Post Processing SDK v4.6.5 Release Notes

These Release Notes describe version 4.6.5 of the Post Processing Software Development Kit (PPSDK) developed by Septentrio NV/SA. This software package delivered on the Companion Disk contains an installer, the API documentation, release notes and legal license documents. The installer contains the actual copy of the Post Processing SDK and the Post Processing SDK related tools (PostNav, pvtcalc, sbf2sbf, sbf2rin, sbf2kml, rin2sbf, etc).

Release References

Post-Processing SDK version:	4.6.5
PostNav:	4.6.5
Release Date:	July 06, 2022
Post-Processing Software version:	4.6.5, composed of:
GNSS firmware version:	6.6.4
Applications:	composed of:
PostNav:	4.6.5
RINEX Converter version:	4.6.5
pvtcalc:	4.6.5
basefinder:	4.6.5
basefinder GUI:	4.6.5
rin2sbf:	4.6.5
sbf2cmd:	2.3.34
sbf2gpx:	3.0.38
sbf2kml:	2.8.7
sbf2rin:	13.4.8
sbf2sbf:	1.2.6
ngs2bin:	1.0.7
User Manuals:	composed of:
API Reference	4.6.5
PPSDK Software User Manual	4.6.5
PPSDK 4.6.5 Command and Log Reference Card	4.6.5
PPSDK 4.6.5 Command Line Interface Reference Guide	4.6.5
SBF Reference Guide	4.6.5
Rinex to SBF Conversion	4.6.5

Deliverables

Companion USB Flash Drive Content

1. Release Notes.pdf, these Release Notes
2. LICENSE.TXT, the License Agreement for the Post Processing SDK
3. Septentrio-Post_Processing_SDK-4.6.5-Setup.exe, the Post Processing SDK installer
4. PPSDK Software User Manual.pdf, the Post Processing SDK Software Manual
5. Directory License\, containing the files related to the SDK license

Installer Content

1. The Post Processing SDK Documentation (doc\ directory)
   - ReleaseNotes.pdf, these Release Notes
   - LICENSE.TXT, the License Agreement for the Post Processing SDK
   - Directory doc\manual, containing the Post Processing SDK SW documentation (PPSDK Software User Manual, SBF Reference Guide, Command Line Reference Guide, etc)
   - Directory doc\api, containing the Post Processing SDK function API reference
2. The Post Processing SDK C header files (includes\ directory)
   - basefinderconstellations.h: Contains constellation definitions to be used in BaseFinder
   - mscinttypes.h: ISO C9x compliant inttypes.h for Microsoft Visual Studio
   - mscssntypes.h: Declaration of Septentrio types for Microsoft C/C++ compiler
   - mscstdint.h: ISO C9x compliant stdint.h for Microsoft Visual Studio
   - sbfdef.h: Declarations of SBF (Septentrio Binary Format) blocks
   - sbfsigtypes.h: Declaration of the GNSS signal types used in sbfdef.h
   - snmpclient.h: S)imple (N)etwork (M)anagement (P)rotocol client
   - snmptypes.h: (S)imple (N)etwork (M)anagement (P)rotocol type description
   - ssnbasefinder.h: API of the BaseFinder module
   - ssnenv.h: Declaration of Septentrio retrieval of Enviroment Settings.
   - ssnerror.h: Declaration of types, macros and functions used by the SSN error module
   - ssngetop.h: Septentrio's own implementation of the Posix "getopt()" function
   - ssnhandle.h: SSN safe handle declaration
   - ssnppengine.h: The Post Processing Engine
   - ssnprogressbar.h: Septentrio's implementation of a textual progress bar
   - ssnrnxdec.h: RINEX decoder functionality
   - ssnsbfanalyze.h: Septentrio SBF Analysis library
   - ssnsbfstream.h: The SBF stream module provides easy manipulation of SBF files
   - ssnsdk.h: SSN SDK module declaration
   - ssntypes.h: Declaration of Septentrio types
3. The Post Processing SDK library (library\ directory)
   - library\dynamic\mingw_x64\ppsdk.dll: Dynamic Link Library (DLL file) for MinGW GCC 4.9.2 64 bit
   - library\dynamic\mingw_x64\ppsdk.dll.a: Dynamic Link Library (.a file) for MinGW GCC 4.9.2 64 bit
   - library\dynamic\mingw_x64_6.2.0\ppsdk.dll: Dynamic Link Library (DLL file) for MinGW 6.2.0 GCC 64 bit

