



Rinexmod & Autorino: Two Tools to Enable multi-GNSS and Near Real-time Data Acquisition and Pre-processing

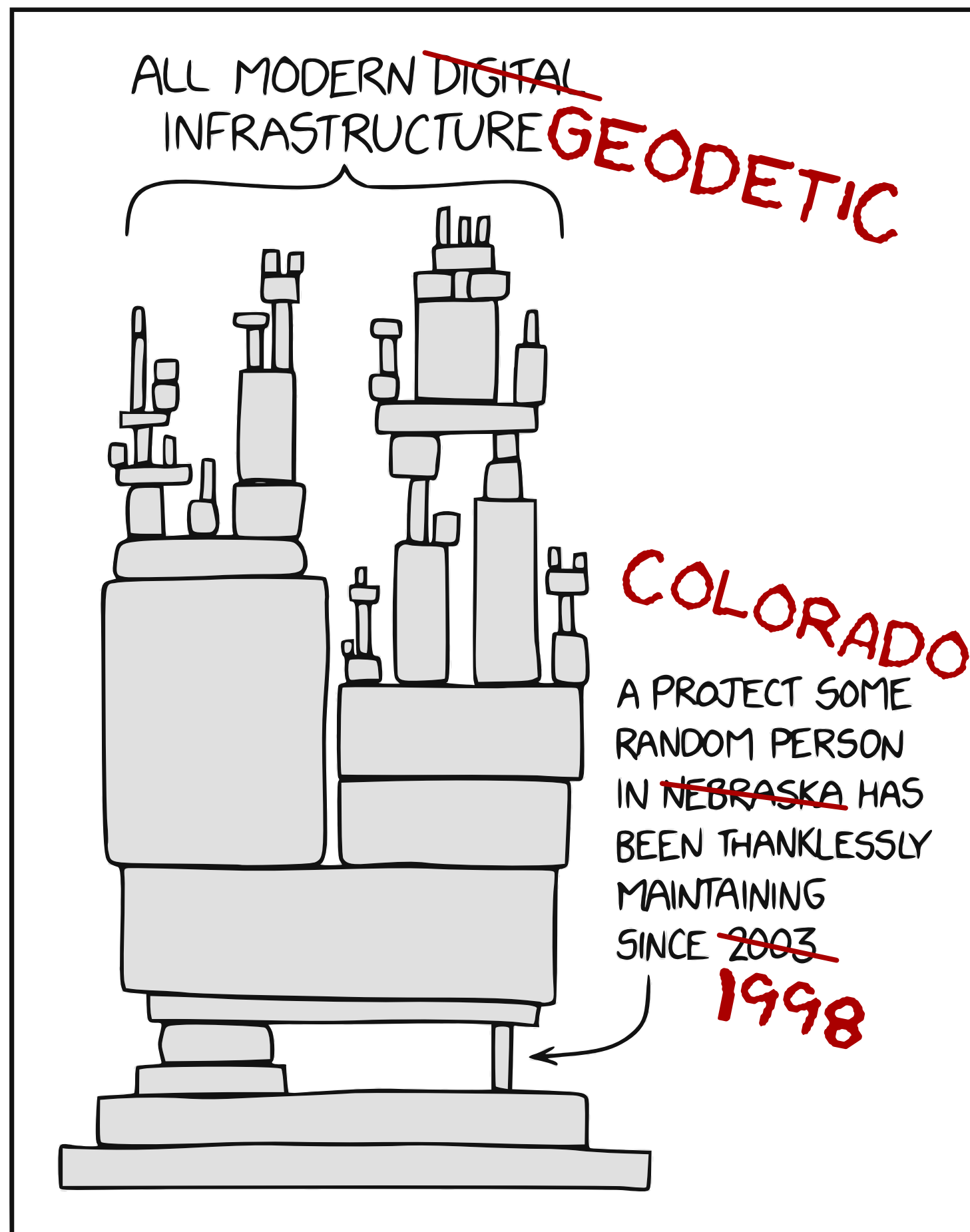


Pierre Sakic (sakic@ipgp.fr)¹, Patrice Boissier^{1,2}, Jean-Marie Saurel¹,
Cyprien Griot^{1,2}, Diane Pacaud^{1,2}, Aurélie Panetier¹

