scisys

Distributed, Real-Time, Embedded Systems

The SOIS Plug-and-Play Architecture and Its Proposed Mapping onto SpaceWire

S.D. Fowell, SciSys Ltd, UK

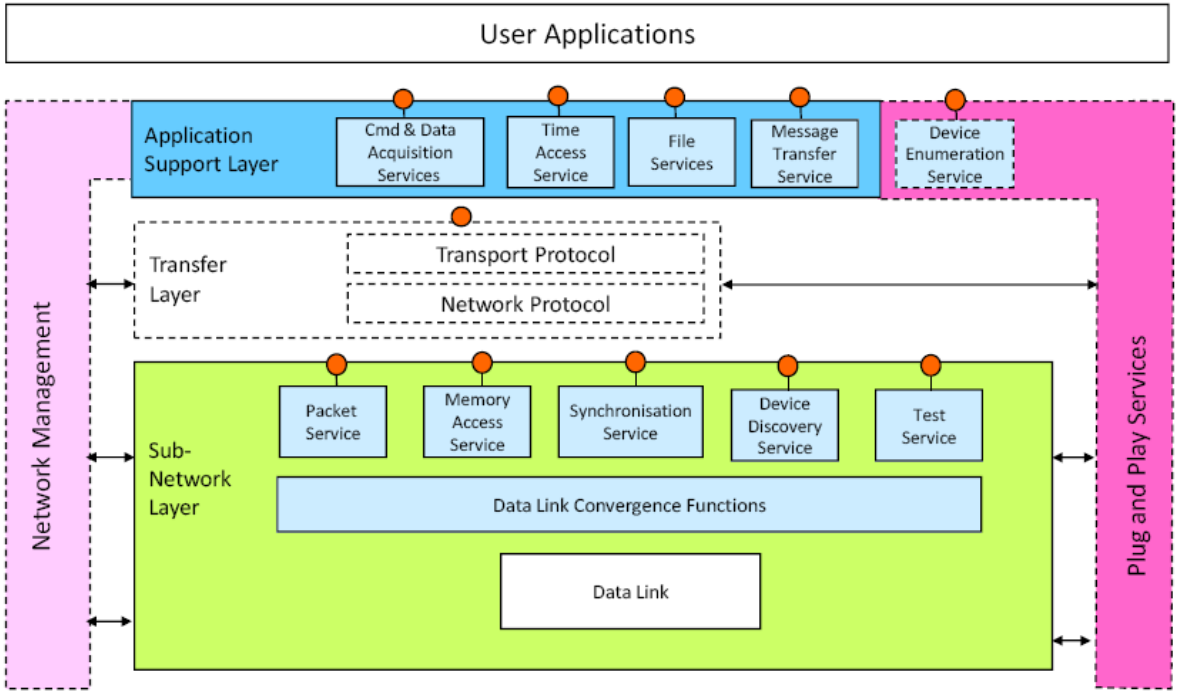
Presented by:

Stuart D Fowell

Overview

- Existing CCSDS SOIS Architecture
- Plug-and-Play Requirements
  - ⇨ Definition of “Plug-and-Play”
  - ⇨ SOIS Plug-and-Play Use Cases
  - ⇨ SOIS Plug-and-Play Requirements
- SOIS Plug-and-Play Architecture
  - ⇨ Existing Plug-and-Play Technologies and Studies
  - ⇨ Device and Service Discovery, Device Adaptation
- Mapping onto SpaceWire & Prototyping
- Example of Powering up a SpaceWire Device

Existing CCSDS SOIS Architecture



- **Command and Data Acquisition Services**, that provide mechanism for commanding of and acquiring data from devices within a spacecraft;
- **Message Transfer Service**, that provides transfer of messages between software applications within a spacecraft;
- **Packet Service**, that provides transfer of packets between data systems within a subnetwork of a spacecraft;
- **Memory Access Service**, that provides access to memory locations of a data system from another data system within a subnetwork of a spacecraft.

Existing CCSDS SOIS Architecture

- The first set of standards have been reviewed by the various Space Agencies and are now available as Red Books and awaiting prototyping before publication as Blue Books.
- ECSS WGs are currently developing protocols to provide the mappings onto SpaceWire and MIL-STD-1553B and a similar exercise is planned in 2008/9 for CAN.
  - ⇨ For SpaceWire, the SpaceNet project is prototyping SpaceWire-RT protocols suitable for use by the SOIS Packet and Memory Access Services.
- However, these standards only address a statically or top-down configured communications architecture – the networks are not self-describing.
- In addition, support for “wireless” capabilities is being considered and by their nature can result in a more dynamic communications architecture.
- To address these issues, it has been identified that the SOIS architecture needs extending to support “plug-and-play” concepts and a “Birds-of-a-Feather” (BoF) grouping has been organised to address this.

