Bernese GNSS Software: Processing Examples in Version 5.2

R. Dach and the
Bernese GNSS Software development team

Astronomical Institute, University of Bern, Switzerland
Sidlerstrasse 5, CH-3012 Bern

Bern, July 2, 2013

To all users of the Bernese GNSS Software, Version 5.2
that are familiar with the previous version 5.0 .

Overview

Directory Structure

The Processing Examples in Version 5.2

Example Dataset

Example BPE: General Aspects

Example BPE: Specific Aspects

Configure Example to Run User Data

R. Dach et al.: Bernese GNSS Software: Processing Examples in Version 5.2
Document compiled: Bern, July 2, 2013

Directory Structure

Directory Structure
Main Directories
Dataflow in Version 5.2
Consequences from the Example Dataflow

Directory Structure

Main Directories:

- `$C=BERN52`
  Program area with source code, executables, and supporting files
- `$U=GPSUSER52`
  User area with user–specific settings for interactive processing and the BPE configurations running for this user
- `$T=GPSTEMP52`
  Temporary file area for the users BPE processing

R. Dach et al.: Bernese GNSS Software: Processing Examples in Version 5.2
Document compiled: Bern, July 2, 2013

Directory Structure

Main Directories:

- $D=GPSDATA/DATAPOOL[1]
  Local database with all external files needed for GNSS data processing
- $P=GPSDATA/CAMPAIGN52
  Campaign area where the processing with the Bernese GNSS Software takes place
- $S=GPSDATA/SAVEDISK[1]
  Product archive containing all GNSS derived products for further analysis

[1] New in Version 5.2

Dataflow in Version 5.2

Why do we need two more directories?

- The idea was already behind the processing examples of Version 5.0 . For simplicity the data have been distributed in the campaign and the "copy" and "save" scripts have been provided as dummy scripts.
- It was not clear to many users why these two scripts exist.
  The campaign should not be used as a long–term storage area for input or result files.
- With the examples of Version 5.2 we start to give the user the full structure of dataflow as we think it is the best:

DATAPOOL → CAMPAIGN → SAVEDISK

Motivation for the DATAPOOL area

The idea of the DATAPOOL area is to place local copies of external files somewhere on your file system. It has several advantages compared to downloading the data each time when starting the processing:

- The files are downloaded only once, even if they are used for several campaigns.
- The data download can be organized with a set of scripts running independently from the Bernese GNSS Software environment, scheduled by the expected availability of the external files to download.
- The processing itself becomes independent from the availability of external data sources.

Structure of the DATAPOOL area

The DATAPOOL area is proposed to be organized as follows, reflecting CODE's own data and product organization:

`/RINEX`:	navigation and Hatanaka-compressed observation RINEX files
`/HOURLY`:	same as RINEX but for near real-time applications
`/STAT_LOG`:	station information files (site logs)
`/COD`:	orbits, EOP, and clock products from the CODE; fully combined multi-GNSS solutions
`/IGS`:	like COD but for IGS combined products (IGS/IGL products)
`/BSW52`:	external files in BSW format, such as ionosphere maps and DCB files, neutral with respect to the data a user will process
`/REF52`:	collection of BSW files useful to several campaigns such as `.CRD`, `.VEL`, and `.STA` files
`/VMF1`:	grids for the Vienna Mapping Function
`/...`:	according to needs, as per VMF1

(see `${D}/README_DATAPOOL.TXT` for a detailed description)

Populating the DATAPOOL area

Let us assume that ${D} points to your DATAPOOL directory:

/RINEX:

see http://igscb.jpl.nasa.gov/components/data.html

```
wget ftp://cddis.gsfc.nasa.gov/gps/data/daily/2011/206/11d/onsa2060.11d.Z
mv onsa2060.11d.Z ${D}/RINEX/ONSA2060.11d.Z
```

/STAT_LOG:

see http://igscb.jpl.nasa.gov/network/netindex.html

```
wget http://igscb.jpl.nasa.gov/igscb/station/log/onsa_20120116.log
mv onsa_20120116.log ${D}/STAT_LOG
```


Populating the DATAPOOL area

Let us assume that ${D} points to your DATAPOOL directory:

/COD:

see ftp://ftp.unibe.ch/aiub/AIUB_AFTP.TXT

```
wget ftp://ftp.unibe.ch/aiub/CODE/2011/COD16461.EPH.Z
wget ftp://ftp.unibe.ch/aiub/CODE/2011/COD16461.CLK.Z
mv COD16461.EPH.Z ${D}/COD/COD16461.EPH.Z
mv COD16461.CLK.Z ${D}/COD/COD16461.CLK.Z

wget ftp://ftp.unibe.ch/aiub/CODE/2011/COD16467.ERP.Z
wget ftp://ftp.unibe.ch/aiub/CODE/2011/COD16467.SUM.Z
mv COD16467.SUM.Z ${D}/COD/COD16467.SUM.Z
mv COD16467.ERP.Z ${D}/COD/COD16467.ERP.Z
```

Populating the DATAPOOL area

Let us assume that ${D} points to your DATAPOOL directory:

/IGS:

see seehttp://igscb.jpl.nasa.gov/components/prods.html

```
wget ftp://cddis.gsfc.nasa.gov/gps/products/1646/igs16461.sp3.Z
wget ftp://cddis.gsfc.nasa.gov/gps/products/1646/igs16461.clk.Z
wget ftp://cddis.gsfc.nasa.gov/gps/products/1646/igs16461.clk_30s.Z
wget ftp://cddis.gsfc.nasa.gov/glonass/products/1646/igl16461.sp3.Z
mv igs16461.sp3.Z ${D}/IGS/IGS16461.SP3.Z
mv igs16461.clk.Z ${D}/IGS/IGS16461.CLK.Z
mv igs16461.clk_30s.Z ${D}/IGS/IGS16461.CLK_30S.Z
mv igl16461.sp3.Z ${D}/IGS/IGL16461.SP3.Z

wget ftp://cddis.gsfc.nasa.gov/gps/products/1646/igs16467.erp.Z
wget ftp://cddis.gsfc.nasa.gov/gps/products/1646/igs16467.sum.Z
wget ftp://cddis.gsfc.nasa.gov/glonass/products/1646/igl16467.sum.Z
mv igs16467.erp.Z ${D}/IGS/IGS16467.ERP.Z
mv igs16467.sum.Z ${D}/IGS/IGS16467.SUM.Z
mv igl16467.sum.Z ${D}/IGS/IGL16467.SUM.Z
```


Populating the DATAPOOL area

Let us assume that ${D} points to your DATAPOOL directory:

/BSW52:

see ftp://ftp.unibe.ch/aiub/AIUB_AFTP.TXT

```
wget ftp://ftp.unibe.ch/aiub/CODE/2011/CGIM2060.11N.Z
mv CGIM2060.11N.Z ${D}/BSW52

wget ftp://ftp.unibe.ch/aiub/CODE/2011/P1C11107_ALL.DCB.Z
mv P1C11107_ALL.DCB.Z ${D}/BSW52
```

