PROCEEDINGS OF SCIENCE

iNetTest a 10 Gigabit Ethernet Test Unit

Richard Hughes-Jones[1]
DANTE
City House, 126-130 Hills Road, Cambridge CB2 1PQ, UK
And
Visiting Fellow, School of Physics and Astronomy, The University of Manchester,
Oxford Rd, Manchester UK
E-mail: `Richard.Hughes-Jones@dante.net`

Jonathan Hargreaves[2]
JIVE
Oude Hoogeveensedijk 4, 7991 PD, Dwingeloo, The Netherlands
E-mail: `hargreaves@jive.nl`

As part of the work of the FABRIC/JRA1 activity of EXPReS, a design called iNetTest was produced to allow the iBoB Field Programmable Gate Array (FPGA) hardware to perform as a two port 10 Gigabit Ethernet network test device. The requirements of the test unit are presented together with discussion of the design and its implementation for the FPGA, the control software and the human interfaces. The tests carried out to verify the performance of the iNetTest unit are presented together with some of the tests performed on a 4 Gigabit international link.

Science and Technology of Long Baseline Real-Time Interferometry:
The 8th International e-VLBI Workshop - EXPReS09
Madrid, Spain
June 22-26, 2009

[1] Speaker

[2] Previously at: Jodrell Bank Centre for Astrophysics, The University of Manchester, Manchester, UK

1.Introduction

The FABRIC/JRA1 work package of EXPReS investigated moving VLBI data at 4 Gigabit/s from the telescope at Onsala to the new WIDAR correlator at Jodrell Bank. It was realised that there was a need for a simple device that would be able to test network components and the performance of the network path. As a first step in creating the firmware designs for the units that would move the VLBI data, a design called iNetTest was produced to allow the iBoB [1] hardware to perform as a two port 10 Gigabit Ethernet network test device. The two 10 Gigabit Ethernet ports are independent and full duplex, thus four independent test flows are available. As well as providing a network test device, the aims were to explore the capabilities and performance of the iBoB hardware and libraries [2], produce and test the firmware building blocks required for VLBI data moving, and create a flexible software architecture to allow control and monitoring of the iBoB systems.

Section 2 considers the requirements and discusses the design and its implementation, while Section 3 presents the design of the control software and human interfaces. Section 4 describes some of the tests carried out to verify the performance of the iNetTest unit and Section 5 presents some of the tests performed on the 4 Gigabit link.

2.Requirements and Design of the iNetTest Unit

Earlier e-VLBI work [3], confirmed by the studies in ESLEA [4]and EXPReS [5], indicate that it is best to use the UDP/IP network protocol for moving real-time VLBI data. Also UDP/IP has been shown to be most useful in characterising the performance of end hosts and network components [6]. Thus the iNetTest hardware was designed to transmit streams of UDP/IP packets at regular, carefully controlled intervals and the throughput and packet arrival times were measured at the iNetTest receiver. Figure 1 shows the network view of the stream of UDP packets.

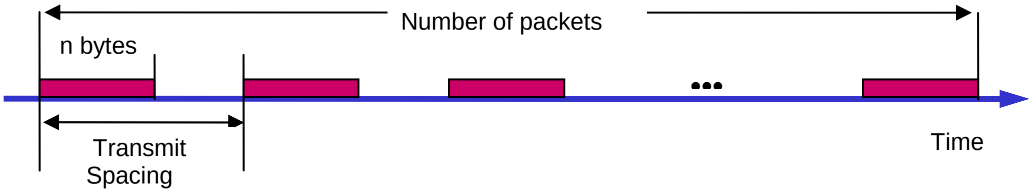


Figure 1 *The network view of the spaced UDP frames that are transmitted from the source iNetTest to the destination iNetTest.*

Within the UDP data section of the packet, an application transport header was constructed by the firmware consisting of a 64 bit packet sequence number with bit 63 set to indicate that this packet was a test packet and not VLBI data. There was no other application data header. The use of an application transport header follows the current discussion of the VLBI Data Interchange Format (VDIF) as proposed by the VDIF Task Force [7].

