G-Nut/Anubis

Version 3 (4)

User manual

Jan Douša and Pavel Václavovic
email: support@gnutsoftware.com

Contents

1. Introduction

1.1   Purpose

G-Nut/Anubis [1] is a software designed for file header checking and quantitative, qualitative and complex control (QC) of data collected by one or more receivers of Global Navigation Satellite Systems (GNSS) and their regional augmentations.

The software provides the following data characteristics:

- Statistics of observations, signals, frequency bands, tracking types, satellites, etc..
- Standard point positioning of standalone GPS, GLONASS, Galileo and BeiDou systems.
- Pre-processing – detecting receiver clock jumps and cycle slips and other interruptions.
- Satellite visibility above the horizon and a user elevation angle mask cut-off.
- Satellite elevations and azimuths, tracking bands etc..
- Satellite navigation data availability and satellite health status.
- Expected counts of observations, elevation-dependent statistics.
- Presence of data gaps and small pieces.
- Code pseudorange multipath and signal noise.
- Phase signal-to-noise density.
- Receiver clock corrections.
- Dilution of precision (DOPs).
- Horizontal and vertical positioning indicators.
- Header metadata and format validation.
- Metadata editing and data filtering.

The software supports these input formats (including CRX, gz and Z compression):

- Data files: RINEX 2/3 and BNCOBS (observation data), RINEX 2/3 (navigation data), SP3 (precise ephemeris), SINEX (metadata).
- Data streams: RTCM 3 (observation and navigation data), BNCOBS (observation data).

The software supports these output formats (including CRX, gz and Z compression):

- Data files: RINEX 2/3 and BNCOBS (observation data), RINEX 2/3 (navigation data), SP3 (precise ephemeris), SINEX (metadata).
- Data streams: RTCM 3 (observation and navigation data), BNCOBS (observation data).
- QC outputs: native Anubis' QC data format (XTR), satellite- and epoch-specific data (JSON), and XML (QC summary).

The software can be executed from the console user interface (CUI). It can be configured using an XML file, from a command-line arguments, or both. It has no graphic user interface (GUI).

The software is currently available for the following operating systems: Linux, Windows and MacOS (experimental).

1.2   Versions

The G-Nut/Anubis is being released in three different software versions:

Anubis Free

is an open-source version with a fundamental functionality:

- Observation statistics (signals, bands, satellites, etc.).
- Data availability (data gaps and small pieces).
- Pre-processing of carrier phases (cycle slips, clock jumps).
- Satellite information (azimuth/elevations).
- Single point positioning (repeatability, GDOP).
- Code multipath and phase signal-to-noise ratio.
- Navigation data, information on satellite health status.
- Header and format validation.

Anubis Pro

is a commercial version with advanced functionality. Compared to the Free version, it brings additional benefits:

- Support for compressed files (Hatanaka + gzip/Z) reading/writing from/to a repository.
- Metadata and data editing, filtering, concatenation, and splitting.
- Navigation and observation data file encoding (RINEX 3/2).
- XML/SINEX input for modifying/validating site metadata.
- JSON output format for an easy plotting and the G-Nut online visualisation service.
- Enhanced configuration/help, predefined setting for specific processing modes.
- Satellite health status used in positioning, observation statistics, KPIs etc.
- Configuration for parallel processing (CPU, RAM, sites, files).

Anubis RT

is a commercial version with advanced and real-time functionality. Compared to the Pro version, it brings additional benefits:

- Real-time data QC monitoring (format, content, quality, and latency).
- Stream observation data format converter (streams from/to RTCM/BNC).
- Stream product data format converter (streams from/to RTCM-SSR/IGS-SSR/BNC).
- Stream-to-data converter (files from/to RINEX/BNC).
- RTCM and BNC format decoder/encoder.
- TCP/NTRIP client.

*Important note: individual functionalities are not always explicitly distinguished within this manual. General use of **Anubis** term in this document refers to any version relevant for the requested operation.*

1.3   Licences

The Anubis-Free is a free-source software provided AS IS and without a support.

The Anubis-Pro and Anubis-RT are provided under a commercial licence with a support relevant for standard issues.

1.4   Installation

The software is distributed from *http://www.gnutsoftware.com*. It can be downloaded as a pre-compiled binary (dynamic/static, 32b/64b) and it does not need any specific compilation or installation procedure.

1.5   Development

The software has been developed using the G-Nut core library [2] initially designed at the Geodetic Observatory Pecný of the Research Institute of Geodesy, Topography and Cartography. Since 2018, both the library and user applications has been maintained as well as further developed by the G-Nut Software s.r.o..

In case of bug reporting, feedback or wish-list, and other questions, contact us via the email: *support@gnutsoftware.com*. Note: a support is currently on the best-effort basis only. In case you need any specific support, contact us for negotiation.

1.6   Acknowledgements

We acknowledge the availability of third-party libraries used within the G-Nut:

- Newmat - C++ Matrix library (*http://www.robertnz.net/nm_intro.htm*).
- Pugixml - a light-weight, simple and fast XML parser for C++ (*http://pugixml.org/*).
- RNXCMP - a Hatanaka's RINEX compression (*https://terras.gsi.go.jp/ja/crx2rnx/*) [3].
- md5 - a reference implementation of RFC 1321 converted to C++ by Frank Thilo for bzflag (*http://www.bzflag.org*).
- rapidjson - a fast JSON parser/generator for C++ with both SAX/DOM style API (*https://rapidjson.org/*).
- zlib - a free lossless data-compression C library (*https://www.zlib.net*).
- gzstream - a small C++ wrapper providing the functionality of the zlib C-library (*https://www.cs.unc.edu/Research/compgeom/gzstream/*).

2. Operating modes (pre-setting)

The Anubis operates in several modes (format checking, metadata/data quality control or editing). These are briefly described in a relation to the software background operations such as input data decoding, data filtering, data containers, data quality control (QC), and outputs encoding. Relevant pre-setting for any user-requested operating mode can be triggered with a single command-line argument:

```
Anubis          --help | --thin | --edit | --summ | --full | --lite | --kpis
```

- **Help** for configuration options and default setting.
- **Thin** for pure metadata control and header check.
- **Edit** for pure metadata editing and data filtering.
- **Lite** for pure quantitative control with optional navigation data.
- **Full** for qualitative and complex control with mandatory navigation data.
- **Summ** for a minimum output, but still supporting qualitative and complex control.
- **KPIs** for a horizontal and vertical positioning and DOP key-parameter indicators (KPIs).

Additionally, the real-time operation mode can be simultaneously applied, see Section 7.

Figure 2.1 shows relations of internal operations according to user inputs, decoding procedures, observation and navigation data representation, QC operations and outputs. Obviously, the navigation data are mandatory for the full QC. Thin mode operates purely on a data decoding. Data decoding includes data filtering according to user setting (selecting systems, satellites, observation types, frequency bands, tracking attributes, navigation data types, etc.). Both edit and QC modes use then pre-filtered data from the internal container representation.
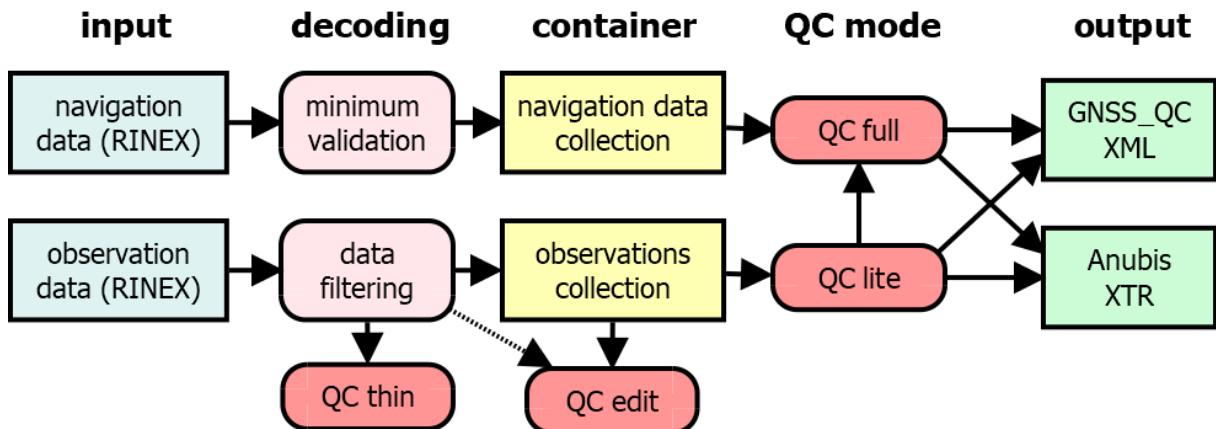

Figure 2.1: Anubis operating modes and relation to inputs, outputs and internal procedures

2.1   User help

A basic help (*–help*) shows principal command-line arguments.

```
G-Nut/Anubis Pro [3.1] compiled: Jan 11 2021 18:41:07 (Rev: 3113)

Usage:

    -h|--help              .. this help
    -V                     .. software version
    -x file                .. configuration input file
    -X                     .. configuration stdout (default)
    -Z                     .. log-file without time stamps
    -z file                .. configuration output file
    xml-setting            .. XML setting via a command line (see below)


XML setting via command line:

:elem:subelem    val    .. set xml element value   (e.g. :inp:rinexo <rinexo file>)
:elem:attribute=val     .. set xml attribute value (e.g. :out:verb=1)
:elem[ID]:subelem   val .. set xml element value   (e.g. :sys[GPS]:sat "G01␣G02␣G03")
:elem[ID]:attribute=val .. set xml attribute value (e.g. :site[GOPE]:rec="RECEIVER")
```

An extended help ($-\boldsymbol{X}$) then gives a detailed list of applicable configuration options including a brief description and default setting values. [1]

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE config>
<config>
  <!--general setting:
   max_cores=integer          .. max cores to be used (0=any)
   max_files=integer          .. max files in a batch (0=all)
   max_sites=integer          .. max sites in a batch (0=all)
 * max_epoch="0"              .. maximum number of epochs in containers (0=all)

 * run_rtime=bool             .. process in real-time (empty=false)
 * run_delay=int              .. delay of real-time processing [s]

   beg     %Y-%m-%d %H:%M:%S   .. beg time % (empty=auto)
   end     %Y-%m-%d %H:%M:%S   .. end time % (empty=auto)
   int     integer             .. data sampling interval (empty=all)
   rec     list of strings     .. receiver(s) (empty=all)
   sys     list of strings     .. GNSS system(s) (empty=all)
  -->
  <gen max_cores="1" max_sites="1" max_files="1"><beg /><end /><int /><rec /><sys /></gen>

  <!--quality check setting:
   sec_sum=int        .. summary statistics
   sec_hdr=int        .. header metadata check
   sec_obs=int        .. observation statistics
   sec_est=int        .. estimated values
   sec_gap=int        .. gap & pieces
   sec_bnd=int        .. observation bands
   sec_pre=int        .. cycle-slip, clock-jumps
   sec_ele=int        .. azimuth/elevation (if navigation)
   sec_mpx=int        .. multipath calculation
   sec_snr=int        .. signal-to-noise ratio
   sec_sat=int        .. satellite information
   sec_kpi=int        .. key-parameter indicators

   int_stp=int        .. [s] interval for time-spacing
   int_gap=int        .. [s] interval for gap identification
   int_pcs=int        .. [s] interval for small pieces identification
   mpx_nep=int        .. number of epochs for multipath calculation
   mpx_lim=double     .. sigma-multiplicator for MP cycle-slip & outlier detection
   pos_kin=bool       .. kinematic receiver (true = kinematic)
   pos_int=int        .. positioning interval
   pos_cut=double     .. positioning elevation angle cut-off (degrees)
   ele_cut=double     .. user elevation cut-off (only for expt/have, degrees)
   col_sat=int        .. number of columns for sat-specific reporting
   sat_rec=bool       .. expected observations from satellites (true:any | false:signal)
   use_health=enum    .. use of satellite health (position|statistics|all)
 * use_qcmode=enum    .. section pre-setting (thin|edit|summ|lite|full|kpis)
 * ref_sig=double     .. [m] maximum sigma of reference coordinates (default:0, estimated median)
 * max_vpe=double     .. [m] maximum vertical position error
```

---

[1] '*' represents a configuration of a non-free functionality

```
 * max_hpe=double        .. [m] maximum horizontal position error
 * max_dop=double        .. [-] maximum PDOP
   -->  ......
</config>
```

2.2   Thin mode – metadata control

The **thin** mode is ultra fast because it exits immediately after reading RINEX file header, i.e. prior any data decoding. It can be effectively applied for a single file or multiple files giving errors or warning messages on a standard error output (*/dev/stderr*). Every present error trigger an increase of the software return code.

Each file header is confronted with a user metadata introduced either from the XML configuration file (the <site> element), see Section 3, or from a SINEX input file (or both).

For a set of RINEX files, the bellow command generates errors and warning messages and, optionally, a non-zero return CODE if a serious problem (error) occurs. A silent output, and a zero return CODE, characterises a fully compliant RINEX header format and metadata corresponding with user requests.

```
Anubis  --thin  :inp:rinexo '\$(ls␣*crx.gz)' -x My-Site-Meta-Data.xml
```

Examples of RINEX header metadata inconsistency:

```
*** Warning: AZGR00PRT - object inconsistent DOMES: 31902M004 !! AZGR
*** Warning: MPTB00MOZ - object inconsistent DOMES: 39308M001 !! Unknown
*** Error: ABFC00NGA - object inconsistent ECC: 0.0000 0.0000 0.1800 !! 0.0000 0.0000 0.1710
*** Error: PAGU00PRT - object inconsistent ANT: TRM55971.00    NONE !! TRM57971.00    NONE
```

Examples of RINEX header format incompatibility:

```
*** Warning: BDS band changed: B1->B2 (RNX 3.02) .. SUBI00PRT_R_20203000000_01D_30S_MO.crx.gz
*** Warning: INTERVAL not available! .. ABFC00NGA_R_20203000000_01D_30S_MO.crx.gz
*** Error: GLO BIASES not available! .. FTNA00WLF_R_20203000000_01D_30S_MO.crx.gz
*** Error: GLO SLOT/FREQ not available! .. PAGU00PRT_R_20203000000_01D_30S_MO.crx.gz
```

2.3   Edit mode – data filtering, metadata modifying

In principle, the **edit** mode can be fast if file header metadata are modified only, i.e. without touching the file body, though the entire file still needs to be decompressed and compressed for the corresponding input or output files. Such a fast metadata edit option (i.e. 'untouched' body) is foreseen for a future software release and, currently, the metadata modification still includes the data full decoding and encoding procedure. Actual data editing, filtering then includes window editing, down-sampling, selecting systems, satellites, and signals. Thus reorganised data are written in the output file according to the software encoder in both cases.

Output file compression (Z/gz) can be applied independently from the input one, and arbitrarily with/without utilizing GNSS data Hatanaka's compression, i.e. any combination is supported.

Input navigation and observation RINEX files decompression is handled automatically, (more

precisely, if it concerns the Hatanaka compression and the RINEX format version). The Z/gz compressions of input and output files, and for any format including log files, are handled automatically according to input/output file name extensions:

.gz for GZ file compression,
.Z for LZH file compression.

However, formats of output file, including the Hatanaka compression, are requested explicitly within the specification of the output format (*Note: input files are handled automatically*):

:out:rinexo2 for RINEX 2 output file,
:out:rinexo3 for RINEX 3 output file,
:out:crinexo2 for RINEX 2 Hatanaka output file,
:out:crinexo3 for RINEX 3 Hatanaka output file,
:out:bncobs for BKG NTRIP Client (BNC, [4]) observation output file.

Some limitations existed for the **edit mode** in Anubis 3.0-3.1:

1. format conversion is not supported (from RINEX 3 to RINEX 2 and vice versa).
2. editing can be performed on data from a single station only, although multiple input files can be concatenated into a single one.

Since Anubis 3.2, a multi-file editing is experimentally enabled including conversions from RINEX 2 to RINEX 3 and vice versa.

For metadata modification, user setting can be provided through either XML configuration file <site> element or from a SINEX input file. Both can be combined too, see 3, while in case of conflict, the priority is modified with the **priority=** ('GNUT', 'SINEX', 'RINEX') of the <site> element. Data filtering and editing can be requestedy with the XML setting or command-line arguments.

Example of header metadata modification using XML <site> configuration, and SINEX:

```
Anubis     --edit -x My-Site-Meta-Data.xml                          \
               :inp:rinexo GOPE3660.20D.Z                           \
               :out:rinexo2  gope3660.20o.gz                        \
               :out:crinexo2 gope3660.20d.gz
```

or

```
Anubis     --edit :inp:rinexo GOPE3660.20D.Z                        \
               :inp:sinex  My-Site-Meta-Data.snx.gz                 \
               :out:rinexo2  gope3660.20o.Z                         \
               :out:crinexo2 gope3660.20d.Z
```

Example of metadata modification (receiver type change) from a command-line arguments:

```
Anubis     --edit :inp:rinexo GOPE3660.20D.Z :out:crinexo2 gope3660.20d.gz \
               :site[GOPE]:rec=TRIMBLE
```

Example of data filtering (down-sampling, window of 1h, GNSS with selected observations):

```
Anubis     --edit :inp:rinexo   GOPE00CZE_R_20203660000_01D_30S_MO.crx.gz \
                 :out:rinexo3  GOPE00CZE_R_20203660100_01H_60S_MO.rnx    \
                 :out:crinexo3 GOPE00CZE_R_20203660100_01H_60S_MO.crx.gz \
                 :gen:sys 'GPS␣GAL␣GLO␣BDS'                              \
                 :gen:beg '2020-12-31␣00:00:00'                         \
                 :gen:end '2020-12-31␣01:00:00'                         \
                 :gen:int 60                                            \
                 :sys[GPS]:gobs 'C1␣P2␣L1␣L2'
```

2.4 QC mode – quantitative and qualitative control

The quality control (QC *lite, full, summ* and *kpis*) modes can be used on single/multiple files and stations. The mode can also be specified in the XML setting (*use_qcmode*).

Various modes differ in QC section pre-setting (modify default setting, see Table 2.1), however, these may be still changed with individual section setting either within the XML or configuration command-line arguments.

Table 2.1: Anubis pre-setting for different operating modes

Operating mode	Thin	Edit	Lite	Full	Summ	KPIs
XML-setting				(quality control)		
:qc:sec_sum=	0	0	1	2	2	3
:qc:sec_hdr=	0	0	1	1	1	1
:qc:sec_obs=	0	0	1	1	1	1
:qc:sec_gap=	0	0	1	1	-1	1
:qc:sec_bnd=	0	0	1	1	-1	1
:qc:sec_pre=	0	0	1	1	-1	0
:qc:sec_est=	0	0	0	1	0	0
:qc:sec_ele=	0	0	0	1	0	0
:qc:sec_mpx=	0	0	0	1	-1	-1
:qc:sec_snr=	0	0	0	1	0	0
:qc:sec_sat=	0	0	0	1	0	0
:qc:sec_kpi=	0	0	0	0	0	9

Anubis supports three complementary QC output formats, which contents are specifically described in sections 4, 5, and 6.

XTR QC output – a native ASCII format organised in QC sections supporting adaptable verbosity; easy-to-use for multi-file grep commands; for details see Section 4.

XML QC output – a QC summary using extensible markup language (XML) format and containing data optimal for populating a relation database system (e.g. MySQL, PostgreSQL); efficient for visualising time series of QC key parameters, suitable for monitoring an evolution of site quality in time; for details see Section 5.

JSON QC output – a JSON high-resolution format designed primarily for an efficient archive of satellite-/epoch-wise data; convenient for visualisation of positioning and sky-plots from a single data file with QC results; optionally for populating non-relational database systems (e.g. MongoDB); for details see Section 6.

Example of QC lite operation without navigation data file, requesting multipath and signal-to-noise outputs, and eliminating a majority of other quantitative sections:

```
Anubis     --lite :inp:rinexo GOPE3660.20D.Z   \
                  :out:xtr    GOPE3660.xtr.gz  \
                  :out:log    anubis.log        \
                  :qc:sec_mpx=2 :qc:sec_snr=2  \
                  :qc:sec_hdr=0 :qc:sec_obs=0 :qc:sec_bnd=0 :qc:sec_gap=0 :qc:sec_pre=0
```

Example of QC full operation supported with a global navigation file and all QC outputs:

```
Anubis     --full :inp:rinexn BRDC00GOP_R_20203660000_01D_MN.rnx.gz \
                  :inp:rinexo GOPE3660.20D.Z                        \
                  :out:xtr    GOPE3660.xtr.gz                       \
                  :out:xml    GOPE3660.xml.gz                       \
                  :out:json   GOPE3660.json.gz                      \
                  :out:log    /dev/stdout                           \
                  :out:verb=2
```

Example of QC summary targeting a full XML output for populating QC database:

```
Anubis     --summ :inp:rinexn BRDC00GOP_R_20203660000_01D_MN.rnx.gz \
                  :inp:rinexo GOPE3660.20D.Z                        \
                  :out:xml    GOPE3660.xml
```

2.5 KPI mode - position performance

This mode has been developed for a purpose of monitoring individual GNSS constellations through site(s) horizontal and vertical positioning performance, and DOP values. The mode obviously depends on complete and correct navigation data. It is able to measure Signal-in-Space (SiS) errors based on processing data from one or more station(s) and reflecting the quality of navigation data together with actually exploited signals. However, the results might be affected by site environment and instrumentation.

The mode is enabled by exploiting a reference station position, given either from a SINEX file or from XML <site> configuration, for estimating horizontal and vertical position differences in epoch-by-epoch single point positioning.

Figure 2.2 shows example of improved performance of E1+E5a positioning supported by Galileo FNAV navigation data after four new Galileo satellites became active on February 11, 2019, first enabling a stable $GDOP < 6$ every day all over the globe. Prior the date, the positioning solution was regularly unavailable at some areas due to few observed satellites only. With data filtering, such positioning can be performed for any individual constellations, combination of tracking signal and navigation data types. The output is described in details in the KPI section 4.12.

Example of KPI estimates exploiting globally merged (GOP BRDC) navigation data file:

