

I N T E G R A L S C I E N C E D A T A

C E N T R E

INTEGRAL OFF-LINE SCIENTIFIC ANALYSIS



INSTALLATION GUIDE

Issue 11

September 2018

CONTENTS

1. WELCOME	3
2. INSTALLING THE SOFTWARE	4
3. INSTALLING THE INSTRUMENT CHARACTERISTICS	5
4. INSTALLING THE REFERENCE CATALOGUES	6
5. INSTALLING AND USING THE TEST DATA	6
6. OSA THROUGH DOCKER	9
7. GETTING AND INSTALLING INTEGRAL DATA	15
APPENDIX A: TROUBLESHOOTING	15
APPENDIX B: INSTALLING THE SOFTWARE FROM SOURCE CODE	16
APPENDIX C: DOCUMENT CHANGE HISTORY	22

1. Welcome

This document describes the installation of the INTEGRAL Off-line Scientific Analysis (OSA) version 11 on Linux and using docker.

Requirements:

As from OSA11, OSA depends on HEASOFT for rebinning the IBIS/ISGRI response matrix. It is thus required to have it installed an initialized (see <https://heasarc.nasa.gov/lheasoft/>)

OSA is composed of the following packages:

- Off-line Scientific Analysis Software (OSA_SW)
- Instrument Characteristics (OSA_IC)
- 'High-energy' Catalogue (OSA_CAT)
- Test Observations, which consist of a set of data and related scripts (OSA_TESTDATA)

In order to successfully analyse INTEGRAL data, you will have to download and install at least the three packages

- **OSA_SW**
- **OSA_IC**
- **OSA_CAT**

via the following URL <http://www.isdc.unige.ch/integral/analysis#Software> and retrieve observation data from the INTEGRAL archive (<http://www.isdc.unige.ch/integral/archive>)

- **i** The download and use of OSA_TESTDATA is optional. It is meant to check the correct installation of OSA by running test scripts on a small data set.
- **i** OSA software comes as a binary package or may be installed from the source code. *We strongly recommend that you first install and test the binary package for your operating system(s).*
- **i** You can use the binary packages to compile and link your own software with OSA libraries. If you have problems doing so, please read Appendix A on 'Troubleshooting'.

This document is part of the INTEGRAL analysis user documentation available at <http://www.isdc.unige.ch/integral/analysis>. In particular, detailed information about the software is provided in the instrument specific user manuals.

If you have any problems installing OSA please consult the 'Frequently Asked Questions' at <http://www.isdc.unige.ch/integral/analysis#FAQ> to check for an answer to your question or send an E-mail to the INTEGRAL help-desk at inthelp@sciops.esa.int. See the appendix on 'Troubleshooting' for information you should provide to help us solving the problem you have.

2. Installing the Software

The INTEGRAL OSA software is available via the following URL

<http://www.isdc.unige.ch/integral/analysis#Software>. Packages containing the OSA binaries for the following operating systems are provided for your convenience:

- Linux

The OSA software requires as run-time library the **libgfortran.so.3** and the **libc.so.6** libraries. These library have to be installed before the OSA software can be used. The libc.so library is most likely already installed with recent operating systems. The libgfortran.so library can be received by various packages, depending on the Linux distribution. For example:

- CentOS 5 - x86_32
- CentOS 6 - x86_64
- CentOS 7 - x86_64
- Ubuntu 16.04 - x86_64

If you are running an operating system for which no binary package is available, you can use the docker image, which is giving full portability to OSA (Section 6). Testers have successfully used the Ubuntu version on Open Suse 42.2 and 42.3. You can also download and install the source code package. Please see Appendix B for the detailed system requirements and instructions to install the OSA software from the source code.

We strongly recommend that you first install and possibly test the binary package for your operating system(s) or use the docker image.

Linux users:

Once you have downloaded the binary package for a Linux operating system,

- Move it to a directory to which you have write access and
- unpack the downloaded tar-file using e.g. the following command
`tar xfzv "Binary-file"`

Mac OS X users:

Docker is a system that enables to run a software within its native environment without modification of the users' operating system. The only requirement is to install the docker server and download the docker image for a specific task. We provide OSA and Heasoft in a docker image built on Linux CentOS7.

Mac OS X users are invited to use docker distribution, and are referred to 6 for full instructions.

Linux users.

Resulting from the above command(s) you will find the directory `osa11` containing all the OSA software. This directory is referred to as `ISDC_ENV` in the INTEGRAL OSA user documentation.

- i** You will have to set the environment variable ISDC_ENV to the full pathname of this directory to use the just installed OSA software. Details on setting up the run-time environment can be found in the cookbook sections of the instrument specific OSA user manuals.

For any platform, you need to install the ‘Instrument Characteristics’ and the ‘Reference Catalogue’. Their installation is described in the following two sections. Then, you need to fetch data and perform the analysis.