- library\dynamic\mingw_x64_6.2.0\ppsdk.dll.a: Dynamic Link Library (.a file) for MinGW 6.2.0 GCC 64 bit
- library\dynamic\vs2013\ppsdk.dll: Dynamic Link Library (DLL file) for Visual Studio 2013
- library\dynamic\vs2013\ppsdk.lib: Dynamic Link Library (LIB file) for Visual Studio 2013
- library\dynamic\vs2013_x64\ppsdk.dll: Dynamic Link Library (DLL file) for Visual Studio 2013 64 bit
- library\dynamic\vs2013_x64\ppsdk.lib: Dynamic Link Library (LIB file) for Visual Studio 2013 64 bit
- library\dynamic\vs2015\ppsdk.dll: Dynamic Link Library (DLL file) for Visual Studio 2015
- library\dynamic\vs2015\ppsdk.lib: Dynamic Link Library (LIB file) for Visual Studio 2015
- library\dynamic\vs2015_x64\ppsdk.dll: Dynamic Link Library (DLL file) for Visual Studio 2015 64 bit
- library\dynamic\vs2015_x64\ppsdk.lib: Dynamic Link Library (LIB file) for Visual Studio 2015 64 bit
- library\dynamic\vs2017\ppsdk.dll: Dynamic Link Library (DLL file) for Visual Studio 2017
- library\dynamic\vs2017\ppsdk.lib: Dynamic Link Library (LIB file) for Visual Studio 2017
- library\dynamic\vs2017_x64\ppsdk.dll: Dynamic Link Library (DLL file) for Visual Studio 2017 64 bit
- library\dynamic\vs2017_x64\ppsdk.lib: Dynamic Link Library (LIB file) for Visual Studio 2017 64 bit
- library\static\mingw_x64\GCC-4.X\MT\ppsdk.lib: Static library for MinGW GCC version 4.x
- library\static\mingw_x64_6.2.0\GCC-6.2.0\MT\ppsdk.lib: Static library for MinGW GCC version 6.2.0
- library\static\vs2013\MD\ppsdk.lib: Static library (LIB file compiled with the multithread and DLL-specific version of the Microsoft C++ run-time library) for Visual Studio 2013
- library\static\vs2013\MT\ppsdk.lib: Static library (LIB file compiled with the multithread and static version of the Microsoft C++ run-time library) for Visual Studio 2013
- library\static\vs2013_x64\MD\ppsdk.lib: Static library (LIB file compiled with the multithread and DLL-specific version of the Microsoft C++ run-time library) for Visual Studio 2013 64 bit
- library\static\vs2013_x64\MT\ppsdk.lib: Static library (LIB file compiled with the multithread and static version of the Microsoft C++ run-time library) for Visual Studio 2013 64 bit
- library\static\vs2015\MD\ppsdk.lib: Static library (LIB file compiled with the multithread and DLL-specific version of the Microsoft C++ run-time library) for Visual Studio 2015
- library\static\vs2015\MT\ppsdk.lib: Static library (LIB file compiled with the multithread and static version of the Microsoft C++ run-time library) for Visual Studio 2015
- library\static\vs2015_x64\MD\ppsdk.lib: Static library (LIB file compiled with the multithread and DLL-specific version of the Microsoft C++ run-time library) for Visual Studio 2015 64 bit

- library\static\vs2015_x64\MT\ppsdk.lib: Static library (LIB file compiled with the multithread and static version of the Microsoft C++ run-time library) for Visual Studio 2015 64 bit
- library\static\vs2017\MD\ppsdk.lib: Static library (LIB file compiled with the multithread and DLL-specific version of the Microsoft C++ run-time library) for Visual Studio 2017
- library\static\vs2017\MT\ppsdk.lib: Static library (LIB file compiled with the multithread and static version of the Microsoft C++ run-time library) for Visual Studio 2017
- library\static\vs2017_x64\MD\ppsdk.lib: Static library (LIB file compiled with the multithread and DLL-specific version of the Microsoft C++ run-time library) for Visual Studio 2017 64 bit
- library\static\vs2017_x64\MT\ppsdk.lib: Static library (LIB file compiled with the multithread and static version of the Microsoft C++ run-time library) for Visual Studio 2017 64 bit

4. Library data requirement (libdata\ directory)
   - libdata\ppsdk.antinfo: The latest available binary antenna description file
5. Post Processing SDK sample applications (samples\sources directory) with sample projects for all compilers.
   - C samples (c folder)
     - ssnsdksetup: The ssnsdksetup sample application is a simple source code that does not execute any particular post-processing operation, but demonstrates how to open an SSN SDK handle, the entry point for any operation with the PPSDK API.
     - computepvt: The computepvt sample application is a command line executable that demonstrates the functionality used to compute a PVT solution.
     - simplepvt: The simplepvt sample application is a simpler usage of the PPEngine which only focuses on a few functions to demonstrate how to compute a new PVT from a rover SBF file and a reference data file.
     - getmibinfo: The getmibinfo sample application is a command line executable that demonstrates how to retrieve the MIB description of the Post Processing SDK.
     - listblocks: The listblocks sample application is a command line executable that demonstrates how to list the contents of an SBF file and print out the GNSS time stamp and ID of the blocks.
     - sendcommand: The sendcommand sample application is a command line executable that demonstrates how to send an ASCII command to the PP Engine.
     - basefindergetssnref: The basefindergetssnref sample application is a command line executable that demonstrates how to get a reference file for SSN's headquarter using SSN BaseFinder.
     - ssnerrorusage: The ssnerrorusage sample application is a very simple command line program which demonstrates how error codes are reported by the functions of the PPSDK API, and how to retrieve specific information from them.
     - sbfstreamupdate: The sbfstreamupdate sample application shows how an input SBF file can be manipulated and modified as you please, before to be saved into a new output file.
     - rinexconversion: The rinexconversion sample program demonstrates how to use the RINEX Decoder module to convert a set of RINEX files into a single output SBF stream which is in turn saved into a file.
     - sbfanalysis: The sbfanalysis sample application uses the SBF Analyze module to launch two different analyses on an SBF stream that is previously opened.

