Powered by tecplot

Quick search
Go

Table of Contents

Installation
- Python and Tecplot
  - Package Install
  - Tecplot 360 EDU or Tecplot 360 GUI
  - Tecplot Server
- Windows
  - Installation
  - Environment Setup
  - Updating
- Linux
  - Installation
  - Environment Setup
  - Updating
- Mac OS X
  - Installation
  - Environment Setup (Batch Only)
  - Updating
- Troubleshooting
- Environment Variables

Examples
Release Notes
Glossary
Functions
Classes

This Page

Show Source

Installation

Python and Tecplot

Package Install

It is recommended to use pip to install PyTecplot, along with all of the dependencies, from Python's Package Index. This will install the latest version of PyTecplot and its dependencies. Tecplot 360 must be installed and in path of the installed machine that came with recent version of Python. Please refer to pip's documentation for information if you have issues using pip.

Depending on the options you set when installing Python, you may already have the folder containing pip in your path environment variable. If this is the case, then you may use pip directly instead of prepending it with python -m. The rest of the following commands will assume this.

```
python -m pip install --user pytecplot
```

Installing from Local Source

For those with a restricted internet connection, it is necessary to "manually" install all the required dependencies as listed in the section "Software Requirements" above. This sometimes involves downloading these packages from https://pypi.org/, transferring them to the target system and running pip install --user <package_name>. Alternatively, you may wish to install PyTecplot into your system's site-packages or prefix. A complier may be required if there are no pre-compiled binaries for your specific operating system and Python version.

PyTecplot ships with Tecplot 360 and can be found under the pytecplot directory. You may run pip from within this directory to install pytecplot as follows. Note the use of "." indicates the current working directory.

```
cd /path/to/tecplot/360/pytecplot
pip install .
```

Installing Without Root Access

If you get a "permission denied" error, this likely means you are attempting to install PyTecplot into a system-controlled Python package directory. If this is what you want to do, then you must prepend the above pip command with sudo. Alternatively, you may wish to install PyTecplot into your user-space or home directory. This can be done by add the option --user to the install step (see the output of the command pip help install):

```
pip install --user pytecplot
```

Optional Dependencies

All required dependencies will be installed along with PyTecplot. There are optional dependencies such as Numpy and Python which you may want to install as well. These can be installed by appending [optional] to the installation command:

```
pip install pytecplot[numpy]
```

Environment Setup

PyTecplot scripts can be run in two distinct modes: "batch" in which PyTecplot manages it's own interaction with Tecplot 360, or "connected" where the PyTecplot script communicates with a running instance of Tecplot 360 through the Tecplot Sockets Framework. For more details, see the Tecplot Sockets Framework (tecplot.sockets.connect). For more details, no further environment setup is required. Currently, when running in "batch" mode, we need to use environment variables to point to the installation of Tecplot 360.

Depending on the options you selected when installing Tecplot 360, you may need to setup your environment to point to the Tecplot 360 installation you want to use. By default, the Tecplot 360 directory is not added to the system's path. You will have to add it and make sure it is before any other Tecplot 360 installation. With a standard installation of Tecplot 360, the path is usually something like the following. Again, "2009R2" should be replaced with the installed version of Tecplot 360.

```
"C:/Program Files/Tecplot/Tecplot 360 EX 2009R2/bin"
```

To view the current path, run the following command in the command console:

```
echo %PATH%
```

To edit it globally for all consoles you will have to navigate to "Control Panel" -> "System" -> "Advanced System Settings" -> "Environment Variables". From there, you should find the path environment variable, edit it, and click "OK" to record it. Required. After changing the state, be sure to close and re-open your console window.

Updating

To update PyTecplot after you have already installed it once, you can run the same installation command with the option --upgrade. For example:

```
python -m pip install --upgrade pytecplot
```

When installing a new version of Tecplot 360, you must ensure that the tecio environment variable gets updated accordingly.

Linux

Note: We recommend using the operating system's package manager to install and update Python along with pip. Once this is done, you can use sudo pip to manage the installation of system-wide python modules.

