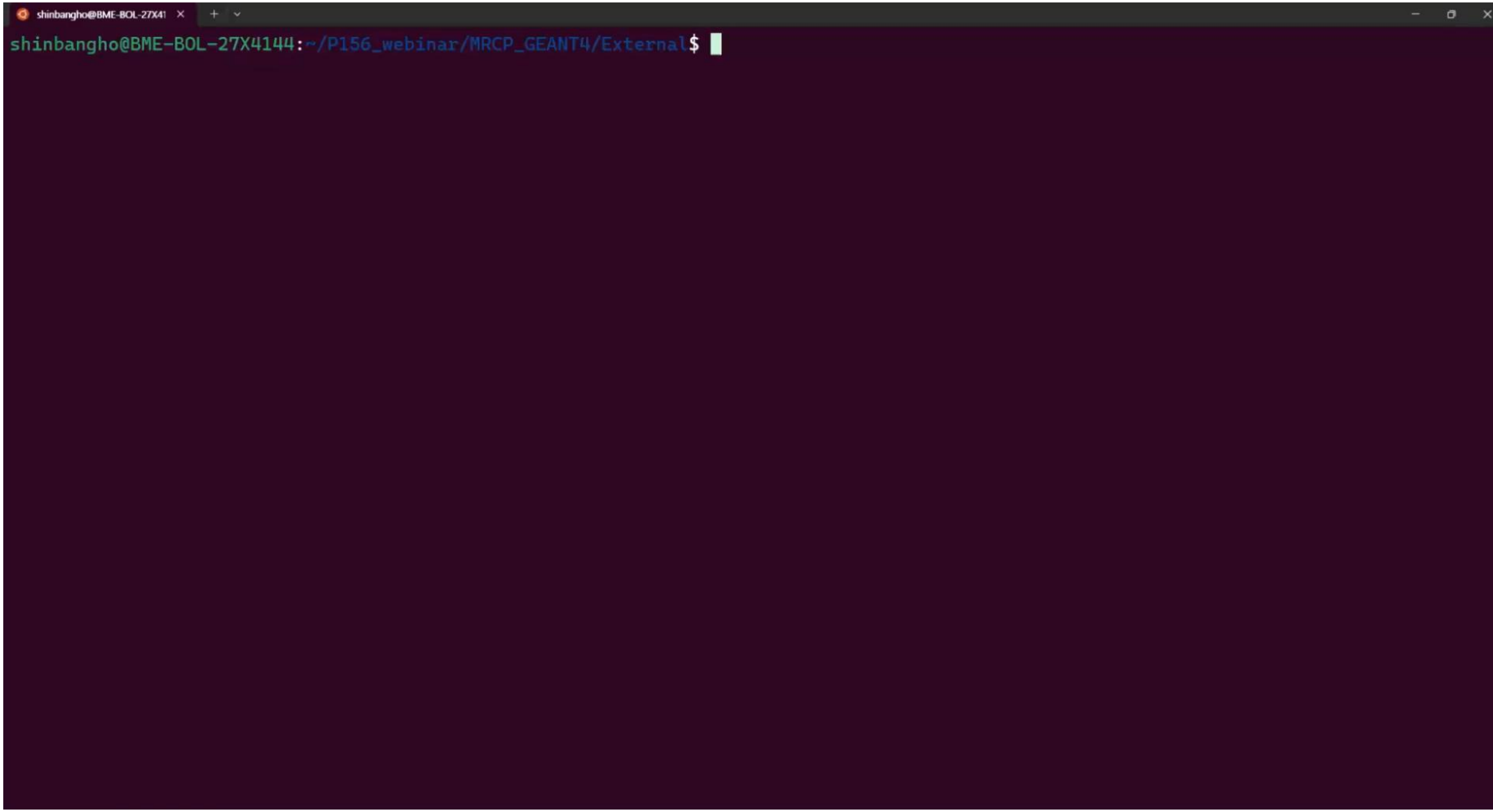
①: `./External` ②: `-m example.in` ③: `-o example.out` ④: `-p MRCP-00F`

Execution example for internal exposure

```
example_build$ ./Internal -i 9500 -m example.in -o example.out -p MRCP-00F
```

