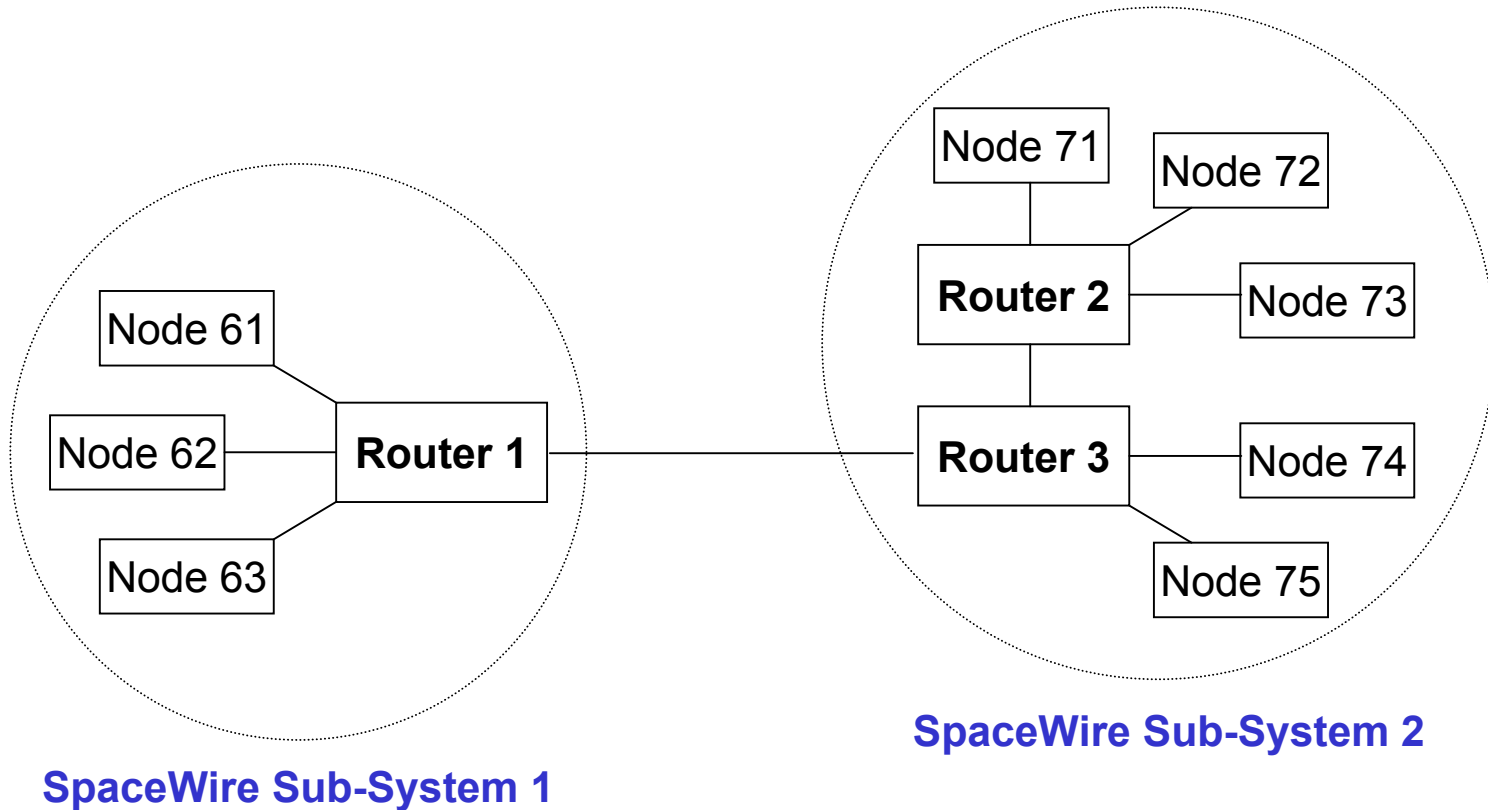


TopNet implementation: SpW IP Tunnel and Protocol Analyser

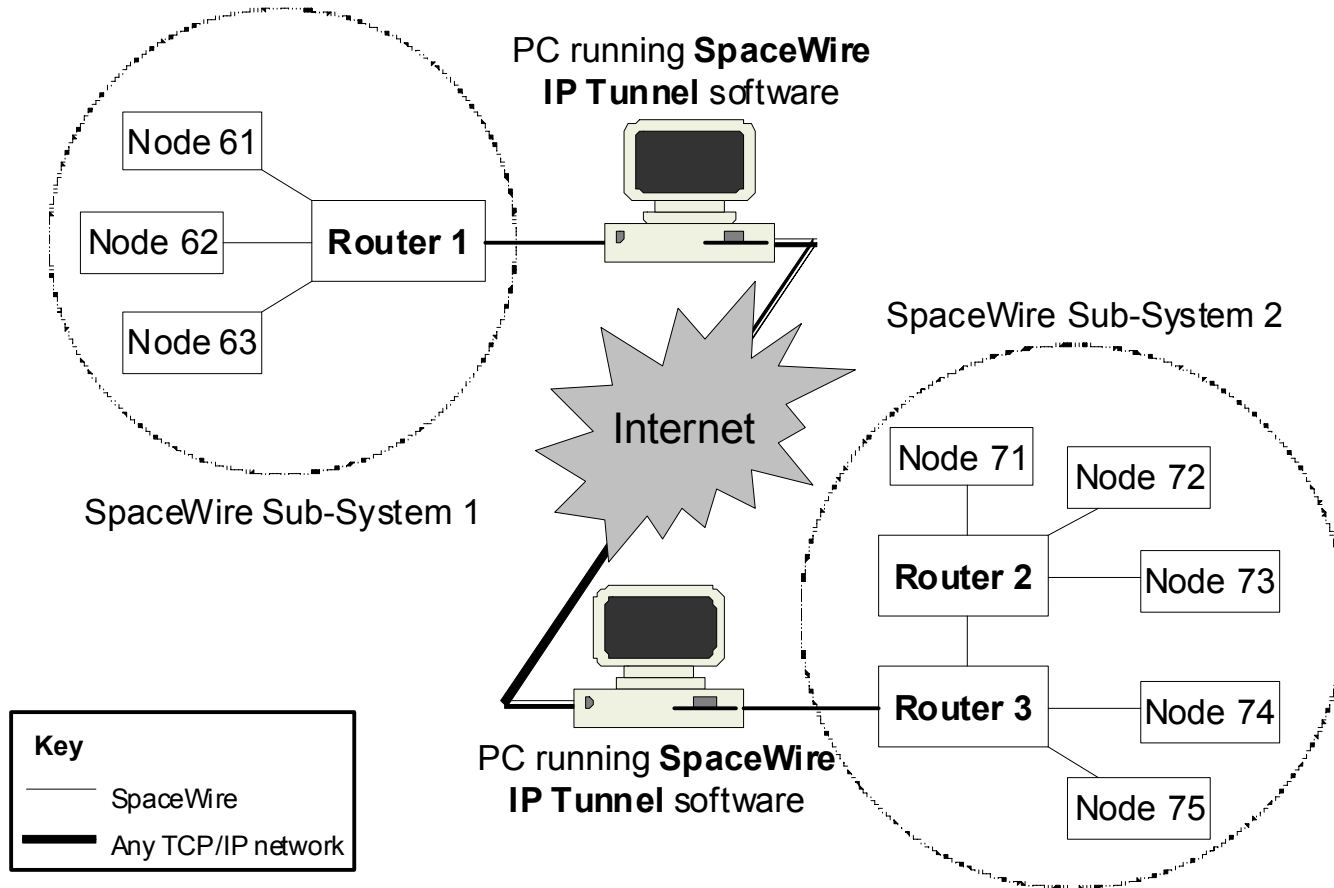
Vitulli R. TEC-EDP

Email: Raffaele.Vitulli@esa.int

SpaceWire System



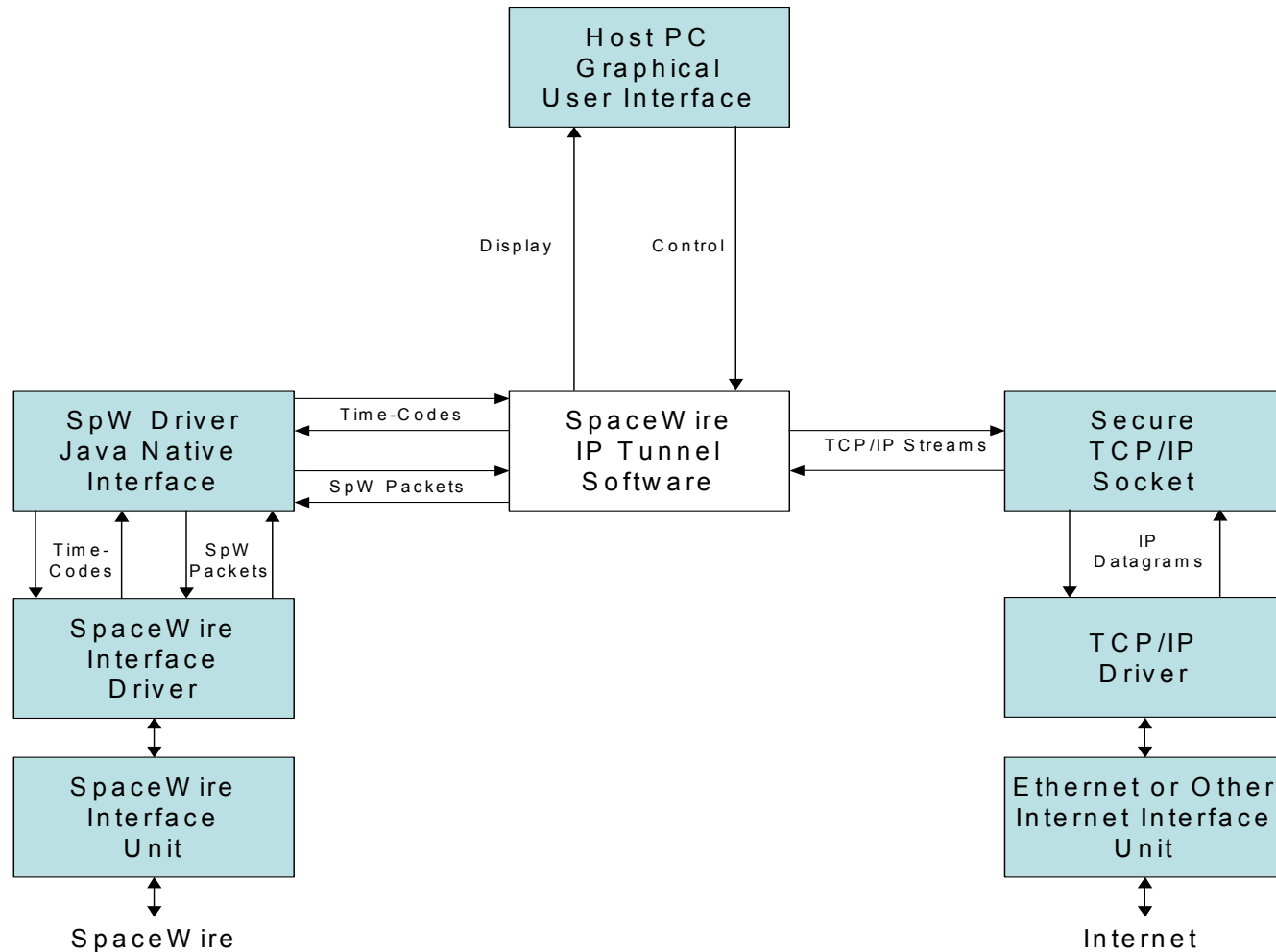
