



The GPS Toolkit

A User's Guide for Scientists, Engineers and Students

Tracie Conn, Tom Gaussiran, R. Benjamin Harris, Jon Little
Richard Mach, David Munton, Brent Renfro, Brian Tolman
Timothy Craddock

Applied Research Laboratories, The University of Texas at Austin

Martin Vermeer

Department of Surveying, Helsinki University of Technology

August 8, 2011
GPSTk Revision 1038

Contents

I	Theory	3
1	The Global Positioning System in a Nutshell	5
1.1	GPS in a Nutshell	5
1.1.1	The Position Solution	5
1.2	GPS Data Sources	7
1.2.1	GPS File Formats	7
1.2.2	Receiver Protocols	7
2	GPS File Formats	9
2.1	RINEX	9
2.2	FIC	9
2.3	SP-3	10
II	Usage, Examples & Notes	11
2.4	<i>ash2mdp ash2xyz</i>	14
2.4.1	Overview	14
2.4.2	Usage	14
2.4.3	Notes	14
2.5	<i>ats2mdp</i>	15
2.5.1	Overview	15
2.5.2	Usage	15
2.6	<i>bc2sp3</i>	16
2.6.1	Overview	16
2.6.2	Usage	16
2.7	<i>CalcDOPs</i>	17
2.7.1	Overview	17
2.7.2	Usage	17
2.7.3	Notes	18
2.8	<i>calgps</i>	19
2.8.1	Overview	19
2.8.2	Usage	19
2.8.3	Examples	19
2.8.4	Notes	19

2.9	<i>compSatVis compStaVis</i>	20
2.9.1	Overview	20
2.9.2	Usage	20
2.9.3	Examples	20
2.10	<i>ConstellationList</i>	22
2.10.1	Overview	22
2.10.2	Usage	22
2.10.3	Examples	22
2.10.4	Notes	22
2.11	<i>daa</i>	24
2.11.1	Overview	24
2.11.2	Usage	24
2.12	<i>DiscFix</i>	26
2.12.1	Overview	26
2.12.2	Usage	26
2.12.3	Examples	27
2.13	<i>DOPcalc</i>	28
2.13.1	Overview	28
2.13.2	Usage	28
2.14	<i>EditRinex</i>	29
2.14.1	Overview	29
2.14.2	Usage	29
2.14.3	Examples	31
2.15	<i>ephdiff</i>	32
2.15.1	Overview	32
2.15.2	Usage	32
2.15.3	Examples	32
2.15.4	Notes	32

The goal of the GPSTk project is to provide a world class, open source computing suite to the satellite navigation community. It is our hope that the GPSTk will empower its users to perform new research and to create new applications.

GPS users employ practically every computational architecture and operating system. Therefore the design of the GPSTk suite is as platform-independent as possible. Platform independence is achieved through use of the ANSI-standard C++ programming language. The principles of object-oriented programming are used throughout the GPSTk code base in order to ensure that the code is modular, extensible, and maintainable.

The GPSTk suite consists of a core library and a set of applications. The library provides a wide array of functions that solve processing problems associated with GPS such as processing or using RINEX. The library is the basis for the more advanced applications distributed as part of the GPSTk suite.

The GPSTk is sponsored by Space and Geophysics Laboratory, within the Applied Research Laboratories at the University of Texas at Austin (ARL:UT). GPSTk is the by-product of GPS research conducted at ARL:UT since before the first satellite launched in 1978; it is the combined effort of many software engineers and scientists. In 2003 the research staff at ARL:UT decided to open source much of their basic GPS processing software as the GPSTk.

Part I

Theory

Chapter 1

The Global Positioning System in a Nutshell

The Global Positioning System is actually a U.S. government satellite navigation system that provides a civilian signal. As of this writing, the signal is broadcast simultaneously by a constellation of 29 satellites each with a 12 hour orbit. From any given position on the Earth, 8 to 12 satellites are usually visible at a time.

1.1 GPS in a Nutshell

Each satellite broadcasts spread spectrum signals at 1575.42 and 1227.6 MHz, also known as L1 and L2, respectively. Currently the civil signal is broadcast only on L1. The signal contains two components: a time code and a navigation message. By differencing the received time code with an internal time code, the receiver can determine the distance, or range, that the signal has traveled. This range observation is offset by errors in the (imperfect) receiver clock; therefore it is called a pseudorange. The navigation message contains the satellite ephemeris, which is a numerical model of the satellite's orbit.

GPS receivers record, besides the pseudorange, a measurement called the carrier phase (or just phase); it is also a range observation like the pseudorange, except (1) it has an unknown constant added to it (the phase ambiguity) and (2) it is much smoother (about 100 times less measurement noise than the pseudorange!), which makes it useful for precise positioning. Because of the way it is measured, the phase is subject to random, sudden jumps; these discrete changes always come in multiples of the wavelength of the GPS signal, and are called cycle slips.

1.1.1 The Position Solution

The standard solution for the user location requires a pseudorange measurement and an ephemeris for each satellite in view. At least four measurements are

required as there are four unknowns: 3 coordinates of position plus the receiver clock offset. The basic algorithm for the solution is described in the official GPS Interface Control Document, or ICD-GPS-200. The position solution is corrupted due to two sources of error: errors in the observations and errors in the ephemeris.

Reducing Measurement Errors

The GPS signal travels through every layer of the Earth's atmosphere. Each layer affects the signal differently. The ionosphere, which is the high-altitude, electrically charged part of the atmosphere, introduces a delay, and therefore a range error, into the signal. The ionosphere delay can be predicted using a model. However, the accuracy of ionosphere models is limited. A better alternative is to measure and remove the ionosphere delay. Measurement of the ionosphere delay is possible by taking advantage of the fact that the delay is frequency dependent. It can be directly computed if you have data on both the GPS frequencies. There is also a delay due to the troposphere, the lower part of the atmosphere. Like the ionosphere delay, the atmosphere delay can be either predicted or derived from measurements. There are many other errors associated with the GPS signal: multipath reflections and relativistic effects are two examples.

More precise applications reduce the effect of error sources by a technique referred to as differential GPS (DGPS). By differencing measurements simultaneously collected by the user and a nearby reference receiver, the errors that are common to both receivers (most of them) are removed. The result of DGPS positioning is a position relative to the reference receiver; adding the reference position to the DGPS solution results in the absolute user position.

The alternative to DGPS is to explicitly model and remove errors. Creating new and robust models of phenomena that effects the GPS signal is an area of active research at ARL:UT and other laboratories. The positioning algorithm can be used to explore such models. Essentially, the basic approach is to turn the positioning algorithm inside out to look at the corrections themselves. For example, observations from a network of receivers can create a global map or model of the ionosphere.

Improved Ephemerides

The GPS position solution can be directly improved by using an improved satellite ephemeris. The U.S National Geospatial-Intelligence Agency (NGA) generates and makes publicly available a number of precise ephemerides, which are more accurate satellite orbits [?], [?]. Satellite orbits described by the broadcast navigation message have an error on the order of meters; the precise ephemeris has decimeter accuracy. The International GPS Service (IGS) is a global, civil cooperative effort that also provides free precise ephemeris products [?]. Global networks of tracking stations produce the observations that make generation of the precise ephemerides possible.

1.2 GPS Data Sources

GPS observation data from many tracking stations are freely available on the Internet. Many such stations contribute their data to the IGS. In addition, many networks of stations also post their data to the Internet; for example the Australian Regional GPS Network (ARGN) [?] and global cooperatives such as NASA's Crust Dynamics Data Information System (CDDIS) [?].

1.2.1 GPS File Formats

Typically GPS observations are recorded in a standardized format developed by and for researchers. Fundamental to this format is the idea that the data should be independent of the type of receiver that collected it. For this reason the format is called Receiver INdependent Exchange, or RINEX. Another format associated with GPS is SP-3, which records the precise ephemeris. The GPSTk supports both RINEX and SP-3 formats.

1.2.2 Receiver Protocols

GPS receivers have become less expensive and more capable over the years, in particular handheld and mobile GPS receivers. The receivers have many features in common. All of the receivers output a position solution every few seconds. All receivers store a list of positions, called waypoints. Many can display maps that can be uploaded. Many can communicate with a PC or handheld to store information or provide position estimates to plotting software.

Typically communication with a PC and other system follows a standard provided by the National Marine Electronics Association called NMEA-0183. NMEA-0183 defines an ASCII based format for communication of position solutions, waypoints and a variety of receiver diagnostics. Here is an example of a line of NMEA data, or sentence:

```
$GPGLL,5133.81,N,00042.25,W*75
```

The data here is a latitude, longitude fix at 51 deg 33.81 min North, 0 deg 42.25 min West; the last part is a checksum.

As a public standard, the NMEA-0183 format has given the user of GPS freedom of choice. NMEA-0183 is the format most typically used by open source applications that utilize receiver-generated positions.

Closed standards are also common. SiRF is a proprietary protocol that is licensed to receiver manufacturers. Many receiver manufacturers implement their own binary protocols. While some of these protocols have been opened to the public, some have been reverse engineered.

Chapter 2

GPS File Formats

A variety of file formats are supported within the GPSTk. The file formats generally store GPS observation data or data related to processing of GPS observables. In this section, a summary of the file formats supported within the GPSTk is presented along with a brief rationale of why each format is supporting within the GPSTk and where to find additional information on the format.

2.1 RINEX

The Receiver INdependent EXchange (RINEX) format was developed by the National Geodetic Survey (NGS) in the U.S. and the University of Berne in Switzerland. RINEX is actually three format definitions that allow storage of GPS observations, GPS navigation message information, and meteorological data associated with GPS observations. GPSTk contains classes to both read and write RINEX V2.1 data files of all types (observation, navigation message, and meteorological). RINEX has undergone a number of revisions since its inception. Each revision is defined using a standard [?], [?], [?], [?].

2.2 FIC

The Floating, Integer, Character (FIC) format was developed in the mid-80s as a relatively machine-independent way to store GPS observation and navigation message data while retaining receiver specific characteristics. Over time, the RINEX format (see above) proved more popular with users and use of the observation records within the FIC format faded away. However, the FIC records associated with GPS navigation message data are still supported within the GPSTk because these records retain some data quantities that are not contained within the RINEX navigation message file. For example, RINEX makes few provisions for storing the almanac data contained in Subframe 4 and Subframe 5. Like RINEX, a standards document defines FIC [?].

2.3 SP-3

The SP-3 format stores ephemeris information for satellites. Usually SP-3 is used for storage of GPS precise ephemerides. GPSTk supports both SP-3a and SP3-c formats. SP-3 was originally designed by NGS. Standards documents describe the specific details of the SP-3 formats [?], [?].

Part II

Usage, Examples & Notes

	Tool	Description	Execution Example
Transforms	calgps	generates a GPS calendar	calgps -Y 2004
	poscvrt	converts a given input position to other position formats	poscvrt --geodetic="30.28 262.26700 167.64"
	timeconvert	converts given input time to other time formats	timeconvert --calendar="07 04 2006"
	wheresat	outputs expected location of a satellite	wheresat -b arl2100.06n -p 3
Collecting & Converting	rtAshtech	records observations from an Ashtech receiver	rtAshtech -p /dev/ttyS1 -o "minute%03j%02H%02m.%06yo"
	ficfica ficafic fic2rin	convert fic files between ASCII, binary, and RINEX formats	fic2rin fic2100.06 rin121.06n
	mdp2fic mdp2rinex	convert MDP files to FIC or RINEX files	mdp2rinex -i mdpfile -o arl2100.06o
	novaRinex	convert Novatel files to RINEX	novaRinex --input nova2100.06 --obstype L1
	navdmp	dumps information from nav files to human readable formats	navdmp -i arl2100.06n -o arl2100.06.dmp
	RinexDump	dumps observation data for specified satellites from a RINEX file	RinexDump arl2100.06o 3 4 L1 L2
Comparing & Validating	ephdiff	compares the satellite positions from two ephemeris sources	ephdiff arl2100.06n fic2100.06
	ficdiff	compares contents of two FIC files	ficdiff fic12100.06 fic22100.06
	ficcheck ficacheck	reads a FIC file and checks it for errors reporting the first found	ficcheck fic2100.06 -t "07/20/2006 11:00:00"
	rowdiff rnwdiff rnwdiff	compares contents of two RINEX files	rowdiff arl1210.06o arl22100.06o
	rowcheck rnwcheck rnwcheck	read Rinex files and checks it for errors reporting the first found	rnwcheck arl210.06n -e "07/20/2006 11:00:00"
	navsum RinSum	summarizes the contents of nav/Rinex files	RinSum -i arl2100.06o --EpochBeg 2006,07,20,13,20,00
	mdptool	summarizes MDP data	mdptool -i mdpfile --pvt --obs
	reszilla	computes range residuals or zero baseline differences	reszilla -o arl210.06o -e arl2100.06n
Editing Data	mergeFIC	sorts and merges input FIC files into a single file	mergeFIC -i fic12100.06 -i fic22100.06 -o ficmerge2100.06
	mergeRinObs, -Nav, -Met	sorts and merges RINEX files	mergeRinNav -i arl2100.06n -i arl2110.06n arl210-211.06n
	NavMerge	merges RINEX nav files into a single file	NavMerge -oarlnavs.06n arl2100.06n arl2110.06n
	rinexthin	decimates an input RINEX observation files to desired data rate	rinexthin -f arl2100.06o -s 30 -o arl2100thin.06n
	ResCor	edits RINEX files and computes corrections	ResCor -IFar12100.06o -OFar12100mod.06o -DS12,12:00:00
	DiscFix	cycle slip corrector	DiscFix --inputfile arl2100.06o --dt 1.5
Iono	IonoBias	solves interfrequency biases and a simple ionosphere model	IonoBias --input arl2100.06o --nav arl2100.06n --XSat 3
	TECMaps	creates maps of Total Electron Content (TEC)	TECMaps --input arl2100.06o --nav arl2100.06n --LinearFit
Positioning	PRSolve	generates autonomous position solution	PRSolve -o arl2100.06o -n arl2100.06nn --XPRN 12
	rinexpvt	generates autonomous position solution	rinexpvt -o alr2100.06o -n arl2100.06n
	DDBase	computes a network solution using carrier phase	DDBase ... --ObsFile arl2100.06o --PosXYZ x,y,z,1 --Fix
	vecsol	estimates short baseline using range or carrier phase	vecsol station12100.06o station22100.06o

Table 2.1: GPSTk Applications, categorized, with execution examples.

2.4 *ash2mdp ash2xyz*

2.4.1 Overview

These applications process Ashtech Z(Y)-12 observation and ephemeris data and output satellite positions and ionospheric corrections in either MDP or XYZ format.