Definition of “Plug-and-Play”

- **“Plug and play** is a computer feature that allows the addition of a new device, normally a peripheral, without requiring reconfiguration or manual installation of device drivers. ... Modern plug-and-play includes both the traditional boot-time assignment of I/O addresses and interrupts to prevent conflicts and identify drivers, as well as hotplug systems such as USB and Firewire.” – www.wikipedia.com

- In the context of the Spacecraft domain, **Peripherals** should include:
  - devices traditionally associated with avionics
    - simple (e.g. thrusters, magnetometers, thermistors)
    - more complex (e.g. star trackers)
  - simple instruments

- **Plug-and-Play** doesn’t extend to full integration of whole sub-systems or computers (as this includes software not device integration)
  - However, it does have a role in simplifying integration at the subnetwork layer

- **“SOIS Device Plug-and-Play** is the set of automated mechanisms used to discover, learn the capabilities of, and provide access to a *device* in a spacecraft’s onboard (sub-)network.” – SOIS Plug-and-Play BoF

SOIS Plug-and-Play Use Cases

- **Dynamic Spacecraft Network Reconfiguration** – activation of redundant devices upon a flying spacecraft in response to faults. A Fault Detection, Isolation and Recovery (FDIR) system application simply powers up replacement. Reconfiguration happens automatically (bottom-up), rather than hierarchically (top-down)
- **Spacecraft Integration & Test** – Electrical Ground Support Equipment (EGSE) connection to Spacecraft under test using wireless technologies
- **Rapid Spacecraft Assembly of Devices** – to reduce/eliminate the need for aspects of Spacecraft database for configuring OBSW
- **Roaming Heterogenous Devices** (e.g. Biometric Health Monitoring of ISS/Orbiter crew) – characterised as facilitating the incorporation of heterogeneous sensing and control devices in a wireless, heterogeneous communications network

Use Cases out-of-scope for SOIS Plug-and-Play (though that is not to say that SOIS Plug-and-Play may not have a role to play within them):

- **Onboard Software Upgrade or Reconfiguration** – covering mode changes or software updates. This is purely a software change with no new data systems introduced
- **Rapid Spacecraft Assembly of Subsystems** – SOIS Plug-and-Play simplifies at the subnetwork integration of subsystems, but also requires exchange of info. using perhaps a s/w framework or middleware, beyond the present scope of SOIS. However, a s/w framework would exchange messages using Message Transfer Service so SOIS Plug-and-Play aids but does not fully solve this

