GIPSY 6.3 Release Notes

Jet Propulsion Laboratory
California Institute of Technology

March 18, 2014

Change Log:
March 18, 2014: Original version

Table of Contents

1 Introduction

Release 6.3 of GIPSY contains updates to release 6.2. Release 6.3 has been compiled with the gfortran compiler and is available as 64-bit and 32-bit executables. Release 6.3 is rootless, so it can be installed in any directory and uses environment variables to find the installed files.

GIPSY is naturally divided into two directory trees. The first directory tree is a read-only tree that contains the programs, scripts, libraries, and some tables that never change or would only change with a new release. The second directory tree contains tables that may change (Earth Orientation, PRN_GPS, atmos_flux, antenna calibrations). Some of these are files can be customized by the user, but most are files which occasionally need to be updated from a GIPSY ftp site.

Several different instances of rootless GIPSY can be installed on a single computer. In fact rootless GIPSY can be installed on a computer that also has an instance of the 4.X rooted instance of GIPSY. If users wish to run more than a single instance of GIPSY, we recommend that a single writable table directory be shared among the installed instances of GIPSY.

Under the read-only GIPSY installation directory, there are two startup scripts: rc_gipsy.csh for C-shell users, and rc_gipsy.sh for bash, Bourne, and Korn shell users. Users can set the environment by source-ing the appropriate startup script, either explicitly before each use of GIPSY, or by including this source command in a .cshrc file (or similar) so that it is executed automatically at computer log in. For example:

```
source rc_gipsy.csh
```

Once this is done, GIPSY modules can be executed. Users with locally developed scripts that have hard coded paths to GIPSY executables can either remove the paths or install the rootless version of GIPSY in the same path as previous versions of GIPSY.

Extensive testing of release 6.3 has been performed for typical applications, including precise point positioning of terrestrial receivers, and orbit/clock determination for low-Earth orbiters and Global Navigation Satellite System spacecraft (GPS, GLONASS, and DORIS). Feedback on possible defects or bugs with user applications, or general feedback, is appreciated. For release 6.3 feedback should be directed to the Release 6.3 topic on the GIPSY Forum. (https://gipsy-oasis.jpl.nasa.gov/index.php?page=forum)

2 New Features

2.1 New or Updated Models and Software

- tropnominal - added GPT2 troposphere model.
- qregres - added GPT2 troposphere model.
- qregres - added ocean load pole tide model.
- add_ocnld - changed default ocean load tide model to GOT4.8ac.
- add_cmotidenml.py - changed default tidal geocenter model to GOT4.8ac.

- postbreak - added –pb_min_elev and –pb_percent_of_largest options.
- gd2p.pl – added –add_ocnldpoltid to invoke ocean load pole tide model.
- gd2p.pl – added –pb_min_elev and –pb_percent_of_largest options.
- gd2p.pl – added option for orbit output interval.
- gd2p.pl – updates for single frequency processing and use of ionosphere models.
- gd2p.pl – replaced –gim option with –tec_mdl option.
- gd2p.pl – added –ion_integrate option to control computation of slant TEC.
- ninja - default removes data from GLONASS, Galileo, and SBAS satellites.
- pr2p - various modifications to pr2p to improve robustness. Allow usage of all data without decimation; allow user to provide apriori position and clock, use position and clock solution from previous epoch for initial guess for current epoch.
- ambigon4 - allow for L2/P2 single frequency data.
- rnx_time - accommodate other constellations and allow longer headers.
- stacov - output X, Y, and R_OFF if they appear to be estimated in smcov.nio.
- antex2xyz.py - allow DORIS antex files.
- rmsta, stacut, zeta_max – dimension increase.
- statransform – added extra precision to output.
- apply2tdp – modified output format to match output format of tdp files generated by gd2p.pl.
- transform and stacov2x – expanded space of position difference column in output by 1 to reduce cases of overflows.
- stacov2xfile.py – new program to compute X-files.
- oi - added model for GPS satellite transmitter antenna thrust as replacement for old UCL model. Invoked used ANT_THRUST_ON_OFF='ON', and ANT_THRUST_FILE = '$GOA_VAR/etc/antennaPower_IGS_2011_09_21.pm'.
- oi - added GSPM13 solar radiation pressure model for GPS satellites. Invoked using GPS_SRP_MODEL='GSPM_XYZ13', GPSR_SRP_MODEL='GSPM_XYZ13', GPSF_SRP_MODEL='GSPM_XYZ13'
- oi - updated solar radiation pressure model for GLONASS satellites.
- oi - increased maximum allow degree for ocean tide potential to 90.
- oi - added namelist parameter ALLOWABLE_TOL_CHANGE for users to manage tolerance changes in integrator (diva).
- brdc2gipsy.py - process RINEX 2 and 3 navigation files.
- match_qm - new version.
- pos2quat.pl – new program to use pos_goa or eci file to generate nominal quaternions for LEOs.
- res2pdop.pl – new program to compute PDOP for LEOs.