/REF52:

see ftp://ftp.unibe.ch/aiub/AIUB_AFTP.TXT

```
wget ftp://ftp.unibe.ch/aiub/BSWUSER52/STA/IGB08_R.CRD
wget ftp://ftp.unibe.ch/aiub/BSWUSER52/STA/IGB08_R.VEL
wget ftp://ftp.unibe.ch/aiub/BSWUSER52/STA/IGB08.FIX
mv IGB08_R.CRD ${D}/REF52
mv IGB08_R.VEL ${D}/REF52
mv IGB08.FIX ${D}/REF52
```

Populating the DATAPOOL area

Let us assume that ${D} points to your DATAPOOL directory:

/VMF1:

see ${X}/DOC/README_VMF.TXT

```
wget http://ggosatm.hg.tuwien.ac.at/DELAY/GRID/VMFG/2011/VMFG_20110725.H00
wget http://ggosatm.hg.tuwien.ac.at/DELAY/GRID/VMFG/2011/VMFG_20110725.H06
wget http://ggosatm.hg.tuwien.ac.at/DELAY/GRID/VMFG/2011/VMFG_20110725.H12
wget http://ggosatm.hg.tuwien.ac.at/DELAY/GRID/VMFG/2011/VMFG_20110725.H18
wget http://ggosatm.hg.tuwien.ac.at/DELAY/GRID/VMFG/2011/VMFG_20110726.H00
gzip VMFG_20110725.H00; mv VMFG_20110725.H00.gz ${D}/VMF1
gzip VMFG_20110725.H06; mv VMFG_20110725.H06.gz ${D}/VMF1
gzip VMFG_20110725.H12; mv VMFG_20110725.H12.gz ${D}/VMF1
gzip VMFG_20110725.H18; mv VMFG_20110725.H18.gz ${D}/VMF1
gzip VMFG_20110726.H00; mv VMFG_20110726.H00.gz ${D}/VMF1
```


Motivation for the SAVEDISK area

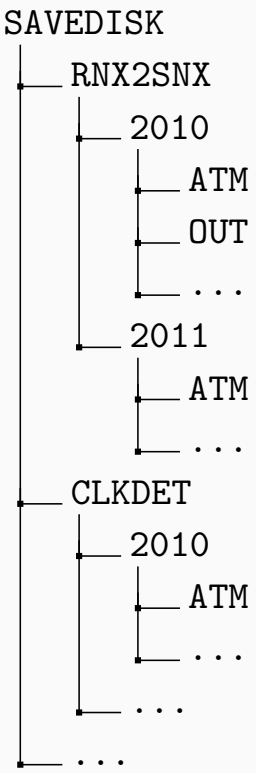
The main result files from the data analysis are collected in the SAVEDISK area. To take them out of the processing campaign area has several advantages:

- When processing GNSS data, a lot of files from various processing steps will populate your campaign directories. Most of them are not needed in a long–term archive.
- If the long–term archive area is separated and only consists of the result files from the processing they can be better protected (backup and so on).
- Because the result files are stored in the SAVEDISK area, you can easily clean up your campaign area without loosing important files. Please keep in mind that the computing performance decreases if you have several thousands of files in a directory.

Structure of the SAVEDISK area

We propose to build subdirectories in the SAVEDISK area for each of your projects. If these projects collect data over several years yearly subdirectories are recommended. It is also practical to use further subdirectories like ATM, ORB, OUT, SOL, STA to distribute the files and to get shorter listings if you are looking for a file.
On the right hand side you find an example how the directory tree may look if two of the example BPEs are assumed as different projects.

(see `${S}/README_SAVEDISK.TXT` for a detailed description)



Realization of the Example Dataflow

Putting data from the DATAPOOL area into the campaign

Each BPE starts with a few copy scripts:

- `RNX_COP`: copies the RINEX files for all selected stations of the session being processed into the campaign.
- `XXX_COP`: calls the BPE-utility (`$BPE/bpe_util.pm`) "copyRef" to copy all necessary files from the DATAPOOL area into the campaign.

Putting data from the campaign into the SAVEDISK area

Each BPE ends with a sequence of scripts:

- `XXX_SUM`: generates the processing protocol
- `XXX_SAV`: puts the result files from the processing CAMPAIGN into the SAVEDISK area
- `XXX_DEL`: cleans up the campaign directory structure

Realization of the Example Dataflow

```
# RNX2SNX.PCF
# ===========
#
PID SCRIPT   OPT_DIR  CAMPAIGN CPU      F WAIT FOR....
3** 8******* 8******* 8******* 8******* 1 3** 3** 3** 3** 3** 3** 3**
#
# Copy required files
# -------------------
001 R2S_COP  R2S_GEN           ANY      1
011 RNX_COP  R2S_GEN           ANY      1 001

...

599 DUMMY    NO_OPT            ANY      1 512 513 514 522
#
# Create summary file and delete files
# ------------------------------------
901 R2S_SUM  R2S_GEN           ANY      1 599
902 R2S_SAV  R2S_GEN           ANY      1 901
903 R2S_DEL  R2S_GEN           ANY      1 902
991 BPE_CLN  R2S_GEN           ANY      1 903
#
# End of BPE
# ----------
999 DUMMY    NO_OPT            ANY      1 991
```


Example for a XXX_COP–script: R2S_COP

```
# Construct list of reference files
# ---------------------------------
  my @filLst = ();

  # Station related input files:
  # Reference frame stations:
  push @filLst,"${dirCrd}${refInf}_R.${extCrd} 1";
  push @filLst,"${dirVel}${refInf}_R.${extVel} 1";
  push @filLst,"${dirCrd}${crdInf}.${extCrd} 1";
  push @filLst,"${dirVel}${crdInf}.${extVel} 1";
  push @filLst,"${dirAbb}${crdInf}.${extAbb} 1";
  push @filLst,"${dirSta}${crdInf}.${extSta} 1";
  push @filLst,"${dirCrx}${crdInf}.${extCrx} 1" if (${crxInf} ne "");
  push @filLst,"${dirBlq}${blqInf}.${extBlq} 1" if (${blqInf} ne "");
  push @filLst,"${dirAtl}${atlInf}.${extAtl} 1" if (${atlInf} ne "");
  push @filLst,"${dirClu}${crdInf}.${extClu} 1";

# Copy Reference Files
# --------------------
  $refDir = "$ENV{D}/" . $refDir;
  copyRef($refDir,@filLst);

# Construct list of external product files
# ----------------------------------------
  @filLst = ();
...
```

Observation selection with RNX_COP

Three opportunities to select stations for processing are implemented (considered in the following order):

1. A file {CAMPAIGN}/STA/{V_OBSSEL}.CLU exists.
   The station names from this file are translated into the four-character-IDs using the abbreviation table. These four-character-IDs are used to copy the corresponding RINEX files from the source directory in the data base.
2. A file {CAMPAIGN}/{V_OBSSEL} contains the RINEX abbreviations of the files to be processed.
3. If V_OBSSEL is empty all RINEX files of the specified session(s) are processed.