3. Installing the Instrument Characteristics

Since OSA 8.0, the IC data is no more provided as a pre-packaged tar-file nor via a script to download the files.

1. Define an archive directory to store the instrument characteristics and the reference catalogues. Then execute one of the following commands:

```
setenv REP_BASE_PROD __archive_directory__                                -- for csh family  
REP_BASE_PROD=__archive_directory__ ; export REP_BASE_PROD -- for sh family
```

(If the setenv commands fail with a message like: ’setenv: command not found’ or ’setenv: not found’ please read the appendix on ‘Troubleshooting’.)

2. with the following command you will retrieve the IC data from the ISDC server :

```
rsync -Lzrtv isdcarc.unige.ch::arc/FTP/arc_distr/ic_tree/prod/ $REP_BASE_PROD
```

A total disk space of some 3+ GBytes is needed to install the full IC data containing all INTERAL instruments.

⚠ The file names of the IC data are significant and must not be changed. Otherwise internal references in the IC data will be broken.

To learn more about ’ISDC compliant Data Repositories’ (REP_BASE_PROD) and ’Instrument Characteristics’, please refer to the OSA User Manuals (especially the ’Introduction to the INTEGRAL Data Analysis’) at the following URL <http://www.isdc.unige.ch/integral/analysis>

4. Installing the Reference Catalogues

The INTEGRAL reference catalogue and the OMC reference catalogue are available via the following URL <http://www.isdc.unige.ch/integral/analysis#Software>.

A total disk space of 85 MB is needed to install the two catalogues.

Once you have downloaded the OSA_CAT package,

- cd to a directory to which you have ‘write’ access.
- i** Typically, the directory where you unpack the reference and OMC catalogue is as well used for the installation of the instruments characteristics, see above.
- Unpack the downloaded tar-file using e.g. the following command
`gzip -dc osa_cat-VV.tar.gz | tar xvf -`

where VV is the available version number, e.g. 41.

To learn more about ’ISDC compliant Data Repositories’, ’OSA Catalogues’ and DOLs, please refer to the OSA User Manuals (especially the ’Introduction to the INTEGRAL Data Analysis’) at the following URL <http://www.isdc.unige.ch/integral/analysis>

5. Installing and Using the Test Data

Now you may want to test the installed OSA software. To be able to do so, OSA comes with a test data packages running an analysis on small data sets from Crab observations for 5 instruments. Those packages are available via the following URL
<http://www.isdc.unige.ch/integral/analysis#Software>

A total disk space of about 3 GB is needed to install the test data packages. After running the test scripts, the disk space required grows to about 5 GB.

2. Installing the Test Data

Once you have downloaded the test data package,

1. Create a directory to which you have write access and `cd` to this directory.
2. Unpack the downloaded tar-file using e.g. the following command
`gzip -dc osa_testdata-11.0.tar.gz | tar xvf -`
3. `cd` to the `testdata` sub-directory that was created via the above command. This directory will be referred to as `REP_BASE_PROD` for the rest of this section.
4. Create links to the already downloaded OSA_IC package in the archive directory.

```
ln -s __archive_directory__/ic ic
mkdir idx                                     - if the idx directory does not exist yet
cd idx
ln -s __archive_directory__/idx/ic ic
```

2. Running the Test Scripts

Once a consistent set of the OSA software, IC, CAT and the test data packages has been successfully installed, the available test scripts can be used to verify the correct installation, set-up and functioning of the INTEGRAL Off-line Scientific Analysis.

To do that, the following command sequence should be followed

1. setenv ISDC_ENV __directory_of_OSA_SW_installation__ – for csh-family
export ISDC_ENV=__directory_of_OSA_SW_installation__ – for sh-family
e.g. “/opt/osa/”
2. setenv ISDC_REF_CAT __DOL_of_the_reference_catalogue__
e.g. “__directory_of_cat_installation__/cat/hec/gnrl_refr_cat_0041.fits[1]”

⚠ The double-quotes (“”) are important if you are using a shell of the csh-family. Otherwise the shell will try to interpret the [1] as a regular expression.
3. setenv ISDC_OMC_CAT __DOL_of_the_OMC_catalogue__
e.g. “__directory_of_cat_installation__/cat/omc/omc_refr_cat_0005.fits[1]”

⚠ The double-quotes (“”) are again important (see above).
4. setenv REP_BASE_PROD __TEST_DATA_install_dir__/testdata
5. source \$ISDC_ENV/bin/isdc_init_env.csh – for csh-family
. \$ISDC_ENV/bin/isdc_init_env.sh – for sh-family

⚠ The ‘source’ or the leading ‘.’ are important!
6. cd __TEST_DATA_install_dir__/testdata
7. Make a symbolic link to the `ic` and `idx` locations on your system
`ln -s _my_location/ic`
`ln -s _my_location/idx`
8. cd ..
9. It’s always a good idea to remove old parameter files that are stored in your user pfiles directory – usually the pfiles subdirectory in your home directory. You actually only need to remove the entry-level parameter files, `ibis_science_analysis.par`, `jemx_science_analysis.par`, `omc_science_analysis.par` or `spi_science_analysis.par`
10. make test INSTR=*instr*
instr should be either `ibis`, `picsit`, `jemx`, `spi` or `omc` to perform the test for a specific instrument.
The test runs of the different instruments are independent of each other and can be launched in any order.
11. make dircmp INSTR=*instr* – optional. See below for details.
instr should be either `ibis`, `picsit`, `jemx`, `spi` or `omc` to verify the test for a specific instrument.