The packet length, spacing and number of packets to send was specified to the transmitting iNetTest unit via the control software, but the data transmission was performed in hardware and was thus deterministic. Besides counting the received packets, the receiving iNetTest unit also checked that the 64 bit packet sequence number monotonically increased thus checking for lost packets. For each incoming packet stream the FPGA histogramed the difference between the successive packet arrival times. The hardware also logged the transmission and arrival times of a snapshot of 2048 packets on each channel along with the application header identifying the packet. This allowed investigation of the one-way network transit delay of the packets.

2.1The FPGA Design

The iBoB is constructed around a Xilinx Virtex II Pro Field Programmable Gate Array (FPGA) clocked at 200 MHz. As well as an array of reconfigurable logic, the ‘fabric’ of the FPGA, the chip contains two PowerPC processors. Two sets of four RocketIO transceivers are routed to the CX4 10 Gigabit Ethernet connectors.

Figure 2 shows the Simulink design of the iNetTest design. The 10 Gigabit Ethernet ports are implemented using a core from the Berkeley libraries [2] which includes the PHY, MAC and a XAUI interface built from the RocketIO transceivers. The Berkeley block also includes a state machine to add and remove the IP and UDP headers and logic to separate software packets (those to/from the PowerPC) from hardware packets (those destined for the fabric of the FPGA). Registers mapped onto the PowerPC’s OPB bus allow software control of the local MAC, IP and Gateway address, port number and ARP table.

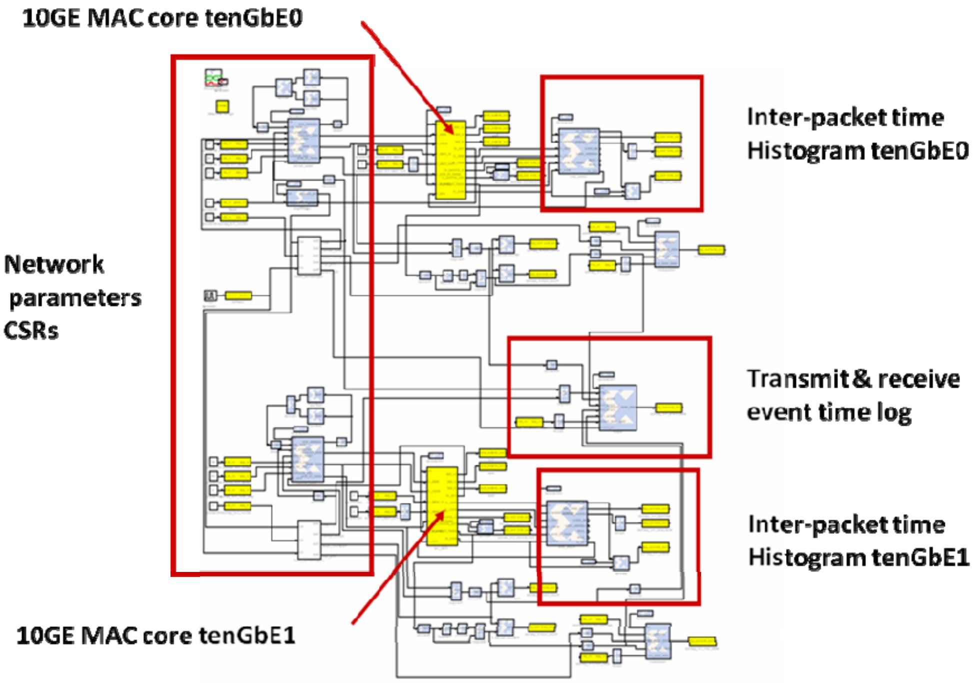

10GE MAC core tenGbE0
Inter-packet time Histogram tenGbE0
Network parameters CSRs
Transmit & receive event time log
Inter-packet time Histogram tenGbE1
10GE MAC core tenGbE1

