THE SPACEWIRE-PNP PROTOCOL IN THE SOIS PLUG-AND-PLAY ARCHITECTURE

Session: SpaceWire Networks and Protocols

Short Paper

Peter Mendham, Stuart Fowell

SciSys UK Ltd, Clothier Road, Bristol, BS4 5SS, UK

Chris Taylor

ESA/ESTEC, 2200 AG Noordwijk ZH, The Netherlands

E-mail: peter.mendham@scisys.co.uk, stuart.fowell@scisys.co.uk, chris.taylor@esa.int

ABSTRACT

The SOIS architecture has been steadily evolving for a number of years and includes an effort to incorporate the features of “plug-and-play” systems which are relevant to onboard communications. This paper reviews the role that plug-and-play can play in an onboard system architecture, and within SOIS more specifically. The SOIS plug-and-play architecture is described and the role of the subnetwork is highlighted specifically. The SpaceWire-PnP protocol is assessed against the requirements for a SOIS implementation, and is presented as an exemplar of a subnetwork plug-and-play protocol.

INTRODUCTION

The SOIS plug-and-play architecture has been steadily evolving for a number of years, with input from a range of studies and prototyping activities. Additionally, recent work has benefited from the experience of NASA with Space Plug-and-Play Avionics (SPA), clarifying the alignment of such international efforts with the SOIS architecture. As an onboard communications technology with growing popularity, SpaceWire has always been carefully considered within the SOIS plug-and-play architecture and previous work has described potential mappings of SOIS plug-and-play onto SpaceWire [1]. Concurrently, the SpaceWire-PnP protocol is being developed with the intention of providing standard mechanisms for carrying out crucial network management and configuration tasks on SpaceWire networks, including device and network topology discovery.

Section 00 begins this paper by reviewing the plug-and-play problem within the larger context of onboard system architecture, from applications to hardware. The SOIS plug-and-play architecture is the focus of this discussion, described in Section 0, where the roles that a subnetwork plug-and-play protocol must fulfil to meet the requirements of SOIS are highlighted. In Section 00, SpaceWire-PnP is applied within the SOIS architecture as an exemplar subnetwork protocol, with an examination of the portions of the architecture that are subnetwork specific, and those that are mission specific. Section 00 concludes the paper.

ONBOARD PLUG-AND-PLAY

The SOIS Green Book [2] defines plug-and-play as “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”. One of the goals of SOIS is provide standards for such mechanisms to improve the interoperability and portability of onboard systems. An implementation of SOIS is typically software-centric, although this does not have to be the case. We will use software terminology here, beginning with onboard applications.

The focus of device access by an onboard application is the functional characteristics of that device type. For example, accesses on a sun sensor will be largely concerned with acquiring the position of the sun, whilst those on a reaction wheel will be largely concerned with commanding the wheel torque. The application is not concerned, except by necessity, with the underlying mechanism required to acquire the sun position or command the wheel torque. By abstracting the generic operations of a device type, or class, from its underlying access mechanisms, application software can be made more portable. Such a technique is known as *device virtualisation*, and forms an integral part of plug-and-play strategies on many platforms.

The presentation of a virtual device interface to an application relies on two processes: access to the device, potentially using some device-specific protocol; the adaptation of the semantics of the device access mechanisms to match the expected semantics of the virtual device. In turn, the device-specific protocol will rely on the mechanisms provided by the technology used to communicate with the device, i.e. the subnetwork. The definition of the Device-Specific Access Protocol (DSAP) and translation mechanisms for virtualisation could be implemented directly into software (or hardware). More flexibly, however, they could be described in a structured, machine-readable form such as an Electronic Data Sheet (EDS). The use of an EDS permits the automatic generation or tailoring of implementations handling the DSAP and/or virtualisation translations.

Which devices are present in the system, may be statically “hard-coded” into the communications software, or it maybe dynamically determined at run time. In the case of a static system, it may be still be necessary to verify the configuration of services against which devices are logically present on the subnetwork. This verification requires the same underlying subnetwork mechanism: to be able to discover and uniquely identify all devices on the subnetwork. If the subnetwork requires management and configuration in order to be able to access and use devices to their full potential, this must be carried out, according to some mission-specific policy. The dynamic detection of devices also raises the possibility of dynamically utilising an EDS, hosted by the device, to suitably adapt or tailor the DSAP and/or virtualisation translations associated with the device.

The subnetwork must therefore support standard mechanisms to support the access of devices, which must be supported by plug-and-play mechanisms such as device discovery and the management of subnetwork configuration.