SOIS Plug-and-Play Requirements

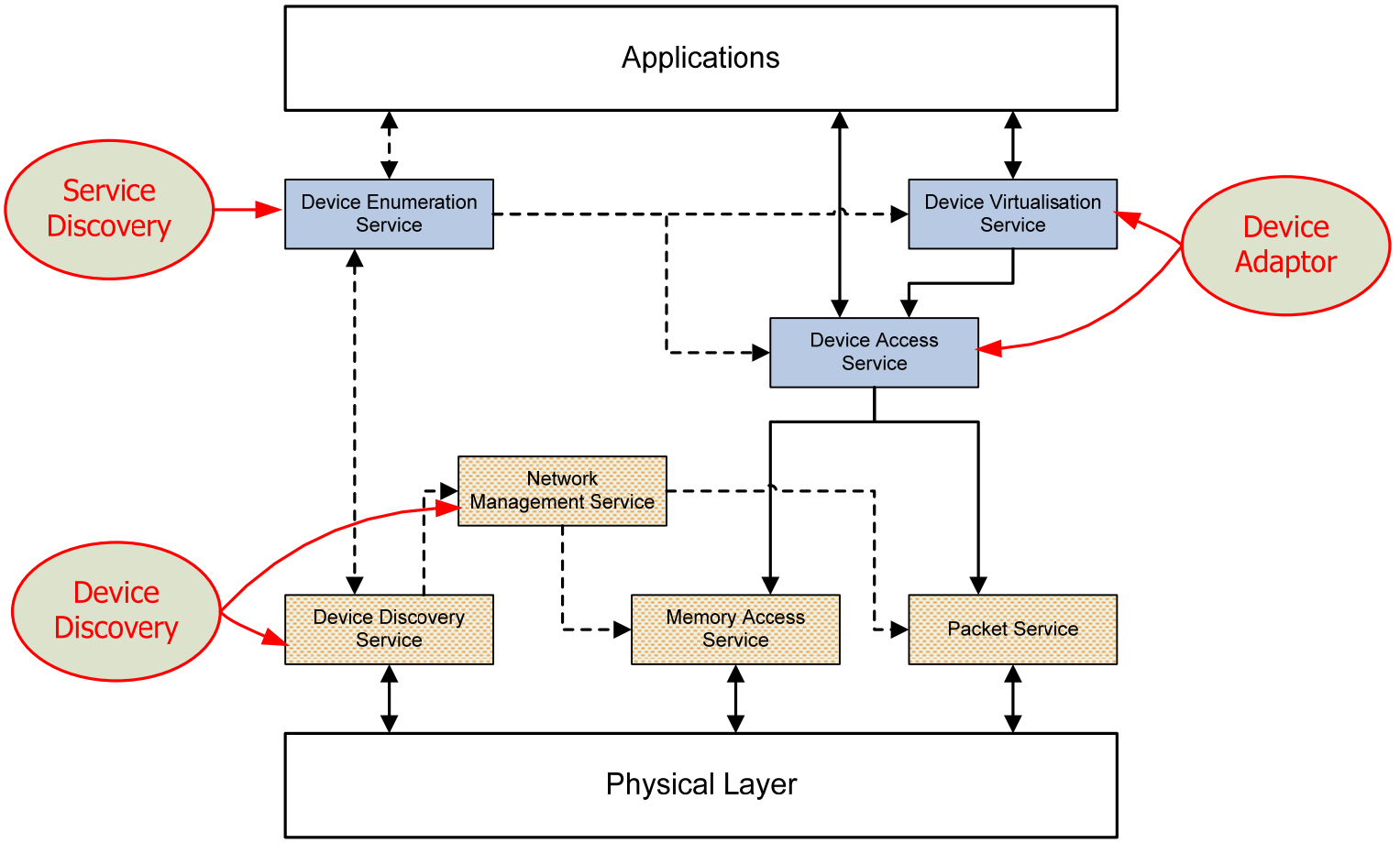
Support mechanisms to:

- discover devices added to a SOIS subnetwork;
  - ⇨ powered up, mechanically inserted, electing to enter e.g. sending announcement packet
- discover devices removed from a SOIS subnetwork;
  - ⇨ switched off, failed, mechanically removed, out of range, electing to withdraw
- discovery of capabilities of added devices;
- reconfigure SOIS services to allow command of and/or acquisitions from added devices;
- reconfigure SOIS services to disallow command of and/or acquisitions from removed devices;
- to notify users (applications and higher layer services) of added and removed devices and their capabilities.

Reference SOIS Device Plug-and-Play Architecture

- Large number of buses, data links and networks considered
  - ⇨ Inc. MIL-STD-1553B, SpaceWire, Serial, 802.11, Bluetooth, Ethernet, IEEE 1451, USB, FireWire, 1-wire
- Different subnetwork types have differing existing levels of capability.
- Generally split into 3 functions (some or all of which may be present):
  - ⇨ **Device Discovery** – discovery of initial and subsequent changes to subnetwork topology and devices added to/removed from subnetwork
  - ⇨ **Service Discovery** – discovery of capability of devices added to subnetwork
  - ⇨ **Device Adaptation** – adaptation of specific devices to generic classes and functions

Reference SOIS Device Plug-and-Play Architecture



Device Discovery

- Discovers and enables communication using SOIS subnetwork services with added device and notifies higher layers of changes
- Major subnetwork-specific functions consist of:
- **Device Discovery Service**:
  - ⇨ Discovery of initial subnetwork topology, including current devices on subnetwork
  - ⇨ Discovery of any changes to subnetwork topology, including addition or removal of devices through failure or control disconnection (inc. power down)
    - ↳ E.g. specific discovery mechanism, e.g. by broadcasting for new devices, or react to a subnetwork event, e.g. a trigger that a new device has been powered up or inserted into the subnetwork.
  - ⇨ Reconfiguration of other SOIS Subnetwork Layer Services to allow communication with new device
  - ⇨ Notification of device changes (addition/removal)
- **Network Management Service**:
  - ⇨ Assignment of addresses to newly added devices
  - ⇨ Configuration of any routing functions of subnetwork, e.g. routers
- Typically this is already provided by existing subnetworks