The “`make test INSTR=instr`” command will run the data analysis for the test observation. If you have the FTOOL `fdiff` available on your system, you may run as well the “`make dircmp INSTR=instr`” command. A comparison of the output data of the current test run with the reference output data will be performed. The reference output data is included in the OSA test data package in the directory ‘*instr_outref*’. It was generated running OSA on a 64-bit Linux

platform. The `make test` command should exit with a return code of 0. Users may check the return code by typing `echo '$?'` immediately after the `make test` or `make dircmp` command. If `make test` terminated with an error and you cannot figure out the problem yourself, please consult the 'Frequently Asked Questions' at <http://www.isdc.unige.ch/integral/analysis#FAQ> to check for a solution to your problem or send an E-mail to the INTEGRAL help-desk at inthelp@sciops.esa.int. See the appendix on 'Troubleshooting' for information you should provide to help us solving the problem you have.

6. OSA through Docker

To ensure portability of OSA with evolving operating systems, we provide an OSA docker image, suitable for any platform for which docker is available. Docker is a tool in which one can ship an OS with the SW. In ourcase, it allows to run OSA in a Linux environment, independently from the user's OS. As Mac OS-X platforms would require significant software development for portability, this has become not feasible with the available resources after more than 15 years since the launch of INTEGRAL. Therefore, we discontinued our effort to provide Mac OS binary images. Here, we provide a brief introduction to the use of docker for OSA.

6.1 General installation of OSA docker image to run the analysis

First of all, it is necessary to **install Docker on your platform**. We do not provide support on this, instructions can be retrieved from the docker project to which we refer. Instructions are specific to your OS distribution (see <https://docs.docker.com/install/>). For a quick use, we suggest to read the first two paragraphs below (Graphical interface for Linux and Mac, Downloading the OSA docker image) and then just to Section 6.4 .

Graphical interface, special instructions for Mac users

The interaction with the graphical environment is platform dependent and needs to be properly set-up. For Linux you need to make sure that X11 is installed and that you are in a graphical session. For OSX, the following instructions have been developed by CERN for the “root” program and are also valid for OSA, which uses “root” for its graphical interface (we refer for details an updates to the page <https://hub.docker.com/r/rootproject/root-ubuntu16/>).

1. *Make sure that [XQuartz](https://www.xquartz.org) is installed. If it isn't install it from here: <https://www.xquartz.org>*
2. *Open XQuartz, go to XQuartz, Preferences, select the Security tab, and tick the box "Allow connections from network clients". Then, quit XQuartz.*
3. *Open Terminal and issue the following command:
`ip=$(ifconfig en0 | grep inet | awk '$1=="inet" {print $2}')`
This will grab your IP address on the local network.*
4. *Run `echo $ip` to make sure it was successful. If nothing is displayed, replace `en0` by `en1` or a higher number in the above command. Try again until it works.*
5. *When you have successfully defined the variable `ip` to store your IP address, issue the command: `xhost + $ip`
This will start XQuartz and whitelist your local IP address. A message should be displayed saying that the IP address has been “added to the access control list”.*

Downloading the OSA docker image

Download the Docker image from the public Docker Hub to which ISDC has uploaded the software. (This requires 2.6 Gb of disk space and an Internet connection.)

From the Terminal command line type:

```
docker pull integralsw/osa:11.0
```

(Note that the number after the colon, here 11.0, is a tag to the version. There is also a 10.2 docker image if you want to download that version. Omitting the tag will result in downloading the latest version.

Docker images will be stored in a system directory. Images can be listed from the command line with the instruction:

```
docker image
```

(see documentation for details, e.g.,

<https://docs.docker.com/engine/reference/commandline/images/>)

The command:

```
docker rmi "Image-name"
```

will remove an image.

Defining the arguments and options

Any OSA command can be executed in the docker container after some settings.

The container acts as a separate OS in which the OSA initialization is performed internally with the script “/init.sh”. However, we have to provide through our local OS input data and a place to write the results. This can be done in the way explained below.