***Figure 2**. The Simulink Design of the iNetTest Unit*

Outside the Berkeley core, each 10 Gigabit Ethernet port has its own set of control and status registers (CSRs) that allow setting of the parameters required for sending packets to the destination: the destination IP address, the UDP port number, the packet length, the inter packet spacing, and the number of packets. Other CSRs allow control of the packet flows and monitoring of the data packets received. The histograms and packet time log are created in hardware during a flow, and stored in on-chip dual port memory which the embedded PowerPC can read back at the end of the flow.

The network test flows use hardware packets which are generated and received by state machines implemented in the fabric of the FPGA. The flows can be run in two modes: one shot and continuous. In one shot the flow stops after transmission of the number of packets set in a 32 bit register, so the flow can be up to 2^{32}-1 packets long. Continuous mode simulates the flow of astronomical data: transmission continues as long as a control bit remains high. Another control bit allows a flow in either mode to be cancelled at any time.

Jumbo frames are supported by the Berkeley MAC, but the iNetTest firmware imposes a maximum packet length of 8192 bytes. The data source is a repeating pattern stored in on-chip memory. After adding the VDIF and transport headers the frame is clocked across a 64 bit wide FIFO into the Berkeley 10 Gigabit Port block.

When the hardware assembles a packet for a given IP address it uses an ARP table in hardware to determine the correct MAC address and then clocks the packet via the PHY core into the XAUI for transmission on the CX4 link. If the destination IP address is not on the local network, then the hardware selects the designated router MAC address. Ethernet packets that arrive on the 10 Gigabit links that do not have the UDP port number of the hardware fabric are sent to the PowerPC CPU. The PowerPC can also create packets to be sent on the 10 Gigabit Ethernet ports, but these have lower priority than the hardware generated packets.

3.The Control Software and Human Interfaces

The iNetTest software design was divided into several areas: the software for the embedded PowerPC CPU, and the control software external to the iBoB which is formed from core code and three different human interfaces.

3.1Software for the embedded PowerPC CPU

The software that runs in the embedded PowerPC was written in C and provides read and write access, where appropriate, to four types of components:

- One word transfers to or from registers defined in the FPGA.
- Bulk transfers to or from memory blocks defined in the FPGA firmware.
- Access to the VHDL network modules.
- Control of software actions in the PowerPC CPU e.g. initiating an ICMP echo request (ping) or generating an ARP request sequence.

The embedded PowerPC permits control of the above operations via a simple command line interface (CLI) available on the serial port, or from remote systems using UDP/IP packets over the iBoB's 10/100 Ethernet port. UDP/IP was used as there is insufficient memory available to allow use of a TCP stack.The embedded PowerPC is also responsible for initialising all the Ethernet ports with their addresses and initiating a set of ARP requests to create the ARP hardware table. It is assumed that a class C IP network exists on each Ethernet port, but an IP packet for a non-local address will be sent to a specified gateway address.

3.2Control Software

The architecture of the control software is shown in Figure 3. The design separates the human interface from the core code that interacts with the software in the PowerPC embedded in the FPGA. As indicated in Figure 3 three styles of human interface were provided:

- A simple "terminal" style menu system.
- A CLI allowing the iNetTest to be controlled from a perl script for making a series of measurements.
- A Graphical User Interface