THE SOIS PLUG-AND-PLAY ARCHITECTURE

The SOIS Plug-and-Play architecture instantiates the major features of the mechanisms described in Section 0 in explicit service interface descriptions. It is expected that these service interfaces will be implemented by suitable services, relying on the underlying hardware or driver software for a subnetwork such as SpaceWire. An example onboard software architecture implementing the SOIS plug-

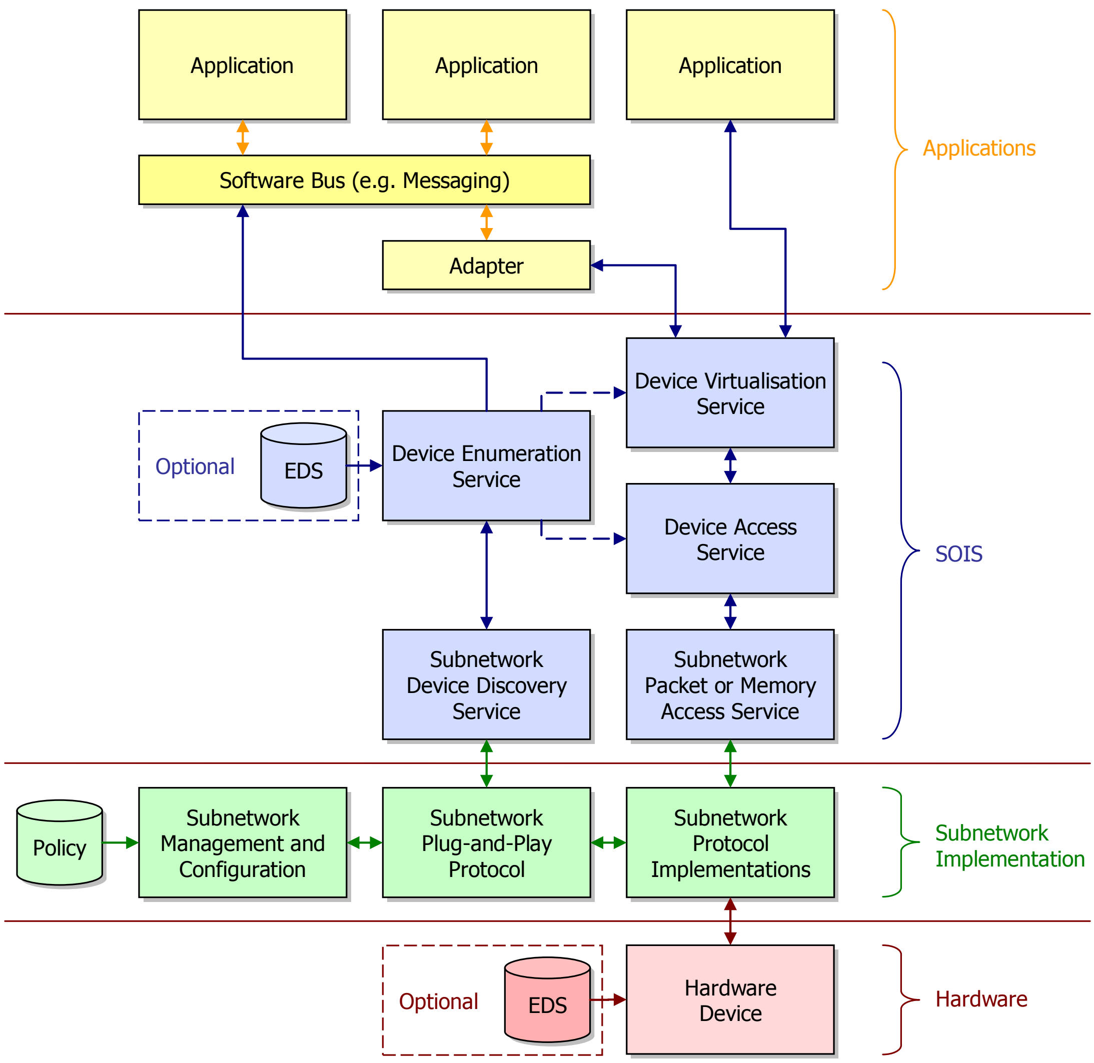

Figure 1: Onboard Software Architecture Using SOIS Plug-and-Play

The area marked as "Subnetwork Implementation" corresponds to the SpaceWire interface, and SpaceWire protocol handling. This layer is the appropriate place to handle subnetwork plug-and-play which must support: device discovery functions sufficient to support the subnetwork Device Discovery Service (DDS) and support for managing and configuring all standard subnetwork features. To permit dynamic configuration using an EDS, subnetwork plug-and-play should support a standard mechanism for querying an EDS from a device.