Any OSA command inside the docker image is run from the command line as:

```
docker run [arguments] integralsw/osa:11.0 command
```

Let's first create temporary directories to mount the `home` and `pfiles` to be visible within the docker image. This will avoid “polluting” our working directory with unnecessary outputs (`$$` is a variable that stores the shell's process ID number¹):

```
mkdir -pv /tmp/osa-home-$$
mkdir -pv /tmp/osa-home-$$-pfiles
```

These directories will be identified and mounted using the docker options

```
-v /tmp/osa-home-$$-pfiles:/pfiles
-v /tmp/osa-home-$$:/home/integral
```

It is necessary to specify the user name in the docker image to be the same as in the native OS, which is done with the options

```
--rm -it --user `id -u`
```

The graphical interface options are defined as

```
-e DISPLAY=$DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix (Linux)
-e DISPLAY=$ip:0 -v /tmp/.X11-unix:/tmp/.X11-unix (OS-X)
```

We need an OSA compliant directory, as described in

https://www.isdc.unige.ch/integral/download/osa/doc/11.0/osa_um_intro/index.html where to download and store IC files and data following the instructions in Sections 4 and 7. This directory will be referred to using the environmental variable `REP_BASE_PROD`, and will contain the folders `ic`, `idx`, `scw`, `aux`, as well as the output data (see the end of this section for an example on how to separate input and output directories).

¹ At the terminal, enter `echo $$`, and then `ps`. You will see that the active shell in which you are writing these commands has the process number given by `$$`.

We can mount this directory with the option:

```
-v $PWD:/home/integral/work
```

In conclusion, these are the arguments and options we need to run OSA 11 within the docker:

Linux:

```
--rm -it --user $(id -u) \
-e DISPLAY=$DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix \
-v tmp/osa-home-$$_pfiles:/pfiles \
-v /tmp/osa-home-$$_:/home/integral \
-v $PWD:/home/integral/work
```

OS-X (for a graphical interface, remember to issue `xhost + $ip` on your shell before running OSA):

```
--rm -it --user $(id -u) \
-e DISPLAY=$ip:0 -v /tmp/.X11-unix:/tmp/.X11-unix \
-v tmp/osa-home-$$_pfiles:/pfiles \
-v /tmp/osa-home-$$_:/home/integral \
-v $PWD:/home/integral/work
```

(Depending on your system, you might need to set `-e DISPLAY=$ip:1`, `-e DISPLAY=$ip:2` or using a number even greater than 2.)

6.2 Running OSA Commands

We are now ready to work on the <command> to run OSA tasks. Since we need to define some environment variables, it is better to run a shell that will 1) define variables, 2) initialize OSA, and 3) cd into the working directory.

Our command will then be:

```
bash -c
`export HOME=/home/integral
REP_BASE_PROD=/home/integral/work;
source /init.sh;
cd $REP_BASE_PROD;
<our OSA commands>'
```

Catalogs are shipped with the docker image. However, it is possible to use custom catalogs. What is needed is to place them in a folder mounted in the docker image.

We suggest to place the catalogs in the standard locations in your **local** file system:

`$PWD/cat/my_gnrl_refr_cat.fits` and `$PWD/cat/my_omc_refr_cat.fits`
And use them in the call within docker as

```
bash -c
`export HOME=/home/integral
REP_BASE_PROD=/home/integral/work;
ISDC_REF_CAT=/home/integral/work/cat/my_gnrl_refr_cat.fits;
INTEGRAL Off-line Scientific Analysis - Installation Guide
```

```

ISDC_OMC_CAT=/home/integral/work/cat/my_omc_refr_cat.fits;
source /init.sh;
cd $REP_BASE_PROD;
<our OSA commands>

```

We are ready to run `og_create` program to build our OSA observation group. We use IBIS as a example and assume that we have stored our list of science windows in the file `list_sdw.txt`.

The OSA command is

```

og create idxSwg=list_sdw.txt ogid=my_og \
baseDir=$REP_BASE_PROD instrument=IBIS

```

However, we have to prepend all the necessary arguments for docker to run properly. For Mac OS-X:

```

docker run \
--rm -it --user `id -u` \
-e DISPLAY=$ip:0 -v /tmp/.X11-unix:/tmp/.X11-unix \
-v /tmp/osa-home-$$.pfiles:/pfiles -v /tmp/osa-home-$$/home/integral \
-v $PWD:/home/integral/work \
integralsw/osa:11.0 \
bash -c 'export HOME=/home/integral \
REP_BASE_PROD=/home/integral/work; \
source /init.sh; cd $REP_BASE_PROD; \
og_create idxSwg=list_sdw.txt ogid=my_og \
baseDir=$REP_BASE_PROD instrument=IBIS obs_id=""'

```

This will create a directory `obs/my_og`, with the observation group for OSA. It is now possible to run `ibis_science_analysis` using

```

docker run \
--rm -it --user `id -u` \
-e DISPLAY=$ip:0 -v /tmp/.X11-unix:/tmp/.X11-unix \
-v /tmp/osa-home-$$.pfiles:/pfiles -v /tmp/osa-home-$$/home/integral \
-v $PWD:/home/integral/work \
integralsw/osa:11.0 \
bash -c 'export HOME=/home/integral \
REP_BASE_PROD=/home/integral/work; \
source /init.sh; cd $REP_BASE_PROD/obs/my_og; \
ibis_science_analysis'

```

The graphical interface opens and you can run the `ibis_science_analysis` as explained in the relative the user manual (remember to issue `xhost+$ip` in the shell before running the commands). With these commands, we moved into our observation group within the docker container to run `ibis_science_analysis`. Results will be locally stored in `$PWD/obs/my_og`.

