

e-VLBI with the SFXC correlator

Mark Kettenis**JIVE**E-mail:* kettenis@jive.nl**Aard Keimpema***JIVE**E-mail:* keimpema@jive.nl**Des Small***JIVE**E-mail:* small@jive.nl**Damien Marchal***University of Amsterdam[†]**E-mail:* damien.marchal@lifl.fr

The SFXC software correlator developed at JIVE has reached state of maturity where we are concentrating on optimization and testing on “real-world” data sets. In this context, distribution of sampled data is an important issue. We will discuss an approach that makes use of existing Grid middleware suitable for near-realtime correlation that can potentially be distributed over multiple clusters. We will also discuss an entirely different approach, more suitable for real-time correlation where data is streamed directly into a cluster using the GEANT2 AutoBAHN and Internet2 DCN infrastructure.

*The 8th International e-VLBI Workshop
22-26 June 2009
Madrid, Spain*

*Speaker.

[†]Currently at Laboratoire d'Informatique Fondamentale de Lille

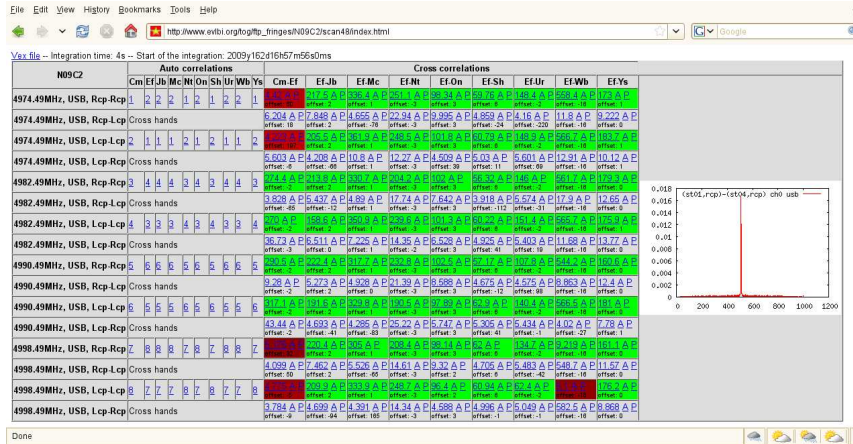


Figure 1: Web page presenting the results of an FTP fringe test

1. Introduction

Like most software correlators, SFXC is an FX-correlator. It has its origin in the correlator that was used to track the Huygens probe on its descent to the surface of Titan [2]. Since then a lot of work has been put into turning it into a general purpose VLBI correlator for astronomy [4]. Most of this work was done as part of the SCARIE (Software Correlator Architecture Research and Implementation for e-VLBI) and FABRIC (Future Arrays of Broadband Radio-telescopes on Internet Computing) projects at JIVE. In its current state the correlator is VEX-driven and can read Mark4, VLBA and Mark5B input data. The correlator model is based on Calc 10 [3]; identical to the model used by the EVN Mark IV correlator operated by JIVE. The correlator output can be easily translated into FITS-IDI such that it can be read into AIPS for further processing [5].

2. FTP-fringe tests

One of the first e-applications of the SFXC software correlator have been the so-called FTP-fringe tests. These tests have become a standard part of the observing sessions of the European VLBI Network (EVN). They are conducted to spot any problems with the setup at the telescopes before the actual science observation starts. SFXC has been used for this purpose since May 2007. Before that date, we used the KSRC/NICT software correlator package [1]. Using a correlator that is developed “in-house” for this purpose has many benefits. It is much easier to add support for new input data formats and new correlator modes.

For these tests telescopes transfer a small amount of data (typically less than 10 seconds) to JIVE using the FTP protocol. This transfer is started automatically by the field system running at the telescope. After the data arrives at JIVE a support scientist runs the correlator with the desired parameters on a single four-way SMP machine. The results are then published through automatically generated web pages (see Figure 1). These web pages have been designed to make it easy for the telescope operators to spot problems. This approach works quite well for small amounts of data but isn’t really suitable for complete experiments.

