IEEE MIC 2022

STIR Users Group 10/11/2022

Connecting STIR and the SIMIND Monte Carlo Simulator

R. Gillen[1,2], S. Porter[1], M. Ljungberg[3], K. Thielemans[1]

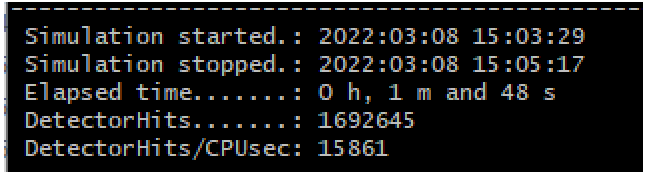
[1] Institute of Nuclear Medicine, University College London Hospital, London, UK,
[2] Marine, Medical and Nuclear Department, National Physical Laboratory, Teddington, UK
[3] Lund University, Medical Radiation Physics, Lund, Sweden

Acknowledgements: RG is completing a PhD with funding from the Industrial Strategy Challenge Fund via NPL. STIR is maintained with the support from the Computational Collaborative Project in Synergistic
Reconstruction for Biomedical Imaging, CCP SyneRBI, UK EPSRC grant EP/T026693/1.
Thanks to Dr Kjell Erlandsson and Ashley Morahan for assistance in getting started with SIMIND.

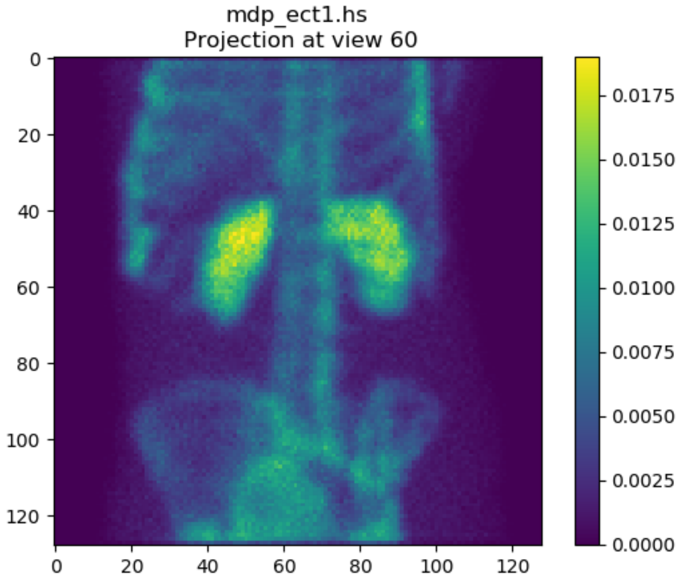
Contact Rebecca.Gillen.18@ucl.ac.uk

Introduction - SIMIND

- Monte Carlo (MC) simulation software [1]
- Simulates typical SPECT systems only
  - Less configurable but faster to set up and run than GATE

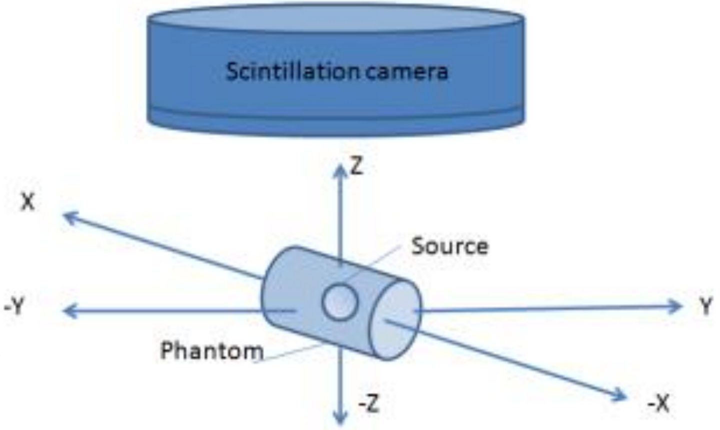


```
------------------------------------------------------
 Simulation started.: 2022:03:08 15:03:29
 Simulation stopped.: 2022:03:08 15:05:17
 Elapsed time.......: 0 h, 1 m and 48 s
 DetectorHits.......: 1692645
 DetectorHits/CPUsec: 15861
```



[1] M. Ljungberg and S.-E. Strand, “A Monte Carlo program for the simulation of scintillation camera characteristics,” *Comput. Methods Programs Biomed.*, vol. 29, pp. 257–272, 1989.