- C# samples (c# folder)
  - Ppsdk: To be able to use the Post Processing SDK API in a C# application the functions and types used have to be declared. For the samples this is done in a separate C# library called PpSdk.
  - ListMissingEpochs: The ListMissingEpochs sample application is a GUI application that demonstrates how to check for possible missing SBF blocks at certain epochs.
6. Assets (samples\assets directory)
   - samples\assets\demo.sbf: This demo SBF file which can be used for testing the Post-Processing SDK. The file contains 100% standalone. You can compute an RTK solution by using the same file as a Base Reference file when using pvtcalc or PostNav. If you post-process the file alone with the default settings of the PPSDK engine then an SBAS solution will be obtained. The same file can be used with other applications such as sbf2sbf, sbfblocks, etc.
7. Septentrio tools (tools\ directory)
   - tools\binaries\win32\pvtcalc.exe: Septentrio Post Processing Command Line application (source code included in tools\sources\pvtcalc)
   - tools\binaries\win32\rin2sbf.exe: Converter of RINEX into SBF format (source code included in tools\sources\rin2sbf)
   - tools\binaries\win32\sbf2cmd.exe: Converter SBF to CMD format
   - tools\binaries\win32\sbf2gpx.exe: Converter SBF to GPX format
   - tools\binaries\win32\sbf2kml.exe: Converter SBF to KML format
   - tools\binaries\win32\sbf2rin.exe: Converter SBF to RINEX format
   - tools\binaries\win32\sbf2sbf.exe: Editor of SBF files with generic functions (source code included in tools\sources\sbf2sbf)
   - tools\binaries\win32\sbfblocks.exe: Program that lists all SBF blocks in an SBF file
   - tools\binaries\win32\ngs2bin.exe: Program that converts NGS Antenna files into binary format for use in the PPSDK
8. Septentrio Applications
   - applications\postnav.exe: The Septentrio Graphical Post Processing application
   - applications\rinconverter.exe: The Septentrio Graphical RINEX Converter
   - applications\ppsdksettingsmanager: The Septentrio PPSDK Settings Manager
   - applications\bin2asc: The Septentrio SBF to ASCII command line tool
9. Directory prerequisites\, containing the Microsoft Visual C++ 2013, 2013_x64, 2015, 2015_x64 , 2017 and 2017_x64 Re-distributable Packages, installs runtime components of Visual C++ Libraries required to run applications developed with Visual C++ on a computer that does not have Visual C++ 2013, 2015 or 2017 installed.
10. uninstall.exe, the uninstaller for the Post Processing SDK

Note that since the 4.6.0 release of PPSDK support for Visual Studio 2008 and 2010 is dropped. The 4.6.3 release of PPSDK is the last release that will have support for Visual Studio 2013 and MinGW GCC 4.9.2.

Legal Notice

Septentrio does not authorize the use of its products as critical components in devices or systems intended for safety-of-life applications or in devices or systems, which failure may endanger life or cause injuries, without express written approval.

All the software and documentation delivered with the Post Processing SDK is licensed as explained in the LICENSE.TXT file.

A note about compatibility

The software is compatible with:

- SBF files from Septentrio AsteRx, Altus and PolaRx family of receivers
- RINEX observation and navigation files versions 2.1x and 3.0x

Minimum System Requirements

The following operating systems are supported:

- Windows 7®
- Windows 8®
- Windows 10®

The minimal system requirements[1] are:

- 1 GHz processor
- 1 GB RAM
- 1024×768 or higher resolution

Installation

The Post Processing SDK installer package can be run on MS Windows by executing the "Septentrio-Post_Processing_SDK-4.6.5-Setup.exe" file. The installer will guide the user through the different installation steps.

The Post Processing SDK does not install any 3rd party software and does not require an active internet connection with exception of BaseFinder where a connection is needed to download data files..

Please consult the Installation section of the PPSDK Software User Manual for further details.

Post Processing SDK v4.6.5

Post-processing engine

Post Processing SDK differences to previous versions

Post Processing SDK differences introduced in PPSDK v4.6.5

---

[1]   Improved CPU performance will be obtained using an up to date PC.

No differences

Post Processing SDK differences introduced in PPSDK v4.6.4

Issues resolved in Post Processing SDK:

1. The functions "SSNSBFStream_insertCommandOverTime" and "SSNSBF-Stream_insertFileCommandOverTime" now only consider the time of SBF bocks with receiver time to search for the correct insertion location.

