Introduction to Basic Simulations with SIXTE

Esin G. Gulbahar on behalf of the SIXTE team

Lea Dauner, Thomas Dauser, Christian Kirsch, Maximilian Lorenz, Karan Pal, Philipp Thalhammer, Jakob Stierhof, Franziska Haluschka, Jörn Wilms

SIXTE Online Workshop, 2026

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

ERLANGEN CENTRE FOR ASTROPARTICLE PHYSICS

Outline

- Set-up
- Preparation of input
  - `simputfile`
  - `xspec`
- Running the simulation
  - `sixtesim`
- Analysing the results
  - `imgev`
  - `makespec`
  - `makelc`

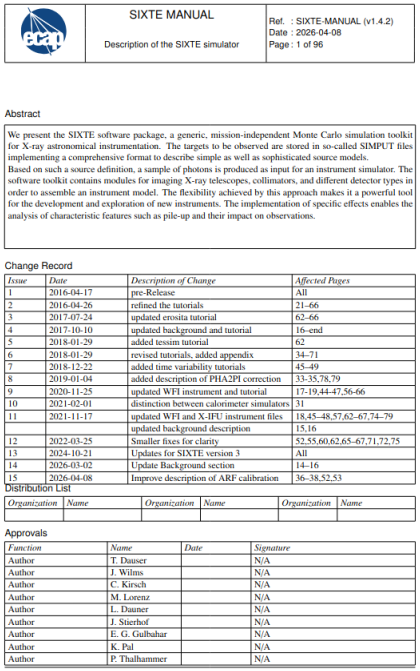
Before we begin...

Check your set-up! Does everything work?

To follow along make sure to have SIXTE, SIMPUT, and XSPEC (or HEASoft) installed!

Before we begin...

You can refer to the SIXTE simulator manual at any point! Checkout chapter 10 for basic simulation introduction.



SIXTE MANUAL

Description of the SIXTE simulator

Ref. : SIXTE-MANUAL (v1.4.2)
Date : 2026-04-08
Page : 1 of 96

Abstract

We present the SIXTE software package, a generic, mission-independent Monte Carlo simulation toolkit for X-ray astronomical instrumentation. The targets to be observed are stored in so-called SIMPUT files implementing a comprehensive format to describe simple as well as sophisticated source models. Based on such a source definition, a sample of photons is produced as input for an instrument simulator. The software toolkit contains modules for imaging X-ray telescopes, collimators, and different detector types in order to assemble an instrument model. The flexibility achieved by this approach makes it a powerful tool for the development and exploration of new instruments. The implementation of specific effects enables the analysis of characteristic features such as pile-up and their impact on observations.

Change Record

Issue	*Date*	*Description of Change*	*Affected Pages*
1	2016-04-17	pre-Release	All
2	2016-04-26	refined the tutorials	21–66
3	2017-07-24	updated erosita tutorial	62–66
4	2017-10-10	updated background and tutorial	16–end
5	2018-01-29	added tessim tutorial	62
6	2018-01-29	revised tutorials, added appendix	34–71
7	2018-12-22	added time variability tutorials	45–49
8	2019-01-04	added description of PHA2PI correction	33-35,78,79
9	2020-11-25	updated WFI instrument and tutorial	17-19,44–47,56-66
10	2021-02-01	distinction between calorimeter simulators	31
11	2021-11-17	updated WFI and X-IFU instrument files	18,45–48,57,62–67,74–79
		updated background description	15,16
12	2022-03-25	Smaller fixes for clarity	52,55,60,62,65–67,71,72,75
13	2024-10-21	Updates for SIXTE version 3	All
14	2026-03-02	Update Background section	14–16
15	2026-04-08	Improve description of ARF calibration	36–38,52,53