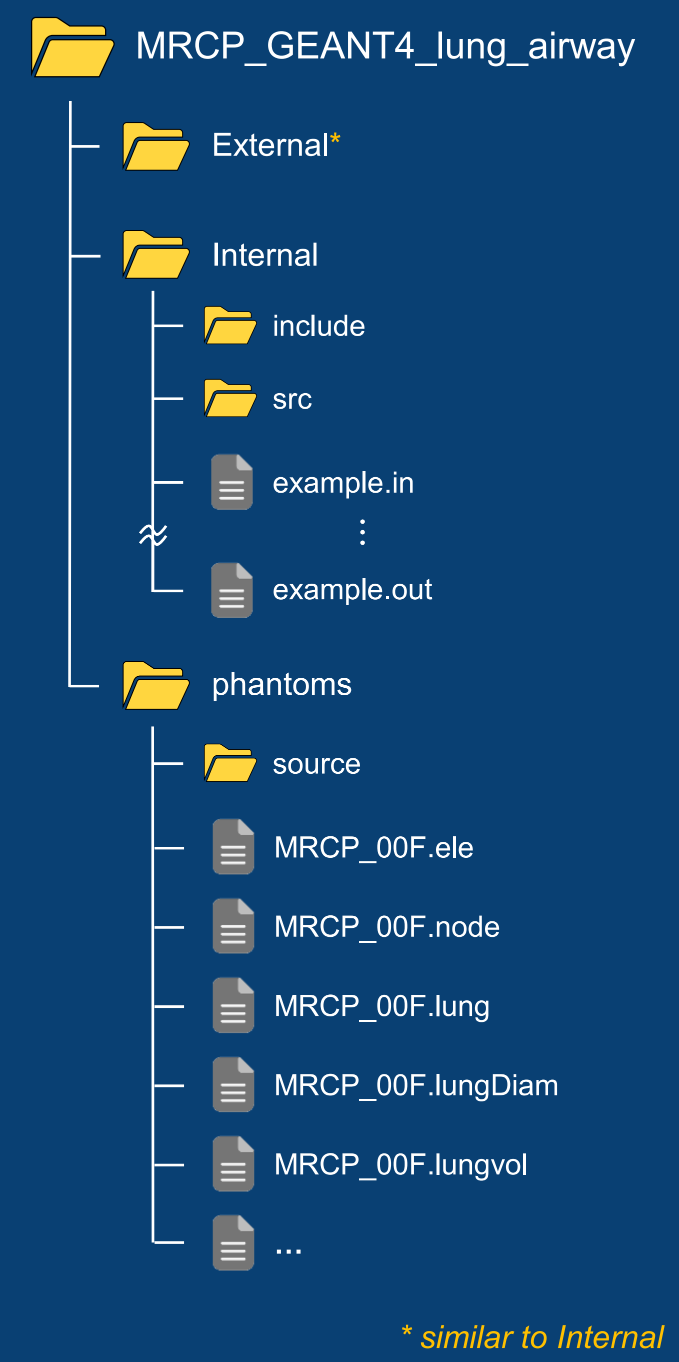
①: `./Internal` ⑤: `-i 9500` ②: `-m example.in` ③: `-o example.out` ④: `-p MRCP-00F`