2.4.2 Usage

Optional Arguments		<i>ash2mdp ash2xyz</i>
Short Arg.	Long Arg.	Description
-i		Where to get data from. The default is to use stdin.
-o		Where to send the output. The default is to use stdout.
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-w	-week=NUM	The full GPS week in which this data starts. Use this option when the start time of the data being processed is not during this week.
-c	-code=ARG	Restriction for source of observation data collected via L1/L2 Y code tracking will be used. Options are "Y", "P", and "codeless." XYZ only.
-s	-offset=NUM	Output SV positions at a time offset from the current time. Give a positive or negative integer of seconds. XYZ only.
-n	-num_points=NUM	Width of the exponential filter moving window, in number of points (default is 36). XYZ only.

2.4.3 Notes

Input is on the command line, or of the same format in a file (-f<file>).

2.5 *ats2mdp*

2.5.1 Overview

This application converts ATS binary format data to MDP format.

2.5.2 Usage

Optional Arguments		<i>ats2mdp</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-i	-input=ARG	A file from which to take the input. The default is stdin.
-o	-output=ARG	A file from which to receive the output. The default is stdout.

2.6 *bc2sp3*

2.6.1 Overview

This application reads RINEX navigation file(s) and writes to SP3 (a or c) file(s).

2.6.2 Usage

Optional Arguments		<i>bc2sp3</i>
Short Arg.	Long Arg.	Description
-in		Read the input file (repeatable).
-out		Name the output file. Default is sp3.out.
-tb		Output beginning epoch; <time> = week, sec-of-week (earliest in input).
-te		Output ending epoch; <time> = week, sec-of-week (latest in input).
-outputC		Output version c (no correlation) (otherwise a).
-msg		Add message as a comment to the output header (repeatable).
-verbose		Output to screen: dump headers, data, etc.
-help		Print this message and quit.

2.7 *CalcDOPs*

2.7.1 Overview

This application reads SV almanac data (one file per day of observation) from a FIC, FICA or a RINEX navigation file, then computes and displays visibility information. Dilution of precision values from that data are calculated using standard methods. See for example:

- AIAA GPS Theory and Applications vol. 1, Ed. Parkinson & Spilker, pp. 414.
- GPS Signals, Measurements, and Performance, 2ed., Misra & Enge, pp. 203.

2.7.2 Usage

<i>CalcDOPs</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
-i<inputfile>		Input file for day to be calculated.
Optional Arguments		
-p <inputfile>		Input file for previous day (ephemeris mode only).
-o <outputfile>		Grid output file (default DOPs.out).
-sf <outputfile>		Stats output file (default DOPs.stat).
-tf <outputfile>		Time steps output file (default DOPs.times).
-l <outputfile>		Log output file (default DOPS.log).
-rs		Read from stats file.
-a		Work in almanac mode (ephemeris mode is default).
-w -s <week> <sow>		Starting time tag.
-x <prn>		Exclude satellite PRN.
-t <dt>		Time spacing.
-na		North America only.
-d		Dump grid results at each time step (time-intensive).
-h	-help	Output options info and exit.
-v		Print version info and exit.

2.7.3 Notes

Abort/failure codes given on return:

-1	could not open input data file
-2	could not identify input data file type
-3	fewer than 4 satellite almanacs available
-4	could not allocate GridStats data types
-5	could not open input stats file
-6	could not open output grid file
-7	could not open output stats file
-8	could not open output log file

Essential variables not documented below at declaration:

NtrofN	number of cells/times with < 5 SVs visible during the time period
NpeakH	number cells/times w/ HDOP > 10
NpeakP	number cells/times w/ PDOP > 10
IworstN	index in Grid[] of cell with worst nsvs (number of satellites)
IworstH	index in Grid[] of cell with worst HDOP
IworstP	index in Grid[] of cell with worst PDOP
WorstN	value of nsvs at IworstN
WorstH	value of HDOP at IworstH
WorstP	value of PDOP at IworstP
TworstN	time tag (CommonTime class) of WorstN
TworstH	time tag (CommonTime class) of WorstH
TworstP	time tag (CommonTime class) of WorstP

1. GPS only, using PRNs hard-wired to SV numbers 1-32.
2. Elevation limit is hard-wired to 5 degrees above horizon.
3. "North America" means the northern half-hemisphere: -180 to 0 deg long., 0 to 90N latitude.
4. Ephemeris mode is default, almanac mode is optional. Ephemeris mode is preferred, because it excludes unhealthy satellites for any time when they transmitted an unhealthy flag. Almanac mode will generally not exclude SVs when they were unhealthy (typical), or may erroneously exclude them for an entire day (rarely).
5. If 2 input files are given, the default start time is midnight on the day to be calculated. A previous-day input file can be given only in ephemeris mode, not almanac.
6. The code uses geodetic coordinates for all calculations.
7. The -d option is useful for e.g. making movies of DOPs throughout a day.

2.8 *calgps*

2.8.1 Overview

This application generates a dual GPS and Julian calendar to either the command line or to a graphics file. The arguments and format are inspired by the UNIX ‘cal’ utility. With no arguments, the current argument is printed. The last and next month can also be printed. Also, the current or any given year can be printed.

2.8.2 Usage

calgps

Optional Arguments

Short Arg.	Long Arg.	Description
-h	-help	Generates help output.
-3	-three-months	Prints a GPS calendar for the previous, current, and next month.
-y	-year	Prints a GPS calendar for the entire current year.
-Y	-specific-year=NUM	Prints a GPS calendar for the entire specified year.
-p	-postscript=ARG	Generates a postscript file.
-s	-svg=ARG	Generates an SVG file.
-e	-eps=ARG	Generates an encapsulated postscript file.
-v	-view	Try to launch an appropriate viewer for the file.
-n	-no-blurb	Suppress GPSTk reference in graphic output.

2.8.3 Examples

```
> calgps -3
```

```

                                Jun 2011
1638                                1-152  2-153  3-154  4-155
1639    5-156  6-157  7-158  8-159  9-160 10-161 11-162
1640   12-163 13-164 14-165 15-166 16-167 17-168 18-169
1641   19-170 20-171 21-172 22-173 23-174 24-175 25-176
1642   26-177 27-178 28-179 29-180 30-181
```

```

                                Jul 2011
1642                                1-182  2-183
1643    3-184  4-185  5-186  6-187  7-188  8-189  9-190
1644   10-191 11-192 12-193 13-194 14-195 15-196 16-197
1645   17-198 18-199 19-200 20-201 21-202 22-203 23-204
1646   24-205 25-206 26-207 27-208 28-209 29-210 30-211
1647   31-212
```

```
. . .
```

2.8.4 Notes

If multiple options are given only the first is considered.

2.9 *compSatVis compStaVis*

2.9.1 Overview

compSatVis computes satellite visibility. *compStaVis* computes station visibility.

2.9.2 Usage

<i>compSatVis compStaVis</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
-o	-output-file=ARG	Name of the output file to write.
-n	-nav=ARG	Name of navigation file.
-c	-mscfile=ARG	Name of MS coordinates file.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-p	-int=ARG	Interval in seconds.
-e	-minelv=ARG	Minimum elevation angle.
-t	-navFileType=ARG	FALM, FEPH, RNAV, YUMA, SEM, or SP3.
-m	-min-sta=ARG	Minimum number of stations visible simultaneously. <i>compStaVis</i> only.
-m	-max-SV=ARG	Maximum number of SVs tracked simultaneously. <i>compSatVis</i> only.
-D	-detail	Print SV count for each interval.
-x	-exclude=ARG	Exclude station.
-i	-include=ARG	Include station.
-s	-start-time=TIME	Start time of evaluation ("m/d/y H:M").
-z	-end-time=TIME	End time of evaluation ("m/d/y H:M").

2.9.3 Examples

Generating satellite visibility statistics using the SEM almanac from the USCG Navigation Center.

This example loads SEM almanac data from the file *current.al3* and a list of station locations from the file *stations.msc*. It then calculates the number of satellites visible to each station found at each 60 sec interval from 0000Z to 2356Z of Jan 13, 2008. using a 10 degree minimum elevation angle. The results are written to the file *visout.txt*. Note the use of a specific start time. The SEM and Yuma almanac formats contain an almanac reference week, which is generally in the range 0-1023 (the existing format definitions are ambiguous and SEM and Yuma almanacs with full week numbers have been reported, at least anecdotally). If the -s command is not specified, *compSatVis* will use whatever reference time is given in the almanac file, which may result in unexpected results.


```
user@host:~$ compSatVis -ovisout.txt -ncurrent.al3 -tSEM  
-cstations.msc -e10 -p60 -s"01/16/2008 00:00"
```

Generating station visibility statistics using the SEM almanac from the USCG Navigation Center.

Same as the previous example, however, the values calculated and the statistics will reflect the number of stations visible to each satellite.

```
user@host:~$ compSatVis -ovisout.txt -ncurrent.alm -tYUMA -cstations.msc -e10 -p60 -s"01/13/2008 00:00"
```

Generating satellite visibility statistics using the Yuma almanac from the USCG Navigation Center.

Similar to the first example, but the statistics are computed over four complete days.

```
user@host:~$ compSatVis -ovisout.txt -ncurrent.alm -tYUMA -cstations.msc -e10 -p60 -s"01/13/2008 00:00"
```

Generating satellite visibility statistics using SP3 files.

Similar to the first example, however, navigation message data are from three SP3 files. It is necessary to load three SP3 files to cover the default sidereal day period because the methods that calculate SV positions from the SP 3 data use interpolation and need data from the previous day and the following day in order to have sufficient points for the interpolation. In this example in which no evaluation period is specified, compSatVis derives coverage for the "middle day" for the period.

```
user@host:~$ compSatVis -ovisout.txt -napc14622 -napc14623 -napc14624 -tSP3 -cstations.msc -e10 -p60 -s"01/13/2008 00:00"
```

2.10 *ConstellationList*

2.10.1 Overview

ConstellationList provides lists of the GPS SV PRN ID active/inactive on a given day.

2.10.2 Usage

		<i>ConstellationList</i>
Required Arguments		
Short Arg.	Long Arg.	Description
-i	-input-file=<arg>	The name of the Constellation Definition file(s) to read.
-y	-year=<arg>	Year of interest.
-j	-day-of-year=<arg>	Day of year.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-O	-OpsAd	Assume input file is Op Advisory format (CSV is default).
-b	-Base24	List PRNs in Base 24 Constellation.
-x	-excessSVs	List PRNs in use, but in excess of the Base 24 Constellation.
-n	-notBase24	List PRNs NOT used in Base 24 Constellation.
-s	-SVN Output	Output SVN in place of PRN (not valid for -O).

2.10.3 Examples

```
>ConstellationList -iSlot2008.csv -tC -y2008 -j001 -b
2, 3, 4, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 27, 28, 30, 31
>ConstellationList -iSlot2008.csv -tC -y2008 -j001 -n
1, 5, 7, 24, 25, 26, 29, 32
>ConstellationList -iSlot2008.csv -tC -y2008 -j001 -n
1, 5, 24, 25, 26
>ConstellationList -iSlot2008.csv -tC -y2008 -j001 -b -s
61, 33, 34, 36, 38, 39, 40, 46, 58, 43, 41, 55, 56, 53, 54, 59, 51, 45, 47, 60, 27, 44, 30, 52
```

2.10.4 Notes

In particular, *ConstellationList* provides a means of determining which SVs are members of the “Base 24” constellation and which are not. This is usually of no interest to the general user, but is important in cases where programs are evaluating GPS performance against a defined standard such as the Standard Positioning Service (SPS) Performance Standard (PS) which defines GPS performance in terms of the “official” constellation as opposed to the superset which is normally available. The results are provided as a text list of comma separated values on a single line, suitable for piping into another process.

The complication in this process is that the information regarding the orbit plane/slot of each SV is not available from the broadcast message. It must be obtained “external to the system”. One source of such information is the USCG Navigation Center website which stores the Operational Advisories. These advisories provide the relationship between SVs and plane/slot assignments. ConstellationList is programmed to read the advisories as an input format, as long as the format of the advisories does not change.

As an alternative to the Operational Advisories, ARL:UT has prepared files of the assignments for specific years as comma separated value files. Each line in these files represents the status on a given day and includes the mapping between the PRN IDs and the NAVSTAR numbers. These files have been hand-checked and are available in the GPSTk repository as Slot2007.csv and Slot2008.csv.

2.11 *daa*

2.11.1 Overview

This application performs a data availability analysis of the input data. In general, availability is determined by station and satellite position.

2.11.2 Usage

		<i>daa</i>
Required Arguments		
Short Arg.	Long Arg.	Description
-e	-eph=ARG	Where to get the ephemeris data. Acceptable formats include RINEX nav, FIC, MDP, SP3, YUMA, and SEM. Repeat for multiple files.
-o	-obs=ARG	Where to get the observation data. Acceptable formats include RINEX obs, MDP, smooth, Novatel, and raw Ashtech. Repeat for multiple files. If a RINEX obs file is provided, the position will be taken from the header unless otherwise specified.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
	-output=ARG	Output location (default is stdout).
-x	-independent=ARG	The independent variable in the analysis. The default is time.
-c	-msc=ARG	Station coordinates file.
-m	-msid=ARG	Station for which to process data. Used to select a station position from the msc file.
-t	-time-format=ARG	CommonTime format specifier used for times in the output. The default is "%Y %j %02H:%02M:%04.1f".
	-mask-angle=ARG	Ignore anomalies on SVs below this elevation. The default is 10 degrees.
	-track-angle=ARG	Assume the receiver starts tracking at this elevation. The default is 10 degrees.
	-time-mask=ARG	Ignore anomalies on SVs that haven't been above the mask angle for this number of seconds. The default is 0 seconds.
	-snr=ARG	Discard data with an SNR less than this value. The default is 20 dB-Hz.
-p	-position=ARG	Receiver antenna position in position (x,y,z) coordinates. Format as a string: "X Y Z".
-l	-time-span=ARG	How much data to process, in seconds.
	-ignore-prn=ARG	Specify the PRN of an SV to not report on in the output. Repeat to specify multiple SVs.
	-obs-interval=ARG	Specify the time interval, in seconds, between observations. The default is to scan the file to discover this via examination of the file.
-b	-bad-health	Ignore anomalies associated with SVs that are marked unhealthy.
-s	-smash-adjacent	Combine adjacent lines from the same PRN.

<code>-start-time=TIME</code>	Ignore data before this time.
<code>-stop-time=TIME</code>	Ignore any data after this time.