```
Anubis     --kpis :inp:rinexn BRDC00GOP_R_20203660000_01D_MN.rnx.gz \
                  :inp:sinex  MetaData+ReferenceCRD.snx             \
                  :inp:rinexo GOPE3660.20D.Z                        \
                  :out:xtr    GOPE3660.xtr                          \
                  :out:json   GOPE3660.json
```

GAL_FNAV/GOP [2019/02] - SITE time-series of position/height errors (PDOP<6)
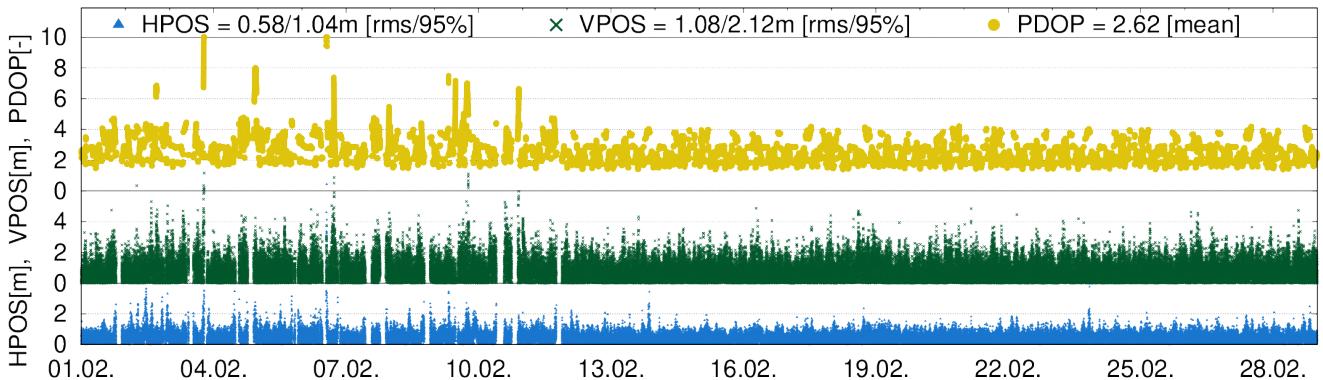
Figure 2.2: Monthly positioning performance with Galileo E1+E1a signals and FNAV data

2.6 Handling multiple files

Anubis was initially designed for a single RINEX input file operation, which can be optimal for a file metadata control and header compliance (interest in a return code) or for a single-file data filtering or metadata editing:

```
Anubis     --thin -x Metadata.xml  :inp:rinexo GOPE3660.20D.Z  :out:xtr      GOPE3660.xtr
Anubis     --edit -x Editting.xml  :inp:rinexo GOPE3660.20D.Z  :out:crinexo2 GOPE3660.20D.gz
```

Anubis however supports also editing and QC operations on a massive repository such as data centres, large network operator's or data analyst's repositories. A huge number of input files can still be handled efficiently including an easy and flexible configuration.

The following example shows Anubis thin operation on files (or a mask of files) in several user directories (files are listed via a shell command-line argument expansion):

```
Anubis     --thin :inp:rinexo  "\$(ls␣dir1/2020/366/*crx.gz␣␣␣dir2/2020/366/*crx.gz)"
```

Anubis 3.2 then introduced an alternative (experimental) approach when supporting generic mask(s) for predefined input file names according to the user <gen> settings:

```
Anubis     --summ :inp:rinexn   "brdc/BRDC00GOP_R_20003660000_rnx.gz)"              \
                 :inp:rinexo   "dir/2020/366/(rec)_R_20203000000_01D_30S_MO.crx.gz" \
                 :out:xtr      "(rec).xtr.gz"                                       \
                 :out:log      "/std/out"                                          \
                 :gen:rec      "GOPE00CZE␣GOP700CZE␣BRUX00BEL␣WTZR00DEU␣ONSA00SWE"
```

Note the difference between both approaches. The former, the OS 'ls' command expands into a list of available files in directories allowing regular expressions. The latter, Anubis is asked to search specific predefined file names defined with site and time-dependent variables.

Anyway, for operations with multiple files, user have to specify a generic file name, at least for outputs, i.e. by defining site name and, optionally, time-specific variables. The site name was first represented by a single *(rec)* variable (until Anubis 3.1), but more variants are available as of Anubis 3.2:

- '(rec)' – original station name (lower case),
- '(REC)' – original station name (upper case),

- '(rec4)' – short (or original) station name (lower case),
- '(REC4)' – short (or original) station name (upper case),
- '(rec9)' – long (or original) station name (lower case),
- '(REC9)' – long (or original) station name (upper case).

Additionally, Anubis is designed to guess an original site name adaptively (short / long by comparing first 4 characters) according to user setting (or SINEX), RINEX file name, and MARKER in the RINEX header.

A user can also enforce a short/long site name by user setting which is considered with a highest priority. Finally, a user can easily extend a short file/site name of RINEX input files into long ones for RINEX output files, with the same user settings. See the following multi-file edit example:

```
Anubis     --edit :inp:rinexo   repo/2020/366/(rec)3000.20d.gz                              \
               :out:crinexo3 repo/2020/366/(REC9)_R_20203000000_01D_30S_MO.crx.gz       \
               :out:log      /dev/stdout                                                \
               :gen:rec      'GOPE00CZE␣GOP700CZE␣GOP600CZE␣BRUX00BEL␣WTZR00DEU␣ONSA00SWE'
```

Second, several time-dependent variables can be used for actual reference epoch(s) of the processed data:

- '%Y' – 4-char year,
- '%y' – 2-char year,
- '%j' – day of year (DoY),
- '%m' – month of year,
- '%d' – day of month,
- '%w' – day of GPS week,
- '%W' – GPS week,
- '%I' – hour ID (upper case),
- '%i' – hour ID (lower case),
- '%H' – hours of day,
- '%M' – minutes of day,
- '%S' – seconds of day.

When site/time-dependent variable is identified within the input (output) format, the file name is considered as a mask instead of a single name. Such mask is then expanded into a site-/time-specific list of input (output) candidates that are further analysed – only those relevant (i.e. existing for inputs and applicable for outputs) – are read (written), others are ignored silently.

Using time-dependent variables are specified for any input files, both <beg> and <end> elements are requested to be defined, thus enabling a time-dependent mask expansion. The list of input RINEX files are thus generated by Anubis:

```
Anubis     --thin :inp:rinexo  inp/%Y/%d/(REC9)_R_%Y%d00000_01D_30S_MO.crx.gz \
               :out:xtr      out/(rec4)_%Y-%m-%d.xtr.gz                    \
               :out:xml      out/(rec4)_%Y-%m-%d.xml.gz                    \
               :out:json     out/(rec4)_%Y-%m-%d.xml.gz                    \
               :out:log      anubis.log                                   \
               :gen:beg      2020-01-01T00:00:00                          \
               :gen:end      2020-01-31T00:00:00                          \
               :gen:rec      'GOPE00CZE␣GOP600CZE␣GOP700CZE'
```

There exists also a specific setting to control efficient operation on a multiple files, which can be processed 1) in parallel, or in a sequence, 2) possibly restricting a number of used cores (understand processing threads), or a maximum of files processed within a single batch. The following configuration enables to configure these operations:

- ***max_cores*** defines a maximum number of cores to be used (0=auto),
- ***max_files*** defines a maximum number of files to be used in a single batch (0=auto),
- ***max_sites*** defines a maximum number of sites to be used in a single batch (0=auto).

3. Configuration

The software can be configured from a GNUT XML configuration file, using command-line arguments, or mixing both. In principal, the XML configuration represents an native structure for the software settings, while command-line arguments enable user to a) configure fully the software, b) modify a user XML (background) configuration, or c) adapt default setting.

This section describes only the XML configuration since any command-line argument corresponds the native setting by applying (recursively) the following rules:

```
:elem:subelem    val    .. set xml element value   (e.g. :inp:rinexo <rinexo file>)
:elem:attribute=val     .. set xml attribute value (e.g. :out:verb=1)
:elem[ID]:subelem   val .. set xml element value   (e.g. :sys[GPS]:sat "G01␣G02␣G03")
:elem[ID]:attribute=val .. set xml attribute value (e.g. :site[GOPE]:rec="RECEIVER")
```

Contrary to the XML setting, additional conditions that have to be considered:

- any duplicitous argument (e.g. :elem:attribute="value") overwrites an existing one.
- multi-elements (:sys[id], :site[id]) require an element identifier (e.g. $[GPS]$, $[GOPE]$).

It is important to note that for avoiding duplicitous command-line arguments, user has to enclose multiple values in quotation marks. Obviously, any input previously set from either the XML or command-line setting is finally replaced with the last (unique) command-line argument.

A basic help (*--help*) as well as a full help (*-X*) has already been introduced in Section 2.1.

The software execution using the XML setting only can be simple:

```
Anubis -x my_config.xml
```

On the other hand, the command-line setting can be encapsulated in a processing shell script:

```
#!/bin/bash

Anubis  :inp:rinexo  GOPE1730.17o         \
        :inp:rinexn  BRDC1730.17p         \
        :out:xtr     GOPE1730.xtr         \
        :out:xml     GOPE1730.xml         \
        :out:log     GOPE1730.log         \
        :out:verb=1                       \
        :gen:beg     "2017-06-22␣00:00:00" \
        :gen:end     "2017-06-23␣00:00:00" \
        :gen:sys     "GPS␣GLO"            \
        :gen:int 30
```

Mixing both configurations can be useful particularly for specifying user-default static parameters in the XML file (e.g. generic configuration, output file masks, site metadata) with added/adapted dynamic parameters from the command-line (e.g. input files):

```
Anubis -x metadata.xml    :inp:rinexo "$(ls␣data/?????????_R_??????????_01D_30S_MO.crx.gz") \
                          :inp:rinexn "$(ls␣brdc/BRDC?????_R_??????????_01D_MO.rnx.gz")
```

*Hint: in this particular case, it is recommended to store actual configuration along with archiving other output files, i.e. using option **-z final.cfg** .*

3.1   XML setting structure

A structure of the Anubis XML configuration file consists of elements and attributes of the Extensible Markup Language (XML). The configuration is placed within a root element <config> and following XML tree of child elements:

- < *gen* >   for general setting,
- < *sys* >   for GNSS-specific setting,
- < *nav* >   for navigation data setting,
- < *inp* >   for input setting,
- < *out* >   for output setting,
- < *qc* >   for QC setting,
- < *site* >   for site metadata setting.

Selected elements can be omitted (default software values are considered then), however, input and output elements are mandatory. Individual elements are described in following sections.

3.2   General configuration

The element <gen> includes general setting for the software operation such as a start / end time of a processing period, requested GNSS constellations, list of GNSS sites (receivers), and a sampling interval of the processing.

```
<gen
   max_cores="0"                        # maximum number of cores to be used (0=all)
   max_sites="0"                        # maximum number of sites in a batch (0=all)
   max_files="0"                        # maximum number of files in a batch (0=all)
 * max_epoch="0"                        # maximum number of epochs in containers (0=all)

 * run_rtime=bool                       # process in real-time (empty=false)
 * run_delay=int                        # delay of real-time processing [s]
>
   <beg> "2014-03-01␣00:00:00" </beg>
   <end> "2014-03-01␣23:59:59" </end>
   <sys>   GPS GLO GAL BDS QZS   </sys>
   <rec>   GOPE POTS WTZR        </rec>
   <int>   300                   </int>
</gen>
```

Remark: the general setting is applied at the decoding level, thus data/products are filtered by the setting prior any data analysis, see Section 2.

Along with setting sub-elements, the configuration includes several attributes **max_cores** , **max_cores** , **max_files** to configure sequential/parallel processing described in Section 2.6.

Additionally, **run_rtime** and **run_delay** attributes can be used to specify real-time operation and the real-time process delay. The **max_epoch** attribute is then used in the RT version for specifying how much data/products will be kept in internal containers. For its optimal settings, see Section 7.

The two elements (<sys>, <rec>) support multiple string values separated by one or more blank characters.

The elements <beg> and <end> define the start/end of the processing interval, both in the format *'%Y-%m-%d %D:%M:%S'* (or in a single string *'%Y-%m-%dT%D:%M:%S'*).

The <sys> element specifies global and regional systems requested. These 3-char GNSS abbreviations are allowed: **GPS** , **GLO** , **GAL** , **BDS** , **QZS** , **SBS** , **IRN** for GPS, GLONASS, Galileo, BeiDou, QZSS, SBAS, IRNSS, respectively. By default (or empty element), all these above supported GNSS and regional systems are considered.

Remark: while the empty element represents all the systems, the '-' pre-fix (minus) enables rejecting any particular system.

The <rec> element enables a list of receiver (site) identifiers to be processed. The list may be left empty (default), considering any site identified within input data to be processed. It may contain string(s) corresponding to name(s) found in the **MARKER** field of the RINEX observation header(s), or NTRIP [5] **mountpoints** in case of real-time input data stream(s).

Remark: the MARKER of the RINEX file may not be filled properly, or it may be inconsistent with the site name from the file name (e.g. 4-char vs. 9-char long file name), the Anubis guesses a correct name when confronting MARKER, file name, and the <rec> user setting. The guess name is reported in the log file, and it is used for the site variables, see Section 2.6.

The <int> element represents the processing sampling rate expressed in seconds. The sampling rate defines an interval for decoding (and filtering, see Section 2)) input data, thus affecting majority of output QC characteristics.

Remark: in normal cases, the element should contain integer number, however, for processing high-rate data ($> 1Hz$), the sampling rate may be float with a number of extra decimal digits ('0') to define a time resolution of the input data.

3.3 GNSS configuration

Constellation-specific configuration is enabled in the <sys> element, which needs to be identified with a mandatory **id** attribute in order to distinguish individual GNSS constellations, see Section 3.2. Hence, multiple elements are supported.

Each system may contain the following sub-elements:

- < *nav* > for navigation type selection,
- < *sat* > for satellite selection,
- < *type* > for observation type selection,
- < *band* > for observation frequency band selection,
- < *attr* > for observation tracking attribute selection,
- < *gobs* > for observation signal-specific selection (add-on/pure signal setting).

These can be used for defining a set of conditions applied for input data filtering during decoding process. By default, all the setting are empty, thus no filtering is applied. By introducing any single value into any single element then activates the filter and, consequently, all enabled values should be explicitly specified. An example of setting for Galileo (not very meaningful):

```
<sys id='GAL'>
  <sat>   E01 E05 E06 E08 E09 E12 E14   </sat>   # disable unspecified Galileo satellites
  <nav>   FNAV INAV_E07                 </nav>   # disable INAV_E01
  <type>  C L                           </type>  # disable Doppler and signal-to-noise ratio
  <band>  1 2 5 7   8                   </band>  # disable E6 frequency band observations
  <attr>  C L X P W                     </attr>  # disable unspecified observation attributes
  <gobs>  C1C C2P L1C L2P S1C S2P L2P   </attr>  # add-on specifically mentioned signals
</sys>
```

Remark: A combination of <gobs> setting together with observation type, frequency band, or attribute results represents the add-on signal-specific setting with respect to the other elements. Codes representing observation types, frequency bands, tracking attributes or specific signals correspond to the RINEX standards.

The <sat> element specifies a list of 'enabled' satellites (PRNs). If the element is not specified in the configuration file, or is empty, all available (supported) satellites for given constellations contribute as default.

Remark: similarly as for the <sys> , the '-' pre-fix (minus) enables a negative selection, i.e. rejecting only particular satellite.

The <nav> element specifies the navigation data types when supporting:

- **FNAV** for Galileo FNAV from the E5a signal,
- **INAV** for Galileo INAV from any signal,
- **INAV_E01** for Galileo INAV from the E1 signal,
- **INAV_E07** for Galileo INAV from the E5b signal,
- **CNAV** for GPS CNAV navigation type (not implemented yet in RINEX 3),
- **NAV** for any navigation type.

3.4 Navigation data configuration

Handling navigation data is optional setting for two specific purposes:
1. filtering, checking, and handling navigation data within G-Nut applications, and
2. generating merged and quality control navigation data, i.e. BRDC file, [6]).

Selected parameters can be used in any G-Nut application:
- **chk_record** (boolean) – use simple range checking for individual navigation records
- **chk_health** (boolean) – filter out satellites with unhealthy status
- **chk_duplic** (boolean) – filter out duplicitous navigation records
- **chk_tot** (boolean) – check navigation data validity with respect to the ToT epoch
- **key_tot** (boolean) – store/search navigation records according to the ToT epoch
- **ext_nav** (double) – multiplier to extended navigation data validity interval

Remaining parameters are used for the comprehensive navigation data consolidation (merging and checking). These are specific and should be applied only with a sufficient number of input navigation files and enabling duplicitous records for possible data selecting:
- **chk_header** (boolean) – NAV data statistical analysis for BRDC header
- **chk_params** (boolean) – NAV data time-series analysis for key parameters
- **chk_others** (boolean) – NAV data statistical analysis for other parameters
- **sig_bounds** (double) – a sigma multiplier for outliers in NAV time-series analysis

Default values for navigation data setting should be convenient for a majority of Anubis QC data utilisation, and these are provided below:

```
<nav chk_record="true"   chk_health="true"   chk_duplic="true"
     chk_tot="true"       key_tot="true"       ext_nav="1.0"
     chk_header="false"  chk_params="false"  chk_others="false"   sig_bounds="3.0"
/>
```

3.5   Input configuration

Supported input formats were introduced in Section 1.1. The <inp> element includes sub-elements for inputs for observation data, navigation data, and metadata. Each sub-element represents a specific format and may contain a space-separated list of more input paths, e.g.:

```
<inp>
  <!-- input setting: path may include prefix file:// [..tcp,ntrip]
    <rinexn>       path1 [path2 ..]    </rinexn>     # for RINEXN format (file)
    <rinexo>       path1 [path2 ..]    </rinexo>     # for RINEXO format (file)
  * <sinex>        path1 [path2 ..]    </sinex>      # for SINEX  format (file)
  * <sp3>          path1 [path2 ..]    </sp3>        # for SP3    format (file)
  * <rtcm>         path1 [path2 ..]    </rtcm>       # for RTCM   format (file or stream)
  * <bncobs>       path1 [path2 ..]    </bncobs>     # for BNCOBS format (file or stream)
  -->
</inp>
```

The path represents a generic input such as data files, data streams, and file masks.

Remark: the file mask is not a valid path, but contains user- or time-specific variables, see Section 2.6 that are replaced with valid strings prior a file searching.

The path may include one of the following structures (the prefix **file://** is optional):

- **file://basename.ext** – file accessible within a local/network file system,
- **tcp://host:port** – TCP data stream for host+port (* Anubis-RT only),
- **ntrip://mnt/user:pwd@host:port** – NTRIP data stream for mountpoint (mnt) accessible with specified username and password for host+port (* Anubis-RT only).

Remark: the file name doesn't play any role in the format definition. The RINEX format version is automatically recognised in the software as well as the **Hatanaka** *compression if used as an input (Hatanaka format is not supported with Anubis Free). Files are also automatically compressed/decompressed by the software when using the* **.gz/.Z** *extension in the file name (only* **.gz** *is supported with Anubis Free).*

An example below shows mixed inputs for GNSS observation and navigation files using relative and absolute paths and RINEX 2 and 3 input formats:

```
<inp>
  <rinexn> file://../data/brd/brdm0220.15p.gz       </rinexn>
  <rinexo> file://../data/obs/onsa0220.15o.gz
           /home/jan/data/obs/GOPE00CZE_R_201522000000_01D_30S_MO.crx.gz
  </rinexo>
</inp>
```

Multiple files can be handled using the site/time-specific variables as described in Section 2.6,

Remark: Windows users should specify the file path as it is supported by the platform.

3.6   Output configuration

Supported output formats were introduced in Section 1.1. The <out> element includes sub-elements for individual outputs for QC results, log files and, optionally, observation and navigation data.

The configuration is similar as for inputs described in the previous section 3.5. Each sub-element represents a specific format and contain a space-separated list of one or more output paths (the path again represents a generic output, i.e. data file, data stream, file mask):

