

# **GT 4.2.0 pyGlobus: System Administrator's Guide**

DRAFT

## GT 4.2.0 pyGlobus: System Administrator's Guide

### Introduction

pyGlobus provides a Python API to the pre-OGSI GT API (GT2). This guide contains installation and configuration information for system administrators installing pyGlobus. It explains how to install, configure and test the installation.

### Important

This information is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the [Installing GT 4.2.0](#). Read through this guide before continuing!

DRAFT

# Table of Contents

- 1. Building and installing ..... 1
- 2. Configuring ..... 2
- 3. Deploying ..... 3
- 4. Testing ..... 4
- 5. Security considerations ..... 5
  - 1. Security Considerations for PyGlobus ..... 5
- 6. Troubleshooting ..... 6

DRAFT

# Chapter 1. Building and installing

pyGlobus requires python<sup>1</sup> version 2.2 or newer and the GT2 libraries built with a threaded flavor. The full source installation of GT 4.2.0 will build the prerequisite libraries with the correct flavor.

DRAFT

---

<sup>1</sup> <http://www.python.org>

## Chapter 2. Configuring

pyGlobus requires that the environment variables `GLOBUS_LOCATION` and `GPT_LOCATION` are set. These are set to the base of your globus installation and gpt installation. In GT 4.2.0 GPT (Grid Packaging Toolkit) is installed in the same place as globus, so you can set both of these environment variables to the same location.

DRAFT

## Chapter 3. Deploying

To install pyGlobus, set the GLOBUS\_LOCATION and GPT\_LOCATION environment variables. Use the python distutils setup script:

```
%python setup.py install
```

DRAFT

# Chapter 4. Testing

pyGlobus comes with unittests that are arranged by module. To run the unittests you first must be able to create a valid user proxy using `grid-proxy-init`. You must edit `config.txt` in the `pyGlobus/test` directory.

To run all of the tests:

```
%python test_pyGlobus.py
```

If you only wish to run tests from a single module, for example the security module:

```
%python test_security.py
```

DRAFT

---

# Chapter 5. Security considerations

## 1. Security Considerations for PyGlobus

pyGlobus has a security module which allows for proxy creation, signing, encryption, and the creation and inquiry of security contexts. Care must be taken when developing applications which use GSI to ensure that authentication information will not be compromised. When creating a security context, one must ensure that the context will have the properties that they desire. For example, should the context use confidentiality or integrity? These concerns are not specific to pyGlobus but rather to any application developer who is using low level security APIs.

DRAFT

# Chapter 6. Troubleshooting

See our [FAQ](#)<sup>1</sup> and [web page](#)<sup>2</sup> for any problems concerning pyGlobus.

You can also contact us on our mailing list [python-discuss@globus.org](mailto:python-discuss@globus.org)<sup>3</sup>.

DRAFT

---

<sup>1</sup> <http://www.dsd.lbl.gov/gtg/projects/pyGlobus/FAQ.html>

<sup>2</sup> <http://www.dsd.lbl.gov/gtg/projects/pyGlobus/index.html>

<sup>3</sup> <mailto:python-discuss@globus.org>