Université Paris-Cité, Institut de physique du globe de Paris, CNRS, IGN, F-75005 Paris, France

Observatoire volcanologique du Piton de la Fournaise, Institut de physique du globe de Paris, F-97418 La Plaine des Cafres, La Réunion, France

Introduction



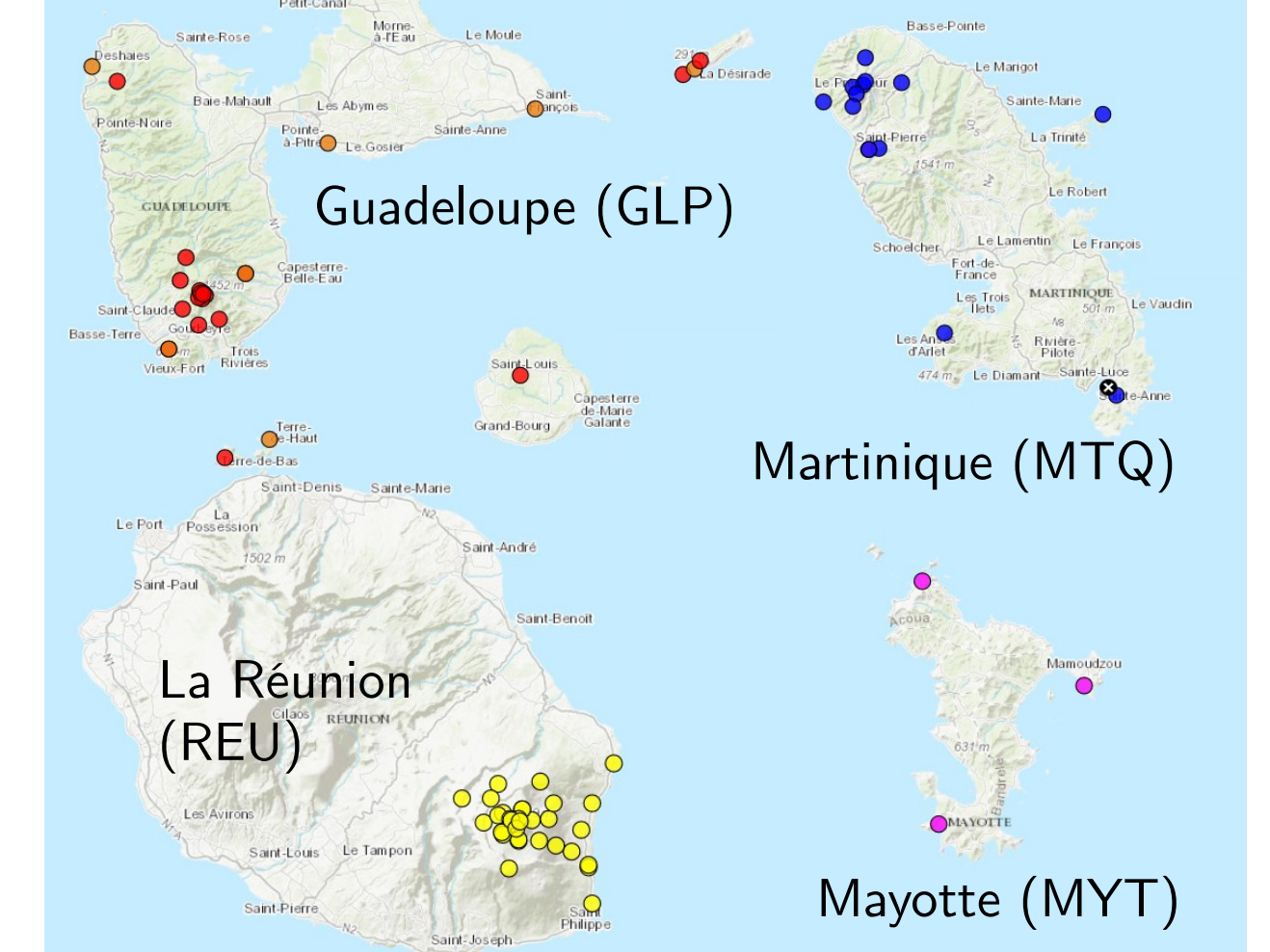
Freely inspired by XKCD #2347, R. Munroe

