John Gipson: IVS Working Group 4: VLBI Data Structures, IVS 2010 General Meeting Proceedings, p.187–191
http://ivscc.gsfc.nasa.gov/publications/gm2010/gipson3.pdf

IVS Working Group 4: VLBI Data Structures

John Gipson

NVI, Inc., Code 698, NASA Goddard Space Flight Center, Greenbelt, MD, 20771

e-mail: john.m.gipson@nasa.gov

Abstract

In 2007 the IVS Directing Board established IVS Working Group 4 on VLBI Data Structures. This note discusses the current VLBI data format, goals for a new format, the history and formation of the Working Group, and a timeline for the development of a new VLBI data format.

1. Introduction

At the 15 September 2007 IVS Directing Board meeting I proposed establishing a "Working Group on VLBI Data Structures". The thrust of the presentation was that, although the VLBI database system has served us very well these last 30 years, it is time for a new data structure that is more modern, flexible, and extensible. This proposal was unanimously accepted, and the board established IVS Working Group 4. Quoting from the IVS Web site [1]: *"The Working Group will examine the data structure currently used in VLBI data processing and investigate what data structure is likely to be needed in the future. It will design a data structure that meets current and anticipated requirements for individual VLBI sessions including a cataloging, archiving, and distribution system. Further, it will prepare the transition capability through conversion of the current data structure as well as cataloging and archiving softwares to the new system."*

Any change to the VLBI data format affects everyone in the VLBI community. Therefore, it is important that the working group have representatives from a broad cross-section of the IVS community. Table 1 lists the current members of WG4 together with their affiliation or function. The initial membership was arrived at in consultation with the IVS Directing Board. While we wanted to ensure that all points of view were represented we also wanted to make sure that the size did not make WG4 unwieldy. The current composition and size of WG4 is a reasonable compromise between these two goals. My initial request for participation in WG4 was enthusiastic: everyone I contacted agreed to participate with the exception of an individual who declined because of retirement.

Table 1. Membership in Working Group 4.

Chair	John Gipson
Analysis Coordinator	Axel Nothnagel
Haystack/Correlator Representative	Roger Cappallo
GSFC/Calc/Solve	David Gordon
JPL/Modest	Chris Jacobs Ojars Sovers
Occam	Oleg Titov Volker Tesmer
TU Vienna	Johannes Böhm
IAA	Sergey Kurdobov
Steelbreeze	Sergei Bolotin
Observatorie de Paris/PIVEX	Anne-Marie Gontier
NICT	Thomas Hobiger Hiroshi Takiguchi

2. History of Working Group 4

WG4 held its first meeting at the 2008 IVS General Meeting in St. Petersburg, Russsia. This meeting was open to the general IVS community. Roughly 25 scientists attended: ten WG4 members and fifteen others. This meeting was held after a long day of proceedings. The number of participants and the lively discussion that ensued is strong evidence of the interest in this subject. A set of design goals, displayed in Table 2, emerged from this discussion. In some sense the design goals imply a combination and extension of the current VLBI databases, the information contained on the IVS session Web-pages, and much more information [2].

During the next year the working group communicated via email and telecon and discussed how to meet the goals that emerged from the St. Petersburg meeting. A consensus began to emerge about how to achieve most of these goals.

Table 2. Key goals of the new format.

Goal	Description
Provenance	Users should be able to determine the origin of the data and what was done to it.
Compactness	The data structure should minimize redundancy and the storage format should emphasize compactness.
Speed	Commonly used data should be able to be retrieved quickly.
Platform/OS/ Language Support	Data should be accessible by programs written in different languages. running on a variety of computers and operating systems.
Extensible	It should be easy to add new data types.
Open	Data should be accessible without the need of proprietary software.
Decoupled	Different types of data should be separate from each other.
Multiple data levels	Data should be available at different levels of abstraction. For example, most users are interested only in the delay and rate observables. Specialists may be interested in correlator output.
Completeness	All VLBI data required to process (and understand) a VLBI session from start to finish should be available: schedule files, email, log-files, correlator output, and final 'database'.
Web Accessible	All data should be available via the Web.

