

Standardising Spacecraft Onboard Interfaces – The CCSDS SOIF Activity

By Chris Plummer, Cotectic Ltd, Joseph Smith, NASA JPL, and Patrick Plancke, ESA ESTEC

August 2002

1. Abstract

CCSDS is best known for its standards for packet telemetry and packet telecommand. Now, CCSDS is setting out to provide new standards for the interchange of information, and the interconnection of subsystems and devices onboard spacecraft. This effort is known as the Spacecraft Onboard Interface (SOIF) activity. SOIF aims to publish standards that will:

- increase the potential for the reuse of spacecraft hardware and software,
- reduce the development time and effort for flight software,
- enhance the communication potential between onboard software components and with flight hardware items,
- open the way for more capable software architectures to be used onboard spacecraft,
- capitalise on existing terrestrial standards and expertise,
- enable significant reductions to be made to the mass, bulk, and complexity of the spacecraft harness.

This paper describes the work currently being performed by the CCSDS SOIF sub-panel, and the reference architecture that has been developed. Some of the services that SOIF provides, how they are used to transport data around a spacecraft, and their expected impact on onboard software development are described. From this we draw some inferences about the effect of SOIF on operational aspects.

2. The Goals of SOIF

The CCSDS P1K sub-panel was given the charter of standardising spacecraft onboard interfaces, and became known as the SOIF sub-panel. Clearly this is an enormous task, and identifying precisely which interfaces should be standardised was the first problem. Each member agency had its own strong views on what was most important, and it was eventually agreed that SOIF should initially concentrate on standardising a set of services used for onboard communication and access to onboard sensors and actuators, since these produced the most tangible and significant benefits in the short term. Subsequently, SOIF would turn its attention to the physical electrical interfaces for sensors and actuators, as well as defining more general onboard services that would further simplify the development of onboard software applications.

Because of its scope, the SOIF sub-panel is made up of experts from a variety of disciplines, ranging from flight software specialists, through communications and protocol engineers, to electrical and RF specialists. We therefore set out to partition the SOIF task using a reference model that enabled these experts to work most efficiently within their domain of expertise. Based on this reference model, we have established a number of special interest groups, each dealing with one key aspect. The SOIF reference model is shown in Figure 1 and described in the next section.

Again, because of its broad scope, SOIF may be seen differently by different user communities. Some of those user communities, and the way in which they will see SOIF, are as follows:

Flight application software developers will see SOIF as a set of services that are provided to ease their application development tasks, with these services being accessed through software calls to an application programming interface (API). SOIF isolates the application software developers from the specifics of the underlying hardware and the spacecraft configuration, which will increase the potential for application software re-use in the future.

Spacecraft platform developers will see SOIF as providing consistent specifications for standard services that must commonly be provided onboard the spacecraft. SOIF will therefore reduce the burden of developing specifications of platform services for each new spacecraft project.

Spacecraft instrument and component developers will see SOIF as providing the specifications for the electrical and communication interfaces through which they connect their products to the spacecraft. This will reduce the unnecessary diversity seen in instrument and component interfaces at present and should ensure that SOIF compliant equipment can be re-used across a variety of different platforms and missions.

3. The SOIF Reference Model

The SOIF reference model is layered according to the principles of the ISO OSI Reference Model, and is depicted in Figure 1.

The SOIF **application layer** contains user oriented services which are presented to SOIF user applications that reside outside of the model. Typically a SOIF user application is an onboard software entity that makes use of the SOIF services to access onboard sensors and actuators, as well as other onboard applications. The SOIF application layer combines the application and presentation layers of the OSI 7-layer model.

The SOIF **transport layer** contains services that enable end-to-end transfer of messages between users. The SOIF transport layer combines the transport and session layers of the OSI 7-layer model.