2.12 *DiscFix*

2.12.1 Overview

This application reads a RINEX observation data file containing GPS dual-frequency pseudorange and carrier phase measurements, divides the data into ‘satellite passes’, and finds and fixes discontinuities in the phases for each pass.

Output is a list of editing commands for use with program EditRinex. DiscFix will (optionally) write the corrected pseudorange and phase data to a new RINEX observation file. Other options will also smooth the pseudorange and/or debias the corrected phase.

DiscFix calls the GPSTk Discontinuity Corrector (GDC vers 5.3 7/14/2008).

2.12.2 Usage

		<i>DiscFix</i>
Required Arguments		
Short Arg.	Long Arg.	Description
	-inputdir	File containing more options.
	-dt	Time space in seconds of the data.
Optional Arguments		
Short Arg.	Long Arg.	Description
-f	-file	File containing more options.
	-beginTime	Start time of processing (BOF).
	-endTime	End time of processing (EOF).
	-decimate	Decimate data to specified time interval, in seconds.
	-forceCA	Use C/A code range, NOT P code. Default only if P absent.
	-gap	Minimum data gap in seconds separating satellite passes (600).
	-onlySat	Process only satellite (GPS SatID, e.g. G21).
	-exSat	Exclude satellite(s) (GPSSatID).
	-smoothPR	Smooth pseudorange and output in place of raw pseudorange.
	-smoothPH	Debias phase and output in place of raw phase.
	-smooth	Same as -smoothPR AND -smoothPH.
	-DClabel	Set Discontinuity Corrector parameter ‘label’ to ‘value’.
	-DChelp	Print a list of GDC parameters and their defaults, then quit.
	-logOut	Output log file name (df.log).
	-cmdOut	Output file name, for editing commands (df.out).
	-format	Output time format (gpstk::CommonTime) (%4F %10.3g).
	-RinexFile	RINEX (obs) file name for output of corrected data.
	-RunBy	RINEX header ‘RUN BY’ string for output.
	-Observer	RINEX header ‘OBSERVER’ string for output.
	-Agency	RINEX header ‘AGENCY’ string for output.
	-Marker	RINEX header ‘MARKER’ string for output.

	-Number	RINEX header 'NUMBER' string for output.
-h	-help	Print this syntax page and quit.
	-verbose	Print extended output to the log file.

2.12.3 Examples

```
> DiscFix --dt 1.5 --inputfile ar12800.06o
```

```
DiscFix, part of the GPS ToolKit, Ver 5.0 8/20/07, Run 2011/07/22 11:17:25  
DiscFix is writing to log file df.log  
DiscFix is writing to output file df.out  
DiscFix timing: 0.960 seconds.
```

2.13 *DOPcalc*

2.13.1 Overview

This application computes position, time, and geometric dilution of precision (DOP) parameters.

2.13.2 Usage

		<i>DOPcalc</i>
Required Arguments		Description
Short Arg.	Long Arg.	
-e	-eph=ARG	Where to get the ephemeris data. Acceptable formats include RINEX nav, FIC, MDP, SP3, YUMA, and SEM. Repeat for multiple files.
-o	-obs=ARG	Where to get the observation data. Acceptable formats include RINEX obs, MDP, smooth, Novatel, and raw Ashtech. Repeat for multiple files. If a RINEX obs file is provided, the position will be taken from the header unless otherwise specified.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-p	-position=ARG	User position in ECEF (x,y,z) coordinates. Format as a string: "X Y Z".
	-el-mask=ARG	Elevation mask to apply, in degrees. The default is 0.
-c	-msc=ARG	Station coordinate file.
-m	-msid=ARG	Monitor station ID number.

2.14 *EditRinex*

2.14.1 Overview

This application will open and read one RINEX file, apply editing commands, and write the modified RINEX data to another RINEX file(s). Input is on the command line, or of the same format in a file (-f<file>).

2.14.2 Usage

<i>EditRinex</i>		
Optional Arguments		
Short Arg.	Long Arg.	Description
-f	-file <file>	File containing more options.
-l	-log <file>	Output log file name.
-h	-help	Print syntax and quit.
-d	-debug	Print extended output info.
-v	-verbose	Print extended output info.
	<REC>	Rinex editing commands - following:

Rinex Editor Commands

Commands consist of an identifier and a comma-delimited data field; they may be separated by space(s) '-id <data>' (two minuses) or not '-id<data>' (one minus).

Examples are '-IF myFile' or '-IFmyFile'; '-HDc msg' or '-HD cmsg' or '-HDcmsg'; -BZ or -BZ; '-DD +<SV,OT,t>' or '-DD+ <SV,OT,t>' or '-DD+<SV,OT,t>'.

The data field contains no whitespace and sub-fields are comma-delimited. <SV> is a RINEX 'system and id' identifier, e.g. G27 (= GPS PRN 27); satellite system alone denotes 'all satellites this system', e.g. 'R' (GLONASS). <OT> is a RINEX observation type, e.g. L1 or P2, and is case sensitive. <time> is either <GPSweek,GPSsecOfWeek> or <year,mon,day,hour,min,second>.

File I/O:

-
- IF<file> Input RINEX observation file name [may be repeated] (required).
 - ID<dir> Directory in which to find input file.
 - OF<file> Output RINEX file name (required, or -OF<file>,<time>).
 - OF<f>,<time> At RINEX epoch <time>, close output file and open another named <f>.
 - OD<dir> Directory in which to put output file(s).

Output RINEX header:

-
- HDf If present, fill optional records in the output RINEX header
 - HDp<program> Set output RINEX header 'program' field.
 - HDr<run_by> Set output RINEX header 'run by' field.
 - HDo<observer> Set output RINEX header 'observer' field.

-HDA<agency> Set output RINEX header 'agency' field.
 -HDx<x,y,z> Set output RINEX header 'position' field to ECEF position (x,y,z).
 -HDm<marker> Set output RINEX header 'marker' field.
 -HDn<number> Set output RINEX header 'number' field.
 -HDc<comment> Add comment to output RINEX header (more than one allowed).
 -HDdc Delete all comments in output RINEX header.
 (NB -HDdc cannot delete comments created by *subsequent* -HDc commands).

Output RINEX observation types (also see 'Specific edit commands' below):

-AO<OT> Add observation type OT to header and observation data.
 -DO<OT> Delete observation type OT entirely (including in header).

Time-related edit commands:

-TB<time> Begin time: reject data before this time (also used for decimation).
 -TE<time> End time: reject data after this time.
 -TT<dt> Tolerance in comparing times, in seconds (default=1ms).
 -TN<dt> Decimate data to epochs = Begin + integer*dt (within tolerance).

Specific edit commands:

(Generally each '+' command (e.g DA+<time>) has a corresponding '-' command, and vice-versa; if not, end-of-file or beginning-of-file is assumed. Note that one-time commands are applied AFTER other commands of the same type.)

Delete commands:

-DA+<time> Delete all data beginning at this time.
 -DA-<time> Stop deleting data at this time.
 -DO<OT> Delete observation type OT entirely (including in header).
 -DS<SV> Delete all data for satellite SV entirely (SV may be system only).
 -DS<SV>,<time> Delete all data for satellite SV at this single time only.
 -DS+<SV>,<time> Delete all data for satellite SV beginning at this time.
 -DS-<SV>,<time> Stop deleting all data for satellite SV at this time.
 -DD<SV,OT,t> Delete a single RINEX datum(SV,OT,t) at time <t>.
 -DD+<SV,OT,t> Delete all (SV,OT) data, beginning at time <t>.
 -DD-<SV,OT,t> Stop deleting all (SV,OT) data at time <t>.
 (Deleting data for one OT means setting it to zero - as RINEX requires.)

Set commands:

-SD<SV,OT,t,d> Set data(SV,OT,t) to <d> at time <t>.
 -SS<SV,OT,t,s> Set ssi(SV,OT,t) to <s> at time <t>.
 -SL+<SV,OT,t,l> Set all lli(SV,OT,t) to <l> at time <t>.
 -SL-<SV,OT,t,l> Stop setting lli(SV,OT,t) to <l> at time <t> ('<l>' is optional).
 -SL<SV,OT,t,l> Set lli(SV,OT,t) to <l> at the single time <t> only.

Bias commands:

(BD commands apply only when data is non-zero, unless -BZ appears.)

-BZ Apply BD commands even when data is zero (i.e. 'missing').

-BD<SV,OT,t,d> Add the value of <d> to data(SV,OT,t) at time <t>.

-BD+<SV,OT,t,d> Add value <d> to data(SV,OT) beginning at time <t>.

-BD-<SV,OT,t,d> Stop adding <d> to data(SV,OT) at time <t> (',<d>' optional).

-BS<SV,OT,t,s> Add the value of <s> to ssi(SV,OT,t) at time <t>.

-BL<SV,OT,t,l> Add the value of <l> to lli(SV,OT,t) at time <t>.

2.14.3 Examples

Changing the APPROX position in the file acor1480.08o to the center of the Earth. Writes a new file called acor1480.08o.mod

```
user@host:~$ EditRinex -IFacor1480.08o -OFacor1480.08o.mod -HDx0,0,0
```

Removing a satellite, PRN 29, from an observation file, onsa2240.05o. Creates a new file, temp.o

```
EditRinex -IFonsa2240.05o -OFtemp.o
```

2.15 *ephdiff*

2.15.1 Overview

The application compares the contents of two files containing ephemeris data.

2.15.2 Usage

<i>ephdiff</i>		
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-f	-fic=ARG	Name of an input FIC file.
-r	-rinex=ARG	Name of an input RINEX NAV file.

2.15.3 Examples

```
> ephdiff -f fic06.187 -r arl2800.06n
```

Broadcast Ephemeris (Engineering Units)

PRN : 11

	Week(10bt)	SOW	DOW	UTD	SOD	MM/DD/YYYY	HH:MM:SS
Clock Epoch:	1382(358)	417600	Thu-4	187	72000	07/06/2006	20:00:00
Eph Epoch:	1382(358)	417600	Thu-4	187	72000	07/06/2006	20:00:00
Transmit Week:	1382						
Fit interval flag :	0						

SUBFRAME OVERHEAD

	SOW	DOW:HH:MM:SS	IOD	ALERT	A-S
SF1 HOW:	411426	Thu-4:18:17:06	0x17D	0	on
SF2 HOW:	411432	Thu-4:18:17:12	0x7D	0	on
SF3 HOW:	411438	Thu-4:18:17:18	0x7D	0	on

CLOCK

. . .

2.15.4 Notes

Both files can either be a RINEX or a FIC file.

2.16 *ephsum*

2.16.1 Overview

ephsum summarizes contents of a RINEX navigation message or FIC file and outputs to text file. The summary contains the transmit time, time of effectivity, end of effectivity, IODC, and health as a one-line-per ephemeris summary. The number of ephemerides found per SV is also provided. The number of ephemerides per SV is also summarized at the end. The default is to summarize all SVs found. If a specific PRN ID is provided, only data for that PRN ID will be summarized.

2.16.2 Usage

			<i>ephsum</i>
Required Arguments			
Short Arg.	Long Arg.		Description
-i	-input-file=ARG		Input file name(s)
-o	-output-file=ARG		Output file name
Optional Arguments			
Short Arg.	Long Arg.		Description
-d	-debug		Increase debug level.
-v	-verbose		Increase verbosity.
-h	-help		Print help usage.
-p	-PRNID=ARG		The PRN ID of the SV to process (default is all SVs).
-x	-xmit		List in order of transmission (default is TOE).

2.16.3 Examples

```
# Output file from EphSum
# Processing input specification: anavfic06.120 - Success(FIC)
# Processing input specification: anavfic06.121 - Success(FIC)
#
#PRN: 01, # of eph: 31
#PRN !      Xmit      !      Toe/Toc      !      End of Eff      !      IODC      Health
  01 ! 1373 146640 05/01/06 121 16:44:00 ! 1373 151184 05/01/06 121 17:59:44 !...
  01 ! 1373 151200 05/01/06 121 18:00:00 ! 1373 158384 05/01/06 121 19:59:44 !...
  01 ! 1373 165570 05/01/06 121 21:59:30 ! 1373 165584 05/01/06 121 21:59:44 !...
  01 ! 1373 158400 05/01/06 121 20:00:00 ! 1373 165600 05/01/06 121 22:00:00 !...
  01 ! 1373 165600 05/01/06 121 22:00:00 ! 1373 172784 05/01/06 121 23:59:44 !...
#
    - - - PRN 2-30 omitted for brevity - - -
#
#PRN: 31, # of eph: NONE
#
#PRN: 32, # of eph: NONE
```

```
#
#Summary of Counts by PRN
# PRN      Count
# 01       31
# 02       26
# 03       27
# 04       26
# 05       26
# 06       26
# 07       26
# 08       27
# 09       27
# 10       26
# 11       26
# 12        0
# 13       26
# 14       27
# 15       28
# 16       26
# 17       27
# 18       26
# 19       26
# 20       26
# 21       26
# 22       26
# 23       26
# 24       26
# 25       26
# 26       26
# 27       27
# 28       26
# 29       26
# 30       26
# 31        0
# 32        0
```

2.17 *fic2rin*

2.17.1 Overview

This application converts navigation messages between the FIC format, a format for GPS observations established by ARL:UT, and the RINEX format.

2.17.2 Usage

```
fic2rin usage: fic2rin <input FIC file> <output RINEX file name>
```

2.17.3 Examples

File Snippets

Binary FIC File

```
0000000
*
0000020          B   L   K           m \0 \0 \0
0000030 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 f 005 \0 \0
0000040 022 \0 \0 \0 > f 301 " 260 i { ! f \0 d 026
0000050 335 344 8 \t 002 b C 035 205 7 4 027 241 372 210 006
0000060 006 } Y / 301 374 ? \0 \ S 021 8 > f 301 "
. . .
```

RINEX NAV File

2.10	NAVIGATION	RINEX VERSION / TYPE
fic2rin	07/13/2006 11:48:58	PGM / RUN BY / DATE
END OF HEADER		
5 06 7 6 19 59 44.0	.199091155082D-03 .356976670446D-10 .000000000000D+00	
	.118000000000D+03 -.656250000000D+00 .538879589355D-08 .997594152841D+00	
	-.409781932831D-07 .710751442239D-02 .655464828014D-05 .515355578804D+04	
	.417584000000D+06 -.104308128357D-06 -.249936238139D+01 .707805156708D-07	
	.938194464982D+00 .241750000000D+03 .105751234129D+01 -.843570852398D-08	
	.600024993449D-10 .100000000000D+01 .138200000000D+04 .000000000000D+00	
	.240000000000D+01 .000000000000D+00 -.419095158577D-08 .118000000000D+03	
	.411426000000D+06 .400000000000D+01	
. . .		

