26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

A Representative Application of a Layered Interface Modeling Pattern

Peter Shames, Marc Sarrel,
Sanford Friedenthal (Affiliate)
Jet Propulsion Laboratory, California Institute of Technology

Introduction

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Interfaces define
  - How a system interacts with its environment
  - How the components of a system interact
- Interfaces are specified in many domains
  - Electrical, mechanical, thermal, fluid, human, data
- System engineers must specify, analyze and verify this range of interfaces
- In addition, individual interfaces can be very complex
  - Specification for USB 2 is 650 pages
- Model Based Systems Engineering offers a method to support accurate interface specification, and allow common representation across disciplines

Layered Interface Modeling Approach

- Leverage layered interface concepts used to define communication interfaces (e.g., OSI stack)
  - This layered ISO interface model has functioned well as the basis for the Internet for decades
- Specify the interface in terms of what is exchanged
- Realize the interface by transforming the exchange from application layer to physical layers
- Define interactions between peer layers of the interacting components, and between vertical layers of each component.

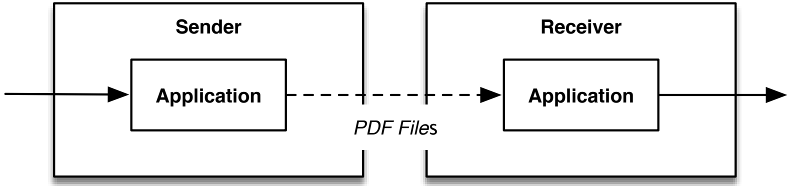
Why Layered Interfaces

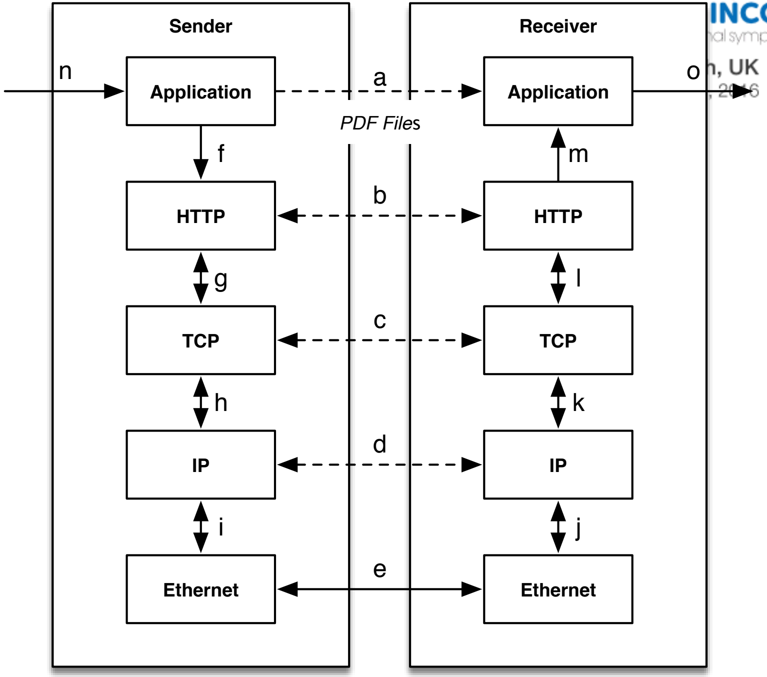
- Separation of concerns
- Each layer of the interface describes it's own functionality and addresses its own set of concerns
- Each layer may be considered separately, or in combination with others
- Layers are independent of each other, and can be combined in permissible ways
- All layers must function correctly for the interface to work as a whole.

Why Layered Interfaces (cont.)

- A rich and flexible representation in the model allows construction of many different views
  - Different interface abstractions are provided, from logical flows to complete protocol stacks
  - End-to-end data flows, connectivity, and data transformations
  - Physical connections
  - Protocol specifications
  - Message definitions
  - Complete interface specifications that span discipline concerns
- The generation of these views from a single model ensures their consistency.