The observation files are expected in RINEX format (Hatanaka compressed) in the directory {D}/{V_RNXDIR}/ .

Example for a XXX_SUM–script: R2S_SUM

```
# Append protocol files
# ---------------------
# - Error/warning messages concerning RINEX inconsistencies
  my $sumFil = "${dirOut}RNX${yyssss}.ERR";
  if (-s $sumFil) {
    appFile("PART 0: RINEX INCONSISTENCIES",2,$sumFil,$prcFil,1) }

# - RINEX pseudo-graphics
  $sumFil = "${dirSmc}GRA${yyssss}.${extSmc}";
  appFile("PART 1: RINEX PSEUDO-GRAPHICS",2,$sumFil,$prcFil,1);

  $sumFil = "${dirDel}GRA${yyssss}.${extDel}";
  if (-s $sumFil) {
    appFile("LIST OF REJECTED RINEX DATA FILES",2,$sumFil,$prcFil,1) }

# - Orbit generation summary
  $sumFil = "${dirLst}ORB${yyssss}.${extLst}";
  appFile("PART 2: ORBIT GENERATION SUMMARY",2,$sumFil,$prcFil,1);

# - Single-point-positioning summary
  $sumFil = "${dirOut}SPP${yyssss}.${extOut}";
  appFile("PART 3: SINGLE-POINT-POSITIONING SUMMARY",2,$sumFil,$prcFil,1);

  $sumFil = "${dirDel}SPP${yyssss}.${extDel}";
  if (-s $sumFil) {
    appFile("LIST OF REJECTED BINARY DATA FILES",2,$sumFil,$prcFil,1) }
```

Example for a XXX_SAV–script: R2S_SAV

```
# Some directories
# ----------------
    my $dirSav = "$ENV{S}/" . $result . "/$yyyy/";
    $bpe->setVar('DIRSAV',$dirSav);
    my $outSav = "$dirSav/OUT/";
    my $atmSav = "$dirSav/ATM/";
    my $solSav = "$dirSav/SOL/";
    my $staSav = "$dirSav/STA/";
    check_dir($outSav,$atmSav,$solSav,$staSav) if ( $sav eq "Y" );
...
# Copy files to savedisk
# ----------------------
    my @cpyLst = (
    "${dirOut}R2S${yyssss}.PRC␣␣␣␣␣␣␣␣${outSav}␣f␣",
    "${dirTrp}${e}${yyssss}.${extTrp}␣${atmSav}␣f␣",
    "${dirTro}${e}${yyssss}.${extTro}␣${atmSav}␣f␣",
    "${dirSnx}${e}${yyssss}.${extSnx}␣${solSav}␣z␣",
    "${dirNeq}${e}${yyssss}.${extNeq}␣${solSav}␣z␣",
    "${dirNeq}${f}${yyssss}.${extNeq}␣${solSav}␣z␣",
    "${dirCrd}${e}${yyssss}.${extCrd}␣${staSav}␣f␣");

# Do the distribution
# -------------------
    my $iErr = 0;
    map { $iErr += copy2archive( split("␣",$_),$tit ) } @cpyLst;
    die() if $iErr;            # Stop in case of errors
```


Example for a XXX_DEL–script: R2S_DEL

```
# Delete Files from Previous Run
# ------------------------------
    unlink glob "$dirOut*.J_lk";    # Remove potential "left-over" lock files

# Observation files
    unlink glob("${dirRxn}????${ssss}.${extRxn}");
    unlink glob("${dirRxg}????${ssss}.${extRxg}");
    unlink glob("${dirRxo}????${ssss}.${extRxo}");
    unlink glob("${dirSmt}????${ssss}.${extSmt}");

    unlink glob("${dirPzh}????${ssss}.${extPzh}");
    unlink glob("${dirPzo}????${ssss}.${extPzo}");
    unlink glob("${dirCzh}????${ssss}.${extCzh}");
    unlink glob("${dirCzo}????${ssss}.${extCzo}");
    unlink glob("${dirPsh}????${ssss}.${extPsh}");
    unlink glob("${dirPso}????${ssss}.${extPso}");
    unlink glob("${dirCsh}????${ssss}.${extCsh}");
    unlink glob("${dirCso}????${ssss}.${extCso}");

# Troposphere and Ionosphere
    unlink glob("${dirIon}HOI${yyssss}.${extIon}");
    unlink glob("${dirTrp}${c}${yyssss}.${extTrp}");
    unlink glob("${dirTro}${c}${yyssss}.${extTro}");
    unlink glob("${dirTrp}${e}${yyssss}.${extTrp}");

...
```

Example for a BPE_CLN–script

```
# Get variables
# -------------
  my ($yyssssm30, $ssssm30,
      $taskId,
      $dirBpePrt, $dirBpeLog,
      $extBpePrt, $extBpeLog) =
    $bpe->getKeys(
      '$YSS-(30)', '$S-(30)',
      'TASKID',
      'DIR_BPEPRT', 'DIR_BPELOG',
      'EXT_BPEPRT', 'EXT_BPELOG');

# Delete PRT/LOG and BPE files
# ----------------------------
  my @delLst = ("${dirBpePrt}${taskId}${yyssssm30}_???_???.${extBpePrt}",
                "${dirBpeLog}${taskId}${yyssssm30}_???_???.${extBpeLog}",
                "${dirBpePrt}???${ssssm30}???.BPE");

  map { unlink(glob($_)) } @delLst;
```


Closing Remarks on the Scripts to Realize the Dataflo

- To modify the list of result files or their target directory needs adaptions in the XXX_SAV–script.
- If you have another structure in your datapool you need to revise the source directories in the XXX_COP–script.
- The scripts shall be as generic as possible, it means in particular use variables where all variables are used in the BPE processing.
- The deletion scripts are very important and must be kept consistent with the processing part. No left–overs from the processing are allowed for a permanently running process!
- To finish a BPE with a DUMMY script at PID 999 is beneficial to check whether the processing has successfully finished.

Consequences from the Example Dataflow

The full intended dataflow is now realized in the processing examples of Version 5.2 .

The consequence is that a few additional tools need to be installed before the processing examples can be started:

- extracting UNIX-compressed files
- RINEX Hatanaka compression tools

Potential sources are listed at `http://www.bernese.unibe.ch/download/#TLS` .

Our assumption is that these (or at least equivalent) tools are needed anyhow if you want to process GNSS data. This means that this does not imply any additional work load to the users.