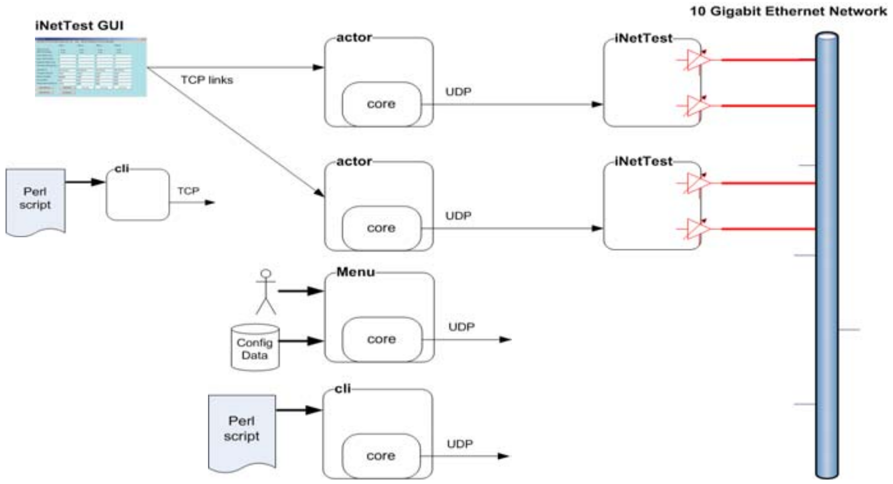


***Figure 3** The architecture of the control software*

The core code shares knowledge of the iNetTest resources or cores with the software in the PowerPC, and they use symbolic references to denote the FPGA cores due to the way the Xilinx tools allocate physical addresses. The PowerPC maps the symbolic reference onto the current address for that FPGA core. Access to a resource is performed by sending a UDP packet containing the request and waiting for the response packet. If no packet is received within a certain time, the core software re-sends the request.

The core software is also aware of the sequence of commands that are required to perform a certain action with a given FPGA resource, and translates the request from the human interface into the required sequence of requests. For example to start a data flow, several CSR accesses are required in a specific order to clear counters, initialise histograms, set the parameters of the flow, and then start the UDP/IP flow.

The “terminal” style menu and the CLI are stand alone programs which are usually run on a host local to the iNetTest unit.

The graphical user interface is designed to be used at a distance from the iNetTest units. It uses a TCP/IP link to connect to an actor on a host local to the iNetTest unit. It is written in C# and uses a series of “tab” windows to provide control of: setting the network locations of the actors and the iNetTest units, requesting and displaying the 10 GE network parameters, showing the transmitter and receiver statistics, managing ARP on the iNetTest units, sending pings, control of the UDP flows, setting and displaying histograms. Figure 4 provides an impression of these displays and Figure 5 shows the window that allows control of the UDP flows.

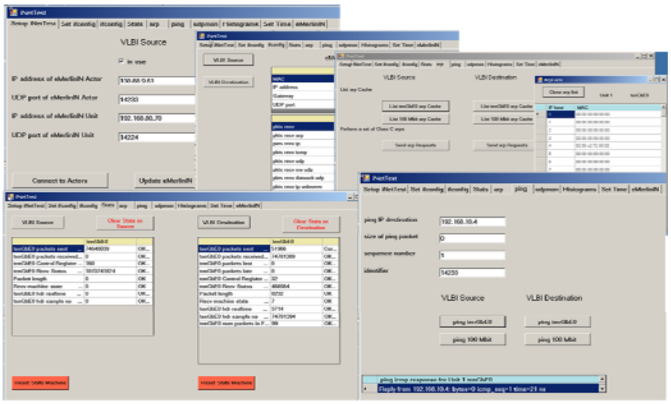
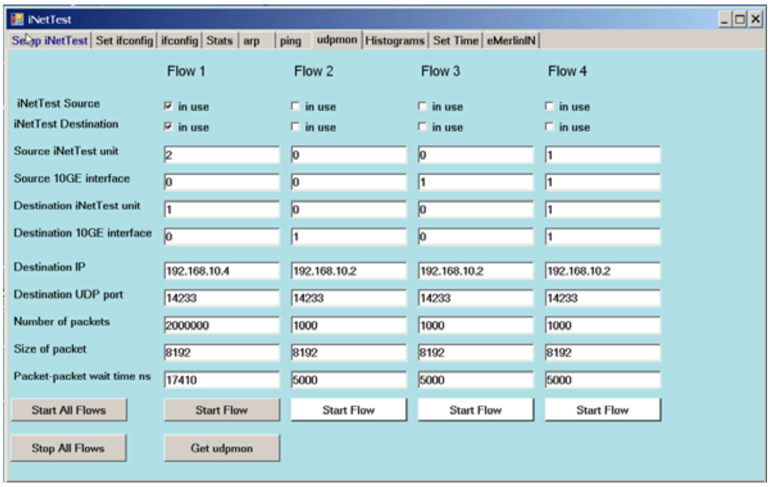