Compile and Execution of Geant4 – Demo Video



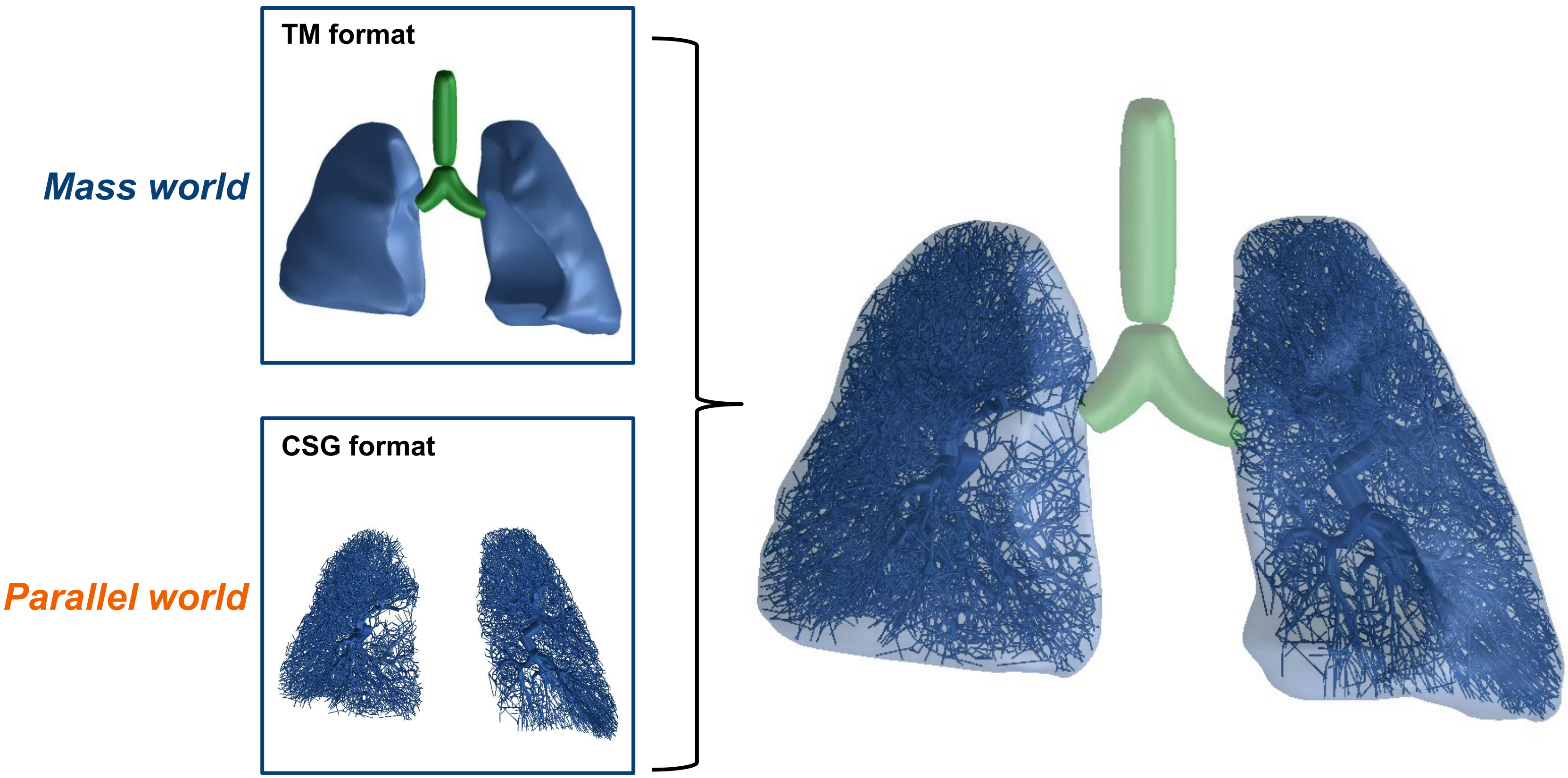
Monte Carlo Code Input Examples

Geant4 code – Lung Airway



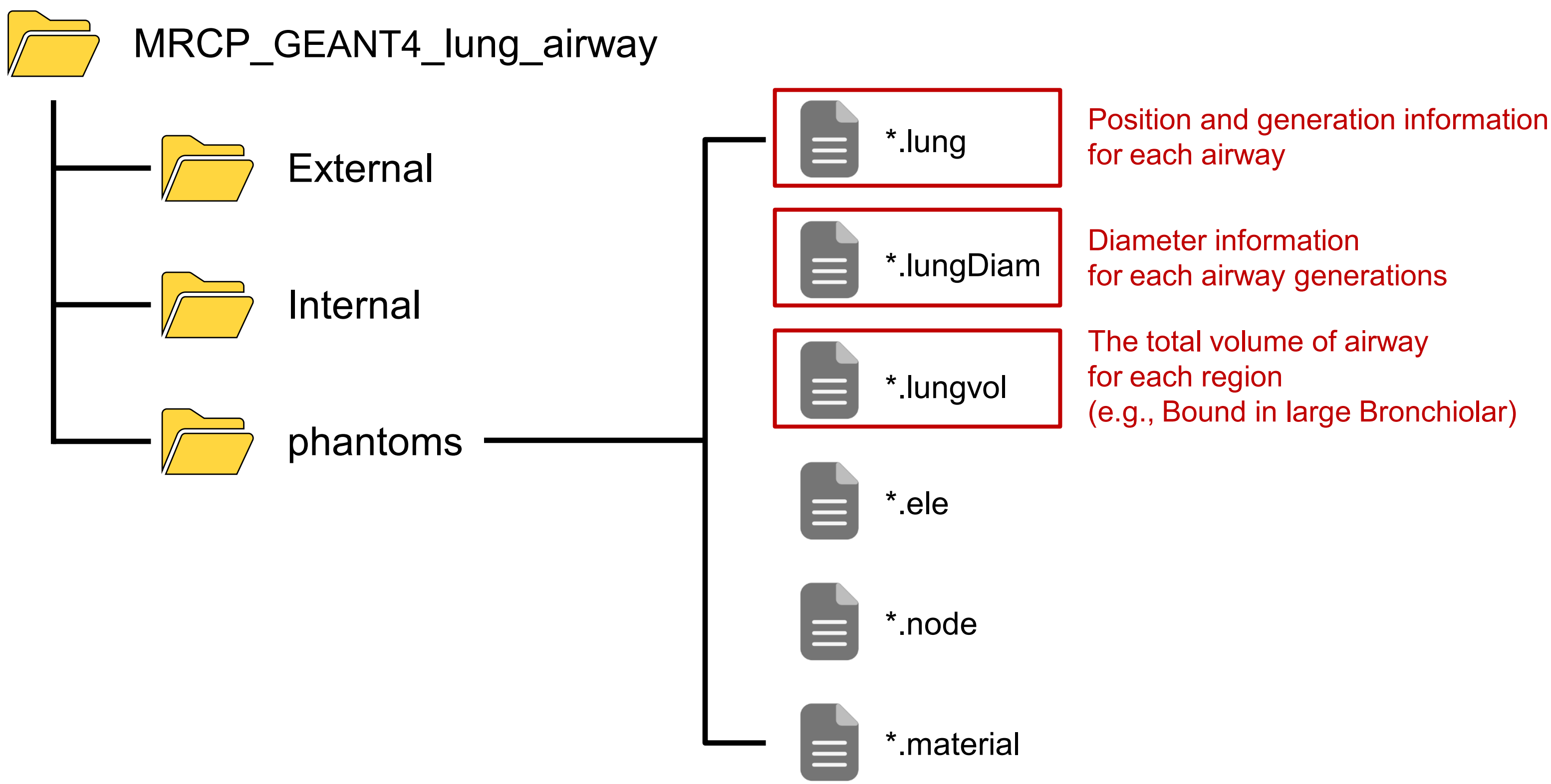
Installation of Lung Airway in MRCPs

- Lung airway is developed in **constructive solid geometry (CSG) format**, while the MRCPs are in tetrahedral-mesh (TM) format.
- To install the lung airways in the MRCPs, **"*G4VUserParallelWorld*" class** of the Geant4 code was used.



Lung Airway Files in “phantoms” Folder

- The “phantoms” folder contains additional information files for the lung airway with existing files.
- **Information list: position, generation, diameter, volume**



How to Execute Geant4 – External Exposure

- ./External -m [macro] -o [output] –p [phantom name]
  - -m: macro file name
    - ✓ example.in (for only execution) and init_vis.mac (for visualization).
  - -o: output file name
  - -p: option to change the phantom (default: MRCP-00M)

Phantom information

Organ ID	# of Tet	vol [cm3]	d [g/cm3]	mass [g]	organ/tissue
100	4630	3.503	1.028	3.601	Adrenal_left
200	2374	3.503	1.028	3.601	Adrenal_right
300	3973	0.001	1.061	0.001	ET1(0-8)
301	4002	0.003	1.061	0.003	ET1(8-40)
302	3969	0.001	1.061	0.001	ET1(40-50)
303	3273	0.090	1.061	0.095	ET1(50-Surface)
400	25272	0.033	1.000	0.033	ET2(-15-0)
