THE SPACEWIRE INTERNET TUNNEL AND THE ADVANTAGES IT PROVIDES FOR SPACECRAFT INTEGRATION

Session: SpaceWire Test and Verification

Long Paper

Stuart Mills, Steve Parkes

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

Raffaele Vitulli

European Space Agency, ESTEC, Keplerlaan 1, 2201 AS Noordwijk, The Netherlands

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

ABSTRACT

The SpaceWire Internet Tunnel is a tool developed by the University of Dundee and STAR-Dundee to allow the components of a spacecraft utilising SpaceWire to be integrated virtually. This concept is known as "virtual spacecraft integration" and involves connecting the components remotely using a network such as the Internet. A pilot study was conducted by ESA to investigate the benefits of such a system.

This paper describes the SpaceWire Internet Tunnel in detail, reporting some of the technical hurdles which were overcome to achieve virtual spacecraft integration. Advantages and limitations of the system, identified by both the University of Dundee and those involved in the pilot study are described, with the reasons for any limitations explained.

1 INTRODUCTION

The concept of virtual spacecraft integration has been described in previous papers by the University of Dundee [1] [2] [3]. It provides a means by which integration testing of spacecraft components can be performed without the need to bring each of the components to one physical location. The SpaceWire standard [4] aims to improve reusability, promote compatibility and reduce system integration costs. Virtual spacecraft integration has the potential to reduce system integration costs still further, by reducing travel and by identifying problems at an earlier stage of spacecraft development than is currently the case.

Virtual integration is achieved through the use of a network such as the Internet. A section of the spacecraft's onboard bus is replaced with a virtual connection over the network, allowing components to communicate with one another despite potentially being great distances apart.

2 THE SPACEWIRE INTERNET TUNNEL

The SpaceWire Internet Tunnel is a tool for performing virtual spacecraft integration in a SpaceWire network. Originally developed by the University of Dundee under ESA contract, it is now a commercial product available from STAR-Dundee [5]. An example SpaceWire network which could benefit from virtual spacecraft integration is shown in Figure 1. This network contains two separate sub-systems, which may be developed by different companies, possibly in different countries. This is quite common in European missions, for example.

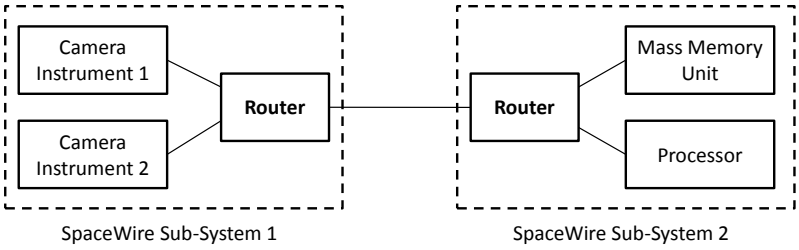

Figure 1: Example SpaceWire network containing two distinct sub-systems

A SpaceWire Internet Tunnel replaces a SpaceWire link in an onboard network, and consists of both software and hardware components. A SpaceWire cable representing one end of the link to be replaced by the SpaceWire Internet Tunnel is connected to a SpaceWire IP-Tunnel device. This device is then connected to a PC by a USB cable. Software running on the PC manages the Tunnel and allows traffic crossing the Tunnel to be monitored and recorded. A similar set-up is used at the other end of the link being replaced, and the software running on the two PCs tunnels traffic received on the SpaceWire links over a network to the other end. This arrangement is shown in Figure 2, where the two sub-systems from the example network in Figure 1 have been connected virtually using a SpaceWire Internet Tunnel. These two sub-systems may be in the same lab, or may be in different continents.