2.18 *ficacheck* *ficcheck*

2.18.1 Overview

These applications read input ASCII or binary FIC and check them for errors. *ficcheck* checks binary files and *ficacheck* checks ASCII files.

2.18.2 Usage

Optional Arguments

Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-t	-time=TIME	Time of first record to count (default BOT).
-e	-end-time=TIME	End of time range to compare (default EOT).

ficacheck usage: *ficacheck* [options] <FICA file>

ficcheck usage: *ficcheck* [options] <FIC file>

2.18.3 Examples

```
>ficcheck fic06.187
```

```
Checking fic06.187
```

```
Read 252 records.
```

```
> ficacheck brokenfica
```

```
Checking brokenfica
```

```
text 0:Bad block header, record=2 location=484
```

```
text 1:blkHdr=[   ]
```

```
text 2:In record 2
```

```
text 3:In file brokenfica
```

```
text 4:Near file line 10
```

```
location 0:src/FICData.cpp:963
```

```
location 1:src/FFStream.cpp:159
```

```
location 2:src/FFStream.hpp:208
```

```
location 3:src/FFStream.hpp:208
```

2.18.4 Notes

Only the first error in each file is reported. The entire file is always checked regardless of time options.

2.19 *ficafic ficfica*

2.19.1 Overview

These applications convert navigation message data between variations of the FIC format, a format for GPS observations established by ARL:UT. *ficacheck* works with ASCII FIC files and *ficcheck* works with binary FIC files.

2.19.2 Usage

```
ficafic usage: ficafic <input fica file> <output fic file name>
ficfica usage: ficfica <input fic file> <output fica file name>
```

2.19.3 Examples

File Snippets

Binary FIC File

```
00000000
*
0000020          B  L  K          m \0 \0 \0
0000030 \0 \0 \0 \0          \0 \0 \0 \0 \0 \0 \0 f 005 \0 \0
0000040 022 \0 \0 \0 > f 301 " 260 i { ! f \0 d 026
0000050 335 344 8 \t 002 b C 035 205 7 4 027 241 372 210 006
0000060 006 } Y / 301 374 ? \0 \ S 021 8 > f 301 "
. . .
```

ASCII FIC File

```
BLK      109      0      32      0
        1382          18  583099966  561736112  375652454  154723549
        490955266  389298053  109640353  794393862  4193473  940659548
        583099966  561744492  792779231  218793822  800301952  12009725
        793943984  14182503  56922219  427630416  583099966  561753060
        1073203199  309077037  1329639  15188054  182084772  733918588
        1072216082  792738524
BLK      9      60      0      0
.13900000000000D+03 .35800000000000D+03 .41142600000000D+06 .10000000000000D+01
.10000000000000D+01 .13820000000000D+04 .10000000000000D+01 .00000000000000D+00
.00000000000000D+00 .91136000000000D+06 .00000000000000D+00-.10244548320770D-07
.41760000000000D+06 .00000000000000D+00-.14779288903810D-11-.24207541719079D-03
.00000000000000D+00 .00000000000000D+00 .00000000000000D+00 .18000000000000D+02
. . .
```

2.20 *ficdiff*

2.20.1 Overview

The application compares the contents of two FIC files containing ephemeris data.

2.20.2 Usage

Optional Arguments		<i>ficdiff</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-t	-time=TIME	Start of time range to compare (default BOT).
-e	-end-time=TIME	End of time range to compare (default EOT).

ephdiff usage: ficdiff [options] fic1 fic2

2.20.3 Examples

```
> ficdiff -t "08/01/2006 12:00:00" fic1 fic2
<FIC BlockNumber: 9
floats: 139 362 172806 1 1 1386 1 0 0 55296 0 -4.19095e-09 180000 0 . . .
integers:
chars:

<FIC BlockNumber: 9
floats: 139 362 172806 1 1 1386 1 0 0 59392 0 -6.98492e-09 179984 0 . . .
integers:
chars:
. . .
```

2.21 *findMoreThan12*

2.21.1 Overview

This application finds when there are simultaneously more than 12 SVs above a given elevation.

2.21.2 Usage

Required Arguments			<i>findMoreThan12</i>
Short Arg.	Long Arg.	Description	
-e	-eph-files=ARG	Ephemeris source file(s). Can be RINEX nav, SP3, or FIC.	
-p	-position=ARG	Antenna position in ECEF (x,y,z) coordinates. Format as a string: "X Y Z".	
-m	-min-elev=NUM	Give an integer for the elevation (degrees) above which you want to find more than 12 SVs at a given time.	
Optional Arguments			
Short Arg.	Long Arg.	Description	
-h	-help	Print help usage.	
-v	-verbose	Increase verbosity.	

2.22 *IonoBias*

2.22.1 Overview

The application will open and read several preprocessed RINEX observation files (containing obs types EL,LA,LO,SR or SS) and use the data to estimate satellite and receiver biases and to compute a simple ionospheric model using least squares and the slant TEC values.

2.22.2 Usage

<i>IonoBias</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
	-input	Input RINEX obs file name(s).
Optional Arguments		
Short Arg.	Long Arg.	Description
-f		File containing more options
	-inputdir	Path for input file(s).
Ephemeris Input		
Short Arg.	Long Arg.	Description
	-navdir	Path of navigation file(s).
	-nav	Navigation (RINEX (nav) OR SP3) file(s).
Output		
Short Arg.	Long Arg.	Description
	-datafile	Data (AT) file name, for output and/or input.
	-log	Output log file name.
	-biasout	Output satellite+receiver biases file name.
Time Limits		
Short Arg.	Long Arg.	Description
	-BeginTime	Start time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	-BeginGPSTime	Start time, arg is of the form GPSweek,GPSsow.
	-EndTime	End time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	-EndGPSTime	End time, arg is of the form GPSweek,GPSsow.
Processing		
Short Arg.	Long Arg.	Description
	-NoEstimation	Do NOT perform the estimation (default=false).
	-NoPreprocess	Skip preprocessing; read (existing) AT file (false).
	-NoSatBiases	Compute Receiver biases ONLY (not Rx+Sat biases) (false).
	-Model	Ionospheric model: type is linear, quadratic or cubic.
	-MinPoints	Minimum points per satellite required.
	-MinTimeSpan	Minimum timespan per satellite required (minutes).

-MinElevation	Minimum elevation angle (degrees).
-MinLatitude	Minimum latitude (degrees).
-MaxLatitude	Maximum latitude (degrees).
-MinLongitude	Minimum longitude (degrees).
-MaxLongitude	Maximum longitude (degrees).
-TimeSector	Time sector (day — night — both).
-TerminOffset	Terminator offset (minutes).
-IonoHeight	Ionosphere height (km).

Other Options

Short Arg.	Long Arg.	Description
	-XSat	Exclude this satellite (<sat> may be <system> only).
-v	-verbose	Print extended output info.
-d	-debug	Increase debug level.
-h	-help	Print syntax and quit.

2.22.3 Examples

```
> IonoBias --inputdir data_set --navdir data_set --input s081213a.99o --input s081214a.99o
--input s081215a.99o --nav s081213a.99n --nav s081214a.99n --nav s081215a.99n --datafile output}
Ionobias, built on the GPSTK ToolKit, Ver 1.0 6/25/04, Run 2006/08/17 09:50:59
Ionobias output directed to log file IonoBias.log
Ionobias timing: 6.210 seconds.
```

Output File Snippet

```
      3      3 Number (max, good) stations in this file
010101101100001111110111011101110
010101101100001111110111011101110
010100101100001111110111011101110
Npt 9737 Sta 85408 LLH 30.2160 262.2746 163.4226
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 1 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 14 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 15 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 21 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 22 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 25 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 29 1
1021 0.0 0.00000 -463513.64930 0.32 0.000 1 30 1
1021 30.0 0.00000 -463513.52430 0.32 0.000 1 1 1
1021 30.0 0.00000 -463513.52430 0.32 0.000 1 14 1
```

2.22.4 Notes

Input can be either on the command line or put in a file and then input using the -f option. The file is formatted just as if it were the command line.

2.23 *mdp2fic mdp2rinex*

2.23.1 Overview

The applications convert a variety of GPS related observations from the MDP format to FIC and RINEX formats. MDP is a format for network receiver interfaces derived by ARL:UT that can be used to serve observations over networks.

2.23.2 Usage

<i>mdp2fic</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
-i	-mdp-input=ARG	Filename to read MDP data from. The filename of '-' means to use stdin.
-n	-nav=ARG	Filename to which FIC nav data will be written.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-l	-log=ARG	Filename for (optional) output log file.

<i>mdp2rinex</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
-i	-mdp-input=ARG	Filename to read MDP data from. The filename of '-' means to use stdin.
-n	-obs=ARG	Filename to write RINEX obs data to. The filename of '-' means to use stdout.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-n	-nav=ARG	Filename to write RINEX nav data.
-p	-pos=ARG	Antenna position to write into obs file header. Format as string: "X Y Z"
-t	-thinning=ARG	A thinning factor for the data, specified in seconds between points.
-c	-l2c	Enable output of L2C data in C2.
-a	-any-nav-source	Accept subframes from any code/carrier.

2.23.3 Examples

```
> mdp2fic -i mdp183.06 -o fic183.06 -l mdp2ficlog183.06
```

```
> mdp2rinex -i mdp183.06 -o rin183.06o -n rin183.06n -t 60
```

2.24 *mdptool*

2.24.1 Overview

The application performs various functions on a stream of MDP data.

2.24.2 Usage

Optional Arguments		<i>mdptool</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-i	-input=ARG	Where to get the MDP data from. The default is to use stdin. If the file name begins with "tcp:" the remainder is assumed to be a hostname[:port] and the source is taken from a tcp socket at this address. If the port number is not specified a default of 8910 is used.
	-output=ARG	Where to send the output. The default is stdout.
-p	-pvt	Enable pvt output.
-o	-obs	Enable obs output.
-n	-nav	Enable nav output.
-t	-test	Enable selftest output.
-x	-hex	Dump all messages in hex.
-b	-bad	Try to process bad messages also.
-a	-almanac	Build and process almanacs. Only applies to the nav style.
-e	-ephemeris	Build and process engineering ephemerides. Only applies to the nav style.
	-min-alm	This allows a complete almanac to be constructed from fewer than 50 pages. It is required for Ashtech Z(Y)-12. The default is to require all 50 pages.
-f	-follow	Follow the input file as it grows.
-s	-output-style=ARG	What type of output to produce from the MDP stream. Valid styles are: brief, verbose, table, track, null, mdp, nav, and summary. The default is summary. Some modes aren't quite complete.
-l	-timeSpan=NUM	How much data to process, in seconds.
-m	-bug-mask=NUM	What RX bugs: 1 SV count, 2 nav parity/fmt, 4 HOW/hdr time equal.
	-startTime=TIME	Ignore data before this time. (%4Y/%03j/%02H:%02M:%05.2f).
	-stopTime=TIME	Ignore any data after this time.
	-time-format=ARG	CommonTime format specifier used for times in the output. The default is %4Y %3j %02H:%02M:%04.1f.

2.24.3 Notes

In the summary mode, the default is to only summarize the observation data above 10 degrees. Increasing the verbosity level will also summarize the data below 10 degrees.

2.25 *mergeFic*

2.25.1 Overview

This application merges multiple FIC files into a single FIC file.

2.25.2 Usage

		<i>mergeFIC</i>
Required Arguments		
Short Arg.	Long Arg.	Description
-i	-input=ARG	An input RINEX observation file, can be repeated as many times as needed.
-o	-output=ARG	Name for the merged output RINEX observation file. Any existing file with that name will be overwritten.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.

2.25.3 Examples

```
> mergeFIC -i fic1 -i fic2 -o ficm
```

2.26 *mergeRinObs mergeRinNav mergeRinMet*

2.26.1 Overview

These applications merge multiple RINEX observation, navigation, or meteorological data files into a single coherent RINEX obs/nav/met file, respectively.

2.26.2 Usage

		<i>mergeRinObs</i>
Required Arguments		
Short Arg.	Long Arg.	Description
-i	-input=ARG	An input RINEX observation file, can be repeated as many times as needed.
-o	-output=ARG	Name for the merged output RINEX observation file. Any existing file with that name will be overwritten.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.

mergeRinNav and *mergeRinMet* have the same usage.

2.26.3 Examples

```
> mergeRinObs -i arl280.06o -i arl2810.06o -o arl280-10.06o
> mergeRinNav -i arl280.06n -i arl2810.06n -o arl280-10.06n
> mergeRinMet -i arl280.06m -i arl2810.06m -o arl280-10.06m
```

2.27 *navdmp*

2.27.1 Overview

The application prints the contents of an FIC or RINEX navigation file into a human readable file and allows filtering of the data.

2.27.2 Usage

<i>navdmp</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
-i	-input=ARG	Name of an input navigation message file.
-o	-output=ARG	Name of an output file.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-a	-all-records	Unless otherwise specified, use default values for record filtration.
-t	-time=TIME	Start time (of data) for processing.
-e	-end-time=TIME	End time (of data) for processing.
-p	-prn=NUM	PRN(s) to include.
-b	-block=NUM	FIC block number(s) to process ((9)109 (Engineering) ephemerides, (62)162 (engineering) almanacs).
-r	-RINEX	Assume input file is a RINEX navigation message file.