SpaceWire Tunnel Concept



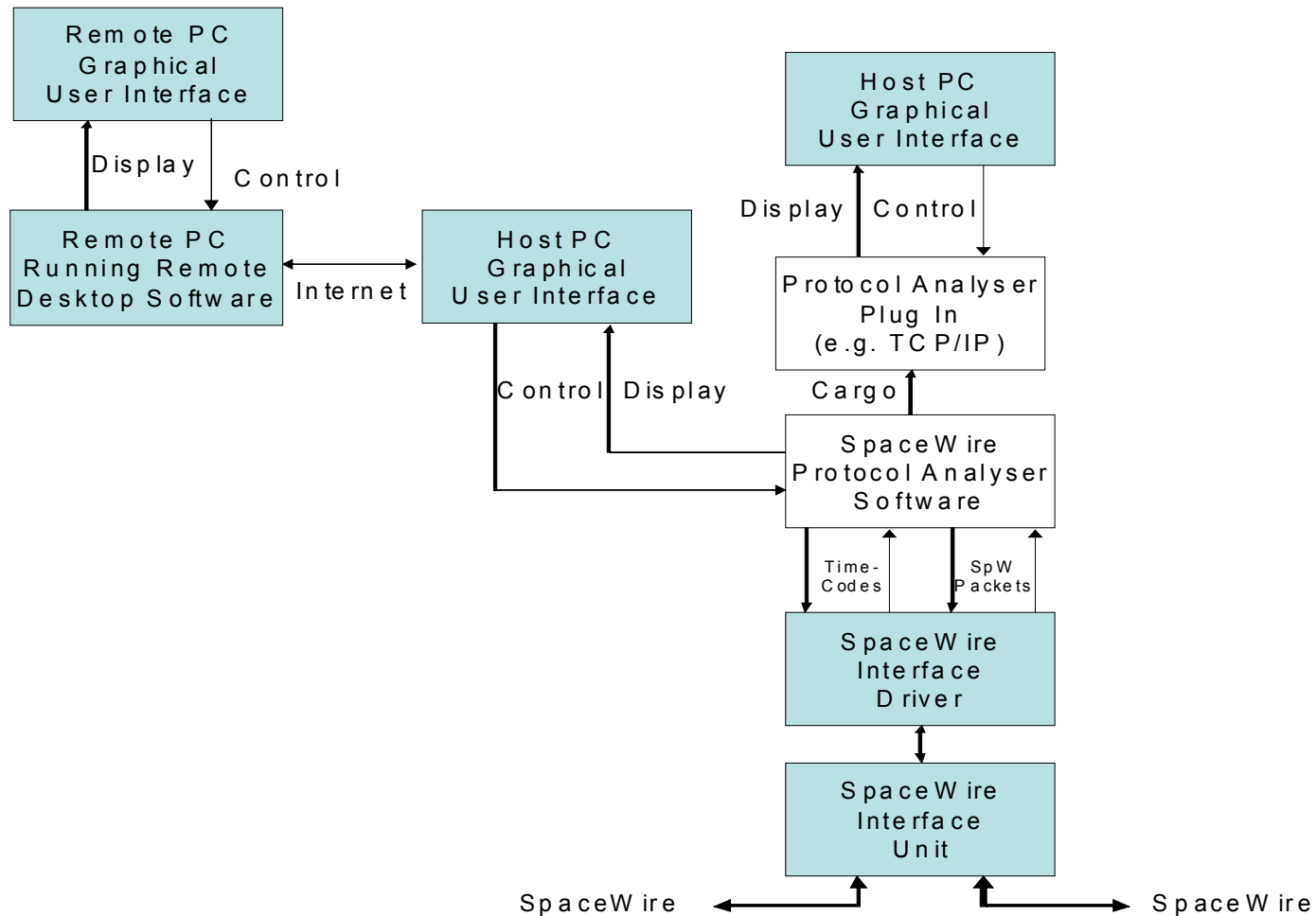
Aims of IP Tunnel

- To provide support for remote, decentralised integration of SpW-based, satellite onboard data-handling sub-systems using the internet to connect the geographically separated sub-systems.
- To provide a means for monitoring the traffic flowing through the SpaceWire IP Tunnel.
- To provide a way of monitoring SpaceWire traffic remotely using available multi-port SpaceWire to PC interface units.
- To enable higher level protocols running over SpaceWire, through a SpaceWire IP Tunnel or through a multi-port SpaceWire interface, to be analysed and displayed.
- Due to bandwidth limitations and latency of the connection, real-time communication is not possible.