Installation

Once Python is installed along with the pip module, you may install PyTecplot from Python's official PyPI repository by running the following command with root privileges (sudo) if needed:

```
pip install pytecplot
```

Installing from Local Source

For those with a restricted internet connection, it is necessary to "manually" install all the required dependencies as listed in the section "Software Requirements" above. This sometimes involves downloading these packages from https://pypi.org/, transferring them to the target system and running pip install <package_name>. Alternatively, you may wish to install PyTecplot into your system's site-packages or prefix. A complier may be required if there are no pre-compiled binaries for your specific operating system and Python version.

PyTecplot ships with Tecplot 360 and can be found under the pytecplot directory. You may run pip from within this directory to install pytecplot as follows. Note the use of "." indicates the current working directory.

```
cd /path/to/tecplot/360/pytecplot
pip install .
```

Installing Without Root Access

If you get a "permission denied" error, this likely means you are attempting to install PyTecplot into a system-controlled Python package directory. If this is what you want to do, then you must prepend the above pip command with sudo. Alternatively, you may wish to install PyTecplot into your user-space or home directory. This can be done by add the option --user to the install step (see the output of the command pip help install):

```
pip install --user pytecplot
```

Optional Dependencies

All required dependencies will be installed along with PyTecplot. There are optional dependencies such as Numpy and Python which you may want to install as well. These can be installed by appending [optional] to the installation command:

```
pip install pytecplot[numpy]
```

Environment Setup (Batch Only)

PyTecplot scripts can be run in two distinct modes: "batch" in which PyTecplot manages it's own interaction with Tecplot 360, or "connected" where the PyTecplot script communicates with a running instance of Tecplot 360 through the Tecplot Sockets Framework. For more details, see the Tecplot Sockets Framework (tecplot.sockets.connect). For more details, no further environment setup is required. Currently, when running in "batch" mode, we need to use environment variables to point to the installation of Tecplot 360.

Depending on the options you selected when installing Tecplot 360, you may need to setup your environment to point to the Tecplot 360 installation you want to use. By default, the Tecplot 360 directory is not added to the system's path. You will have to add it and make sure it is before any other Tecplot 360 installation. With a standard installation of Tecplot 360, the path is usually something like the following. Again, "2009R2" should be replaced with the installed version of Tecplot 360.

```
"C:/Program Files/Tecplot/Tecplot 360 EX 2009R2/bin"
```

To view the current path, run the following command in the command console:

```
echo %PATH%
```

To edit it globally for all consoles you will have to navigate to "Control Panel" -> "System" -> "Advanced System Settings" -> "Environment Variables". From there, you should find the path environment variable, edit it, and click "OK" to record it. Required. After changing the state, be sure to close and re-open your console window.

Updating

To update PyTecplot after you have already installed it once, you can run the same installation command with the option --upgrade. For example:

```
python -m pip install --upgrade pytecplot
```

When installing a new version of Tecplot 360, you must ensure that the tecio environment variable gets updated accordingly.

Mac OS X

Note: We highly recommend using a package management tool such as Macports, Brew or Fink to install and update Python along with pip. Once this is done, you can use sudo pip to manage the installation of system-wide python modules.

Installation

Once Python is installed along with the pip module, you may install PyTecplot from Python's official PyPI repository by running the following command with root privileges (sudo) if needed:

```
pip install pytecplot
```

Installing from Local Source

For those with a restricted internet connection, it is necessary to "manually" install all the required dependencies as listed in the section "Software Requirements" above. This sometimes involves downloading these packages from https://pypi.org/, transferring them to the target system and running pip install <package_name>. Alternatively, you may wish to install PyTecplot into your system's site-packages or prefix. A complier may be required if there are no pre-compiled binaries for your specific operating system and Python version.