The next face-to-face meeting of WG4 was held at the 2009 European VLBI Meeting in Bordeaux, France. This meeting was also open to the IVS community. At this meeting a proposal was put forward to split the data contained in the current Mark III databases into smaller files which are organized by a special ASCII file called a wrapper. I summarized some of the characteristics and advantages of this approach. Overall the reaction was positive.

In the summer of 2009 we worked on elaborating these ideas, and in July a draft proposal was circulated to Working Group 4 members. Concurrently I began a partial implementation of these ideas and wrote software to convert a subset of the data in a Mark III database into the new format. This particular subset included all data in NGS cards and a little more. The subset was chosen because many VLBI analysis packages including Occam, Steelbreeze, and VieVS use NGS cards as input. In August 2009 we made available, via anonymous ftp, three VLBI sessions in the

new format: an Intensive, an R1, and an RDV.

In the fall of 2010, Andrea Pany of the Technical University of Vienna developed an interface to VieVS working with the draft proposal. During this process the definition of a few of the data items needed to be clarified, which emphasizes the importance of working with the data hands on. At NASA's Goddard Space Flight Center, Sergei Bolotin interfaced a variant of this format to Steelbreeze. Steelbreeze uses its own proprietary format, and one motivation for interfacing to the new format was to see if there was a performance penalty associated with using the new format. Bolotin found a performance penalty of 40 μs/observation[1]. There are currently $\simeq$ 6 million VLBI observations, which translates into an extra 6 minutes to process all of the VLBI data—a modest price to pay for the many advantages the new format brings.

3. Overview of New Organization

In a paper of this size it is impossible to completely describe the new organization and format. Instead, I will briefly describe three of the key components: 1) modularization; 2) storing data in NetCDF files; and 3) using wrapper files to organize the data.

3.1. Modularization

A solution to many of the design goals of Table 3 is to modularize the data, that is to break up the data associated with a session into smaller pieces. These smaller pieces are organized by 'type'; e.g., group delay observable, met-data, editing criteria, station names, and station positions. In many, though not all, cases, each 'type' corresponds to a Mark III database L-code. Different data types are stored in different files, with generally only one or a few closely related data types in each file. For example, it might be convenient to store all of the met-data for a station together in a file. However, there is no compelling reason to store the met-data together with pointing information. Splitting the data in this way has numerous advantages, some of which are outlined below. The first three directly address the design goals. The remaining are other advantages not originally specified, but are consequences of this design decision.

1. **Separable.** Users can retrieve only that part of the data in which they are interested.
2. **Extensible.** As new data types become used, for example, source maps, they can be easily added without having to rewrite the whole scheme. All you need to do is specify a new data type and the file format.
3. **Decoupled.** Different kinds of data are separated from each other. Observables are separated from models. Data that won't change is separated from data that might change.
4. **Flexible.** Since different data is kept in different files, it is easy to add new data types.
5. **Partial Data Update.** Instead of updating the entire database, as is currently done, you only need to update that part of the data that has changed.[2]

Data will also be organized by 'scope', that is how broadly applicable it is: Does it hold for the entire session, for a particular scan, for a particular scan and station, or for a particular observation. The current Mark III database is observation oriented: all data required to process a

[1] No effort was made to optimize the interface. With optimization this figure should be less.

[2] This is done by making a new version of the relevant file, keeping the old one intact.

given observation is stored once for each observation. This results in tremendous redundancy for some data. For example, in an N-station scan, each station will participate in $N-1$ observations. Station met-data, which is the same for all observations in a scan, is stored once for each observation instead of once per scan. This results in an $(N-1)$-fold redundancy. Organizing data by scope allows you to reduce redundancy.

3.2. Organizing Data by Wrappers

The main disadvantage of breaking up the VLBI data into many smaller files is that you need some way of organizing the files. This is where the concept of a wrapper comes in. A wrapper is an ASCII file that contains pointers to VLBI files associated with a session. VLBI analysis software parses this file and reads in the appropriate data. As new data types are added, or as data is updated, new versions of the wrapper can be generated. The wrapper concept is illustrated schematically in Figure 1. The wrapper can serve several different purposes:

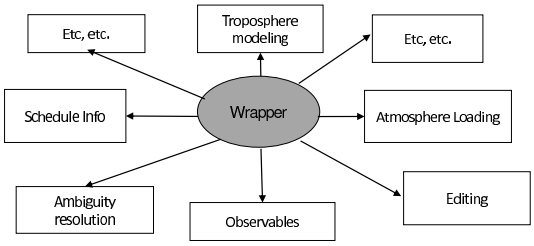

Figure 1. Wrappers organize the data.