Post Processing SDK differences introduced in PPSDK v4.6.2

Issues resolved in Post Processing SDK:

1. When both the reference position and antenna offset specified set, the antenna offset is no longer rejected.

Post Processing SDK differences introduced in PPSDK v4.6.1

Issues resolved in Post Processing SDK:

1. An issue has been fixed that could trigger a segmentation fault when an ssnhandle was closed twice. Now the ssnhandle will internally refer to an element that keeps track of the status, and an operation on an invalid file handle no longer causes crashes.
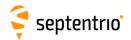
Post Processing SDK differences introduced in PPSDK v4.6.0

New features in Post Processing SDK:

1. BeiDou or Galileo navigation files are now supported by the RINEX converter.
2. The PVT engine now also supports Beidou and Galileo constellations in RTK.
3. The ExtEventBaseVectGeod SBF block was added to output the base vector computation triggered by external events.
4. A silent installation option was added to allow automatic update in a distributed environment.
5. Support for Microsoft Visual Studio 2017 was added (support for Microsoft Visual Studio 2008 and 2010 has been dropped).
6. BaseFinder is a new application and API component to automate the retrieval of reference station data. BaseFinder optimizes the use of public data for best GNSS performance and is an optional feature.
7. Dual antenna GNSS heading computation is now supported.
8. Horizontal and vertical precision of the extrapolated position was added to the SBF messages ExtEventPVTCartesian and ExtEventPVTGeodetic.

Issues resolved in Post Processing SDK:

1. Support for RTCMv3 message 1033 insertion was added. This message is to be used for improving the antenna phase centre compensation and satellite bias for GLONASS constellation.
2. The SBF stream merge options have been improved. Where before inserting multiple reference streams lead to an exponentially increase in the filesize this is no longer the case.
3. When closing the last ppengine handle the diffcorr database is now removed were before is it was kept. We consider the risk that the database contents changes is too big to re-use it.

4. An option has been introduced in the PPSDK API for fast processing by adding the SSNPPENGINE_OPTIONS_FAST flag to ssn_ppengine_options_t. This fast processing option skips some pre-processing steps that are used to improve PVT availability and accuracy.

   Because of this change the options SSNPPENGINE_OPTIONS_SERIALIZE and SSNPPENGINE_OPTIONS_DESERIALIZE are no longer used and deprecated.
5. The PPSDK environment variable SSN_NEQUICK_PATH has no more effect on post-processing and does no longer need to be defined.
6. PPSDK has been made more robust to misbehaving GLONASS satellites.
7. When satellites or signals in the input SBF file are unknown to the software, that information is ignored and the software will no longer stop functioning.
8. The backwards post-processing engine has been updated to support long baseline processing (previously this was only supported in forward mode).
9. The command-line option -N of pvtcalc works again as expected.
10. The performance of the PPSDK is optimized when using RTCM3 messages. The usage of RTCM2 for insertion of base/reference data is thus discouraged. The RTCM2 option was removed from the help and documentation, but is still available.
11. The antenna information file has been updated with the calibration data included in the NGS file of 12/07/2019.

Post Processing SDK differences introduced in PPSDK v4.1.8

New features in Post Processing SDK:

1. GPS, GLONASS and BeiDou MSM4 messages are now supported in the diffcorr insertion process.

Issues resolved in Post Processing SDK:

1. For best RTK performance, it is recommended to use RTCM3 format in the diffcorr insertion process. Please note that RTCM2 is no longer the default diffcorr stream format in either pvtcalc or PostNav. Also note that the aliases SSNSBFSTREAM_RTCMV2_MESSAGE_ALL and SSNSBFSTREAM_RTCMV3_MESSAGE_ALL are deprecated. They are still functional but it is recommended to use instead the aliases SSNSBFSTREAM_RTCMV2_MESSAGE_DEFAULT and SSNSBFSTREAM_RTCMV3_MESSAGE_DEFAULT.
2. An issue where the multipath mitigation of weak GNSS signals could influence RTK availability has been solved.
3. When adding SSNSBFSTREAM_RTCMV2_MESSAGE_DEFAULT (or SSNSBFSTREAM_RTCMV2_MESSAGE_ALL) to the messages parameter in the SSNSDK_SBFStream_insertReferenceStream call it is now possible to add other messages as well. Before all other messages were removed. Note that this wrong behaviour was not present when using RTSSNSBFSTREAM_RTCMV3_MESSAGE_DEFAULT (or SSNSBFSTREAM_RTCMV3_MESSAGE_ALL).
4. Output of local positions and coordinates through the PosLocal and PosProjected SBF blocks is now supported if RTCMv3 MT1021-1027 messages are provided.
5. Some warnings were persistent in the PPSDK when not closing the handles. This made that if such an warning was triggered once, the next runs seemed to trigger the same warning since it was not reset. In order to reset the warnings before a next run a new API function has been added: SSNPPengine_ResetWarningCounters().
6. The API call to set the RINEX conversion Doppler computation mode option (SSNRNXDec_setParameters with parameter SSNRNXDEC_PARAM_FORCEDOPPLERFROMCARRIERPHASE) has now an extra option

to automatically compute the doppler from the carrier phase measurements if not available in the RINEX file: SSNRNXDEC_DOPPLERFROMLX_AUTO.