2.27.3 Examples

```
> navdmp -i data_set/s081213a.99n -o summary --RINEX
```

```
Current filtering options:
```

```
Start time: 01/06/1980 00:00:00
End time:   01/01/4713 00:00:00
PRNs:       using all PRNs
```

```
Choose an option by number then push enter:
```

- 1) Change the start time
- 2) Change the end time
- 3) Select specific PRNs
- 5) Process the file

```
use ctrl-c to exit
```

```
? 5
```

```
processing...
```

```
Summary File Snippet
```

```
*****
```

Broadcast Ephemeris (Engineering Units)

PRN : 14

	Week(10bt)	SOW	DOW	UTD	SOD	MM/DD/YYYY	HH:MM:SS
Clock Epoch:	1021(1021)	7200	Sun-0	213	7200	08/01/1999	02:00:00
Eph Epoch:	1021(1021)	7200	Sun-0	213	7200	08/01/1999	02:00:00
Transmit Week:1021							
Fit interval flag : 0							

SUBFRAME OVERHEAD

	SOW	DOW:HH:MM:SS	IOD	ALERT	A-S
SF1 HOW:	6	Sun-0:00:00:06	0x023	0	off
SF2 HOW:	6	Sun-0:00:00:06	0x23	0	off
SF3 HOW:	6	Sun-0:00:00:06	0x23	0	off

CLOCK

Bias T0: 2.82567926E-05 sec
 Drift: 1.02318154E-12 sec/sec
 Drift rate: 0.00000000E+00 sec/(sec**2)
 Group delay: -2.32830644E-09 sec

ORBIT PARAMETERS

Semi-major axis: 5.15359685E+03 m**5
 Motion correction: 4.44732811E-09 rad/sec
 Eccentricity: 8.10711295E-04
 Arg of perigee: 2.16661714E+00 rad
 Mean anomaly at epoch: 1.75307843E-01 rad
 Right ascension: 2.02857661E+00 rad -8.31963226E-09 rad/sec
 Inclination: 9.77089255E-01 rad 2.20723480E-10 rad/sec

HARMONIC CORRECTIONS

Radial	Sine:	1.31875000E+01 m	Cosine:	3.31593750E+02 m
Inclination	Sine:	5.77419996E-08 rad	Cosine:	-1.86264515E-08 rad
In-track	Sine:	2.74367630E-06 rad	Cosine:	6.27711415E-07 rad

SV STATUS

Health bits: 0x00 URA index: 7
 Code on L2: P only L2 P Nav data: on

2.28 *NavMerge*

2.28.1 Overview

The application merges RINEX navigation files into a single file.

2.28.2 Usage

		<i>NavMerge</i>
Optional Arguments		
Short Arg.	Long Arg.	Description
-o		Write all data to an output RINEX nav file. If omitted, a data summary is written to the screen.
-tb		Output only if epoch is within 4 hours of the interval (tb,te).
-te		If te or tb is missing, they are made equal. Time tags have the form year,mon,day,HH,min,sec OR GPSweek,sow.

NavMerge usage: NavMerge [options] <RINEX nav file> <RINEX nav file>

2.28.3 Examples

```
> NavMerge -o s081213-214.99n s081213a.99n s081214a.99n
```

```
Output file name is
Exception: text 0:Unexpected EOF
text 1:In record 0
text 2:In file s081213-214.99n
text 3:Near file line 0
location 0:src/FFTextStream.hpp:244
location 1:src/FFStream.cpp:159
location 2:src/FFStream.hpp:208
location 3:src/FFStream.hpp:208
```

```
Read    0 ephemerides from file s081213-214.99n
Read  200 ephemerides from file s081213a.99n
Read   197 ephemerides from file s081214a.99n
Read   397 total ephemerides.
```

2.28.4 Notes

NavMerge corrects data for output when the GPS full week number is inconsistent with the epoch time.

2.29 *navsum*

2.29.1 Overview

This application lists the block contents of a FIC file and prints summary count information.

2.29.2 Usage

Required Arguments		<i>navsum</i>
Short Arg.	Long Arg.	Description
-i	-input=ARG	Name of an input FIC file.
-o	-output=ARG	Name of an output file.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-a	-all-records	Unless otherwise specified, use default values for record filtration.
-t	-time=TIME	Start time (of data) for processing.
-e	-end-time=TIME	End time (of data) for processing.
-p	-prn=NUM	PRN(s) to include.
-b	-block=NUM	FIC block number(s) to process ((9)109 (Engineering) ephemerides, (62)162 (engineering) almanacs).
-f	-use-alternate-format	Use alternate output format.

2.29.3 Examples

```
> navsum -i s081213a.99n -o summary --RINEX
```

```
Current filtering options:
```

```
Start time: 01/06/1980 00:00:00
End time:   01/01/4713 00:00:00
PRNs:       using all PRNs
```

```
Choose an option by number then push enter:
```

- 1) Change the start time
- 2) Change the end time
- 3) Select specific PRNs
- 5) Process the file

```
use ctrl-c to exit
```

```
? 5
```

```
processing...
```

```
Summary of data processed
```

```
Block Type Summary
```

```
Type # Blocks Found
9      0
```

109	0
62	0
162	0

Ephemeris Blocks by PRN

PRN	Block	Num
01	9	0
01	109	0
02	9	0
02	109	0
03	9	0
03	109	0
04	9	0
04	109	0
05	9	0
. . .		

2.30 *novaRinex*

2.30.1 Overview

The application will open and read a binary Novatel file (OEM2 and OEM4 receivers are supported), and convert the data to RINEX format observation and navigation files. The RINEX header is filled using user input (see below), and optional records are filled.

2.30.2 Usage

novaRinex

Required Arguments

Short Arg.	Long Arg.	Description
	-input	Novatel binary input file.

Optional Arguments

Short Arg.	Long Arg.	Description
-f		Name of file containing more options ('#' to EOL : comment).
	-dir	Directory in which to find input file (default ./).
	-obs	RINEX observation output file (RnovaRINEX.obs).
	-nav	RINEX navigation output file (RnovaRINEX.nav).

Output RINEX Header Fields

Short Arg.	Long Arg.	Description
	-noHDopt	If present, do not fill optional records in the output RINEX header.
	-HDp	Set output RINEX header 'program' field ('novaRINEX v2.1 9/07').
-HDr		Set output RINEX header 'run by' field ('ARL:UT/GPSTk').
-HDo <obser>		Set output RINEX header 'observer' field.
-HDa <agency>		Set output RINEX header 'agency' field ('ARL:UT/GPSTk').
-HDm <marker>		Set output RINEX header 'marker' field.
-HDn <number>		Set output RINEX header 'number' field.
-HDrn <number>		Set output RINEX header 'Rx number' field.
-HDrt <type>		Set output RINEX header 'Rx type' field ('Novatel').
-HDrv <vers>		Set output RINEX header 'Rx version' field ('OEM2/4').
-HDan <number>		Set output RINEX header 'antenna number' field.
-HDat <type>		Set output RINEX header 'antenna type' field.
-HDc <comment>		Add comment to output RINEX header (>1 allowed).

Output RINEX Observation Data

Short Arg.	Long Arg.	Description
	-obstype <OT>	Output this RINEX (standard) obs type (i.e. <OT> is one of L1,L2,C1,P1,P2,D1,D2,S1,or S2); repeat for each type. NB default is ALL std. types that have data.

Output Configuration

Short Arg.	Long Arg.	Description
	<code>-begin <arg></code>	Start time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	<code>-beginGPS <arg></code>	Start time, arg is of the form GPSweek,GPSsow.
	<code>-end <arg></code>	End time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	<code>-endGPS <arg></code>	End time, arg is of the form GPSweek,GPSsow
	<code>-week <week></code>	GPS Week number of this data, NB: this is for OEM2; this command serves two functions, resolving the ambiguity in the 10-bit week (default uses <code>-begin</code> , <code>-end</code> , or the current system time) and ensuring that ephemeris records that precede any obs records are not lost.
	<code>-debias</code>	Remove an initial bias from the phase.
<code>-h</code>	<code>-help</code>	Print this message and quit.
	<code>-verbose</code>	Print more information.
<code>-d</code>	<code>-debug</code>	Print extended output info.

2.30.3 Notes

Input is on the command line, or of the same format in a file (`-f<file>`).

2.31 *poscvt*

2.31.1 Overview

This application allows the user to convert among different coordinate systems on the command line. Coordinate systems handled include Cartesian, geocentric, and geodetic.

2.31.2 Usage

Optional Arguments		<i>poscvt</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
	-ecef=POSITION	ECEF “X Y Z” in meters.
	-geodetic=POSITION	Geodetic “lat lon alt” in deg, deg, meters.
	-geocentric=POSITION	Geocentric “lat lon radius” in deg, deg, meters.
	-spherical=POSITION	Spherical “theta, pi, radius” in deg, deg, meters.
-l	-list-formats	List the available format codes for use by the input and output format options.
-F	-output-format=ARG	Write the position with the given format.

2.31.3 Examples

```
> poscvt --ecef="4345070.59253 45619878.26297 803.598856837"
```

ECEF (x,y,z) in meters	4345070.5925 45619878.2630 803.5989
Geodetic (llh) in deg, deg, m	0.00100566 84.55926933 39448197.4795
Geocentric (llr) in deg, deg, m	0.00100472 84.55926933 45826334.4795
Spherical (tpr) in deg, deg, m	89.99899528 84.55926933 45826334.4795

2.31.4 Notes

If no options are given *poscvt* assumes XYZ 0 0 0.

2.32 *PR*Solve

2.32.1 Overview

The application reads one or more RINEX observation files, plus one or more navigation (ephemeris) files, and computes an autonomous pseudorange position solution, using a RAIM-like algorithm to eliminate outliers. Output is to the log file, and also optionally to a RINEX observation file with the position solutions in auxiliary header blocks.

2.32.2 Usage

<i>PR</i> Solve		
Required Arguments		
Short Arg.	Long Arg.	Description
-o	-obs	Input RINEX observation file(s).
-n	-nav	Input navigation (ephemeris) file(s) (RINEX or SP3).
Optional Arguments: Input		
Short Arg.	Long Arg.	Description
-f		File containing more options.
	-obsdir	Directory of input observation file(s).
	-navdir	Directory of input navigation file(s).
	-metdir	Directory of input meteorological file(s).
-m	-met	Input RINEX meteorological file(s).
	-decimate	Decimate data to time interval dt.
	-BeginTime	Start time: arg is 'GPSweek,sow' OR 'YYYY,MM,DD,HH,Min,Sec'.
	-EndTime	End time: arg is 'GPSweek,sow' OR 'YYYY,MM,DD,HH,Min,Sec'.
	-useCA	Use C/A code pseudorange if P1 is not available.
	-forceCA	Use C/A code pseudorange regardless of P1 availability.
Optional Arguments: Configuration		
Short Arg.	Long Arg.	Description
	-Freq	Frequency to process: 1, 2, or 3 for L1, L2, of iono-free combination.
	-MinElev	Minimum elevation angle in degrees (only if -PosXYZ).
	-exSat	Exclude this satellite.
	-Trop	Trop model, one of ZR, BL, SA, NB, NL, GG, GGH (gpstk::TropModel), with optional weather T(c), P(mb),RH(%).
Optional Arguments: PRSolution Configuration		
Short Arg.	Long Arg.	Description
	-RMSlimit	Upper limit on RMS post-fit residuals (m) for a good solution.
	-SlopeLimit	Upper limit on RAIM 'slope' for a good solution.
	-Algebra	Use algebraic algorithm (otherwise linearized LS).
	-DistanceCriterion	Use distance from a priori as convergence criterion (else RMS).

-ReturnAtOnce	Return as soon as a good solution is found.
-NReject	Maximum number of satellites to reject.
-NIter	Maximum iteration count (linearized LS algorithm).
-Conv	Minimum convergence criterion (m) (LLS algorithm).

Optional Arguments: Output

Short Arg.	Long Arg.	Description
	-Log	Output log file name (prs.log).
	-PosXYZ <X,Y,Z>	Known position (ECEF,m), used to compute output residuals.
	-APSout	Output autonomous pseudorange solution (APS - no RAIM).
	-TimeFormat	Output time format (ala CommonTime) (default: %4F %10.3g).

Optional Arguments: RINEX Output

Short Arg.	Long Arg.	Description
	-outRinex	Output RINEX observation file name.
	-RunBy	Output RINEX header 'RUN BY' string.
	-Observer	Output RINEX header 'OBSERVER' string.
	-Agency	Output RINEX header 'AGENCY' string.
	-Marker	Output RINEX header 'MARKER' string.
	-Number	Output RINEX header 'NUMBER' string.

Optional Arguments: Help

Short Arg.	Long Arg.	Description
	-verbose	Print extended output.
	-debug	Print very extended output.
	-helpRetCodes	Print return codes (implies -help).
-h	-help	Print syntax and quit.

2.32.3 Examples

```
> PRSolve -o arl2800.06o -n arl2800.06n
```

```
PRSolve, part of the GPS ToolKit, Ver 2.3 11/09, Run 2011/07/22 11:39:15
Opened log file prs.log
```

```
Weighted average RAIM solution for file: arl2800.06o
(2880 total epochs, with 2880 good, 0 rejected.)
 918129.266960 -4346070.850055 4561977.615781
Covariance of RAIM solution for file: arl2800.06o
 0.000150      -0.000061      0.000058
-0.000061      0.000427     -0.000248
 0.000058     -0.000248      0.000493
```

2.32.4 Notes

In the log file, results appear one epoch per line with the format:

TAG Nrej week sow Nsat X Y Z T RMS slope nit conv sat sat .. (code) [N]V
TAG denotes solution (X Y Z T) type:

- RPF Final RAIM ECEF XYZ solution
- RPR Final RAIM ECEF XYZ solution residuals [only if `-PosXYZ` given]
- RNE Final RAIM North-East-Up solution residuals [only if `-PosXYZ`]
- APS Autonomous ECEF XYZ solution [only if `-APSout` given]
- APR Autonomous ECEF XYZ solution residuals [only if both `-APS` `-Pos`]
- ANE Autonomous North-East-Up solution residuals [only if `-APS` `-Pos`]

Where:

- `Nrej` = number of rejected sats
- `(week,sow)` = GPS time tag
- `Nsat` = sats used
- `XYZT` = position+time solution(or residuals)
- `RMS` = RMS residual of fit
- `slope` = RAIM slope
- `nit` = of iterations
- `conv` = convergence factor
- `'sat sat ...'` lists all sat. PRNs (- : rejected)
- `code` = return value from `PRSolution::RAIMCompute()`
- `NV` means NOT valid

2.33 *ResCor*

2.33.1 Overview

The application will open and read a single RINEX observation file, apply editing commands using the RinexEditor package, compute any of several residuals and corrections and register extended RINEX observation types for them, and then write the edited data, along with the new extended observation types, to an output RINEX observation file.