```
<out
     verb="[integer]"                           # log verbosity (default: 0)
   append="[boolean]"                           # log appending (default: false)
>
   <!-- output setting: path may include prefix file:// [..tcp,ntrip]
   <log>         path1 [path2 ..]    </log>        # for LOG format      (file)
   <xml>         path1 [path2 ..]    </xml>        # for XML format      (file)
   <xtr>         path1 [path2 ..]    </xtr>        # for XTR format      (file)
 * <json>        path1 [path2 ..]    </json>       # for JSON format     (file)
 * <crinexo2>    path1 [path2 ..]    </crinexo2>   # for CRINEXO2 format (file)
 * <crinexo3>    path1 [path2 ..]    </crinexo3>   # for CRINEXO3 format (file)
 * <rinexn2>     path1 [path2 ..]    </rinexn2>    # for RINEXN2 format  (file)
 * <rinexn3>     path1 [path2 ..]    </rinexn3>    # for RINEXN3 format  (file)
 * <rinexo2>     path1 [path2 ..]    </rinexo2>    # for RINEXO2 format  (file)
 * <rinexo3>     path1 [path2 ..]    </rinexo3>    # for RINEXO3 format  (file)
 * <bncobs>      path1 [path2 ..]    </bncobs>     # for BNCOBS format   (file or stream)
 * <rtcm>        path1 [path2 ..]    </rtcm>       # for RTCM format     (file or stream)
   -->
</out>
```

The <out> element contains also two attributes relevant only to the log file:

- **verb** (integer) defines a verbosity level with values 0-9 representing details of printed information.
- **append** (boolean) may enforce a continuation of writing into the existing file (the file is created if does not exist). The default is 'false' for creating (new) or overwriting (existing) file.

Remark: Anubis automatically creates any non-existing directory within the path.

The example below shows the output setting for the Anubis QC operation when multiple files are enabled using the site/time-specific variables, see Section 2.6:

```
<out verb=2 append="false">
  <log>    OUT/anubis-%Y-%d.log.gz    </log>
  <xtr>    OUT/(rec4)-%Y-%d.xtr.gz    </xtr>
  <xml>    OUT/(rec4)-%Y-%d.xml.gz    </xml>
  <json>   OUT/(rec4)-%Y-%d.json.gz   </json>
</out>
```

Supported QC data output formats are described in Sections 4, 5, and 6.

Using the edit mode, the output format for RINEX observations and navigation files needs to be set explicitly, using a specific encoder keyword, see Section 2.3, e.g.:

```
<out verb=2 append="false">
  <log>      /dev/stdout              </log>
  <rinexn3>  OUT/BRDC2200.15p.gz      </rinexn3>
  <rinexo2>  OUT/GOPE2200.15o.Z       </rinexo2>
  <crinexo2> OUT/GOPE2200.15d.Z       </crinexo2>
</out>
```

Remark: any output file, a log in particular, may be also redirected to the standard output (/dev/stdout) or the error output (/dev/stderr), on a Linux platform. Note also the variable $(rec) enabling a single mask for a common output setting for all simultaneously processed files/stations.

The Anubis log file reports individual lines started with an event epoch (system time), followed

with an object identifier (the one throwing the message) and its verbosity, and the message itself. An example of a log file with **verb=2** is given below:

```
2017-08-10 06:25:19 [main:1] READ: file://../data/brdc/yel22370.16p started
2017-08-10 06:25:19 [rinexn:1] skipped sat G04 warning: unhealthy satellite  G04 2016-08-23 22:00:00
2017-08-10 06:25:19 [rinexn:1] skipped sat G04 warning: unhealthy satellite  G04 2016-08-23 21:59:28
2017-08-10 06:25:19 [main:0] READ: file://../data/brdc/yel22370.16p   0.171 sec
2017-08-10 06:25:19 [main:1] READ: file://../data/obs/yel22370.16o started
2017-08-10 06:25:19 [rinexo2:0] MARKER NAME: YEL2 (4-CH)
2017-08-10 06:25:19 [gallobj:1] add new obj YEL2
2017-08-10 06:25:19 [rinex:1] Object created, using RINEX header: YEL2 2016-08-24 00:00:00
2017-08-10 06:25:19 [gobj:0] Warning: object YEL2 completed (Name): YEL2
2017-08-10 06:25:19 [gobj:0] Warning: object YEL2 completed (Domes): 40127M006
2017-08-10 06:25:19 [gobj:0] Warning: object YEL2 completed (Antenna): LEIAR25.R4      NONE
2017-08-10 06:25:19 [gobj:0] Warning: object YEL2 completed (Coordinates): -1224441.973 -2689174.628  5633660.332
2017-08-10 06:25:19 [gobj:0] Warning: object YEL2 completed (NEU Eccentricity): 0.000  0.000  0.050
2017-08-10 06:25:19 [gobj:0] Warning: object YEL2 completed (XYZ Eccentricity): -0.010  -0.021  0.044
2017-08-10 06:25:19 [grec:0] Warning: object YEL2 completed (Receiver): SEPT POLARX4TR
2017-08-10 06:25:21 [main:0] READ: file://../data/obs/yel22370.16o   1.974 sec
2017-08-10 06:25:21 [main:0] Error: RUN BY not available!
2017-08-10 06:25:21 [main:0] Warning: LEAPSEC not available!
2017-08-10 06:25:21 [main:1] Single-thread summary: YEL2 started
2017-08-10 06:25:21 [gxtrqc:0] SITE: YEL2 YEL2.xtr
2017-08-10 06:25:21 [gxtrqc:1] Sync XTR step 2016-08-24 00:00:00 -> 2016-08-24 00:00:00
2017-08-10 06:25:21 [gxtrqc:0] Warning: no receiver setting.
2017-08-10 06:25:21 [gxtrqc:0] YEL2 header [1]:  0.000
2017-08-10 06:25:21 [gxtrqc:0] YEL2 obsstt [0]:  0.000
2017-08-10 06:25:21 [gxtrqc:0] YEL2 estima [1]:  0.000
2017-08-10 06:25:21 [gxtrqc:0] YEL2 observ [1]:  0.000
2017-08-10 06:25:22 [gxtrqc:0] YEL2 nbands [1]:  1.000
2017-08-10 06:25:22 [gxtrqc:0] YEL2 pieces [1]:  0.000
2017-08-10 06:25:24 [gxtrqc:0] YEL2 prepro [1]:  2.000
2017-08-10 06:25:24 [gxtrqc:0] YEL2 skyplt [1]:  0.000
2017-08-10 06:25:25 [gxtrqc:0] YEL2 mlpath [1]:  1.000
2017-08-10 06:25:25 [gxtrqc:0] YEL2 snoise [0]:  0.000
2017-08-10 06:25:25 [gxtrqc:0] YEL2 summar [0]:  0.000
2017-08-10 06:25:25 [main:0] Single-thread summary: YEL2   4.370 sec
2017-08-10 06:25:26 [main:0] total time:   6.545 sec
```

3.7   Quality check configuration

The QC setting is managed using attributes of the <qc> element:

```
<qc
   <!--quality check setting:
   use_health="[enumerate]"    # use of satellite health (position/statistics/all)
 * use_qcmode="[enumerate]"    # section pre-setting (thin/edit/summ/lite/full/kpis)
   sec_sum="[integer]"         # section for summary
   sec_hdr="[integer]"         # section for metadata information
   sec_est="[integer]"         # section for estimated values
   sec_obs="[integer]"         # section for observation statistics
   sec_gap="[integer]"         # section for data gap and small pieces
   sec_bnd="[integer]"         # section for availability frequency bands
   sec_pre="[integer]"         # section for detecting cycle slips and clock jumps
   sec_ele="[integer]"         # section for azimuths and elevations of visible satellites
   sec_mpx="[integer]"         # section for code multipath and signal noise estimates
   sec_snr="[integer]"         # section for signal-to-noise ratio observations
   sec_sat="[integer]"         # section for satellite health status and navigation records
   sec_kpi="[integer]"         # section for positioning performance indicators
   ele_cut="[double]"          # [deg] user elevation cut-off for expt/have ratio estimates
   pos_cut="[double]"          # [deg] elevation angle cut-off mask for positioning estimates
   pos_int="[integer]"         # [s] interval for single point positioning estimates
   pos_kin="[boolean]"         # enables processing of data from a receiver in kinematic mode
   sat_rec="[boolean]"         # satellite observations expected (true:any, false:signal)
   mpx_lim="[double]"          # [-] sigma multiplier for phase pre-processing in multipath
   mpx_nep="[integer]"         # [#] number of epochs used for code multipath estimates
   col_sat="[integer]"         # [#] number of columns for satellite-specific columns
   int_stp="[integer]"         # [s] interval for QC parameters sampling
   int_gap="[integer]"         # [s] interval for reporting data gaps
   int_pcs="[integer]"         # [s] interval for reporting small data pieces
 * ref_sig="[double]"          # [m] maximum sigma of reference coordinates
 * max_vpe="[double]"          # [m] maximum vertical position error
 * max_hpe="[double]"          # [m] maximum horizontal position error
 * max_dop="[double]"          # [m] maximum PDOP value in positioning performance
   -->
/>
```

The attributes for individual sections (**sec_...**) correspond with the structure of the Anubis' native XTR output, see Section 4, where the value defines a verbosity level for each individual part of the XTR format. The higher value, the more detailed information are reported in the log file. The zero verbosity deactivates any corresponding section for the reporting as well as the data processing. A negative value then deactivates only the reporting, and is used for triggering any mandatory process on the background (done by the software for some closely related operations).

The **use_qcmode** attribute represents individual sections' verbosity pre-setting, and it has been described in Section 2.

The **use_health** attribute, together with the **chk_record** from the <nav> element, controls the usage of unhealthy satellites in QC functioning within different QC sections, see Table 3.1.

Table 3.1: Satellites used in different sections when considering the health status

setting attribute	Value	Value	Value	Value
:nav:chk_health=	"false"	"true"	"true"	"true"
:qc:use_health=	n/a	"position"	"statistics"	"all"
QC functioning in section	Satellites	Satellites	Satellites	Satellites
Positioning	all used	healthy	healthy	healthy
Position KPIs	all used	healthy	healthy	healthy
Observations	all used	all used	healthy	healthy
Summary statistics	all used	all used	healthy	healthy
Azimuth/elevation	all used	all used	all used	healthy
Frequency bands	all used	all used	all used	healthy
Pre-processing	all used	all used	all used	all used
Code multipath	all used	all used	all used	all used
Signal-to-noise	all used	all used	all used	all used
Satellite information	all used	all used	all used	all used

The **int_stp** attribute defines a synchronous sampling of QC parameters reported within the most sections (**sec_ele** , **sec_bnd** , **sec_mpx** , **sec_snr**), e.g. necessary for sky-plots QC visualisations. The **pos_int** attribute then defines a specific sampling for a position estimates and positioning indicators (**sec_est** , **sec_kpi**).

Similarly, while the **ele_cut** attribute specifies a user elevation angle for ratio of expected and existing observations, the **pos_cut** affects the elevation cut-off mask in a single point positioning.

The **mpx_lim** attribute is used for a threshold definition for a cycle-slip detection within the code multipath calculation. Average code multipath is estimated exploiting a specific linear combination of a code and dual-frequency phase observations, see Section 4.9 requiring a sequence of n-epochs (**mpx_nep**) without phase cycle slips. The algorithm exploits the threshold **mpx_lim*sigma** for identifying a potential cycle slip, while the sigma has been estimated from the noise of a consecutive (and continuous) multipath observations.

The **max_vpe** , **max_hpe** , and **max_dop** attributes specify maximum values for the positioning performance indicators – vertical error, horizontal error, and PDOP value, respec-

tively. The **ref_sig** attribute specifies maximum sigma for coordinates allowed to be used as reference for KPI estimates.

When processing data from a kinematic receiver (**pos_kin="true"**), the position is estimated every epoch on the basis of code observations using the ionosphere-free linear combination. The processing time increases when analysing kinematic data, thus for a permanent station, the default value ("false") should be optimally used for estimating coordinates just once.

3.8 Site metadata configuration

A site metadata can be obtained either from the SINEX [7] or defined in the XML setting, see Section 2.2. The latter enables specification of the following metadata types:

- **id** (string) – unique site identifier (internal software identifier)
- **name** (string) – site name (4-char/9-char station name)
- **domes** (string) – site DOMES number
- **desc** (string) – site description
- **rec** (string) – site receiver name
- **ant** (string) – site antenna+radome name
- **priority** (string) – site metadata priority (GNUT—SINEX—RINEX)
- **x | y | z** (double) – site cartesian coordinates [m]
- **dn | de | du** (double) – site North, East, and Up eccentricities [m]
- **sx | sy | sz** (double) – site coordinates sigmas [m]

Selected metadata types (**rec, ant, x/y/z, dn/de/du, sx/sy/sz**) might be attributed with a time validity if defined using a time-dependent sub-element. Common (or static) metadata are set simply in the <site> parent element, i.e. without a time-dependent sub-element:

```
<site id="GOPE" name="GOPE00CZE" domes="11502M002" desc="Geodetic␣Observatory␣Pecny"
      x="3979316.1" y="1050312.6" z="4857067.4"  dn="0.0" de="0.0" du="0.1114"
      rec="TPS␣NETG3"
      ant="TPSCR.G3␣␣␣␣␣␣␣␣␣TPSH"
      priority="GNUT"
/>
```

Remark: the **id** *should be unique site identifier, possibly linked with the rec IDs specified within the* <gen><site> *element.*

In case of multiple metadata inputs (GNUT, SINEX, RINEX), a priority for station is given with the attribute **site["ID"]:priority** (default for all can be changed with **[""]**):

```
Anubis          -x Metadata.xml  \
      :inp:sinex Metadata.snx  \
      :site[""]:priority=GNUT  \
      :site["GOP7"]:priority=SINEX \
      :site["GOP6"]:priority=RINEX
/>
```

As long as any individual site metadata are valid for a particular time interval, it should be set using one or more time-dependent elements (<set>) with possibility to specify explicitly the **beg** and the **end** timestamp attributes, e.g.:

```
<site id="GOPE00CZE" name="GOPE00CZE" domes="11502M002" desc="Pecny,␣Ondrejov/Czech␣Republic"  x="3979316.1" y="1050312.5" z="4857067.1" >
  <set beg="2019-04-10T00:00:00"                            rec="TRIMBLE␣ALLOY"      ant="TPSCR.G3␣␣␣␣␣␣␣␣␣TPSH"  du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2018-08-08T00:00:00" end="2019-04-10T00:00:00" rec="TRIMBLE␣SPS855"     ant="TPSCR.G3␣␣␣␣␣␣␣␣␣TPSH"  du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2009-12-14T00:00:00" end="2018-08-08T00:00:00" rec="TPS␣NETG3"          ant="TPSCR.G3␣␣␣␣␣␣␣␣␣TPSH"  du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2006-07-14T00:00:00" end="2009-12-14T00:00:00" rec="ASHTECH␣Z18"        ant="TPSCR3_GGD␣␣␣␣␣␣␣CONE"  du="0.1114" dn="0.0000" de="0.0000" />
  <set beg="2006-01-24T00:00:00" end="2006-07-14T00:00:00" rec="ASHTECH␣Z18"        ant="ASH701946.3␣␣␣␣␣␣SNOW"  du="0.0467" dn="0.0000" de="0.0000" />
  <set beg="2000-10-04T00:00:00" end="2006-01-24T00:00:00" rec="ASHTECH␣Z18"        ant="ASH701946.3␣␣␣␣␣␣SNOW"  du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="2000-07-24T00:00:00" end="2000-10-04T00:00:00" rec="TRIMBLE␣4000SSE"    ant="TRM14532.00␣␣␣␣␣␣NONE"  du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1999-11-04T00:00:00" end="2000-07-24T00:00:00" rec="ASHTECH␣Z18"        ant="ASH701073.3␣␣␣␣␣␣SNOW"  du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1997-06-20T00:00:00" end="1999-11-04T00:00:00" rec="TRIMBLE␣4000SSE"    ant="TRM14532.00␣␣␣␣␣␣NONE"  du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1997-06-11T00:00:00" end="1997-06-20T00:00:00" rec="SPP␣GEOTRACER100"   ant="TRM14532.00␣␣␣␣␣␣NONE"  du="0.0464" dn="0.0000" de="0.0000" />
  <set beg="1995-05-13T00:00:00" end="1997-06-11T00:00:00" rec="TRIMBLE␣4000SSE"    ant="TRM14532.00␣␣␣␣␣␣NONE"  du="0.0464" dn="0.0000" de="0.0000" />
</site>
```

The time evolution metadata setting should facilitate validating or editing station metadata in the RINEX header. The IGS site-log files can be extracted into the G-Nut XML setting, and introduced as a common input for the QC operation or file editing, e.g. by combining 'static' XML setting (e.g. metadata) and 'dynamic' input/output configurations from the software command-line arguments.

4. XTR QC output

The XTR file is organised in sections consisting of related QC results corresponding to QC setting. Each section is introduced with an initialisation string ('#======'), the section name and its verbosity level. Each data lines starts with a unique 6-char keyword helpful for an easier QC results searching, handling, and filtering. It can be effectively used for convenient single-/multi-file QC data processing or plotting within a script.

Within each individual section, a level of output details can be controlled via its verbosity level (see Section 2.4). The verbosity level 1 (e.g. **sec_est=1**) provides usually a summary statistics only, while the verbosity 2 (e.g. **sec_est=2**) gives more concrete details including time-dependent QC results. The zero verbosity (e.g. **sec_est=0**) then corresponds to a switching off both processing and output related to the section (and could speed up the solution). The Anubis may however still trigger some operations on the background (e.g. **sec_est=-1**) if these are mandatory for other QC output. Next sections describe a content of the XTR file when using the highest verbosity.

4.1   Summary statistics

The summary section contains a collection of the most interesting statistics obtained during the QC processing. Constellation-/satellite-/signal-specific outputs are included together with an overall observation statistics. Basic characteristics over all GNSS global constellations are represented with a single summary line (with individual columns described below):

```
#====== Summary statistics
#TOTSUM First_Epoch_____ Last_Epoch_____ Hours_ Sample MinEle #_Expt #_Have %Ratio o/slps woElev Exp>10 Hav>10 %Rt>10
=TOTSUM 2020-12-31 00:00:00 2020-12-31 23:59:30  24.00  30.00   0.02 148580 120525  81.12    161   6201 114833 104528  91.03
```

> **Hours** – data window length in hours (estimated from a total number of epochs and sampling rate)
> **Sample** – data sampling interval (estimated from existing observations)
> **MinEle** – data minimum elevation angle (estimated from observation elevation angles, if possible)
> **#_Expt** – number of expected observations above the horizon (carrier-phase signal with a maximum count for each GNSS)
> **#_Have** – number of existing observations above the horizon (carrier-phase signal with a maximum count for each GNSS)
> **%Ratio** – ratio of existing/expected observations above the horizon (calculated from values given above)
> **o/slps** – number of carrier-phase observations per one cycle slip
> **woElev** – number of carrier-phase observations without available elevation angle (missing valid satellite ephemeris)
> **Exp>10** – number of expected observations above user mask (carrier-phase signal with a maximum count for each GNSS)
> **Hav>10** – number of existing observations above user mask (carrier-phase signal with a maximum count for each GNSS)
> **%Rt>10** – ratio of existing/expected observations above user elevation-cut of mask (calculated from values given above)

The line is identified with the keyword ('=TOTSUM'), which is followed with the first and the last observation epoch of available GNSS observations. Epoch strings are reported in the form – '%Y-%m-%d %D:%M:%S' , see Section 2.6.

A data window length ('Hours') is represented in hours when estimated from a total number of data epochs multiplied by an estimated sampling interval ('Sample').

The 'MinEle' is estimated as a minimum elevation angle over all collected observations (and constellations) provided the valid complete navigation data are available.

Total counts of carrier-phase observations are calculated as sums of representative signals for each individual GNSS (the one with a maximum number of carrier-phase observations). *Note:*

a traditional signal at the first frequency is often selected as it usually has a good signal-to-noise density and is transmitted by all satellites.

The *'#_Expt'* , *'#_Have'* and *'%Ratio'* values express expected number of observations, existing numbers of observations, and their ratio, respectively. All these values characterise the availability of observations above the horizon.

The *'Exp>10'* , *'Hav>10'* , and *'%Rt>10'* parameters represent similar counts as described above, however, reflecting a user elevation angle setting (10 degrees in the example).

The *'o/slps'* reflects a number of carrier-phase observations per single cycle slip when considering the same carrier-phase signals selected for individual constellations.

The *'woElev'* represents a number of satellite carrier-phase observations (count over a signal per GNSS) without valid ephemeris and, consequently, without calculated satellite positions and elevation/azimuth angles. A large value (> 200) indicates a problem with available navigation data (for one or more constellations).

Additional lines report a summary statistics on data quality and availability for each individual constellation (specified in the keyword prefix, e.g. *'=GPSSUM'*):