Figure 4 *The tab windows clockwise from the top for: defining the actors, showing the interface parameters, managing ARP, sending a ping and displaying statistics.*



	Flow 1	Flow 2	Flow 3	Flow 4
iNetTest Source	☑ in use	□ in use	□ in use	□ in use
iNetTest Destination	☑ in use	□ in use	□ in use	□ in use
Source iNetTest unit	2	0	0	1
Source 10GE interface	0	0	1	1
Destination iNetTest unit	1	0	0	1
Destination 10GE interface	0	1	0	1
Destination IP	192.168.10.4	192.168.10.2	192.168.10.2	192.168.10.2
Destination UDP port	14233	14233	14233	14233
Number of packets	2000000	1000	1000	1000
Size of packet	8192	8192	8192	8192
Packet-packet wait time ns	17410	5000	5000	5000

Figure 5 *View of the window that allows control of the UDP flows.*

4.Performance of the iNetTest Unit

4.1iNetTest to iNetTest

Figure 6 shows the distribution of the time between the arrivals of successive 8192 byte UDP packets when 10 M packets were transmitted between two iNetTest units in the lab. The graph is plotted in 5 ns bins and has a full width half maximum (FWHM) of 15ns with no outlying points or tails. This confirms the deterministic behaviour of the FPGA design.

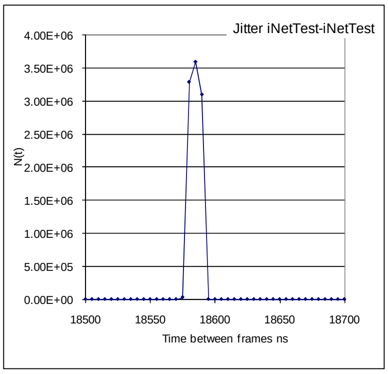


Figure 6 The distribution of the time between the arrivals of successive 8192 byte UDP packets transmitted between two iNetTest units in the lab plotted in 5 ns bins.

4.2PC to iNetTest

The iNetTest units have been used to investigate the distribution of the time between successive packets transmitted between two high-performance PCs running Linux with the 2.6.20-web100_pktd-plus kernel. The PCs were built using the Supermicro X7DBE motherboard, two Dual Core Intel Xeon Woodcrest 5130 2GHz CPUs, 4 G Bytes of memory, and the PCI-Express 10G-PCIE-8A-R 10 Gigabit Ethernet NIC from Myricom.

The left hand plot in Figure 7 shows the histogram distribution of the time between successive packets transmitted between two PCs. The requested packet separation was 100 µs, which corresponds to the main peak in the histogram. However about 1% of the packets are delayed or advanced by approx. 34 µs as indicated by the side lobes. The right hand plot shows that these peaks are absent when packets are transmitted from the iNetTest unit to the PC suggesting that they are an artefact of the how the PC operating system works.