Distribution List

Organization	*Name*	*Organization*	*Name*	*Organization*	*Name*

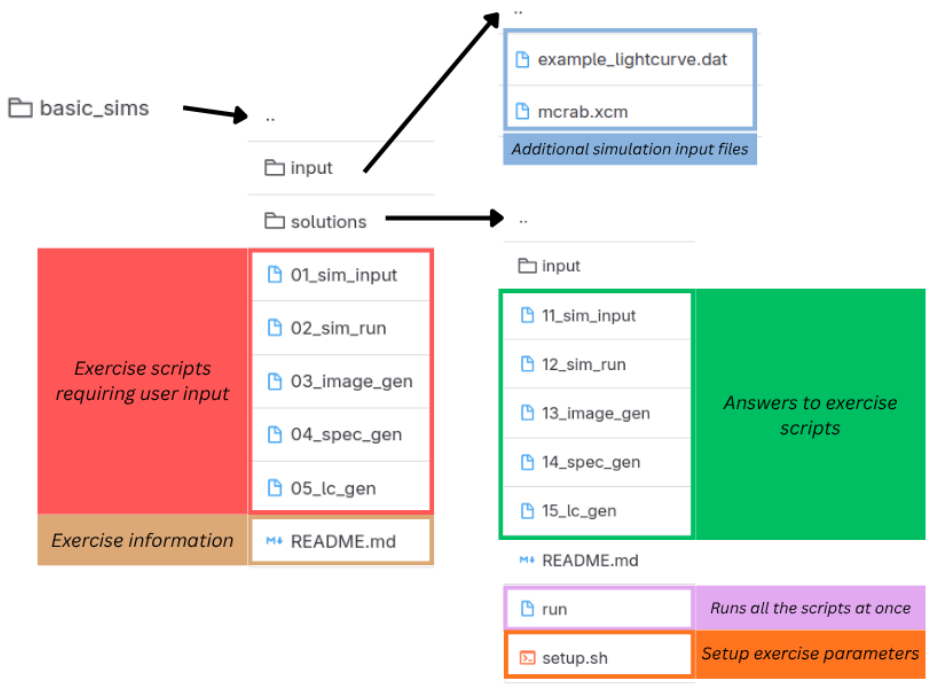
Approvals

Function	*Name*	*Date*	*Signature*
Author	T. Dauser		N/A
Author	J. Wilms		N/A
Author	C. Kirsch		N/A
Author	M. Lorenz		N/A
Author	L. Dauner		N/A
Author	J. Stierhof		N/A
Author	E. G. Gulbahar		N/A
Author	K. Pal		N/A
Author	P. Thalhammer		N/A

Available @ `https://www.sternwarte.uni-erlangen.de/sixte/sixte-manual/`

Before we begin...

You can find the exercise scripts, extra material, and answers for this presentation at the git repository of the workshop.



Scripts available @ `https://github.com/stierhofj/Sixte_Workshop_2026`

For SciServer: `Storage/sixte/sixte_volume/sixtedata/examples/sixte-workshop-26`

Before we begin...

0.1 Set environment variables:

.bashrc:

```
export SIMPUT=simputdir
export SIXTE=sixtedir
.  $SIXTE/bin/sixte-install.sh
```

.cshrc:

```
setenv SIMPUT=simputdir
setenv SIXTE=sixtedir
source $SIXTE/bin/sixte-install.sh
```

SciServer:

```
source $HOME/workspace/Storage/sixte/sixte_volume/sixte_setup.sh
```

Docker: `docker run -it fausixte/<image name>:<version tag>`

Before we begin...