To reduce data from other instruments, you should change the call to `og_create` appropriately and then call `jemx_science_analysis`, `spi_science_analysis`, or `omc_science_analysis`.

6.4 Data and catalog separated from output location

In case you would like to run your analysis in a location different from the place you stored the archive, IC tree, and catalog, we suggest that you mount a separate volume to the docker image and you create symbolic links **inside the docker image**.

Suppose that in your local file system you have stored data, IC tree, and catalog in

```
/integral/arc/aux  
/integral/arc/scw  
/integral/arc/ic  
/integral/arc/idx  
/integral/arc/cat
```

You should add the option `-v /integral/arc:/archive` to **all** your docker commands and create symbolic links before running `og_create`, for instance with the following command:

```
docker run \  
--rm -it --user `id -u` \  
-e DISPLAY=$ip:0 -v /tmp/.X11-unix:/tmp/.X11-unix \  
-v /tmp/osa-home-$$_pfiles:/pfiles -v /tmp/osa-home-$$_:/home/integral \  
-v $PWD:/home/integral/work \  
-v /integral/arc:/archive \  
integralsw/osa:11.0 \  
bash -c 'export HOME=/home/integral \  
ISDC_REF_CAT=/home/integral/work/cat/hec/gnrl_refr_cat_0041.fits \  
ISDC_OMC_CAT=/home/integral/work/cat/omc/omc_refr_cat_0005.fits \  
REP_BASE_PROD=/home/integral/work; \  
source /init.sh; cd $REP_BASE_PROD; \  
ln -sf /archive/* .'
```

This will create internally in docker the links `aux`, `scw`, `ic`, `idx`, and `cat` so that you can run `og_create` and your analysis commands from an OSA compliant location. Note that with this command, all that is stored locally in `/integral/arc` will be linked and that in the local file system, these will be broken links, unless locations are identical in the local file system and in the docker image.

One should remember to add the option `-v /integral/arc:/archive` to mount the volume and the declaration of environmental variables for the catalogs

```
export ISDC_REF_CAT=/home/integral/work/cat/hec/gnrl_refr_cat_0041.fits  
ISDC_OMC_CAT=/home/integral/work/cat/omc/omc_refr_cat_0005.fits  
to all your following commands such as og_create, ibis_science_analysis, spe_pick  
....
```

6.4 Contributed script

We provide also a script to wrap commands around docker: see

<https://gitlab.astro.unige.ch/savchenk/osa-docker/blob/master/osa-docker.sh>

In this script, you can define the local data location and the local archive location (they can be the same) by setting the environmental variables

```
export CURRENT_IC=/isdc/arc/rev_3
```

```
export REP_BASE_PROD=/isdc/arc/rev_3/
```

Then, you can create the science window list in your local directory and OSA as, for example:

```
./osa-docker.sh og_create scw.list IBIS test ./
./osa-docker.sh cd obs/test\; ibis_science_analysis
```

The graphical interface in the script needs to be adapted to your platform (Linux or Mac OS-X), by setting the display variable for Mac OS-X while executing the script as (in bash):

```
DISPLAY=$ip:0 ./osa-docker.sh cd obs/test\; ibis_science_analysis
(you might need $ip:1 or higher on some Mac OS-X)
```

7. Getting and Installing Integral Data

1. Public Data

Requests for public INTEGRAL data can be made via the ISDC archive browser at the following URL:
<http://www.isdc.unige.ch/integral/archive>

Once you have selected the data you want to download, you will receive an e-mail with detailed instructions on how to install the public INTEGRAL data.

2. Private Data

Once the data for your observation has been fully processed, the PI will receive an e-mail with detailed instructions on how to download and install the private INTEGRAL data. The PI will receive first the message for the NRT data and then for the consolidated version. Since AO12, all the INTEGRAL data are available straightforwardly for sources that were not assigned to a scientist, with the exception of the observations assigned to PIs from the Russian federation. To download these data, it is necessary to visit the page: http://www.isdc.unige.ch/integral/archive#NRT_CONS_data

and agree to the “terms and conditions”. Starting from AO13, all the consolidated data are made available, regardless of the PI provenance. For example, scientists interested in a new transient appearing on a private observation can download and use all the available INTEGRAL data for their publication.