Using SIMIND with STIR



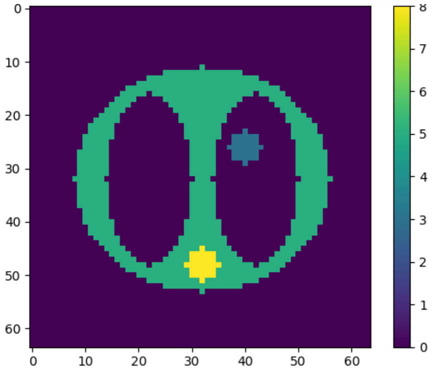
- SIMIND can define emission and density data with pre-set shapes
- More flexibility possible using “voxel based” input datasets (e.g. from STIR)
- There are differences in conventions between STIR and SIMIND:
  - Coordinate systems, attenuation/density maps, accepted input data types, interfile headers…etc

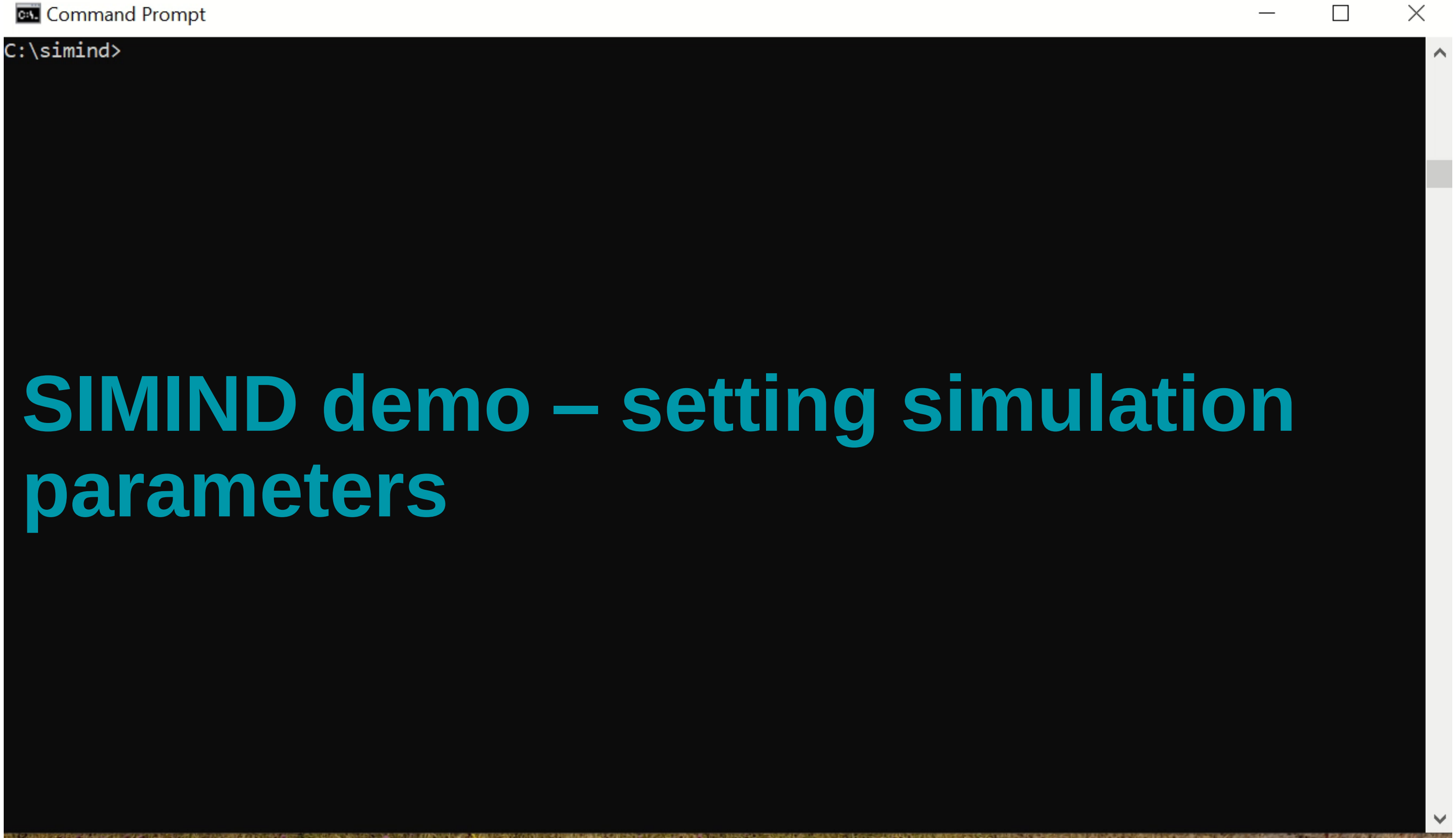
SIMIND demo – input data and other required files

- Emission data: generate in STIR and save *.v file as *.smi
- Density data: generate in STIR and save *.v file as *.dmi