2.33.2 Usage

		<i>ResCor</i>
Required Arguments		
Short Arg.	Long Arg.	Description
-IF		Input RINEX observation file.
-OF		Name of output RINEX observation file.
Configuration Arguments		
Short Arg.	Long Arg.	Description
-f<file>		File containing more options.
	-nav <file>	Navigation (RINEX Nav OR SP3) file(s).
	-navdir <dir>	Directory of navigation file(s).
Reference Position Input		
Short Arg.	Long Arg.	Description
	-RxLLH <l,l,h>	1.Receiver position (static) in geodetic lat, lon(E), ht (deg,deg,m).
	-RxXYZ <x,y,z>	2.Receiver position (static) in ECEF coordinates (m).
	-Rxhere	3.Reference site positions(time) from this file (i.e. -IF<RINEXFile>).
	-RxRinex <fn>	4.Reference site positions(time) from another RINEX file named <fn>.
	-RxFlat <fn>	5.Reference site positions and times given in a flat file named <fn>.
	-Rxhelp	(Enter -Rxhelp for a description of the -RxFlat file format).
	-RAIM	6.Reference site positions computed via RAIM (requires P1,P2,EP).
		NB the following two options apply only if -RAIM is found.
	-noRAIMedit	Do not edit data based on RAIM solution.
	-RAIMhead	Output average RAIM solution to RINEX header (if -HDf also appears).
	-noRefout	Do not output reference solution to RINEX.
	-MinElev	Minimum satellite elevation in degrees for output.
Residual/Correction Computation		
Short Arg.	Long Arg.	Description
	-debias <OT,l>	Debias new output type <OT>; trigger a bias reset with limit <l>.
	-Callow	Allow C1 to replace P1 when P1 is not available.

-Cforce	Force C/A code pseudorange C1 to replace P1.
-IonoHt <ht>	Height of ionosphere in km (default 400) (needed for LA,LO,VR,VP).
-Tgd	Apply the Tgd from BC ephemeris to SR,SP,VR, and VP.
-SVonly <prn>	Process this satellite ONLY.

Output Files

Short Arg.	Long Arg.	Description
-Log <file>		Output log file name (rc.log)

Help

Short Arg.	Long Arg.	Description
	-verbose	Print extended output
	-debug	Print debugging information.
-h	-help	Print syntax and quit.
	-REChelp	Print syntax of RINEXEditor commands and quit.
	-ROThelp	Print list of extended RINEX observation types and quit.

List of Available RINEX Observation Types

OT	Description	Units	Required input (EP=ephemeris,PS=Rx Position)	
ER	Ephemeris range	meters		EP PS
RI	Iono Delay, Range	meters		P1
PI	Iono Delay, Phase	meters	L1 L2	
TR	Tropospheric Delay	meters		EP PS
RL	Relativity Correct.	meters		EP
SC	SV Clock Bias	meters		EP
EL	Elevation Angle	degrees		EP PS
AZ	Azimuth Angle	degrees		EP PS
SR	Slant TEC (PR)	TECU		P1
SP	Slant TEC (Ph)	TECU	L1 L2	
VR	Vertical TEC (PR)	TECU		P1 EP PS
VP	Vertical TEC (Ph)	TECU	L1 L2	EP PS
LA	Lat Iono Intercept	degrees		EP PS
L0	Lon Iono Intercept	degrees		EP PS
P3	TFC(IF) Pseudorange	meters		P1
L3	TFC(IF) Phase	meters	L1 L2	
P4	GeoFree Pseudorange	meters		P1
L4	GeoFree Phase	meters	L1 L2	
P5	WideLane Pseudorange	meters		P1
L5	WideLane Phase	meters	L1 L2	
MP	Multipath (=M3)	meters	L1 L2	P1
M1	L1 Range minus Phase	meters	L1	P1
M2	L2 Range minus Phase	meters	L2	
M3	IF Range minus Phase	meters	L1 L2	P1
M4	GF Range minus Phase	meters	L1 L2	P1

M5 WL Range minus Phase	meters	L1 L2	P1
XR Non-dispersive Range	meters	L1 L2	P1
XI Ionospheric delay	meters	L1 L2	P1
X1 Range Error L1	meters	L1 L2	P1
X2 Range Error L2	meters	L1 L2	P1
SX Satellite ECEF-X	meters		EP
SY Satellite ECEF-Y	meters		EP
SZ Satellite ECEF-Z	meters		EP

2.34 *reszilla*

2.34.1 Overview

Reszilla is an application that computes various residuals from GPS pseudorange, phase, and doppler data. These data are often referred to as raw observations. The two types of residuals that are currently computed are an Observed Range Deviation (ORD), and a double difference (DD). Once these residuals are computed, statistical summaries of these differences are computed and output to the user. Optionally, the residuals themselves may be output.

2.34.2 Observed Range Deviations

An ORD is basically the observed range to an SV differenced from the estimated range to that SV. There are many terms that go into computing the estimated range and/or correcting the observed range for known effects. When all of these effects are accounted for (as *reszilla* is capable of doing) ORDs can be in the 10-30 cm range for a geodetic quality GPS receiver. Pretty impressive when you consider that the range to the SV is somewhere between 20 to 26 million meters.

For many GPS receivers, the most significant effect to account for is the receiver clock offset. This is the difference between the receiver's internal time and true GPS time. This parameter is often computed as part of a PVT solution. This is not how *reszilla* works. *Reszilla* is provided a surveyed position of the receiver antenna, and it makes a more accurate estimate of the receiver clock offset by averaging the residuals of all SVs in track.

2.34.3 Usage

OrdApp

Required Arguments		
Short Arg.	Long Arg.	Description
-i	-input	Where to read the ord data. The default is stdin.
-r	-output	Where to write the output. The default is stdout.
-t	-time-format	CommonTime format specifier used for times in the output.
Optional Arguments		
Short Arg.	Long Arg.	Description
	-ns	Report the clock in ns, not meters.

ordClock

ordClock generates clock estimates for each epoch of ORDs.

Optional Arguments		<i>ordClock</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-w	-use-warts	Use warts in the clock solution. The default is to not use warts.
-e	-estimate-only	Only compute the receiver clock bias. Don't remove this bias from the ords. The default is to both estimate the bias and remove the it from the ords.
-c	-clock-source=ARG	An ord file to read the receiver clock offsets from.
-i	-input=ARG	Where to read the ord data. The default is stdin.
-r	-output=ARG	Where to write the output. The default is stdout.
-t	-time-format=ARG	CommonTime format specifier used for times in the output. The default is "%4Y %3j %02H:%02M:%04.1f".
	-ns	Report the clock in ns, not meters.

ordEdit

ordEdit edits an ORD file based on various criteria.

Optional Arguments		<i>ordEdit</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-k	-clock-est	Remove ORDs that do not have corresponding clock estimates.
-c	-no-clock	Remove all clock offset estimate warts. Give this option twice to remove all clock data.
-m	-elev=NUM	Remove data for SVs below a given elevation mask.
-p	-PRN=NUM	Filter data by PRN number. Repeat option for multiple satellites. Negative PRN numbers mean exclude these PRNs. Positive PRN numbers mean only include these satellites. Zero removes all.
-w	-warts=NUM	Include/Exclude warts from the indicated PRN. Repeat option for multiple PRNs. Negative numbers exclude, positive numbers include, zero excludes warts from all PRNs. The default is to include all warts.

-e	-be-file=ARG	Remove data for unhealthy SVs by providing broadcast ephemeris source: RINEX nav or FIC file.
	-start=ARG	Throw out data before this time. Format as string: "yyyy ddd HH:MM:SS".
	-end=ARG	Throw out data after this time. Format as string: "yyyy ddd HH:MM:SS".
-s	-size=ARG	Remove clock residuals with absolute values greater than this size (meters).
-l	-ord-limit=ARG	Remove ords with absolute values greater than this size (meters).
-i	-input=ARG	Where to read the ord data. The default is stdin.
-r	-output=ARG	Where to write the output. The default is stdout.
-t	-time-format=ARG	CommonTime format specifier used for times in the output. The default is "%4Y %3j %02H:%02M:%04.1f".
	-ns	Report the clock in ns, not meters.

ordGen

ordGen generates observed range deviations.

Required Arguments		<i>ordGen</i>
Short Arg.	Long Arg.	Description
-o	-obs=ARG	Where to get the obs data.
-e	-eph=ARG	Where to get the ephemeris data. Acceptable formats include RINEX (nav), FIC, MDP, SP3, YUMA, and SEM.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-w	-weather=ARG	Weather data file name (RINEX met format only).
-c	-msc=ARG	Station coordinate file.
	-omode=ARG	Specifies what observations are used to compute the ORDs. Valid values are:p1p2, z1z2, c1p2, c1c2, c1y2, c1z2, y1y2, c1, p1, y1, z1, c2, p2, y2, z2, smo, dynamic, and smart. The default is smart.
	-trop-model=ARG	Specify the trop model to use. Options are zero, simple, nb, and gg. The default is nb.
-p	-pos=ARG	Location of the antenna in meters ECEF.
-m	-msid=NUM	Station to process data for. Used to select a station position from the msc file or data from a SMODF file.
-n	-near	Allows the program to select an ephemeris that is not strictly in the future. Only affects the selection of which broadcast ephemeris to use.
	-sv-time	Assume that the data is time-tagged according to each SV's clock, not a common receiver clock. The is set by default only for omode=smo.

-i	-input=ARG	Where to read the ord data. The default is stdin.
-r	-output=ARG	Where to write the output. The default is stdout.
-t	-time-format=ARG	CommonTime format specifier used for times in the output. The default is "%4Y %3j %02H:%02M:%04.1P".
	-ns	Report the clock in ns, not meters.

ordLinEst

ordLinEst computes a linear clock estimate.

Optional Arguments		<i>ordLinEst</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-m	-max-rate=ARG	Rate used to detect a clock jump. Default is 10,000 m/day.
-i	-input=ARG	Where to read the ord data. The default is stdin.
-r	-output=ARG	Where to write the output. The default is stdout.
-t	-time-format=ARG	CommonTime format specifier used for times in the output. The default is "%4Y %3j %02H:%02M:%04.1P".
	-ns	Report the clock in ns, not meters.

ordStats

ordStats computes ORD statistics.

Optional Arguments		<i>ordStats</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-b	-elev-bin=ARG	A range of elevations, used in computing the statistical summaries. Repeat to specify multiple bins. The default is "-b 0-10 -b 10-20 -b 20-60 -b 10-90".
-s	-sigma=NUM	Multiplier for sigma stripping used in statistical computations. The default value is 6.
-w	-wonky	Use wonky data in stats computation. The default is to not use such data.

	<code>-stats-only</code>	Only output stats to stdout.
<code>-i</code>	<code>-input=ARG</code>	Where to read the ord data. The default is stdin.
<code>-r</code>	<code>-output=ARG</code>	Where to write the output. The default is stdout.
<code>-t</code>	<code>-time-format=ARG</code>	CommonTime format specifier used for times in the output. The default is "%4Y %3j %02H:%02M:%04.1f".
	<code>-ns</code>	Report the clock in ns, not meters.

2.34.4 Double Difference Residuals

While many double differences exist, *reszilla* computes an the first difference to a master SV and the second difference to a second receiver. This double difference removes receiver clock error, iono, trop, and SV clock errors. When the two receivers are connected to a common antenna (often referred to as a zero-baseline setup) and are of the same type, even the multipath is differenced out. What is left is basically receiver tracking noise and receiver tracking errors.

One complicating factor in computing this DD is that while the clock errors in the receivers cancel out, there is still an error associated with the motion of the satellite during the interval between when the two receivers computing their observation. To remove this error, an estimate of the clock offset between the two receivers is need. *Reszilla* can get this estimate in one of two ways; estimates this by computing a clock estimate for each receiver as described under the ORD section or reading the estimates from the rinex obs data files. These two estimates are then differenced to get the offset between the two receivers.

Another complicating factor is that the phase observations normally have an "integer ambiguity" associated with them. When the DD phase observation is computed, it will have the difference between the two receivers ambiguity. Often this number can be quite big. Removing this ambiguity is often referred to as debiasing the data. This process involves much black magic and slight of hand. Do not delve into this or even look too closely at the details or you will be sullied.

2.34.5 Usage

ddGen

ddGen computes double-difference residuals from raw observations.

Required Arguments		
Short Arg.	Long Arg.	Description
<code>-1</code>	<code>-obs1=ARG</code>	Where to get the first receiver's obs data.
<code>-2</code>	<code>-obs2=ARG</code>	Where to get the second receiver's obs data.
<code>-e</code>	<code>-eph=ARG</code>	Where to get the ephemeris data. Acceptable formats include RINEX nav, FIC, MDP, SP3, YUMA, and SEM.

Optional Arguments

Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
	-ddmode=ARG	Specifies what observations are used to compute the double difference residuals. Valid values are: all, phase. The default is all.
	-omode=ARG	Specifies what observations to use to compute the ORDs. Valid values are: p1p2, z1z2, c1p2, c1y2, c1z2, y1y2, c1, p1, y1, z1, c2, p2, y2, z2 smo, dynamic, and smart. The default is smart.
	-min-arc-time=ARG	The minimum length of time (in seconds) that a sequence of observations must span to be considered as an arc. The default value is 60.0 seconds.
	-min-arc-gap=ARG	The minimum length of time (in seconds) between two arcs for them to be considered separate arcs. The default value is 60.0 seconds.
	-min-arc-length=ARG	The minimum number of epochs that can be considered an arc. The default value is 5 epochs.
	-noise=ARG	The noise threshold used in finding discontinuities. The default is 0.1000 cycles.
-b	-elev-bin=ARG	Range of elevations to use in computing the statistical summaries. Repeat to specify multiple bins. The default is "-b 0-10 -b 10-20 -b 20-60 -b 60-90".
-c	-msc=ARG	Station coordinate file.
-p	-pos=ARG	Location of the antenna in meters ECEF.
-E	-health-src=ARG	Do not use data from unhealthy SVs as determined using this ephemeris source. Can be RINEX navigation or FIC file(s).
	-strip=ARG	Factor used in stripping data prior to computing descriptive statistics. The default value is 3.2.
	-phase=ARG	Only compute phase double differences.
	-S	Only included observables with a raw signal strength, or SNR, of at least the
3 -m	-msid=NUM	Station to process data for. Used to select a station position from the msc file or data from a SMODF file.
-w	-window=NUM	Compute mean values of the double differences over this time span (seconds). (15 min = 900)
-r	-raw	Output the raw double differences in addition to the descriptive statistics.
-a	-all-combos	Compute all combinations, don't just use one master SV.
-n	-near	Allow the program to select an ephemeris that is not strictly in the future. Only affects the selection of which broadcast ephemeris to use. i.e. use a close ephemeris.
	-zero-trop	Disables trop corrections.

2.34.6 Data Input

Several different types of data are required to compute these residuals; the raw observations, the receiver antenna position, the satellite position, and optionally weather observations. The raw observations may be supplied to *reszilla* in one of several formats; *rinex obs* (see *RinexObsData* class), *smodf* (see *SMODFData* class), and *MDP* (see *MDPObsEpoch* class in *apps/MDPtools*). The receiver antenna position may be specified in the *rinex obs* header or via a station coordinates file (see *MSCData* class).