Whereas the plug-and-play protocol and mechanisms can be generic for a subnetwork, actual subnetwork management activities may well be specific to a mission. To reflect this, the policy guiding the management and configuration of a subnetwork has been explicitly depicted in Figure 1.

SPACEWIRE-PNP FOR SUBNETWORK PLUG-AND-PLAY

The SpaceWire-PnP proposal [3] was initially developed by one of the authors whilst at the University of Dundee [4-7] and has continued with input from SciSys [8-9] and the SpaceWire community. The core of SpaceWire-PnP is the following services:

- Device Identification, which provides basic information about devices such as their vendor ID and how many ports they have.
- Network Management, which permits the discovery of devices and network topology.
- Link Configuration, which permits the configuration of SpaceWire links on a device.
- Router Configuration, which permits the configuration of routing features and applies only to SpaceWire routers.

Additionally, SpaceWire-PnP is extensible: so-called “capability services” provide for, amongst other things, generic data sources and sinks. The simplest source service was designed considering the requirement for hosting EDS data in a device.

As can clearly be seen, the services provided by SpaceWire-PnP meet the requirements for the SOIS plug-and-play architecture. Device Identification and Network Management services of SpaceWire-PnP provide all functions necessary for the subnetwork independent SOIS DDS, although there is an opportunity to make these correlate more closely. The Link and Router Configuration services provide for the subnetwork specific configuration and management of all SpaceWire features identified in the SpaceWire standard.

SUMMARY AND CONCLUSIONS

The SOIS plug-and-play architecture supports the application of plug-and-play principles, including virtualisation and device discovery using static and or dynamic methods. This architecture relies on device discovery features provided by the subnetwork, which must also take responsibility for management and configuration of subnetwork resources. This combination of discovery, configuration and management functions must be met by suitable subnetwork plug-and-play mechanisms, such as SpaceWire-PnP.

The flexible topology, peer-to-peer approach and distributed resources of SpaceWire are, in many ways, a superset or more general case of the characteristics provided by other communications media. The SpaceWire-PnP proposal is therefore a good model for the development of other plug-and-play protocols. Ongoing work at SciSys is prototyping key elements of SpaceWire-PnP, including the principles behind device discovery and electronic data sheet use. This work clearly indicates the potential power and flexibility of applying SpaceWire-PnP within the SOIS plug-and-play architecture.

REFERENCES

1. Fowell and Taylor, 'The SOIS Plug-and-Play Architecture and its Proposed Mapping onto SpaceWire', Proceedings of the 1st International SpaceWire Conference, Dundee, UK, 2007.

2. CCSDS, 'Spacecraft Onboard Interface Services – Informational Report', CCSDS 850.0-G-1, Green Book, Issue 1.0, 2007.

3. University of Dundee, "SpaceWire-PnP: Protocol Definition', Version 2.1, Available from the minutes of the 14th SpaceWire Working Group, http://spacewire.esa.int/WG/SpaceWire/, 2010.

4. Mendham *et al*, 'Plug-and-Play Technology for SpaceWire: Drivers and Alternatives', IAC-07-B4.7.06, Proceedings of the 58th International Astronautical Congress, Hyderabad, India, 2007.

5. Mendham *et al,* 'Interoperability and Rapid Spacecraft Development Using SpaceWire Plug-and-Play', IAC-08-B4.7.02, Proceedings of the 59th International Astronautical Congress, Glasgow, UK, 2008.

6. Mendham *et al*, 'Network Management and Configuration using RMAP', Proceedings of the 1st International SpaceWire Conference, Dundee, UK, 2007.

7. Mendham and Parkes, 'SpaceWire Plug-and-Play: A Roadmap', Proceedings of the 2nd International SpaceWire Conference, Nara, Japan, 2007.

8. Mendham *et al*, 'Standardised Sensor and Actuator Interfaces with SpaceWire-PnP', IAC-09-B4.7.02, Proceedings of the 60th International Astronautical Congress, Daejeon, South Korea, 2009.

9. Mendham, 'The SpaceWire-PnP Protocol', Available from the minutes of the 14th SpaceWire Working Group, http://spacewire.esa.int/WG/SpaceWire/, 2010.