RequirementsExperts

Training and Services for Project Success

Everything you wanted to know about interfaces, but were afraid to ask

Lou Wheatcraft

Presented at INCOSE 2010 July 14, 2010

Recent Email:

- I am looking into methodologies for a process that supports interface control requirements development with a particular emphasis how one set-ups and maintains traceability analysis of the interfaces.
- There seams to be little or no write-up in the literature on this.
  - Most text books discuss ICD working groups but none address the traceability analysis which could be complex.

Objectives

Help you to understand:

- What constitutes an interface
- How to identify interfaces
- How to define and document interface definition
- What constitutes a good interface requirement
- Where and how to document interface requirements

Help you to write better interface requirements.

Why are Interfaces Important?

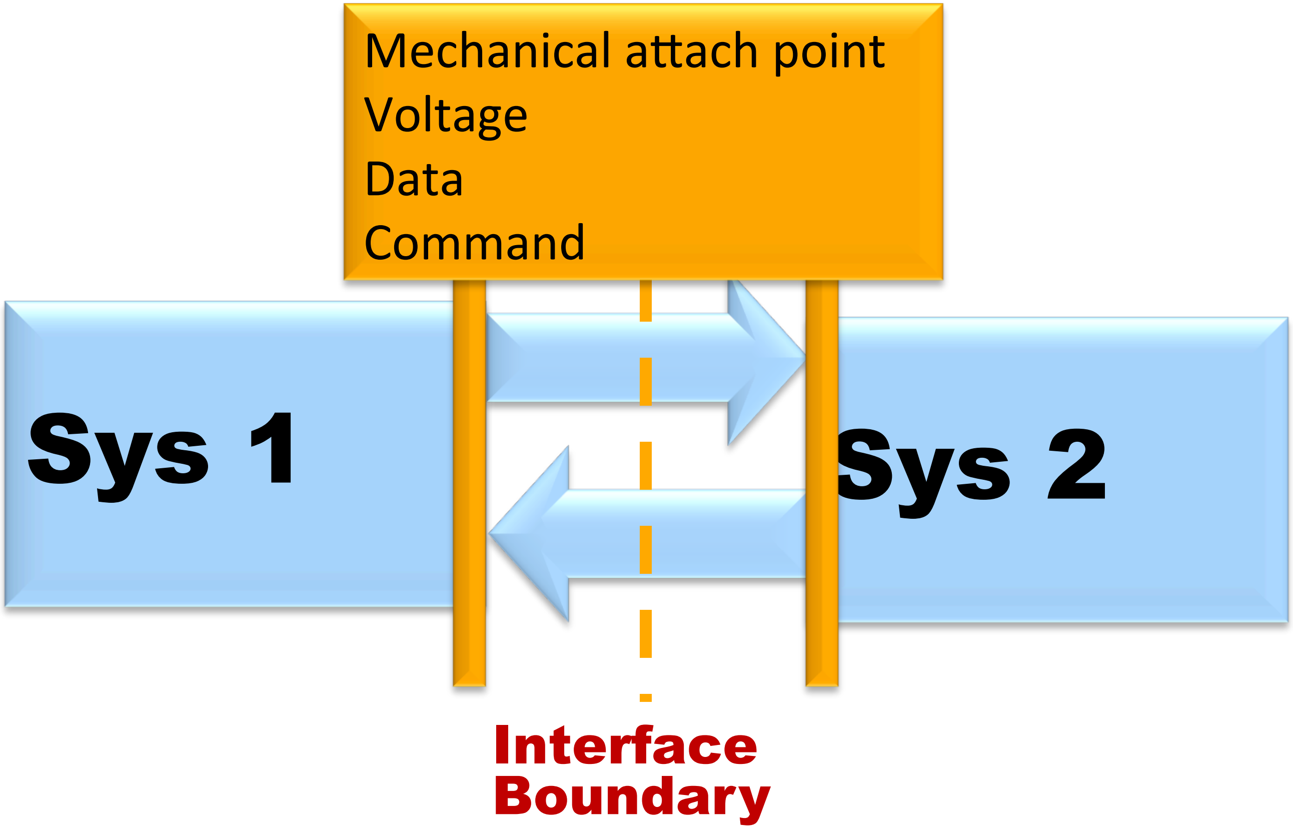
- Define your system's boundaries
- Understand the dependencies your system has with other systems and dependencies other systems have with your system
- Ensure compatibility between your system and other systems you need to interact with
- Provide a source of system requirements
- Expose potential problem areas and risks to your project
  - Missing or incorrectly defined interfaces are often a major cause of cost overruns and product failures

Because of the importance of identifying, defining, developing interface requirements, and managing these activities, interfaces need to be a prime concern of the project Systems Engineer

- What is an Interface?