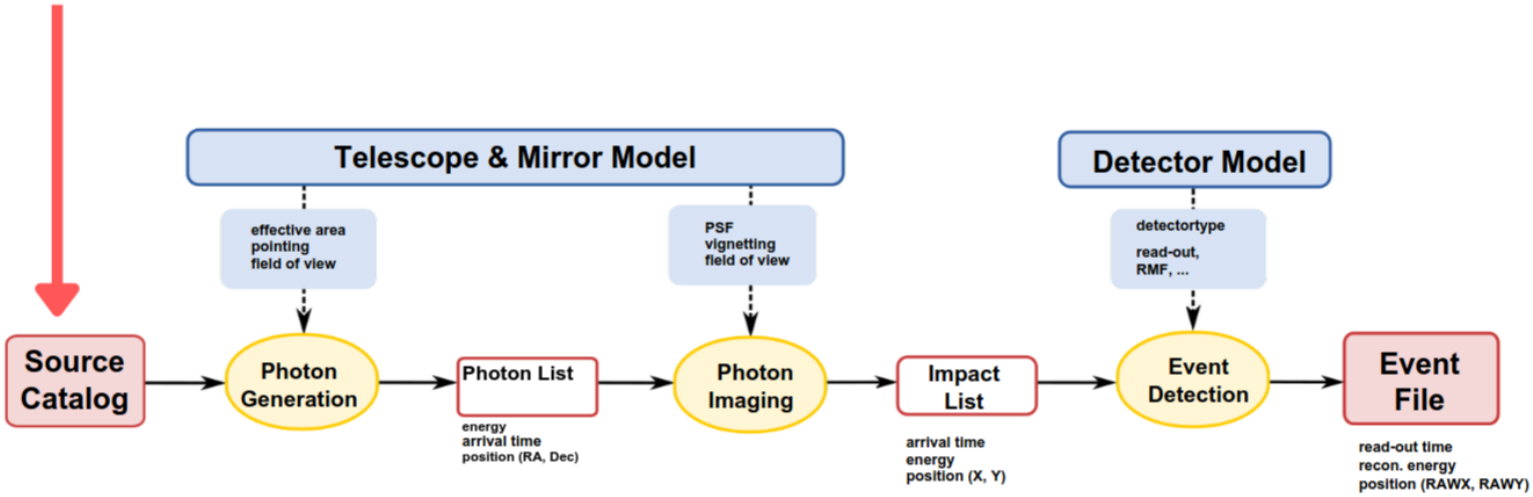
0.2 Test SIMPUT, SIXTE and XSPEC

Using `plist` command from FTOOLS we can check if SIXTE and SIMPUT will run and get information on the parameters required by the tasks:

- `plist simputfile`
- `plist sixtesim`
- `sixteversion`
- `xspec`

Important: `sixteversion` should be 3.4.0!

PART I.: Preparation of the simulation inputs using SIMPUT



Simulation Input — Preparation

1.1 Run `plist simputfile` to see the command parameters

1.2 Which parameters do you think we should change for:

Name of source	
Source position	
Source flux	
Energy	
Logarithmic energy grid & number of bins	
Define xspec file	

See manual section 10.2.2

Simulation Input — `simputfile`

1.3 Write shell script to create a `simputfile` (*see* `01-sim-input`):

Name of source	`Src_Name=<name>`
Source position	`Ra=0.0, Dec=0.0`
Source flux	`srcFlux=2.1157e-11`
Energy	`Elow=0.1, Eup=15, Emin=2, Emax=10`
Logarithmic energy grid & number of bins	`logEgrid=yes, Nbins=1000`
Define xspec file	`XSPECFile=<name>.xcm`

Manual section 10.2.2

Simulation Input — XSPEC

1.4 We need to create a spectrum for our source (we can use `xspec`)

Note for Sciserver users:

- Create folder in: `/$HOME/workspace/Storage/`*username*`/persistent/`*name_of_folder*
- Go to that directory.

1.5 Type `xspec` into a terminal to prepare an xcm-file for the source:

```
model        phabs*pegpwrlw
nH>          0.2
PhoIndex>    2.05
eMin>        2
eMax>        10
norm>        21.6
```

1.6 Check the flux: `XSPEC12>flux 2 10`

1.7 Save your model: `XSPEC12>save model mcrab.xcm`

1.8 Run `simputfile`

Manual section 10.2.2

Simulation Input — `simputfile` shell script

Example `simputfile` shell script (`01-sim-input`):