Service Discovery

- Discovers and enables use of capability of added devices and notifies higher layers of change
- Major functions consist of:
- **Device Enumeration Service**, which is responsible for managing the discovery of added device's capabilities and its insertion into the SOIS communications architecture.
  - ⇨ Discovering the capabilities of the device
  - ⇨ Reconfiguration of other SOIS Application Support Layer Services to allow use of new device
- Differing levels of capabilities provided by existing subnetworks. More opportunities for existing or new generic solutions
- Number of options for discovering the capability of the device, all based in part on reading information from the device itself, including:
  - ⇨ **Device class and type** information read from device, match with expected device classes and types, e.g. spacecraft database, etc
  - ⇨ **Electronic Data Sheet** (EDS) defines the device type and capabilities (e.g. functions, protocols and classes-of-service supported), e.g. xTEDS

Device Adaptation

- Provide generic interface to functions of classes of devices, adapting between generic and specific interfaces, e.g. information encoding and command/access protocols
- Can/should be used independently of plug-and-play!

- **Device Virtualisation Service**
  - ⇨ Mapping of standard functions onto device specific functions, including calibration/unit conversions, formatting of commands and data
  - ⇨ Requires **standardisation** of framework and **class types and functions**
  - ⇨ ECSS SSDHI standard used as a starting point for classification
  - ⇨ Needs to be extensible to add:
    - ↳ New classes
    - ↳ New specific devices

- **Device Access Service**
  - ⇨ Existing SOIS Service that provides “access” to device through defined command and data acquisition protocols, i.e. subnetwork type and address and packet/memory access service
  - ⇨ Needs to be extensible to add:
    - ↳ New subnetwork types
    - ↳ New subnetwork protocols?

- Both services updated by Device Enumeration Service when a device is added or removed
- May be **dynamic**, e.g. XML Interpreter using xTEDS, or **static**, e.g. hard-coded device classes with MIB updated when new device discovered

Mapping onto SpaceWire & Prototyping

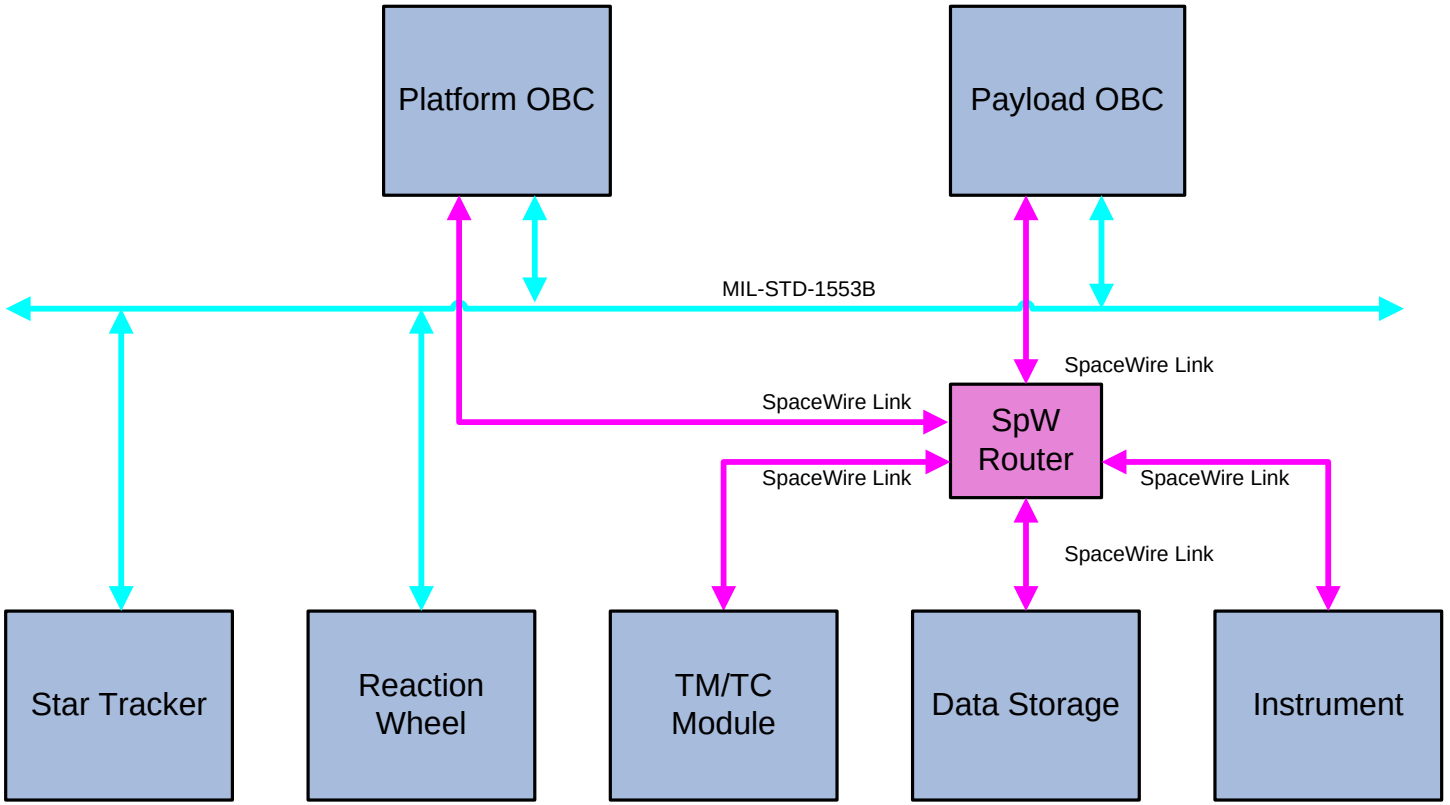
- **Developing SpaceWire Plug-and-Play standard has been considered and seems to provides Device Discovery and simple Service Discovery capabilities required by SOIS**
  - ⇨ SpaceWire Plug-and-Play, Draft A, 31st Jan 2008, Peter Mendham (UoDundee) & Glenn Rakow (NASA-GSFC)

