GSFC Technology Development Center

John Gipson, Ed Himwich, Mario Bérubé

Abstract This report summarizes the activities of the GSFC Technology Development Center (TDC) and describes plans for the future. The GSFC TDC develops station software including the Field System (FS), IVS session webpage software, and scheduling software (*sked*); hardware including tools for station timing and meteorology; scheduling algorithms, and operational procedures. It provides a pool of individuals to assist with station implementation, check-out, upgrades, and training.

1 General Information

The IVS GSFC Technology Development Center (TDC) develops hardware, software, algorithms, and operational procedures. It provides manpower for station visits for training and upgrades. Other technology development areas at GSFC are covered by other IVS components such as the GSFC Analysis Center. The current staff of the GSFC TDC consists of John Gipson, Ed Himwich, and Mario Bérubé. The remainder of this report covers the status of the main areas supported by the TDC.

2 Field System

The GSFC TDC is responsible for the development, maintenance, and documentation of the Field System (FS) software package. The FS provides equipment control at VLBI stations. It interprets the *.snp* schedule and *.prc* procedure files (both as prepared by *drudg* from the *.skd* schedule file). The FS controls the antenna, data acquisition hardware, and related ancillary equipment needed for making VLBI measurements. All major VLBI data acquisition backends are supported. The FS is customizable to allow it to control station-specific equipment. It is used at almost all of the IVS Network Stations (more than 35) and also at many stations that perform VLBI only for astronomical observations. The only major VLBI facilities not using the FS are the DSN, LBA, VLBA, and VERA.

2.1 Work This Period

The first 32/64-bit version of the FS, 10.0.0, was released during this period. This version also merged the main branches (9.13.x) and VGOS branches (9.12.x). The code was imported into *git* and is now distributed via *GitHub*. The import into *git* included all existent versions, over 130 FS9 versions, 17 FS8 versions (VENIX), and two older versions (HP-RTE/1000/A), back to version 5.5.

The first version with extensive support for DBBC3s, FS 10.1.0, was released. That work was funded by the EVN. This version also had many other improvements including streamlined *onoff* output, clean-up of *gnplt* comment handling, and changing the default *git* branch to *main*.

NVI, Inc./NASA Goddard Space Flight Center

2.2 Plans for the Future

A new FS version, 10.2, is expected to be released in 2023. It will have many improvements including support for longer experiment names (including IVS 12-character session codes), support for FSL11, and expanded DBBC3 support (funded by the EVN). Also, the *plotlog* utility for plotting the data in the logs has been expanded and brought up to date for VGOS.

Support for R2DBEs is expected, in FS 10.3, during 2023.

The new FS Linux distribution, FSL11, was developed in collaboration with the EVN during the period of this report but was released afterwards. This distribution is based on Debian *Stretch* and is expected to have support through June 2026.

3 Automation

The GSFC TDC is responsible for maintaining the IVS session webpages, displaying the Master Schedule, and providing information about the analysis and the scheduling of IVS sessions. A fully automated system has been operational for a few years with no major problems (see the 2019+2020 Biennial Report for a detailed description of the system). Some applications were modified to accept version 2 of the master files and the new naming convention for vgosDB files.

4 Ingest

GSFC has supported the IVS Data Centers by providing the initial version of the "ingest" software used to validate files uploaded by the VLBI community. The operational version is maintained by GSFC and the IVS Data Centers.

5 VLBI Communications Center (VCC)

The current VLBI communication method was developed more than 25 years ago and relies mainly on emails, the archiving system, and the dedicated people monitoring the information relevant to them. This is no longer a suitable communication system for operational VLBI. To improve the actual system, the GSFC group at NASA has developed a VLBI Communications Center (VCC) and tools for near real-time, machine-to-machine, two-way communication between IVS components. The VCC is a web service supported by a database and a message broker using formatted information designed for access by computers. The database keeps up-to-date data on schedules, catalogs, and all relevant information on various IVS components (e.g., station availability, latest SEFDs). The message broker is used to inform any IVS components that some data/information at the VCC are relevant for them. The VCC knows who acknowledges the message and who uploaded the schedule, allowing full traceability of data/information exchanges.

The VCC concept was presented at the IVS 2022 General Meeting, and a detailed description is available at `https://ivscc.gsfc.nasa.gov/publications/gm2022/25_berube_etal.pdf`.