16 bit integer data

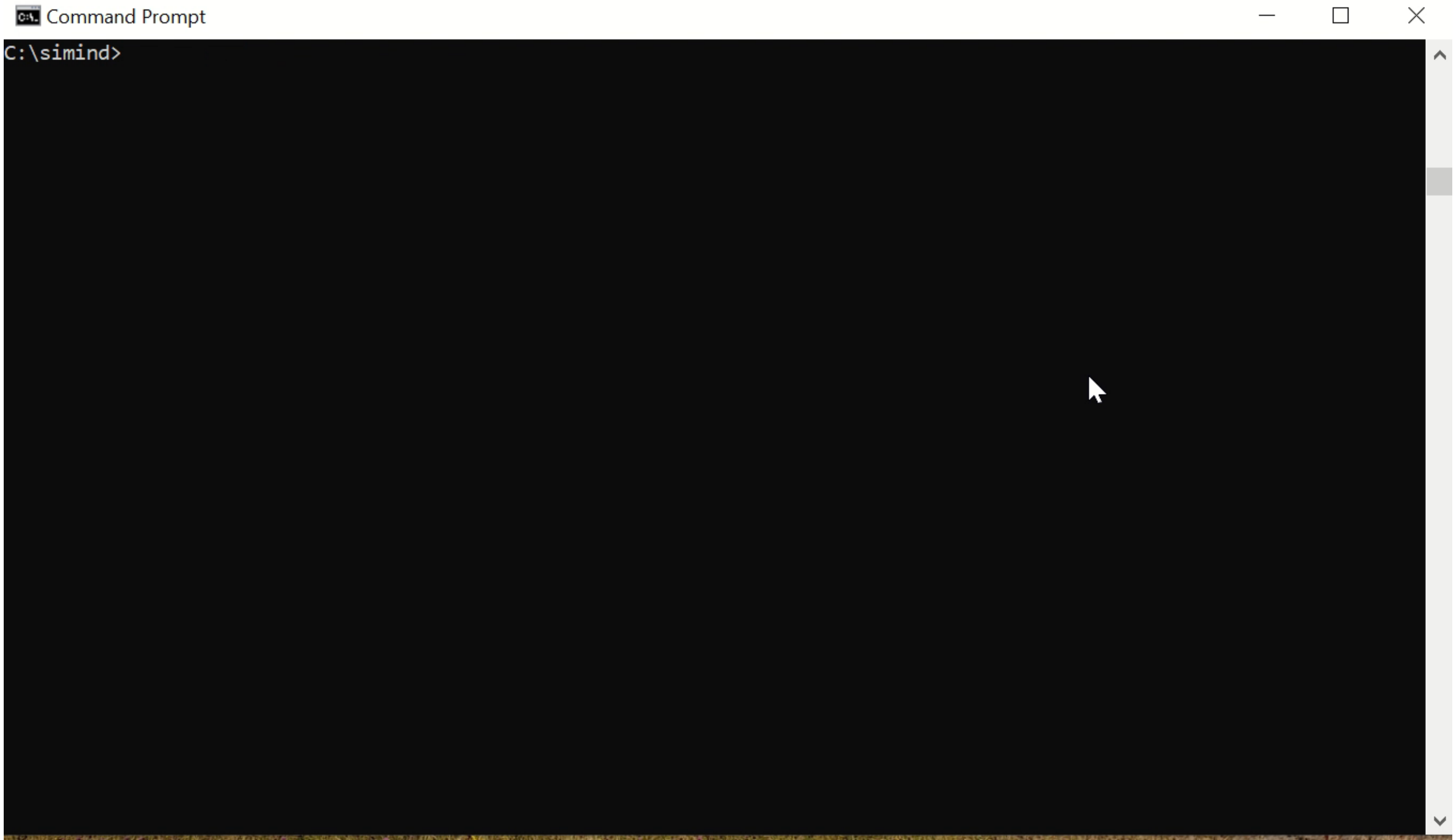
- Simulation parameters: *.smc file
- Other optional input files – see SIMIND manual