3. Distributed correlation

A system was designed that includes a workflow manager, a VLBI grid broker and some additional modules that run at the data source and on the clusters that will be used to do the correlation. During the implementation of this design we encountered two places where VLBI-specific knowledge was required. Since our collaborators in Poznan who implemented the workflow manager and broker are not VLBI experts, we decided to hide some of the details inside additional web services.

The VLBI community uses the VEX file format to describe VLBI experiments. Parsers for this file format are available for a few languages like C, FORTRAN and Perl, but not for Java, which is the de-facto standard language to implement Grid services. On top of that interpreting VEX requires a substantial amount of domain-specific knowledge. So a webservice was implemented that creates a template for a correlation control file from a VEX file. The workflow manager then turns this template into a full correlator control file using input from the operator. For the benefit of the workflow manager, this correlator control file is also annotated with some additional information from the VEX file. For example, the positions of the telescopes are provided such that the workflow manager can place the telescopes on a map presented to the user. More details about this system can be found in [6].

Unfortunately, in this system data will hit disk storage several times in the process of being correlated: when the data is divided into chunks; when the data is copied to storage nodes using grid-ftp; and when the data is copied onto local storage on the cluster where the correlation jobs are run. While we have succeeded in correlating data using two clusters, the overall speed seems not to be sufficient to do real-time e-VLBI.

4. Bandwidth-on-Demand

Whereas hardware correlators are pretty much tied to a specific location, software correlators can run anywhere, as long as there are enough computing, storage and networking resources. Especially for real-time e-VLBI we need enough bandwidth between the telescopes and the correlator to guarantee timely delivery of the baseband data. This is further complicated by the fact that these network connections are likely to cross multiple networking domains because of the long-baseline nature of VLBI.

Given the fact that many VLBI arrays (for example the EVN) do not operate continuously, dedicated network circuits are not strictly necessary. Compared to a traditional VLBI array, operation of an e-VLBI array tends to be much more dynamic. Liberated from the logistics of shipping disks to the telescopes it becomes far easier to organize Target of Opportunity (ToO) and ad-hoc observations. And provided the network connections are in place, including additional telescopes becomes a lot easier as well.

This makes e-VLBI with a software correlator a splendid application for the various bandwidth-on-demand systems that are being developed around the world. One of such systems is AutoBAHN, which is being developed in Europe as part of the GEANT2 project. AutoBAHN is particularly interesting because it is already intrinsically cross-domain. It allows circuits across networks of participating NRENs and across the GEANT backbone. It also interfaces to other comparable systems such as Internet2's DCN services.