- To confirm this, the SOIS Reference Implementation project is prototyping SOIS Plug-and-Play architecture using SpaceWire

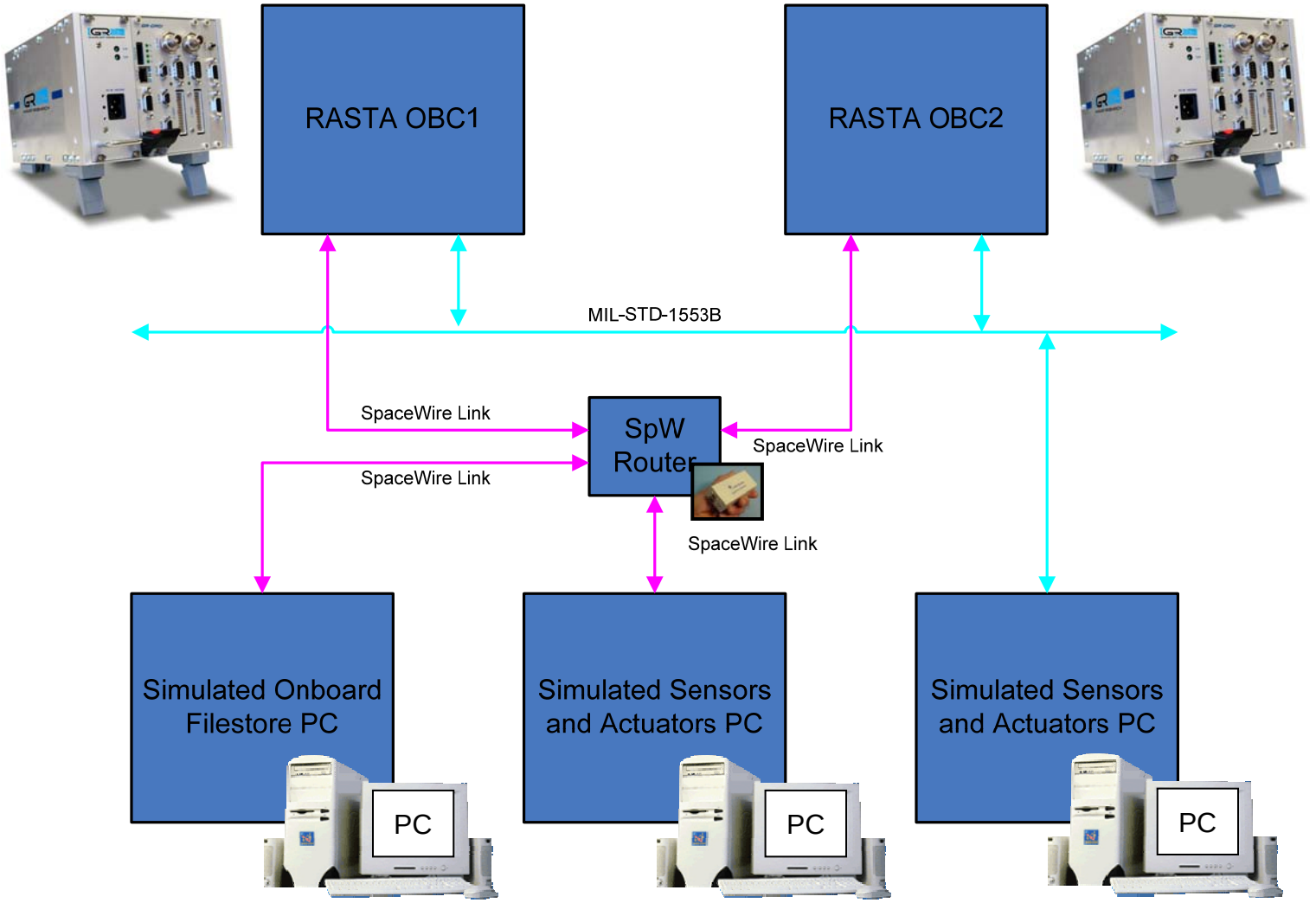
- Using RASTA and UoD USB-SpaceWire Router (SpW_10X)