Appendix A: Troubleshooting

1. Problems with the `setenv` command

If the `setenv` command fails with a message like: ‘`setenv: command not found`’ or ‘`setenv: not found`’, you are very likely not using a command interpreter (shell) of a C-Shell flavour (e.g. `csh` or `tcsh`). Please replace the command `setenv my_variable my_value` by the following command sequence
`my_variable=my_value ; export my_variable`

 No blank characters ‘ ’ must be present in the `my_variable=my_value` command. Otherwise you will get an error message like ‘`my variable not found`’.

2. Problems with the `source` command

If the `source` command fails with a message like: ‘`source: not found`’ or you get many error messages indication that the command actually failed, you are very likely not using a command interpreter (shell) of a C-Shell flavour (e.g. `csh` or `tcsh`). Please replace the command ‘`source my_script`’ by the following command ‘`. my_script`’

 You really must type the ‘.’ followed by a blank character before the name of the script. Otherwise the setting of the environment variables inside the script will not be kept after the script terminated.

3. Problems linking your own software with OSA libraries

If you are using a binary installation and you are having problems to link your own software with OSA libraries the following command sequence might help:

```
cd $ISDC_ENV ; mv lib lib_orig ; ln -s libalt lib
```

If you still have problems, you probably need to compile the OSA software from the source code and link your software using the newly built libraries. However, we suggest to use docker containers.

4. Reporting problems to the INTEGRAL help desk

For an efficient trouble-shooting, please include the following information in your E-mail to the help-desk inthelp@sciops.esa.int.

- the version of the OSA software package
- the name and version of the operating system. Type `uname -a` to retrieve this information. Linux users should in addition provide the contents of file `/proc/version`.
- the version of the OSA_IC package used
- the version of the OSA_CAT package used
- if you are having a problem running the test scripts:
 - in the terminal window where the make test was run, execute the `env` or `printenv` command and provide the output to the ISDC.
 - the contents of the following files `make_test_failed.log` or `make_dircmp_failed.log` depending on whether the ‘`make test`’ or ‘`make dircmp`’ failed.
- if you are building the OSA software from the source code:
 - the name and version of the compiler(s). For the GNU compilers, type ‘`gcc -v`’ and ‘`g++ -v`’
 - the name and version of the make program. Type ‘`make -v`’ or ‘`gmake -v`’ depending on the name of the GNU make on your system.

Appendix B: Installing the Software from Source Code

If you are running an operating system for which no binary package is available, or you cannot use the binary packages for another reason, do not hesitate to contact us via the INTEGRAL help-desk at inthelp@sciops.esa.int for troubleshooting or to check for the possibility to obtain a custom made binary package.

We strongly recommend that you first install and test the binary package for your operating system(s).

5. Supported Platforms

The OSA software was checked by ISDC to correctly compile, install and run on the following platforms:

- Linux
 - CentOS 5.11 64 bit gcc 4.4.7
 - CentOS 6.9 64 bit gcc 4.4.7
 - CentOS 7.5 64 bit gcc 4.4.7
 - Ubuntu 16.04 64 bit gcc 4.4.7

The binary OSA software packages that we provide at the ISDC are generated for these platforms. In general, the OSA software will also compile and run on a variety of other Linux platforms. As we at the ISDC do not necessarily have access to those platforms, we unfortunately can only offer limited support for those platforms.

6. Third Party Software

Before compiling and installing the OSA SW from the source code you have to make sure that the following packages are installed:

1. GNU make version 3.79.1 (or higher)

⚠ You really need this version of GNU make for all platforms. Most Linux distributions will already include the correct make. Earlier versions of GNU make are known not to work.

2. ROOT version 5.34.34

available via the ISDC website at URL <http://www.isdc.unige.ch/integral/analysis#Software> in the section on ‘Download OSA 10.2 for Developers’.

i Since OSA software version 3.0 you can choose between an installation with and without ROOT. If you choose to install without ROOT, you will NOT benefit from all OSA functionalities, i.e. GUI support is not available. To learn more about how to install OSA without ROOT please see section ‘Setting up the Environment’.

⚠ For Mac OS X users it is required to install ROOT from the source code.

For other operating systems it is recommended to install ROOT from the source code. If you want to download ROOT as a binary package, you must make sure that the compiler used to compile ROOT is identical to the one you are using for the installation of the OSA SW. Otherwise, please download the ROOT source code package and compile and install it yourself. Using different compilers may result in serious problems with your X-system.

7. Hardware Requirements

- **Disk Space**

To build and install the OSA software from source code a total of some 1.7 GByte of disk space is needed – depending on the operating system and compiler used.

i After the successful installation from the source code you may reduce the disk space needed by OSA software by executing ‘make distclean’ in the same directory where you executed ‘make install’. This will reduce the amount of disk space needed to some 890 MB. You may additionally remove the source code directories (support-sw, analysis-sw, contrib-sw). This will reduce the disk space needed to some 790 MB.

