VIRTUAL SATELLITE INTEGRATION AND THE SPACEWIRE INTERNET TUNNEL

Session: SpaceWire Test and Verification

Long Paper

Stuart Mills

STAR-Dundee,

c/o School of Computing, University of Dundee, Dundee, DD1 4HN, Scotland, UK

Steve Parkes

University of Dundee,

School of Computing, Dundee, DD1 4HN, Scotland, UK

Raffaele Vitulli

European Space Agency,

ESTEC, Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands

E-mail: stuart@star-dundee.com, sparkes@computing.dundee.ac.uk, Raffaele.Vitulli@esa.int

ABSTRACT

A major area of risk in spacecraft development is the integration of sub-systems late in the development cycle. Sub-systems are often developed by different organisations in different countries, making it expensive to bring them all together for testing. Although the SpaceWire standard has helped reduce incompatibility problems at the data link and physical layers, there is still the potential for problems at the higher layers. These problems may only be found during integration testing, which is late in the development cycle, and may therefore be very costly to address.

This paper presents a solution to reduce that risk: virtual satellite integration and the SpaceWire Internet Tunnel. Virtual satellite integration is a process whereby the sub-systems of a satellite are brought together virtually for testing. The SpaceWire Internet Tunnel is a tool which makes it possible to virtually integrate sub-systems which are to be connected using SpaceWire.

1 INTRODUCTION

1.1 LIMITATIONS OF EXISTING METHODS OF INTEGRATION

A common feature of spacecraft development, particularly in Europe, is that the partners involved are separated by great distances. Development teams may be in different states, countries or continents, and may be separated by culture and language, in addition to their geographical separation.

To cope with these boundaries and ensure the correct operation of the system, a great emphasis is placed on using documentation during spacecraft design. Interface specifications are written for each subsystem and the accuracy of these documents is of great importance. Any errors or inaccuracies in these specifications could result in sub-systems being developed which cannot communicate without major modification.

Once the sub-systems have been developed, integration and testing is performed. This requires each unit to be moved to a common location. Any faults or errors identified at this late stage can be very costly to correct. This late integration is therefore a major area of risk in any project.

One method of reducing this risk is through the use of standard interfaces. SpaceWire [1] [2] provides a standard for the physical and data link layers of the interface, greatly reducing the likelihood of issues at this level during integration testing. However, there is still the potential for problems at the higher layers, for example in the interface between applications, each of which may be developed and tested in isolation.

1.2 VIRTUAL SATELLITE INTEGRATION

Virtual satellite integration [3] can be used to greatly reduce these risks. It provides a means of interconnecting the sub-systems at an earlier stage of the development life cycle, without moving the units to a common location. Instead the Internet is used to remotely connect the sub-systems, allowing them to be virtually integrated.

An example of a very simple spacecraft network using SpaceWire and containing two sub-systems is shown in Figure 1. These sub-systems could be developed by separate organisations in different countries.

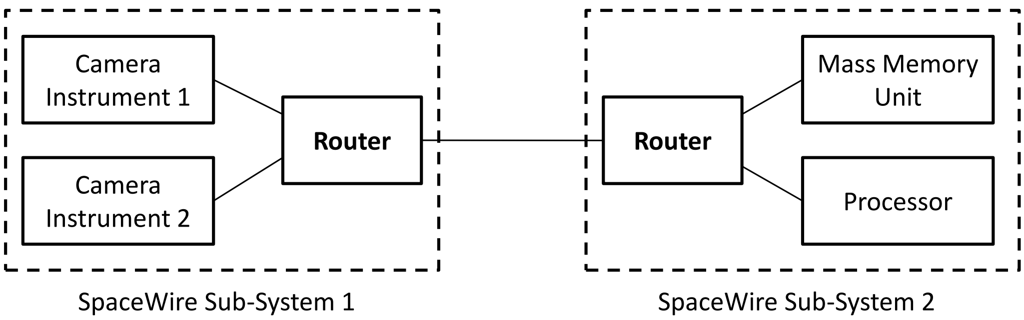

Figure 1: SpaceWire network with two distinct sub-systems

In the past these sub-systems may not be tested together until late in the development cycle. Using virtual satellite integration, these two sub-systems can be tested together at a much earlier stage, as shown in Figure 2. In this updated network diagram, the link between the two routers has been replaced with links to two SpaceWire IP-Tunnel devices each connected to a PC, with the PCs connected over the Internet. Each PC runs the SpaceWire IP Tunnel software which manages the connection between the two PCs, sending and receiving traffic on the Internet connection and the SpaceWire link.