401	25234	0.089	1.061	0.094	ET2(0-40)
402	25291	0.022	1.061	0.024	ET2(40-50)
403	25305	0.011	1.061	0.012	ET2(50-55)
404	25240	0.023	1.061	0.024	ET2(55-65)
405	25751	1.588	1.061	1.685	ET2(65-Surface)
500	2562	0.008	1.052	0.008	Oral_mucosa_tongue
501	1819	0.009	1.052	0.010	Oral_mucosa_moth_floor
600	833	0.006	1.052	0.006	Oral_mucosa_lips_and_cheeks
700	2461	0.511	1.061	0.542	Trachea
800	13048	0.002	1.000	0.002	BB(-11--6)
801	13042	0.002	1.061	0.002	BB(-6-0)
802	12964	0.003	1.061	0.003	BB(0-10)
803	12946	0.008	1.061	0.009	BB(10-35)
804	13060	0.002	1.061	0.002	BB(35-40)
805	13000	0.003	1.061	0.003	BB(40-50)
806	12978	0.003	1.061	0.004	BB(50-60)
807	12953	0.003	1.061	0.004	BB(60-70)
808	13203	0.330	1.061	0.350	BB(70-surface)
900	3274	0.072	1.070	0.077	Blood_in_large_arteries_head
910	2669	0.264	1.070	0.282	Blood_in_large_veins_head
1000	66204	8.842	1.070	9.460	Blood_in_large_arteries_trunk
1010	66498	22.213	1.070	23.768	Blood_in_large_veins_trunk
1100	37289	1.645	1.070	1.760	Blood_in_large_arteries_arms
1110	30248	6.355	1.070	6.800	Blood_in_large_veins_arms
1200	57951	5.703	1.070	6.103	Blood_in_large_arteries_legs
1210	48229	19.954	1.070	21.351	Blood_in_large_veins_legs
1300	11028	1.248	1.542	1.924	Humeri_upper_cortical
1400	3864	2.803	1.265	3.545	Humeri_upper_spongiosa