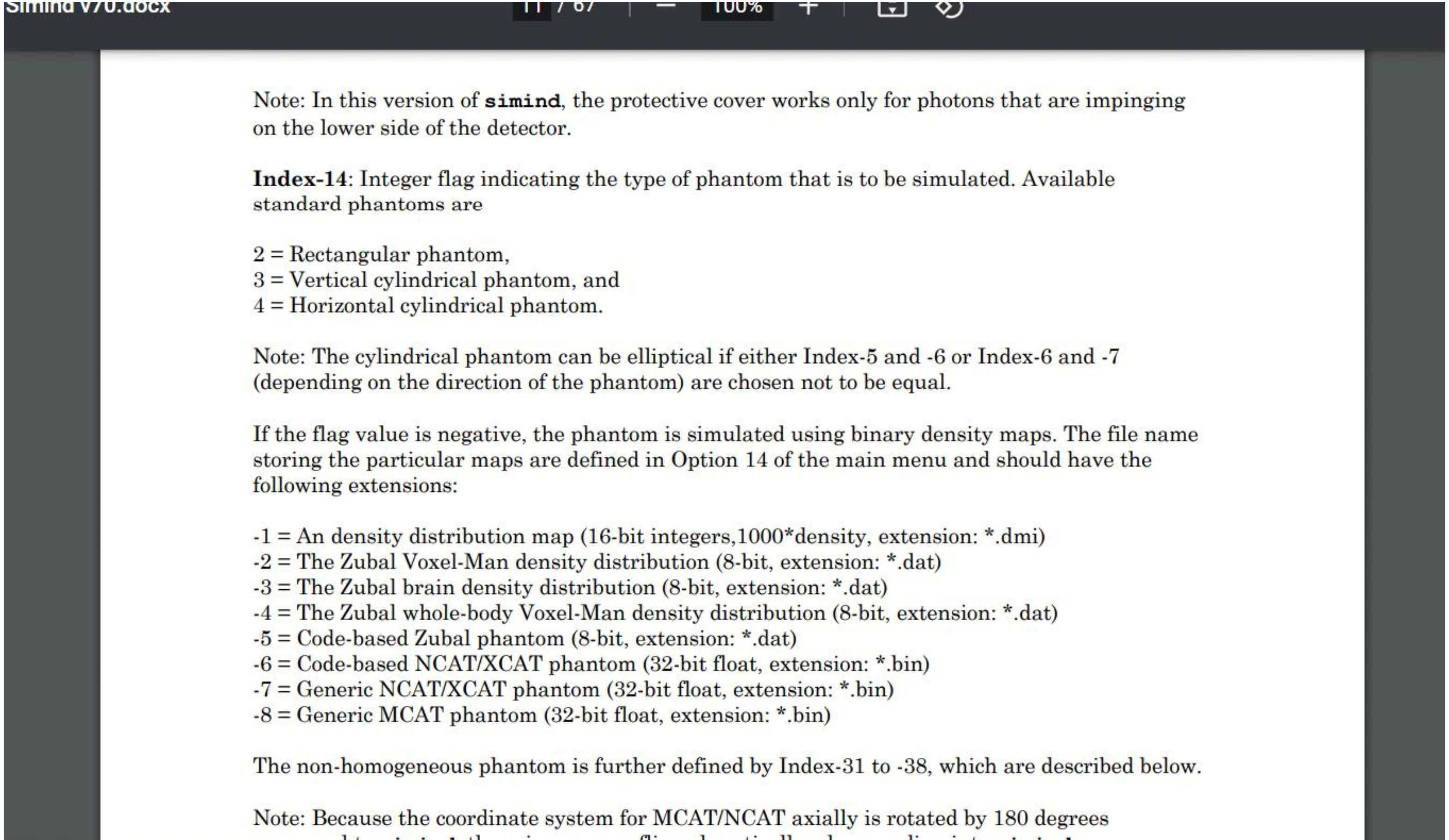


```
C:\simind>
```

SIMIND demo – setting simulation parameters



```
C:\simind>
```



Note: In this version of **simind**, the protective cover works only for photons that are impinging on the lower side of the detector.

Index-14: Integer flag indicating the type of phantom that is to be simulated. Available standard phantoms are

2 = Rectangular phantom,
3 = Vertical cylindrical phantom, and
4 = Horizontal cylindrical phantom.

