GSFC Technology Development Center Report

Ed Himwich, John Gipson

Abstract

This report summarizes the activities of the GSFC Technology Development Center (TDC) for 2008 and forecasts planned activities for 2009. The GSFC TDC develops station software including the Field System, 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. Technology Center Activities

The GSFC IVS 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 and Ed Himwich, both employed by NVI, Inc. The remainder of this report covers the status of the main areas of development that are currently being pursued.

2. Field System

The GSFC TDC is responsible for 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). 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 all the IVS network stations (over 30) and also at many stations that do VLBI only for astronomical observations. The only major VLBI facilities not using it are the VLBA and VERA.

During this period some of the new features that were released in FS version 9.10 were:

- A `-No_X` option was added to the *fs* program to allow it to be started from a non-X-server terminal. The *oprin* program can be run in a separate terminal window and has a new `end_oprin` command to allow it to be terminated gracefully. In this case, the user can examine the *logfile* using the

$$\texttt{tail -f log } \textit{logfile}$$

  command in another window.

- The `TNX` command was expanded to distinguish multiple instances of the same error mnemonic.

- The *plotlog* utility was expanded to handle PCal amplitude Tsys normalization, improved phase difference plots, K5 PCal plotting, and plot selection.

- The new *monpcal* utility for realtime PCal display was added.

- The *logpl* log plotting utility was upgraded with a new expanded version written in Python. The new version includes: XY plots, plotting of pcal output from decode4 commands, and many new plotting capabilities.

- The FS Linux 7 (FSL7) distribution (based on Debian *etch*) was developed and deployed. This effort included a complete update of the installation and upgrade documentation.

- A new version of the *gnplt* gain analysis utility was developed in Python. This version is much faster and provides more flexibility in handling the data, as well as better graphics. It will be distributed next year.

- Various upgrades were made to the source code to fix bugs and to make the code compatible with more compilers and specifically the new FSL7 distribution.

In the next year, several other improvements are expected; among these are: (1) Support for DBBC and DBE racks, (2) a complete update to the documentation and conversion to a more modern format that will be easier to use; (3) conversion of the FORTRAN source to use the *gfortran* compiler; this will enable use of the source level debugger, *gdb*, for development and field debugging; (4) use of *idl2rpc* for remote operation; (5) *chekr* support for Mark 5A and 5B systems; (6) use of the Mark IV Decoder for phase-cal extraction in the field; (7) FS Linux 8 (based on Debian *lenny*) development; (8) support for periodic firing of the noise diode during observations; and (9) distribution of the new *gnplt*.

3. 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 geodetic schedules, first SKED is run at the Operation Centers to generate the .skd file that contains the full network observing schedule. Then stations use the .skd file as input to DRUDG for making the control files and procedures for their station. Catalogs are used to define the equipment, stations, sources, and observing modes which are selected when writing a schedule with SKED.

Changes to SKED and DRUDG are driven by changes in equipment and by feedback from the users. The following summarizes some of the important changes to these programs in 2008.

3.1. SKED

The following changes were made to SKED.

- The installation procedure for SKED was simplified. Previously users had to modify makefiles for SKED and four associated libraries (skdrut, matrix, lnfch, and curses) so that they would point to the correct compilers and system libraries. This procedure was modified so that the paths to the compilers and libraries are now environment variables and are set via the script `set_misc`. All the user has to do is to modify this script, execute it, and then run `sked_make`, which will make all of the required SKED libraries and then compile and link SKED.

- For about 10 scans during CONT08, a station arrived late on source. A seemingly unrelated problem was that occasionally a station would be scheduled to be in two scans at the same

time. Both issues were traced to a bug in the 'tag-along' mode. This bug was fixed.

- Previously when the `Remove` command was used to remove stations from scans, SKED would not check how many stations were left in the scan. This could lead to a situation where only one station was observing. This bug was fixed.

- The scheduling algorithm was improved to give 10-15% more observations.

- New symbolic time names were introduced: `Start` or `Begin` will move you to the start of the schedule; `End` to the end.

- Some obsolete commands were eliminated, and some new commands were introduced.

Plans for the next year include: (1) updating documentation, (2) making VEX format native, and (3) supporting *CLEAN* components for source flux models.

3.2. DRUDG

DRUDG is a mature program, and there were not many changes during 2008. The following changes were made to DRUDG.

- Dymo printing was modified to give stations greater flexibility. Some options which were hard-coded can now be specified in the `skedf.ctl` file. Also, the *print2dymo* script was modified to work with Linux versions 5, 6, and 7. All the user has to do is comment out or uncomment the appropriate lines. In addition, this script can now print out debugging information.

- The format of the antenna pointing files generated by SKED, and used by NRAO, was modified. Apparently NRAO had been hand editing these files for years. This was not a big issue previously, when the RDV schedules' recording was `CONTINUOUS`. However, starting with RDV70, NRAO requested that the schedules be generated to use `ADAPTIVE` recording, and the number of changes increased dramatically. These pointing files will no longer need to be edited.

- There were several other minor bug fixes and enhancements which are covered in the DRUDG release notes.

Plans for the next year include: (1) a documentation update and (2) support for new rack types.