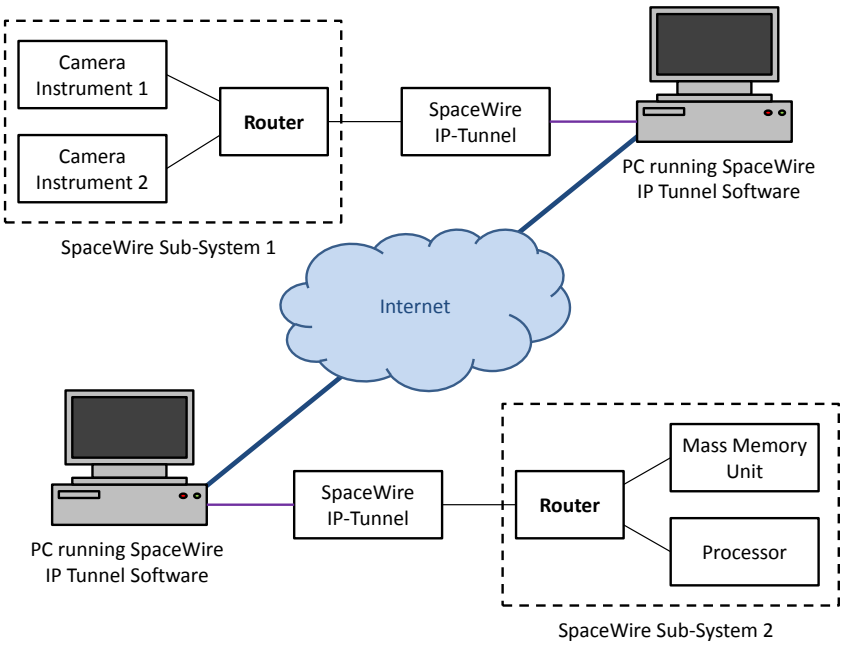

Figure 2: Example SpaceWire network containing two sub-systems integrated virtually

As well as exchanging data packets, the Tunnel also ensures that the link state is reflected at each end of the Tunnel. This means that if a link is disconnected at one end of the Tunnel, the link at the other end will also be disconnected. Other than the increased latency and reduced bandwidth, the Tunnel is almost transparent.

The hardware component of the Tunnel, shown in Figure 3, has two SpaceWire ports, allowing up to two Tunnels to be established. The software component is a Java application which can be run on Windows or Linux, as drivers for the SpaceWire-IP Tunnel have been written for these platforms. Any number of connections can be established using the Tunnel software, up to two for each connected Tunnel device.

Figure 3: The SpaceWire IP-Tunnel hardware

The SpaceWire Protocol Analyser is a module provided with the SpaceWire IP Tunnel Software which allows the traffic crossing a Tunnel to be monitored and recorded. As the SpaceWire Internet Tunnel is intended for use during integration testing, it is important that tools are provided to identify and correct any problems in the system.

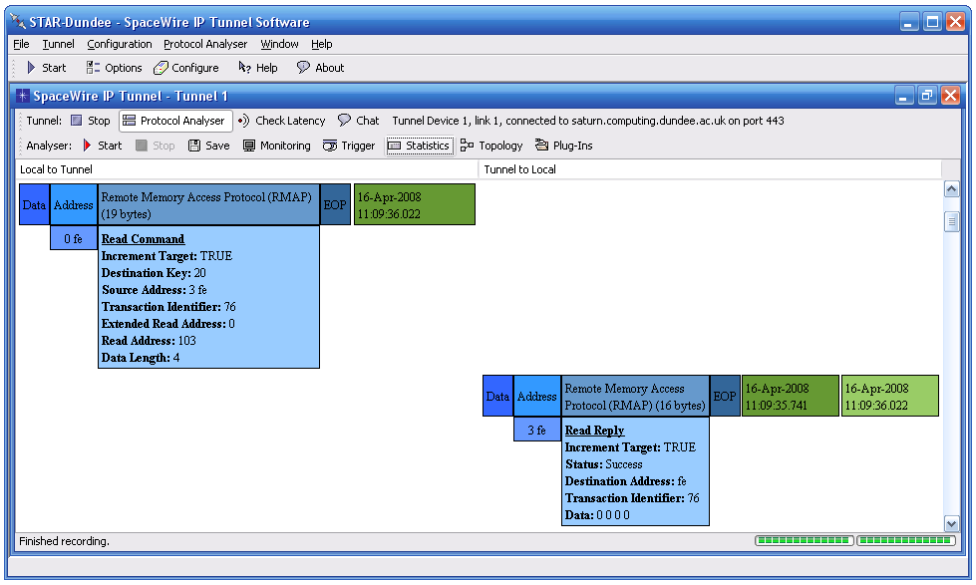

Figure 4: The main window of the SpaceWire IP Tunnel Software, showing recorded RMAP packets

The Protocol Analyser allows plug-ins for specific protocols to be loaded so that packets containing these protocols can be monitored and recorded. Higher-level protocol plug-ins can be written by users and loaded at run time, so potentially any protocols can be supported. A plug-in for the Remote Memory Access Protocol (RMAP) [6] is included with the Protocol Analyser. This allows the individual fields of recorded RMAP packets to be displayed, and also permits triggering on RMAP packets. A screenshot of the Tunnel Software showing some traffic recorded using the Protocol Analyser and formatted using the RMAP plug-in is shown in Figure 4.