2.34.7 Output

There are two general types of output that *reszilla* produces - statistical summaries and the raw residuals. The mean, standard deviation, and maximum value of the residuals are calculated as a function of specified elevation ranges and are output in a statistics table. Looking at the results for each elevation bin is useful as ORDs tend to be much higher when satellites are lower on the horizon. For a more thorough analysis, the ORD or DD residuals calculated by *reszilla* may be output in a matrix format to a file with columns for time, PRN, elevation, ORD or clock residual, IODC, satellite health, and a flag for the residual type. The flag specifies exactly which of the 13 possible residual types the data on that row represent, depending on the method used for calculation.

One benefit of this output feature is that residuals can be looked at for particular time periods or PRNs. Fortunately there is a companion plotting tool that makes this simple. Given a *reszilla* output file, the *dplot* program will plot residuals and, if specified, receiver clock estimates versus time using *gnuplot*. A user may specify the time range, stripping value, and PRN(s) to use in the plot, as well as a filename for saving the result.

Types in the raw output files:

- 0 - c1p2 observed range deviation
- 50 - computed clock, difference from estimate, strip
- 51 - linear clock estimate, abdev

Double difference types:

- 10 - c1 20 - c2
- 11 - p1 21 - p2
- 12 - l1 22 - l2
- 13 - d1 23 - d2
- 14 - s1 24 - s2

2.34.8 Notes

The criteria min-arc-time and min-arc-length are both required to be met for an arc to be valid in double difference mode. All output quantities (stddev, min, max, ord, clock, double difference, ...) are in meters.

2.35 *rmwcheck rnwcheck rowcheck*

2.35.1 Overview

These applications read a RINEX observation (*rowcheck*), navigation(*rnwcheck*), or meteorological (*rmwcheck*) data file and check it for errors.

2.35.2 Usage

Optional Arguments

Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-l	-quit-on-first-error	Quit on the first error encountered.
-t	-time=TIME	Time of first record to count (Default = BOT).
-e	-end-time=TIME	End of time range to compare (Default = EOT).

rmwcheck usage: *rmwcheck* [options] <RINEX Met file>

rnwcheck usage: *rnwcheck* [options] <RINEX Nav file>

rowcheck usage: *rowcheck* [options] <RINEX Obs file>

2.35.3 Examples

```
> rnwcheck -t "08/01/2006 12:00:00" -e "08/01/2006 15:00:00" s081214a.99n
```

```
Checking s081213a.99n
```

```
Read 200 records.
```

2.35.4 Notes

Only the first error in each file is reported. The entire file is always checked regardless of time options.

2.36 *rmwdiff rnwdiff rowdiff*

2.36.1 Overview

These applications difference RINEX observation, navigation, and meteorological data files.

2.36.2 Usage

Optional Arguments

Short Arg.	Long Arg.	Description
-d	--debug	Increase debug level.
-v	--verbose	Increase verbosity.
-h	--help	Print help usage.
-l	--quit-on-first-error	Quit on the first error encountered.
-t	--time=TIME	Start of time range to compare (Default = BOT.)
-e	--end-time=TIME	End of time range to compare (Default = EOT.)

rmwdiff usage: rmwdiff [options] <RINEX Met file> <RINEX Met file>

rnwdiff usage: rnwdiff [options] <RINEX Nav file> <RINEX Nav file>

rowdiff usage: rowdiff [options] <RINEX Obs file> <RINEX Obs file>

2.36.3 Notes

Only the first error in each file is reported. The entire file is always checked regardless of time options.

2.37 *RinexDump*

2.37.1 Overview

The application reads a RINEX file and dumps the observation types in columns. Output is to the screen, with one time tag and one satellite per line.

2.37.2 Usage

Optional Arguments		<i>RinexDump</i>
Short Arg.	Long Arg.	Description
	-pos	Output only positions from aux headers; sat and obs are ignored.
-n	-num	Make output purely numeric (no header, no system char on sats).
	-format <file>	Output times in CommonTime format (Default: %4F %10.3g).
	-file <file>	RINEX observation file; this option may be repeated.
	-obs <obs>	RINEX observation type, found in file header.
	-sat <sat>	RINEX satellite ID (e.g. G31 for GPS PRN 31).
-h	-help	Print this and quit.

RinexDump usage: *RinexDump* [-n] <rinex obs file> [<satellite(s)> <obstype(s)>]

The optional argument -n tells *RinexDump* its output should be purely numeric.

2.37.3 Examples

```
> RinexDump algo1580.06o 3 4 5

# Rinexdump file: algo1580.06o Satellites: G03 G04 G05 Observations: ALL
# Week GPS_sow Sat      L1 L S      L2 L S      C1 L S
1378 259200.000 G03 -3843024.647 0 3 -2994560.443 0 1 23796436.087 0 0
1378 259230.000 G03 -3954052.735 0 3 -3081075.654 0 2 23775308.750 0 0
1378 259260.000 G03 -4064994.465 0 2 -3167523.561 0 3 23754197.617 0 0

. . .

      P2 L S      P1 L S      S1 L S      S2 L S
23796439.457 0 0 23796436.350 0 0    21.100 0 0    11.000 0 0
23775311.168 0 0 23775308.182 0 0    22.100 0 0    17.800 0 0
23754199.648 0 0 23754196.550 0 0    17.000 0 0    18.600 0 0

. . .
```

2.37.4 Notes

MATLAB and Octave can read the purely numeric output.

2.38 *Rinex3Dump*

2.38.1 Overview

The application reads a RINEX3 file and dumps the observation data for the given satellite(s) to the standard output.

2.38.2 Usage

Optional Arguments		<i>Rinex3Dump</i>
Short Arg.	Long Arg.	Description
-f	-file <file>	Input file is a RINEX observation file. This option may be repeated. Optional, but may be needed in case of ambiguity.
	-format <format>	The format of the time output. Default is %4F %10.3g.
-h	-help	Prints out this help and exits.
-n	-num	Make output purely numeric, ie. no header, no system char on satellites.
-o	-obs <obs>	RINEX observation type (eg. C1C) found in the file header. Optional, but may be needed in case of ambiguity.
-p	-pos	Only output positions from aux headers, ie. sat and obs are ignored.
-s	-sat <sat>	RINEX satellite ID (eg. For GPS PRN 31, <sat> = G01). Optional, but may be needed in case of ambiguity.
-v	-verbose	Prints out verbose output.

Rinex3Dump usage: Rinex3Dump [-n] <rinex obs file> [<satellite(s)> <obstype(s)>]

The optional argument -n tells Rinex3Dump its output should be purely numeric.

2.38.3 Notes

MATLAB and Octave can read the purely numeric output.

2.39 *rinexpvt*

2.39.1 Overview

The application generates a user position based on RINEX observation data with the option of including navigation and meteorological data to aid error correction.

2.39.2 Usage

Required Arguments			<i>rinexpvt</i>
Short Arg.	Long Arg.		Description
-o	-obs-file=ARG		RINEX observation file.
Optional Arguments			
Short Arg.	Long Arg.		Description
-d	-debug		Increase debug level.
-v	-verbose		Increase verbosity.
-h	-help		Print help usage.
-n	-nav-file=ARG		RINEX navigation file. Required for single frequency ionosphere correction.
-p	-pe-file=ARG		SP3 Precise Ephemeris file. Repeat this for each input file.
-m	-met-file=ARG		RINEX meteorological file.
-t	-time-format=ARG		Alternate time format string.
-e	-enu=ARG		Use the following as origin to solve for East/North/Up coordinates, formatted as a string: "X Y Z".
-l	-elevation-mask=ARG		Elevation mask (degrees).
-g	-logfile=ARG		Write logfile to this file.
-r	-rate=ARG		Observation interval (Default = 30 seconds or Rinex Header specification).
-y	-yuma=ARG		Yuma almanac file.
-a	-sem=ARG		SEM almanac file.
-s	-single-frequency		Use only C1 (SPS).
-f	-dual-frequency		Use only P1 and P2 (PPS).
-i	-no-ionosphere		Do NOT correct for ionosphere delay.
-x	-no-closest-ephemeris		Allow ephemeris use outside of fit interval.
-c	-no-carrier-smoothing		Do NOT use carrier phase smoothing.
-z	-no-glonass		Exclude GLONASS Satellites from PVT solution.

2.39.3 Examples

```
> rinexpvt -o arl2800.06o -n arl2800.06n
2006 1 1 09 41 00 918130.968492 -4346073.94224 4561982.02123 333.303358692
2006 1 1 09 41 30 918130.956684 -4346073.91529 4561982.01659 333.317002144
2006 1 1 09 42 00 918130.924146 -4346073.83279 4561982.01338 333.279239604
```

2.39.4 Notes

Though not stated in the required options lists, either a RINEX navigation file or an SP3 Precise Ephemeris File is needed, using the -n or -p option respectively. When using precise ephemeris, three files must be included: the previous day, the current day, and the next day.

Although -z argument appears as optional, in this release, it is always turned on, but implementation will occur in a later release.

2.40 *RinSum*

2.40.1 Overview

The application reads a RINEX file and summarizes its content.

2.40.2 Usage

Optional Arguments		<i>RinSum</i>
Short Arg.	Long Arg.	Description
-i	-input	Input file name(s).
-f		File containing more options.
-o	-output	Output file name.
-p	-path	Path for input file(s).
-R	-Replace	Replace header with full one.
-s	-sort	Sort the PRN/Obs table on begin time.
-g	-gps	Print times in the PRN/Obs table as GPS times.
	-gaps	Print a table of gaps in the data, assuming specified interval dt.
	-start	Start time: <time> is 'GPSweek,sow' OR 'YYYY,MM,DD,HH,Min,Sec'.
	-stop	Stop time: <time> is 'GPSweek,sow' OR 'YYYY,MM,DD,HH,Min,Sec'.
-b	-brief	Produce a brief (6-line) summary.
-h	-help	Print syntax and quit.
-d	-debug	Print debugging information.

2.40.3 Examples

```
> RinSum -i data_set/s081213a.99o --EpochBeg 2006,08,1,12,0,0'

+++++++ RinSum summary of Rinex obs file data_set/s081213a.99o ++++++
Rinex header:
----- REQUIRED -----
Rinex Version 2.10, File type Observation, System G (GPS).
Prgm: RinexObsWriter, Run: 11-14-01 10:04:27, By: NIMA
Marker name: 85408.
Obs'r : Monitor Station, Agency: NIMA
Rec#: 1, Type: ZY12, Vers:
Antenna # : 85408, Type : AshTech Geodetic 3
Position (XYZ,m) : (-740289.7851, -5457071.6555, 3207245.8294).
Antenna offset (ENU,m) : (0.0000, 0.0000, 0.0000).
Wavelength factors (default) L1:1, L2: 1.
Observation types (7) :
Type #0 = L1 L1 Carrier Phase (L1 cycles).
Type #1 = L2 L2 Carrier Phase (L2 cycles).
Type #2 = C1 C/A-code pseudorange (meters).
Type #3 = P1 Pcode L1 pseudorange (meters).
Type #4 = P2 Pcode L2 pseudorange (meters).
Type #5 = D1 Doppler Frequency L1 (Hz).
Type #6 = D2 Doppler Frequency L2 (Hz).
```

```

Time of first obs 1999/08/01 00:00:00.0000000 GPS
(This header is VALID 2.1 Rinex.)
----- OPTIONAL -----
Comments (3) :
The AS bit flag is set if receiver is in Z mode
Signal to Noise ratio information is omitted
This file contains SMOOTHED obs data
----- END OF HEADER -----

WARNING: Computed first time does not agree with header

Computed interval is 0.00
Computed first epoch is -4713/01/01 00:00:00.0000000
Computed last epoch is 1999/08/01 23:59:30.0000000
There were 0 epochs (-0.00% of -2147483647 possible epochs in this timespan) and 0 inline header blocks.

      Summary of data available in this file: (Totals are based on times and interval)
PRN/OT:  L1  L2  C1  P1  P2  D1  D2 Total Begin - End time
TOTAL      0   0   0   0   0   0   0

```

WARNING: ObsType L1 should be deleted from header.

WARNING: ObsType L2 should be deleted from header.

WARNING: ObsType C1 should be deleted from header.

WARNING: ObsType P1 should be deleted from header.

WARNING: ObsType P2 should be deleted from header.

WARNING: ObsType D1 should be deleted from header.

WARNING: ObsType D2 should be deleted from header.

+++++++ End of RinSum summary of data_set/s081213a.99o ++++++

2.41 *Rin3Sum*

2.41.1 Overview

The application reads a RINEX3 file and summarizes its content.

2.41.2 Usage

Optional Arguments		<i>Rin3Sum</i>
Short Arg.	Long Arg.	Description
-i	-input	Input file name(s).
-f		file containing more options.
-o	-output	Output file name.
-p	-path	Path for input file(s).
-R	-Replace	Replace header with full one.
-s	-sort	Sort the PRN/Obs table on begin time.
-g	-gps	Print times in the PRN/Obs table as GPS times.
	-EpochBeg	Start time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	-GPSEnd	End time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	-GPSBeg	Start time, arg is of the form GPSweek,GPSsow.
	-GPSEnd	End time, arg is of the form GPSweek,GPSsow.
-h	-help	Print syntax and quit.
-d	-debug	Print debugging info.

2.42 *rtAshtech*

2.42.1 Overview

This application logs observations from an Ashtech Z-XII receiver. It records observations directly into the RINEX format. A number of optional outputs are possible. The raw messages from a receiver can be recorded. Observations can also be recorded in a format that is easily imported into numerical packages.

2.42.2 Usage

Optional Arguments		<i>rtAshtech</i>
Short Arg.	Long Arg.	Description
-h	-help	Print help usage.
-v	-verbose	Increased diagnostic messages.
-r	-raw	Record raw observations.
-l	-log	Record log entries.
-t	-text	Record observations as simple text files.
-o	-rinex-obs=ARG	Naming convention for RINEX obs files.
-n	-rinex-nav=ARG	Naming convention for RINEX nav message files.
-T	-text-obs=ARG	Naming convention for obs in simple text files.
-i	-input	Where to read ashTech data. Can be a file or a serial device (ser:/dev/ttyS0), a tcp port (tcp:hostname:port), or standard input (the default).

2.42.3 Examples

```
> rtAshtech -p /dev/ttyS1
```

```
> rtAshtech -o "minute\%03j\%02H\%02M.\%02yo"
```

2.42.4 Notes

rtAshtech only works on UNIX systems with POSIX compliant serial ports.

2.43 *sp32bc*

2.43.1 Overview

This application reads an SP3 file (either a or c format) and writes to RINEX navigation file(s).