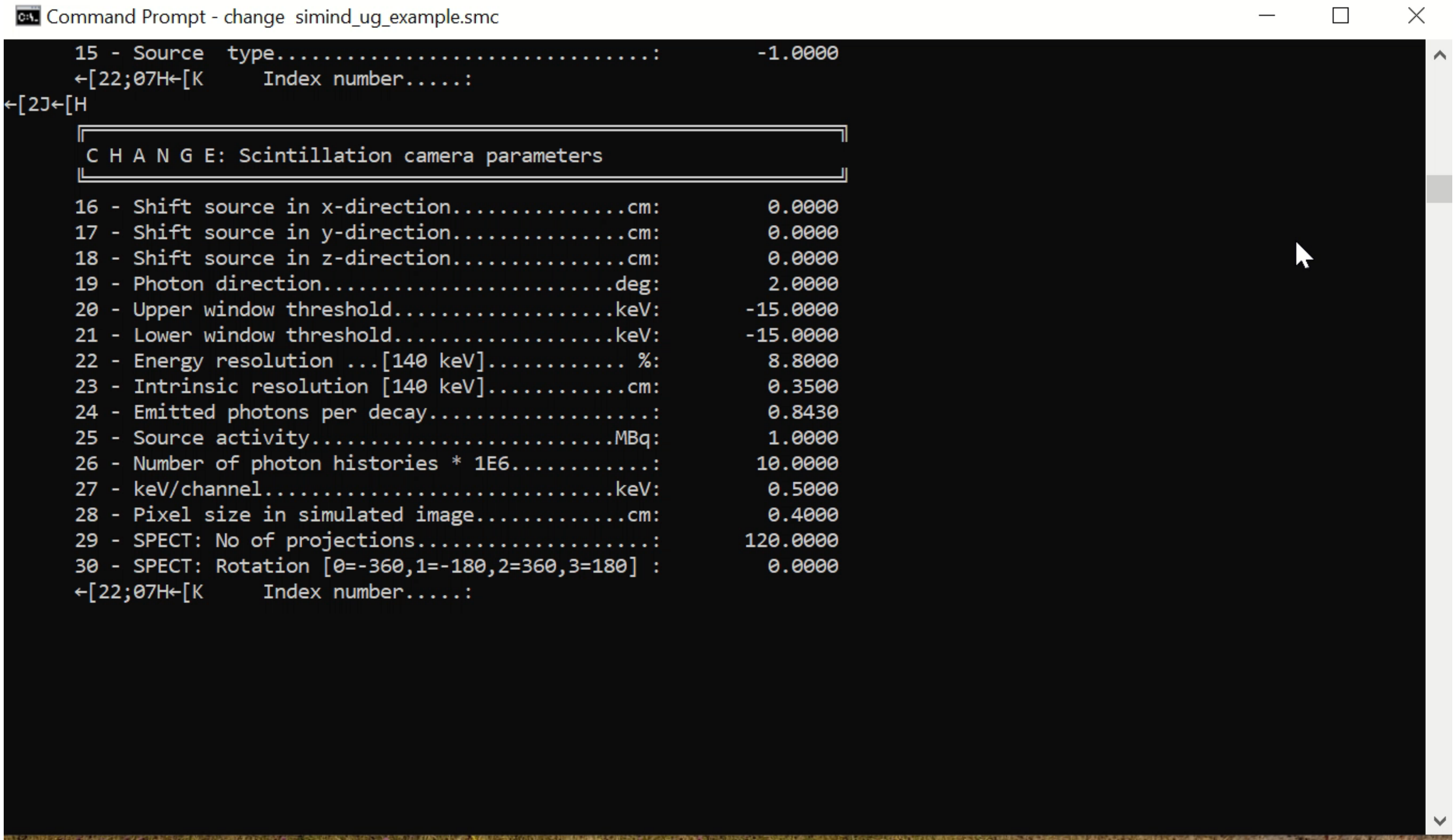
Note: The cylindrical phantom can be elliptical if either Index-5 and -6 or Index-6 and -7 (depending on the direction of the phantom) are chosen not to be equal.

If the flag value is negative, the phantom is simulated using binary density maps. The file name storing the particular maps are defined in Option 14 of the main menu and should have the following extensions:

-1 = An density distribution map (16-bit integers,1000*density, extension: *.dmi)
-2 = The Zubal Voxel-Man density distribution (8-bit, extension: *.dat)
-3 = The Zubal brain density distribution (8-bit, extension: *.dat)
-4 = The Zubal whole-body Voxel-Man density distribution (8-bit, extension: *.dat)
-5 = Code-based Zubal phantom (8-bit, extension: *.dat)
-6 = Code-based NCAT/XCAT phantom (32-bit float, extension: *.bin)
-7 = Generic NCAT/XCAT phantom (32-bit float, extension: *.bin)
-8 = Generic MCAT phantom (32-bit float, extension: *.bin)