2.2 Bug Fixes

- gps_shadow – convert time tags from input shadow file or oi.nio from TDT to GPS time when creating list of points to pass to del_qm.
- tpmerge – added option to extended Earth orientation (EO) file.

- Station.py (library) – various bug fixes.
- Nav.py (library) – bug fixes for voting on navigation data.
- oi – bug fix for small error in station transition matrix when using DTM2000 drag model.
- oi – bug fix for getting yaw bias for satellite with bias type +1. (None since 1995-10-22).
- Earth orientation library - bug fix for small omission error in X and Y polar motion partials (S' matrix).
- GEOP file format – bug fixes in many functions for proper application. NOTE: We intend to replace tpnml files with GEOP files for earth orientation parameter time series within the next 2 years. See section 5 below.

3 Converting GIPSY Binary Files from 32-bit to 64-bit Platforms

GIPSY routinely uses two types of binary files. These files must be compatible with the native computer platform (32-bit or 64-bit).

1. Qmfile (output from ninja)
   The program qm_2_native converts any qmfile to the native architecture.
2. Navio (or nio) files
   These files are converted in a two-step process:
   a. Use GIPSY program nio2text to convert the nio file to a text file on the platform that it was generated.
   b. Use GIPSY program text2nio to convert the text file to a new nio file on the target platform.

4 Programs Not Available in Release 6.3

All program available in version 6.2 are also available in version 6.3.

5 Deprecated Software

Deprecated software means GOA programs that are likely to be deleted in the following release. They are in this release, but users are advised to prepare to do without them in the future.

- We will no longer generate and provide 32-bit builds of the GIPSY software starting with the 6.4 release.
- PvsCA – CA-P biases are now explicitly applied inside ninja software.

- amb_nml – Was used with ambigon2.
- Option to provide Extended Earth Orientation (UT1, and polar motion tidal) models (EpochEO, EOTIDE, UMD0, XMD0, and YMD0) through $earth_orientation namelist block, and option to provide timing and polar motion information using $tpnml option, will not be supported in future releases. This capability will be replaced by the GEOP file format (see $GOA/file_formats/GEOP). $GOA/file_formats/tp_array will not be supported in the future.
- Legacy Flinn products (in eci format) will not be supported in future releases. The option –orb_clk in gd2p.pl will no longer support types “flinn” and “ql”. Legacy products will be removed from the GIPSY ftp site (ftp://sideshow.jpl.nasa.gov/pub/gipsy_products). Reprocessed GPS orbit and clock products spanning August 16, 1992 onward are now available in the current flinnR and qlR (pos_goa) product formats at: ftp://sideshow.jpl.nasa.gov/pub/JPL_GPS_Products/Final.

6 Orbit/Clock products

6.1 Operational Products

GIPSY-OASIS is used to routinely generate precise estimates of the GPS satellite orbit positions and clock offsets. The latencies of these products vary from < 2 hours (near real time) to 14-days. The accuracy of the products improves with longer latencies.

Table 1. JPL’s Precise Orbit and Clock Estimates for GPS Constellation in GIPSY Format.