Spacewire IP Tunnel Model



Protocol Analyzer Model



Time-Code and N-Char Sequencing

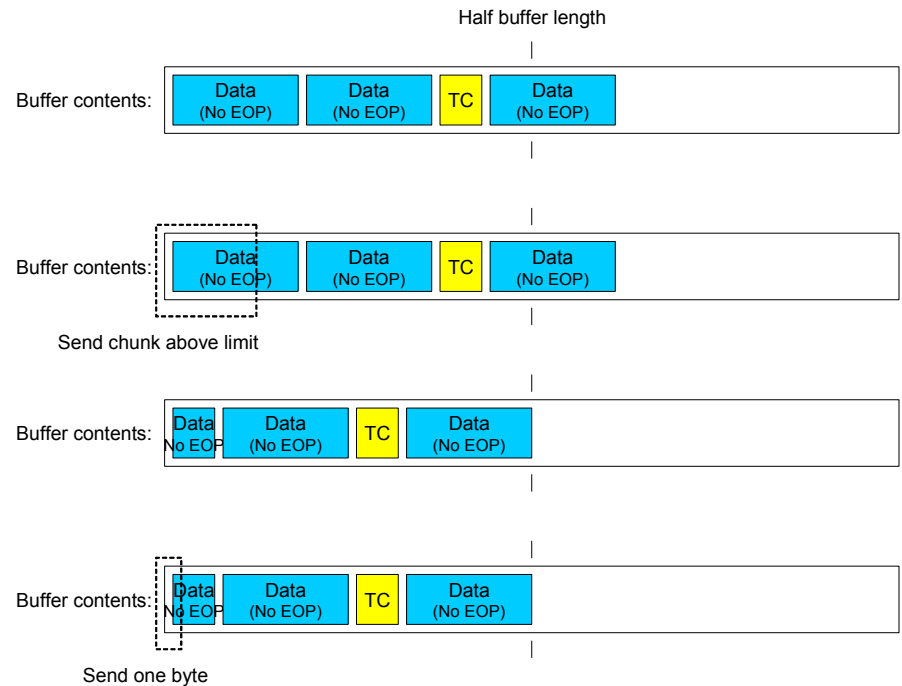
- Inside SpaceWire CODEC
 - Time-codes have priority
 - they jump the transmit and receive queues
- Not possible to ensure full synchronisation of time-codes and data without modifying the SpaceWire CODEC
- IP Tunnels must preserve N-Char / Time-code ordering
- The best way is to use a modified SpaceWire-USB Brick, in order to keep time-code to N-Char ordering

Link Start/Disconnect

- IP Tunnels must be completely transparent
- Tunnel devices shall support notification of “link start” and “link disconnect” to the other end of the Tunnel
- Also link start/disconnect notification has been implemented in the modified SpaceWire-USB Brick

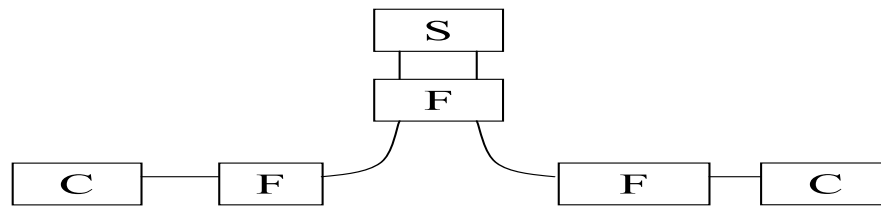
Tunnel in a network using routers with timeouts

- the gap between packet chunks could be greater than the timeouts at the router
- A proper buffer is necessary at the tunnel exit



Operation of Tunnel behind firewall

- IP Tunnel is using port 80 as a default. One of the 2 Tunnel should be outside the firewall.
- Alternative solutions:
 - Use of SSH port 22 to forward traffic to a non-privileged port (> 1024)
 - to have a centralised server which provides connection between pairs of Tunnel clients



C = Client, F = Firewall, S = Server

Tunnel Hardware

- In order to provide the full functionality, a modified device is required.
- The IP Tunnel software can also be used with existing SpaceWire devices, such as the SpaceWire-USB Brick.
- When used with these devices the Tunnel has some limitations (order of time-codes and links disconnect/start)

Tunnel Software

- Changes were required to the driver which provides the interface with the device.
- The SpaceWire IP Tunnel application provides the man-machine interface and the tunnelling functionality (coded in Java).
- Windows and Linux drivers are currently available for the SpaceWire devices.

Present Status

- The SpaceWire driver is completed.
- Detailed testing of the driver was performed, and identified bugs were fixed
- The Java interface to the driver is completed
- Tunnel Software is completed and under testing

To be done:

- Complete Protocol Analyzer user interface
- Complete the User Manuals for both the Tunnel and Protocol Analyzer
- Delivery date: end of July

Follow-on activities

Involvement of different actors (industry, university, agency) in a pilot activity for decentralized integration of SpW-based data handling sub-systems that are geographically separated