```
#GNSSUM 2020-12-31 00:00:00 Epoch_Statistics_ Excl_Epochs&Satellites_   CycleSlips/Interruptions_And_Other_Discontinuities Code_Multipath_Mean_...
#GNSSUM 2020-12-31 00:00:00 ExpEp HavEp UseEp xCoEp xPhEp xCoSv xPhSv  csAll  csEpo  csSat  csSig  nSlp  nJmp  nGap  nPcs   mp1   mp2   mp3  ...
=GPSSUM 2020-12-31 00:00:00 2880  2880  2880    0     0    360   400   471     0     171    271    29     0     0     0   26.0  24.3    -   ...
=GALSUM 2020-12-31 00:00:00 2880  2880  2880    0     0     15    15   228     0     200     22     6     0     0     0   19.7    -     -   ...
=GLOSUM 2020-12-31 00:00:00 2880  2880  2880    0     0   2997  3016   639     0     254    118   267     0     0     0   30.1  34.7    -   ...
=BDSSUM 2020-12-31 00:00:00 2880  2880  2880    0     0   2435  2488  3471     0     790    491  2190     0     0     0     -   46.8    -   ...
```

ExpEp – number of expected epochs (user pre-defined data window and sampling, otherwise estimated from data)

HavEp – number of available epochs (counting actual data epochs)

UseEp – number of usable epochs (criterion is 4 satellites in the epoch with dual-frequency data for each GNSS constellation)

xCoEp – number of epochs with less than 4 satellites providing dual-frequency pseudorange observations

xPhEp – number of epochs with less than 4 satellites providing dual-frequency carrier-phase observations

xCoSv – number of single-frequency pseudorange data only

xPhSv – number of single-frequency carrier-phase data only

csTot – number of total phase cycle slips or interruptions (corresponds to setting a new initial phase ambiguity)

csEpo – number of interruptions due to a missing epoch (counting all previously observed satellites)

csSat – number of interruptions due to a missing satellite at a specific epoch (when it is expected)

csSig – number of interruptions due to a missing signal from a single satellite (and others are available)

nSlp – number of identified phase cycle slips during a continuous tracking (counting for satellite and signal)

nJmp – number of identified receiver clock jumps (or an incoherence of code and phase observations)

nGap – number of data total gaps (according to the setting $int\_gap = 600$ in seconds)

nPcs – number of small data pieces (according to the setting $int\_pcs = 1800$ in seconds)

mpX – code multipath moving average RMS [cm] (mean value over all all signals at a given frequency band)

All *'expected'* values are estimated according to the window specified in the <beg>, <end> elements of the <gen> setting. If the window is not set up (by default), the interval is defined from the first and the last observation epochs. *Note: the window definition by user is highly recommended, else any missing epoch at the beginning/end of a data file will distort 'expected' parameters, and thus all the quantitative QC indicators.*

A minimum number of four satellites is requested in the single point position, hence a number of *'usable'* epochs (*'UseEp'*) represents those epochs providing dual-frequency observations from 4 satellites at least. Numbers of *'unusable'* epochs with few insufficient code and carrier-phase observations are reported in *'xCoEp'* and *'xPhEp'* parameters, and numbers of single-frequency code and carrier-phase measurements for individual satellites are reported in *'xCoSv'* and *'xPhSv'* parameters, respectively.

Pre-processing results are characterised by the number of receiver clock jumps 'nJmp' and numbers of carrier-phase interruptions such as cycle slips 'nSlp', discontinuities of observations for individual satellite 'csSat', epochs 'csEpo' and signals 'csSig'. Epoch-/signal-specific information details on data pre-processing can be found in Section 4.7.

Data gap may cause a significant problem in the processing as the carrier-phase observation continuity is lost, and all ambiguities need to be re-initialised, and the solution converges again. Overcoming data gaps or connecting short data pieces is possible only with advanced processing techniques. Numbers of data gaps 'nGap' and short data pieces 'nPcs' are reported according to user setting 3. Details are provided in Section 4.6.

Mean code multipath and noise characterise a quality of an observed signal and the site environment. Mean values per frequency band and constellation are reported here. A detailed description of such values (epoch-/satellite-/signal-specific) is given in the section 4.9.

A verbose summary gives characteristics for individual observations and each constellation:

```
#GNSxxx 2020-12-31 00:00:00 nSat ExpObs HavObs %Ratio Exp>10 Hav>10 %Rt>10 wo/Ele  Ele>0  Ele>5 Ele>10 Ele>15 ...
=GPSC1C 2020-12-31 00:00:00   31  33515  29709  88.64  25388  25365  99.91      0  29709  28452  25365  22200 ...
=GPSC2W 2020-12-31 00:00:00   31  33515  29048  86.67  25388  25364  99.91      0  29048  28144  25364  22200 ...
=GPSC2X 2020-12-31 00:00:00   23  33515  21706  64.77  25388  18553  73.08      0  21706  20852  18553  16178 ...
=GPSC5X 2020-12-31 00:00:00   16  33515  14952  44.61  25388  12728  50.13      0  14952  14304  12728  11135 ...
=GPSL1C 2020-12-31 00:00:00   31  33515  29542  88.15  25388  25364  99.91      0  29542  28370  25364  22200 ...
=GPSL2W 2020-12-31 00:00:00   31  33515  28971  86.44  25388  25363  99.90      0  28971  28102  25363  22200 ...
=GPSL2X 2020-12-31 00:00:00   23  33515  21706  64.77  25388  18553  73.08      0  21706  20852  18553  16178 ...
=GPSL5X 2020-12-31 00:00:00   16  33515  14952  44.61  25388  12728  50.13      0  14952  14304  12728  11135 ...
=GALC1X 2020-12-31 00:00:00   24  26275  22111  84.15  19650  19363  98.54      0  22111  21352  19363  17005 ...
=GALC5X 2020-12-31 00:00:00   24  26275  22067  83.98  19650  19363  98.54      0  22067  21341  19363  17005 ...
=GALC6X 2020-12-31 00:00:00   24  26275  22074  84.01  19650  19362  98.53      0  22074  21345  19362  17004 ...
=GALC7X 2020-12-31 00:00:00   24  26275  21950  83.54  19650  19363  98.54      0  21950  21313  19363  17005 ...
=GALC8X 2020-12-31 00:00:00   24  26275  22094  84.09  19650  19363  98.54      0  22094  21346  19363  17005 ...
=GALL1X 2020-12-31 00:00:00   24  26275  22101  84.11  19650  19363  98.54      0  22101  21351  19363  17005 ...
=GALL5X 2020-12-31 00:00:00   24  26275  22067  83.98  19650  19363  98.54      0  22067  21341  19363  17005 ...
=GALL6X 2020-12-31 00:00:00   24  26275  22065  83.98  19650  19362  98.53      0  22065  21344  19362  17004 ...
=GALL7X 2020-12-31 00:00:00   24  26275  21950  83.54  19650  19363  98.54      0  21950  21313  19363  17005 ...
=GALL8X 2020-12-31 00:00:00   24  26275  22094  84.09  19650  19363  98.54      0  22094  21346  19363  17005 ...
=GLOC1C 2020-12-31 00:00:00   24  27183  24345  89.56  21630  21329  98.61    942  23403  22571  20492  17659 ...
=GLOC1P 2020-12-31 00:00:00   24  27173  24293  89.40  21621  21312  98.57    932  23361  22543  20484  17659 ...
=GLOC2C 2020-12-31 00:00:00   21  27170  20927  77.02  21618  18641  86.23    929  19998  19406  17816  15481 ...
=GLOC2P 2020-12-31 00:00:00   21  27173  21336  78.52  21621  18925  87.53    932  20404  19768  18097  15694 ...
=GLOL1C 2020-12-31 00:00:00   24  27183  24345  89.56  21630  21329  98.61    942  23403  22571  20492  17659 ...
=GLOL1P 2020-12-31 00:00:00   24  27173  24287  89.38  21621  21311  98.57    932  23355  22540  20483  17658 ...
=GLOL2C 2020-12-31 00:00:00   21  27170  20924  77.01  21618  18639  86.22    929  19995  19404  17814  15480 ...
=GLOL2P 2020-12-31 00:00:00   21  27173  21313  78.43  21621  18917  87.49    932  20381  19752  18089  15693 ...
=BDSC2I 2020-12-31 00:00:00   42  61618  44565  72.32  48175  38495  79.91   5270  39295  37792  33811  29719 ...
=BDSC6I 2020-12-31 00:00:00   40  59396  42117  70.91  46200  36307  78.59   3048  39069  37573  33598  29506 ...
=BDSC7I 2020-12-31 00:00:00   11  56348  12797  22.71  43491  11129  25.59      0  12797  12405  11129   9954 ...
=BDSL2I 2020-12-31 00:00:00   42  61607  44537  72.29  48165  38472  79.88   5259  39278  37777  33798  29706 ...
=BDSL6I 2020-12-31 00:00:00   40  59371  42046  70.82  46178  36246  78.49   3023  39023  37532  33559  29467 ...
=BDSL7I 2020-12-31 00:00:00   11  56348  12770  22.66  43491  11103  25.53      0  12770  12378  11103   9928 ...
```

nSat – number of satellites transmitting the signal

ExpObs – number of expected observations above the horizon (user-defined window, data sampling, visible satellites)

HavObs – number of available observations above the horizon (counting actual observations for tracked satellites)

%Ratio – ratio of existing/expected observations above the horizon (calculated from above values)

wo/Elev – number of observations without knowing an elevation angle (missing or invalid satellite ephemeris)

Exp>10 – number of expected observations above user mask (e.g. above 10-deg elevation angle cut-off)

Hav>10 – number of existing observations above user mask (e.g. above 10-deg elevation angle cut-off)

%Rt>10 – ratio of existing/expected observations above the user elevation angle -cut-off (using values above)

Ele> X – number of existing observations above the X-deg elevation angle

Each line is introduced with a keyword generated from GNSS system and RINEX observation code, and provides similar statistics as described in the 'TOTSUM' line. The 'nSat' expresses counts of satellites tracking a particular signal, e.g. the third GPS frequency signals (C5/L5) are obviously transmitted by modern satellites only (16). The 'wo/Ele' represents numbers

of observations without calculated elevation angle (e.g. BeiDou), and higher numbers may indicate an increased uncertainty for *'expected'* counts.

Finally, the summary reports times of satellite ascending/descending for horizon (*'SKYxxx'*) and a user elevation angle mask (*'MSKxxx'*):

```
#SKYxxx Ascending_Horizon__ Descending_Horizon_ Time[h] ExptObs
=SKYC02 2020-12-31 00:00:00 2020-12-31 23:59:30  23.992    2879
=SKYC05 2020-12-31 00:00:00 2020-12-31 23:59:30  23.992    2879
=MSKC05 2020-12-31 00:00:00 2020-12-31 23:59:30  23.992    2879
=SKYC06 2020-12-31 00:00:00 2020-12-31 07:01:21   7.023     843
=SKYC06 2020-12-31 22:47:54 2020-12-31 23:59:30   1.193     143
=MSKC06 2020-12-31 00:00:00 2020-12-31 05:43:24   5.723     687
=MSKC06 2020-12-31 23:38:18 2020-12-31 23:59:30   0.353      42
=SKYC07 2020-12-31 06:36:53 2020-12-31 14:52:36   8.262     991
=MSKC07 2020-12-31 07:30:26 2020-12-31 13:30:11   5.996     720
=SKYC08 2020-12-31 14:38:50 2020-12-31 23:05:10   8.439    1013
=MSKC08 2020-12-31 15:22:32 2020-12-31 21:52:50   6.505     781
=SKYC09 2020-12-31 00:00:00 2020-12-31 11:20:08  11.336    1360
=SKYC09 2020-12-31 23:23:48 2020-12-31 23:59:30   0.595      71
=MSKC09 2020-12-31 00:12:23 2020-12-31 09:31:59   9.327    1119
........
```

A satellite visibility interval (*'Time[h]'*) and counts of expected observations (*'ExptObs'*) complete each record. The C02 and C05 are BeiDou geostationary satellites visible above the horizon whole day, but C05 above the user mask as the station is located in Europe.

4.2 Header information

Selected station metadata are stored in the header of the RINEX observation file, and their correctness is very critical for precise GNSS application. Data analysts usually perform an independent cross-validation of the RINEX metadata against external metadata source. The International GNSS Service (IGS, https://www.igs.org) has developed a station log file for the purpose of a long-term maintenance of all changes at GNSS site instrumentation, environment and other information.

Currently, Anubis is able to check the RINEX header metadata against external information provided either in Anubis configuration or from the SINEX input file. Anubis is not capable of handling the IGS site-log format, but a support of its future replacement – GeodesyML (http://geodesyml.org/) is on Anubis' TODO list. A user metadata can be utilised for a cross-validation of RINEX header files. Both RINEX header and user metadata are reported in the XTR file within the **QC mode** and this can be archived or exploited in a post-processing. However, header metadata inconsistencies can directly be reported as warnings and errors on the standard error output when using the **thin mode** , see 2.2.

The RINEX header metadata reported in XTR output are introduced with:

```
#====== Header information
#RNXHDR 2020-12-31 00:00:00 _RINEX_HEADER_____ _RINEX_HEADER_____ _RINEX_HEADER_____
=RNXVER 2020-12-31 00:00:00 3.03                    M                       2021-01-01 00:16:35
=RNXPGM 2020-12-31 00:00:00 ALLOY 5.45&OBSRINEX     RIGTC, GO PECNY
=RNXAGE 2020-12-31 00:00:00 GOP OC STAFF            RIGTC, GO PECNY
```

  '=*RNXVER'* – RINEX file version, observation data type, and the date of file creation.
  '=*RNXPGM'* – program name and the operator who created the RINEX.
  '=*RNXAGE'* – observer name and the agency name.

The second part contains RINEX header (RINEX_HEADER) and user metadata (USER_REQUEST) provided in parallel for easy grepping through an archive of XTR files:

```
#RNXHDR 2020-12-31 00:00:00 _RINEX_HEADER_____ _USER_REQUEST_____
=BEGEND 2020-12-31 00:00:00 2020-12-31 00:00:00    xxxx-xx-xx xx:xx:xx   2020-12-31 00:00:00    2020-12-31 23:59:30
=INTHDR 2020-12-31 00:00:00 30.000                                     30.000
=MARKER 2020-12-31 00:00:00 GOPE 11502M002                            GOPE00CZE 11502M002
=RECEIV 2020-12-31 00:00:00 TRIMBLE ALLOY       5.45          40016   TRIMBLE ALLOY
=ANTENN 2020-12-31 00:00:00 TPSCR.G3      TPSH30161                   TPSCR.G3        TPSH
```

'=BEGEND' – start and end epoch of the file.
'=INTHDR' – sampling interval.
'=MARKER' – marker name (site name).
'=RECEIV' – receiver type, version and the firmware.
'=ANTENN' – antenna type, radome and serial number.

The last part consists of coordinate and eccentricity metadata:

```
#RNXHDR 2020-12-31 00:00:00 _RINEX_HEADER_____ _USER_REQUEST_____
=XYZAPR 2020-12-31 00:00:00    3979316.4389   1050312.2534   4857066.9036    3979316.4389   1050312.2534   4857066.9036
=XYZECC 2020-12-31 00:00:00          0.0694         0.0183         0.0852          0.0694         0.0183         0.0852
=ENUECC 2020-12-31 00:00:00          0.0000         0.0000         0.1114          0.0000         0.0000         0.1114
```

'=XYZAPR' – approximate site coordinates.
'=XYZECC' – XYZ eccentricities (from marker to antenna reference point).
'=ENUECC' – ENU eccentricities (from marker to antenna reference point).

4.3 Estimated values

The complex overview on quality of GNSS observations and navigation messages can be obtained with data analysis. Epoch-wise single point positioning exploiting code pseudorange observations only is used to estimate site coordinates and receiver clock offsets.

The section for estimated values contains summary information:

```
#====== Estimated values
=PERIOD 2020-12-31 00:00:00 2020-12-31 00:00:00    2020-12-31 23:59:30
=SAMPLE 2020-12-31 00:00:00 30.000
=XYZGPS 2020-12-31 00:00:00    3979316.8216   1050312.7641   4857068.3343    1.4    0.9    1.5    96    0
=XYZGAL 2020-12-31 00:00:00    3979315.9539   1050312.7662   4857067.3737    0.7    0.4    0.7    96    0
=XYZGLO 2020-12-31 00:00:00    3979315.6761   1050311.1717   4857067.1897    3.5    3.3    7.5    96    0
=XYZBDS 2020-12-31 00:00:00    3979322.6273   1050310.6425   4857071.7062    6.1    3.6   13.2    96    0
=BLHGPS 2020-12-31 00:00:00   49.913706685   14.785623749    594.0187    1.0    1.2    1.9    96    0
=BLHGAL 2020-12-31 00:00:00   49.913706891   14.785626860    592.7439    0.6    0.6    0.8    96    0
=BLHGLO 2020-12-31 00:00:00   49.913710472   14.785606383    592.1682    4.0    4.6    7.5    96    0
=BLHBDS 2020-12-31 00:00:00   49.913691320   14.785574563    599.8646    4.5    4.7   13.9    96    0
```

'=PERIOD' – first and last observations of available data.
'=SAMPLE' – estimated sampling rate (using a statistical approach).
'=XYZGPS' – mean XYZ coordinates estimated using GPS system, uncertainties, numbers of used and excluded epochs.
'=BLHGPS' – mean BLH coordinates estimated using GPS system, uncertainties, numbers of used and excluded epochs.

Earth-centred, Earth-fixed (XYZ) and ellipsoidal (BLH) site coordinates are estimated from individual GNSS constellations. Mean values are reported together with uncertainties and numbers of samples/outliers (96 position epochs in a 15-min sampling defined in setting 3.7).

A higher verbosity gives epoch-wise coordinate estimates for user setting:

```
#POSGNS 2020-12-31 00:00:00        X [m]        Y [m]        Z [m]       B [deg]      L [deg]    H [m]   GDOP ... REC_CLK[m] #Sat #Excl
 POSGPS 2020-12-31 00:00:00   3979317.8132  1050313.7215  4857069.2804  49.913703888  14.785633113  595.5173   1.7 ... 130455.0   11    0
 POSGPS 2020-12-31 00:15:00   3979317.7813  1050312.8226  4857071.5150  49.913718613  14.785621126  597.0594   1.6 ... 130455.8   11    0
 POSGPS 2020-12-31 00:30:00   3979318.2924  1050313.8926  4857070.2627  49.913706087  14.785633713  596.5954   1.7 ... 130455.6   12    2
 POSGPS 2020-12-31 00:45:00   3979320.1232  1050314.1277  4857071.6890  49.913701756  14.785630374  598.8651   1.7 ... 130457.6   11    1
 POSGPS 2020-12-31 01:00:00   3979319.2242  1050314.4574  4857070.1991  49.913698531  14.785638005  597.2196   1.4 ... 130456.7   11    0
 POSGPS 2020-12-31 01:15:00   3979319.0981  1050314.7318  4857068.3179  49.913687998  14.785642147  595.6872   1.9 ... 130455.2   11    2
 POSGPS 2020-12-31 01:30:00   3979317.4467  1050314.0329  4857068.7116  49.913702485  14.785638607  594.9051   1.5 ... 130454.5   11    0
 ...

 POSGAL 2020-12-31 00:00:00   3979316.4862  1050312.7723  4857067.8515  49.913706107  14.785625050  593.4419   1.8 ... 130456.8    9    0
 POSGAL 2020-12-31 00:15:00   3979315.2428  1050312.9022  4857066.8290  49.913708228  14.785631216  591.9068   1.8 ... 130456.0   10    1
 POSGAL 2020-12-31 00:30:00   3979316.2332  1050312.7339  4857068.0952  49.913709267  14.785625432  593.4645   1.8 ... 130457.1    9    0
 POSGAL 2020-12-31 00:45:00   3979315.2297  1050312.3404  4857068.0463  49.913716348  14.785623700  592.7376   1.8 ... 130456.6    9    0
 POSGAL 2020-12-31 01:00:00   3979314.6228  1050312.1410  4857067.4003  49.913716994  14.785623173  591.8328   2.5 ... 130456.0    9    1
 POSGAL 2020-12-31 01:15:00   3979315.8496  1050312.7399  4857068.1353  49.913712040  14.785626876  593.2574   3.0 ... 130457.0    7    0
 POSGAL 2020-12-31 01:30:00   3979315.9424  1050312.7756  4857068.3518  49.913712613  14.785627026  593.4866   2.8 ... 130457.3    7    0
 ...
```

'X', 'Y', 'Z' – cartesian coordinates estimated for the given GNSS constellation and time epoch.

'B', 'L', 'H' – ellipsoidal coordinates estimated for the given GNSS constellation and time epoch.

'GDOP', 'PDOP', 'HDOP', 'VDOP' – geometric, position, horizontal and vertical dilution of precision (DOP).

'REC_CLK[m]' – receiver clock offsets given in length unit (meters).

'#Sat', '#Excl' – numbers of all and excluded satellites.

4.4   Observation types

A quantitative statistics of actually observed data is provided within this section. *Note: actual data are product of filtering applied during data decoding.* The below example shows results stemming from the verbosity ***sec_obs=1*** :

```
#====== Observation types
=GNSSYS 2020-12-31 00:00:00      4 GPS GAL GLO BDS

=GPSSAT 2020-12-31 00:00:00     31 G01 G02 G03 G04 G05 G06 G07 G08 G09 G10   - G12 G13 G14 G15 G16 G17 G18 G19 G20 G21 G22 G23 G24 G25 G26 G27 G28 ...
=GALSAT 2020-12-31 00:00:00     24 E01 E02 E03 E04 E05   - E07 E08 E09   - E11 E12 E13 E14 E15   -   - E18 E19   - E21   -   - E24 E25 E26 E27   - ...
=GLOSAT 2020-12-31 00:00:00     24 R01 R02 R03 R04 R05 R06 R07 R08 R09 R10 R11 R12 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22 R23 R24   -   -   -   - ...
=BDSSAT 2020-12-31 00:00:00     42   -   -   -   - C05 C06 C07 C08 C09 C10 C11 C12 C13 C14   - C16   -   - C19 C20 C21 C22 C23 C24 C25 C26 C27 C28 ...