- No support in Router for notification of changes to links
- No support in Router for PnP protocol, but does provide equivalent features
- Instead polling of router and node registers using RMAP and Configuration Port 0 is used
- Single master (other redundancy mechanisms can be used to failover to redundant master)
- Read device class and type from device (not complicated xTEDS!)
- Using fixed SpaceWire network topology as focussing of Device Plug-and-Play for prototype

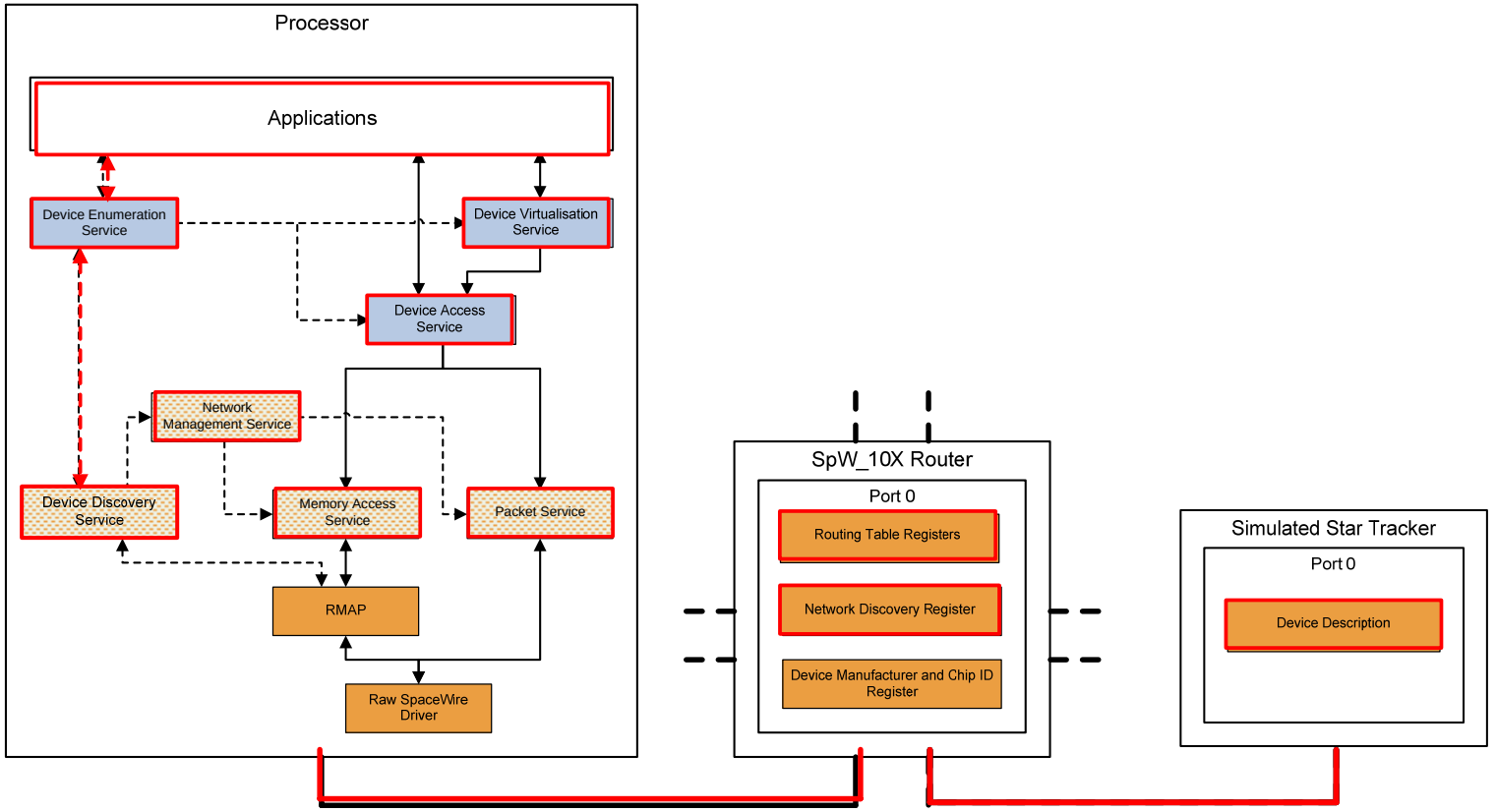
SOIS Plug-and-Play Prototype Demonstrator