3 TECHNICAL DIFFICULTIES AND OVERCOMING THEM

During development of the SpaceWire Internet Tunnel, a number of potential limitations were noted. Each of these limitations was identified as being in one of the following three areas:

- Issues relating to the bandwidth and latency restrictions of the Internet
- Security concerns related to sending data across the Internet
- Problems establishing connections over the Internet

The SpaceWire Internet Tunnel has mechanisms to address each of these areas, reducing or eliminating each limitation.

The first category of limitation is particularly problematic when tunnelling SpaceWire traffic. For example, if the time between routing two bytes of a packet is greater than the timeout period within a router, that router will terminate the packet and indicate there was a timeout error. To avoid such issues when tunnelling, the SpaceWire Internet Tunnel Software contains a number of mechanisms to greatly reduce the chances of timeouts being introduced due to the bandwidth or latency restrictions of the terrestrial network in use. These mechanisms are present at both the entrance and the exit of the Tunnel, as timeouts may also occur when a router is unable to send bytes as a link is already in use.

Despite these mechanisms there are some systems where the SpaceWire Internet Tunnel is not suitable. Such systems include those which expect packets within a bounded time period, or which have high bandwidth requirements.

To cope with the security concerns of tunnelling sensitive information over the Internet, all Tunnel traffic is sent using Transport Layer Security (TLS) [7], the successor to the Secure Sockets Layer (SSL). TLS is the protocol used by secure web pages which transmit sensitive information such as credit card details. All traffic is encrypted, and the protocol ensures the contents cannot be viewed or modified. To ensure that the Tunnel at the other end of a Tunnel connection is who they claim to be, all Tunnels can also be configured to use a password. When sent over the network, this password is encrypted using the same TLS protocol.

A number of problems were identified by users when establishing network connections between two Tunnel ends. These mainly related to firewall restrictions and proxy issues. Some organisations expect their users to connect to the Internet via a proxy server. Support was added to the Tunnel Software for proxy servers and a number of the authentication systems used by proxy servers.

Other organisations only allow Internet traffic on a limited number of TCP (Transmission Control Protocol) ports, such as those used for web pages (normally port 80). The SpaceWire Internet Tunnel allows any port to be used. In addition to avoiding problems with firewalls which only allow certain ports to be used, this also avoids problems with firewalls or proxy servers which check that the format of packets on a particular TCP port is in the correct format for that port. Using port 443, which is assigned for secure HTTP traffic being sent using TLS/SSL, the firewall or proxy server sees the encrypted Tunnel traffic and assumes the traffic to be in the correct format, as it cannot view the encrypted contents.

Another limitation when establishing network connections is the client-server nature of network connections: one end of a Tunnel acts as a server listening for connections, while the other end acts as a client and connects to that server. Most organisations are not happy for their users to run servers behind their firewalls, and block all attempts to connect to those servers through their firewall. The only exceptions to this rule are dedicated servers, such as web servers, which normally run on PCs which have been granted special firewall permissions to accept connections from outside the firewall.

Although some network administrators are happy to allow a Tunnel PC to accept Tunnel connections, some users found this was not permitted. This meant they could only establish Tunnels within their organisation, or with users in other organisations who had been granted the required firewall permissions. To address this limitation, the SpaceWire IP Tunnel Server was developed.

4 THE SPACEWIRE INTERNET TUNNEL SERVER

As mentioned in the above section, a limitation of the SpaceWire IP Tunnel Software is that one end of the Tunnel must act as a server, and therefore can require special firewall privileges. The SpaceWire Tunnel Server was developed to address this issue, allowing both ends of a Tunnel to act as a client.

