

# Mark 5/e-VLBI Newsletter

MIT Haystack Observatory

December 2004

Issue #7

The Mark 5/e-VLBI Newsletter is issued from time to time to keep users informed regarding Mark 5/e-VLBI progress, plans, problems, solutions and workarounds. All back issues of the Newsletter are available at the Mark 5 web site at [www.haystack.edu/mark5](http://www.haystack.edu/mark5). We invite input from anyone on subjects we should discuss or questions that need answers; please send them to [mark5@haystack.mit.edu](mailto:mark5@haystack.mit.edu). General information about the Mark 5 and e-VLBI is available at the Haystack web site at <http://www.haystack.edu/>.

## Contents of this Issue

1. New system for disk-module labeling
2. Importance of system pre-checks (particularly at 1 Gbps)
3. Suggested list of maintenance spares for Mark 5
4. Recovering data from overwritten or abnormally terminated scans
5. Disk-FIFO mode now available
6. Mark 5B status update
7. New Mark 5 memos available
8. 512 Mbps real-time e-VLBI demo at SuperComputing 2004
9. Production e-VLBI facility set up at Haystack
10. VSI-E development progress
11. Experiment-Guided Adaptive Endpoint (EGAE) progress
12. e-VLBI file-naming conventions
13. 10 Gbps link established between Westford and Haystack

## 1. New system for disk-module labeling

Almost everyone knows the difficulty of getting adhesive labels to stick securely to the crinkly-paint front panel of a Mark 5 disk module – they have a tendency to fall off after a while. After consulting with several adhesive companies, it became clear that we can get a label with a really *permanent* adhesive on that surface that would work well for permanent VSN labels, but no company was able to suggest an adhesive for experiment labels that need to stick well but can be removed as well.

Recently, a new approach has been suggested for experiment labels – *use string-attached shipping labels to which standard experiment labels are adhered*. (When this was suggested to Steve Parsley, he noted that he had already seen this method in action on some modules at JIVE, but we don't know who to credit; at Haystack, suggestions by Dan Smythe and Peter Bolis led us to this approach). In any case, this appears to be a good possible solution and we suggest its adoption on a trial basis. The rules are:

1. One experiment label per tag.
2. Tag shall be of a sturdy, preferable tear-proof, material
3. Tag shall be attached with sturdy *string* (not metal wire)

4. When a module is shipped to a station, its status should be clearly stated on a tag ("Unerased", "Erased", etc.); existing experiment tags should be removed.
5. At a station, the current module status should always be indicated on a tag ('Unerased', 'Erased and Tested', 'Bad')

Shipping tags are readily available locally everywhere; the size of the tags is not critical and can be chosen to match the size of the locally produced adhesive experiment labels.

As for VSN labels, we are experimenting with a permanent-adhesive backing label upon which a VSN label would be placed.

## **2. Importance of system pre-checks (particularly at 1 Gbps)**

To help stations prepare for reliable 1024 Mb/s Mark5 recordings, I have prepared some test procedures which are available at <ftp://web.haystack.edu/pub/mark5/SNAP>. Procedures `chk1024.snp` and `chk1024.prc` are for systems with a Mark 4 formatter. The files for VLBA4 systems are `vchk1024.snp` and `vchk1024.prc`.

The `.snp` file records three 4-minute scans

512 Mb/s with `mode=mark4:32`

512 Mb/s with `mode=mark4:64`

1024 Mb/s with `mode=mark4:64`

with a lot of extra procedures to verify that the recordings are good and to help diagnose problems if any of the scans are bad.

The VLBA4 versions of the files are untested, since I do not have a VLBA4 rack. I have tested the Mark 4 version of the files. The only difference between the Mark4 version and the VLBA4 version should be that the `vc . . .` procedures have been replaced with `bbc . . .` procedures.

The files have been updated to work only with Version 9.7 of the Field System, and will not work with Version 9.6.

If you have difficulty interpreting the log messages produced by these files, or if you have any questions about the results, please e-mail me a copy of the log file, or put the log on CDDISA, with an e-mail telling me where to find it. Also please let me know if you have any other problems with these files or procedures.

For more information on testing at 1Gbps, see Mark 5 Newsletter #4 article "Notes on 1 Gbps Recording", available at <http://web.haystack.mit.edu/mark5/newsletter.html>.

Dan Smythe [DSmythe@haystack.mit.edu](mailto:DSmythe@haystack.mit.edu)

## **3. Suggested list of maintenance spares for Mark 5**

Some owners of Mark5 Units have asked what spare parts they should keep on the shelf. In our experience, the parts that fail most often are the system disks and the power supplies. Since the Mark 5 unit shares most of its components with a standard desktop personal computer, you can use your own experience with PCs to help you decide what components are most likely to fail.

