SIMDIS Guide

What is SIMDIS?

SIMDIS is a set of Government-off-the-Shelf software tools that provide 2D and 3D interactive graphical and video display of live and post processed simulation, test, and operational data. SIMDIS allows an integrated view of both time-space position information and telemetry data. SIMDIS provides an intuitive view of complex system interactions before, during and after an event.

The SIMDIS toolset includes custom tools to analyze and display data for:

- Equipment Models
- Spatial Grids
- Ranges
- Angles
- Antenna Patterns
- RADAR Cross-Section
- Line-of-Sight
- Lasers
- RF Propagation
- Lines of Bearing

How to install SIMDIS

SIMDIS supports Windows 7, 10 and RHEL 6 and 7 (Linux). SIMDIS has also been tested on RHEL 8, Ubuntu 16.04 and 18.04.

For OSX users, you will need to run SIMDIS in a VM. WARNING: There may be performance issues running a VM on OSX.

There are several ways to install SIMDIS. For an easy installation, run the installer ***SIMDIS_Installer***. For Linux users, you will need to change permission to allow the installer to run. For custom installations, please see the ***SIMDIS Download Document*** or the SIMDIS user manual.

SIMDIS - Extra contains extra features that can be loaded into the start-up sequence of SIMDIS. To add the extra features, copy the contents of this folder into the main SIMDIS directory, usually located at %SIMDIS_DIR%. WARNING: Do not replace the current SIMDIS files.

PluginAPI are external plugins that can be used with SIMDIS. The folder does not need to be loaded in a particular location.

On the Tuesday, 16 November, a link will be sent out on Slack for you to download SIMDIS. The link will be valid for 72 hours. If you have any issues with the link, please contact the Track 3 technical team on Slack.

Creating Python/C++ Applications

C++ applications can be build using a Plug-in API. The Plug-in API is a C++ software library. The API can be used to integrate a user's software with the SIMDIS 3D Visualization, Analysis and Display package. SIMDIS loads the plug-in clients in the form of shared objects under UNIX or dynamically loaded libraries under Windows systems.

The Plug-in API is supplied as source code only. To compile C++ packages, you will need to install:

- CMake 3.0 or later
- Current C++ compiler. Currently in use is g++ 8.3 (Linux) and VC++ 14.0/14.1/14.2 (Visual Studio 2015/2017/2019 respectively on Windows)

Python applications use the SIMDIS embedded Python interpreter to allow written Python code to interact with SIMDIS Plug-in API. The Python API is enabled by wrapping the current Plug-in API with SWIG. You can install SWIG from http://www.swig.org/download.html.

Custom applications can be developed in the HOST environment separate from the SIMDIS interpreter. In order to connect the SIMDIS environment with your host machine there are several steps:

1. Make sure you have Python 3.9 installed. Python 3.9 can be downloaded from https://www.python.org/downloads/. Make sure to note where Python is installed.
    a. Add your download location to the beginning of your system PATH environment variable
2. Download pip from https://bootstrap.pypa.io/get-pip.py. Use your new Python version (python or python3 command) to set up pip.
3. Add the folder where pip downloads packages to your system PYTHONPATH environment variable.
    a. To find where the folder, run *pip show pip* in the command line. The folder will be located under the 'Location' value.
4. Add '%SIMDIS_DIR%/bin/pythonScripts' to your system PYTHONPATH environment variable.
    a. pythonScripts contains python packages designed to interact with the SIMDIS application
    b. SIMDIS_DIR is set up during the installation of SIMDIS.

For more information about the Plug-in API, please read the official documentation included in the downloaded folder at ***PluginAPI/PluginClient/doc/PluginAPI.doc***. More information about the Python API can be found in the SIMDIS user manual.

Loading Python/C++ Application into SIMDIS

Applications loaded to SIMDIS are managed by the Plug-in Manager. The easiest way to load your application is the drag and drop the file into SIMDIS. To start your application, you will need to select your application from the Plug-ins dropdown.

Another alternative to load applications is the use the Plug-in Manager. The Plug-in Manger instructions are in the document ***PluginAPI/PluginClient/doc/PluginAPI.doc.***

The SIMDIS download zip package contains example applications to run with SIMDIS. To run the examples:

1. The example code requires pandas. Run pip (or pip3) to install pandas. Make sure to add --user to install into your home directory (instead of system directory requiring admin privileges)
2. 'drugboat.py' and 'forcepackage.py' are the two example applications that are provided. Each file contain two hardcoded file paths with token '<path-to-example-folder>' that must be changed. Open both Python files with a text editor and change the path variables to your download location (lines 14/15 and 16/17).
    a. Example: "C:\Users\<NAME>\<where did you download?>\simdis-<host>\Example-Code\"
3. Open the SIMDIS application.
4. Drag and drop the application file (eg. drugboat.py) into SIMDIS.
5. Open the Plug-ins tab and select your Plug-in (named DrugBoatScenario and ForcePackageScenario respectively).
6. SIMDIS will change the view point and run the code

```
ERROR   Notifications   < PY EXC > Runtime failure on DrugBoat: Unable to run plug-in.
ERROR   Plug-ins        ERROR: Plug-in returned bad value on StartUp(), stopping C:\Users\18312\Downloads\simdis-windows\simdis-windows\Example-Code\DrugBoat.py.
```

NOTE: If you see any errors similar to the example below, please double check the above list. If you still have any issues, please contact Tech Support.