The Processing Examples in Version 5.2

The Processing Examples in Version 5.2
The Examples BPEs
Characteristics of the Examples BPEs

The Processing Examples in Version 5.2

The Bernese GNSS Software, Version 5.2 is distributed with several ready–to–use example BPEs:

`PPP_BAS.PCF`	Standard PPP for coordinate, troposphere, and receiver clock determination based only on GPS data or a combined GPS/GLONASS solution
`PPP_DEMO.PCF`	PPP containing several extended processing examples, like pseudo-kinematic, high-rate troposphere, or ionosphere solutions
`RNX2SNX.PCF`	Standard double-difference network solution based only on GPS data or a combined GPS/GLONASS solution with an extended ambiguity resolution scheme
`BASTST.PCF`	Baseline by baseline processing for trouble shooting.
`CLKDET.PCF`	Zero-difference network solution based only on GPS data or a combined GPS/GLONASS solution providing clock corrections (e.g., w.r.t. an existing coordinate and troposphere solution)
`LEOPOD.PCF`	Precise Orbit Determination for a Low Earth Orbiting Satellites based on on-board GPS-measurements (e.g., for GRACE)
`SLRVAL.PCF`	Validation of an existing GNSS or LEO orbit using SLR measurements

Characteristics of the Examples BPEs

- The Version 5.0 contained the first time a set of BPEs as processing examples. They were intended to demonstrate the users how a BPE can look like and should give a starting point for the own development of BPEs.
- The practice has shown that most of the users have used these BPEs as a ready–to–use processing scheme.

→ The new examples for Version 5.2 shall be designed as "ready–to–use" processing examples.

  - The user scripts must not be a demonstrator on how to develop good user scripts but they should become able to handle all relevant problems.

What Applications Shall be Covered?

Analysing the questions from users and the current developments, we have identified the following wish list:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Processing Example: PPP_BAS.PCF

Precise point positioning (PPP) limited to an optimized coordinate estimation:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Processing Example: PPP_DEMO.PCF

Extended Precise point positioning (PPP) including several special applications:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Processing Example: RNX2SNX.PCF

Double–difference standard network solution for coordinate and troposphere estimation with ambiguity resolution:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Processing Example: CLKDET.PCF

Zero–difference standard network solution for receiver and satellite clock determination:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Processing Example: LEOPOD.PCF

Precise orbit determination (POD) for a Low Earth Orbiter (LEO) mission based on GPS data:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Processing Example: SLRVAL.PCF

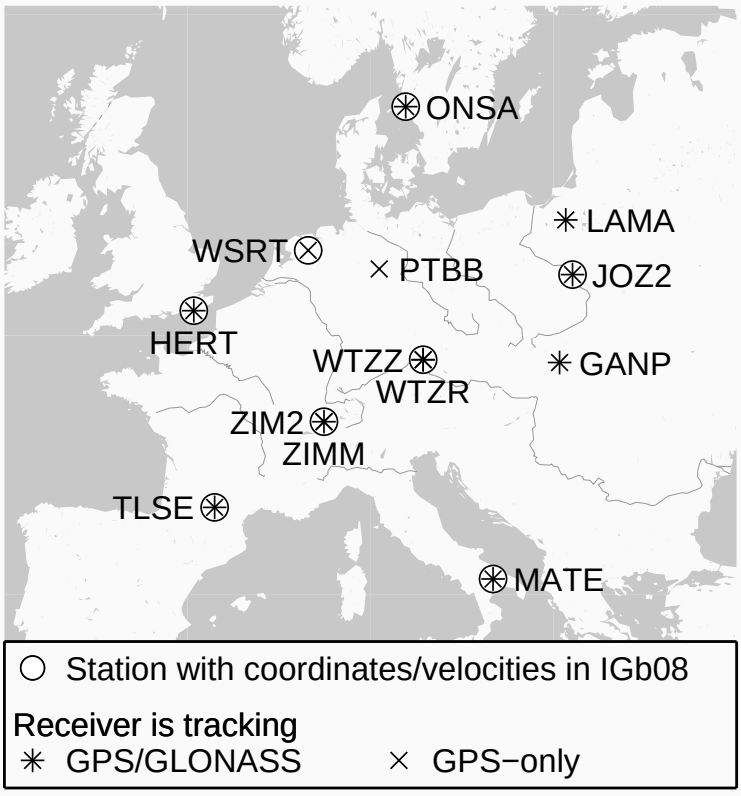
Validate an existing orbit of GNSS or LEO satellites using independent SLR measurements:

- allow the combined GPS/GLONASS processing
- full ambiguity resolution scheme with all steps for GPS and so far supported also for GLONASS
- optimized preprocessing of the zero–difference examples
- cover more applications by additional example BPEs (LEO orbit determination; SLR processing to validate orbits)
- for demonstration purposes equipment changes and a repositioning event shall be included
- kinematic or at least pseudo–kinematic processing
- high–rate troposphere estimation
- ionosphere model determination
- prepared for processing many stations (clustering)
- support hourly (sliding window) and daily processing as well as reprocessing (daily independent) schemes

Example Dataset

Example Dataset

Example Dataset



Stations
13 total
10 GPS+GLONASS
3 GPS–only
2 pairs forming short baselines

Datum definition
8 coordinates in IGb08

Antenna calibrations
3 individually calibrated
9 separate GPS/GLONASS calibrations
11 `ROBOT` calibrations
1 calibration `ADOPTED FROM NONE`

7 high performance clock

Selected days:

1:	2010–Jul–26	Day of year: 2010/207	GPS–week: 1594–1
2:	2010–Jul–27	Day of year: 2010/208	GPS–week: 1594–2
3:	2011–Jul–24	Day of year: 2011/205	GPS–week: 1646–0
4:	2011–Jul–25	Day of year: 2011/206	GPS–week: 1646–1

Criteria for selecting the four days:

- Two times two days with about one year inbetween to allow velocity estimation.
- The full availability of the observation files of the selected stations.
- Repositioning event for satellite G25 for day 208 of year 2010 (day 2).
- Optimal availability for SLR tracking data to GNSS and GRACE (used in the `LEOPOD.PCF` example) for `SLRVAL.PCF`.

Example BPE: General Aspects

Example BPE: General Aspects
- README–Files on the Example BPEs
- Standard–Variables for the Dataflow
- Defining Standard Processing Files
- Defining the Reference Frame
- Set of Project–Specific Files
- Selecting Orbit/Clock Products
- Define Basic Modeling Aspects
- Hourly Processing
- Limitation of the BPE Variables

README–Files on the Example BPEs

Each example BPEs is accomplished by an extensive README file:

- explaining the main purpose
- providing a detailed description on the realization of the purpose,
- showing where to find the key quality indicators for the results and giving some ideas about potential sources of problems
- listing of the BPE server variables in the BPE example PCF with a short explanation on their usage
- listing of the necessary input files for the processing and where they are expected in the DATAPOOL area
- listing of the result files and where they are located in the SAVEDISK area with the current configuration

Standard–Variables for the Dataflow

BPE–variables for the dataflow (DATAPOOL→CAMPAIGN52):
(all BPEs)

- V_RNXDIR (default: "RINEX" or "HOURLY")
  Directory in the ${D} (DATAPOOL) area where the RINEX files are expected.

- V_OBSSEL (select according to your needs)
  This variable may contain a filename for a cluster file or a file with the RINEX station abbreviations.

Standard–Variables for the Dataflow

BPE–variables for the dataflow (DATAPOOL→CAMPAIGN52):
(all BPEs)

- V_REFDIR (default: "REF52")
  Directory in the ${D} (DATAPOOL) area where the basic Bernese files for the processing are located.