Virtual satellite integration can also be used to test sub-systems with simulators. In Figure 2, for example, sub-system 1 could be replaced with a camera simulator in order to test sub-system 2.

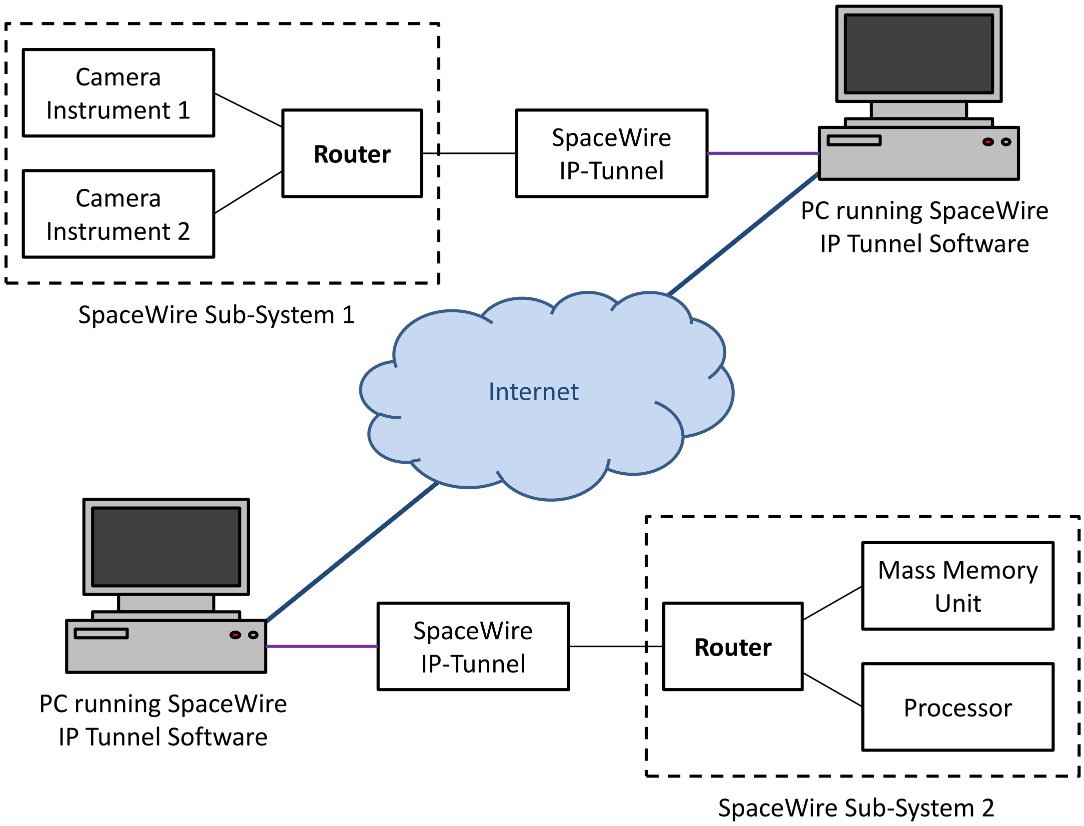

Figure 2: SpaceWire Internet Tunnel between two distinct sub-systems

There are some scenarios in which the use of virtual satellite integration is not appropriate. As a network such as the Internet is used to replace a link in the network, traffic passing across this link will be subjected to the bandwidth limitations and latency issues of this network. The SpaceWire Internet Tunnel provides features to limit the effect of these problems but despite this, virtual satellite integration may not be suitable for systems in which traffic must be transferred within a bounded time period, for example. Many systems can be used with the Tunnel, however, and indeed the slow response times of the Internet can often be useful in identifying problems in devices and applications.

2 THE SPACEWIRE INTERNET TUNNEL

The SpaceWire Internet Tunnel [4] consists of hardware to send and receive traffic from the SpaceWire link, and software to route traffic between the hardware component and the Internet. Protocol Analysis software is also provided to allow the traffic crossing the Tunnel to be monitored.

A SpaceWire Internet Tunnel transparently replaces a SpaceWire link. All traffic entering a Tunnel will exit the Tunnel in exactly the same order. This means, for example, that a time-code will exit the Tunnel between the same two data characters it entered.

2.1 SpaceWire IP-Tunnel Hardware

The hardware component of the SpaceWire Internet Tunnel requires a STAR-Dundee SpaceWire IP-Tunnel device [5]. The reason for the use of a specialised device is that the device does not simply send and receive packets. The Space IP-Tunnel device must perform a number of tasks in order to cope with the expected reduction in bandwidth and increase in latency introduced by the Internet. This is necessary to avoid timeouts at routers, for example, which occur when the time between two data characters in the same packet exceeds a defined limit.