It is advisable to create a clone of the system disk for backup in case emergency. A separate document available at <http://web.haystack.mit.edu/mark5/operations.html> describes a procedure for making an exact copy of your system disk.

Dan Smythe [DSmythe@haystack.mit.edu](mailto:DSmythe@haystack.mit.edu)

#### 4. Recovering data from overwritten or abnormally terminated scans (updated 3 Jan 05)

A new command has been added to *Mark5A* Version 2.7 to aid in the recovery of overwritten or abnormally terminated scans. Instances have been reported of the accidental execution of the utilities ‘*sstest*’ or ‘*WRSpeed Test*’, both of which will overwrite any existing data near the beginning of a disk module; in both cases, access to scans beyond the overwritten section is prevented. Also, occasionally a scan is terminated abnormally during recording (due to a power failure, or a keyswitch being accidentally turned to the ‘off’ position, for example): in this case, all of the data of scan in progress are normally lost. Conduant has written special recovery software, now integrated into the most recent version of *Mark5A*, that allows recovery from both of these ‘failure’ modes.

The new ‘*recover=<recovery mode>*’ command allows recovery of most or all of the otherwise lost data. A ‘*recover=1*’ command may be used in the case of overwritten data (accidentally using ‘*sstest*’ or ‘*WRSpeedTest*’) and will restore the module to readability, recovering all data but the few seconds overwritten by the inadvertent test. A ‘*recover=0*’ command may be used to recover data in which a scan terminated prematurely; all data up to the point of the abnormal termination will be recovered. Details are available in the Mark 5A command set memo (Rev. 2.7) at <http://web.haystack.mit.edu/mark5/software.html>.

#### 5. Mark 5 Disk-FIFO mode now available

A new ‘disk-FIFO’ mode augments ‘*in2net*’ to use a Mark 5 disk pack as the FIFO buffer in the case where network transfer during an experiment is desired, but network transfer rates are slower than real-time. The disk-FIFO mode operates up to a maximum of 512 Mbps with an 8-disk module.

##### *Usage*

Use disk FIFO mode on a transmitting Mark 5, along with ‘*net2disk*’, ‘*net2out*’ or ‘*net2file*’ on a receiving Mark 5, to transfer data to remote machines during an experiment, when network transfer speeds are slow compared to the real-time data rate from the telescope. In disk-FIFO mode, the disk module is used only to augment FIFO buffer storage and does not result in usable recorded data at the end of the experiment (though this might change in the future).

##### *Setup*

Disk-FIFO mode requires a scratch disk module in Bank A. Before starting, the disk module should be erased (by *SSErase* or *Mark5A*) before restarting *Mark5A* in disk-FIFO mode. To start *Mark5A* in disk-FIFO mode:

```
Mark5A -m 0 -d 1 &
```

The ‘*-m 0*’ and ‘*&*’ are optional, as usual. The ‘*-d 1*’ initiates the special disk-FIFO mode of operation. When *Mark5A* is running in this mode, a ‘*status?*’ query will return 0x20 – ‘Disk FIFO mode’, and many of the normal *Mark5A* commands and queries will return errors or will not function properly. The ‘*in2net*’ function will use the scratch disk module as a FIFO.

Details of operation of the disk FIFO mode are documented in the Mark 5A command set memo (Rev. 2.7) at <http://web.haystack.mit.edu/mark5/software.html>.

#### 6. Mark 5B status update

The PCB for the Mark 5B interface board is now out for prototype fabrication. The design of the PCB proved to be more challenging than we had expected, primarily due to rigid signal

constraints imposed by the Virtex 2 Pro Xilinx FPGA. In order to meet these signal constraints, several hundred terminating resistors had to be added to the already crowded area around the 896-pin ball-grid-array Xilinx chip. In order to meet board thickness constraints of 0.062 inches (1.58 mm) and maintain proper impedance control, the designers were forced to the design of a 12-layer board with 4-mil (0.1 mm) tracewidths. The delays in the PCB board design indicate that the first assembled prototype boards will be available for test sometime in January 2005.

## 7. New Mark 5 memos available

Several new memos have been recently added to the Mark 5 memo series at <ftp://web.haystack.edu/pub/mark5/index.html>, including Mark 5B design specifications and digital backend design and analysis memos. Take a look at them if you are interested.

