

## SIXTE Installation

A detailed description of how to install SIXTE from **source code** can be found below, as well as on our website: <https://www.sternwarte.uni-erlangen.de/sixte/installation-steps/>.

We furthermore provide SIXTE through **Docker**. Instructions on how to install Docker can be found below as well and on our website: <https://www.sternwarte.uni-erlangen.de/sixte/installation/>.

We primarily recommend installing locally on your machine. If this is not possible, you may use **Sci-Server** instead. In this case, please familiarize yourself with this setup in advance, as users often need some time to get comfortable with it.

### Installing from Source Code

#### Required Packages

The installation of the simulation software package requires the following packages:

```
gcc, g++, gfortran
gsl
cmake
cgal
boost
automake
```

If you are using macOS, we recommend using homebrew (eg.: brew install gcc, if you are not using homebrew yet, further information can be found here). On Linux we recommend using apt.

We also recommend that you have a current version of [NASA's HEASOFT package](#). Please be aware that the HEASOFT version 6.32 does currently not work with SIXTE.

#### Installing SIMPUT

To install SIMPUT and SIXTE, please visit our website and download the latest versions of SIXTE and SIMPUT: <https://www.sternwarte.uni-erlangen.de/sixte/installation/>

Once you have downloaded the SIMPUT release archive, extract it:

```
tar xvzf simput-x.y.z.tar.gz
cd simput-x.y.z
```

Then execute the following CMake commands in the top-level SIMPUT source directory:

1. Configure the project and create the build directory (where `simputdir` is your chosen installation directory):

```
cmake -S . -B build -DCMAKE_INSTALL_PREFIX=simputdir
```

2. Compile and link:

```
cmake --build build
```

### 3. Install:

```
cmake --install build
```

## Installing SIXTE

Installing SIXTE works the same way as SIMPUT. Extract the SIXTE tar archive and follow the standard build process. We recommend installing SIXTE in the same location as SIMPUT (i.e., into `simputdir`).

Execute the following CMake commands in the top-level SIXTE source directory:

4. Configure the project and create the build directory (where `sixtedir` is your chosen installation directory):

```
cmake -S . -B build -DCMAKE_INSTALL_PREFIX=sixtedir
```

5. Compile and link:

```
cmake --build build
```

6. Install:

```
cmake --install build
```

## Using the Tools

To use the tools, set the environment variables `$$SIMPUT` and `$$SIXTE` to the respective installation directories and source the `sixte-install.csh/sixte-install.sh` script:

### In a C shell (csh/tcsh):

```
setenv SIMPUT simputdir
setenv SIXTE sixtedir
source ${SIXTE}/bin/sixte-install.csh
```

### In a Bash shell (bash/sh/zsh):

```
export SIMPUT=simputdir
export SIXTE=sixedir
. ${SIXTE}/bin/sixte-install.sh
```

You can also add these lines to your `~/.cshrc` or `~/.bashrc` file. More detailed instructions can be found in the `INSTALL.txt` file in the project directory or on the SIXTE website: <https://www.stern-warte.uni-erlangen.de/sixte/installation-steps/>.

## Installing the instrument files

To use the files, please follow the steps below. They will be installed in `$$SIXTE/share/sixte/instruments` and can be used from there.

### **NewAthena WFI**

The instrument files for the *NewAthena* WFI can be downloaded [here](#).

```
mv instruments_athena-wfi-1.11.4.tar.gz $SIXTE/  
cd $SIXTE  
tar xvzf instruments_athena-wfi-1.11.4.tar.gz
```

### ***NewAthena* X-IFU**

The instrument files for the *NewAthena* X-IFU can be downloaded [here](#). This needs to be unzipped and the tar-file necessary for SIXTE simulations can be found in the folder SIXTE. Using this tar-file, please follow the instructions below.

```
mv new_athena_xifu_1.11.0.tar.gz $SIXTE/  
cd $SIXTE  
tar -xvzf new_athena_xifu_1.11.0.tar.gz --strip-components=1 \  
new_athena_xifu_1.11.0/share/sixte/instruments/new-athena-xifu \  
new_athena_xifu_1.11.0/share/sixte/instruments/new-athena-xifu.version
```

## **Installing from Docker**

SIXTE is also available on Docker Hub in the [fausixte](#) repository.

To use the Docker images, first pull the image:

```
docker pull fausixte/<image name>:<version tag>
```

Then run it interactively (press Ctrl+C to stop the container):

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

## **SIXTE on SciServer**

SIXTE is also available on [SciServer](#). Detailed instructions on how to run simulations on the platform can be found in the [SIXTE on SciServer User Guide](#).