The device must also treat traffic differently than the SpaceWire-USB Brick [5] on which it is based. In a normal SpaceWire device, time-codes should be treated with higher priority than data packets. However, a SpaceWire Internet Tunnel must not alter the order of time-codes and data characters. Similarly, link start and disconnect events must be passed up to the application, in sequence with data characters and time-codes.

2.2 SpaceWire IP Tunnel Software

The SpaceWire IP Tunnel application is written in Java, which means it can be run without recompilation on any platform on which Java is supported and a driver is available for the SpaceWire IP-Tunnel device. Currently Windows and Linux drivers are provided. This driver provides the interface between the application and the hardware and is an updated version of the SpaceWire USB Driver provided with STAR-Dundee SpaceWire-USB Bricks and SpaceWire Router-USBs [5]. Modifications to the driver were required in order to support the interface used by the Tunnel device. The updated driver also keeps data characters, time-codes and link start and disconnect events in sequence when passing them up to the user. In addition, changes were made to the driver to improve performance with the unusual traffic flow produced by the SpaceWire IP-Tunnel device, when compared with the SpaceWire-USB Brick and SpaceWire Router-USB.

When the user creates a Tunnel in the SpaceWire IP Tunnel application, a secure connection is established over the network to the other end. The secure connection ensures that no-one can view or modify the traffic crossing the network, while a password can be provided for each Tunnel to provide further authentication. Any form of network interface (typically Ethernet) can be used by the Tunnel software, provided there is a network driver available for the operating system.

Once the connection is established, traffic received on the SpaceWire link is sent over this network connection to the other end. At the same time, the software is also receiving traffic from the network connection, which it then sends over the SpaceWire link. Traffic received from both the SpaceWire link and the network connection is also passed to the SpaceWire Protocol Analyser for analysis.

2.3 SpaceWire Protocol Analyser

As the SpaceWire Internet Tunnel is intended for use during testing, it is important to be able to identify and investigate any problems encountered while performing a test. The SpaceWire Protocol Analyser is integrated in to the SpaceWire IP Tunnel software and allows the user to view the statistics of traffic crossing the Tunnel in real time, as shown in Figure 3.

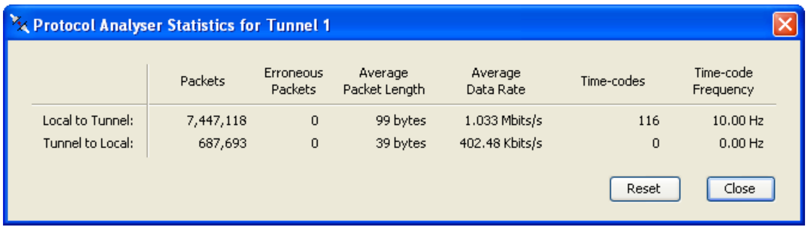

Figure 3: Protocol Analyser Statistics dialog

Traffic crossing the Tunnel can also be recorded using the Protocol Analyser. The type and characteristics of traffic to be recorded (e.g. data packets with a particular address or data pattern, or time-codes with specific flag values) can be specified, as can a trigger point and the amount of traffic to record before and after the trigger point. Once traffic is recorded, it is displayed in the Protocol Analyser for further analysis, as shown in Figure 4.