SOIS Plug-and-Play Prototype Demonstrator



Example of Powering up a SpaceWire PnP Device



Example of Powering up a SpaceWire PnP Device

1. Device is powered up and activates its SpaceWire link. Through normal SpaceWire mechanism, link to SpaceWire router automatically starts and this is indicated in the Network Discovery Register
2. SpaceWire Device Discovery Service on OBC1 polls router's Network Discovery Register and discovers a newly active port. Uses path address routing and RMAP to attempt to read device class register on port 0 of node at end of newly active link. RMAP reply packet successfully returns read class and type
3. SpaceWire Network Management Service on OBC1 allocates it a SpaceWire logical address and updates the SpaceWire Routers routing tables
4. SpaceWire Device Discovery Service notifies the Device Enumeration Service on OBC1 of the new device, including its subnet type, address, class and type
5. Device Enumeration Service allocates the new device a Device Identifier, looks up the access service (e.g. SpaceWire Memory Access Service -> RMAP) for a device of that class and type in its MIB, and updates the MIB of the Device Access Service with the device identifier, address and access service
6. Device Enumeration Service updates the MIB of the Device Virtualisation Service so that the device identifier is associated with the preallocated functions of the device class and the adaptation mechanisms of the device type
7. Device Enumeration Service notifies a registered OBSW application that a new device of class X has been added to the system. This OBSW application may perform an initial configuration of the device and/or configure the software system to start using the device, e.g. FDIR to recover from a failed device

SOIS Plug-and-Play BoF

- Active group defining SOIS Plug-and-Play
- Current members from Europe and USA include:
  - ⇨ Scott Burleigh (NASA/JPL), Max Ciccone (ESA), Stuart Fowell (BNSC/SciSys), Kevin Gifford (UoColorado), Paul Jaffe (NRL), Ramon Krosley (DesignNet), Jim Lyke (USAF), Glenn Rakow (NASA-GSFC), Gary Rod (sysRAND), Francisco Tortosa (ESA) and others...!

- Completed Technology Survey for Device Discovery
- Current focus is on Service Definition and Discovery
- Finally we will categorise Device Classes and Types

- Roughly fortnightly teleconferences discussing and refining concepts
- Next teleconference is on 17th June 16:00 BST – send me an email if you want to join in! (stuart.fowell@scisys.co.uk)

- The next CCSDS SOIS meeting is mid October 2008 at Berlin, Germany, where the SOIS Plug-and-Play Demonstrator will be demonstrated!

And finally the credits…

- The SOIS Plug-and-Play standardisation is sponsored in Europe by ESA and BNSC

- The SOIS Plug-and-Play prototyping in support of this is carried out as part of the **SOIS Reference Implementation** project, an ESA TRP project, being undertaken by SciSys and Astrium Satellites SAS in 2008