Lung airway information

*********************************Lung Information********************************

Organ ID	vol [cm3]	d [g/cm3]	mass [g]	organ/tissue
818	1.266	1.061	1.343	BB(70-surface)_2to8gen
817	0.021	1.061	0.022	BB(60-70)_2to8gen
816	0.021	1.061	0.023	BB(50-60)_2to8gen
815	0.020	1.061	0.021	BB(40-50)_2to8gen
814	0.010	1.061	0.010	BB(35-40)_2to8gen
813	0.045	1.061	0.048	BB(10-35)_2to8gen
812	0.018	1.061	0.019	BB(0-10)_2to8gen
811	0.059	1.061	0.063	BB(-6-0)_2to8gen
810	0.008	1.000	0.008	BB(-11--6)_2to8gen
819	0.430	0.001	0.000	BB(air)_2to8gen
826	0.250	1.061	0.266	bb(25-surface)_9to15gen
825	0.118	1.061	0.126	bb(20-25)_9to15gen
824	0.180	1.061	0.191	bb(12-20)_9to15gen
823	0.168	1.061	0.178	bb(4-12)_9to15gen
822	0.080	1.061	0.085	bb(0-4)_9to15gen
821	0.077	1.061	0.081	bb(-4-0)_9to15gen
820	0.037	1.000	0.037	bb(-6--4)_9to15gen
829	1.015	0.001	0.001	bb(air)_9to15gen

Lung Data

[mm]	#	layer0	layer1	layer2	layer3	layer4	layer5	layer6	layer7	layer8	layer9
gen-1	2	4.983	4.013	3.993	3.973	3.953	3.943	3.893	3.873	3.861	3.851
gen-2	4	3.853	2.883	2.863	2.843	2.823	2.813	2.763	2.743	2.731	2.721
gen-3	8	3.079	2.109	2.089	2.069	2.049	2.039	1.989	1.969	1.957	1.947
gen-4	15	2.530	1.560	1.540	1.520	1.500	1.490	1.440	1.420	1.408	1.398
gen-5	27	2.272	1.302	1.282	1.262	1.242	1.232	1.182	1.162	1.150	1.140
gen-6	48	2.046	1.076	1.056	1.036	1.016	1.006	0.956	0.936	0.924	0.914
gen-7	92	1.885	0.915	0.895	0.875	0.855	0.845	0.795	0.775	0.763	0.753
gen-8	173	1.755	0.785	0.765	0.745	0.725	0.715	0.665	0.645	0.633	0.623
gen-9	272	0.601	0.581	0.571	0.555	0.539	0.531	0.523	0.519	0.000	0.000
gen-10	520	0.502	0.482	0.472	0.456	0.440	0.432	0.424	0.420	0.000	0.000
gen-11	959	0.418	0.398	0.388	0.372	0.356	0.348	0.340	0.336	0.000	0.000
gen-12	1702	0.350	0.330	0.320	0.304	0.288	0.280	0.272	0.268	0.000	0.000
gen-13	2779	0.296	0.276	0.266	0.250	0.234	0.226	0.218	0.214	0.000	0.000
gen-14	3888	0.258	0.238	0.228	0.212	0.196	0.188	0.180	0.176	0.000	0.000
gen-15	3648	0.236	0.216	0.206	0.190	0.174	0.166	0.158	0.154	0.000	0.000
gen-16		0.236	0.216	0.206	0.190	0.174	0.166	0.158	0.154	0.000	0.000

How to Execute Geant4 – External Exposure

- ./External -m [macro] -o [output] –p [phantom name]
  - -m: macro file name
    - ✓ example.in (for only execution) and init_vis.mac (for visualization).
  - -o: output file name
  - -p: option to change the phantom (default: MRCP-00M)

Output file example

Run #0 / Number of event processed : 10000000

organ ID	Organ Mass (g)	Dose (Gy/source)	Relative Error
BB_basal	0.037	3.444e-17	0.576
BB_secretory	0.069	5.176e-17	0.573
bb_secretory	0.167	4.937e-17	0.573
Lung(AI)_left	28.689	3.625e-17	0.092
Lung(AI)_right	31.311	3.223e-17	0.092