1. The wrapper can be used by analysis programs to specify what data to use.
2. The wrapper allows analysts to experiment with 'what if' scenarios, for example, to use another analysts editing criteria which is stored in a file. All you need to do is to obtain the editing file and modify the wrapper to point to it.
3. Because of the general structure of the wrapper, different analysis packages can use different wrappers that point to different subsets of the VLBI data.
4. The wrapper is a convenient means of signaling to the IVS data center what information is required. In this scenario, a user writes a wrapper with pointers to the relevant files and sends it to the IVS data center. The data center packages the data in tar-file and makes it available. (Since all data is available via FTP this is a convenience.)

3.3. NetCDF as Default Storage Format

Working Group 4 reviewed a variety of data storage formats including NetCDF, HCDF, CDF, and FITS. In some sense, all of these formats are equivalent—there exist utilities to convert from one format to another. Ultimately we decided to use NetCDF, because it has a large user community and because several members of the Working Group have experience with using NetCDF. At its most abstract, NetCDF is a means of storing arrays in files. The arrays can be of different sizes and shapes, and contain different kinds of data—strings, integer, real, double, etc. Most VLBI data used in analysis is some kind of array. From this point of view

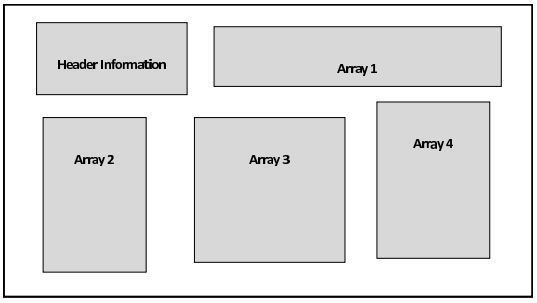

Figure 2. A NetCDF file is a container for arrays.

using NetCDF is a natural choice. These files can contain history entries which aid in provenance. Storing data in NetCDF format has the following advantages:

1. **Platform/OS/Language Support.** NetCDF has interface libraries to all commonly used computer languages running on a variety of platforms and operating systems.
2. **Speed.** NetCDF is designed to access data fast.
3. **Compactness.** The data is stored in binary format, and the overhead is low. A NetCDF file is much smaller than an ASCII file storing the same information.
4. **Open.** NetCDF is an open standard, and software to read/write NetCDF files is freely available.
5. **Transportability.** NetCDF files use the same internal format regardless of the machine architecture. Access to the files is transparent. For example, the interface libraries take care of automatically converting from big-endian to little-endian.
6. **Large User Community.** Because of the large user community, there are many tools developed to work with NetCDF files.

4. Next Steps

The immediate next step is for various VLBI software developers to develop interfaces to the new format. It is likely that this will lead to further refinement.

- VieVS already has an interface to the new format.
- At Goddard we plan on interfacing solve to the new format in the fall of 2010. Initially the new format will be a replacement for solve superfiles and will be used in global solutions. Gradually we will modify *calc/solve* to use the new format at earlier stages of analysis.
- Oleg Titov (private communication) plans on developing an interface to Occam in the winter of 2010 or the spring of 2011.

Beginning in the spring of 2011 we will make a subset of the data in the current Mark III databases available in the new format. This subset will include all data currently in the NGS cards as well as all data required for *calc/solve* analysis. The size of this subset will expand through 2011 until by the end of the year all of the data that is currently in the Mark III database format will be available in the new format. We will also work on gathering correlator output files and making these available on the IVS data center.

In March 2011, there will be an open meeting of IVS Working Group 4 at the European VLBI meeting. This will provide another opportunity for the VLBI community to provide further feedback and for fine-tuning the specifications if they are required. Working Group 4 will also work on its final report, which will be presented to the IVS Directing Board. We anticipate that the last meeting of WG4 will be at the 2012 General Meeting, at which point it will dissolve.

References

[1] http://ivscc.gsfc.nasa.gov/about/wg/wg4/index.html

[2] Gipson, J., IVS Working Group 4 on VLBI Data Structures, The 5th IVS General Meeting Proceedings, 2008, p. 143-152.