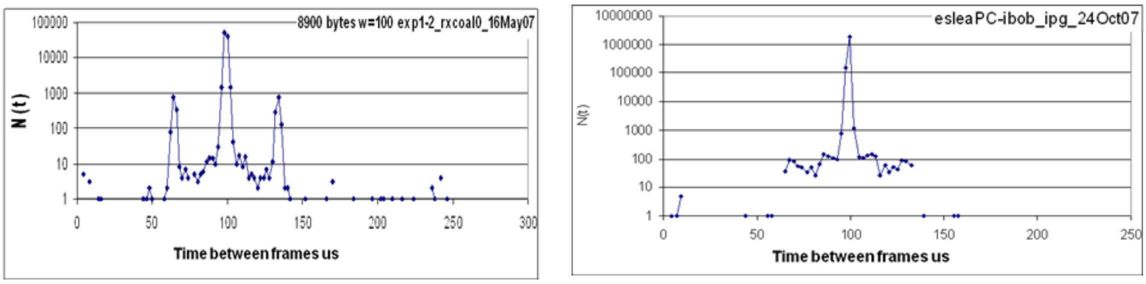


Figure 7 Distributions of the time between the arrivals of successive packets.
Left plot: PC to PC Right plot: PC to iNetTest

4.3iNetTest Throughput on the 4 Gigabit link

Throughput and packet loss and packet jitter tests were performed on the multi-domain 4 Gigabit end-to-end path from Onsala to Jodrell Bank [8] using iNetTest units at either end of the path. The network path crosses multiple administrative boundaries, and on different portions of the path it uses equipment from different manufacturers. Some sections of the path used 10 Gigabit Ethernet technology, while others used SDH. The SDH sections of the path were provisioned as 28 VC-4, giving a transfer rate of 4.193 Gbit/s. This was sufficient to carry the VLBI data rate of 4.096 Gbit/s as well as the protocol and framing overheads.

Figure 8 shows the received "wire" rate UDP throughput and packet loss as a function of the transmitted packet spacing for UDP packets sent from Onsala to Jodrell Bank for various packet sizes as measured by the iNetTest units. The flat portions of the curves indicate that the maximum throughput was 4.179 Gbit/s, in agreement with that available for moving data over the 28 VC-4 SDH link after allowing for the GFP overheads. The packet loss for spacing less than ~17 µs is expected as more packets are being sent than can be carried by the 28 VC-4s. Note there was no packet loss for 8192 byte packets at 16 µs spacing.

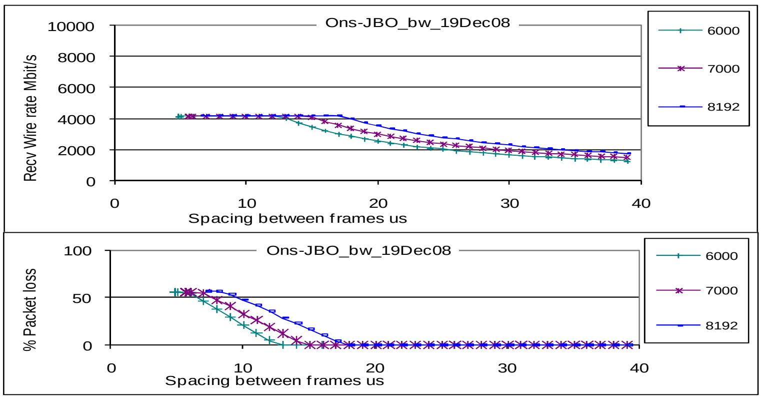


Figure 8 UDP throughput and packet loss, measured by the iNetTest units, as a function of the spacing between the packets when sending packets from Onsala to Jodrell Bank.

4.4 Stability of the 4 Gig link

The stability of the multi-domain network path was determined by using the iNetTest units to send a set of trials of 100M 8192 Byte packets with a spacing of 16µs from Onsala to Jodrell and measuring the achievable UDP throughput, the packet loss and the inter-packet jitter. Each trial took about 27 minutes and consecutive trials were repeated immediately. For each trial the data throughput was measured as 4.094 Gbit/s, with no variation between trials. Over a typical set of about 40 trials, the loss rate was ≤ 10^{-9} (approx bit error rate better than10^{-13}).