Product	Latency	3-D RMS Accuracy (cm)	Product Location (anonymous ftp)	High-Rate Clock Product Availability	Ambiguity Resolution Product (wlpb) Availability
Ultra-Rapid	< 2 hours	5	ftp://sideshow.jpl.nasa.gov/pub/JPL_GPS_Products/Ultra	None	All dates
Rapid	Next-day (16:00 UTC)	3.5	ftp://sideshow.jpl.nasa.gov/pub/JPL_GPS_Products/Rapid	July 1, 2011 to present	All dates
Final	< 14 days	2.5	ftp://sideshow.jpl.nasa.gov/pub/JPL_GPS_Products/Final	May 5, 2000 to present	August 16, 1992 to present

- All products are provided in formats native to GIPSY.
- Products include information to enable single receiver ambiguity resolution with the GIPSY-OASIS software.

- A reanalysis of GPS data from August 16, 1992 to generate GPS orbit and clock products was completed in late 2012 using IERS2010 standards and the IGS08 reference frame. All reprocessed products include ambiguity resolution product.
- A new reanalysis of historic data also spanning August 16, 1992 to present to generate GPS orbit and clock products is expected to be completed by mid-2014.
- Daily Rapid and Final products span a 30-hour window centered at noon of each day.
- Ultra-Rapid products span a 30-hour moving window, and are updated (overwritten) every hour. The file name prefix of the products (YYYY-MM-DD) is consistent with the last epoch of the 30-hour moving window. The exception is from 00:00 to 03:00 UTC of each day, when products from the current and previous day are both updated. As such, the last update for a particular day's Ultra-Rapid product covers a 30-hour period centered at noon of that day, consistent with the Rapid and Final products.
- Orbit estimates are delivered in Earth-fixed coordinates in GIPSY's pos_goa format (see $GOA/file_formats/pos_goa).
- Ultra-Rapid, Rapid and Final orbit products are all provided in a reference frame indicated by the .frame file. Final orbit products are also available in a so-called fiducial-free product (nf) along with a 7-parameter transformation (in a so-called x file) to the reference frame indicated by the .frame file.
- A new no-net-rotation (nnr) Final orbit and clock product will be introduced in mid-2014 and also provided in the next release of reanalyzed orbit and clock products. The nnr product will also include an associated 7-parameter transformation in a so-called x-file. This product is similar to the nf product, but is constrained to have no net rotation with respect to the reference frame indicated by the .frame file. As such, rotations in the associated x-file are negligible.
- All orbit estimates are intended to be with respect to the center of mass of the total Earth system. As such, when applying ocean load tide corrections, users are advised to use a model that is defined with respect to the center of mass of the total Earth system including the ocean tides. The option "-add_ocnld" in gd2p.pl allows the user to specify a specific coefficient file with ocean loading coefficients.
- Ultra-Rapid, Rapid, and Final clock products all include 5-minute clock solutions. Final products also include high-rate (hr) 30-second clock estimates from May 5, 2000 onward. Rapid products also include high-rate (hr) 30-second clock estimates starting in July 2011. The high-rate clock estimates are available about two hours after delivery of the standard Rapid products.
- All three products may be automatically fetched with gd2p.pl using the –orb_clk flag.
- Orbit/clock products for the GLONASS constellation are expected to be available in 2015. GIPSY 6.3 should be capable of performing precise positioning with those GLONASS products.

6.2 Compatibility Between GIPSY Versions and Orbit/Clock Products

Users are encouraged to transition to GIPSY6.1.x and higher when using orbit and clock products generated with the IERS2010 standards and IGS08 reference

frame for maximum consistency. This includes all products currently available at ftp://sideshow.jpl.nasa.gov/pub/JPL_GPS_Products.

7 GOA_VAR

We are releasing updates to the GOA_VAR area together with GIPSY 6.3. These encompass the following changes:

1. The files sta_pos, sta_svec, sta_id, and pcenter in $GOA_VAR/sta_info have been updated.
2. The ocean loading coefficients file $GOA_VAR/sta_info/ocnld_coeff_cm_fes04 has been updated to include additional sites and was generated with more accurate position information.
3. Added a new ocean loading coefficient file:
   $GOA_VAR/sta_info/ ocnld_coeff_cm_got48ac_wtpxo8ofunc
   This is the new default in add_ocnld.
4. Added a coefficient file for the ocean load pole tide model:
   $GOA_VAR/sta_info/ oceanpoletide_load_0.5deg.txt.
   This is the default when invoking the –add_ocnldpoltid option in gd2p.pl.
5. Added coefficient files for the GPT2 troposphere model:
   $GOA_VAR/etc/gpt2/
6. Added newly available GNSS antenna calibrations (ANTEX files) from the IGS, and corresponding native GIPSY format calibration files (*.xyz). The GNSS transmitter phase center variations in
   $GOA_VAR/etc/antenna_cals_xmit/igsXX_wwww.xyz,
   which are derived from the IGS Antex files, are extrapolated to a boresight angle of 30 degrees (elevation angle of 60 degrees). The original IGS transmitter calibrations in the ANTEX file are defined only for boresight angles of 0-14 degrees (i.e., elevation angles of 76-90 degrees) for igs08_1740.atx and earlier, and for boresight angles of 0-17 degrees (i.e., elevation angles of 73-90 degrees) for igs08_1748.atx and later. This extrapolation has no impact on terrestrial positioning but improves precise orbit determination of low-Earth orbiters. Presently, XX=”05” for files in the IGS05 reference frame, and XX=”08” for files in the IGS08 reference frame.
7. Updated International Reference Ionosphere 2012 (IRI2012) coefficient files in $GOA_VAR/etc/iri. This is the default location of the IRI model that is assumed by GIPSY. The set of IRI coefficients provided with this version is valid through October 31, 2013. Updates are made available through the update_gipsy_files.py script (see below).
8. Added IGS recommended antenna thrust model:
   $GOA_VAR/rtg/satelliteInfo/antennaPower_IGS_2011_09_21.pcm.

Users with current GOA_VAR installations who are running the update_gipsy_files.py

script will automatically get the updated antenna calibration files when the files are changed on the sideshow ftp site.

The update_gipsy_files.py script will not update all the new files in $GOA_VAR so users should install these files manually from the tar file posted on the GIPSY 6.3 download page.

WARNING: The GIPSY 6.3 verify test answers were generated using this release of the GOA_VAR files. Please use verify script and answers that are provided with the specific version of GIPSY. If your tests fail please verify that you are using this latest release of the GOA_VAR files.

8 Installing Required Third Party Software

GIPSY has been built and runs on Red Hat Enterprise Linux release 4 (RHEL4) 32-bit and release 5 (RHEL5) 64-bit systems with the following additions:

1. Perl 5.8.8. This is standard with RHEL5. We install this in /usr/local/bin on our RHEL4 hosts, but it may be installed anywhere on the path.
2. Python 2.4.1 is used and installed in /usr/local/bin on our RHEL4 hosts. This must be built with zlib enabled. Python 2.4.3 is standard with RHEL5 and is used. We also provide libraries built for Python 2.6 and 2.7.
3. Numpy. The version needed depends on the version of Python used and the architecture. This must be built for both RHEL4 and RHEL5. This is only used by the program pr2p.
4. Tcsh 6.12 or newer. Standard with both RHEL4 and RHEL5.
5. Gnuplot 4.0. This is standard with RHEL5, but must be built for RHEL4
6. Ncompress 4.2.4. Standard with both RHEL4 and RHEL5.
7. teqc – This program is distributed by UNAVCO and only needed when using the –F option in ninja. See: http://facility.unavco.org/software/teqc/teqc.html.

Note that older versions of Perl and Python are delivered with RHEL4. Our experience is that GIPSY will run with these older versions of Perl and Python, however these are not the version that we tested against, and we are not supporting this configuration.

Architecture	Python version	Numpy version	Notes
32-bit	2.4	1.3.0	
32-bit	2.6.2	1.4.1	Works for Scientific Linux 6
32-bit	2.7.1	1.5.1	
64-bit	2.4.3	1.3.0	
64-bit	2.6.2	1.6.1	Default 2.6 libraries
64-bit	2.7.1	1.6.2	
64-bit	2.6.6	1.4.1	Alternate for Scientific Linux 6. Required additional configuration to use.