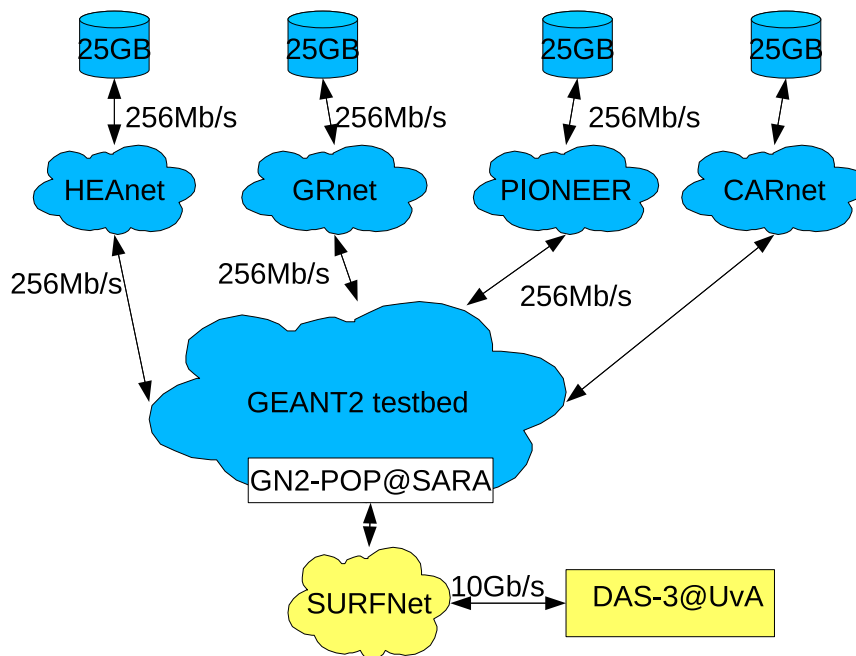


Figure 2: Network setup of AutoBAHN demo

In the second half of 2008, we set up two demonstrations together with the AutoBAHN team. In the first demo, at the GLIF workshop in Seattle, a few minutes worth of baseband data was distributed across test machines in five different European countries and correlated on the DAS-3 cluster at the University of Amsterdam in the Netherlands. Shortly before starting the correlator, circuits between the participating “telescopes” and the correlator were requested using the web services interface provided by AutoBAHN. Data was streamed into the correlator directly, never hitting disk storage again after being read on the test machines.

For the second demo, at the Supercomputing ’09 conference in Austin, we added an additional site near Boston, in the United States, to demonstrate the interface between AutoBAHN and Internet2.

Since SURFNet, the NREN of the Netherlands, does not implement AutoBAHN and does not yet interface their system for allocating bandwidth on demand to the AutoBAHN system, we had to set up static routing between the GEANT POP in Amsterdam and the DAS-3 cluster (which actually are in the same building).

Both demonstrations were successful, with data from four “telescopes” correlated at 256 Mb/s on about 60 nodes of the DAS-3 cluster. The bottleneck here was still the computation and not the network. By bypassing parts of the correlation algorithm a throughput of 512 Mb/s was obtained.

5. Conclusions

The simple FTP-based scheme used for the EVN FTP fringe tests works well for the purpose it was developed for, but the approach does not scale to complete observations.

The grid-based approach to distributed correlation has been shown to work. However, so far the additional overhead seems to be higher than the possible increase in performance gained from using more computational resources.

The bandwidth on demand system seems to be a good match for the software correlator application. We have shown that a throughput of 256 Mb/s can be easily sustained and that data rates of at least 512 Mb/s should be within reach given enough computational resources. How well this will work out for real-time e-VLBI still remains to be seen.

6. Acknowledgements

We thank the AutoBAHN team for their collaboration during the joint demonstrations. The FABRIC project is part of EXPReS (Express Production Real-time e-VLBI Service). EXPReS is an Integrated Infrastructure Initiative (I3), funded under the European Commission's Sixth Framework Programme (FP6), contract number 026642, from March 2006 through August 2009. The SCARIE project is funded by the Netherlands Organisation for Scientific Research (NWO).

References

- [1] T. Kondo, M. Kimura, Y. Koyama, H. Osaki, *Current Status of Software Correlators Developed at Kashima Space Research Center*, in proceedings of *International VLBI Service for Geodesy and Astrometry 2004 General Meeting*, Ottawa, 2004
- [2] L. I. Gurvits, S. V. Pogrebenko, I. M. Avruch, *VLBI as a tool for planetary science missions*, in proceedings of *European Planetary Science Congress 2006*
- [3] D. Gordon, D. MacMillan, K. Baver, *Calc 10 Implementation*, in proceedings of *International VLBI Service for Geodesy and Astrometry 2006 General Meeting*
- [4] N. Kruithof, *e-VLBI using a Software Correlator*, in proceedings of *INGRID 2007*
- [5] Y. Pidopryhora, A. Keimpema and M. Kettenis, *The latest tests of the SFXC correlator*, in proceedings of *The 8th International e-VLBI Workshop*, PoS (EXPReS09) 046
- [6] D. Stokłosa, N. Meyer, *User interaction and workflow management in Grid enabled e-VLBI experiments*, in proceedings of *The 8th International e-VLBI Workshop*, PoS (EXPReS09) 047