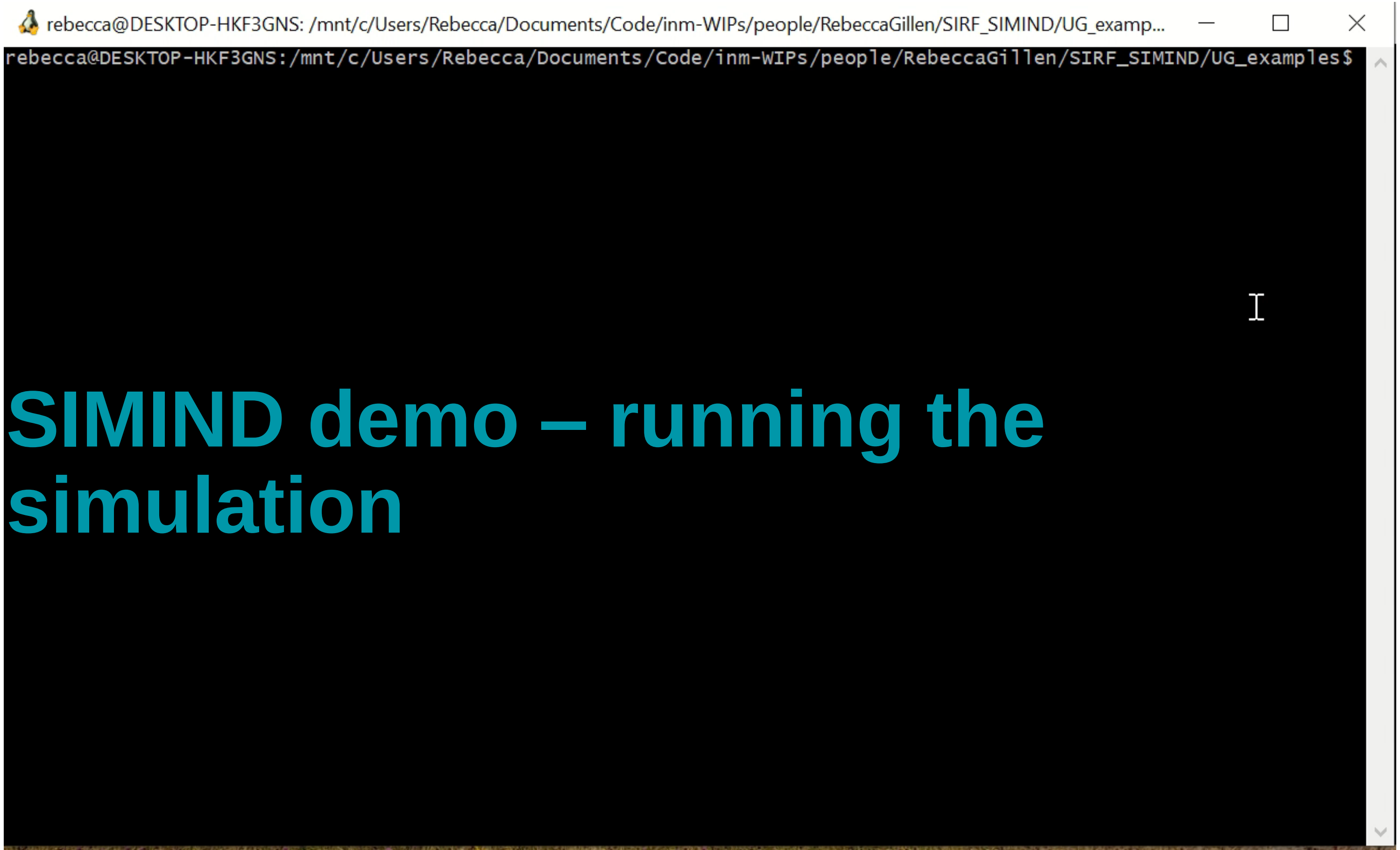
The non-homogeneous phantom is further defined by Index-31 to -38, which are described below.

Note: Because the coordinate system for MCAT/NCAT axially is rotated by 180 degrees