=BDSHDR 2020-12-31 00:00:00     12 C5X L5X S5X C2I L2I S2I C7I L7I S7I C6I L6I S6I
=GALHDR 2020-12-31 00:00:00     15 C1X L1X S1X C5X L5X S5X C7X L7X S7X C8X L8X S8X C6X L6X S6X
=GPSHDR 2020-12-31 00:00:00     12 C1C L1C S1C C2W L2W S2W C2X L2X S2X C5X L5X S5X
=IRNHDR 2020-12-31 00:00:00      3 C5A L5A S5A
=QZSHDR 2020-12-31 00:00:00     15 C1C L1C S1C C1X L1X S1X C1Z L1Z S1Z C2X L2X S2X C5X L5X S5X
=GLOHDR 2020-12-31 00:00:00     12 C1C L1C S1C C1P L1P S1P C2C L2C S2C C2P L2P S2P
=SBSHDR 2020-12-31 00:00:00      3 C1C L1C S1C
=GPSOBS 2020-12-31 00:00:00     12 C1C C2W C2X C5X L1C L2W L2X L5X S1C S2W S2X S5X
=GALOBS 2020-12-31 00:00:00     15 C1X C5X C6X C7X C8X L1X L5X L6X L7X L8X S1X S5X S6X S7X S8X
=GLOOBS 2020-12-31 00:00:00     12 C1C C1P C2C C2P L1C L1P L2C L2P S1C S1P S2C S2P
=BDSOBS 2020-12-31 00:00:00      9 C2I C6I C7I L2I L6I L7I S2I S6I S7I
```

'=GNSSYS' – count and list of observed systems.

'=GPSSAT' – count and list of observed satellites for the GPS system.

'=GPSHDR' – count and list of observation types for the GPS system (from a file header).

'=GPSOBS' – count and list of observation types for the GPS system (from actual data).

It may happen that the measurements reported in a file header are actually not present in RINEX data. It can be seen from a comparison of the '=GPSHDR' and '=GPSOBS' lines.

Nowadays, series of satellite block types carrying different equipment are common to GNSS constellation. All signals don't have to be transmitted by all satellites, e.g. the L5 frequency is transmitted by modern GPS satellites only at the moment. A signal selection thus becomes important for any data analysis, in particular for a method utilising difference data. By increasing the section verbosity, more information is provided about available satellites and signals from data of each GNSS constellation.

```
GPSC1C 2020-12-31 00:00:00     G01 G02 G03 G04 G05 G06 G07 G08 G09 G10   - G12 G13 G14 G15 G16 G17 G18 G19 G20 G21 G22 G23 G24 G25 G26 G27 G28 ...
GPSC2W 2020-12-31 00:00:00     G01 G02 G03 G04 G05 G06 G07 G08 G09 G10   - G12 G13 G14 G15 G16 G17 G18 G19 G20 G21 G22 G23 G24 G25 G26 G27 G28 ...
GPSC2X 2020-12-31 00:00:00     G01   - G03 G04 G05 G06 G07 G08 G09 G10   - G12   - G14 G15   - G17 G18   -   -   - G23 G24 G25 G26 G27   - ...
GPSC5X 2020-12-31 00:00:00     G01   - G03 G04   - G06   - G08 G09 G10   -   -   - G14   -   -   - G18   -   -   - G23 G24 G25 G26 G27   - ...
......
GALL1X 2020-12-31 00:00:00     E01 E02 E03 E04 E05   - E07 E08 E09   - E11 E12 E13 E14 E15   -   - E18 E19   - E21   -   - E24 E25 E26 E27   - ...
GALL5X 2020-12-31 00:00:00     E01 E02 E03 E04 E05   - E07 E08 E09   - E11 E12 E13 E14 E15   -   - E18 E19   - E21   -   - E24 E25 E26 E27   - ...
GALL6X 2020-12-31 00:00:00     E01 E02 E03 E04 E05   - E07 E08 E09   - E11 E12 E13 E14 E15   -   - E18 E19   - E21   -   - E24 E25 E26 E27   - ...
GALL7X 2020-12-31 00:00:00     E01 E02 E03 E04 E05   - E07 E08 E09   - E11 E12 E13 E14 E15   -   - E18 E19   - E21   -   - E24 E25 E26 E27   - ...
GALL8X 2020-12-31 00:00:00     E01 E02 E03 E04 E05   - E07 E08 E09   - E11 E12 E13 E14 E15   -   - E18 E19   - E21   -   - E24 E25 E26 E27   - ...
......
GLOS1C 2020-12-31 00:00:00     R01 R02 R03 R04 R05 R06 R07 R08 R09 R10 R11 R12 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22 R23 R24   -   -   -   - ...
GLOS1P 2020-12-31 00:00:00     R01 R02 R03 R04 R05 R06 R07 R08 R09 R10 R11 R12 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22 R23 R24   -   -   -   - ...
GLOS2C 2020-12-31 00:00:00     R01 R02 R03 R04 R05   - R07 R08 R09   - R11 R12 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22   - R24   -   -   -   - ...
GLOS2P 2020-12-31 00:00:00     R01 R02 R03 R04 R05   - R07 R08 R09   - R11 R12 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22   - R24   -   -   -   - ...
......
BDSD2I 2020-12-31 00:00:00       -   -   -   - C05 C06 C07 C08 C09 C10 C11 C12 C13 C14   - C16   -   - C19 C20 C21 C22 C23 C24 C25 C26 C27 C28 ...
BDSD6I 2020-12-31 00:00:00       -   -   -   - C05 C06 C07 C08 C09 C10 C11 C12 C13 C14   - C16   -   - C19 C20 C21 C22 C23 C24 C25 C26 C27 C28 ...
BDSD7I 2020-12-31 00:00:00       -   -   -   - C05 C06 C07 C08 C09 C10 C11 C12 C13 C14   - C16   -   -   -   -   -   -   -   -   -   -   -   - ...
......
```

'GPSC1C' – list of GPS satellites for code observations (C1C etc.).

'GALL1X' – list of Galileo satellites for phase observations (L1X etc.).

'GLOS1C' – list of GLONASS satellites for signal-to-noise density observations (C1C etc.).

'BDSD2I' – list of BeiDou satellites for Doppler observations (D2I etc.).

```
G01OBS 2020-12-31 00:00:00      12 C1C C2W C2X C5X L1C L2W L2X L5X S1C S2W S2X S5X
G02OBS 2020-12-31 00:00:00       6 C1C C2W L1C L2W S1C S2W
......
G05OBS 2020-12-31 00:00:00       9 C1C C2W C2X L1C L2W L2X S1C S2W S2X
G06OBS 2020-12-31 00:00:00      12 C1C C2W C2X C5X L1C L2W L2X L5X S1C S2W S2X S5X
......
E01OBS 2020-12-31 00:00:00      15 C1X C5X C6X C7X C8X L1X L5X L6X L7X L8X S1X S5X S6X S7X S8X
E02OBS 2020-12-31 00:00:00      15 C1X C5X C6X C7X C8X L1X L5X L6X L7X L8X S1X S5X S6X S7X S8X
......
R01OBS 2020-12-31 00:00:00      12 C1C C1P C2C C2P L1C L1P L2C L2P S1C S1P S2C S2P
R02OBS 2020-12-31 00:00:00      12 C1C C1P C2C C2P L1C L1P L2C L2P S1C S1P S2C S2P
R03OBS 2020-12-31 00:00:00      12 C1C C1P C2C C2P L1C L1P L2C L2P S1C S1P S2C S2P
R04OBS 2020-12-31 00:00:00      12 C1C C1P C2C C2P L1C L1P L2C L2P S1C S1P S2C S2P
R05OBS 2020-12-31 00:00:00      12 C1C C1P C2C C2P L1C L1P L2C L2P S1C S1P S2C S2P
R06OBS 2020-12-31 00:00:00       6 C1C C1P L1C L1P S1C S1P
......
C05OBS 2020-12-31 00:00:00       9 C2I C6I C7I L2I L6I L7I S2I S6I S7I
C06OBS 2020-12-31 00:00:00       9 C2I C6I C7I L2I L6I L7I S2I S6I S7I
......
C19OBS 2020-12-31 00:00:00       6 C2I C6I L2I L6I S2I S6I
C20OBS 2020-12-31 00:00:00       6 C2I C6I L2I L6I S2I S6I
......
```

'=G01OBS' – list of observations collected from the GPS satellite (G01 etc.).
'=E01OBS' – list of observations collected from the Galileo satellite (E01 etc.).
'=R01OBS' – list of observations collected from the GLONASS satellite (R01 etc.).
'=C01OBS' – list of observations collected from the BeiDou satellite (C01 etc.).

4.5 Bands available

A traditional strategy for GNSS data processing is based on analysing dual-frequency observations for largely eliminating ionospheric delay. Anubis thus considers valid satellite observations whenever at least two frequency carrier-phase and code observations are available. This section reports availability of frequencies and focuses on a lack of dual-frequency data. The section report consists of:

```
#====== Band available
#GNSxEP 2020-12-31 00:00:00 FewBand x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
=GPSCEP 2020-12-31 00:00:00    360  99  98 100  99  99 100  99 100 100 100   - 100  89  99  98  95  99 100  98  96  95  98  99 100 100 100  99  93 ...
=GPSLEP 2020-12-31 00:00:00    400  99  97 100  99  99 100  98 100 100  99   - 100  89  99  97  95  99  99  98  96  95  98  99 100  99 100  99  93 ...
=GALCEP 2020-12-31 00:00:00     15 100 100 100 100 100   - 100 100  99   - 100 100 100  99 100   -   - 100 100   - 100   -   - 100 100 100 100   - ...
=GALLEP 2020-12-31 00:00:00     15 100 100 100 100 100   - 100 100  99   - 100 100 100  99 100   -   - 100 100   - 100   -   - 100 100 100 100   - ...
=GLOCEP 2020-12-31 00:00:00   2997  99  99  99  99  99   0  99  99  99   0  99  99 100  99  99  99  99  99 100 100  99  88   0  99   -   -   -   - ...
=GLOLEP 2020-12-31 00:00:00   3016  99  99  99  99  99   0  99  99  99   0  99  99 100  99  99  99  99  99 100  99  87   0  99   -   -   -   - ...
=BDSCEP 2020-12-31 00:00:00   2435   -   -   -   -  92 100 100 100 100 100 100  99 100 100   - 100   -   - 100 100  99  99  99  99 100  99 100 100 ...
=BDSLEP 2020-12-31 00:00:00   2488   -   -   -   -  91 100 100 100 100 100 100  99 100 100   - 100   -   - 100 100  99  99  99  99 100  99 100 100 ...
```

'=GPSCEP' – percentage of dual-frequency code observations available for individual GPS satellites.
'=GPSLEP' – percentage of dual-frequency carrier-phase observations available for individual GPS satellites.
'FewBand' – count of satellites with single-frequency code and phase observations (see also 'xCoSv' and 'xPhSv'), 4.1.

A higher verbosity reports actual frequency bands available at given sampling epochs:

```
#NxBAND 2020-12-31 00:00:00 nSatell x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
GPSCBN 2020-12-31 00:00:00     11   -   -   -   -   -   -   2   3   -   3   -   -   -   -   2   2   -   3   -   2   2   -   3   -   -   3   3   - ...
GPSCBN 2020-12-31 00:15:00     11   -   -   -   -   -   -   2   3   -   3   -   -   -   -   2   2   -   3   -   2   2   -   3   -   -   3   3   - ...
GPSCBN 2020-12-31 00:30:00     12   -   -   -   -   -   -   2   3   -   3   -   -   -   -   2   2   -   3   -   2   2   -   3   -   -   3   3   - ...
GPSCBN 2020-12-31 00:45:00     11   -   -   -   -   -   -   -   3   -   3   -   -   -   -   2   2   -   3   -   2   2   -   3   -   -   3   3   - ...
......
GALLBN 2020-12-31 00:00:00      9   5   -   -   -   -   -   5   5   -   -   -   5   5   -   -   -   -   -   -   -   -   -   5   -   5   -   -   - ...
GALLBN 2020-12-31 00:15:00     10   5   -   -   -   -   -   5   5   -   -   -   5   5   5   -   -   -   -   -   -   -   -   5   -   5   -   -   - ...
GALLBN 2020-12-31 00:30:00      9   5   -   -   -   -   -   5   5   -   -   -   5   -   5   -   -   -   -   -   -   -   -   5   -   5   -   -   - ...
GALLBN 2020-12-31 00:45:00      9   5   -   -   -   -   -   5   5   -   -   -   5   -   5   -   -   -   -   -   -   -   -   5   -   5   -   -   - ...
......
GLOCBN 2020-12-31 00:00:00      9   2   2   -   -   -   -   2   -   -   -   -   2   2   2   2   -   -   -   -   -   1   2   -   -   -   -   - ...
GLOCBN 2020-12-31 00:15:00     10   2   2   -   -   -   -   2   -   -   -   -   2   2   2   2   2   -   -   -   -   1   2   -   -   -   -   - ...
GLOCBN 2020-12-31 00:30:00     10   2   2   -   -   -   -   2   -   -   -   -   2   2   2   2   2   -   -   -   -   1   2   -   -   -   -   - ...
GLOCBN 2020-12-31 00:45:00     10   2   2   -   -   -   -   2   -   -   -   -   2   2   2   2   2   -   -   -   -   1   2   -   -   -   -   - ...
......
BDSLBN 2020-12-31 00:00:00     18   -   -   -   -   3   3   -   -   3   -   3   3   3   -   -   3   -   -   2   -   2   2   -   2   2   -   -   - ...
BDSLBN 2020-12-31 00:15:00     19   -   -   -   -   3   3   -   -   3   -   3   3   3   -   -   3   -   -   2   -   2   2   2   2   2   -   -   - ...
BDSLBN 2020-12-31 00:30:00     19   -   -   -   -   3   3   -   -   3   -   3   3   3   -   -   3   -   -   2   -   2   2   2   2   2   -   -   - ...
BDSLBN 2020-12-31 00:45:00     19   -   -   -   -   3   3   -   -   3   -   3   3   3   -   -   3   -   -   2   -   2   2   2   2   2   -   -   - ...
```

'=GPSCBN' – count of frequency bands for code observations available at a given epoch for individual GPS satellites.
'=GPSLBN' – count of frequency bands for carrier-phase observations available at a given epoch for individual GPS satellites.
'nSatell' – count of satellites providing dual-frequency observations at a given epoch.

Epochs with a single-frequency data or number of satellites less than 4 are eventually reported:

```
#GNSBND 2020-12-31 00:00:00 C<2 L<2 nSv SvList
 GPSXBN 2020-12-31 00:30:00   1   1  12 G32
 GPSXBN 2020-12-31 00:31:00   0   1  12 G32
 GPSXBN 2020-12-31 00:35:00   0   1  12 G07
 GPSXBN 2020-12-31 00:36:30   0   1  12 G07
 GPSXBN 2020-12-31 01:05:00   1   2  12 G15 G22
 GALXBN 2020-12-31 10:54:30   1   1   8 E14
 GALXBN 2020-12-31 11:44:30   1   1   8 E14
 GALXBN 2020-12-31 11:45:00   1   1   8 E14
 GALXBN 2020-12-31 14:17:00   1   1   8 E31
 GALXBN 2020-12-31 22:32:30   1   1   8 E36
 GALXBN 2020-12-31 23:35:30   1   1   8 E09
 GALXBN 2020-12-31 23:36:00   1   1   8 E09
 GLOXBN 2020-12-31 00:00:00   1   1   9 R23
 ......
```

'$C < 2$', '$L < 2$' – count of satellites at a given epoch providing only single-frequency code and phase observations.

'nSv' – count of satellites available in a given epoch.

'$SvList$' – list of satellites not providing dual-/multi-frequency observations.

4.6 Gaps & Pieces

GNSS observations with many data gaps are more difficult to process because some parameters, e.g. ambiguities, need to be reinitialised. Also within a short data block, parameters cannot be estimated precisely due to their long convergence (e.g. in Precise Point Positioning). This section reports large data gaps and short data pieces:

```
#====== Gaps & Pieces
#GAPLST 2020-12-31 begTime     endTime   >600s
 GAPLST 2020-12-31 00:22:00   00:39:00    1020
 GAPLST 2020-12-31 01:05:00   01:22:00    1020
 GAPLST 2020-12-31 02:02:30   02:24:00    1290

#PCSLST 2020-12-31 begTime     endTime  <1800s
 PCSLST 2020-12-31 00:00:00   00:22:00    1320
 PCSLST 2020-12-31 00:39:00   01:05:00    1560

#SMPLST 2020-12-31 00:00:00   sampling    count
 SMPLST 2020-12-31 00:00:00         30     2744
 SMPLST 2020-12-31 00:00:00         90        2
 SMPLST 2020-12-31 00:00:00        120        1
 SMPLST 2020-12-31 00:00:00        420        1
 SMPLST 2020-12-31 00:00:00       1020        2
 SMPLST 2020-12-31 00:00:00       1290        1
```

'$GAPLST$' – list of large data gaps (defined by user).

'$PCSLST$' – list of small data pieces (defined by user).

'$SMPLST$' – list of frequencies of intervals between consecutive observations.

The first time ('$begTime$') and the last time ('$endTime$') of a data gap or small data piece are reported followed by the length in seconds. The length criteria is defined by the user and written in the header line. The last block summarises frequencies of intervals between consecutive epochs collected from observation data. The sampling rate is then estimated as the most frequent interval between observations.

4.7 Pre-processing results

The observation pre-processing is an essential part of precise GNSS analysis. It consists of detecting receiver clock jumps and detecting and possibly repairing carrier-phase cycle slips.

Along with real cycle slips (*'CS_Slip'*), i.e. jumps in carrier-phase observations, additional three kinds of discontinuities are defined and counted: due to missing epochs (*'CS_Epoch'*), due to missing satellites *'CS_Satell'* , and due to missing signal (*'CS_Signal'*) when considering two consecutive epochs, see Figure 4.1.
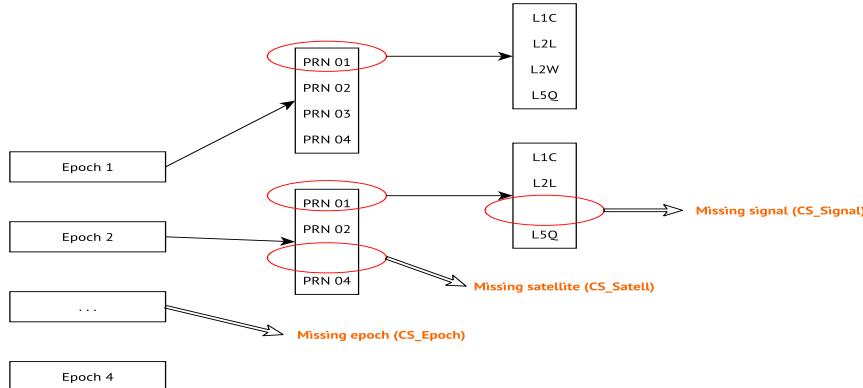
Figure 4.1: Signal discontinuities (epoch, satellite, and observation).

A minimum output (**sec_pre=1**) reports a summary over all constellations and signals:

```
#====== Preprocessing results
#GNSPRP 2020-12-31 00:00:00     CS_Total     CS_Slip     CS_Epoch     CS_Satell     CS_Signal
=GPSPRP 2020-12-31 00:00:00          471          29            0           171           271
=GALPRP 2020-12-31 00:00:00          228           6            0           200            22
=GLOPRP 2020-12-31 00:00:00          639         267            0           254           118
=BDSPRP 2020-12-31 00:00:00         3471        2190            0           790           491

#GNSxxx 2020-12-31 00:00:00     CS_Total     CS_Slip     CS_Epoch     CS_Satell     CS_Signal
=GPSL1C 2020-12-31 00:00:00          208          12            0            70           126
=GPSL2W 2020-12-31 00:00:00          152          12            0            27           113
=GPSL2X 2020-12-31 00:00:00           62           3            0            30            29
=GPSL5X 2020-12-31 00:00:00           49           2            0            44             3
=GALL1X 2020-12-31 00:00:00           49           1            0            46             2
=GALL5X 2020-12-31 00:00:00           46           2            0            42             2
=GALL6X 2020-12-31 00:00:00           44           0            0            34            10
=GALL7X 2020-12-31 00:00:00           43           1            0            37             5
=GALL8X 2020-12-31 00:00:00           46           2            0            41             3
=GLOL1C 2020-12-31 00:00:00          167          66            0           101             0
=GLOL1P 2020-12-31 00:00:00          158          69            0            67            22
=GLOL2C 2020-12-31 00:00:00          123          56            0            32            35
=GLOL2P 2020-12-31 00:00:00          191          76            0            54            61
=BDSL2I 2020-12-31 00:00:00         1069         669            0           376            24
=BDSL6I 2020-12-31 00:00:00         1290         775            0           235           280
=BDSL7I 2020-12-31 00:00:00         1112         746            0           179           187
```

'GPSPRP' – count of carrier-phase interruptions counted over all signals for GPS.
'GPSL1C' – count of carrier-phase interruptions counted for GPS signals (L1C etc.).
'CLKJMP' – count of receiver clock jumps (no clock jump in the above example).

The principles of cycle-slip detection algorithm implemented in Anubis are referred to [8]. The algorithm has been improved for supporting all available multi-GNSS frequencies and signals, and cycle slips are reported for individual constellations.

For receivers equipped with a low-quality oscillators, the clocks may diverge fast compared to a stable satellite atomic clocks. If the difference becomes too large, the receiver corrects the clocks by a particular number of milliseconds. This correction causes a jump in receiver clocks and, consequently, also in carrier-phase observations. A cycle slip common to all satellites can be then observed. Anubis analyses GNSS observations and provides estimated number of milliseconds at all epochs where a receiver generates such a clock jump. Actual clock jumps are reported with **sec_pre=2** as in example below:

```
#CLKJMP 2010-04-15 00:00:00   Phase[ms]
 CLKJMP 2010-12-26 00:07:30   1
 CLKJMP 2010-12-26 00:19:30   2
 CLKJMP 2010-12-26 00:31:00   3
 CLKJMP 2010-12-26 00:42:30   4
 CLKJMP 2010-12-26 00:54:30   5
 CLKJMP 2010-12-26 01:06:00   6
 CLKJMP 2010-12-26 01:17:30   7
 CLKJMP 2010-12-26 01:29:30   8
 CLKJMP 2010-12-26 01:41:00   9
 ..
```

The most comprehensive report of cycle slips provides lists of all identified cycle slips for individual satellites and constellations:

```
#GNSSLP 2016-08-24 00:00:00  PRN      L1C      L2C      L2I      L2L      L2P      L2W      L3Q      L5Q   ...
 GPSSLP 2016-08-24 00:39:00  G08    -12.0        -        -        -        -     -9.0        -     -9.0   ...
 GPSSLP 2016-08-24 00:39:30  G08     16.0        -        -        -        -     12.0        -     12.0   ...
 GPSSLP 2016-08-24 00:40:00  G30      4.0        -        -        -        -      3.0        -      3.0   ...
 ......
 GALSLP 2020-12-31 08:22:30  E18        -        -       12        -        -        -        -        -   ...
 GALSLP 2020-12-31 11:56:30  E15        -        -        -        -        -        -        -        -   ...
 ......
 GLOSLP 2020-12-31 00:30:00  R14       -9       -9        -       -7        -       -7        -        -   ...
 GLOSLP 2020-12-31 00:32:30  R18       -9       -9        -       -7        -       -7        -        -   ...
 GLOSLP 2020-12-31 00:42:00  R16       -9       -9        -       -7        -       -7        -        -   ...
 ......
 BDSSLP 2020-12-31 00:00:30  C05        -        -        -        -       -6        -        -        -   ...
 BDSSLP 2020-12-31 00:01:00  C05        -        -        -        -        1        -        -        -   ...
 BDSSLP 2020-12-31 00:03:00  C05        -        -        -        -        -        -        -        -   ...
```

'GPSSLP' – cycle slips [unit of cycle] for individual GPS observations, given epoch and satellite.

4.8 Elevation & azimuth

Satellite ephemeris (navigation data or precise orbits in SP3) are needed for qualitative and complex control, however, it can be useful also for plotting satellite skyplots and when calculating the ratio of existing/expected observations, see 4.1, within the quantitative control.

Satellite elevations and azimuths are calculated in user-defined sampling interval:

```
#====== Elevation & Azimuth
#GNSELE 2020-12-31 00:00:00  Mean x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
 GPSELE 2020-12-31 00:00:00    37   -   -   -   -   -   -   9  32   -  57   -   -   -   -  15  49   -  31   -  55  11   -  56   -   -  22  72   - ...
 GPSELE 2020-12-31 00:15:00    37   -   -   -   -   -   -   7  38   -  62   -   -   -   -  14  42   -  25   -  51  16   -  53   -   -  16  79   - ...
 GPSELE 2020-12-31 00:30:00    33   -   -   -   -   -   -   4  44   -  66   -   -   -   -  12  36   -  19   -  46  22   -  48   -   -  10  85   - ...
 GPSELE 2020-12-31 00:45:00    34   -   -   -   -   -   -   -  51   -  67   -   -   -   -  10  29   -  13   -  40  28   -  43   -   -   4  83   - ...
 GPSELE 2020-12-31 01:00:00    33   8   -   -   -   -   -   -  58   -  66   -   -   -   -   7  22   -   8   -  34  33   -  37   -   -   -  76   - ...
 GPSELE 2020-12-31 01:15:00    32  13   -   -   -   -   -   -  64   -  62   -   -   -   -   -  16   -   3   -  28  39   4  31   -   -   -  69   - ...
 GPSELE 2020-12-31 01:30:00    32  19   -   -   -   -   -   -  69   -  56   -   -   -   9   -   9   -   -   -  22  45  10  25   -   -   -  61   - ...
 ......
 BDSELE 2020-12-31 00:00:00    27   -   -   -   -  21  13   -   -   7   -  11  61  16   -   -  13   -   -  33   -  12  42   -  15  22   -   -   - ...
 BDSELE 2020-12-31 00:15:00    27   -   -   -   -  21  15   -   -  10   -  16  67  14   -   -  16   -   -  30   -  18  47   8  10  20   -   -   - ...
 BDSELE 2020-12-31 00:30:00    28   -   -   -   -  21  18   -   -  13   -  21  73  12   -   -  19   -   -  27   -  24  51  10   5  17   -   -   - ...
 BDSELE 2020-12-31 00:45:00    29   -   -   -   -  21  20   -   -  17   -  26  79  11   -   -  21   -   -  23   -  30  53  11   0  14   -   -   - ...
 BDSELE 2020-12-31 01:00:00    30   -   -   -   -  21  22   -   -  20   -  32  84   9   -   -  23   -   -  18   -  37  54  11   -  10   -   -   - ...
 BDSELE 2020-12-31 01:15:00    30   -   -   -   -  21  23   -   -  23   -  37  86   8   -   -  25   -   -  13   -  43  54  10   -   6   -   -   - ...
 BDSELE 2020-12-31 01:30:00    31   -   -   -   -  21  25   -   -  26   -  43  81   6   -   -  27   -   -   8   -  50  51   9   -   2   -   -   - ...
 ......
#GNSAZI 2020-12-31 00:00:00  Mean x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
 GPSAZI 2020-12-31 00:00:00   180   -   -   -   -   -   -   - 307 299   - 147   -   -   -   -  51 207   -  66   -  78 254   -  83   -   - 185 300   - ...
 GPSAZI 2020-12-31 00:15:00   176   -   -   -   -   -   -   - 302 300   - 137   -   -   -   -  45 203   -  69   -  69 258   -  72   -   - 184 295   - ...
 GPSAZI 2020-12-31 00:30:00   167   -   -   -   -   -   -   - 296 300   - 123   -   -   -   -  39 200   -  72   -  62 262   -  64   -   - 183 264   - ...
 GPSAZI 2020-12-31 00:45:00   145   -   -   -   -   -   -   - 300   - 105   -   -   -   -  34 197   -  75   -  58 266   -  58   -   - 181 177   - ...
 GPSAZI 2020-12-31 01:00:00   148 264   -   -   -   -   -   - 297   -  87   -   -   -   -  29 195   -  79   -  56 270   -  54   -   -   - 160   - ...
 GPSAZI 2020-12-31 01:15:00   163 268   -   -   -   -   -   - 291   -  73   -   -   -   - 194   -  82   -  55 274 220  52   -   -   - 157   - ...
 GPSAZI 2020-12-31 01:30:00   184 272   -   -   -   -   -   - 280   -  65   -   -   - 329   - 192   -   -   -  54 278 222  51   -   -   - 157   - ...
 ......
 BDSAZI 2020-12-31 00:00:00   149   -   -   -   - 127  70   -   -  97   - 284 296  60   -   -  73   -   -  57   - 175 128   - 233 290   -   -   - ...
 BDSAZI 2020-12-31 00:15:00   162   -   -   -   - 127  68   -   -  96   - 287 301  63   -   -  72   -   -  52   - 174 121 331 230 284   -   -   - ...
 BDSAZI 2020-12-31 00:30:00   162   -   -   -   - 127  67   -   -  95   - 290 308  66   -   -  70   -   -  47   - 173 112 326 227 278   -   -   - ...
 BDSAZI 2020-12-31 00:45:00   162   -   -   -   - 127  65   -   -  94   - 293 317  69   -   -  68   -   -  43   - 172 102 321 223 273   -   -   - ...
 BDSAZI 2020-12-31 01:00:00   167   -   -   -   - 127  63   -   -  93   - 296 340  72   -   -  66   -   -  40   - 171  91 316   - 268   -   -   - ...
 BDSAZI 2020-12-31 01:15:00   150   -   -   -   - 127  61   -   -  91   - 298  75  75   -   -  64   -   -  38   - 170  81 310   - 264   -   -   - ...
 BDSAZI 2020-12-31 01:30:00   156   -   -   -   - 127  59   -   -  90   - 300 114  78   -   -  62   -   -  37   - 168  71 305   - 260   -   -   - ...
 ......
```

'GPSELE' – elevation angles of individual GPS satellites.
'GPSAZI' – azimuth angles of individual GPS satellites.

4.9 Code multipath

GNSS tracking is affected by the so-called multipath effect due to the site environment and surrounding objects. In such a case, an original signal is not the only one received by the antenna, but also various reflected signals characteristics with extended paths. The multipath effect can be reduced by choke-ring or ecosorb material placed under the antenna. The multipath affects both code and carrier-phase observations, however, for code observations the effect is significantly higher.

Anubis calculates code multipath and the signal noise on the basis of the so-called multipath linear (MP) combinations for MP_1 and MP_2, [9]

$$MP_1 = P_1 - L_1 - \frac{2f_2^2}{(f_1^2 - f_2^2)}(L_1 - L_2) \tag{4.1}$$

$$MP_2 = P_2 - L_2 - \frac{2f_1^2}{(f_2^2 - f_1^2)}(L_2 - L_1) \tag{4.2}$$

where indices 1 and 2 represent the first and the second original (GPS) frequencies.

In principal, the MP linear combination is a difference of code and phase observations for removing a geometric term and, additionally, an ionospheric path delay eliminated by using carrier-phase observations from two frequencies. The L_1 and L_2 terms however still contain unknown ambiguities. The MP observation needs to be thus accumulated over several consecutive epochs (15-30; configurable with the attribute ***mpx_nep*** , see 3) for estimating a mean value that can be subtracted as unknown ambiguity. Anyway, a strategy for detecting cycle slips should be applied because any change in the ambiguity during the interval would disable its subtracting from the MP values. If phase observations are continuous (free of cycle slips), a root-mean-square error can be estimated to characterise a combined signal noise and code multipath effect (both cannot be further separated).

Anubis handles any multipath for any code observations if dual-frequency carrier phases are available for eliminating the ionospheric effect. Any code signal can be combined with a selected (unique) dual-frequency carrier phases using the generalised formula of the MP linear combination:

$$MP_k = P_k - L_i - \beta(L_i - L_j) = P_k + \alpha L_i + \beta L_j, \tag{4.3}$$

with

$$\alpha = -\frac{(f_j^2 + f_k^2)}{(f_i^2 - f_j^2)}\frac{f_i^2}{f_k^2} \qquad \beta = \frac{(f_i^2 + f_k^2)}{(f_i^2 - f_j^2)}\frac{f_j^2}{f_k^2}, \tag{4.4}$$

where k, i and j are indices for any GNSS frequency band.

Mean multipath values for all satellites of all constellations are reported:

```
#====== Code multipath
#GNSMxx 2020-12-31 00:00:00    mean x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
=GPSM1C 2020-12-31 00:00:00   25.96  23  25  23  24  31  29  26  29  20  28   -  24  22  27  27  27  26  25  27  32  22  23  27  23  23  28  26  31 ...
=GPSM2W 2020-12-31 00:00:00   22.76  16  26  16  21  35  27  21  23  18  24   -  17  17  23  24  27  25  23  28  30  15  20  24  15  15  22  19  31 ...
=GPSM2X 2020-12-31 00:00:00   25.87  23   -  17  35  32  29  27  28  21  25   -  21   -  25  30   -  28  24   -   -   -   -  25  26  20  27  29   - ...
=GPSM5X 2020-12-31 00:00:00   16.31  13   -  11  31   -  20   -  15  11  20   -   -   -  14   -   -   -  14   -   -   -   -  15  18   9  17  22   - ...
=GALM1X 2020-12-31 00:00:00   19.67  27  15  15  15  18   -  26  21  20   -  19  21  23  35  18   -   -  21  17   -  30   -   -  20  19  19  20   - ...
=GALM5X 2020-12-31 00:00:00   15.33  20   9  13  11  13   -  22  15  15   -  15  15  17  59  13   -   -  17  13   -  24   -   -  15  13  16  13   - ...
=GALM6X 2020-12-31 00:00:00   21.73  30  15  19  17  20   -  30  24  19   -  21  20  24  70  19   -   -  24  19   -  34   -   -  20  19  29  21   - ...
=GALM7X 2020-12-31 00:00:00   22.22  32  12  18  17  20   -  35  23  22   -  24  26  25  22  20   -   -  26  19   -  40   -   -  21  19  27  22   - ...
=GALM8X 2020-12-31 00:00:00    6.26   7   3   4   3   5   -   9   5   6   -   5   6   5  50   4   -   -   7   4   -  12   -   -   5   4   5   7   - ...
=GLOM1C 2020-12-31 00:00:00   34.97  36  30  34  32  31   -  33  36  32   -   -  32  46  31  28  40  31  34  46  46  30  51   -  31   -   -   -   - ...
=GLOM1P 2020-12-31 00:00:00   25.20  20  19  24  25  24   -  28  35  20   -   -  24  30  22  19  22  23  22  39  34  25  37   -  23   -   -   -   - ...
=GLOM2C 2020-12-31 00:00:00   51.04  61  48  49  47  52   -  49  54  46   -   -  44  61  46  43  68  41  52  58  47  45  78   -  46   -   -   -   - ...
=GLOM2P 2020-12-31 00:00:00   18.42  17  19  17  17  15   -  20  21  17   -   -  18  15  15  16  20  14  17  19  15  15  47   -  18   -   -   -   - ...
=BDSM2I 2020-12-31 00:00:00   46.80   -   -   -   - 276  36  42  33  35  34  29  28  35  22   -  38   -   -  33  24  26  28  28  30  29  30  29  26 ...
=BDSM6I 2020-12-31 00:00:00   41.79   -   -   -   - 355  31  35  28  23  25  19  11  22   9   -  37   -   -  18  13  14  19  14  20  17  20  15  14 ...
=BDSM7I 2020-12-31 00:00:00   91.64   -   -   -   - 378  48  46  37  34  34  26  22  35  17   -  51   -   -   -   -   -   -   -   -   -   -   -   - ...
```

'=GPSM1C' – average value of code multipath [cm] for all GPS satellites and signal on L1 frequency (C1C etc.).

'=GALM5X' – average value of code multipath [cm] for all Galileo satellites and signal on E5a frequency (C4X etc.).

Epoch-wise code multipath values are reported with **sec_mpx=2** :

```
GPSM1C 2020-12-31 00:00:00   26.26   -   -   -   -   -   -   - 38 30   - 20   -   -   -   - 37 16   - 22   - 18 32   - 20   -   - 39 17   - ...
GPSM1C 2020-12-31 00:15:00   25.67   -   -   -   -   -   -   - 39 23   - 13   -   -   -   - 34 24   - 27   - 18 30   - 18   -   - 38 18   - ...
GPSM1C 2020-12-31 00:30:00   26.66   -   -   -   -   -   -   - 21   - 15   -   -   -   - 39 37   - 22   - 21 21   - 24   -   - 44 13   - ...
GPSM1C 2020-12-31 00:45:00   26.37 35   -   -   -   -   -   - 16   - 16   -   -   -   - 43 21   - 37   - 19 27   - 19   -   - 35 15   - ...
GPSM1C 2020-12-31 01:00:00   31.02 54   -   -   -   -   -   - 14   - 16   -   -   -   -   - 31   - 45   - 28 20 65 25   -   -   - 13   - ...
GPSM1C 2020-12-31 01:15:00   25.89 28   -   -   -   -   -   - 14   - 17   -   -   -   -   - 44   -   -   - 26 22 33 32   -   -   - 19   - ...
GPSM1C 2020-12-31 01:30:00   25.62 23   -   -   -   -   -   - 13   - 15   -   -   - 35   - 36   -   -   - 45 17 27 34   -   -   - 12   - ...
......
GPSM2W 2020-12-31 00:00:00   18.89   -   -   -   -   -   -   - 27 15   - 7   -   -   -   - 33 10   - 42   - 11 34   - 9   -   - 14 7   - ...
GPSM2W 2020-12-31 00:15:00   19.00   -   -   -   -   -   -   - 23 7   - 4   -   -   -   - 56 9   - 42   - 6 19   - 11   -   - 23 8   - ...
GPSM2W 2020-12-31 00:30:00   20.80   -   -   -   -   -   -   - 8   - 3   -   -   -   - 44 23   - 36   - 12 26   - 11   -   - 31 6   - ...
GPSM2W 2020-12-31 00:45:00   24.94 44   -   -   -   -   -   - 7   - 5   -   -   -   - 41 23   - 38   - 14 17   - 16   -   - 58 6   - ...
GPSM2W 2020-12-31 01:00:00   26.80 22   -   -   -   -   -   - 6   - 5   -   -   -   -   - 32   - 43   - 22 18 87 25   -   -   - 6   - ...
GPSM2W 2020-12-31 01:15:00   23.22 30   -   -   -   -   -   - 5   - 8   -   -   -   -   - 27   -   -   - 29 13 57 24   -   -   - 12   - ...
GPSM2W 2020-12-31 01:30:00   21.47 29   -   -   -   -   -   - 7   - 7   -   -   - 23   - 29   -   -   - 39 8 25 41   -   -   - 6   - ...
......
GALM1X 2020-12-31 00:00:00   21.18 15   -   -   -   -   -   - 23 45   -   -   - 20 51 15   -   -   -   -   -   -   -   -   - 15   - 7   -   - ...
GALM1X 2020-12-31 00:15:00   19.21 25   -   -   -   -   -   - 20 43   -   -   - 21   - 11   -   -   -   -   -   -   -   -   - 19   - 10   -   - ...
GALM1X 2020-12-31 00:30:00   15.92 19   -   -   -   -   -   - 19 32   -   -   - 22   - 6   -   -   -   -   -   -   -   -   - 15   - 10   -   - ...
GALM1X 2020-12-31 00:45:00   20.10 49   -   -   -   -   -   - 26 29   -   -   - 16   - 11   -   -   -   -   -   -   -   -   - 21   - 11   -   - ...
GALM1X 2020-12-31 01:00:00   17.83 41   -   -   -   -   -   - 17   -   -   -   - 15   - 15   -   -   -   -   -   -   -   -   - 17   - 11   -   - ...
GALM1X 2020-12-31 01:15:00   13.26   -   -   -   -   -   -   - 12   -   -   -   - 16   - 12   -   -   -   -   -   -   -   -   - 13   - 15   -   - ...
GALM1X 2020-12-31 01:30:00   18.33   -   -   -   -   -   -   - 17   -   -   -   - 16   - 25   -   -   -   -   -   -   -   -   - 14   - 23   -   - ...
......
```

'GPSM1C' – values of code multipath [cm] for all GPS satellites and signal on L1 frequency (C1C etc.).

'GALM5X' – values of code multipath [cm] for all Galileo satellites and signal on E5a frequency (C4X etc.).

4.10 Signal to noise ratio

The signal strength is reported as a summary for each satellite and signal over the data period, and, optionally (**sec_snr=2**), using original values with user-defined sampling interval:

```
#====== Signal to noise ratio
#GNSSxx 2020-12-31 00:00:00    mean x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
=GPSS1C 2020-12-31 00:00:00   43.54  45  43  45  43  43  42  43  42  46  44   -  46  44  42  43  42  43  42  42  41  44  46  42  45  45  42  43  42 ...
=GPSS2W 2020-12-31 00:00:00   36.96  44  29  45  33  38  38  39  39  46  38   -  46  34  29  39  27  39  30  29  27  34  35  29  45  44  38  40  27 ...
=GPSS2X 2020-12-31 00:00:00   43.65  46   -  46  45  41  42  41  43  46  43   -  45   -  44  41   -  41  44   -   -   -  43  47  45  43  45   -   - ...
=GPSS5X 2020-12-31 00:00:00   46.50  49   -  48  47   -  45   -  46  48  45   -   -   -  45   -   -  46   -   -   -  45  48  48  45  47   -   - ...
=GALS1X 2020-12-31 00:00:00   44.73  43  47  46  47  46   -  42  43  46   -  43  43  41  43  45   -  46  43   -  42   -   -  46  46  45  45   -   - ...
=GALS5X 2020-12-31 00:00:00   43.86  42  48  45  47  45   -  41  42  45   -  41  42  40  45  45   -  44  42   -  41   -  45  45  44  44   -   - ...
=GALS6X 2020-12-31 00:00:00   40.61  39  44  41  45  42   -  38  38  42   -  38  39  36  41  41   -  41  39   -  37   -   -  42  42  41  40   -   - ...
=GALS7X 2020-12-31 00:00:00   42.77  41  46  44  46  44   -  40  40  44   -  40  42  39  45  43   -  43  41   -  40   -   -  43  44  43  43   -   - ...
=GALS8X 2020-12-31 00:00:00   46.84  45  50  48  50  48   -  44  45  48   -  43  45  43  47  48   -   -  47  45   -  44   -   -  48  48  47  47   -   - ...
=GLOS1C 2020-12-31 00:00:00   45.78  45  48  48  48  49  43  47  46  47  46  48  49  39  48  49  43  49  46  38  38  48  42  47  49   -   -   -   - ...
=GLOS1P 2020-12-31 00:00:00   44.42  44  47  46  46  47  42  46  44  46  45  47  47  37  47  47  41  47  45  37  36  47  42  46  47   -   -   -   - ...
=GLOS2C 2020-12-31 00:00:00   41.40  39  42  42  41  43   -  42  42  44   -  43  43  38  43  43  38  44  42  40  43  42  34   -  43   -   -   -   - ...
=GLOS2P 2020-12-31 00:00:00   40.58  37  42  41  40  42   -  41  41  43   -  42  43  40  42  42  37  43  41  39  43  42  30   -  42   -   -   -   - ...
=BDSS2I 2020-12-31 00:00:00   42.17   -   -   -   - 35  36  36  37  38  38  42  42  40  44   -  38   -   -  42  45  45  43  44  43  44  43  44  44 ...
=BDSS6I 2020-12-31 00:00:00   39.66   -   -   -   - 34  34  34  35  36  35  41  43  36  44   -  32   -   -  39  41  43  41  42  41  42  40  40  40 ...
=BDSS7I 2020-12-31 00:00:00   38.56   -   -   -   - 37  35  36  37  38  37  42  44  37  45   -  35   -   -   -   -   -   -   -   -   -   -   -   - ...