PyTecplot ships with Tecplot 360 and can be found under the pytecplot directory. You may run pip from within this directory to install pytecplot as follows. Note the use of "." indicates the current working directory.

```
cd /path/to/tecplot/360/pytecplot
pip install .
```

Installing Without Root Access

If you get a "permission denied" error, this likely means you are attempting to install PyTecplot into a system-controlled Python package directory. If this is what you want to do, then you must prepend the above pip command with sudo. Alternatively, you may wish to install PyTecplot into your user-space or home directory. This can be done by add the option --user to the install step (see the output of the command pip help install):

```
pip install --user pytecplot
```

Optional Dependencies

All required dependencies will be installed along with PyTecplot. There are optional dependencies such as Numpy and Python which you may want to install as well. These can be installed by appending [optional] to the installation command:

```
pip install pytecplot[numpy]
```

Environment Setup (Batch Only)

PyTecplot scripts can be run in two distinct modes: "batch" in which PyTecplot manages it's own interaction with Tecplot 360, or "connected" where the PyTecplot script communicates with a running instance of Tecplot 360 through the Tecplot Sockets Framework. For more details, see the Tecplot Sockets Framework (tecplot.sockets.connect). For more details, no further environment setup is required. Currently, when running in "batch" mode, we need to use environment variables to point to the installation of Tecplot 360.

Depending on the options you selected when installing Tecplot 360, you may need to setup your environment to point to the Tecplot 360 installation you want to use. By default, the Tecplot 360 directory is not added to the system's path. You will have to add it and make sure it is before any other Tecplot 360 installation. With a standard installation of Tecplot 360, the path is usually something like the following. Again, "2009R2" should be replaced with the installed version of Tecplot 360.

```
"/Applications/Tecplot 360 EX 2009R2/bin"
```

To view the current path, run the following command in the command console:

```
echo $PATH
```

To edit it globally for all consoles you will have to navigate to "/etc/paths". Required. After changing the state, be sure to close and re-open your console window.

Updating

To update PyTecplot after you have already installed it once, you can run the same installation command with the option --upgrade. For example:

```
python -m pip install --upgrade pytecplot
```

When installing a new version of Tecplot 360, you must ensure that the tecio environment variable gets updated accordingly.

Troubleshooting

Note: We highly recommend using a package management tool such as Macports, Brew or Fink to install and update Python along with pip. Once this is done, you can use sudo pip to manage the installation of system-wide python modules.

Installation

Once Python is installed along with the pip module, you may install PyTecplot from Python's official PyPI repository by running the following command with root privileges (sudo) if needed:

```
pip install pytecplot
```

Installing from Local Source

For those with a restricted internet connection, it is necessary to "manually" install all the required dependencies as listed in the section "Software Requirements" above. This sometimes involves downloading these packages from https://pypi.org/, transferring them to the target system and running pip install <package_name>. Alternatively, you may wish to install PyTecplot into your system's site-packages or prefix. A complier may be required if there are no pre-compiled binaries for your specific operating system and Python version.

PyTecplot ships with Tecplot 360 and can be found under the pytecplot directory. You may run pip from within this directory to install pytecplot as follows. Note the use of "." indicates the current working directory.

```
cd /path/to/tecplot/360/pytecplot
pip install .
```

Installing Without Root Access

If you get a "permission denied" error, this likely means you are attempting to install PyTecplot into a system-controlled Python package directory. If this is what you want to do, then you must prepend the above pip command with sudo. Alternatively, you may wish to install PyTecplot into your user-space or home directory. This can be done by add the option --user to the install step (see the output of the command pip help install):

```
pip install --user pytecplot
```

Optional Dependencies

All required dependencies will be installed along with PyTecplot. There are optional dependencies such as Numpy and Python which you may want to install as well. These can be installed by appending [optional] to the installation command:

```
pip install pytecplot[numpy]
```

Environment Setup (Batch Only)

PyTecplot scripts can be run in two distinct modes: "batch" in which PyTecplot manages it's own interaction with Tecplot 360, or "connected" where the PyTecplot script communicates with a running instance of Tecplot 360 through the Tecplot Sockets Framework. For more details, see the Tecplot Sockets Framework (tecplot.sockets.connect). For more details, no further environment setup is required. Currently, when running in "batch" mode, we need to use environment variables to point to the installation of Tecplot 360.