- V_REFINF (default: "IGB08")
  Reference frame files (CRD and VEL), used as master files when updating coordinate and velocity files.

- V_CRDINF (any project name, e.g., "EXAMPLE")
  The basename of the files used as basic Bernese input files for this example BPE.

Standard–Variables for the Dataflow

BPE–variables for the dataflow (CAMPAIGN52→SAVEDISK):
(all BPEs)

- V_RESULT (any project name, e.g., "EPNFINAL")
  Directory in the ${S} (SAVEDISK) area where the result files from the BPE processing are collected in yearly directories.

- V_SAV (default: "Y")
  Save result files into the ${S}/{V_RESULT} area if V_SAV=='Y'. This variable is managed by the BPE user script XXX_SAV.
  Typically the results are copied in the product archive. For test purposes it might be beneficial if test result files do not overwrite the product files in the long-term archive.

Standard–Variables for the Dataflow

BPE–variables for the dataflow (CAMPAIGN52→DATAPOOL):
(PPP_BAS.PCF, PPP_DEMO.PCF)

- V_UPD (default: "Y")
  Update the files in the ${D}/{V_REFDIR} area with the results from the BPE process (only in the PPP–examples).

BPE–variables for the dataflow (cleaning CAMPAIGN52):
(all BPEs)

- V_DEL (default: "Y")
  Delete all files from the currently processed session from the campaign if V_DEL=='Y'; managed in XXX_DEL user script.
  Usually the deletion should be enabled to keep the campaign clean. For debugging purposes and to identify sources for errors in the processing it may be useful to keep all files.

Defining Standard Processing Files

BPE–variables defining "General Input Files":
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, BASTST.PCF, CLKDET.PCF, LEOPOD, and SLRVAL.PCF)

- V_PCV (default: "I08")
  Selection of the antenna phase center model identifier.

- V_SATINF (default: "SATELLIT")
  Basename of the satellite information file. The resulting filename is ${X}/GEN/{V_SATINF}.{V_PCV}.

- V_PCVINF (default: "PCV")
  Basename of the antenna phase center correction file. The resulting filename is ${X}/GEN/{V_PCVINF}.{V_PCV}.

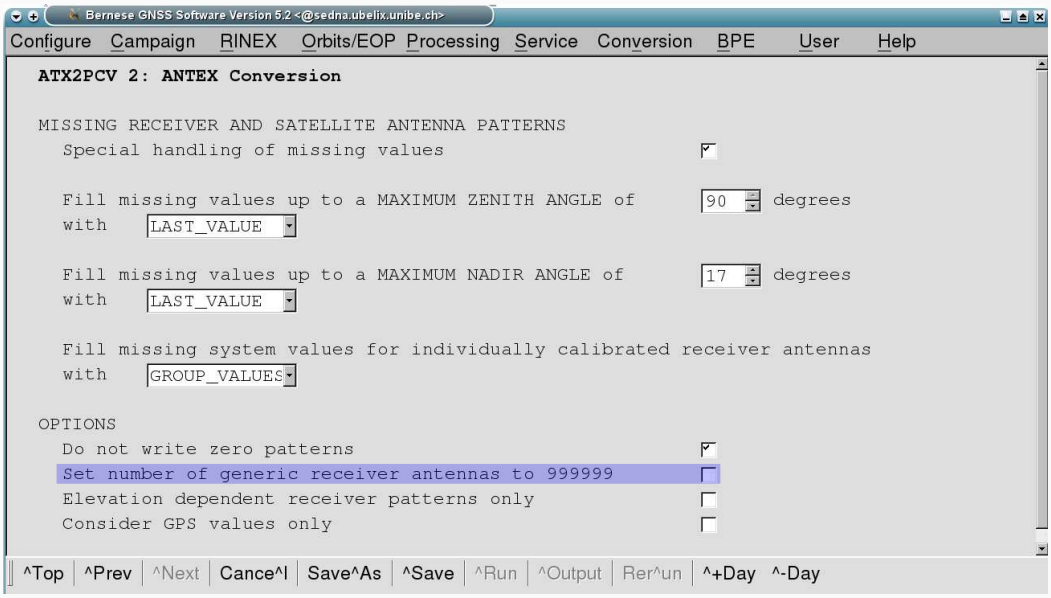
- V_SATCRX (default: "SAT_$Y+0")
  Name of the satellite problem file.

Exchange/Complete ATX Corrections

Updating the antenna phase center correction file:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, and CLKDET.PCF)

- V_MYATX (default: empty)
  Filename (including extension) for an ANTEX file containing receiver and possibly satellite antenna phase center corrections (expected in ${X}/GEN). The file is used to update the Bernese phase center file in ${X}/GEN/{V_PCVINF}.{V_PCV}, e.g., in case of new antennas in the network.
  Note: if you plan to use a file containing individually calibrated antennas you need to change the settings for "Set number of generic receiver antennas to 999999" in panel "ATX2PCV 2".
  Please remind that calibration values for all antennas with number different from 999999 in the station information file must be available with individual calibrations.

Defining Standard Processing Files



- Check this box to switch from type–mean to individual calibration.

Defining the Reference Frame

BPE–variable to define the reference frame:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, and CLKDET.PCF)

- V_REFDIR (default: "REF52")
  Directory in the ${D} (DATAPOOL) area where the coordinate and velocity files for the reference frame are located.

- V_REFINF (default: "IGB08")
  Reference frame files, used as master files when updating coordinate and velocity files. The corresponding files are expected as
  - ${D}/{V_REFDIR}/(V_REFINF)_R.CRD
  - ${D}/{V_REFDIR}/(V_REFINF)_R.VEL

- V_REFEPO, only PPP–examples (default: "2005 01 01")
  The epoch of the coordinates in the reference frame coordinate file (see above) needs to be specified here. In case of IGSb08 it is "2005 01 01".

Defining the Reference Frame

BPE–variable to define the reference frame:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, and CLKDET.PCF)

The related sequence in the BPE is:

```
#
# Copy required files
# -------------------
...
003 COOVEL    R2S_GEN            ANY       1 001
004 COOVEL    R2S_GE2            ANY       1 001
005 CRDMERGE  R2S_GEN            ANY       1 003 004
...
```

- COOVEL propagates the coordinates from the reference frame coordinate file (PID 004) and the project–specific coordinate file (PID 003, see below) to the processed session.
- CRDMERGE (PID 005) takes all stations from the reference frame coordinate file and adds all additional stations from the project–specific coordinate file.