- **Memory/CPU**
- i* You will need approximately 10 GB of disk space to hold the data from one day of INTEGRAL observations.

8. Installing the Software

The INTEGRAL OSA software source code package (osa10.2-source.tar.gz) is available via the following URL <http://www.isdc.unige.ch/integral/osa/current/developers>

Once you have downloaded the source code package

- cd to a directory to which you have write access and
- unpack the downloaded tar-file using e.g. the following command:
`gzip -dc osa11.0-source.tar.gz | tar xvf -`

Resulting from the above command(s) you will find the directory osa11 containing all the OSA source code. For a typical installation, this directory is referred to as ISDC_ENV in the INTEGRAL OSA user documentation.

9. Setting up the Environment

This section describes the setting of the installation relevant environment variables.

It is not intended to provide a description of the environment that is needed to run the software. The installation environment will not be sufficient to run the software. Please see the instrument specific cookbooks for a detailed description on how to set-up the environment to actually run the OSA software.

10. Compiler

Make sure that your compiler's bin and lib directories are contained in environment variables PATH and LD_LIBRARY_PATH respectively. A way to check this, is to use the 'which' command followed by the name of your compiler. If the compiler is not found, type the following commands:

```
setenv PATH "__path_to_your_compiler__/bin:$PATH"  
setenv LD_LIBRARY_PATH "__path_to_your_compiler__/lib:$LD_LIBRARY_PATH"
```

i Users of Mac OS X need to use environment variable DYLD_LIBRARY_PATH instead of LD_LIBRARY_PATH

i If one or both of the above commands fail with a message like: 'LD_LIBRARY_PATH: Undefined variable', omit the ':\$PATH' or ':\$LD_LIBRARY_PATH' part in the commands respectively.

i If the setenv commands fail with a message like: 'setenv: command not found' or 'setenv: not found' please read the appendix on 'Troubleshooting'.

In addition, you have to make sure that all compiler relevant environment variables are correctly set up. Depending on the compiler and OS used, those variables are:

Linux (including Ubuntu) using GNU C/C++ and Fortran Compilers

```
setenv CC "gcc -Df2cFortran"
setenv CXX "g++ -Df2cFortran"
setenv F90 gfortran
```

- i* The ISDC default setting on Linux is to switch the code optimiser on (option -O2).
- ⚠** You must have a FORTRAN90 compiler installed on your system. Otherwise the build will fail.
- i* If the `setenv` commands fail with a message like: 'setenv: command not found' or 'setenv: not found' please read the appendix on 'Troubleshooting'.

11. ROOT

You can choose between an installation with and without ROOT. If you choose to install without ROOT, you will NOT benefit from all OSA functionalities, i.e. GUI support is not available. Once you have installed OSA software without ROOT and would like to use the GUIs, you will have to re-install OSA software from the source code.

If you choose NOT to install with ROOT, you have to make sure that environment variable ROOTSYS is NOT set. Otherwise you will run into problems during the OSA software compilation.

If you choose to install with ROOT, make sure that environment variable ROOTSYS is set to where your ROOT installation is located. The ROOTSYS environment variable should typically point to a directory with at least the following sub-directories: `bin`, `etc`, `icons`, `include` and `lib`. Make sure that your ROOT's `bin` and `lib` directories are contained in environment variables `PATH` and `LD_LIBRARY_PATH` respectively. A way to check that is to actually execute the '`root`' command and verify the version of root. To exit root, please type '`.q`'. If root is not found, type the following commands:

```
setenv PATH "$ROOTSYS/bin:$PATH"
setenv LD_LIBRARY_PATH "$ROOTSYS/lib:$LD_LIBRARY_PATH"
```

- i* If one or both of the above commands fail with a message like: 'LD_LIBRARY_PATH: Undefined variable', omit the '`:$PATH`' or '`::$LD_LIBRARY_PATH`' part in the commands respectively.
- i* If the `setenv` commands fail with a message like: 'setenv: command not found' or 'setenv: not found' please read the appendix on 'Troubleshooting'.

12. ISDC_ENV

Decide where the ISDC software package should be installed. This is controlled via the environment variable `ISDC_ENV`. You may set the value of `ISDC_ENV` to any location where you have write permission. It is however recommended to set `ISDC_ENV` to the directory where the source code of the OSA software package is located.

i When installing on a multi-platform system, you might find it useful to have a common source code tree for the various binary installations you maintain. This can be achieved by setting the ISDC_ENV environment variable to e.g. __osa_src_rep__/Linux_2.4_i686 for an installation under Linux.

⚠ Make sure that \$ISDC_ENV/bin is contained in the environment variable PATH – even if this directory does not yet exist at this stage of the installation. If \$ISDC_ENV/bin is not included, execute the following command: setenv PATH "\$ISDC_ENV/bin:\$PATH". If the setenv command fails, please read the appendix on ‘Troubleshooting’.