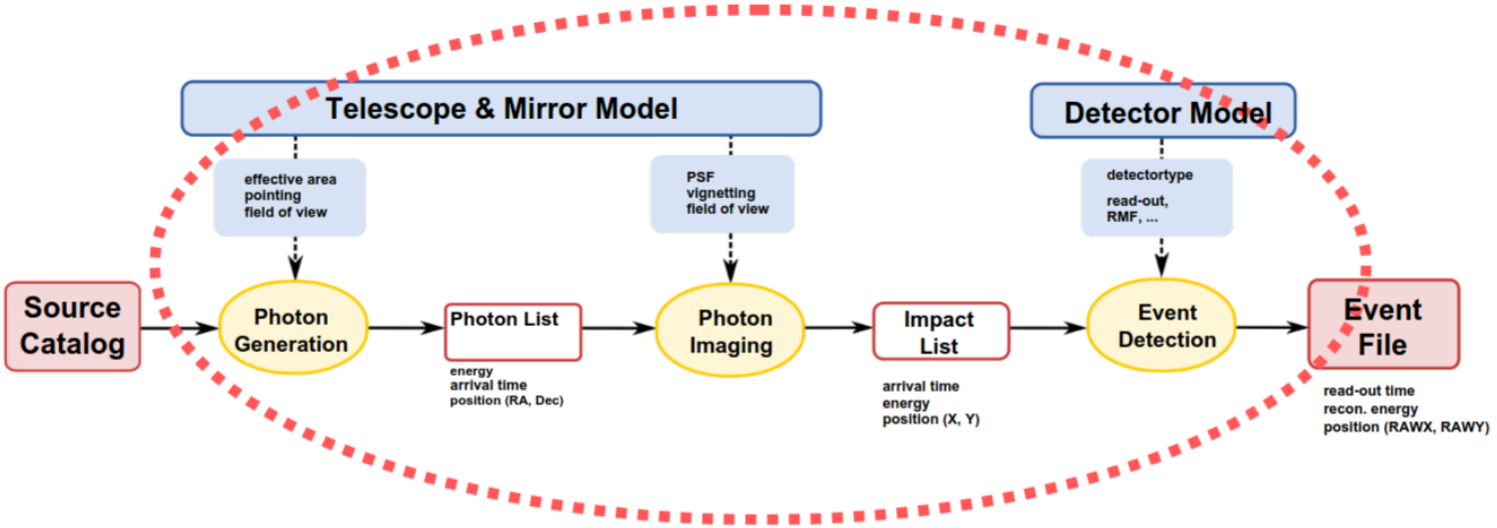
```
#!/bin/sh
base=mcrab

$SIMPUT/bin/simputfile \
      Simput=${base}.fits \
      Src_Name=first \
      RA=0.0 Dec=0.0 \
      Emin=2 Emax=10 \
      srcFlux=2.1157e-11 \
      Elow=0.1 Eup=15 \
      Nbins=1000 \
      logEgrid=yes \
      XSPECFile=${base}.xcm \
      clobber=yes
```

→ Let us investigate the output file.

Manual section 10.2.2

PART II.: Running the simulation



Simulation Run — `sixtesim`

2.1 Run `plist sixtesim` to investigate the parameters.

2.2 Which parameters do you think we should change for:

Path to .xml file	
Simput Catalog	
Telescope pointing	
Exposure time	
Output file	
Prefix for output file	

Note: XML-files are used to describe instrument properties, such as pixel size, quantum efficiency, and other detector properties.

See manual section 10.2.4

Quick glance at NewAthena instrument XMLs

WFI Example XML Set-up

```
<?xml version="1.0"?>
<instrument telescop="Athena"
instrume="WFI">
<telescope>
Information on arf, focal length, FOV, PSF,
vignetting etc.
<detector type="depfet">
Information on dimensions, rmf, and background
etc.
<readout mode="time">
Readout mechanics ...
</readout>
</detector>
</instrument>
```

X-IFU Example XML Set-up

```
<?xml version="1.0"?>
<instrument telescop="Athena"
instrume="XIFU">
<telescope>
Telescope description...
</telescope>
<detector type="microcal">
Microcalorimeter detector geometry and set-up
description...
<readout mode="event"
samplefreq="130.2083e+3"/>
X-IFU requires additional information relevant to
its specific readout scheme
</detector>
</instrument>
```

See manual sections 6.1, 6.2, Appendix C

Simulation Run — sixtesim

2.3 Use the example shell script and run a simulation for one large chip of the WFI

2.4 Run a second simulation with an off-axis pointing of the source (*see* 02-sim-run):

Path to .xml file	XMLFile=<*xmlfile*>
Simput Catalog	Simput=<*name*>.fits
Telescope pointing	Ra=0.0, Dec=0.0
Exposure time	Exposure=1000
Output file	EvtFile=<*name*>.fits
Prefix for output file	Prefix=<*name*>