Two major developments in GNSS geodesy over the past 5 years

- New constellations reach maturity
 - Galileo declared operational in 2016, Beidou in 2018, 1st GPS satellite Block III satellite in 2018
 - ⇒ Increasing number of observations and new signals, rendering obsolete the RINEX2 format, dating back to 1993 (!)
- The end of teqc development in 2019
 - the "Swiss army knife" of data conversion and quality control
 - Widely used until now by the community
 - Delayed switchover to RINEX3
 - ⇒ Need to turn to OEM converters (one per manufacturer)

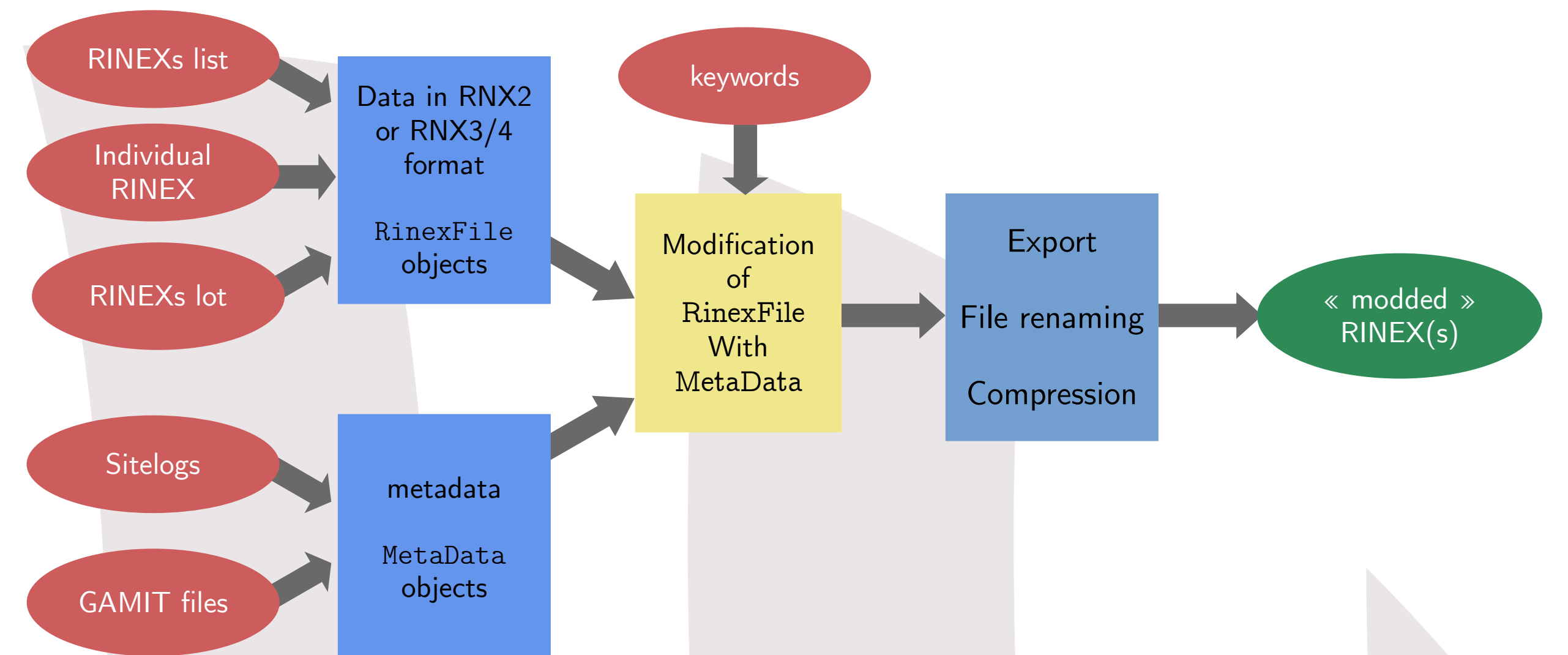
Identified needs

- A unified download/processing chain for the 3 IPGP's volcanological and seismological observatories (OVS), i.e. 85 GNSS stations, to enable a transition to near-real time
- Be as teqc-free as possible:
 - For header editing (⇒ *rinexmod*)
 - Convert raw data to RINEX3 (⇒ *autorino*)