404	25240	0.023	1.061	0.024	ET2(55-65)
405	25751	1.588	1.061	1.685	ET2(65-Surface)
500	2562	0.008	1.052	0.008	Oral_mucosa_tongue
501	1819	0.009	1.052	0.010	Oral_mucosa_moth_floor
600	833	0.006	1.052	0.006	Oral_mucosa_lips_and_cheeks
700	2461	0.511	1.061	0.542	Trachea
800	13048	0.002	1.000	0.002	BB(-11--6)
801	13042	0.002	1.061	0.002	BB(-6-0)
802	12964	0.003	1.061	0.003	BB(0-10)
803	12946	0.008	1.061	0.009	BB(10-35)
804	13060	0.002	1.061	0.002	BB(35-40)
805	13000	0.003	1.061	0.003	BB(40-50)
806	12978	0.003	1.061	0.004	BB(50-60)
807	12953	0.003	1.061	0.004	BB(60-70)
808	13203	0.330	1.061	0.350	BB(70-surface)
900	3274	0.072	1.070	0.077	Blood_in_large_arteries_head
910	2669	0.264	1.070	0.282	Blood_in_large_veins_head
1000	66204	8.842	1.070	9.460	Blood_in_large_arteries_trunk
1010	66498	22.213	1.070	23.768	Blood_in_large_veins_trunk
1100	37289	1.645	1.070	1.760	Blood_in_large_arteries_arms
1110	30248	6.355	1.070	6.800	Blood_in_large_veins_arms
1200	57951	5.703	1.070	6.103	Blood_in_large_arteries_legs
1210	48229	19.954	1.070	21.351	Blood_in_large_veins_legs
1300	11028	1.248	1.542	1.924	Humeri_upper_cortical
1400	3864	2.803	1.265	3.545	Humeri_upper_spongiosa

826	0.250	1.061	0.266	bb(25-surface)_9to15gen
825	0.118	1.061	0.126	bb(20-25)_9to15gen
824	0.180	1.061	0.191	bb(12-20)_9to15gen
823	0.168	1.061	0.178	bb(4-12)_9to15gen
822	0.080	1.061	0.085	bb(0-4)_9to15gen
821	0.077	1.061	0.081	bb(-4-0)_9to15gen
820	0.037	1.000	0.037	bb(-6--4)_9to15gen
829	1.015	0.001	0.001	bb(air)_9to15gen