Set of Project–Specific Files

BPE–variables for project–specific station–related files:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, BASTST.PCF, CLKDET.PCF, and SLRVAL.PCF)

- V_REFDIR (default: "REF52")
  Directory in the ${D} (DATAPOOL) area where the project–related files are located.
- V_CRDINF (default: "EXAMPLE")
  The basename of the station–related files for the particular project used for the BPE. Typically these are:
  - ${D}/{V_REFDIR}/(V_CRDINF).CRD
  - ${D}/{V_REFDIR}/(V_CRDINF).VEL
  - ${D}/{V_REFDIR}/(V_CRDINF).STA
    (for PPP the separate variable V_STAINF is used)

Set of Project–Specific Files

BPE–variables for project–specific station–related files:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, CLKDET.PCF, and SLRVAL.PCF)

- V_STAINF, only PPP–examples (default: "EXAMPLE")
  Usually the header information in the RINEX observation files is compared to the expected equipment information provided in the station information file. The basename of this file is given in this variable: ${D}/{V_REFDIR}/(V_STAINF).STA.
  For a "quick-look" PPP this variable may be empty to skip the check of the RINEX header information.

- V_CRXINF (default: empty)
  Inconsistencies between the RINEX observation file headers and the station information file can be recorded in a file with "Accepted station information" inconsistencies in program RXOBV3. If you need such a file for your processing, you can specify its name here. It will be copied from ${D}/REF52/${V_CRXINF}.CRX

Set of Project–Specific Files

BPE–variables for project–specific station–related files:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, BASTST.PCF, CLKDET.PCF, and SLRVAL.PCF)

- V_BLQINF (default: "EXAMPLE")
  The basename of the file containing the ocean tidal loading corrections for the stations and center-of-mass corrections. The file ${D}/REF52/${V_BLQINF}.BLQ will be copied into your campaign for processing.
  If the variable is empty, no displacement corrections to the stations or CMC for the orbit generation are applied.

- V_ATLINF (default: "EXAMPLE")
  Same as V_BLQINF but for atmospheric tidal loading. The filename is ${D}/REF52/${V_BLQINF}.ATL.

Selecting Orbit/Clock Products

BPE–variable to select orbit/clock products:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, CLKDET.PCF, and LEOPOD.PCF)

- V_B (default: "COD")
  Name of the external product for GNSS-orbit, ERP and satellite clock information. It is expected that the products are available in the ${D} (DATAPOOL) area in a specific subdirectory V_B:
  - ${D}/{V_B}/{V_B}wwwwd.PRE
  - ${D}/{V_B}/{V_B}wwwwd.CLK (not for RNX2SNX.PCF)
  - ${D}/{V_B}/{V_B}wwww7.ERP

  Remark: Alternatively you may also use the products from the IGS or any other IGS analysis center or even from other sources if they are precise enough. From our point of view, there is a clear preference for CODE products ("COD", arguments in the main presentation).

  If you want to process GLONASS data with IGS products you must prepare two orbit files: ${D}/IGS/IGSwwwwd.PRE and ${D}/IGS/IGLwwwwd.PRE.

Define Basic Modeling Aspects

BPE–variables to define basic processing settings:
(PPP_BAS.PCF, PPP_DEMO.PCF, RNX2SNX.PCF, BASTST.PCF, and CLKDET.PCF)

- V_HOIFIL (default: "HOI$YSS+0")
  Specify the filename used for the ionosphere model within your campaign area that is used to compute the higher-order ionosphere (HOI) corrections. The ionosphere model is copied from ${D}/BSW52/CODwwwwd.ION.
  If the variable is empty, the HOI corrections are disabled.

- V_SATSYS (default: "GPS/GLO")
  Select the system(s) (usually "GPS" or "GPS/GLO") to be used for the BPE processing.

- V_SAMPL, also for LEOPOD.PCF (default: "300")
  Sampling interval for introducing the observations from RINEX to Bernese observation file format.

Hourly Processing

Switch from daily to hourly processing:
(PPP_BAS.PCF, PPP_DEMO.PCF, and RNX2SNX.PCF)

The basic switch between daily and hourly processing is done by selecting the corresponding "Session table" ("Menu>Configure>Set session/compute date"). Two BPE variables are specifically relevant to change between daily and hourly processing scheme:

- V_HOURLY (default: "8")
  In case of hourly session definition, a number of hours can be specified that are prepended to stabilize the estimation of ambiguity parameters. It should not be much shorter than the length of a satellite path, e.g., 6 hours. It is obsolete if daily session definition is given.

- V_RNXDIR (default: "HOURLY")
  If you change from daily to hourly processing you should not forget to select the directory with hourly instead of daily RINEX files.

Limitation of the BPE Variables

The BPE variables can be used to take a lot of main settings out from the individual programs and manage them on a central location.

This principle is convenient to the users but has also some limits:

- The numer of BPE variables was dramatically increased w.r.t. the examples in Version 5.0 . A further significant increase to cover more options and switches in this way seems not be reasonable without confusing the users.

- From the technical point of view such variables can cover options that can be managed by directly putting the variables into the program input panels in the BPE option directories.
  A prominent example, what cannot be handled in this way is the change of the troposphere model. To support the users with this change, we have established a description in the FAQ–section of the Bernese Homepage:
  http://www.bernese.unibe.ch/faq/#exm001

Example BPE: Specific Aspects

Example BPE: Specific Aspects
- PPP–Examples
- RNX2SNX–Example
- BASTST–Example
- CLKDET–Example
- LEOPOD–Example
- SLRVAL–Example

PPP–Examples

PCFile name: `${U}/PCF/PPP_BAS.PCF`
Description: `${U}/PCF/PPP_BAS.README`

PRECISE POINT POSITIONING: basic version

Purpose:
This BPE performs an efficient station-wise processing (Precise Point Positioning, PPP). The following results are obtained:

- computation of station coordinates (on the cm-level accuracy), to get, e.g., corresponding a priori information for interferometric analysis (see RNX2SNX PCF),
- estimation of station-specific troposphere parameters,
- determination of phase-consistent receiver clock offsets typically at 5–minute intervals (for time transfer).

PPP–Examples

PCFile name: `${U}/PCF/PPP_DEMO.PCF`
Description: `${U}/PCF/PPP_DEMO.README`

PRECISE POINT POSITIONING: extended version

Purpose:
As the basic version this PPP-BPE:

- computes coordinates, troposphere and receiver clocks
- updates the coordinate/velocity files

It may also compute:

- geocenter coordinates from all datum stations included in the PPP procedure,
- pseudo-kinematic coordinates,
- high-rate troposphere parameters (15 minutes sampling), and
- station-specific/regional ionosphere models together with the corresponding DCB (P1-P2) parameters.

PPP–Examples

PCFile name: `${U}/PCF/PPP_DEMO.PCF`
Description: `${U}/PCF/PPP_DEMO.README`

PRECISE POINT POSITIONING: extended version

Purpose: (cont.)
Furthermore this BPE demonstrates the two opportunities to pre-process phase data for zero-difference processing:

- RNXSMT: consistency check with the code measurements based on Melbourne-Wübbena linear combination (requires dual frequency data)
- MAUPRP: a sophisticated interpretation of epoch-difference residuals based on PPP (requires precise satellite clock corrections with the same sampling as that of the data)

BPE–specific Variables (PPP)

BPE–variable to select the observation type:
(PPP_BAS.PCF, PPP_DEMO.PCF)

- V_OBSTYP (default: "BOTH")
  You may select whether only the "CODE", only the "PHASE", or "BOTH" measurement types together shall be used for the PPP procedure.