More infos: <http://volobis.ipgp.fr/data/access-gnss-data>

rinexmod : schematic workflow



rinexmod

- *rinexmod* is a tool for batch-editing the headers of GNSS data files in RINEX format, and for renaming them correctly.
- supports RINEX versions 2 and 3/4, short and long naming conventions, and Hatanaka compression.
- developed in python3, can be run from the command line; or in API mode, by calling a frontend function or *RinexFile* objects.
- Metadata can be taken from a sitelogs file, GAMIT's station.info (for campaign-mode users) or entered manually. GeodesyML will be implemented in future releases



rinexmod synopsis (command line interface)

```
usage: rinexmod [-h] -i RINEXINPUT [RINEXINPUT ...] -o OUTPUTFOLDER
  [-s SITELOG] [-k KEY=VALUE [KEY=VALUE ...]]
  [-m MARKER] [-co COUNTRY] [-n NINECHARFILE]
  [-sti STATION_INFO] [-lfi LFILE_APRIORI]
  [-r RELATIVE] [-nh] [-c COMPRESSION] [-l] [-fs]
  [-fc] [-fr] [-ig] [-a] [-ol OUTPUT_LOGS] [-w] [-v]
  [-t] [-u] [-tol] [-mp MULTI_PROCESS] [-d]

RinexMod takes RINEX files (v2 or v3/4, compressed or not), rename
them and modify their headers, and write them back to a destination
directory

options:
  -h, --help            show this help message and exit

required arguments:
  -i RINEXINPUT [RINEXINPUT ...], --rinexinput RINEXINPUT
  [RINEXINPUT ...]
      Input RINEX file(s). It can be 1) a list
      file of the RINEX paths to process
      (generated with find or ls command for
      instance) 2) several RINEX files paths 3) a
      single RINEX file path (see -a/--alone for a
      single input file)
  -o OUTPUTFOLDER, --outputfolder OUTPUTFOLDER
      Output folder for modified RINEX files

optional arguments:
  -s SITELOG, --sitelog SITELOG
      Get the RINEX header values from file's
      site's sitelog. Provide a single sitelog
      path or a folder containing sitelogs.
  -k KEY=VALUE [KEY=VALUE ...], --modif_kw KEY=VALUE [KEY=VALUE ...]
      Modification keywords for RINEX's header
  -m MARKER, --marker MARKER
      A four or nine-character site code that will
      be used to rename input files. (apply also
      to the header's MARKER NAME, but a custom -k
      marker_name='XXXXX' overrides it)
  -co COUNTRY, --country COUNTRY
      A three-character string corresponding to the
      ISO 3166 Country code that will be used to
      rename input files.
  -n NINECHARFILE, --ninecharfile NINECHARFILE
      Path of a file that contains 9-char. site
      names (e.g. from the M3G database)
  -sti STATION_INFO, --station_info STATION_INFO
      Path of a GAMIT station.info file to obtain
      GNSS site metadata information (needs also
      -lfi option)
  -lfi LFILE_APRIORI, --lfile_apriori LFILE_APRIORI
      Path of a GAMIT apriori apr/L-File to obtain
      GNSS site position and DOMES information
      (needs also -sti option)
  -nh, --no_hatanaka
      Skip high-level RINEX-specific Hatanaka
      compression (performed per default). See
      also -c 'none'
  -c COMPRESSION, --compression COMPRESSION
      Set low-level RINEX file compression
      (acceptable values: 'gz' (recommended to
      fit IGS standards), 'Z', 'none')
  -tol, --tolerant_file_period
      the RINEX file period is tolerant and stick
      to the actual data content, but then can be
      odd (e.g. 07H, 14H...). A strict file period
      is applied per default (04H or 01D), being
      compatible with the IGS conventions
  -mp MULTI_PROCESS, --multi_process MULTI_PROCESS
      Number of parallel multiprocessing (default: 1)
```



rinexmod is freely
downloadable and usable
under GNU GPLv3 licence

github.com/IPGP/rinexmod

autorino

Assisted Unloading,
Treatment and
Organization of RINEX
Observations



- Centralizes GNSS data download, conversion and pre-processing operations
- Control of actions to be performed for each station from a standardized yaml configuration file
- Integration of station metadata from M3G

Download module

- Detects and manages the need to re-download raw GNSS data from receivers
- Optimized for near-real-time (5min, 15min)