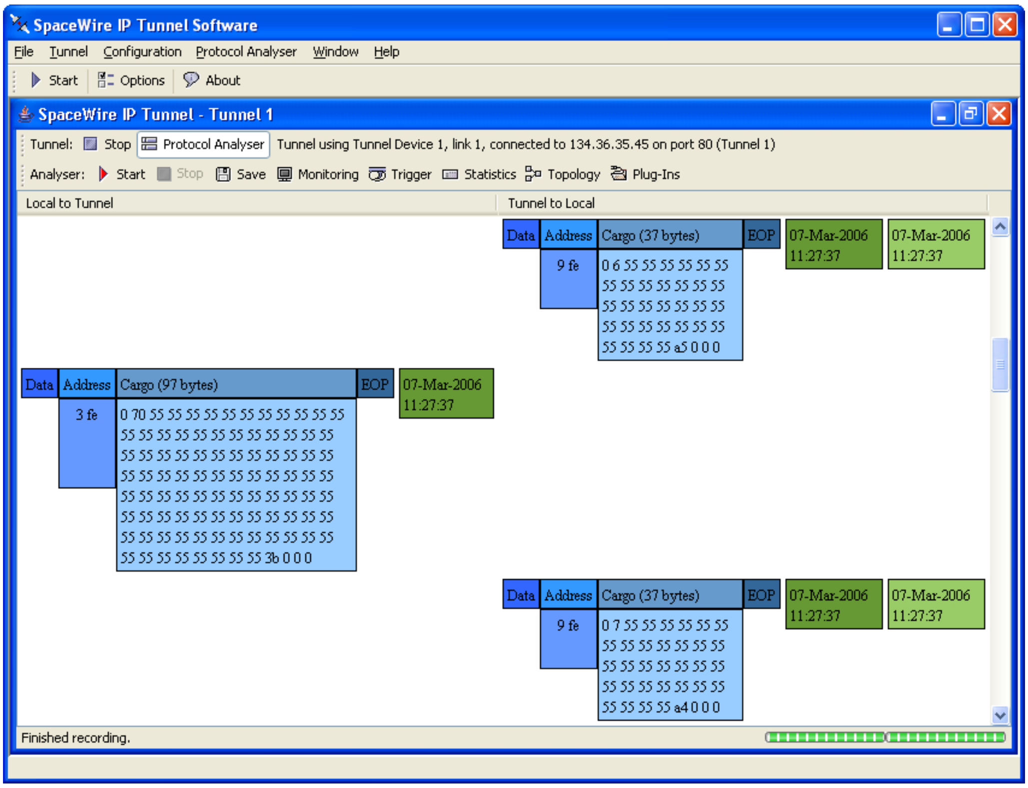

Figure 4: Protocol Analyser showing recorded traffic

The Protocol Analyser also allows higher layer protocol plug-ins to be loaded. These can be written by users to identify and display packets containing a particular higher layer protocol. With a higher layer protocol plug-in loaded, the Protocol Analyser can be used to record and/or trigger on packets containing the protocol defined by the plug-in. Recorded traffic containing the protocol is also displayed using the formatting information provided by the plug-in.

Figure 5 shows traffic recorded by the Protocol Analyser containing Remote Memory Access Protocol (RMAP) [6] packets. The RMAP plug-in (which is provided with the SpaceWire IP Tunnel and Protocol Analyser Software) has been used to format the packet, and so the fields in the RMAP packet can easily be viewed without the need to look at each of the individual data bytes to determine a field's value.

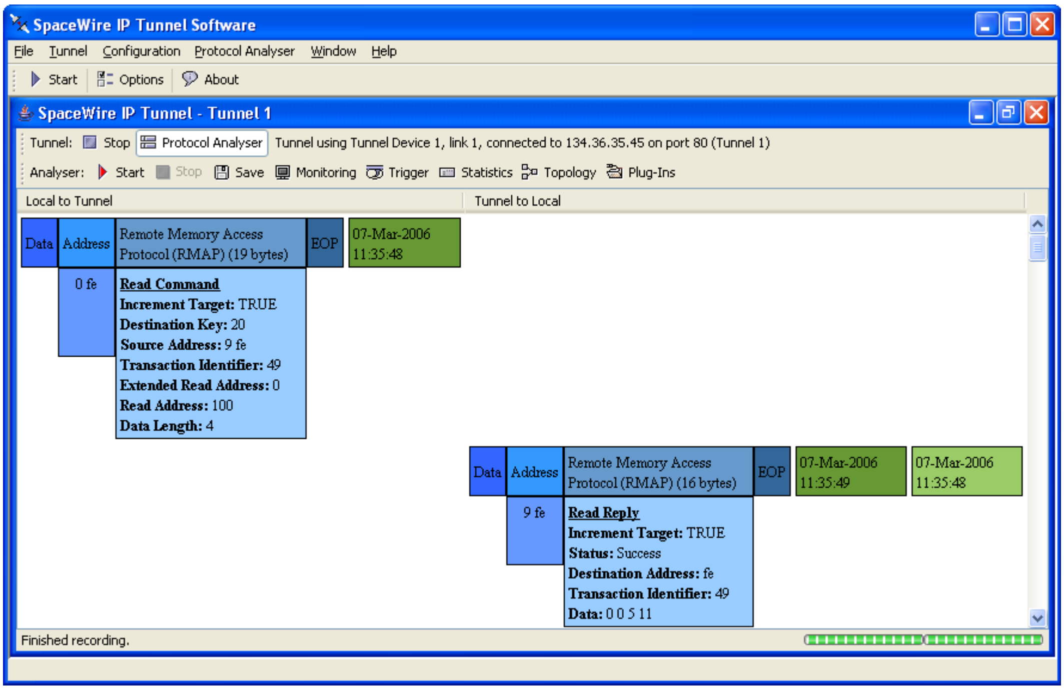

Figure 5: Protocol Analyser showing recorded traffic formatted by the RMAP plug-in

As users can write their own protocol plug-ins, it can be much easier to monitor the traffic crossing the network. For example, if an application sends packets in a particular format, someone can write a higher layer protocol plug-in describing that format, which can then be loaded by anyone using the Tunnel software. Protocol plug-ins are written in Java and can be very simple or as complex as is required. Example source code is provided to assist in development of new plug-ins.