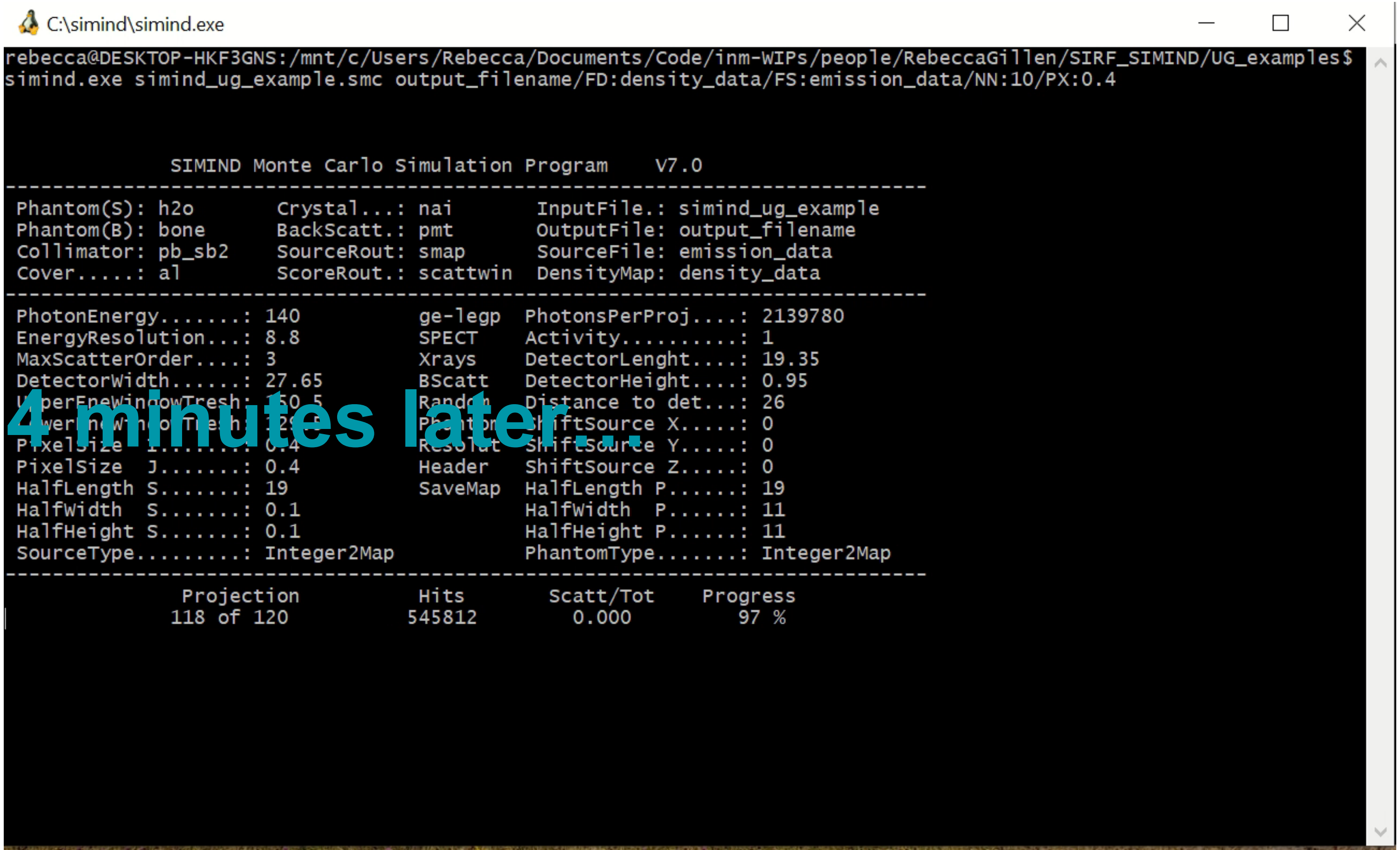
Command Prompt - change simind_ug_example.smc

```
     15 - Source  type...............................:       -1.0000
     ←[22;07H←[K     Index number.....:
←[2J←[H

      C H A N G E: Scintillation camera parameters

     16 - Shift source in x-direction..............cm:        0.0000
     17 - Shift source in y-direction..............cm:        0.0000
     18 - Shift source in z-direction..............cm:        0.0000
     19 - Photon direction........................deg:        2.0000
     20 - Upper window threshold..................keV:      -15.0000
     21 - Lower window threshold..................keV:      -15.0000
     22 - Energy resolution ...[140 keV]........... %:        8.8000
     23 - Intrinsic resolution [140 keV]...........cm:        0.3500
     24 - Emitted photons per decay..................:        0.8430
     25 - Source activity.........................MBq:        1.0000
     26 - Number of photon histories * 1E6...........:       10.0000
     27 - keV/channel.............................keV:        0.5000
     28 - Pixel size in simulated image...........cm:        0.4000
     29 - SPECT: No of projections...................:      120.0000
     30 - SPECT: Rotation [0=-360,1=-180,2=360,3=180] :        0.0000
     ←[22;07H←[K     Index number.....:
```


rebecca@DESKTOP-HKF3GNS: /mnt/c/Users/Rebecca/Documents/Code/inm-WIPs/people/RebeccaGillen/SIRF_SIMIND/UG_examp...

```
rebecca@DESKTOP-HKF3GNS:/mnt/c/Users/Rebecca/Documents/Code/inm-WIPs/people/RebeccaGillen/SIRF_SIMIND/UG_examples$
```