BPE–specific Variables (PPP)

BPE–variable to select the zero–difference preprocessing:
(PPP_DEMO.PCF and CLKDET.PCF)

- V_HRCLK (default: "NO")
  You may select how the phase measurements will be pre-processed. You have two options, depending on the availability of satellite clock corrections:
  1. "NO" checks only the consistency between the code and phase observations which limits the level of screening to the noise level of the code data.
  2. "YES" uses a more involved algorithm but requires high-rate (30 sec) and precise satellite clock corrections

BPE–specific Variables (PPP)

BPE–variable to select solutions:
(PPP_DEMO.PCF)
The additional results of the extended PPP version may be enabled/disabled by specifying a non-blank/blank solution identifier for the following PCF variables:

V_G	default:	"GCC"	Determination of translations
V_H	default:	"KIN"	Pseudo-kinematic solution
V_I	default:	"TRP"	High-rate troposphere solution
V_K	default:	"ION"	Station-specific ION/DCB results

RNX2SNX–Example

PCFile name: ${U}/PCF/RNX2SNX.PCF
Description: ${U}/PCF/RNX2SNX.README

RINEX–TO–SINEX

Purpose:
RINEX-TO-SINEX (RNX2SNX): standard double-difference processing for regional networks for static, dual-frequency stations.

The purpose of this BPE is to compute coordinates and troposphere parameters for all included stations in a given reference frame using the translation conditions from a minimum constraint solution. In addition, normal equation files are generated.

BPE–specific Variables (RNX2SNX)

BPE–variables for clustering of parallel BPE jobs:
(RNX2SNX.PCF)

- V_CLU (default: depends on your computer system)
  Specifies how many files (stations/baselines) are processed together in one of the parallel running BPE jobs.

- V_CLUEDT (default: depends on your computer system)
  For the residual screening step (GPSEDT_P) a separate maximum number of baseline files can be specified that are analyzed in one cluster.
  If the variable V_CLUEDT is not specified or is empty, the number from V_CLU is also used in this step.

- V_CLUFIN (default: "Y")
  The baselines created in SNGDIF are assigned to clusters. They are processed in these clusters with correct correlations (script GPSCLU_P). With changing this variable to "N" no clustering is done.

BPE–specific Variables (RNX2SNX)

BPE–variables to manage the ambiguity resolution:
(RNX2SNX.PCF)

- V_GNSSAR (default: "ALL")
  The content of this variable is filled into the selection of systems for the ambiguity resolution (panel "GPSEST 3.2"). It may either contain a single GNSS ("GPS" or "GLONASS") to allow the ambiguity resolution only for this particular system or it can be "ALL" to resolve all ambiguities.
  If the variable is empty, the value in the input file is kept.

BPE–specific Variables (RNX2SNX)

BPE–variables to manage the ambiguity resolution:
(RNX2SNX.PCF)

- V_BL_AMB (default: "6000")
  The variable contains the maximum length of a baseline included in the Melbourne-Wübbena / Narrow-Lane ambiguity resolution scheme.
- V_BL_QIF (default: "2000")
  The variable contains the maximum length of a baseline included in the Quasi-Ionosphere-Free (QIF) ambiguity resolution algorithm that is carried out on the ambiguities that are not resolved in the first step yet.
- V_BL_L53 (default: "200")
  The variable contains the maximum length of a baseline included in the phase-based wide-/narrow-lane ambiguity resolution scheme.
- V_BL_L12 (default: "20")
  The variable contains the maximum length of a baseline included in the direct L1/L2 ambiguity resolution scheme.

BASTST–Example

PCFile name: ${U}/PCF/BASTST.PCF
Description: ${U}/PCF/BASTST.README

BASELINE TEST

Purpose:
Does GPSEST runs baseline by baseline for trouble shooting.

All baselines for a specific session in the campaign are processed independently. The residuals are stored together with some result files. In the subsequent step, the residuals are analyzed and some statistics are provided.

BASTST–Example

PCFile name: `${U}/PCF/BASTST.PCF`
Description: `${U}/PCF/BASTST.README`

BASELINE TEST

Specific BPE–variables:

- `V_APRFIL` (default: "APR$YSS+0")
  Name of the file with the a priori coordinates. It has to be specified as it is used in the program input panel (with session as variable).

- `V_ORBFIL` (default: "COD$YSS+0")
  The name of the files containing respectively the standard orbits and the ERP information must be given here. The filename should use the menu time variables.

CLKDET–Example

PCFile name: `${U}/PCF/CLKDET.PCF`
Description: `${U}/PCF/CLKDET.README`

CLOCK DETERMINATION

Purpose:
CLOCK DETERMINATION (CLKDET): standard zero difference processing for regional networks for dual-frequency stations starting from broadcast clocks.

The purpose of this BPE is to compute station and satellite clock parameters for all included stations and satellites. The results are provided in clock RINEX format.

Coordinates and troposphere corrections (e.g., from a double difference solution, see RNX2SNX.PCF) may be introduced and a consistent set of clock corrections will be obtained.

BPE–specific Variables (CLKDET)

BPE–variables for clustering of parallel BPE jobs:
(CLKDET.PCF)

- V_CLU (default: depends on your computer system)
  Specifies how many files (stations/baselines) are processed together in one of the parallel running BPE jobs.

- V_CLUCOD (default: depends on your computer system)
  Number of clusters for pre-processing the code measurements in regional clusters for residual screening.

- V_CLUPRP (default: depends on your computer system)
  The number of clusters for steps (MKCLUSAP/TIMEST_P with options CLK_ED0 and CLK_EDT).

BPE–specific Variables (CLKDET)

BPE–variables for clustering of parallel BPE jobs:
(CLKDET.PCF)

- V_CLUSOL (default: "1")
  The final solution (MKCLUSAP/TIMEST_P with options CLK_RES) may also be computed in clusters if many stations are included.

- V_MAXSOL (default: depends on your application)
  For the final solution a maximum number of stations per cluster may be specified. This option limits the number of stations in the final solution in conjunction with the number of clusters.

BPE–specific Variables (CLKDET)

Introduce a CRD/TRP solution:
(CLKDET.PCF) It is possible that the results from an external solution may be introduced to compute a clock solution consistent to those results:

- V_FIXPTH (default: "${S}/RNX2SNX/$Y+0")
  Path from where the external coordinate/troposphere solution files are copied.

- V_FIXCRD (default: "F1_$YSS+0")
  Name of the coordinate solution file to be introduced:
  {V_FIXPTH}/STA/{V_FIXCRD}.CRD

- V_FIXTRP (default: "F1_$YSS+0")
  Name of the troposphere solution file to be introduced:
  {V_FIXPTH}/ATM/{V_FIXTRP}.TRP

LEOPOD–Example

PCFile name: ${U}/PCF/LEOPOD.PCF
Description: ${U}/PCF/LEOPOD.README

PRECISE ORBIT DETERMINATION FOR
LOW EARTH ORBITER

Purpose:
This BPE performs reduced-dynamic and kinematic orbit determinations for one LEO. The following results are obtained:

- screened carrier phase observation files
- reduced-dynamic orbit solution with piece-wise constant accelerations as empirical parameters
- kinematic orbit solution
- orbit comparison between both orbit solutions

SLRVAL–Example

PCFile name: `${U}/PCF/SLRVAL.PCF`
Description: `${U}/PCF/SLRVAL.README`

ORBIT VALIDATION USING SLR DATA

Purpose:
This BPE performs an orbit validation for GNSS or LEO orbit files using SLR tracking data.