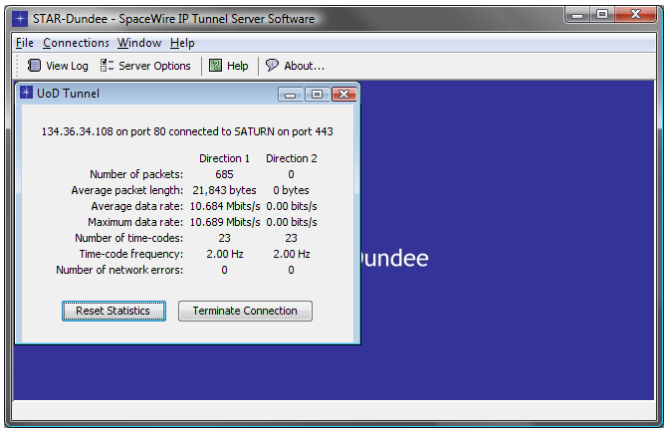

Figure 5: The main window of the SpaceWire IP Tunnel Server Software

The SpaceWire IP Tunnel Server is a software application which establishes connections between Tunnel pairs. A screenshot of this software in operation is shown in Figure 5. The Tunnel Server listens for connections from clients, and so must have the same firewall permissions as other servers. On establishing a connection, the Tunnel Server determines if the other connection in the pair has already been established. If it has, then a virtual connection is established between

the pair, and all traffic received from one client is passed on to the other. There is no difference than if the two clients were connected directly, so this has no effect on the Tunnel's mechanisms to cope with bandwidth and latency restrictions. Latency may be increased, however, as the traffic is now crossing two network connections.

Although a PC acting as a Tunnel Server must still have the same firewall permissions as a Tunnel end acting as a server, a Tunnel Server can manage multiple Tunnel client pairs. This means a single Tunnel Server could manage connections for a number of organisations. Alternatively, an organisation might want to setup their own Tunnel Server, just as they might have web and FTP (File Transfer Protocol) servers.

A further benefit of using a Tunnel Server is that the user does not need to know the address and port of the other end of the Tunnel, only those of the Tunnel Server. Connections are identified by name, and when connecting to a Tunnel Server a user can select a named connection from the list of those presently established. This can be particularly useful when establishing a number of Tunnel connections, or establishing connections with different Tunnel ends for different tests. For example, a user might connect to a connection named "Camera Subsystem" during some tests, and "Camera Subsystem Simulation" during others.

Communication with a Tunnel Server uses TLS to ensure the traffic cannot be viewed or modified, just as with a direct connection between two tunnel ends. In addition to the password used when establishing a connection, each named Tunnel connection can also have an associated password. To avoid connections from unauthorised users the Tunnel Server Software can deny access from specific addresses, or only allow specific addresses. The TCP ports to accept connections on can also be specified, as can the maximum number of connections that can be established.

As with the SpaceWire IP Tunnel Software, the SpaceWire IP Tunnel Server Software is written in Java. This means it can be run on any platform which supports Java and, as no communication with SpaceWire devices is performed, no specific drivers are required on the platform in use.

5 THE TOPNET PILOT STUDY

The theoretical advantages of virtual spacecraft integration and in particular the SpaceWire Internet Tunnel are quite obvious. However, until recently the real-world benefits (and limitations) when using the SpaceWire Internet Tunnel had not been investigated. An ESA funded pilot study, the TopNet Pilot Study, was completed earlier this year to investigate the benefits and limitations when in use within a real project environment. The study involved three consortia, each consisting of partners spread across Europe. Each consortium proposed experiments where SpaceWire devices situated in each of the consortium's partners would be virtually integrated.

The first stage of the study involved each of the consortia establishing Tunnel connections with the University of Dundee and exchanging SpaceWire traffic over that Tunnel. As well as giving the users the opportunity to familiarise themselves with the software and hardware, this also gave the University of Dundee an opportunity to identify any peculiarities in the users' networks. This lead to modifications to the Tunnel Software, with the addition of features such as proxy server support.

After familiarising themselves with the SpaceWire Internet Tunnel, the consortia then went on to conduct their experiments. These experiments were quite varied and covered a number of applications. While some experiments used software and hardware that had previously undergone integration testing, other experiments used software and hardware that was being tested together for the first time. Some experiments involved two forms of integration testing: both virtual integration testing and the traditional method of bringing all components to a single location. Numerous measurements were made during each of the experiments, providing valuable information on the impact a SpaceWire Internet Tunnel has in a SpaceWire network.