On 32-bit Linux, we also provide Python libraries built using Python 2.6 and Numpy 1.4.1 or Python 2.7 and Numpy 1.5.1. On 64-bit Linux,

Since Ubuntu is an increasingly popular Linux distribution, and the packages in release 10.04 are very similar to our configuration, we are delivering beta tarballs of GIPSY built on Ubuntu 10.04 LTS for both the 32- and 64-bit platform. The tarball has been tested on the 12.04 LTS release of Ubuntu. The tarball only requires the default packages for this release plus:

1. python-numpy
2. tcsh
3. gnuplot-x11
4. ncompress

All of the above are available from the Ubuntu repositories, and can be installed with apt-get.

The 32-bit Red Hat build and Ubuntu builds have not been tested as extensively as the 64-bit Red Hat build.

Also GIPSY has been successfully installed and run on other versions of Linux such as various versions of Fedora. Users do this at their own risk as we are currently not officially supporting additional platforms.

No symbolic links in the /usr or /bin directory are required to run rootless GIPSY.

9 Installing GIPSY from a Tarball

All of GIPSY can be installed under two directories: a read-only tree containing executables and default files, and a writable tree that contains tables that must be updated. This delivery includes a cron script that should be run daily to keep these tables updated. There are also startup scripts (rc_gipsy.csh and rc_gipsy.sh) included in this delivery which users should run to set up their environment. You should also create a **gipsy** group that can be used to control write access to the tables directory.

Before installing GIPSY, you must decide where to install GIPSY, how you will restrict write access to the tables directory, and which user will run the cron script to update the tables directory. Following the Filesystem Hierarchy Standard (FHS), we recommend that you install the read-only GIPSY tree in /opt/goa-6.3 and the writable tables directory tree in /var/opt/goa-var. Note that the read-only directory has a version number, while the writable directory is unversioned so that it may be shared among different versions of GIPSY. To control write access to the tables directory, we recommend that you create a **gipsy** group and that all files in /var/opt/goa-var have their permissions set so that they are writable by members of that group. Finally, some member of the **gipsy** group is chosen to run the update cron script. In the following instructions, we assume that you are following the above recommendations.

1. Obtain the gzipped tar file appropriate to your machine. This is goa-6.3-RHEL4-32.tgz for 32-bit Red Hat Enterprise Linux release 4, goa-6.3-RHEL5-64.tgz for 64-bit Red Hat Enterprise Linux release 5, goa-6.3-U10.04-32.tgz for 32-bit Ubuntu 10.04 LTS Linux, or goa-6.3-U10.04-64.tgz for 64-bit Ubuntu 10.04 LTS Linux. Also obtain the goa-var.tgz tarball for the tables (goa-var) directory.
2. Decide where to install the two tarballs. For this example, we assume that the read-only component will be install in /opt/goa-6.3, and the writable component will be installed in /var/opt/goa-var. However, you may choose other locations.
3. Log in as root. This is required to install under /opt, but may not be required if you decide to install in a different location.
4. Extract the GIPSY files into the installation directory.

   ```
   $ cd /opt
   $ tar xvfz  ~/goa-6.3-RHEL5-64.tgz
   ```

5. If you are running on a 64-bit Scientific Linux 6 platform (with numpy 1.4.1), replace the python2.6 library with the alternate library.

   ```
   $ cd goa-6.3/lib/Linux-x86_64

   $ mv python2.6 python2.6-rh5

   $ mv python2.6-sl6 python2.6
   ```

6. Extract the GIPSY tables into the tables directory.

   ```
   $ cd /var/opt/
   $ tar xvfz ~/goa-var.tgz
   ```

7. Make sure that the tables directory is owned by and is writable by members of the team responsible for maintaining this area. For example, the following commands can be used to set permissions for a **gipsy** group.

   ```
   $ chgrp -R gipsy /var/opt/goa-var
   $ chmod -R g+w /var/opt/goa-var
   ```

8. If you did not choose the standard installation directories: /opt/goa-6.3 and /var/opt/goa-var, you must edit the startup scripts (rc_gipsy.csh and rc_gipsy.sh). which are located in the read-only installation directory. This can be done by running the admin/fix_rc_gipsy.py script.