2.43.2 Usage

sp32bc

Required Arguments

Short Arg.	Long Arg.	Description
-p	-pe	Input precise ephemeris.
-r	-r	Rate of broadcast ephemeris output (seconds).

Optional Arguments

Short Arg.	Long Arg.	Description
-h	-help	Display argument list.

2.44 *sp3version*

2.44.1 Overview

This application reads an SP3 file (either a or c format) and writes it to another file (also either in a or c format).

2.44.2 Usage

Optional Arguments		<i>sp3version</i>
Short Arg.	Long Arg.	Description
	-in	A file from which to take the input. The default is stdin.
	-out	A file into which to write the output. The default is sp3.out.
	-outputC	Output version c (otherwise a).
	-msg	Add message as a comment to the output header.
	-verbose	Output to screen: dump headers, data, etc.

2.45 *svvis*

2.45.1 Overview

This application computes when satellites are visible at a given point on the earth.

2.45.2 Usage

Required Arguments		<i>svvis</i>
Short Arg.	Long Arg.	Description
-e	-eph=ARG	Where to get the ephemeris data. Can be RINEX, nav, FIC, MDP, SP3, YUMA, and SEM.
Optional Arguments		
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
	-elevation-mask=ARG	The elevation above which an SV is visible. The default is 0 degrees.
-p	-position=ARG	Receiver antenna position in ECEF (x,y,z) coordinates. Format as string: "X Y Z".
-c	-msc=ARG	Station coordinate file.
-m	-msid=ARG	Station number to use from the msc file.
	-graph-elev=ARG	Output data at the specified interval. Interval is in seconds.
-l	-time-span=ARG	How much data to process, in seconds. Default is 86400.
	-start-time=TIME	When to start computing positions. The default is the start of the ephemeris data.
	-stop-time=TIME	When to stop computing positions. The default is one day after the start time.
	-print-elev	Print the elevation of the sv at each change in tracking. The default is just to output the PRN of the SV.
	-rise-set	Print the visibility data by PRN in rise-set pairs.
	-tabular	Print the visibility data in a tabular format.
	-recent-eph	Use this if the ephemeris data provided uses 10-bit GPS weeks and it should be converted to the current epoch or to the epoch current to the "start-time", if specified.

2.46 *TECMaps*

2.46.1 Overview

Program TECMaps reads RINEX data files containing extended RINEX observation types EL, AZ and SR or VR from several sites and at each epoch fits the vertical TEC data to a model of the ionosphere on a two-dimensional grid surface. Hardware TEC measurement biases are corrected, using input from the program IonoBias. The user can specify the type of grid, the type of TEC data and the model to be used. Output is in the form of files, one per epoch, which can be used to plot the 2D ionospheric TEC surface.

2.46.2 Usage

<i>TECMaps</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
	-input	Input RINEX obs file name(s).
Optional Arguments		
Short Arg.	Long Arg.	Description
	-f	File containing more options.
Reference Station Position (One Required)		
Short Arg.	Long Arg.	Description
	-RxLLH <l,l,h>	Reference site position in geodetic lat, lon (E), ht (deg,deg,m).
	-RxXYZ <x,y,z>	Reference site position in ECEF coordinates (m).
	-inputdir	Path for input file(s).
Ephemeris Input		
Short Arg.	Long Arg.	Description
	-navdir	Path of navigation file(s).
	-nav	Navigation (RINEX navigation OR SP3) file(s).
Output		
Short Arg.	Long Arg.	Description
	-log	Output log file name.
Time Limits		
Short Arg.	Long Arg.	Description
	-BeginTime	Start time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	-BeginGPSTime	Start time, arg is of the form GPSweek,GPSsow.
	-EndTime	End time, arg is of the form YYYY,MM,DD,HH,Min,Sec.
	-EndGPSTime	End time, arg is of the form GPSweek,GPSsow.
Processing		
Short Arg.	Long Arg.	Description
	-noVTECmap	Do NOT create the VTEC map.
	-MUFmap	Create MUF map as well as VTEC map.
	-F0F2map	Create F0F2 map as well as VTEC map.

-Title1 <title>	Title information.
-Title2 <title>	Second title information.
-BaseName <name>	Base name for output files.
-DecorrError <de>	Decorrelation error rate in TECU/1000km (3).
-Biases <file>	File containing estimated sat+rx biases (Prgm IonoBias).
-ElevThresh <ele>	Minimum elevation (6 degrees).
-MinAcqTime<t>	Minimum acquisition time (0 seconds).
-FlatFit	Flat fit type (default).
-LinearFit	Linear fit type.
-IonoHeight <n>	Ionosphere height (km).
-Offset <tec>	Overall bias to add to data (TECU).

Grid

Short Arg.	Long Arg.	Description
	-UniformSpacing	Grid uniform in space (XYZ) (default).
	-UniformGrid	Grid uniform in Lat and Lon.
	-OutputGrid	Output the grid to file <basename.LL>.
	-GnuplotOutput	Write the grid file for gnuplot (default: for Matlab).
	-NumLat <n>	Number of latitude grid points (40).
	-NumLon <n>	Number of longitude grid points (40).
	-BeginLat <lat>	Beginning latitude (21 degrees).
	-BeginLon <lon>	Beginning longitude (230 degrees E).
	-DeltaLat 	Grid spacing in latitude (0.25 degrees).
	-DeltaLon 	Grid spacing in longitude (1.0 degrees).

Other Options

Short Arg.	Long Arg.	Description
	-XSat	Exclude this satellite (<sat> may be <system> only).

Help

Short Arg.	Long Arg.	Description
-v	-verbose	Print extended output info.
-d	-debug	Increase debug level.
-h	-help	Print syntax and summary of input, then quit.

2.46.3 Notes

Input is on the command line, or of the same format in a file (-f<file>).

2.47 *timeconvert*

2.47.1 Overview

This application allows the user to convert between time formats associated with GPS. Time formats include: civilian time, Julian day of year and year, GPS week and seconds of week, Z counts, and Modified Julian Date (MJD).

2.47.2 Usage

Optional Arguments		<i>timeconvert</i>
Short Arg.	Long Arg.	Description
-d	-debug	Increase debug level.
-v	-verbose	Increase verbosity.
-h	-help	Print help usage.
-A	-ansi=TIME	"ANSI-Second".
-c	-civil=TIME	"Month(numeric) DayOfMonth Year Hour:Minute:Second
-R	-rinex-file=TIME	"Year(2-digit) Month(numeric) DayOfMonth Hour Minute Second".
-o	-ews=TIME	"GPSEPOCH 10bitGPSweek SecondOfWeek".
-f	-ws=TIME	"FullGPSWeek SecondOfWeek".
-w	-wz=TIME	"FullGPSWeek Zcount".
	-z29=TIME	"29bitZcount".
-Z	-z32=TIME	"32bitZcount".
-j	-julian=TIME	"JulianDate".
-m	-mjd=TIME	"ModifiedJulianDate".
-u	-unixtime=TIME	"UnixSeconds UnixMicroseconds".
-y	-doy=TIME	"Year DayOfYear SecondsOfDay".
	-input-format=ARG	Time format to use on input.
	-input-time=ARG	Time to be parsed by "input-format" option.
-F	-format=ARG	Time format to use on output.
-a	-add-offset=NUM	Add NUM seconds to specified time.
-s	-sub-offset=NUM	Subtract NUM seconds from specified time.

2.47.3 Examples

Convert RINEX file time.

```
> timeconvert -R "05 06 1985 13:50:02"
```

Month/Day/Year H:M:S	11/06/2010 13:00:00
Modified Julian Date	55506.541666667
GPSweek DayOfWeek SecOfWeek	584 6 565200.000000
FullGPSweek Zcount	1608 376800
Year DayOfYear SecondOfDay	2010 310 46800.000000
Unix: Second Microsecond	1289048400 0
Zcount: 29-bit (32-bit)	306560992 (843431904)

Convert ews time.

```
timeconvert -o "01 1379 500"
```

Month/Day/Year	1/25/2026
Hour:Min:Sec	00:08:20
Modified Julian Date	61065.005787037
GPSTime Week DayOfWeek SecOfWeek	355 0 500.000000
FullGPSTime Week Zcount	2403 333
Year DayOfYear SecondOfDay	2026 25 500.000000
Unix_sec Unix_usec	1769299700 0
Zcount: 29-bit (32-bit)	186122573 (1259864397)

2.47.4 Notes

If no arguments are given it will convert the current time to all formats. When inputting time values, include quotation marks.

2.48 *vecsol*

2.48.1 Overview

The application computes a 3D vector solution using dual-frequency carrier phases. A double difference algorithm is applied with properly computed weights (elevation sine weighting) and correlations. The program iterates to convergence and attempts to resolve ambiguities to integer values if close enough. Crude outlier rejection is provided based on a triple-difference test. Ephemerides used are either broadcast or precise (SP3).

Alternatively, P code processing is additionally provided. The solution is computed using either the ionosphere-free linear combination, or the average of L1 and L2. The ionospheric model included in broadcast ephemeris may be used. A standard tropospheric correction is applied, or tropospheric parameters (zenith delays) may be estimated for the first station (vector mode) or both.

2.48.2 Usage

`vecsol usage: vecsol <RINEX Obs file 1> <RINEX Obs file 2>`

RINEX Observation Files

The two arguments are names of RINEX observation files. They contain the observations collected at the two end points 1 and 2 of the baseline. They must contain a sufficient set of simultaneous observations to the same satellites.

If no separate station coordinate files are provided, the initial station coordinates are taken from the RINEX headers. Upon finishing, `vecsol` creates or updates the coordinate file of the first station (vector mode) or both.

Configuration File `vecsol.conf`

The file `vecsol.conf` contains the input options for the program, one per line.

Options	Value	Meaning
<code>obsMode</code>	3/2/1/0	If 1 or 3, process carrier phase data (instead of P code data). If 0 or 1, iterate on ionosphere-free vector (not L1 + L2).
<code>truecov</code>	1/0	If 1, use true double difference covariances. If 0, ignore any possible correlations.
<code>precise</code>	1/0	If 1, use precise ephemeris, if 0, use broadcast ephemeris.
<code>iono</code>	1/0	If 1, use the 8-parameter ionospheric model that comes with the broadcast ephemeris (.nav) files.
<code>tropo</code>	1/0	If 1, estimate troposphere parameters (zenith delays relative to the standard value, which is always applied).
<code>vecmode</code>	1/0	If 1, solve the vector, i.e. the three coordinate differences between the baseline end points. If 0, solve for the absolute co-ordinates of both end points.
<code>debug</code>	1/0	If 1, produce lots of gory debugging output. See the source for what it all means.

refsat elev	number	Minimum elevation (degs) of the reference satellite used for computing inter-satellite differences. Good initial choice: 30.0.
cutoff elev	number	Cut-off elevation (degs). Good initial choice: 10.0 - 20.0.
rej TP, rej TC reduce	two numbers 1/0	Phase, code triple differences rejection limit (m). Apply post-reduction to combine dependent unknowns.

Ephemeris File Lists

The file `vecsol.nav` contains the names of the navigation RINEX files (“nav files”, extension). Good navigation RINEX files that are globally valid can be found from the CORS website at <http://www.ngs.noaa.gov/CORS/>.

The file `vecsol.eph` contains the names of the precise ephemeris SP3 files (extension `.sp3`) to be used. These should cover the time span of the observations, with time to spare on both ends. Note that the date in the filenames of the SP3 files is given as GPS week + weekday, not year + day of year, as in the observation and nav files.

In the `.nav` and `.eph` files, comment lines have `#` in the first position.

2.48.3 Notes

Currently, `vecsol` does not recover from cycle slips, so the RINEX observation files used have to be fairly clean.

2.49 *WhereSat*

2.49.1 Overview

This application uses input ephemeris to compute the predicted location of a satellite. The Earth-centered, Earth-fixed (ECEF) position of the satellite is reported. Optionally, the topocentric coordinates—azimuth, elevation, and range—can be generated. The user can specify the time interval between successive predictions. Also the output can be generated in a format easily imported into numerical packages.

2.49.2 Usage

<i>WhereSat</i>		
Required Arguments		
Short Arg.	Long Arg.	Description
-e	-eph-files=ARG	Ephemeris source file(s). Can be RINEX nav, SP3, or FIC.
Optional Arguments		
Short Arg.	Long Arg.	Description
-h	-help	Print help usage.
-u	-position=ARG	Antenna position in ECEF (x,y,z) coordinates. Format as string: "X Y Z". used to give user-centered data (SV range, azimuth, and elevation) when SV is in view.
	-start=ARG	Ignore data before this time. Format as string: "MO/DD/YYYY HH:MM:SS".
	-end=ARG	Ignore data after this time. Format as string: "MO/DD/YYYY HH:MM:SS".
-f	-time-format=ARG	CommonTime format specifier used for times in the output. The default is "%4Y %3j %02H:%02M:%4.1f".
-p	-prn=NUM	Which SVs to analyze. Repeat option for multiple satellites. If this option is not specified, all ephemeris data will be processed.
-t	-time=NUM	Time increment in seconds for ephemeris calculation. Default is 900 seconds (15 minutes).

2.49.3 Examples

```
> WhereSat -b aira1720.06n -p 2 -u "918129.01 -4346070.45 803.18"
-s "06/21/2006 17:00:00" -e "06/21/2006 20:00:00" -t 1800
```

```
Antenna Position: 918129 -4.34607e+06 803.18
Navigation File: aira1720.06n
Start Time:      06/21/2006 17:00:00
End Time:        06/21/2006 20:00:00
PRN:             2
```

Prn 2 Earth-fixed position and clock information:

Date	Time(UTC)	X (meters)	Y (meters)	Z (meters)
------	-----------	------------	------------	------------

```
=====
06/21/2006 18:00:00 12758891.971859 18901201.616227 -14049016.596144
06/21/2006 18:30:00 12847888.097031 21541501.416411 -9315422.851798
06/21/2006 19:00:00 12843576.989405 23087218.618683 -3957280.515764
06/21/2006 19:30:00 12450313.769289 23516935.034029 1667186.089065
```

. . .

Clock Correc (s)

```
=====
0.000007
0.000007
0.000007
0.000007
```

Data for user reference frame:

Date	Time(UTC)	Azimuth	Elevation	Range to SV (m)
06/21/2006	18:00:00	130.596202	-43.242769	29627531.177821
06/21/2006	18:30:00	118.680085	-49.681012	29983796.522429
06/21/2006	19:00:00	102.845663	-53.888528	30169796.433699
06/21/2006	19:30:00	84.400419	-55.459042	30197072.648367

Calculated 4 increments for prn 2 .