Post Processing SDK known issues and limitations

1. It is not recommended to use SBF output files from PPSDK as a input for a new PPSDK computation or for measurement analysis.
2. The progress information returned by the PPSDK and/or shown in pvtcalc and PostNav might not always indicate an increasing degree of completion of the current process.
3. When using all RTCM2/RTCM3 messages for the insertion of DiffCorr messages into an SBF stream, the RTK performance may be different from (and in some cases lower than) that obtained when using only the RTK RTCM2/RTCM3 messages. The difference in RTK performance may depend on the Rover signals available, the Reference file data that is chosen for the construction of the RTCM messages, etc. Please select the most appropriate group of RTCM messages for your intended post-processing.
4. The PPSDK cannot handle corrupt input files that contain repeated SBF blocks.
5. The RAIMStatistics block may contain incorrect information after post-processing. It is not recommended to use the block in this version of the PPSDK.
6. The PPSDK does not support files larger than 2GB in size.
7. When using PostNav the Signal Usage screen may show signals which are not supported in either the PPSDK or the user's permissions file. It is recommended to use the Permitted Capabilities screen of PostNav to understand the actual capabilities of the product.
8. While the ReceiverStatus block contains proper Error and Status information about the PPSDK engine, it may also contain certain information (e.g. CPU, UpTime, etc.) which is not relevant to the PPSDK. This irrelevant information corresponds to values which are either 0 or null.
9. RINEX files having names containing multiple '.' (dot) separators might not be accepted as input to the RINEX-to-SBF converter of the PPSDK.
10. The cancel operation introduced by the PPSDK returns SSNERROR_WARNING_OK when invoked; this might be changed in future versions of the PPSDK for better handling of the cancel operation by the top application.
11. PPSDK does not support Moving Base in backwards PVT computation.
12. When the user does not specify the directory in which the diffcorr database should reside, PPSDK tries to create this database in the working directory. Since this is not allowed in all host configurations, it is recommended that the user specifies a directory which is guaranteed to be writable.
13. A short period of RTK Float solution may be visible when sequential RINEX observation files are used with non-continuous lock-time information across the files.
14. The PPSDK engine has been improved for handling basestations with a update rate lower than 1Hz, as a side-effect the PVT calculation duration might increase. Please see the "Obtain a PVT Solution" section in the PPSDK Software User Manual.
15. This relnote should be hidden as long as not fixed!

PostNav v4.6.5

Application for post-processing of SBF files using the PPSDK

PostNav differences to previous versions

PostNav differences introduced in PPSDK v4.6.5

No differences

PostNav differences introduced in PPSDK v4.6.4

Issues resolved in PostNav:

1. The functions "SSNSBFStream_insertCommandOverTime" and "SSNSBF-Stream_insertFileCommandOverTime" now only consider the time of SBF bocks with receiver time to search for the correct insertion location.

PostNav differences introduced in PPSDK v4.6.2

Issues resolved in PostNav:

1. When both the reference position and antenna offset specified set, the antenna offset is no longer rejected.

PostNav differences introduced in PPSDK v4.6.1

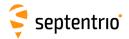No differences

PostNav differences introduced in PPSDK v4.6.0

New features in PostNav:

1. The PVT engine now also supports Beidou and Galileo constellations in RTK.
2. The ExtEventBaseVectGeod SBF block was added to output the base vector computation triggered by external events.
3. Dual antenna GNSS heading computation is now supported.
4. The resolution of the start and stop time input fields in PostNav was increased from minutes to seconds.

Issues resolved in PostNav:

1. Antenna type selection in PostNav is now properly used.
2. When closing the last ppengine handle the diffcorr database is now removed were before is it was kept. We consider the risk that the database contents changes is too big to re-use it.
3. When satellites or signals in the input SBF file are unknown to the software, that information is ignored and the software will no longer stop functioning.
4. The option "ASN.1 file to translate commands" has been removed for simplicity from PostNav since it was not used.
5. The SBF GEO Navigation Data option has been removed for simplicity from PostNav.
6. The Receiver Compatiblity option has been removed for simplicity from PostNav.
7. The SBF Reference File RTCM version option has been removed for simplicity from Post-Nav. It is now always set to RTCM3 default messages.
8. The Measurement interval option has been removed for simplicity from PostNav.
9. The Navigation Data Usage option has been removed for simplicity from PostNav, it is now always set to use all available navigation data blocks.

PostNav differences introduced in PPSDK v4.1.8

New features in PostNav:

1. For best RTK performance, it is recommended to use RTCM3 format in the diffcorr insertion process. Therefore RTCM3 (with default messages) is now the default diffcorr stream format.

Issues resolved in PostNav:

1. In post processing, the preference for GLONASS L2CA phase measurements in RTK resulted in GLONASS not being used when only GLONASS L2P phase measurements were available in the base station RINEX data. This has now been rectified by fully implementing the GLONASS L2P phase measurements in the RINEX file.
2. Output of local positions and coordinates through the PosLocal and PosProjected SBF blocks is now supported if RTCMv3 MT1021-1027 messages are provided.
3. An error in PostNav made that the reference position and antenna (both offset and type) were not cleared properly in the underlying library once they were applied manually and cleared afterwards for another calculation by unchecking the corresponding check-boxes.