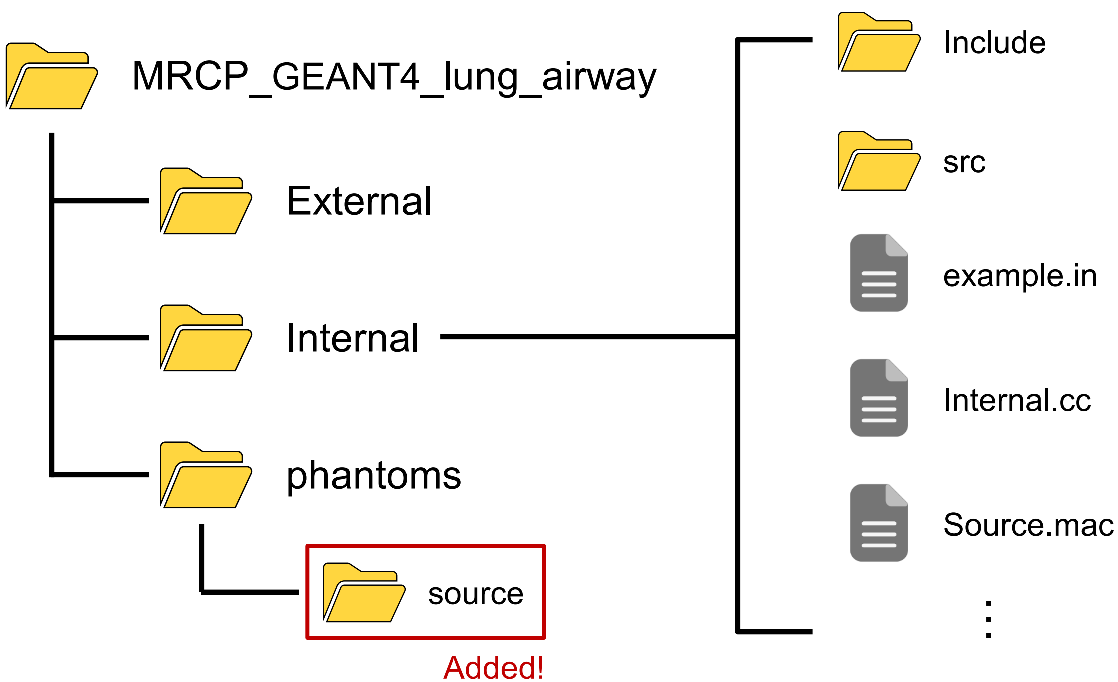
Lung Data

[mm]	#	layer0	layer1	layer2	layer3	layer4	layer5	layer6	layer7	layer8	layer9
gen-1	2	4.983	4.013	3.993	3.973	3.953	3.943	3.893	3.873	3.861	3.851
gen-2	4	3.853	2.883	2.863	2.843	2.823	2.813	2.763	2.743	2.731	2.721
gen-3	8	3.079	2.109	2.089	2.069	2.049	2.039	1.989	1.969	1.957	1.947
gen-4	15	2.530	1.560	1.540	1.520	1.500	1.490	1.440	1.420	1.408	1.398
gen-5	27	2.272	1.302	1.282	1.262	1.242	1.232	1.182	1.162	1.150	1.140
gen-6	48	2.046	1.076	1.056	1.036	1.016	1.006	0.956	0.936	0.924	0.914
gen-7	92	1.885	0.915	0.895	0.875	0.855	0.845	0.795	0.775	0.763	0.753
gen-8	173	1.755	0.785	0.765	0.745	0.725	0.715	0.665	0.645	0.633	0.623
gen-9	272	0.601	0.581	0.571	0.555	0.539	0.531	0.523	0.519	0.000	0.000
gen-10	520	0.502	0.482	0.472	0.456	0.440	0.432	0.424	0.420	0.000	0.000
gen-11	959	0.418	0.398	0.388	0.372	0.356	0.348	0.340	0.336	0.000	0.000
gen-12	1702	0.350	0.330	0.320	0.304	0.288	0.280	0.272	0.268	0.000	0.000
gen-13	2779	0.296	0.276	0.266	0.250	0.234	0.226	0.218	0.214	0.000	0.000
gen-14	3888	0.258	0.238	0.228	0.212	0.196	0.188	0.180	0.176	0.000	0.000
gen-15	3648	0.236	0.216	0.206	0.190	0.174	0.166	0.158	0.154	0.000	0.000
gen-16		0.236	0.216	0.206	0.190	0.174	0.166	0.158	0.154	0.000	0.000

****

Source Files in “Internal” Folder

- For the internal exposure, the **“source” folder containing the location of source particles** is included in the “phantoms” folder.



How to Execute Geant4 – Internal Exposure

- ./Internal **-i [source]** -m [macro] -o [output] -f
  - **-i: organ ID or airway name for generating the internal source particles.**
    - ✓ If the user wants to set the source region in general organs, enter the organ ID (integer format) such as 9500 (liver).
    - ✓ If the user wants to set the source region in lung airways, **enter the specific name below**.

Source region	Source region acronym
Bronchial (BB) fast mucus	BB_fast
BB slow mucus	BB_slow
BB bound layer	BB_bound
BB sequestered layer	BB_sequestered
Bronchial (bb) fast mucus	bb_fast
bb slow mucus	bb_slow
bb bound layer	bb_bound
bb sequestered layer	bb_sequestered
Alveolar-interstitial (AI)	AI

- ✓ -m: macro file name
- ✓ -o: output file name
- ✓ -f: option for female phantom

How to Execute Geant4 – Internal Exposure

- ./Internal **-i [source]** -m [macro] -o [output] -f
  - **-i: organ ID or airway name for generating the internal source particles.**
    - ✓ If the user wants to set the source region in general organs, enter the organ ID (integer format) such as 9500 (liver).
    - ✓ If the user wants to set the source region in lung airways, **enter the specific name below**.

Output file example

```
==========================================================================
Run #0 / Number of event processed : 100000
==========================================================================
          organ ID|      Organ Mass (g)    Dose (Gy/source)      Relative Error
          BB_basal|               0.037          1.078e-14               0.285
      BB_secretory|               0.069          8.766e-15               0.326
      bb_secretory|               0.167          0.000e+00                -nan
     Lung(AI)_left|              28.689          2.049e-12               0.004
    Lung(AI)_right|              31.311          2.133e-12               0.004
==========================================================================
```

bb slow mucus	bb_slow
bb bound layer	bb_bound
bb sequestered layer	bb_sequestered
Alveolar-interstitial (AI)	AI

- ✓ -m: macro file name
- ✓ -o: output file name
- ✓ -f: option for female phantom