   ```
   $ /opt/goa-6.3/admin/fix_rc_gipsy.py –v /var/opt/goa-var
   ```

9. Log in as a user who is a member of the GIPSY group, and create a crontab entry.
   ```
   $ crontab -e
   ```
   The entry should run the update script once daily at some late hour
   ```
   #
   # keep GIPSY /goa/etc files current
   #
   33 5 * * * /opt/goa-6.3/crons/update_gipsy_files.py -u
   ```
   If running GIPSY on a laptop consider using anacron instead of cron.

   This cron script updates various auxiliary files that are used by GIPSY such as antenna calibration files, CA-P bias tables, Earth Orientation Parameter tables, leap second data, receiver type tables, PRN to SVN tables, etc. Users are advised to regularly update these files for optimal results when processing current data. However, these updates are usually not critical for processing historical data, in particular data older than the last time these auxiliary files were updated. Occasional changes to these tables will affect processing of historical data.

10 Run the Verify Tests

Once GIPSY has been installed, the verify script should be run to test basic GIPSY functionality. The verify script tests GIPSY's precise point positioning, precise orbit determination, and ambiguity resolution functionality, comparing the results to expected answers. Network connectivity is not required for any of these positioning tests.

However, the verify script also includes one test that verifies a capability to download JPL orbit and clock products in native GIPSY format from an anonymous JPL ftp server. Failure of this "ftp" test is only a concern for users that will actively use the capability in some GIPSY programs, such as gd2p.pl and pr2p, to automatically download JPL's orbit and clock products when positioning. Note, that such a failure probably indicates that the update_gipsy_files.py script is not updating the auxiliary files used by GIPSY.

1. Log in as some GIPSY user.
2. Source the startup file. For C-shell users:
   ```
   $ source /opt/goa-6.3/rc_gipsy.csh
   ```
   or for bash, ksh, or Bourne shell users:
   ```
   $ . /opt/goa-6.3/rc_gipsy.sh
   ```
3. Update the tables directories:
   ```
   $ $GOA/crons/update_gipsy_files.py -u
   ```
4. Make sure that you don't have a verify directory in the current working directory.
5. Extract a copy of the verify tests.
   ```
   $ tar xvfz $GOA/verify.tar.gz
   ```
6. Enter the verify directory and run the test:
   ```
   $ cd verify
   $ ./verify.py
   ```

Notes:

The verify tests take approximately 15 minutes to complete on a 3GHz Xenon processor.

WARNING: The GIPSY 6.3 verify test answers were generated using updated GOA_VAR files. If your tests fail please first check that you are using the latest release of the GOA_VAR files that accompany this GIPSY release. For details see Section 7.

11 Running Rootless GIPSY

Rootless GIPSY depends on environment variables rather than hard-coded paths to find files. Sourcing one of the provided startup files can set these environment variables.

Environment Variable	Description
GOA	This is the root directory of a complete and installed Rootless GIPSY tree.
GOA_VAR	This is the root directory of the GIPSY tables
ARCH	This defines the system architecture. Currently Linux86 (32-bit Linux) and Linux-x86_64 (64-bit Linux) are supported.
PATH	The search path for executables. To run GIPSY, it must include $GOA/bin, $GOA/bin/$ARCH, and the current working directory (.).
MANPATH	The search path for man pages. It must include $GOA/man to find the GIPSY man pages.

12 Systems Tested On

The minimum system that the RHEL4 version was successfully tested on was a Intel(R) Xeon(TM) CPU 1700MHz with 256 KB cache and 2Gb RAM with 1Gb swap file. It was successfully tested on faster machines with more RAM and unsuccessfully tested on an Intel machine with 256Mb of RAM.

Initial tests run on Fedora and Ubuntu Linux were successful, but our testing was not nearly as extensive as it was on RHEL4 and RHEL5.

13 Installing Release 6.3 on the Same Host as Prior Versions

We have successfully installed many rootless versions of GIPSY on the same host. For the user, the important point to remember is to set the environment correctly. You can have separate runtime table files (goa-var) for each tree with its own update cron, or have a single tables directory shared by the different GIPSY versions. We strongly recommend that you run the newest cron script when you have several versions of GIPSY referencing a common set of configuration files, since we may add files to future releases of GIPSY. If you use a common tables directory for GIPSY 6.3 and prior releases, some of the verify tests may fail for reasons described in the section on GOA_VAR.