On completing their experiments, each consortium presented their findings and compiled a report containing their results and conclusions. The general consensus of those involved in the pilot study is that although there is still the possibility to make improvements, virtual spacecraft integration and the SpaceWire Internet Tunnel is a very useful tool for performing integration testing.

6 ADVANTAGES OF THE SPACEWIRE INTERNET TUNNEL

A number of the advantages of the SpaceWire Internet Tunnel are obvious:

- Integration testing is possible at an earlier stage of development
    - Saves time and money in correcting any problems
- Necessary travel is reduced
    - With associated financial and environmental savings
- Integration testing can be much more flexible
    - Integration testing can be performed at any time
    - Sub-systems can easily be replaced with simulators
- Geographical limitations are reduced
    - Cooperation between organisations involved in a project is improved

There are still limitations, however. There are some systems for which virtual spacecraft integration cannot allow proper integration testing to be performed. This includes those systems which have strict requirements on bandwidth and/or latency. An observation made by some of the contractors involved in the TopNet Pilot Study, however, was that it was often possible to perform limited integration testing for these systems using the SpaceWire Internet Tunnel. For example, by designing software to have configurable latency requirements, or even having a special "Tunnel" mode, initial integration testing could be performed to validate interfaces, etc. Once this testing is completed successfully, full integration testing can be performed.

The only other major limitation identified by the contractors was related to problems establishing connections, which should be addressed by use of the SpaceWire IP Tunnel Server, which was not available during the pilot study. Other issues identified by the contractors related to minor software issues and suggestions for improving the monitoring and recording capabilities of the SpaceWire Protocol Analyser. This emphasised the importance of analysis tools when performing virtual integration testing. Improvements to the software will continue to be made to add new features and address any issues identified.

A number of additional benefits of the SpaceWire Internet Tunnel not previously considered were also reported by the contractors:

- Virtual integration testing can be completed in a fraction of the time required for physical integration testing
- Electronic Ground Support Equipment (EGSE) does not need to be transported to an integration site when correcting problems, it can be virtually integrated
- When a system works using the Tunnel, it gives great confidence in the system
- The SpaceWire Protocol Analyser can be a very useful tool for monitoring and recording traffic, even when not tunnelling

7 CONCLUSIONS

This paper has noted a number of advantages of the SpaceWire Internet Tunnel. The TopNet Pilot Study has shown that these benefits can be realised in real projects. There may be some systems for which virtual spacecraft integration is not an ideal method of performing integration testing, but it has been demonstrated that some testing is still possible in such systems.

In conclusion, although physical integration testing of a system is still essential, virtual integration testing can be a very important stage in future projects.

8 ACKNOWLEDGEMENTS

The authors would like to acknowledge the support of ESA for the work performed on the SpaceWire Internet Tunnel at the University of Dundee. We would also like to thank the contractors involved in the TopNet Pilot Study for their valuable feedback.

9 REFERENCES

1. S. M. Parkes, “Virtual Satellite Integration”, DASIA (Data Systems In Aerospace) 2001, Nice, France, May 2001.

2. S. Mills, S. M. Parkes, R. Vitulli, “SpaceWire Internet Tunnel”, DASIA (Data Systems In Aerospace) 2005, Edinburgh, Scotland, UK, May 2005.

3. S. Mills, S. Parkes, R. Vitulli, “Virtual Satellite Integration and the SpaceWire Internet Tunnel”, International SpaceWire Conference 2007, Dundee, Scotland, UK, September 2007.

4. European Cooperation for Space Standardization, “SpaceWire, Links, Nodes, Routers and Networks”, Standard ECSS-E-50-12A, Issue 1, European Cooperation for Space Data Standardization, February 2003.

5. STAR-Dundee, “STAR-Dundee Website”, STAR-Dundee, http://www.star-dundee.com/.

6. European Cooperation for Space Standardization, “SpaceWire Protocols”, Draft Standard ECSS-E-ST-50-11C, Draft 1.2, European Cooperation for Space Data Standardization, July 2008.

7. T. Dierks, E. Rescorla, “The Transport Layer Security (TLS) Protocol, Version 1.2”, Request for Comments 5246, Internet Engineering Task Force, August 2008.