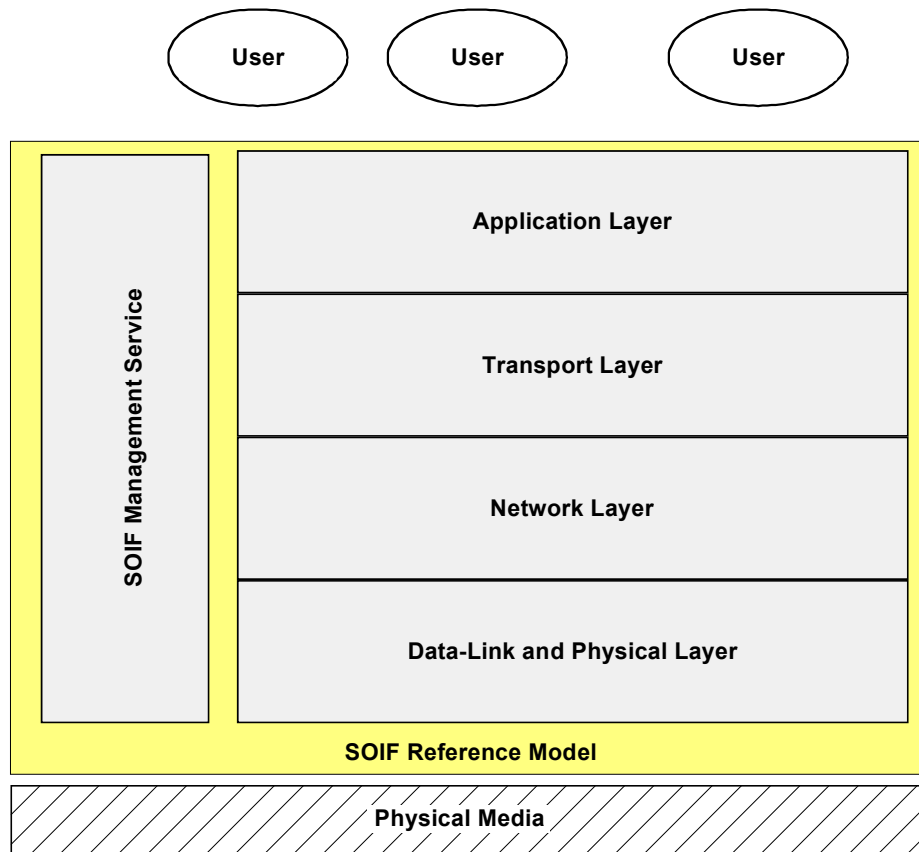


Figure 1 - The SOIF Reference Model

The SOIF **network layer** contains services that control the operation of the underlying sub-networks and enable data to be routed throughout the spacecraft network. This layer corresponds directly with the network layer of the OSI 7-layer model.

The SOIF **data link and physical layer** contains services that implement the onboard sub-network and interfaces to onboard sensors and actuators. Typically, onboard sub-networks comprise onboard buses as well as point-to-point links between flight units. The SOIF data link and physical layer corresponds to the data link layer and the physical layer of the OSI 7-layer model. However, while the OSI model is concerned only with communication networks and does not recognize sensor and actuator interfaces, the SOIF data link and physical layer explicitly includes sensor and actuator interfaces.

The SOIF **management service** provides the capability of managing the SOIF stack. Because this service controls the configuration of each layer, it is represented as a vertical slice spanning all of the layers of the reference model in accordance with accepted OSI convention.

4. SOIF Services

Having established the SOIF layers, these have now been populated with a number of services that we consider need to be provided onboard. These are shown in Figure 2.

We do not have space in this short introduction to SOIF to describe all of these services. However, two services will be described in more detail since they capture the spirit of SOIF and indicate how SOIF will influence different facets of the onboard systems in the future. These services are the command and data acquisition service, and the intra-networking service.