 GPSS1C 2020-12-31 00:00:00   44.14   -   -   -   -   -   - 39  43   - 52   -   -   -   - 40  47   - 43   - 48  35   - 48   -   - 40 50   - ...
 GPSS1C 2020-12-31 00:15:00   43.51   -   -   -   -   -   - 34  46   - 52   -   -   -   - 38  46   - 39   - 47  40   - 48   -   - 38 51   - ...
 GPSS1C 2020-12-31 00:30:00   41.80   -   -   -   -   -   - 30  46   - 54   -   -   -   - 37  46   - 38   - 47  39   - 48   -   - 34 51   - ...
 GPSS1C 2020-12-31 00:45:00   43.32   -   -   -   -   -   - 49   - 52   -   -   -   - 38  43   - 38   - 46  43   - 47   -   - 33 52   - ...
 GPSS1C 2020-12-31 01:00:00   43.18  36   -   -   -   -   - 50   - 53   -   -   -   - 34  41   - 36   - 44  45   - 45   -   -   - 52   - ...
 GPSS1C 2020-12-31 01:15:00   42.45  38   -   -   -   -   - 52   - 52   -   -   -   -   - 39   - 32   - 42  45  34  42   -   -   - 52   - ...
 GPSS1C 2020-12-31 01:30:00   43.12  41   -   -   -   -   - 50   - 52   -   -   - 36   - 38   -   -   - 38  47  38  41   -   -   - 51   - ...
 ......
```

4.11 Satellite information

Satellite information consists of navigation data availability and satellite health.

```
#====== Satellite information
#GNSIxx 2020-12-31 00:00:00            x01 x02 x03 x04 x05 x06 x07 x08 x09 x10 x11 x12 x13 x14 x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 ...
 GPSHLT 2020-12-31 00:00:00             1   1   1   1   1   1   1   1   1   1   0   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1 ...
 GPSHLT 2020-12-31 01:59:44             1   1   1   1   1   1   1   1   1   1   0   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1 ...
 GPSHLT 2020-12-31 02:00:00             1   1   1   1   1   1   1   1   1   1   0   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1 ...
 GPSHLT 2020-12-31 03:59:28             1   1   1   1   1   1   1   1   1   1   0   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1 ...
 GPSHLT 2020-12-31 03:59:44             1   1   1   1   1   1   1   1   1   1   0   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1 ...
 ......
```

```
GALHLT 2020-12-31 00:00:00     1  1  1  1  1  -  1  1  1  -  1  1  1  1  1  -  -  1  1  -  1  -  1  -  -  1  1  1  1  - ...
GALHLT 2020-12-31 00:10:00     1  1  1  1  1  -  1  1  1  -  1  1  1  1  1  -  1  -  1  1  -  1  -  1  -  -  1  1  1  1  - ...
GALHLT 2020-12-31 00:20:00     1  1  1  1  1  -  1  1  1  -  1  1  1  1  1  -  -  1  1  -  1  -  -  1  1  1  1  - ...
GALHLT 2020-12-31 00:30:00     1  1  1  1  1  -  1  1  1  -  1  1  1  1  1  -  -  1  1  -  1  -  -  1  1  1  1  - ...
GALHLT 2020-12-31 00:40:00     1  1  1  1  1  -  1  1  1  -  1  1  1  1  1  -  -  1  1  -  1  -  -  1  1  1  1  - ...
......
GLOHLT 2020-12-31 00:15:00     1  1  1  1  1  1  1  1  1  1  -  1  1  1  1  1  1  1  1  1  1  1  1  1  1  -  -  -  -  - ...
GLOHLT 2020-12-31 00:45:00     1  1  1  1  1  1  1  1  1  1  -  1  1  1  1  1  1  1  1  1  1  1  1  1  1  -  -  -  -  - ...
GLOHLT 2020-12-31 01:15:00     1  1  1  1  1  1  1  1  1  1  -  1  1  1  1  1  1  1  1  1  1  1  1  1  1  -  -  -  -  - ...
GLOHLT 2020-12-31 01:45:00     1  1  1  1  1  1  1  1  1  1  -  1  1  1  1  1  1  1  1  1  1  1  1  1  1  -  -  -  -  - ...
GLOHLT 2020-12-31 02:15:00     1  1  1  1  1  1  1  1  1  1  -  1  1  1  1  1  1  1  1  1  1  1  1  1  1  -  -  -  -  - ...
......
BDSHLT 2020-12-31 00:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
BDSHLT 2020-12-31 01:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
BDSHLT 2020-12-31 02:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
BDSHLT 2020-12-31 03:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
BDSHLT 2020-12-31 04:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
......
GPSRED 2020-12-31 00:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
GPSRED 2020-12-31 01:59:44     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
GPSRED 2020-12-31 02:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
GPSRED 2020-12-31 03:59:28     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
GPSRED 2020-12-31 03:59:44     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
GPSRED 2020-12-31 04:00:00     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 ...
......
GALRED 2020-12-31 00:00:00     2  2  2  2  2  0  2  2  2  0  2  2  2  2  2  0  0  2  2  0  2  0  0  2  2  2  2  0 ...
GALRED 2020-12-31 00:10:00     2  2  2  2  2  0  2  2  2  0  2  2  2  2  2  0  0  2  2  0  2  0  0  2  2  2  2  0 ...
GALRED 2020-12-31 00:20:00     2  2  2  2  2  0  2  2  2  0  2  2  2  2  2  0  0  2  2  0  2  0  0  2  2  2  2  0 ...
GALRED 2020-12-31 00:30:00     2  2  2  2  2  0  2  2  2  0  2  2  2  2  2  0  0  2  2  0  2  0  0  2  2  2  2  0 ...
GALRED 2020-12-31 00:40:00     2  2  2  2  2  0  2  2  2  0  2  2  2  2  2  0  0  2  2  0  2  0  0  2  2  2  2  0 ...
GALRED 2020-12-31 00:50:00     2  2  2  2  2  0  2  2  2  0  2  2  2  2  2  0  0  2  2  0  2  0  0  2  2  2  2  0 ...
......
```

'GPSHLT' – list of navigation records with corresponding signal health status for each satellite.

'GPSRED' – list of navigation records with number of duplicity records for each satellite.

The '-' characterises unavailable navigation data for the given satellite. The satellite health status is represented as binary code (*unhealth* = 0, *health* = 1). Duplicity records are counted with the same IOD number. In case of Galileo satellites in the example, it is represented with FNAV and INAV records of navigation types.

4.12 Key-parameter indicators

Along with a single point positioning, a key-parameter indicators (KPIs) can be estimated from horizontal and vertical positioning errors when the reference coordinates are provided by a user (via setting or an input SINEX file). Alternatively, the errors might represent residuals if reference is given by mean position, see Section 4.3. Criteria for horizontal (**max_hpe**), vertical (**max_vpe**), and PDOP (**max_dop**) values can be specified by user too, see 3.

An example of 30-sec data analysis using Galileo FNAV navigation data and E1+E5a signals:

```
#====== Key-parameter indicators
#GNSKPI 2020-11-16 00:00:00   CountOK  CountXX  %_Expect  %_Exist  %_Posit  %_DopOK
 GAL_EP 2020-11-16 00:00:00     2874        6   99.792    99.792      -        -
 GAL_DH 2020-11-16 00:00:00     2858        0   99.236    99.236   99.443  100.000     0.560
 GAL_DV 2020-11-16 00:00:00     2858        0   99.236    99.236   99.443  100.000     0.729
 GAL_PD 2020-11-16 00:00:00     2858       16   99.236    99.236   99.443  100.000
 GAL_DF 2020-11-16 00:00:00     2880        0  100.000   100.000      -        -
```

'GAL_EP' – number of epochs usable for KPI calculation, i.e. supported with valid navigation data.

'GAL_DH' – number of epochs with horizontal positioning error smaller than criterion (e.g. 10 m).

'GAL_DV' – number of epochs with vertical positioning error smaller than criterion (e.g. 10 m).

'GAL_PD' – number of epochs with PDOP satisfying user-defined criterion (e.g. *PDOP* < 6).

'GAL_DF' – number of epochs with available dual-frequency data.

Numbers of 'fine' epochs (*'CountOK'*) and 'outlier' epochs (*'CountXX'*) in different lines reflect various bases: 1) *'GAL_EP'* – epochs usable for KPI calculation equals to a count of positioning results *'POSGAL'* in Section 4.3, 2) *'GAL_DH'* , *'GAL_DV'* , and *'GAL_DP'* – epochs of horizontal, vertical, PDOP results satisfying the criteria, and 3) *'GAL_DF'* – epochs with dual-frequency observations. From the above example, the total number of expected

epochs is 2880 (all providing dual-frequency data), 2854/26 epochs are with/without valid navigation data (usable for positioning) and, 2838/16 epochs are with/without positioning results satisfying the criteria (16 epochs doesn't satisfy $PDOP < 6$).

The percentages are represented with respect to different bases (counts of epochs): expected (*'%_Expect'*), existing (*'%_Exists'*), positioning (*'%_Posit'*), and those with satisfying PDOP criteria (*'%_DopOK'*). The last value (*'GAL_DH'* , *'GAL_DV'* lines only) is 1-sigma error of horizontal and vertical positioning, respectively.

```
=XYZREF 2020-11-16 11:59:45    3979319.0580    1050312.4160    4857064.7259    0.0057    0.0047    0.0058
=ECCREF 2020-11-16 11:59:45          0.0107          0.0407          0.0500

#GALKPI 2020-11-16 00:00:00    dN[m]    dE[m]    dU[m]    PDOP    FLAGS
 GALKPI 2020-11-16 04:03:30    0.198    0.347    0.021    3.63     ___
 GALKPI 2020-11-16 04:04:00    0.074    0.068    1.551    3.62     ___
 GALKPI 2020-11-16 04:04:30   -0.153   -0.002    1.947   18.87     D__
 GALKPI 2020-11-16 04:05:00    0.450    0.097    0.388   19.26     D__
 GALKPI 2020-11-16 04:05:30   -0.578    0.037    1.304    3.59     ___
 GALKPI 2020-11-16 04:06:00   -0.285    0.303    0.639    3.58     ___
 GALKPI 2020-11-16 04:06:30   -0.546    0.110   -0.341   19.84     D__
 GALKPI 2020-11-16 04:07:00   -0.114    0.036    0.135   19.79     D__
 GALKPI 2020-11-16 04:07:30   -0.648    0.327    0.387   19.62     D__
 GALKPI 2020-11-16 04:08:00   -0.275    0.165   -0.378   19.33     D__
 GALKPI 2020-11-16 04:08:30   -0.158    0.241   -2.322   18.95     D__
 GALKPI 2020-11-16 04:09:00    0.221    0.479   -5.694   18.49     D__
 GALKPI 2020-11-16 04:09:30    0.669    0.234   -6.403   17.97     D__
 GALKPI 2020-11-16 04:10:00   -0.439    0.177   -4.531   17.40     D__
 GALKPI 2020-11-16 04:10:30   -0.094    0.477   -3.984   16.81     D__
 GALKPI 2020-11-16 04:11:00    0.118    0.256   -1.159   16.21     D__
 GALKPI 2020-11-16 04:11:30    0.219   -0.156    1.066   15.61     D__
 GALKPI 2020-11-16 04:12:00    0.501    0.268   -3.015   15.02     D__
 GALKPI 2020-11-16 04:12:30    0.046   -0.021    1.378   14.44     D__
 GALKPI 2020-11-16 04:13:00    0.330   -0.331   -1.940   13.89     D__
 GALKPI 2020-11-16 04:13:30   -0.031   -0.178    3.138    4.22     ___
 GALKPI 2020-11-16 04:14:00   -1.064   -0.016   -1.299    4.24     ___
 ......
```

'XYZREF' – reference cartesian coordinates of the site.

'ECCREF' – reference eccentricities of the site.

'GALKPI' – epoch-by-epoch North, East and Up differences to reference coordinates and PDOP value.

The North (dN), East (dE) and Up (dU) differences from a given epoch with respect to reference coordinates are confronted with user-defined criteria and the flag is setup for any exceeding values (D=DOP, H=Horizontal, V=Vertical).

5. XML QC output

The eXtensible Markup Language (XML) is a human-readable and machine-readable format.

The Anubis' XML output has been developed within the European Plate Observing System (EPOS) implementing phase. It has been designed for a standard summary of the most important QC parameters stemming from a single file, and for archiving or filling database in support of a continuous long-term monitoring of GNSS site metadata, data, instrumentation and environment.

A minimum of original and non-redundant QC quantitative and qualitative parameters is stored in the XML output for each RINEX file, i.e. system- and signal-specific data, while no epoch-/satellite-specific data. The XML content enables a flexible time-series visualisation of selected parameters (or user-defined QC indicators derived from the original parameters) for easy detection of any significant change of the site evolution.

Figure 5.1 shows a graph of the XML structure which contains four main elements:

- META – metadata on the XML output file generation.
- NAVI – navigation data availability (indicators of qualitative control).
- DATA – system dependent/independent quantitative/qualitative QC parameters.
- HEAD – metadata from RINEX file header.
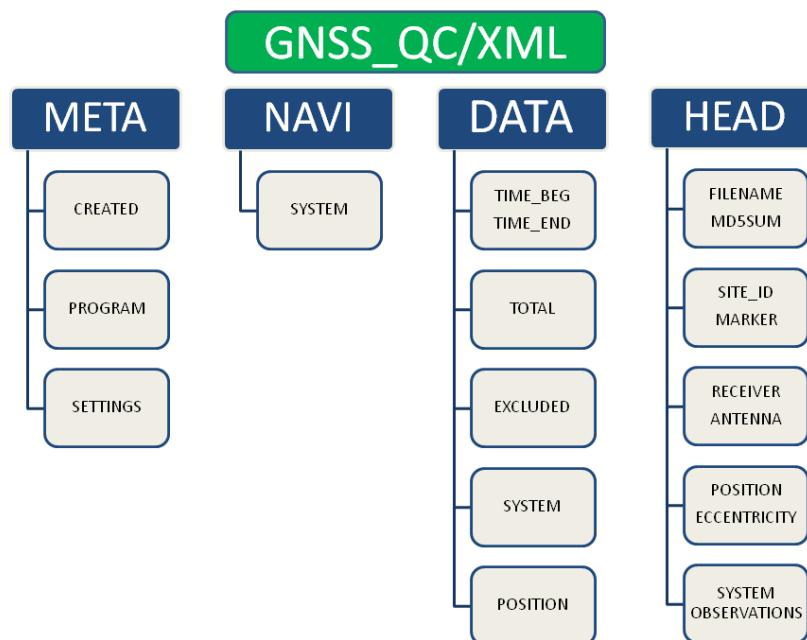
Figure 5.1: GNSS_QC XML output

The XML starts with the file pre-ambule:

```
<?xml version="1.0"?>
<QC_GNSS gml:id="GOPE00CZE_20201231T000000_20201231T235930"       # XML file identifier
        xmlns="https://software.pecny.cz/anubis"                  # XML namespace declared
        xmlns:gml="http://www.opengis.net/gml/3.2"                # GML namespace introduced
        xmlns:xsi="https://www.w3.org/2001/XMLSchema/instance"    # XSI namespace introduced
        schemaVersion="1.1"
...
</QC_GNSS>
```

The XML META element includes the software, its setting and the file creation time:

```
  <meta>
    <created>2021-01-18T10:37:54</created>
    <program>G-Nut/Anubis Pro [3.1]</program>
    <settings>
      <time_beg>2020-12-31T00:00:00</time_beg>
      <time_end>2020-12-31T23:59:30</time_end>
      <data_int>30.000</data_int>
      <elev_min>10.00</elev_min>
      <system type="GPS" />
      <system type="GAL" />
      <system type="GLO" />
      <system type="BDS" />
    </settings>
  </meta>
  ...
</QC_GNSS>
```

The XML NAVI element reports counts of satellites (*'nsat'*) and records (*'have'*) available in navigation data for individual GNSS (*'type'*) which are critical for reliable qualitative control:

```
  <navi>
    <system type="GPS" nsat="32">              # system-specific element
      <have>449</have>                          # count of available navigation records
    </system>
    <system type="GAL" nsat="24">
      <have>5006</have>
    </system>
    ...
  </navi>
```

The XML DATA element starts with estimated generic parameters (constellation-independent):

```
  <data>
    <time_beg>2020-12-31T00:00:00</time_beg>    # data first epoch
    <time_end>2020-12-31T23:59:30</time_end>    # data last epoch
    <data_int>30.00</data_int>                  # data sampling interval
    <data_lag>12.62488</data_lag>               # data file latency
    <numb_epo>2880</numb_epo>                   # number of epochs
    <numb_gap>0</numb_gap>                      # number of data gaps
    <total>
      <expt>148580</expt>                       # sum of expected observations (selected signal and all GNSS)
      <have>120525</have>                       # sum of existing observations (selected signal and all GNSS)
      <expt_usr>114833</expt_usr>               # sum of existing observations above user mask
      <have_usr>104528</have_usr>               # sum of existing observations above user mask
      <elev_min>0.02</elev_min>                 # a minimum observed elevation angle
      <cyc_slps>748</cyc_slps>                  # sum of cycle slips
      <clk_jmps>0</clk_jmps>                    # sum of clock jumps
    </total>
    <excluded>
      <xbeg>0</xbeg>                            # counts of epochs filtered out (prior   user-defined observation interval)
      <xend>0</xend>                            # counts of epochs filtered out (after   user-defined observation interval)
      <xint>0</xint>                            # counts of epochs filtered out (higher  user-defined observation sampling)
      <xsys>0</xsys>                            # counts of epochs filtered out (outside user-defined GNSS constellations)
    </excluded>
```

The XML DATA element contains quantitative and qualitative QC parameters:

```
  <data>
    ...
    <system type="GPS" nsat="31" xele="0">      # system-specific element
      <epo>
        <expt>2880</expt>                       # count of expected data epochs
        <have>2880</have>                       # count of existing data epochs
        <dual>2880</dual>                       # count of dual-frequency data epochs
      </epo>
      <amb>                                      # carrier-phase interruptions
        <nepo>0</nepo>                          # count of interruptions due to gap in epoch
        <nsat>171</nsat>                        # count of interruptions due to gap in satellite
        <nsig>271</nsig>                        # count of interruptions due to gap in signal
        <nslp>29</nslp>                         # count of interruptions due to cycle slips
      </amb>
      <bnd type="code">                          # bands for code observation type
        <xepo>0</xepo>                          # code: count of epochs with single-frequency data
        <xsat>360</xsat>                        # code: count of satellites with single-frequency data
      </bnd>
      <bnd type="phase">                         # bands for phase observation type
        <xepo>0</xepo>                          # phase: count of epochs with single-frequency data
        <xsat>400</xsat>                        # phase: count of satellites with single-frequency data
      </bnd>
      <obs type="C1C">                           # code observation element
        <nsat>31</nsat>                         # C1C: count of observed satellites
        <expt>33515</expt>                      # C1C: count of expected observations
        <have>29709</have>                      # C1C: count of existing observations
        <expt_usr>25388</expt_usr>              # C1C: count of expected observations above user mask
        <have_usr>25365</have_usr>              # C1C: count of existing observations above user mask
```

```
      <mpth>26.0</mpth>                                       # C1C: code multipath
    </obs>
    ...
    <obs type="L2W">                                          # phase observation element
      <nsat>31</nsat>                                         # L2W: count of observed satellites
      <expt>33515</expt>                                      # L2W: count of expected observations
      <have>28971</have>                                      # L2W: count of existing observations
      <expt_usr>25388</expt_usr>                              # L2W: count of expected observations above user mask
      <have_usr>25363</have_usr>                              # L2W: count of existing observations above user mask
      <slps>12</slps>                                         # L2W: count of detected cycle slips
    </obs>
    ...
    <obs type="S5X">                                          # signal strength observation element
      <nsat>16</nsat>                                         # S5X: count of observed satellites
      <expt>33515</expt>                                      # S5X: count of expected observation
      <have>14952</have>                                      # S5X: count of existing observation
      <expt_usr>25388</expt_usr>                              # S5X: count of expected observation above user mask
      <have_usr>12728</have_usr>                              # S5X: count of existing observation above user mask
    </obs>
  </system>
  ...
</data>
```

'system type="GPS"' – GPS-specific output section.

'system nsat="31"' – number of observed satellites for the SYSTEM.

'system xele="0"' – number of observations without available observation azimuth/elevation information.

The XML DATA element contains positioning results:

```
<data>
  ...
  <position type="GPS" source="analysis">                    # GPS single point positioning
    <coordinate>
      <gml:Point gml:id="GPS_analysis">
        <gml:pos>3979316.822 1050312.764 4857068.334</gml:pos>   # X,Y,Z cartesian coordinates [m]
      </gml:Point>
    </coordinate>
    <deviation>
      <axis name="X">1.427</axis>                             # X,Y,Z standard deviations [m]
      <axis name="Y">0.876</axis>
      <axis name="Z">1.501</axis>
    </deviation>
  </position>
  <position type="GAL" source="analysis"> ... </position>     # GAL single point positioning
  <position type="GLO" source="analysis"> ... </position>     # GLO single point positioning
  <position type="BDS" source="analysis"> ... </position>     # BDS single point positioning
</data>
```

The XML HEAD element contains metadata of a single input file (*file_numb=1*):