PostNav known issues and limitations

1. The progress information returned by the PPSDK and/or shown in pvtcalc and PostNav might not always indicate an increasing degree of completion of the current process.
2. The PPSDK does not support files larger than 2GB in size.
3. When the user selects "Start Time (GPS):" from the advanced options of PostNav, the application may show incorrect information. This may happen in the case of files containing certain messages that are not properly supported by the PPSDK. To obtain the proper beginning/end time of a file, the user may use either the sbfblocks tool provided with the PPSDK or the RxTools that is downloadable from the Septentrio website.
4. The PostNav application will change the output settings without notifying the user and will select certain blocks (like the GUI group and other SBF blocks) if no messages, port or rate are set from the application's 'SBF Output' screen.
5. It might not be possible to view the entire 'advanced options' window of PostNav on a screen having low display resolution (e.g. 1366x768).
6. When specifying the reference antenna offset all 3 components must be given.
7. When specifying the reference position via the command line it is not applied during the calculations.

Rinex Converter v4.6.5

Graphical version of the Rinex To SBF converter

Rinex Converter differences to previous versions

Rinex Converter differences introduced in PPSDK v4.6.5

No differences

Rinex Converter differences introduced in PPSDK v4.6.4

Issues resolved in Rinex Converter:

1. A "start moving antenna" special event (epoch flag 2) no longer causes an "Invalid Epoch Flag" error.
2. The special block "epoch 4" no longer causes an "Invalid Epoch Flag" error.

Rinex Converter differences introduced in PPSDK v4.6.2

No differences

Rinex Converter differences introduced in PPSDK v4.6.1

No differences

Rinex Converter differences introduced in PPSDK v4.6.0

New features in Rinex Converter:

1. BeiDou or Galileo navigation files are now supported by the RINEX converter.
2. Mixed Navigation files can now be converted by the RINEX converter.
3. Rin2Sbf can now handle HATANAKA compressed RINEX files.

Issues resolved in Rinex Converter:

1. The option to ad a GLONASS Frequency File is is removed (this functionality is deprecated in the PPSDK API).
2. The option "Do not use built-in GLONASS frequency numbers) is removed (this functionality is deprecated in the PPSDK API).

Rinex Converter differences introduced in PPSDK v4.1.8

New features in Rinex Converter:

1. The option "auto" has been added to compute the Doppler from the carrier phase if not available in the RINEX file, or take it from the RINEX file if available. By default, the Doppler in the SBF file is taken from the RINEX file if present, or set to its do-not-use value if there is no Dx observable in the RINEX file.

Issues resolved in Rinex Converter:

1. RINEX v3.xx observation files with more than 26 observation types per constellation were incorrectly decoded. This is now fixed.
2. The issue of low RTK availability caused by RINEX files containing a mix of either GPS L2P or GPS L2C measurements has been resolved.
3. The error "RINEX file contains data from constellations that are not supported" is now reported when using Galileo or BeiDou RINEX navigation files since those are not supported.

Rinex Converter known issues and limitations

1. RINEX files having names containing multiple '.' (dot) separators might not be accepted as input to the RINEX-to-SBF converter of the PPSDK.

pvtcalc v4.6.5

The application for doing pvt computations

pvtcalc differences to previous versions

pvtcalc differences introduced in PPSDK v4.6.5

No differences

pvtcalc differences introduced in PPSDK v4.6.4

Issues resolved in pvtcalc:

1. The functions "SSNSBFStream_insertCommandOverTime" and "SSNSBF-Stream_insertFileCommandOverTime" now only consider the time of SBF bocks with receiver time to search for the correct insertion location.

pvtcalc differences introduced in PPSDK v4.6.2

Issues resolved in pvtcalc:

1. When both the reference position and antenna offset specified set, the antenna offset is no longer rejected.

pvtcalc differences introduced in PPSDK v4.6.1

No differences

pvtcalc differences introduced in PPSDK v4.6.0

New features in pvtcalc:

1. The PVT engine now also supports Beidou and Galileo constellations in RTK.
2. The ExtEventBaseVectGeod SBF block was added to output the base vector computation triggered by external events.
3. Dual antenna GNSS heading computation is now supported.

Issues resolved in pvtcalc:

1. When satellites or signals in the input SBF file are unknown to the software, that information is ignored and the software will no longer stop functioning.
2. The command-line option -N of pvtcalc works again as expected.
3. The performance of the PPSDK is optimized when using RTCM3 messages. The usage of RTCM2 for insertion of base/reference data is thus discouraged. The RTCM2 option was removed from the help and documentation, but is still available.
4. The ANS.1 file option is now only mandatory when using both the "-x" (execute commands) and "-v" (verbose) options.
5. The option -g geo_navfile has been removed from the pvtcalc documentation and is now considered an advanced option. For backwards compatibility reasons the option still works on pvtcalc.
6. The option -C (receiver compatibility mode) has been removed from the pvtcalc documentation and is now considered an advanced option. For backwards compatibility reasons the option still works on pvtcalc.

7. The option -M (Measurement interval) has been removed from the pvtcalc documentation and is now considered an advanced option. For backwards compatibility reasons the option still works on pvtcalc.
8. The option -N (Navigation data usage) has been removed from the pvtcalc documentation and is now considered an advanced option. For backwards compatibility reasons the option still works on pvtcalc.

pvtcalc differences introduced in PPSDK v4.1.8

New features in pvtcalc:

1. For best RTK performance, it is recommended to use RTCM3 format in the diffcorr insertion process. Therefore RTCM3 (with default messages) is now the default diffcorr stream format.