3 The Pilot Activity

An ESA pilot study is currently being undertaken to test the suitability of the SpaceWire Internet Tunnel and virtual satellite integration. This study involves a number of organisations in different countries, each using tunnels to test concepts, hardware and software. On completion of their experiments they will report back on their experiences.

It is hoped that in addition to proving how useful virtual satellite integration can be, the pilot study will also identify any issues and possible areas of improvement within both the concept and the hardware and software in use. Although the pilot activity is still in its early stages, already some of the organisations involved have successfully tested Tunnels between their locations and a network set-up at the University of Dundee.

4 SPACEWIRE INTERNET TUNNEL SERVER

A limitation of the existing SpaceWire Internet Tunnel is that firewalls in some organisations are configured to stop the Tunnel accessing the Internet, and actions must be taken to allow a Tunnel to pass through the firewall. The reason for this problem is that one end of the Tunnel must operate as a server while the other operates as a client. The server end requires the same firewall settings as a web server, for example, which may not be possible in some organisations.

To address this problem, development has started on a SpaceWire Internet Tunnel Server. A PC acting as a Tunnel Server will run software which accepts connections from clients and establishes connections between pairs of clients. The software will then route all traffic from a client to the other client in the pair. As both ends of the Tunnel are now clients the problems with firewalls should be eliminated. The Tunnel Server must be situated outside the firewall, or given the same access permissions required by a Tunnel client acting as a server. However, a single Tunnel Server will be capable of handling a number of Tunnels, so a single server could handle Tunnels from multiple organisations.

5 SPACEWIRE INTERNET TUNNEL IMPROVEMENTS

Although the SpaceWire Internet Tunnel and Protocol Analyser software performs the task it was designed for, there are a number of improvements that could be made. After some time identifying possible enhancements, development will soon commence on improving the software. These improvements include some minor bug fixes to the Tunnel, attempts to improve performance, and the addition of new features to both the Tunnel and Protocol Analyser.

New features which are being considered include:

- A chat window for communication between the users at each end of a Tunnel
- Support for proxy servers
- Improved facilities for viewing recorded traffic
- Play back of recorded traffic
- Context sensitive help for all areas of the software

6 SUMMARY

The SpaceWire Internet Tunnel is now available to buy from STAR-Dundee. Tunnelling has been successfully demonstrated between a number of locations across Europe. The Pilot Activity will demonstrate and test the virtual satellite integration concept more comprehensively. The feedback from these experiments will drive future improvements to the SpaceWire Internet Tunnel, some of which are currently under development.

Following recent improvements, data rates greater than 25 Mbits/s have been viewed using the Tunnel. As such rates are unlikely to be achieved over the Internet and most wide area networks, the Tunnel hardware and software are unlikely to be the cause of any limits in the data rate when tunnelling. The SpaceWire Tunnel Server will make it easier for users behind firewalls to use the Tunnel, while the improvements being made to the Protocol Analyser will mean that more detailed analysis of tunnelled traffic can be performed.

A demonstration of the SpaceWire Internet Tunnel will be provided at the SpaceWire Conference.

7 Acknowledgements

The authors would like to acknowledge the support of ESA for the work done on the SpaceWire Internet Tunnel and Tunnel Server at the University of Dundee.

8 References

1. European Cooperation for Space Standardization, Standard ECSS-E-50-12A, "SpaceWire – Link, Nodes, Routers and Networks", Issue 1, European Cooperation for Space Standardization, January 2003.

2. European Space Agency, "SpaceWire Web Page", European Space Agency, http://spacewire.esa.int.

3. S. M. Parkes, "Virtual Satellite Integration", Proceedings Data Systems in Aerospace (DASIA) conference, Nice, France, May 2001.

4. S. Mills, S. M. Parkes and R. Vitulli, "SpaceWire Internet Tunnel", Proceedings Data Systems in Aerospace (DASIA) conference, Edinburgh, Scotland, May 2005.

5. STAR-Dundee, "STAR-Dundee Web Site", STAR-Dundee, http://www.star-dundee.com.

6. European Cooperation for Space Standardization, Draft Standard ECSS-E-50-11, "SpaceWire Remote Memory Access Protocol", Draft F, European Cooperation for Space Standardization, November 2006.