```
<head file_numb="1">
  <file_name>GOPE00CZE_R_20203660000_01D_30S_MO.crx.gz</file_name>   # original file name
  <file_md5sum>40622a882a555ce2e6ccd1403632c985</file_md5sum>        # original file md5sum (uncompressed Z/gz)
  <file_format>RINEX3.03</file_format>                                # original file format
  <site_id>GOPE00CZE</site_id>                                        # site name (identified by Anubis)
  <marker_name>GOPE</marker_name>                                     # site marker (RINEX header field)
  <marker_numb>11502M002</marker_numb>                                # site marker number (RINEX field)
  <receiver_type>TRIMBLE ALLOY</receiver_type>                        # receiver type
  <receiver_numb>40016</receiver_numb>                                # receiver number
  <receiver_vers>5.45</receiver_vers>                                 # receiver firmware
  <antenna_type>TPSCR.G3</antenna_type>                               # antenna type
  <antenna_dome>TPSH</antenna_dome>                                   # antenna radome
  <antenna_numb>30161</antenna_numb>                                  # antenna number
  <software>ALLOY 5.45&amp;OBSRINEX</software>                        # software used in RINEX creation
  <data_int>30.000</data_int>                                         # data sampling (RINEX header)
  <position type="GNS" source="rinex_header">
    <coordinate>
      <gml:Point gml:id="rinex_header">
        <gml:pos>3979316.439 1050312.253 4857066.904</gml:pos>        # X,Y,Z cartesian coordinates (RINEX header)
      </gml:Point>
    </coordinate>
    <eccentricity>
      <axis name="N">0.0000</axis>                                    # N,E,U marker eccentricity
      <axis name="E">0.0000</axis>
      <axis name="U">0.1114</axis>
    </eccentricity>
  </position>
  <system type="BDS">                                                 # BeiDou observation types (RINEX header)
    <obs type="C5X" />
    <obs type="L5X" />
    <obs type="S5X" />
    <obs type="C2I" />
    <obs type="L2I" />
    ...
  </system>
  <system type="GAL"> ... </system>                                   # Galileo observation types (RINEX header)
  <system type="GLO"> ... </system>                                   # GLONASS observation types (RINEX header)
  <system type="GPS"> ... </system>                                   # GPS     observation types (RINEX header)
  <system type="IRN"> ... </system>                                   # IRNSS   observation types (RINEX header)
  <system type="SBS"> ... </system>                                   # SBAS    observation types (RINEX header)
  <system type="QZS"> ... </system>                                   # QZSS    observation types (RINEX header)
</head>
```

6. JSON QC output

JSON (JavaScript Object Notation, *https://www.json.org*) is a lightweight data exchange format. The Anubis' JSON output has been developed for optimal high-resolution visualisation of a single QC data file output such as skyplots, positioning charts, parameter high-resolution time series, and for the online QC and visualization service. The JSON structure contains two main elements:

- META – metadata about the XML file (XML creation, software usage and setting),
- SUMM – summary of positioning and observations statistics (Anubis 3.2).
- DATA – parts containing time-, system-, satellite-, and signal-dependent output data,

Metadata part of the Anubis' JSON 1.1 contains:

```
{ "META":
  { "VERSION":"1.1",                        # JSON version
    "CREATED":"2022-01-07T15:11:13",        # JSON created
    "RNXVER":"3.02",                        # RINEX version
    "MARKER":"GOP7",                        # site marker
    "NUMBER":"11502M006",                   # site number
    "ANTTYPE":"SEPCHOKE_B3E6",              # antenna type
    "ANTDOME":"SPKE",                       # antenna radome
    "ANTNUMB":"5287",                       # antenna number
    "RECTYPE":"TRIMBLE␣␣␣␣NETR9",           # receiver type
    "RECNUMB":"49084",                      # receiver number
    "RECVERS":"5.37"                        # receiver firmware
  },
  "SUMM": { "POS": { ... },                 # since JSON 1.1
           "OBS": { ... }                   # since JSON 1.1
  }
  "DATA": { "POS": { ... },                 # since JSON 1.0
           "SAT": { ... }                   # since JSON 1.0
  }
}
```

For easy use in various frameworks, the Anubis' JSON data part can be converted into individual hash structures (system *sys*, epoch *epo*, satellite number *sat*, and signal *sig*,):

- SUMM ["POS"] [*sys*] + [keys: "XYZ"|"BLH"]
- SUMM ["OBS"] [*sys*] [*sig*] + [keys: "NSAT"|"HAVZ"|"EXPZ"|"RATZ"|"HAVU"|"EXPU"|"RATU"|"WELE"|"ELE'x'"]
- DATA ["POS"] [*sys*] [*epo*] + [keys: "XYZ"|"SAT"|"DOP"|"NEU"]
- DATA ["SAT"] [*sys*] [*epo*] [*sat*] + [keys: "AZI"|"ELE"|"CBN"|"LBN"|"Sxx"|"Mxx"]

Summary of positioning and observations statistics includes:

```
{ "SUMM":
  { "POS":
    { "GPS": { "XYZ":[3979318.8913,1050311.8235,4857064.1111,4.46,2.78,5.45],
               "BLH":[49.913670125,14.785603734,33914.619415661,3.07,4.18,6.38] },
      "GAL": { "XYZ":[3979318.9213,1050312.4991,4857064.7901,0.75,0.47,0.82],
               "BLH":[49.913672671,14.785612721,33951.8153328,0.57,0.72,0.95] },
      ....

      "GNS": { "XYZ":[3979318.8515,1050311.9475,4857064.0402,2.76,1.98,3.45],
               "BLH":[49.913669762,14.785605545,33911.258156466,1.51,3.07,4.15] }
    },

    "OBS":
    { "GPS":
      { "C1C": { "NSAT":31,"HAVZ":29383,"EXPZ":33322,"RATZ":88.18,"HAVU":25117,"EXPU":25137,"RATU":99.92,"WELE":0,
                 "ELE0":29383,"ELE5":28175,"ELE10":25117,"ELE15":22034,"ELE20":19297,"ELE30":14737,"ELE50":7777,"ELE70":2701 },
        "C2W": { "NSAT":31,"HAVZ":28608,"EXPZ":33322,"RATZ":85.85,"HAVU":25116,"EXPU":25137,"RATU":99.92,"WELE":0,
                 "ELE0":28608,"ELE5":27811,"ELE10":25116,"ELE15":22033,"ELE20":19297,"ELE30":14737,"ELE50":7777,"ELE70":2701 },
        "C2X": { "NSAT":22,"HAVZ":20695,"EXPZ":33322,"RATZ":62.11,"HAVU":17627,"EXPU":25137,"RATU":70.12,"WELE":0,
                 "ELE0":20695,"ELE5":19854,"ELE10":17627,"ELE15":15375,"ELE20":13666,"ELE30":10445,"ELE50":5705,"ELE70":2044 },
        ....
      },
      "GLO":
      { "C1C": { "NSAT":24,"HAVZ":24018,"EXPZ":27164,"RATZ":88.42,"HAVU":21089,"EXPU":21381,"RATU":98.63,"WELE":0,
                 "ELE0":24018,"ELE5":23329,"ELE10":21089,"ELE15":18301,"ELE20":16109,"ELE30":12137,"ELE50":6256,"ELE70":1702 },
        "C1P": { "NSAT":24,"HAVZ":23946,"EXPZ":27164,"RATZ":88.15,"HAVU":21056,"EXPU":21381,"RATU":98.48,"WELE":0,
                 "ELE0":23946,"ELE5":23272,"ELE10":21056,"ELE15":18279,"ELE20":16106,"ELE30":12137,"ELE50":6256,"ELE70":1702 },
        ....
      }
    }
  }
}
```

'"NSAT"' – number of satellites per signal.

'"HAVZ"' – number of existing observations for signal at the horizon.

'"EXPZ"' – number of expected observations for signal at the horizon.

'"RATZ"' – ratio of existing/expected observations for signal at the horizon.

'"HAVU"' – number of existing observations for signal at the user elevation angle cut-off.

'"EXPU"' – number of expected observations for signal at the user elevation angle cut-off.

'"RATU"' – ratio of existing/expected observations for signal at the user elevation angle cut-off.

'"WELE"' – number of observations without azimuth/elevation information for signal/satellite.

'"ELEx"' – number of observations above the 'x' elevation angle cut-off x=0,5,10,15,20,30,50,70 [deg].

Data part with positioning results contains:

```
{ "DATA":
  { "POS":
    { "GPS":
      { "2020-10-26T00:00:00":
        { "XYZ": [3979317.8786,1050307.6319,4857056.5658,-8.3234],
          "SAT": [9,1],
          "DOP": [1.8,1.6,0.9,1.3],
          "NEU": [-3.2912,-3.7944,-7.0921]
        },
        "2020-10-26T00:15:00":
        { "XYZ": [3979317.2812,1050306.9327,4857058.1024,-8.1434],
          "SAT": [8,0],
          "DOP": [2.0,1.8,0.8,1.5],
          "NEU": [-1.7233,-4.318,-6.4033]
        },
      },
      "GLO":
      { "2020-10-26T00:00:00":
        { "XYZ": [3979317.7124,1050307.3301,4857064.6485,9.0606],
          "SAT": [7,1],
          "DOP": [2.5,2.2,0.6,1.9],
          "NEU": [1.6378,-4.2608,0.363]
        } ...
      } ...
    } ...
    "SAT": { ... }
  }
}
```

'"XYZ"' – array(4) of cartesian coordinates X,Y,Z [m] and receiver clocks [m].

'"SAT"' – array(1) of number of available and excluded satellites in solution.

'"DOP"' – array(4) of dilution of precision values (GDOP, PDOP, HDOP, VDOP).

'"NEU"' – array(3) of N,E,U residuals [m] with respect to reference coordinates.

Part with satellite-/observation-specific data contains:

```
{ "DATA" :
  { "SAT":
    { "GPS":
      { "2020-10-26T00:00:00":
        { "02":
          { "AZI":51.807,
            "ELE":32.372,
            "CBN":2,
            "LBN":2,
            "S1C":45.4,
            "S2W":32.8,
            "M1C":31.2,
            "M2W":25.72
          },
          "05":
          { "AZI":93.013,
            "ELE":10.049,
            "CBN":2,
            "LBN":2,
            "S1C":36.8,
            "S2W":28.4,
            "S2X":34.0,
            "M1C":52.04,
            "M2W":38.04,
            "M2X":34.71
          } ...
        } ...
      } ...
    } ...
  } ...
}
```

'"AZI"' , '"ELE"' – satellite azimuth, elevation angle [deg].

'"CBN"' , '"LBN"' – number of code and phase frequency bands.

'"Mxx"' , '"Sxx"' – code multipath and noise [m] and signal-to-noise density [DbHz] where "xx" represents individual signals.

7. Real-time and user console

The software may also operate in a real-time mode with when supporting all relevant functionalities from the offline mode, i.e. considering any operation modes and corresponding settings, see Section 2.

A specific command-line argument *–rtime* has to be used to activate the real-time operation:

```
Anubis    -x Config.xml    --rtime    :gen:run_delay=30  :gen:max_epoch=300
```

The **run_delay** attribute specifies a data processing delay in seconds considering a latency of data stemming from real-time streams (usual value is a few seconds for observations and tens of seconds for products).

The **max_epoch** attribute defines how much data/products should be kept in internal containers. It should be optimally set by user with respect to its output requests, so all data are available for the processing, while not occupying too much computer memory. Hence, the number of epochs should optimally corresponds to input and output data/product sampling rates multiplied with a requested processing output window (e.g. *max_epoch=300* would correspond to hourly QC and RINEX with 10s data sampling rate).

7.1   Console configuration

A user console is then a dedicated built-in tool intended for a remote access and control of running processes. It is thus applicable mainly for a real-time mode of the software. The console is useful for browsing data in memory of actually running instance of the software and for monitoring input data and its decoding procedure. Multiple users can log in in the same time.

A username, password, and port number are defined for the access (port is only mandatory). All the items can be configured via XML attributes of the <console> element:

```
<console
    user="user"                          # optional
    pasw="pasw"                          # optional
    port="4444"                          # mandatory
/>
```

or via the command-line arguments:

```
:console:user="user"
:console:pasw="pasw"
:console:port="4444"
```

The attributes **user** and **pasw** are optional for defining user credentials for the console, i.e. if not specified, anybody knowing the port can access/use the console.

The console operations are performed on command requests from a user terminal. The tool **rlwrap** is recommended for a comfortable console utilization including an access to the past commands history. A connection to the console at a local host can be initiated with the **telnet** command: *Note: both tools have to be installed externally.*

```
rlwrap telnet localhost <port>
```

Basic console commands are listed by the **help** command request:

```
> help
----------------------
G-Nut console help
----------------------
        help            - this help
        exit            - close client
        info            - show client info and setting
        verb <int>      - select verbosity
        watch <cmd>     - periodically watch a command (see "watch help", for exit use "watch stop")
----------------------
Internal data containers
----------------------
        list data       - list available data containers
        set  data <key> - set active container
        show <args>     - monitor elements of chosen container, start with "show help"
----------------------
Internal data decoders
----------------------
        list coder      - list available data decoders
        set  coder <key> - set active decoder
        show            - monitor data decoding, start with "show help"
----------------------
Configuration (experimental)
----------------------
        config          - show full configuration
        config <commands> - command-line setting (e.g. config :out:verb=2 :gen:int=300)
        reload          - reload original configuration file
```

The **exit** command is called automatically after 10 min of inactivity. Some of the console outputs depend on the level of verbosity (**verb**) (set to 0 by default).

The **info** command returns an actual connection setting, verbosity, **watch** interval (see Section 7.4), and data object selected for working with (see Sections 7.2 and 7.3):

```
G-Nut console connect information:

        IP address:    127.0.0.1
        Port:          4444
        Client:        8
        Verbosity:     2
        Watch int.[s]: 5
        Data object:   orb
```

7.2   Data container browsing

The console allows users to browse actual data stored in the memory. The command **list data** is showing available objects for data representation:

```
> list data
G-Nut console data structures:
        (obs) - GRP_OBSERV | ALLOBS
        (orb) - GRP_EPHEM | ALLPREC
> set data obs
Selected container: obs
```

Selection lines consist of: *('key') - 'data type' | 'container type'.* The **set data 'key'** command enable to select a container to work with.

Data browsing commands corresponding to a selected container structure can be listed with **show help**:

```
> show help
----------------------
Container options
----------------------
  show + ...
        help        - this help
        stats       - basic statistics (verb dependent)
        <key> <args> - show container structure contents (depending on verb, given key and args)
----------------------
Container structure keys
----------------------
        -> sites
        -> epochs (use "last" for last epoch)
        -> satellites (sats)
        -> observations (obs)
```

Any data browsing output also depends on a verbosity settings.

An internal software data structure including a hierarchy is represented with a sequence of keys. These are thus needed to compose data browsing command for various structure depth levels. For example, a list of stations in the data structure is given by the command:

```
> show sites
  Number of sites: 46
        MATE00ITA0     ( epochs: 12)
        ZIM200CHE0     ( epochs: 19)
        WTZR00DEU0     ( epochs: 15)
```

The epochs of the station MATE00ITA0 are then reported with:

```
> show epochs MATE00ITA0
MATE00ITA0
        2021-11-29 10:48:15
        2021-11-29 10:48:20
        2021-11-29 10:48:25
        2021-11-29 10:48:30
```

The same logic is applied to show satellites, and even individual observations at a single epoch:

```
> show obs MATE00ITA0 "2021-11-29 11:05:20" G02
MATE00ITA0 2021-11-29 11:05:20 G02
        C1C 23503970.185938
        C2W 23503967.037639
        L1C 123514198.424547
        L2W 96244784.438200
        S1C 44.187500
        S2W 43.375000
```

The keyword **last** substitutes a particular epoch settings when considering the last epoch in the container. In case of the missing key, the console gives warning,e.g.:

```
> show obs MATE00ITA0 last
Too few arguments for keyword "obs". Should be "show obs <site epoch sat>"
```

7.3   Data decoder monitoring

Similarly as in the previous Section 7.2, console allows user to monitor a procedure of decoding data streams. The commands **list coder** and **set coder** works similarly as with the data

containers:

```
> list coder
G-Nut console coders:
        (0) - path: ntrip://BCEP00BKG0/***:***@***:2101
        (1) - path: ntrip://MATE00ITA0/***:***@***:2101
        (2) - path: ntrip://WTZR00DEU0/***:***@***:2101
        (3) - path: ntrip://ZIM200CHE0/***:***@***:2101
> set coder 2
Selected coder: 2
```

Each line represents a data stream and decoder and the **show help** gives available commands.

The coder monitoring is initiated with the first **show** while the console triggers buffering of decoded data. Executing a **show** command again prints then data stored previously (and cleans the memory).

```
> verb 3
Selected verbosity: 3
> show

> show
Message: Reading RTCM message 1046
Message: Galileo broadcast ephemeris epoch decoded: 2021-11-29 12:50:00 (GAL)  sat: E21
Message: Galileo broadcast ephemeris decoded:
E21 2021-11-29 12:50:00    INAV    759    759 93 107.00  000000000  1 1010000000  5    5
...
```

7.4 Repetitive command calls

While triggering a **watch** mode, the console calls automatically the command in an interval (5 seconds by default). Currently, the **show** command is allowed only available for the watch. The **watch help** provides the following hints:

```
> watch help
----------------------
Watch help
----------------------
Periodically spawn a command. Just write "watch" before a command.
Currently supports 1 watch per client.
----------------------
Watch options
----------------------
watch + ...
        help        - this help
        stop        - end the watch
        int <secs> - set watch interval (default is 5s)
----------------------
Allowed commands
----------------------
watch + ...
        show ...
```

Remark: the watch mode does not block the console, thus user may call another command simultaneously. However, user inactivity causes an exit even when the ***watch*** *is still running.*

7.5 Software reconfiguration (experimental)

The console also supports a selective reconfiguration of the software too when using the **config** command enabling 'command-line' configuration arguments, see Section 3:

```
config :out:verb=4 :gen:int=300
```

The **config** command without argument shows the current configuration.

The **reload** command triggers reloading of the initial configuration file. The path of the original configuration file is remembered (and cannot be altered), hence the original configuration can be modified and read again with this command.

7.6 Non-interactive monitoring

Using the console and an external **expect** linux tool, a user can perform also a non-interactive data monitoring. A regular monitoring task (e.g. a number of stations and observations of the software operated in real-time) can be performed using the following script:

```
#!/usr/bin/expect
spawn telnet localhost 4444
expect "Username"
send   "user\r"
expect "Password"
send   "pasw\r"
expect >
send   "verb␣1\n"
expect >
send   "set␣data␣obs\n"
expect >
send   "show␣stats\n"
expect >
send   "exit\n"
```

By executing the script from the command line, e.g. every hour by the **crontab** scheduler, requested information can be regularly output and stored:

```
spawn telnet localhost 4444
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Username: user
Password: pasw
Success!

Welcome to G-Nut/Anubis RT console! Start with "help".

> verb 1
Selected verbosity: 1
> set data obs
Selected container: obs
> show stats
t_gallobs::_stats
        Number of epochs:   409
        Number of stations: 4
>
```

Bibliography

[1] Václavovic P. and Douša J. G-Nut/Anubis - open-source tool for multi-GNSS data monitoring with a multipath detection for new signals, frequencies and constellations. In *International Association of Geodesy Symposia 150 Years*, volume 143, pages 775–782. International Association of Geodesy, Springer, 2016.

[2] Václavovic P., Douša J., and Györi G. G-Nut software library - state of development and first results. *Acta Geodynamica at Geomaterialia*, 10(4):431–436, 2013.

[3] Hatanaka Y. A compression format and tools for GNSS observation data. *Bulletin of the Geospatioal Information Authority of Japan*, 55(3):21–30, 2008. Available at https://www.gsi.go.jp/ENGLISH/Bulletin55.html.

[4] Weber G., Mervart L., Stürze A., Rülke A., and Stöcker D. BKG Ntrip Client, Version 2.12. In *Mitteilungen des Bundesamtes für Kartographie und Geodäsie*, volume 49. Frankfurt am Main, Bundesamtes für Kartographie und Geodäsie, 2016.

[5] NTRIP: Networked Transport of RTCM via Internet Protocol. Technical report, 2004. Available through https://www.rtcm.org.

[6] Douša J. GOP's consolidated multi-GNSSnavigation data archive, 2019. available at https://doi.org/10.24414/c4ba-kf16.

[7] IERS. SINEX – Solution (Software/technique) INdependent EXchange Format Version 2.02. Technical report, 2006. available at https://www.iers.org/IERS/EN/Organization/AnalysisCoordinator/SinexFormat/sinex.html.

[8] Zhao Q., Sun B., Dai Z., Hu Z., Shi Ch., and Liu J. Real-time detection and repair of cycle slips in triple-frequency GNSS measurements. *GPS Solutions*, 19(3):381–391, Jul 2014.

[9] Estey L. and Meertens C. TEQC: The Multi-Purpose Toolkit for GPS/GLONASS Data. *GPS Solution*, 3(1):42, 1999.