What is an Interface?

A common functional or physical boundary where (across which) two systems interact.



Steps for managing your interfaces

1. Identify the interfaces
2. Define the interfaces
3. Write interface requirements

1. Identify the interfaces

Major Types of Interfaces

- Physical interfaces
  - Mechanical
  - Electrical
  - Electronic
  - Power
  - Propulsion

- Functional interfaces
  - Information transfer
  - Computer-human
  - Maintenance
  - Installation

- Environment*

Major Categories of Interfaces

- Physical
- Electronic
- Electrical
- Hardware/Software
- Software
- Environment
- Human/Machine

Techniques to identify the interfaces

- Operation Concepts
- System Block Diagrams
- N-Squared (N2) diagrams/I-Squared (I2) diagrams
- Allocation Analysis
- External Interface Block Diagrams (Context Diagrams)
- Interface Block Diagrams
- Models

2. Define the interfaces

Interface Definition

- To define an interface both sides must know:
  - characteristic of the thing crossing the interface
  - the characteristics of each system at the interface
  - the media involved in the interaction
- Focus of the definition statement is on one of these
  - not the systems that are involved in the interaction
- Documented in IRDs, IDDs, IADs, ICDs, etc.
- I will use the generic term “DefDoc” for the document used to document the interface definitions.

ICD: Interface Control Document
IRD: Interface Requirement Document
IDD: Interface Definition Document
IAD: Interface Agreement Document

Interface Definition General Format

- [Thing being defined] [will be/is] [whatever the definition is].
  - The communication protocol will be USB 2.0
- [Thing being defined] [will be/is/are] as shown in [Drawing xxxxxx] or [Figure yyyyyy].
  - The mechanical attach points are as shown in drawing xyz.
- [Thing being defined] [will have/has] the characteristics shown in [Table or Graph zzzzzzzz.]
  - The 28 VDC voltage has the characteristics shown in Table 5-1.

Written as a statement of fact – no "shall"

Form depends on whether the definition statement is written before or after design.

Terminology Clarification

- Document names used for interface definition documents:
  - IRD, IDD, IAD, IDAs, Technical Data Package, etc. pre-design definitions and for interface definitions between systems being concurrently developed.
  - ICD post-design, as built definitions and for interface definitions for existing systems
- Some organizations use the an "evolutionary ICD" to document interface definitions
  - The ICD begins with information described earlier in the Interface definition documents for pre-design definitions
  - The ICD evolves to include additional design details from each side of the interface as the design evolves concluding with the as-built configuration
- An organization should be consistent in terminology.

3. Write Interface Requirements

Writing an Interface Requirement

in your Specification

WRONG

- The xyz system shall provide power to my system.

- My system shall interface to the xyz system.

RIGHT

✦ My system shall obtain 28-volt power from the xyz system per ICD 1234 table 2.

Interface Requirement General Format

[My System] shall [use/provide something] [from/to Another System] having the characteristic defined in [Document and section of the document where the something is defined]"

The Payload shall send to the Spacecraft Sensor A data having the characteristics defined in DefDoc #1234, Table TBD.

[My System] shall [interact] [with Another System] as defined in [Document and section of the document where the interface is defined]"