## 8. 512 Mbps real-time e-VLBI demo at SuperComputing 2004

e-VLBI played a prominent role at this year's Super Computer Conference! This year's meeting, dubbed 'SC2004' and held in Pittsburgh 7-12 Nov 2004, hosted live real-time demos for several hours each day during the week long show. E-VLBI shared exhibit space with the [DRAGON project](#), with which MIT Haystack Observatory is collaborating, and set up an exhibit where live results of real-time e-VLBI from the Haystack correlator were displayed. The demo was done in two modes for several hours during each day of the show:

1. Live real-time data was piped from the Westford (near Haystack in Massachusetts) and NASA/GSFC (Maryland) antennas at 512 Mbps to the Haystack Mark 4 VLBI correlator for several hours on each day of the show. The live correlation results were displayed in a 3-dimensional plot (see Figure 1 below) in Pittsburgh as the data were correlated, so the correlation signal could be seen building up and the noise declining as the integration period increased.
2. During periods when the antennas were not available, pre-recorded data were transferred to Haystack from Westford, GGAO, Onsala (Sweden) and Kashima (Japan), followed by immediate correlation, again showing results in Pittsburgh as the correlation proceeded. The transfer rates from all stations were several hundred Mbps.

Many visitors to the booth were interested in seeing a 'real' science application of high-speed networking and learning about VLBI and e-VLBI. More information and photos are posted at <http://evlbi.haystack.mit.edu/sc2004.html>.

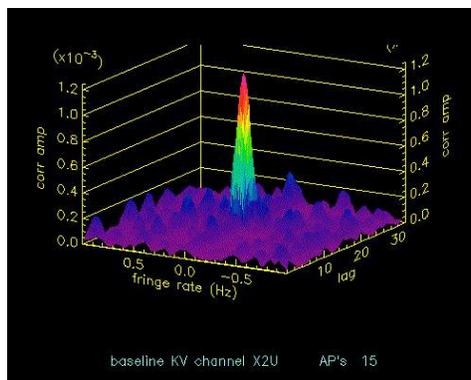


Figure 1: Real-time fringe display at SC2004 from Haystack correlator at SC2004 in Pittsburgh

## 9. Production e-VLBI Facility set up at Haystack

Haystack Observatory has recently established a ‘Production e-VLBI Facility’ which is dedicated to non-real-time e-VLBI transfers. Typically, about three 24-hr data sets per month from Kashima or Tsukuba are now transferred through this facility, with a typical volume of data around 300 GB/session; we expect usage to ramp up in the near future.

The equipment dedicated to the Production e-VLBI Facility’ includes:

- Two high speed servers for the transfer and conversion of data
  - Turtle (1.266 GHz Intel Pentium III Dell PowerEdge 2500)
  - Enterprise (a dual 2.4 GHz Xeon Dell PowerEdge 2600)
- Two 1.0 TB Lacie Bigger Extreme Firewire 800 external hard drives for the temporary storage of data
- A Mark5 for the transfer of data from system disc to Mark5 disc pack.
- 3 Dell PowerConnect 5224 Managed Ethernet Switches

After data are collected at Kashima and Tsukuba on K5 systems, a set of highly automated procedures are followed to transfer, verify, convert to Mark 5 format, and transfer the data to Mark 5 disk modules. Figure 2 summarizes the steps involved in the process; e-VLBI memo #51, available at <ftp://web.haystack.edu/pub/e-vlbi/memoindex.html>, describes the process and XML control files in detail. The process has been structured in a very modular manner to easily accommodate transfers between both heterogeneous and homogeneous data systems. Currently, the system uses ‘bbftp’ for data transfer, but in the near future this will be converted to VSI-E.

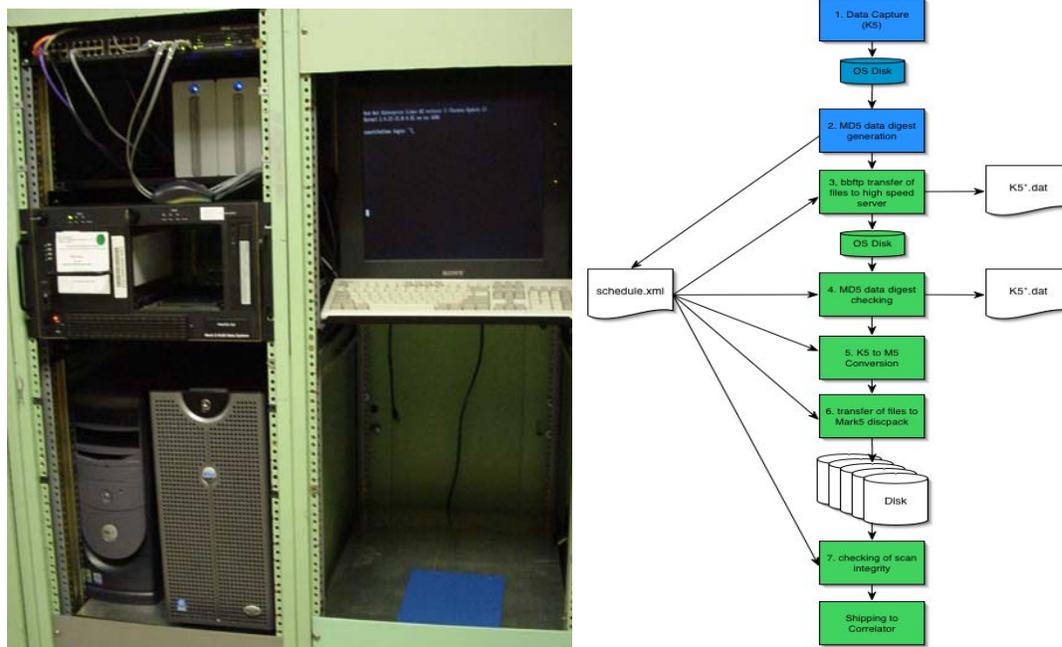


Figure 2: e-VLBI Production Facility (left); K5 to Mark 5A e-VLBI Production Process (right)