Issues resolved in pvtcalc:

1. In post processing, the preference for GLONASS L2CA phase measurements in RTK resulted in GLONASS not being used when only GLONASS L2P phase measurements were available in the base station RINEX data. This has now been rectified by fully implementing the GLONASS L2P phase measurements in the RINEX file.
2. Output of local positions and coordinates through the PosLocal and PosProjected SBF blocks is now supported if RTCMv3 MT1021-1027 messages are provided.

pvtcalc known issues and limitations

1. The progress information returned by the PPSDK and/or shown in pvtcalc and PostNav might not always indicate an increasing degree of completion of the current process.
2. The PPSDK does not support files larger than 2GB in size.
3. The PPSDK engine has been improved for handling basestations with a update rate lower than 1Hz, as a side-effect the PVT calculation duration might increase. Please see the "Obtain a PVT Solution" section in the PPSDK Software User Manual.
4. When specifying the reference antenna offset all 3 components must be given.

BaseFinder v4.6.5

BaseFinder differences to previous versions

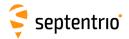**BaseFinder differences introduced in PPSDK v4.6.5**

No differences

BaseFinder differences introduced in PPSDK v4.6.4

No differences

BaseFinder differences introduced in PPSDK v4.6.2

No differences

BaseFinder differences introduced in PPSDK v4.6.1

No differences

BaseFinder differences introduced in PPSDK v4.6.0

New features in BaseFinder:

1. BaseFinder is a new application and API component to automate the retrieval of reference station data. BaseFinder optimizes the use of public data for best GNSS performance and is an optional feature.

BaseFinder differences introduced in PPSDK v4.1.8

No differences

BaseFinder GUI v4.6.5

The GUI version of BaseFinder

BaseFinder GUI differences to previous versions

BaseFinder GUI differences introduced in PPSDK v4.6.5

No differences

BaseFinder GUI differences introduced in PPSDK v4.6.4

No differences

BaseFinder GUI differences introduced in PPSDK v4.6.2

No differences

BaseFinder GUI differences introduced in PPSDK v4.6.1

No differences

BaseFinder GUI differences introduced in PPSDK v4.6.0

New features in BaseFinder GUI:

1. BaseFinder is a new application and API component to automate the retrieval of reference station data. BaseFinder optimizes the use of public data for best GNSS performance and is an optional feature.

BaseFinder GUI differences introduced in PPSDK v4.1.8

No differences

BaseFinder GUI known issues and limitations

1. BaseFinder uses "Open Street Maps" to show a map. When the servers of Open Street Maps are getting to many requests you might see the message "Server busy" on your map.

rin2sbf v4.6.5

The conversion tool on top of the RINEX 2 SBF class

rin2sbf differences to previous versions

rin2sbf differences introduced in PPSDK v4.6.5

No differences

rin2sbf differences introduced in PPSDK v4.6.4

Issues resolved in rin2sbf:

1. A "start moving antenna" special event (epoch flag 2) no longer causes an "Invalid Epoch Flag" error.
2. The special block "epoch 4" no longer causes an "Invalid Epoch Flag" error.

rin2sbf differences introduced in PPSDK v4.6.2

No differences

rin2sbf differences introduced in PPSDK v4.6.1

No differences

rin2sbf differences introduced in PPSDK v4.6.0

New features in rin2sbf:

1. BeiDou or Galileo navigation files are now supported by the RINEX converter.
2. Mixed Navigation files can now be converted by the RINEX converter.
3. Rin2Sbf can now handle HATANAKA compressed RINEX files.

Issues resolved in rin2sbf:

1. Leap seconds handling has been improved in rin2sbf when converting files including the BeiDou constellation.
2. The option -g (GLONASS frequency file) is deprecated and removed from rin2sbf.
3. The option -b (Do not use Backup frequency number list for GLONASS) is deprecated and removed from rin2sbf.

rin2sbf differences introduced in PPSDK v4.1.8

New features in rin2sbf:

1. The Doppler in SBF is now automatically computed from the carrier phase measurements if not available in the RINEX file. The option "-N" has been added to disable this computation: with the "-N" option, the Doppler in SBF is set to its do-not-use value if there is no Dx observable in the RINEX file.

Issues resolved in rin2sbf:

1. RINEX v3.xx observation files with more than 26 observation types per constellation were incorrectly decoded. This is now fixed.

2. The issue of low RTK availability caused by RINEX files containing a mix of either GPS L2P or GPS L2C measurements has been resolved.
3. The error "RINEX file contains data from constellations that are not supported" is now reported when using Galileo or BeiDou RINEX navigation files since those are not supported.

rin2sbf known issues and limitations

1. When converting RINEX files to SBF without carrier phase on L1, the error "Invalid RINEX file" will be returned. It should be ensured that RINEX files with at least L1 carrier phase data are used.
2. RINEX files having names containing multiple '.' (dot) separators might not be accepted as input to the RINEX-to-SBF converter of the PPSDK.
3. A short period of RTK Float solution may be visible when sequential RINEX observation files are used with non-continuous lock-time information across the files.

sbf2cmd v2.3.34

Converter of SBF data to Commands. The tool allows users to visualize the different settings of a receiver over time. The commands are shown in an ASCII file created from the tool.