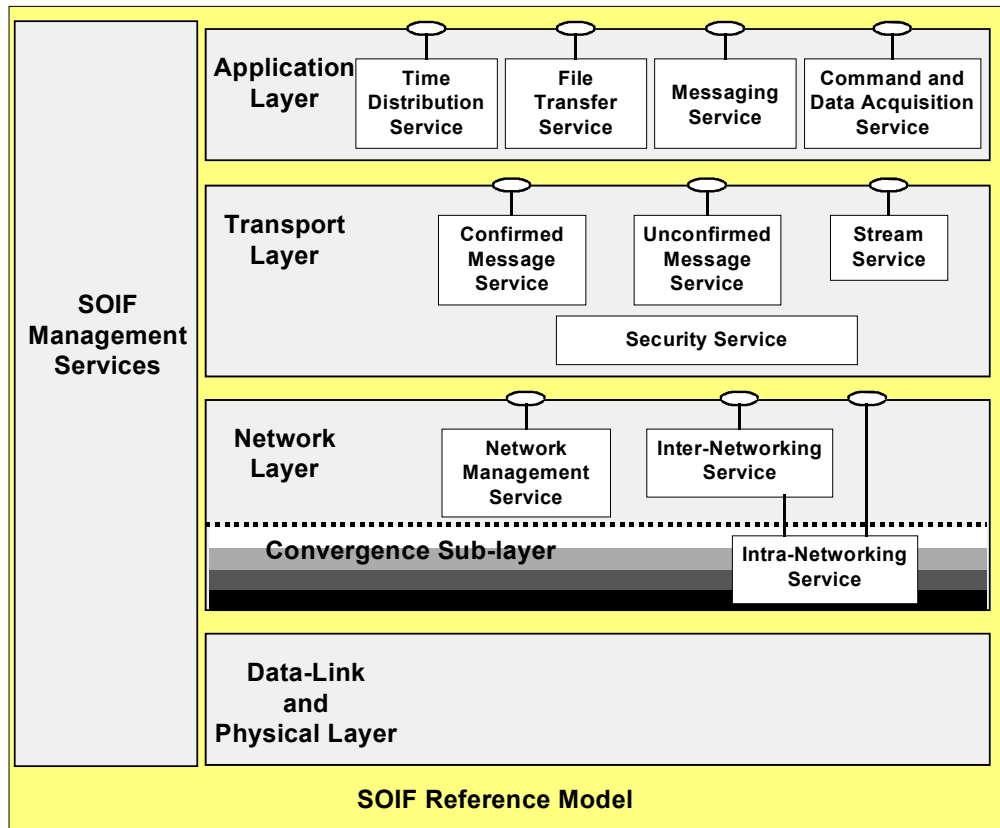


Figure 2 - SOIF Services

4.1. SOIF Command and Data Acquisition Service

The basic SOIF command and data acquisition service enables users to issue commands and retrieve data from sensors and actuators without requiring detailed knowledge of the hardware, the underlying interfaces, or the location of the sensors and actuators. In operation, the service allows users to read or write to a local representation of the sensor or actuator, and the service translates those read and write accesses into the operations that are needed to read or write the real device.

This basic functionality has been recognised as an enabling technology that allows other, more interesting capabilities to be provided to user applications. These include the conversion of raw values into engineering units, the

monitoring of parameters without the user applications involvement, and the implementation of virtual devices. These additional capabilities have now been documented as capability sets that may be provided within the SOIF command and data acquisition service, with the basic device read and write capability described in this paper being called *capability set 1*.

The command and data acquisition service increases the potential for software re-use by hiding the hardware specifics of sensors and actuators, and the underlying interfaces, from the flight application developers. In the short term, the service allows re-use of existing sensors and actuators while offering immediate benefits to the flight application development process. In the longer term, it is expected that sensor and actuator manufacturers will converge towards designs that are increasingly like the local representations, thereby requiring less translation and leading to more efficient implementations of the command and data acquisition service. This in turn will open up new opportunities for sensor and actuator manufacturers while at the same time giving spacecraft developers a wider choice of components.

4.1.1. Architectural Context

Figure 3 shows the command and data acquisition service within the SOIF architecture. The key points to note are:

- The command and data acquisition service resides in the application layer.
- Below the physical layer there are locally connected sensors and the spacecraft bus.
- Locally connected sensors are accessed via P-SAP's. These can be called directly from the application layer.
- The spacecraft bus is accessed via one or more T-SAP's or L-SAP's that can be reached directly from the application layer.
- The command and data acquisition service could use the SOIF messaging service, which also resides in the SOIF application layer.