## 10. VSI-E development progress

Development on VSI-E continues. Currently the focus is on providing translation modules that will convert from Mark4/5 data format to VSI-E data format. Plans are also being made for doing the equivalent for K5 to VSI-E conversion. This will provide for the interoperable transfer of data between different end systems (e.g. Mark5 and K5) as well as providing users with the flexibility to transport data as individual channels or combinations of channels. Work towards

integrating VSI-E with Mark5 and EGAE (see item below) is well underway and should be completed soon. Once completed, this will combine the advantages of a simple user interface to e-VLBI with interoperable and flexible data transport. A reference VSI-E implementation is currently in the final stages of testing.

The latest VSI-E draft specification (Rev 2.7) and a brief description of the VSI-E transport library are available at <http://web.haystack.edu/vsi/index.html>.

## **11. Experiment-Guided Adaptive Endpoint (EGAE) progress**

We are pleased to announce that version 1.0 of the Experiment Guided Adaptive Endpoint (EGAE) is available. The EGAE is software that has been designed to greatly simplify the task of large scale e-VLBI transfers. It provides a simple yet powerful interface for operators and is able to control the transfer of data between Mark5 computers and regular linux servers. The EGAE software was used to automate the e-VLBI transfers between during the non-real-time e-VLBI demonstration for Supercomputing 2004. It has also been used to support production e-VLBI transfers. The EGAE is currently being used to support transfers between Japan and Haystack. The behavior of the EGAE (files to transfer and convert) is controlled by an XML profile. The EGAE is able to automatically transfer scans between regular linux servers and Mark5's, between Mark5's and from Mark5's to regular linux servers. It is also able to convert data between different formats (e.g. K5 to Mark5) and verify the integrity of the data at each stage of transfer using md5 checksums and Mark5 scan\_checks. The EGAE has been implemented using advanced Object-oriented techniques using the Python language. This has helped to simplify the code, enhance readability, modularity and re-usability.

## **12. e-VLBI file-naming conventions**

As e-VLBI continues to develop, it is essential that procedures be adopted to ensure reliable identification of e-VLBI raw data files. One important aspect of these procedures is to adopt standardized file-naming procedures. Agreement has been reached between representatives of NICT, JIVE and Haystack Observatory on this important issue and is proposed for general adoption.

The goals of the file-naming convention are to:

- Identify the experiment, station and scan name.
- Identify the file format (VSI, K5, Mark 5A, Mark 5B, PC-EVN, etc)

The agreed file-naming format for a file containing data from a single scan is

`<exp name>_<station code>_<scan name>[_<data start time>_<aux info1>_<aux info2>...].<file type>`

where

`<exp name>` - experiment name; max 6 chars (consistent with current limit)

`<station code>` - standard 2-character station code

`<scan name>` - assigned scan name (derived from VEX file or other source); max 16 chars

`<data start time>` - (optional) start time of data in file; required if data start time is not unambiguously embedded in the data itself.

`<aux info>` - (optional) auxiliary information field(s)

`<file type>` - identifies high-level data format within file (for example: 'vsi', 'm5a', 'evn', etc. for VSI, Mark5A and PC-EVN formats, respectively)

Details of the new file-naming conventions are available at [ftp://web.haystack.mit.edu/pub/e-vlbi/filenaming\\_conventions.pdf](ftp://web.haystack.mit.edu/pub/e-vlbi/filenaming_conventions.pdf)

### **13. 10 Gbps link established between Westford and Haystack**

An experimental 10 Gbps Ethernet link has been established between the Westford and Haystack Observatories using equipment on loan from [Raptor Networks Technology](#). The Ether-Raptor ER-1010 (one each installed at Westford and Haystack) provides up to six 10GigE ports and 24 GigE ports, which will allow us to easily ramp up our e-VLBI data transfer rates from Westford to Haystack in the near future. We have already used the 10GigE link to transfer Mark 5A data between the two sites at 1024 Mbps; this was done by configuring the Mark 5A at each end with a high-end motherboard that supports ‘bonded’ dual-GigE links. Using this new capability we plan in the near future to demonstrate real-time correlation at the full Mark 5A data rate of 1024Mbps.