The central VCC server has been set up on an NVI computer and is accessible by ssh tunnelling. A VCC-client python package has been developed to facilitate access to the VCC server by any IVS component. Specific applications have been developed for the Coordinating Center, Operation Centers, Analysis Centers, Correlators, and Network Stations. A dashboard is also available for visualizing the status of any VLBI session. The VCC-client package includes a configurable client that monitors the user "inbox" for new messages in near-real time. There is a special application for Network Stations that downloads new schedules and uploads logs, SEFDs, and any other relevant information to the VCC server. A specific interface was developed to notify any affected IVS component about antenna down time.

Because each IVS component has limited roles, the VCC server is able to validate data based on the incoming information and the sender identity. For example, only the Coordinating Center can modify the Master Schedule. Schedule files can only be uploaded by the designated Operation Center. To ensure that these criteria are enforced, each user needs a special key to access the VCC. The keys are provided by the VCC manager (contact Mario Bérubé at mario.berube@nviinc.com for details). Many members of the same organization can have their own key so that everybody can receive the same messages.

The VCC-client package requires Python 3.8 or higher. It has been tested on FS 11 for Network Stations and could be installed on FS 10 by upgrading the Python package. For other components, the package was tested on Ubuntu, MacOS, and Windows.

We plan to test the VCC client at some stations and provide a demonstration at TOW 2023.

6 *sked* and *drudg*

The GSFC TDC is responsible for the development, maintenance, and documentation of *sked* and *drudg*. These two programs are very closely related, and they operate as a pair for the preparation of the detailed observing schedule for a VLBI session and its proper execution in the field. In the normal data flow for a geodetic schedule that is written with *sked*, *sked* is run first at an Operation Center to generate the *.skd* file that contains the full network observing schedule. Then each station uses the *.skd* file as input to *drudg* to make the FS schedule and procedures for that station. Catalogs are used to define the equipment, stations, sources, and observing modes that are selected when writing a schedule with *sked*.

Changes to *sked* and *drudg* are driven by changes to equipment and by feedback from the users. The following sub-section summarizes some of the important changes to these programs during the report period. This summary includes only the most important bugs that were found and fixed over this period. A more complete summary of the changes can be found in the `changelog.txt` files associated with *sked* and *drudg*.

6.1 drudg Changes

Here is a summary of some important changes made in drudg during this period.

- The major change to drudg was adding support for DBBC3.
- Obsolete code dealing with tape recorders was removed on an ad hoc basis—that is, no systematic attempt was made to remove it, but when it was encountered in the normal course of business it was removed. Examples of this include getting rid of references to tape passes, tape length, tape speed, S2 recorders, and so on.
- Drudg was made compatible for both *gfortran* and *fortran77*.
- In support of the new IVS session naming convention, *drudg* now supports 16-character session codes.

6.2 sked Changes

Here is a summary of some important changes to *sked*.

- *sked* was modified to reduce output written to the screen. This has the side effect that the program runs more smoothly.
- To avoid damaging the electronics, the antenna must travel around, not through, the radar mask. This is built into the FS, but *sked* did not know about this. Because of this, in many observations not enough time was allocated for GGAO to slew to the source. We wrote a custom slewing algorithm for GGAO based on what the FS actually does. This eliminated late-on-source error messages in the FS.
- The slew speed for the twin telescopes at Onsala depends not only on the location but on the direction in which the antennas are moving. As an antenna approaches a limit, it slows down. We wrote a custom algorithm to model this behavior correctly.

6.3 sked Catalogs

The GSFC TDC is responsible for maintaining the *sked* catalogs which are used in scheduling VLBI sessions. These catalogs include information about sources, antennas, equipment at stations, and observing modes. These catalogs are used by both *sked* and *VieSched*++.

A major change during this period was to make the catalogs available via *Github*. Any time a change is made, it is first tested locally at GSFC, and then the change is pushed to *Github*.

- flux.cat (which has simple models of source fluxes) was updated on a monthly basis. Timely updates are crucial because the source fluxes change with time.
- equip.cat (which contains information about the equipment at a station) was updated when the equipment changed. This catalog also contains information about station SEFDs, so it was updated whenever a station ran warm and updated again when it ran cold.
- Other catalogs (e.g., lo.cat, modes.cat, rx.cat, rec.cat) were updated as necessary. This might happen if equipment at a station was changed or a mode was not defined for a station.