Configure, Build and Install

When the installation environment is successfully set up you may proceed to configure, build and install the software package. This is done by executing the following command sequence in the directory where you unpacked the OSA software source package:

13. configure

cd to the directory where you have unpacked the downloaded tar-file.

If you want to install OSA software without ROOT execute the command
support-sw/makefiles/ac_stuff/configure --without-cern-root

If you want to install OSA software with ROOT execute the command
support-sw/makefiles/ac_stuff/configure

This command will create all system dependent Makefiles that are needed to build the software.
Please type ‘support-sw/makefiles/ac_stuff/configure --help’ to learn about the configuration options.

⚠ The setting of all environment variables mentioned in section ‘Setting up the Environment’ has to be done prior to executing the configure command. If, after executing the configure command, you change your mind about the value of an environment variable, you must first execute ‘make distclean’ and then redo the configuration step.

⚠ When installing on a multi-platform system and you decided to share the source code tree, you must execute ‘make distclean’ and then redo the configuration step for each platform.

⚠ Make sure that the environment variable MAKEFLAGS is not set.

If the configure step terminated with an error and you cannot figure out the problem yourself, please consult the ‘Frequently Asked Questions’ at <http://www.isdc.unige.ch/integral/analysis#FAQ> to check for a solution to your problem or send an E-mail to the INTEGRAL help-desk at inthelp@sciops.esa.int. For an efficient trouble-shooting, please see appendix ‘Troubleshooting’ for information you should provide to help us solving the problem you have.

14. make install

Type ‘make install’

⚠ You really have to type ‘make install’. A simple make will fail.

This will compile all relevant source files, create the library files, the F90 modules and the C/C++ and F90 executables. The installation terminates by moving the library, the C/C++ header files, the F90 modules, the parameter files and the user manuals to the correct location.

If everything went ok, you should see the following message:

```
***** everything compiled and installed successfully *****  
executables are installed in : $ISDC_ENV/bin  
parameter files are installed in : $ISDC_ENV/pfiles  
F90 modules are installed in : $ISDC_ENV/f90mod  
documentation, help, .txt files are installed in : $ISDC_ENV/help  
C/C++ include .h files are installed in : $ISDC_ENV/include  
.a, .so library files are installed in : $ISDC_ENV/lib  
script files are installed in : $ISDC_ENV/share/default_scripts  
other shared files are installed in : $ISDC_ENV/share  
FITS file template files are installed in : $ISDC_ENV/templates
```

If ‘make install’ terminated with an error and you cannot figure out the problem yourself, please consult the ‘Frequently Asked Questions’ at <http://www.isdc.unige.ch/integral/analysis#FAQ> to check for a solution to your problem or send an E-mail to the INTEGRAL help-desk at inthelp@sciops.esa.int. For an efficient trouble-shooting, please see appendix ‘Troubleshooting’ for information you should provide to help us solving the problem you have.

- i Once the installation is successfully terminated you may recover some disk space by removing temporary compilation products. This is done by executing ‘make distclean’. You may even recover more disk space and remove the entire source code tree by executing ‘rm -rf support-sw analysis-sw other-sw’ in OSA source code directory.

Now OSA software is fully installed and is ready to be used. Please refer to the OSA documentation available at <http://www.isdc.unige.ch/integral/analysis> for details on how to run the OSA software. The ‘Introduction to INTEGRAL Data Analysis’ and the cookbook section of the instrument specific user manual will be of particular use for you.

15. Appendix C: Document Change History

11.0	Updated for OSA 11 with a section on docker usage. Removed references to Mac OS.
10.2	Updated for OSA 10.2 (use of same release number). Addition of an Ubuntu binary and update of Mac OS X versions: for Mavericks (10.9), Yosemite (10.10), and El Capitan (10.11), instead of Lion (10.7) and Mountain Lion (10.8).
2.3	Updated for OSA 10.1. Mac OS X Mountain Lion (10.8) instead of Snow Leopard (10.6) and Mac OS X distribution as `tar.gz` file instead of `dmg.zip`.
2.2	Updated for OSA 10.0. The testdata are one package for all instruments
2.1.2	Updated for OSA 9.0. Removed references to Solaris as this is not supported anymore
2.1.1	Changed version of GNRL_REFR_CAT from 28 to 30
2.1	Updated for OSA 8.0, especially the IC download description
2.0.6	Updated description for CAT and IC installation, especially the symbolic linking part for the IC data.
2.0.5	Highlighted more the OMC catalogue, catalogue version number fixed and removed a couple of typos.
2.0.4	Adapted to OSA 7.0
2.0.3	changed e-mail address of INTEGRAL help desk
2.0.2	Adapted to OSA 6.0
2.0.1	Adapted to OSA 5.1
2.0	Complete re-write