

Preparation for the ECP-AHM tutorial "High Performance I/O Frameworks 101"

Accessing required software:

Either choose 1,2, or 3.

1. Virtual Machine (preferred for VirtualBox and Windows users)
2. Conda (preferred for Linux/macOS 64-bit existing conda users)
3. Build from source (preferred for Linux/macOS 64-bit)

Additional software: VisIt and/or Paraview Installations (preferred for Linux/macOS 64-bit)

Option 1: Virtual Machine (preferred for VirtualBox and Windows users)

Requirements

1. **VirtualBox (highly recommended to install in advance)**
<https://www.virtualbox.org/wiki/Downloads>
2. **30 Gb of disk space**

Steps

1. **Obtain the Virtual Machine (VM) file for this tutorial:** We provide a Virtual Machine (VM) that is about 7 GB in its archival file format. There are two options to obtain the VM:
 - a. USB sticks at the tutorial, so participants can copy it to their laptops (preferred for slow WiFi)
 - b. From Dropbox (requires fast WiFi access)
<https://www.dropbox.com/s/fyh2qn9hp12qj9u/ADIOS%202.5.0-ECP.ova?dl=0>.
2. **Install the VM on VirtualBox**
Run VirtualBox and create your VM by "File/Import Appliance...". The imported VM will take up between 20-25 GB of space. You can change the memory size and

the number of processors for your liking in the System section of the VM settings.
Run the VM. It is a Lubuntu 16.04 LTS Linux machine.
The username and password are both "**adios**".

Notes

1.

For Linux laptop users. If you cannot read the USB stick, check support for exFAT filesystems. From this page:

<http://unixnme.blogspot.com/2016/04/how-to-mount-exfat-partition-in-ubuntu.html>

How to Mount EXFAT partition on Ubuntu 16.04 LTS

To mount exfat partition on Ubuntu, simply install the necessary packages:

```
$ sudo apt-get install exfat-fuse exfat-utils
```

If you need to mount it from the command line, you could do

```
$ sudo mkdir /media/exfat
```

```
$ sudo mount -t exfat /dev/sdxx /media/exfat
```

where /dev/sdxx could be /dev/sda1 or /dev/sda2, or so on. If you are not sure whether the exfat partition you are looking for is /dev/sda1 or /dev/sda2, then you could also run

```
$ sudo lsblk -o NAME,FSTYPE,SIZE,MOUNTPOINT,LABEL
```

Option 2: Conda for Linux/macOS 64-bit systems

Requirements

1. **Anaconda:** latest anaconda with python3 support:
<https://www.anaconda.com/distribution/>

Steps

1. **Set the adios2-examples environment:** The adios2-examples package from anaconda cloud “williamfgc” channel was set for this tutorial
<https://anaconda.org/williamfgc/adios2-examples>

```
                $ conda activate
      (base) $ conda create --name adios2-examples
(adios2-examples) $ conda activate adios2-examples
(adios2-examples) $ conda install adios2-examples -c conda-forge -c
williamfgc
```

2. **Get config files from installation:** Go to a scratch area (e.g. `cd ~/tmp`)

We assume anaconda3 used the default installation directory

```
(adios2-examples) $ cp -r ~/anaconda3/envs/adios2-
examples/share/adios2-examples/gray-scott .
(adios2-examples) $ cd gray-scott
```

3. Follow README.md instructions in the gray-scott directory

Notes:

- The adios2-examples and adios2 binaries should be already accessible in your \$PATH. Do ``which adios2-gray-scott`` to find the bin location for your environment.
- If you have an existing adios2 installation outside of conda, please uninstall it or remove locations from your \$PATH and \$LD_LIBRARY_PATH (macOS: \$DYLD_LIBRARY_PATH).

Option 3: Building ADIOS from source

Required dependencies

Python3, Numpy, Matplotlib, CMake (3.12 or newer)

<https://cmake.org/download/>

Suggested dependencies

These are not required but some parts of the tutorial are using them

- SZ and/or ZFP compression libraries
<https://github.com/disheng222/SZ/releases>
<https://github.com/LLNL/zfp/releases>
- Parallel HDF5 library
<https://www.hdfgroup.org/downloads/hdf5/source-code/>

ADIOS2 source and tutorial

Source code: <https://github.com/ornladios/ADIOS2/releases/tag/v2.5.0>

(Direct link: <https://github.com/ornladios/ADIOS2/archive/v2.5.0.tar.gz>)