Hint: Set the path of XML-directory to
$SIXTE/share/sixte/instruments/athena-wfi/wfi_wo_filter
for SciServer users:
~/workspace/Storage/sixte/sixte_volume/sixtedata/instruments/athena-wfi/wfi_wo_filter

Manual section 10.2.4

Simulation Run — sixtesim

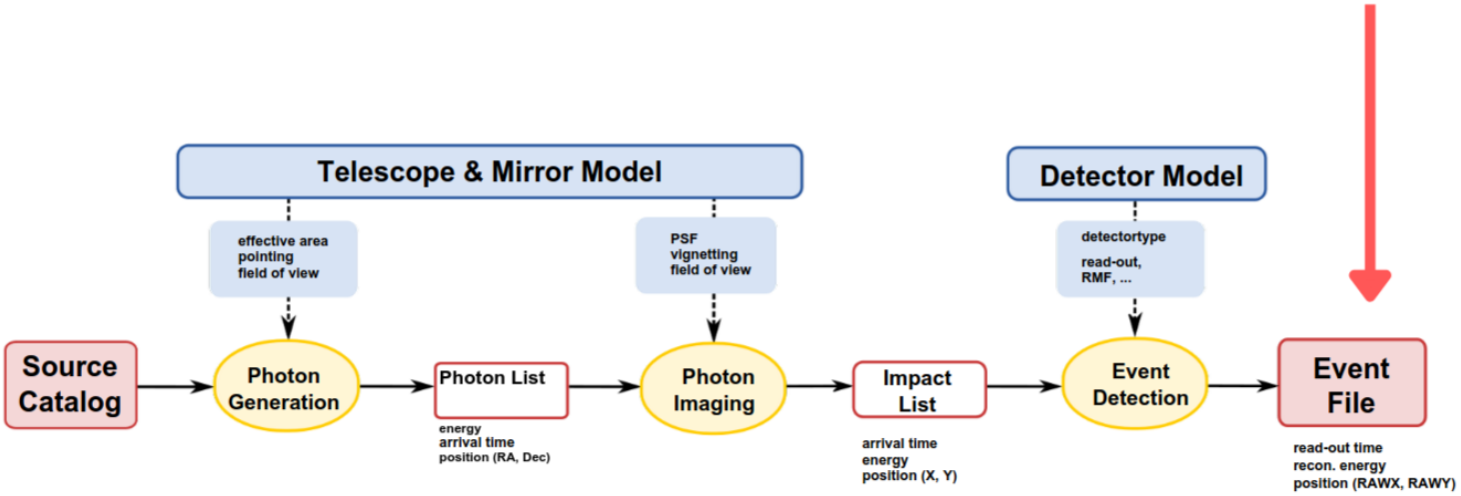
Example sixtesim shell script (02-sim-run):

```
#!/bin/sh
base=mcrab
xmldir=#xmldir
xml=${xmldir}/ld_wfi_ff_large.xml

$SIXTE/bin/sixtesim \
      XMLFile =${xml} \
      RA=0.000 Dec=0.000 \
      Prefix=sim_ \
      Simput=${base}.fits \
      EvtFile=evt_${base}.fits \
      Exposure=1000
```

Manual section 10.2.4

PART III.: Analysing the simulation



Simulation Analysis — FTOOLS

3.1 Take a look at the structure of the event file.

- Use `fstruct` and `fv` or `fdump`
- Let us investigate the meaning of the individual columns in the event file

3.2 Check if the Event File contains a significant fraction of pile-up

- Note the `PILEUP` column. The value of PILEUP is 1 if more than one photon contributed to the event.
- Hint: to get diagnostic information run `fstatistic <name_of_eventfile>.fits PILEUP -`
- **Optional Homework:** Re-do the simulations with a different source flux and check the changes in pile-up.