Configure Example to Run User Data

Configure Example to Run User Data
- Preparing the Campaign
- Filling the DATAPOOL
- Generate Project Related Files

Preparing the Campaign

You have to start with some general steps:

1. Create the campaign
2. Depending on the planned processing scheme you should copy either
   `${X}/SKL/SESSIONS.SES` to `STA/SESSIONS.SES` for daily processing, or
   `${X}/SKL/HOURLY.SES` to `STA/HOURLY.SES` for hourly processing, or
   and activate the session table in "Menu>Configure>Set session/compute date"

Filling the DATAPOOL

Put the needed data into the DATAPOOL–area (`${D}`):

1. Copy your RINEX observation files into a specific subdirectory of the DATAPOOL (e.g., "myRINEX"). They may be UNIX-compressed files in the Hatanaka compact format.
   (Set the BPE–variable "`V_RNXDIR`" to the subdirectory name, if you start one of the example BPEs on these data)
2. Provide GNSS satellite orbits, clock corrections and ERP information
   For instance from CODE:
   `${D}/COD/CODwwwwd.EPH.Z` from `ftp://ftp.unibe.ch/aiub/CODE/yyyy`
   `${D}/COD/CODwwwwd.CLK.Z` from `ftp://ftp.unibe.ch/aiub/CODE/yyyy`
   `${D}/COD/CODwwww7.ERP.Z` from `ftp://ftp.unibe.ch/aiub/CODE/yyyy`
   Set the BPE–variable "`V_B`" to the product identifier (in this case "`COD`"), if you start one of the example BPEs on this products

Filling the DATAPOOL

Put the needed data into the DATAPOOL–area (${D}):

3. Ionosphere models for ambiguity resolution and Higher–order ionosphere corrections

   ${D}/BSW52/CODwwwwd.ION.Z from ftp://ftp.unibe.ch/aiub/CODE/yyyy

4. Differential code biases (DCB)

   ${D}/BSW52/P1P2yymm.DCB.Z for ionosphere determination
   ${D}/BSW52/P1C1yymm.DCB.Z for clock estimation
   from ftp://ftp.unibe.ch/aiub/CODE/yyyy

5. VMF1–coeffcients (if needed)

   ${D}/VMF1/VMFG_yyyymmdd.Hhh.gz
   from http://ggosatm.hg.tuwien.ac.at/DELAY/GRID/VMFG/yyyy

Filling the DATAPOOL

Put the needed data into the DATAPOOL–area (${D}):

6. Reference frame files:

   ${D}/REF52/IGB08_R.CRD
   ${D}/REF52/IGB08_R.VEL
   ${D}/REF52/IGB08.FIX
   from ftp://ftp.unibe.ch/aiub/BSWUSER52/STA

   Set the BPE–variable "V_REFINF" to the basename of the reference frame file (in this case "IGB08", if you start one of the example BPEs on these reference frame files)

Filling the DATAPOOL

Put the needed data into the DATAPOOL–area (`${D}`):

6. Reference frame files:
   Alternatively you may also extract the reference frame files from the `${D}/STAT_LOG/IGB08.snx` file:
   1. Copy the IGB08.snx file into the SOL directory of your campaign
   2. Run SNX2NQ0 ("Menu>Conversion>SINEX to normal equations") to extract `IGB08.CRD/VEL/FIX` from `IGB08.SNX`
      (if you select an epoch, take care on the end-intervals in the SINEX file)
   3. Run CRDMERGE (Menu>Service>Coordinate tools>Merge coordinate/velocity files) to copy the coordinates/velocities to all stations with the same DOMES number (if needed)
   4. Make the new reference files available in the `${D}/REF52` directory as: `IGB08_R.CRD`, `IGB08_R.VEL`, and `IGB08.FIX`

Generate Project Related Files

A set of project related files shall be provided in `${D}/REF52`:

1. Station information file (EXAMPLE.STA):
   - Generate it in a standard way (manually starting from RINEX, from SINEX, ...)
   - If you have a maintained station information file from Version 5.0, you may use the program STA2STA ("Menu>Service>Station information files>Convert station information files") to convert it to the format for Version 5.2.

2. Accepted RINEX header inconsistencies
   (EXAMPLE.CRX, if needed):
   Same procedure as for the station information file.

Generate Project Related Files

A set of project related files shall be provided in `${D}/REF52`:

3. Update the "Antenna phase eccenter" information (PCV.I08):
   a) copy the ANTEX file (e.g., IGS08.ATX) to OUT–directory of your campaign
   b) run the program ATX2PCV ("Menu>Conversion>ANTEX to Bernese format"): use the standard PCV.I08 as Bernese and IGS08.ATX as external input file.
      Take care on the question about individual or group calibration in panel! "ATX2PCV 2"
   c) compare the resulting Bernese formatted Antenna phase eccenter file in the OUT–directory of your campaign with the master file `${X}/GEN/PCV.I08`
   d) update the master file in `${X}/GEN`

   Note, for other project also other files might be needed.

Generate Project Related Files

A set of project related files shall be provided in `${D}/REF52`:

4. Tectonic plate assignment file (EXAMPLE.PLD):
   must be generated manually or with the support of the menu.
   A maintained file from a Version 5.0 campaign can be reused.

Generate Project Related Files

A set of project related files shall be provided in `${D}/REF52`:

5. Ocean tidal loading table EXAMPLE.BLQ:
   - If you have already coordinates for your all stations you can submit a request to `http://froste.oso.chalmers.se/loading`. Otherwise you have to run a "PPP_BAS.PCF" first without ocean tidal loading corrections.
   - If you have a maintained file from a Version 5.0 campaign, you have to add the CMC–section in the header at least. Nevertheless, there are two reasons to submit a request to Chalmers:
     - The loading corrections should refer to the latest ocean tidal models (e.g., FES2004).
     - To compute the corrections with the latest version of the program at Chalmers.
   - Further information can be found in `${X}/DOC/README_BLQ.TXT`.

Generate Project Related Files

A set of project related files shall be provided in `${D}/REF52`:

6. Atmospheric tidal loading table EXAMPLE.ATL:
   - Use the program GRDS1S2 ("Menu>Service>Coordinate tools >Extract atmospheric tidal loading coefficients"). Also for this process you need a complete list of station coordinates in a Bernese formatted coordinate file (see above).
   - Further information can be found in `${X}/DOC/README_ATL.TXT`.

Now you are ready to run the example BPEs from Version 5.2 on your own data.