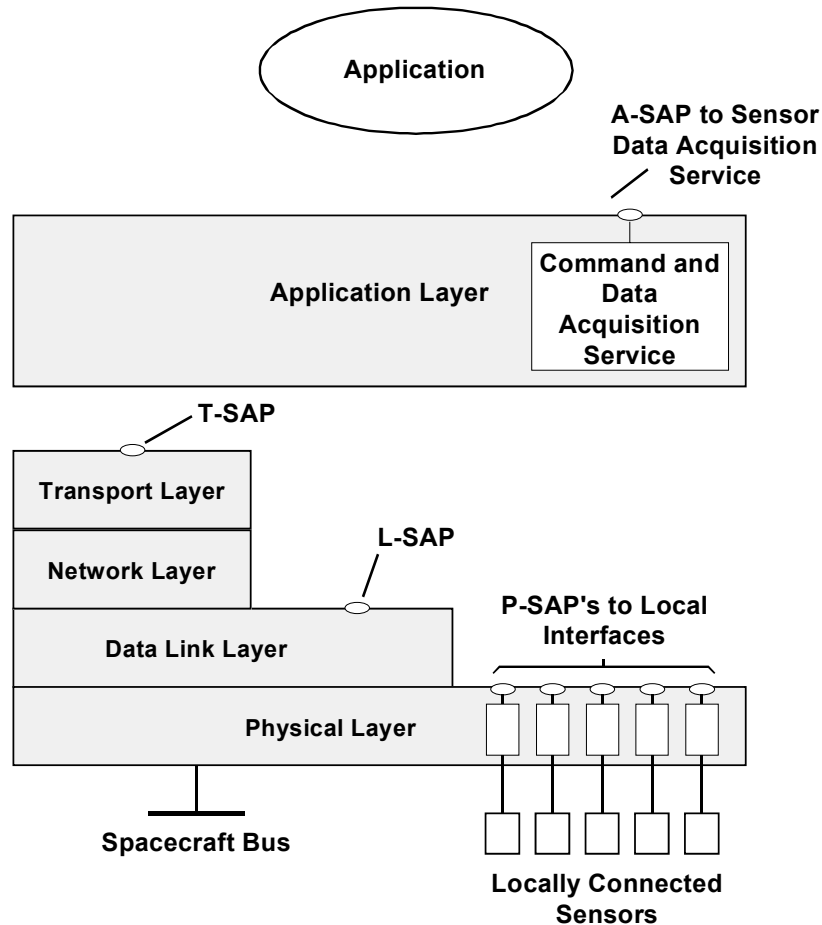


Figure 3 - Architectural Context of the SOIF Command and Data Acquisition Service¹

4.1.2. Service Operation – Service User Perspective

The command and data acquisition service user is an onboard application. Suppose that this application wishes to acquire data from an onboard sensor. In order to do this it issues a ReadDevReq, and provides the device ID as a parameter to this service request. The service responds some time later with a ReadDevInd either containing the requested data, or indicating an error.

The only information that the application must provide when it issues a ReadDevReq is the sensor identifier, which must be unique within the domain of the spacecraft. The user does not need to know the location of the sensor, or how it is physically accessed in the hardware.

In real implementations, there may be issues of timeliness to consider, particularly if the application is performing some real time control function.

¹ **Note:** In this diagram the data link and physical layers are shown as distinct.

However, providing a time constrained service is only a modification of the basic service described above. Also, in reality, critical applications will be aware of the location of sensors, and applications will most probably only expect real-time responses from locally connected sensors.

4.1.3. Service Operation – Service Providers Perspective

The function of the sensor data acquisition service is to acquire data from a specified sensor on receipt of a data acquisition request. We can immediately envisage three possible scenarios based on the location of the sensor:

1. The sensor is locally connected.
2. The sensor is connected directly to the spacecraft bus.
3. The sensor is connected to a remote node that can be reached via the spacecraft bus.

In the first case, the sensor data acquisition service can simply read the sensor via the appropriate local P-SAP. This is the simplest case.

In the second case, the command and data acquisition service accesses the sensor directly via the spacecraft bus by issuing the appropriate request through the L-SAP. If the spacecraft bus is Mil. Std 1553B for example, this might involve issuing a command word and acquiring the response. The L-SAP can be used because there are no transport or network layer services required, the bus address of the sensor must be known by the sensor data acquisition service, and the appropriate command can be easily formatted. The medium access control services of the data link layer are required to avoid conflicts on the bus, hence we need to use an L-SAP rather than a P-SAP.

In the third case, where the sensor is connected to another node of the onboard network, the local command and data acquisition service must transfer the acquisition request to its counterpart in the node to which the sensor is connected. I.e. if the service user is hosted on node A, and the sensor is connected to node B, the command and data acquisition service in node A sends a message to its counterpart in node B, indicating the sensor from which data is to be acquired. The command and data acquisition service in node B then acquires the requested data and sends it back to node A in another message. Finally, the command and data acquisition service in node A responds to the service user.

In this case, we require at least the SOIF network layer services to transfer the message between the command and data acquisition service entities, and possibly also transport layer services. Therefore, a T-SAP could be used for this exchange. More recent work on the SOIF messaging service that resides in the application layer suggests that it may be better to use the messaging services rather than the transport layer services directly. However, this is largely an implementation issue.

4.2. The Intra-networking Service

The SOIF intra-networking service provides the capability of transferring an octet aligned, variable length, delimited data unit across an underlying bus. The implementation of the SOIF intra-networking includes functions such as segmentation and encapsulation for the underlying bus, and is therefore entirely bus dependent, however all users of the intra-networking service are entirely bus independent. This concept is illustrated in Figure 4. The intra-networking service access point therefore represents the convergence point above which any protocol can be used, and below which any bus can be used. This therefore achieves one of the stated aims of SOIF which is to support a variety of higher layer protocols over any underlying bus.