For full output the tool also uses the MIB file of the receiver used while the SBF data file was recorded.

sbf2cmd differences to previous versions

sbf2cmd differences introduced in PPSDK v4.6.5

No differences

sbf2cmd differences introduced in PPSDK v4.6.4

No differences

sbf2cmd differences introduced in PPSDK v4.6.2

No differences

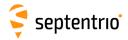**sbf2cmd differences introduced in PPSDK v4.6.1**

No differences

sbf2cmd differences introduced in PPSDK v4.6.0

No differences

sbf2cmd differences introduced in PPSDK v4.1.8

No differences

sbf2gpx v3.0.38

Converter of SBF data to GPX format. The GPX format is a common standard for GPS receivers used for storing position information in an XML-like file. This format can be loaded into Google Earth or can be converted using GPS Visualizer (www.gpsvisualizer.com).

sbf2gpx differences to previous versions

sbf2gpx differences introduced in PPSDK v4.6.5

No differences

sbf2gpx differences introduced in PPSDK v4.6.4

No differences

sbf2gpx differences introduced in PPSDK v4.6.2

No differences

sbf2gpx differences introduced in PPSDK v4.6.1

No differences

sbf2gpx differences introduced in PPSDK v4.6.0

No differences

sbf2gpx differences introduced in PPSDK v4.1.8

No differences

sbf2gpx known issues and limitations

1. Conversion to GPX assumes the SBF file contains positions in WGS84. If the SBF file contains positions in another datum, the GPX file will show position errors introduced by the datum difference.

sbf2kml v2.8.7

Converter of SBF data to KML format. The KML format is the common standard used by Google Earth. This format can be directly loaded into Google Earth. KML allows for several visualization features based on coloring and 3D objects shown directly in Google Earth.

sbf2kml differences to previous versions

sbf2kml differences introduced in PPSDK v4.6.5

No differences

sbf2kml differences introduced in PPSDK v4.6.4

No differences

sbf2kml differences introduced in PPSDK v4.6.2

No differences

sbf2kml differences introduced in PPSDK v4.6.1

No differences

sbf2kml differences introduced in PPSDK v4.6.0

No differences

sbf2kml differences introduced in PPSDK v4.1.8

No differences

sbf2kml known issues and limitations

1. If a file does not contain tracking data at the beginning (e.g. no tracking SV's even if channel status is output) then the Survey track functionality of KML will not work properly.
2. Conversion to KML assumes the SBF file contains positions in WGS84. If the SBF file contains positions in another datum, the KML file will show position errors introduced by the datum difference.

sbf2rin v13.4.8

Converter of SBF data to RINEX format. RINEX is the Receiver INdependent EXchange format. A set of standard definition and formats for ASCII data files to promote the free exchange of GNSS data and facilitate the use of data from any GNSS receiver with any software package. The tool allows for converting from SBF files into RINEX 2.1 and 3.0 formats.

sbf2rin differences to previous versions

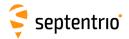**sbf2rin differences introduced in PPSDK v4.6.5**

No differences

sbf2rin differences introduced in PPSDK v4.6.4

No differences

sbf2rin differences introduced in PPSDK v4.6.2

No differences

sbf2rin differences introduced in PPSDK v4.6.1

No differences

sbf2rin differences introduced in PPSDK v4.6.0

No differences

sbf2rin differences introduced in PPSDK v4.1.8

No differences

sbf2sbf v1.2.6

Converter of SBF data to the same SBF format. The idea of this tool is to allow users to manipulate SBF files with different functionalities such as cutting the file, removing specific blocks from a file, merging different SBF files into one, etc.

sbf2sbf differences to previous versions

sbf2sbf differences introduced in PPSDK v4.6.5

No differences

sbf2sbf differences introduced in PPSDK v4.6.4

No differences

sbf2sbf differences introduced in PPSDK v4.6.2

No differences

sbf2sbf differences introduced in PPSDK v4.6.1

No differences

sbf2sbf differences introduced in PPSDK v4.6.0
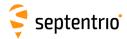
Issues resolved in sbf2sbf:

1. The sbf2sbf application does not require the user to have write permission for the SBF input file anymore.

sbf2sbf differences introduced in PPSDK v4.1.8

No differences

ngs2bin v1.0.7

This is the tool which allows conversion between Antenna Calibration NGS format and Binary format which can be used in the PPSDK. Once you have a binary file then it can be converted into an SREC or SUF file.

ngs2bin differences to previous versions

ngs2bin differences introduced in PPSDK v4.6.5

No differences

ngs2bin differences introduced in PPSDK v4.6.4

No differences

ngs2bin differences introduced in PPSDK v4.6.2

No differences

ngs2bin differences introduced in PPSDK v4.6.1

No differences

ngs2bin differences introduced in PPSDK v4.6.0

No differences

ngs2bin differences introduced in PPSDK v4.1.8

No differences

Contact

Go to Septentrio web site www.septentrio.com or contact the Septentrio support team by email to support@septentrio.com.