Depending on the options you selected when installing Tecplot 360, you may need to setup your environment to point to the Tecplot 360 installation you want to use. By default, the Tecplot 360 directory is not added to the system's path. You will have to add it and make sure it is before any other Tecplot 360 installation. With a standard installation of Tecplot 360, the path is usually something like the following. Again, "2009R2" should be replaced with the installed version of Tecplot 360.

```
/Applications/Tecplot 360 EX 2009R2/bin
```

To view the current path, run the following command in the command console:

```
echo $PATH
```

To edit it globally for all consoles you will have to navigate to "/etc/paths". Required. After changing the state, be sure to close and re-open your console window.

Updating

To update PyTecplot after you have already installed it once, you can run the same installation command with the option --upgrade. For example:

```
python -m pip install --upgrade pytecplot
```

When installing a new version of Tecplot 360, you must ensure that the DYLD_LIBRARY_PATH environment variable gets updated accordingly.

Troubleshooting

1. Verify that you have installed and can run Tecplot 360 version 2011 R2 or later.
2. Verify that you are running 64-bit Tecplot 360 version 11.1 or later.
3. Verify that the Python 3.x version you are using is compatible with the correct pytecplot executable. Installing into the Python 3.x site-packages typically requires elevated privileges. Therefore the pip (sudo) command may need a sudo or "Run as Administrator" type of environment. Alternatively, you may install PyTecplot into a local site-packages. Finally, ensure that your binaries directory is in your path.
4. Verify that the environment variables pointed to by tecio, DYLD_LIBRARY_PATH or DYLD_LIBRARY_PATH for Mac-Osxes, Linux and Windows respectively exists and contains the Tecplot 360 executable and library binaries.
5. Through the package is named "pytecplot" the actual python module that is imported is just "tecplot". If you should have "tecplot" installed and not "pytecplot" installed, verify that your version of pytecplot is the most recent version when you attempt to call pyte-pytecplot API, the most likely error will be something like "AttributeError: module 'tecplot' has no attribute 'connect'". Upgrade your subscription. Type "pip list" and pip to help -> "Tecplot 360 EX Licensing, to verify the license is configured properly.
6. Verify that the version of PyTecplot using pyp lists with a message like "No files were found to uninstall." It may be that tecplot is currently installed in a different Python installation. Verify that the tecplot binary matches one set to use 3.x.bin enter "tecplot -h" to find which version, platform, supportMachine.com.

This has been fixed in Tecplot 360 2020 R1 and updating Tecplot 360 should allow exporting of Python scripts from Tecplot 360 to work. Until then, the workaround is to manually edit the exported .lay package from setup.cfg and make sure you are using it instead of the python that was installed via Macports or Brew.

See the Getting Help section under Quick Start for troubleshooting after installation is successful.

Version Information

Version information of the tecplot Python module can be obtained as a string of the form "Major.Minor.Patch":

```
tecplot.__version__
```

or as a namedtuple with attributes: "major", "minor", "patch". "build" is that order:

```
tecplot.version_info
```

The underlying Tecplot 360 installation has its own version which can be obtained as a str:

```
tecplot.sdk_version
```

or as a namedtuple:

```
tecplot.sdk_version_info
```

PyTecplot adhere to the semantic versioning as outlined in Section 2 of this regard to backwards-compatibility. Specifically, the "Major" version will be incremented if any of the following are made: Function or Class Functions, properties etc.) prepended to the TP API or tecplot.exceptions and does not include methods in the code-base which do not show up in the HTML documentation, even when a deviating is present. Minor will be incremented backwards-compatibility of the public interfaces between of minor versions of the same major version with the exception of internal changes that fix incorrect behavior or bugs.

© Copyright 2021, Tecplot, Inc. Last updated on Jan 05, 2021.