Tip for Skeletal Dosimetry

Doses to Spongiosa and Medullary Cavity

- Like the ICRP-143 voxel phantoms, the skeletal target tissues [active marrow (AM, = red bone marrow) and bone endosteum (TM_{50})] are macroscopically defined as spongiosa and medullary cavity in the MRCPs.
- Therefore, **the doses to skeletal target tissues are approximated by the doses to the spongiosa and medullary cavity.**

$$D_{skel}(AM) = \sum_{x} \frac{m(AM,x)}{m(AM)} D(SP,x) + \sum_{x} \frac{m(AM,x)}{m(AM)} D(MM,x)$$

$$D_{skel}(TM_{50}) = \sum_{x} \frac{m(TM_{50},x)}{m(TM_{50})} D(SP,x) + \sum_{x} \frac{m(TM_{50},x)}{m(TM_{50})} D(MM,x)$$

- $D_{skel}(AM/TM_{50})$: Skeletal-averaged absorbed dose to AM/TM_{50}
- $m(AM/TM_{50},x)$: Masses of AM/TM_{50} in bone site x
- $m(AM/TM_{50})$: Masses of total AM/TM_{50}
- $D(SP/\mathrm{MM},x)$: Absorbed dose to spongiosa/medullary cavity

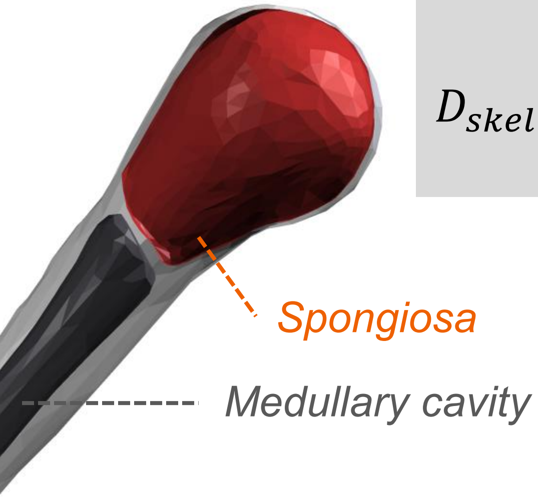
- The bone-site-specific masses of AM and endosteum can be found in **Tables 3.3–3.8 of *Publication 143*.**

Fluence-to-dose Response Functions (DRFs)

- **For photon and neutron, an advanced method, called fluence-to-dose response functions (DRFs)**, can be used to estimate doses to skeletal target tissues (i.e., AM and TM_{50}).
- The AM and TM_{50} doses can be estimated only with the fluence of the photons and/or neutrons passing through spongiosa and medullary cavity.

$$D(r_T, x) = \int_E \Phi(E, r_S, x) R(r_T \leftarrow r_S, x, E) dE$$

$$D_{skel}(r_T) = \sum_x \frac{m(r_T, x)}{m(r_T)} D(r_T, x)$$



- $D(r_T, x)$: Absorbed dose to tissue r_T in bone site x
- $\Phi(E, r_S, x)$: Bone-specific energy-dependent fluence
- $R(r_T \leftarrow r_S, x, E)$: Bone-specific energy-dependent DRFs
- $D_{skel}(r_T)$: Skeletal-averaged absorbed dose to tissue r_T

- The photon fluence-to-dose response functions can be found in **Electronic files of *Publication 155*.**

Monte Carlo Codes – DRFs

- The DRFs can be applied to PHITS, MCNP6, and Geant4 code by using the following functions:

$$D(r_T, x) = \int_E \Phi(E, r_S, x) R(r_T \leftarrow r_S, x, E) dE$$

MC code	Function	Explanation
PHITS	[T-track]	Calculate particle fluence
	[Multiplier]	Apply DRFs
MCNP6	F4 tally	Calculate particle fluence
	DE/DF card	Apply DRFs
Geant4	G4VPrimitiveScorer	Calculate particle fluence and apply DRFs

- Input files for applying the DRFs can be downloaded through MESH-PHANTOM homepage: https://mesh-phantom.com/

MESH-PHANTOM Website (https://mesh-phantom.com/)

MESH PHANTOM

Home | Phantom Information | Phantom Request | Monte Carlo Code Inputs | Q&A

Monte Carlo Code Inputs

Geant4

Geant4-v10.06.p01	.ZIP FILE	.TAR FILE
Geant4-DRF	.ZIP FILE	.TAR FILE

MCNP6

MCNP6.2	.ZIP FILE	.TAR FILE
MCNP6-DRF	.ZIP FILE	.TAR FILE

PHITS

PHITS-v3.10	.ZIP FILE	.TAR FILE
PHITS-DRF	.ZIP FILE	.TAR FILE

Thank you!