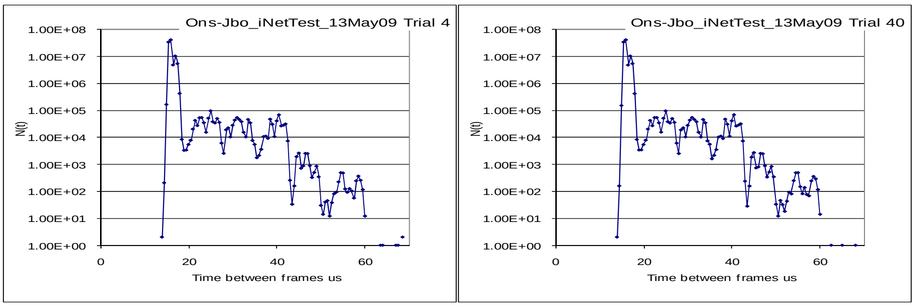


Figure 9 Distributions of the time between the arrivals of successive 8192 byte packets send from Onsala to Jodrell over the 4 Gigabit link for two different trials.

Figure 9 shows plots of the inter-packet arrival times for two of the trials described above. The main peak at 16 µs has a full width half maximum of ~1µs and there are tales extending to ~70 µs, but the tails are a factor of 10^{-3} smaller, and do not change shape. The throughput, packet loss and jitter measurements indicate that the link is extremely stable.

5.Conclusions

The iNetTest IBOBs have proved to be stable and reliable. As expected, the FPGA provides a deterministic way of generating and receiving packets for network testing. In collaboration with the National Research Networks and DANTE, they have been used extensively in testing network devices and the 4 Gbps network between Onsala and Jodrell Bank. The iNetTest units have also been used in field testing and evaluating commercial Ethernet testers from Xena [9].

As well as providing the infrastructure required for iNetTest, the approach of abstracting the firmware and partitioning the software allowed simple extensions to provide control software for the iBoBs programmed for eMerlinIN, On2jbo_tx at Onsala and On2jbo_rx at Jodrell Bank.

6.Acknowledgements

Throughout the development of the iNetTest unit in the EXPReS project we have had encouragement, close co-operation and valuable technical discussions with colleagues at JIVE, Jodrell, Onsala, network personnel from the campuses, the Metropolitan Area Networks, the National Research Networks JANET, SUNET and NORDUnet, DANTE and Internet2. We would like to thank all who have been involved, and this support has been vital in making the project an operational success.

EXPReS is funded by the European Commission (DG-INFSO), under the Sixth Framework Programme, Contract #026642.

References

[1] IBOB web page http://casper.berkeley.edu/wiki/IBOB viewed August 2009

[2] http://casper.berkeley.edu/wiki/Libraries viewed August 2009

[3] Richard Hughes-Jones, Steve Parsley , Ralph Spencer, *High Data Rate Transmission in High Resolution Radio Astronomy - vlbiGRID*, FGCS Special Issue: IGRID2002 Vol 19 (2003) No 6, August 03

[4] R.E. Spencer, P. Burgess, S. Casey, R. Hughes-Jones, S. Kershaw, A. Rushton, M. Strong, A. Szomoru and C. Greenwood, *The contribution of ESLEA to the development of e-VLBI* in “Lighting the Blue Touchpaper for UK e-Science - Closing Conference of ESLEA Project”, PoS, March 2007 (http://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=41#session-5

[5] EXPReS Protocols performance report D150.

[6] R. Hughes-Jones, P. Clarke, S. Dallison, *Performance of 1 and 10 Gigabit Ethernet Cards with Server Quality Motherboards*, Future Generation Computer Systems Special issue, 2004

[7] “VLBI Data Interchange Format (VDIF) Specification”, as proposed by the VDIF Task Force.

[8] R. Hughes-Jones, J. Hargreaves, *Commissioning and Using the 4 Gigabit Lightpath from Onsala to Jodrell Bank*, PoS (EXPReS09) 035

[9] Xena Home Page http://www.xenanetworks.com/