Tutorial examples: <https://github.com/ornladios/ADIOS2-Examples/releases/tag/v2.5.0>

(Direct link: <https://github.com/ornladios/ADIOS2-Examples/archive/v2.5.0.tar.gz>)

Building ADIOS2

Note that installation directories of ZFP, SZ, Parallel HDF5 and add them to the -DCMAKE_PREFIX_PATH option when configuring ADIOS2. If you have built those packages with shared libraries, don't forget to add them to LD_LIBRARY_PATH for your environment.

```
$ tar zxf ADIOS2-2.5.0.tar.gz
$ cd ADIOS2-2.5.0
$ mkdir build
$ cd build
$ cmake -DCMAKE_INSTALL_PREFIX=/opt/adios2 \
    -DCMAKE_BUILD_TYPE=Release \
    -DCMAKE_PREFIX_PATH="/opt/zfp;/opt/SZ;/opt/hdf5-parallel" \
    -DADIOS2_USE_MGARD=OFF \
    -DADIOS2_USE_MPI=ON \
    -DADIOS2_USE_Python=ON \
    -DADIOS2_USE_HDF5=ON \
    -DADIOS2_USE_Profiling=ON \
    -DADIOS2_BUILD_TESTING=OFF \
    ..
...
ADIOS2 build configuration:
ADIOS Version: 2.5.0
C++ Compiler : GNU 5.4.0
    /usr/bin/c++

Fortran Compiler : GNU 5.4.0
```

```
/usr/bin/f95
```

```
Installation prefix: /opt/adios2  
  bin: bin  
  lib: lib  
include: include  
  cmake: lib/cmake/adios2  
  python: lib/python3.5/site-packages
```

Features:

```
Library Type: shared  
Build Type: Release  
Testing: OFF  
Examples: ON
```

Build Options:

```
Blosc      : OFF  
BZip2      : ON  
ZFP        : ON  
SZ         : ON  
MGARD      : OFF  
PNG        : ON  
MPI        : ON  
DataMan    : ON  
Table      : ON  
SSC        : ON  
SST        : ON  
DataSpaces: OFF  
ZeroMQ     : ON  
HDF5       : ON  
Python     : ON  
Fortran    : ON  
SysVShMem  : ON  
Profiling  : ON  
Endian_Reverse: OFF
```

```
RDMA Transport for Staging: Unconfigured
```

```
-- Configuring done  
-- Generating done  
-- Build files have been written to: /home/adios/Software/ADIOS2/build  
$ make -j 4  
$ make install
```

Building ADIOS2-Examples

```
$ tar xzf ADIOS2-Examples-2.5.0.tar.gz  
$ cd ADIOS2-Examples-2.5.0  
$ mkdir build-cmake  
$ cd build-cmake
```

Assuming the installation directory for the examples is /home/adios/Tutorial and the installation directory of ADIOS2 is /opt/adios2

```
$ cmake -DCMAKE_INSTALL_PREFIX=/home/adios/Tutorial \
        -DADIOS2_DIR=/opt/adios2 ..

-- The C compiler identification is GNU 5.4.0
-- The CXX compiler identification is GNU 5.4.0
...
-- Found MPI_C: /usr/lib/openmpi/lib/libmpi.so (found version "3.0")
...
-- Found ADIOS2: /opt/adios2/lib/cmake/adios2/adios2-config.cmake
(found version "2.5.0")
-- Configuring done
-- Generating done
-- Build files have been written to: /home/adios/ADIOS2-
Examples/build-cmake

$ make -j 4
$ make install
```

Add the bin directory in the installation directory to your PATH, so that the example executables can be run anywhere,

```
$ export PATH=$PATH:/home/adios/Tutorial/bin
```

Then run the tutorial example in

```
$ cd /home/adios/Tutorial/share/adios2-examples/gray-scott
```

Additional Software for Option 2 and 3. VisIt and/or Paraview installations

You can download recent versions of VisIt and Paraview for macOS, Linux. Make sure you download a version that is built with MPI. The VM in Option 1 already has them installed.

- VisIt 3.1.0
 - <https://wci.llnl.gov/simulation/computer-codes/visit/downloads>
- ParaView 5.8.0-RC1
 - <https://www.paraview.org/download/>

Notes:

- ParaView only has support for file based (bp3 or bp4) engine outputs.
- Windows users: only ParaView has adios2 support.
- Windows Linux subsystem users: make sure location is accessible via