See manual section 10.2.5

Simulation Analysis — `imgev`

3.3 Generate an image of the event file using `imgev` (*see* `03-image-gen`)

3.4 Run `plist imgev`

- Investigate the different parameters
- Hint:
  - `Projection` – map projection used
  - `NAXIS` – number of X and Y pixels in the image
  - `CRVAL` – defines the center of the map
  - `CRPIX` – points corresponding to the optical axis
  - `CDELT` – detector spatial resolution
- Each instrument will have unique parameter values for `imgev`

For full WFI detector (4 chips)

```
NAXIS1=1063 NAXIS2=1063
CRPIX1=532 CRPIX2=532
CDELT1=-6.207043e-04
CDELT2=6.207043e-04
```

For X-IFU

```
NAXIS1=48 NAXIS2=48
CRPIX1=24.5 CRPIX2=24.5
CDELT1=-0.0015135635084518495
CDELT2=0.0015135635084518495
```

Manual section 10.2.5

Simulation Analysis — imgev

Example imgev shell script (03-image-gen):

```
#!/bin/sh

$SIXTE/bin/imgev \
      EvtFile=sim_evt_mcrab.fits \
      Image=img_mcrab.fits \
      CoordinateSystem=0 Projection=TAN \
      NAXIS1=512 NAXIS2=512 \
      CUNIT1=deg CUNIT2=deg \
      CRVAL1=0.0 CRVAL2=0.0 \
      CRPIX1=256.5 CRPIX2=256.5 \
      CDELT1=-6.207043e-04 CDELT2=-6.207043e-04 \
      history=true clobber=yes
```

→ *Note the parameter values! They are for a single chip of WFI.*

Simulation Analysis — `makespec`

3.5 Generate a spectrum using `makespec` (*see* `04-spec-gen`)
Hint: Use same xmldir as before

3.6 Use `XSPEC` **on your own machine** to plot the spectrum
→ If working on SciServer: download spectrum from SciServer or visualise it using `pyXSPEC`

Manual section 10.2.5

Simulation Analysis — makespec

Example makespec shell script (04-spec-gen):

```
#!/bin/sh
xmldir=#xmldir

$SIXTE/bin/makespec \
      EvtFile=sim_evt_mcrab.fits \
      Spectrum=spec_mcrab.pha \
      EventFilter=
      "(RA>359.95 || RA<0.05) && Dec>-0.05 && Dec<+0.05" \
      RSPPath=${xmldir} \
      clobber=yes
```

Simulation Analysis — `makelc`

3.7 Generate a lightcurve (*see* `05-lc-gen`)

- Check necessary parameters running `plist makelc`.
- What's important for a lightcurve to define?
  - → Duration
  - → Time resolution
- Ways to define a lightcurve:
  - ASCII
    - → Energy dependent → time and flux
    - → Stochastic through power spectrum → frequency and power
  - Lorentzians and zero-centered low frequency QPO

See manual section 10.3.1

Simulation Analysis — `makelc`

- Let's define a lightcurve: ASCII → **Energy dependent** → **time and flux**

3.8 Create a simput file with a TIMING extension
  – Parameter for Date → value is 55000
  – Include lightcurve →
`/simulation_examples/basic_sims/input/example_lightcurve.dat`
or `https://www.sternwarte.uni-erlangen.de/research/sixte/downloads/example_lightcurve.dat`

3.9 Run the simulation → `sixtesim`

3.10 Produce the light curve → `makelc`
  → Eventfile
  → Lightcurve
  → Duration (in s) → 1000.0
  → Time resolution (in s) → 1.0

3.11 Visualise the light curve using `fv`

Manual section 10.3.1

Simulation Analysis — makelc

Example makelc shell script (05-lc-gen):

```
#!/bin/sh
base=mcrab_lightcurve

$SIXTE/bin/makelc \
      EvtFile=sim_evt_${base}.fits \
      Lightcurve=sim_${base}.lc \
      length=1000.0 \
      dt=1.0
```

Congrats, now you know how to run a basic simulation with SIXTE!

In the next sessions, we will demonstrate how to run more advanced simulations.

SIXTE Simulator

- **Helpdesk and support**: `sixte-support@lists.fau.de`
- **Mailing list** for news and updates: `sixte-users@lists.fau.de`
- **Detailed information:** Dauser et al. (2019, A&A 630, A66) and the simulator manual
- **Form to submit questions:**

Thank you :)!