SIMIND demo – running the simulation



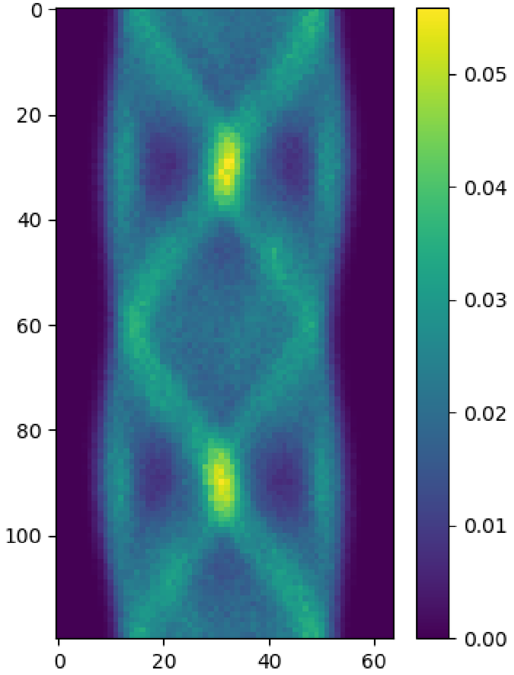
```
rebecca@DESKTOP-HKF3GNS:/mnt/c/Users/Rebecca/Documents/Code/inm-WIPs/people/RebeccaGillen/SIRF_SIMIND/UG_examples$
simind.exe simind_ug_example.smc output_filename/FD:density_data/FS:emission_data/NN:10/PX:0.4



              SIMIND Monte Carlo Simulation Program    V7.0
------------------------------------------------------------------------------
 Phantom(S): h2o        Crystal...: nai        InputFile.: simind_ug_example
 Phantom(B): bone       BackScatt.: pmt        OutputFile: output_filename
 Collimator: pb_sb2     SourceRout: smap       SourceFile: emission_data
 Cover......: al        ScoreRout.: scattwin   DensityMap: density_data
------------------------------------------------------------------------------
 PhotonEnergy.......: 140          ge-legp  PhotonsPerProj....: 2139780
 EnergyResolution...: 8.8          SPECT    Activity..........: 1
 MaxScatterOrder....: 3            Xrays    DetectorLenght....: 19.35
 DetectorWidth......: 27.65        BScatt   DetectorHeight....: 0.95
 UpperEneWindowTresh: 150.5        Random   Distance to det...: 26
 LowerEneWindowTresh: 129.5        Phantom  ShiftSource X.....: 0
 PixelSize  I.......: 0.4          Resolut  ShiftSource Y.....: 0
 PixelSize  J.......: 0.4          Header   ShiftSource Z.....: 0
 HalfLength S.......: 19           SaveMap  HalfLength P......: 19
 HalfWidth  S.......: 0.1                   HalfWidth  P......: 11
 HalfHeight S.......: 0.1                   HalfHeight P......: 11
 SourceType.........: Integer2Map           PhantomType.......: Integer2Map
------------------------------------------------------------------------------
              Projection         Hits       Scatt/Tot     Progress
              118 of 120        545812        0.000          97 %
```

4 minutes later...

SIMIND output

- SIMIND output files:
  - .h00 [header file(s)]
    - Need to convert to STIR-readable form
  - .a00 [sinogram data file(s)]
  - .res [results file – summary of params]

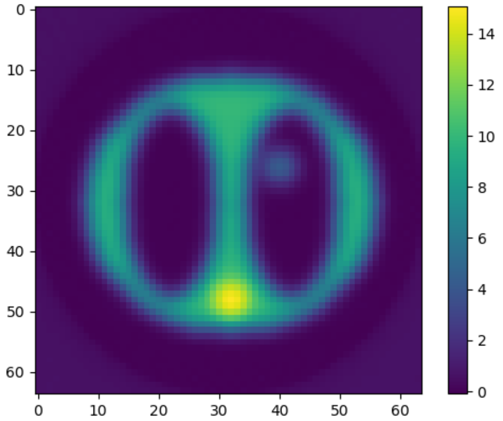
  - Other…. see SIMIND manual



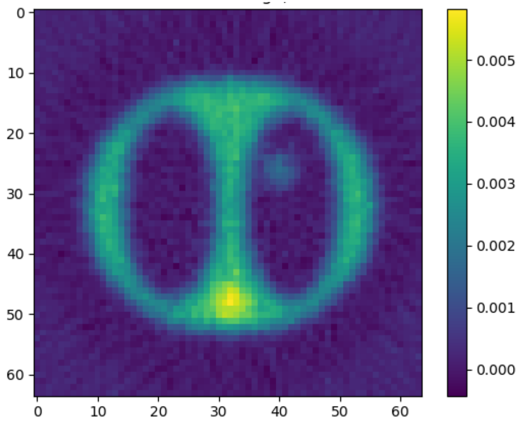
Reconstructing SIMIND output with STIR

- Convert .h00 file to .hs file using script
- Run reconstruction as normal
  - Remember to use STIR-specific attenuation map if AC applied

STIR simulation & STIR recon



SIMIND simulation & STIR recon



Examples and more resources:

- **See STIR_SIMIND Tutorial Repository:**
  - https://github.com/samdporter/STIR_SIMIND
- **Test datasets** include example *.smc file for SIMIND, and *.par and .sh scripts to generate data, convert headers to STIR-friendly format, and do recons with STIR
- **Jupyter Notebook** for those who work with STIR via Python
- **Crib sheet** includes more detail on data types, commands, and recommendations for setting SIMIND parameters for configuring the simulation to work with STIR

Conclusions/Discussion

- It would be useful for a number of applications to combine STIR and SIMIND software systems
- We have established basic requirements to connect STIR & SIMIND
  - Can now use SIMIND perform MC simulations using STIR-generated voxelwise input data
  - Can also use STIR to reconstruct data simulated with SIMIND
- This work is ongoing and we welcome discussion or feedback

Thank you for your attention!