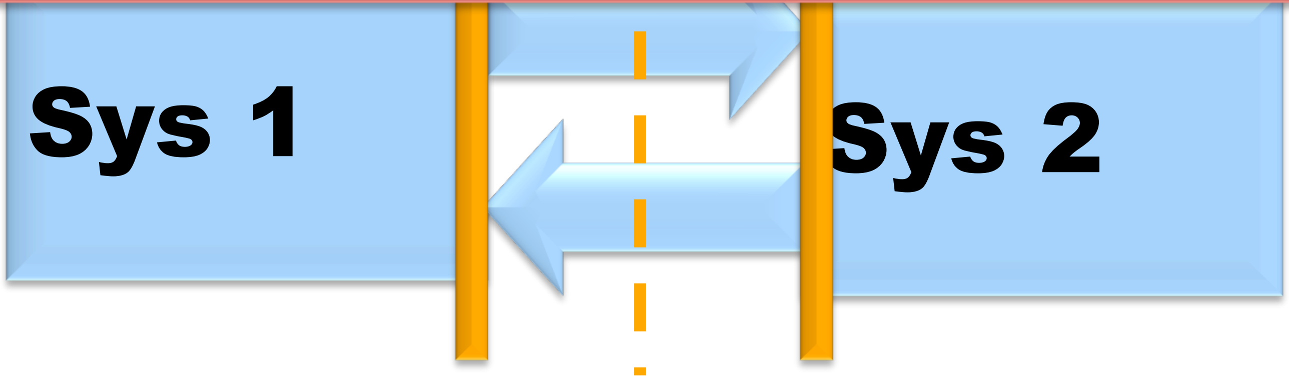
The Payload shall communicate with the Spacecraft processor via a 1553 bus as defined in DefDoc #1234, Figure 6.

Interface requirements are documented in your System Requirement Specification pointing to the common interface definition in DefDoc.

Interface Requirements Between Two Developing Systems

Sys 1 Specification	DefDoc #1234	Sys 2 Specification
Sys 1 shall transmit the [xyz] command defined in DefDoc #1234 Table 2-2 within 5 ms of a MCDS cmd.	***The xyz command [will have] [has] the characteristics described in Table 2-2.***	***Sys 2 shall execute*** **the [xyz] command defined in *DefDoc #1234* Table 2-2 within 4 ms of receipt.**

DefDoc DOES NOT contain a shall statement.



DefDoc – Document where you define the interface boundary

- Interface Design Guidlines

Basic Design Guidelines

- Minimize the number of interfaces
- Simplify all interfaces
- Consider commonality, compatibility, and interchangeability requirements.
- Assess interface risks
- Do you need to protect or control access to your data?

Be sure to ask the question, "What is the worst thing that other elements could do to you across the interface?" DEFEND AGAINST IT

- Closing Thoughts

Closing Thoughts

- Identify all interfaces – early!
- Assign responsibility.
- Obtain copies of all interface documentation.
  - If the interface definitions are not documented, document them!
  - If a new interface, negotiate interface definition, and document those definitions in a common, configuration controlled document.
  - Make all interface documentation available to everyone in your organization as well as to the other systems you are interfacing with.
- Involve interface stakeholders in your project.
- Include interface requirements in your system's requirement document.
- Flowdown (allocate) your interface requirements
- Provide traceability between interface requirement pairs, to parents, and to the common definitions.
- Track/monitor interface change
- Follow the interface design guidelines.
- Plan ahead for interface verification and compliance testing.

References

- CAI 2009, *Requirements Development and Management, Seminar Workbook, October 2009,* Compliance Automation, Inc.
- Grady, J. O., *System Requirements Analysis, Academic Press, 2006*
- Hooks, I. F. & Farry, K. A. 2001, *Customer-Centered Products: creating successful products through smart requirements management; AMACOM Books, NY, NY, 2001.*
- INCOSE 2007, *Systems Engineering Handbook - a guide for system life cycle processes and activities, Version 3.1, INCOSE-TP-2003-002-03.1 , August 2007, ed, Cecilia Haskins.*
- NASA, Training Manual for Elements of Interface Definition and Control, NASA Reference Publication 1370, January 1997, ed, Vincent R. Lalli, Robert E. Kastner, and Henry N. Hartt
- NASA, *System Engineering Handbook, SP-2007-6105 Revision 1, December 2007.*

Biography

- Lou Wheatcraft has over 40 years experience in the aerospace industry, including 22 years in the United States Air Force. Over the last ten years, Lou has worked for Compliance Automation, where he conducts seminars on defining scope, writing good requirements, and managing requirements for NASA, DoD, and industry. Lou has a BS degree in Electrical Engineering, an MA degree in Computer Information Systems, an MS degree in Environmental Management, and has completed the course work for an MS degree in Studies of the Future, Lou is a member of INCOSE, co-chair of the INCOSE Requirements Working Group, a member of PMI, the Software Engineering Institute, the World Futures Society, and the National Honor Society of Pi Alpha Alpha.