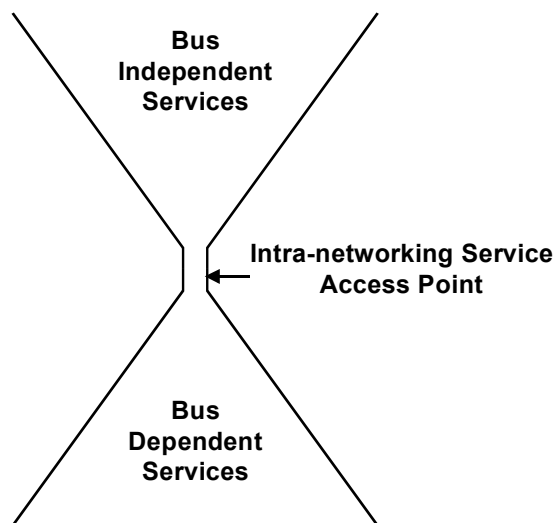


Figure 4 - Hourglass Model showing Bus Dependent and Bus Independent Services

At present, we envisage a variety of different data units being transferred via the intra-networking service, including PUS packets, IP datagrams, SCPS-NP datagrams, CCSDS encapsulation packets, and so on. We expect this list to grow in the future. To ensure that these different data units are handled correctly at the receiving end, the service explicitly transfers the protocol identifier with the data unit.

The service provides two grades of service, expedited and non-expedited transfer. Expedited transfers are handled with priority over non-expedited transfers, which enables high priority data such as urgent commands to be transferred in a timely fashion.

The current status of the definition of the intra-networking service is that both the abstract service interface definition, which describes the way the service is accessed, and the abstract service definition, which describes the capabilities of the service, are in draft, and are expected to be finalised by end 2002. Implementations of the service are being prototyped for Mil. Std 1553B and SpaceWire, and this will be extended to CAN bus in the near future.

4.2.1. Benefits of the SOIF Intra-networking service

The SOIF intra-networking service offers two very important benefits for future spacecraft implementations. Firstly, it enables data to be transferred across any underlying bus. This means that spacecraft applications can be developed without needing to know the details of the underlying bus, and those applications will then be portable across different spacecraft platforms that use different buses. This in turn simplifies application development and increases the potential for software re-use.

Secondly, the intra-networking service allows a variety of high level protocols to be used across any given spacecraft bus. The intra-networking service is thus an enabling technology that will allow designers to choose the most appropriate protocol suites for future applications. For example, the intra-networking service can be used to transfer IP datagrams across the spacecraft bus, enabling flight software to use the TCP/IP communications stack. However, the service could just as easily be used to transfer PUS packets without the need of any other high layer protocols. Furthermore, the intra-networking service can be implemented in such a way as to allow different protocol data units to be mixed and interleaved on the underlying bus, enabling, for example, TCP/IP and PUS based applications to operate on the same node.

We believe that this freedom of choice of overlying protocols will have a profound influence on the software architectures that can be used in future spacecraft.

5. Operational Implications of SOIF

It is expected that the adoption of the SOIF recommendations will significantly reduce the effort, cost, and time required to implement the flight element of a space mission, and will result in more capable and reliable onboard systems being developed.

Moreover, SOIF is an enabling technology for the use of more capable onboard software architectures and affords better connectivity between flight units. Consequently, SOIF can be expected to have a significant impact on the operational aspects of space missions. For example, more capable, distributed onboard software will increase the potential for spacecraft autonomy, thereby reducing the volume of control traffic required between the ground and the spacecraft as well as changing its nature to goal directed commanding. If the software architecture supports application relocation and replacement, this will call for new flight software configuration management techniques.

At the same time, the increased onboard connectivity provided by SOIF will increase the ability to perform contingency operations from the ground. For example, a single onboard application could be used during contingency operation to reach any onboard device via the command and data acquisition service.

Finally, the SOIF communication protocols will improve the end to end connectivity between the ground and the spacecraft enabling payload operators and experimenters to control their onboard equipment using familiar user interfaces, for example based on internet technology.

6. Conclusions

Initially, SOIF will influence the spacecraft development process by reducing the time and engineering effort required to implement the spacecraft flight element, and by increasing the potential for re-use of both hardware and software components.

In the longer term, SOIF will open the way for more capable flight software architectures, for example based on the object-messaging concepts that are already prevalent in terrestrial applications. In the future, this will in turn influence the way spacecraft are operated, and how the flight software is managed during the mission.