Conversion module

- Common interface to run OEM converters seamlessly and uniformly to RINEX3

Manufacturer	Converter
Trimble	Docker trm2rinex
Septentrio	sbf2rin
Leica	mdb2rinex
Topcon	tps2rin
Universal	teqc (« legacy » mode)
BINEX	convbin (from RTKLIB)

RINEX header edition module

- *c.f. rinexmod*

Abstract

Satellite positioning geodesy has been undergoing a major transformation in recent years. The arrival of new GNSS constellations, notably the European Galileo declared operational in 2016 and the Chinese Beidou in 2018, as well as the deployment of GPS Block III the same year, marks the availability of new signals that offer a host of new prospects in terms of localization accuracy and speed.

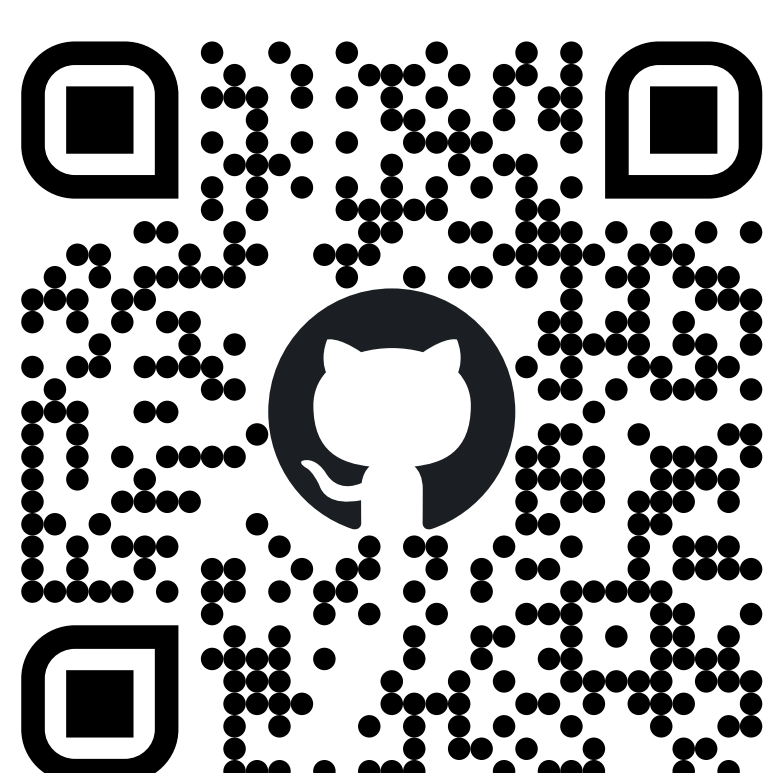
However, these new observables must be appropriately integrated and archived for future use. This problem no longer necessarily arises for worldwide tracking networks: They usually use homogeneous hardware and are components of the unified IGS's global network, benefiting from its well-established standards. However, correctly assimilating the new GNSS signals remains an issue for local and campaign-based network operators. Their networks are very often heterogeneous, comprising different generations of receivers from multiple manufacturers. This situation complicates a uniformized data workflow.

At the same time, the end of support in 2019 of the teqc utility developed by UNAVCO (now EarthScope since 2023) is also changing the game regarding GNSS data pre-processing. teqc was massively used as a "Swiss army knife" for RINEX conversion from proprietary vendor formats, metadata edition and quality control, but since no similar unified utility will probably ever be developed, alternatives must be found to replace it effectively.

Here, we present two acquisition and pre-processing tools designed to replace some teqc functionalities and enable new constellations and their associated signals' observations to be adequately considered in a unified chain.

The first tool is *rinexmod*, a RINEX header editing utility that substitutes teqc. It is compatible with versions 2 and 3/4, correctly handles the long naming convention, and can directly exploit station sitelogs as a source of metadata.

The second tool is *autorino* (for Assisted Unloading, Treatment, and Organization of RINEX Observations), which is designed for automated download and conversion of raw data from the main manufacturers' receivers (Leica, Septentrio, Topcon, Trimble, and BINEX) based on their respective official conversion utilities. A special focus is put on conversion to RINEX3/4 and near real-time capability (download frequency up to 5 min).



autorino is freely
downloadable and
usable under GNU
GPLv3 licence

github.com/IPGP/autorino