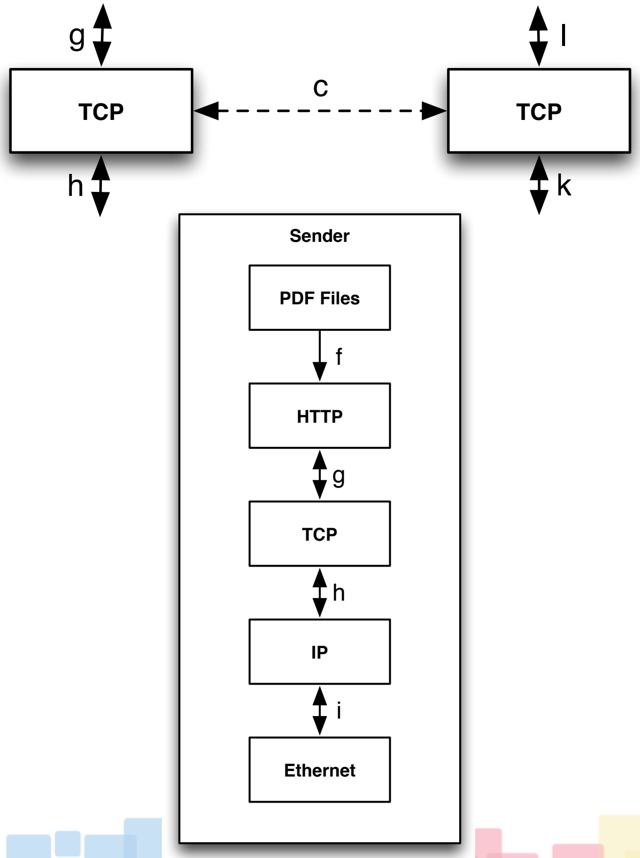
A Simple Example





- The simple view is "boxes and lines"
  - What are the interfaces?
  - What are the protocols?
- End-to-End simple view
  - Send a PDF file from A to B.
  - This is the requirement, what the user sees.
- Interface Specification
  - Implemented with HTTP, TCP, IP and Ethernet.
  - The protocol stacks in each component are connected both horizontally and vertically.
- Simple lists of interface protocols are just not sufficient to understand the architecture.

Selection of Focus



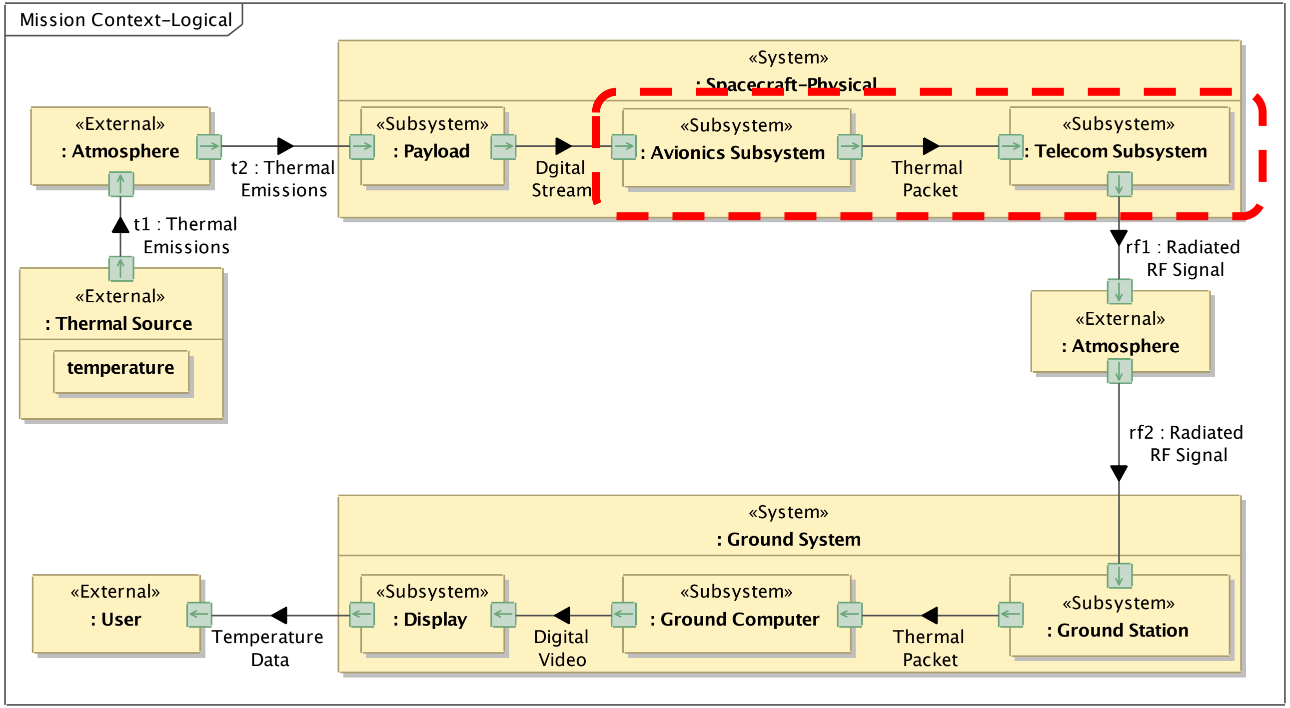
- We can focus on just the TCP layer.
  - How it is connected (horizontally).
  - How it behaves (horizontally).
- We can focus on just the stack.
  - How it is connected (vertically).
  - How it behaves (vertically).
- Data logically flows across the horizontal layers, the TCP spec describes the behavior of the peer protocol entities.
- Data actually flows "down the stack" through each successive layer until it gets to the physical layer where the "real" connections occurs.
- "On the wire" the whole stack is visible.

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

Space Data System Example

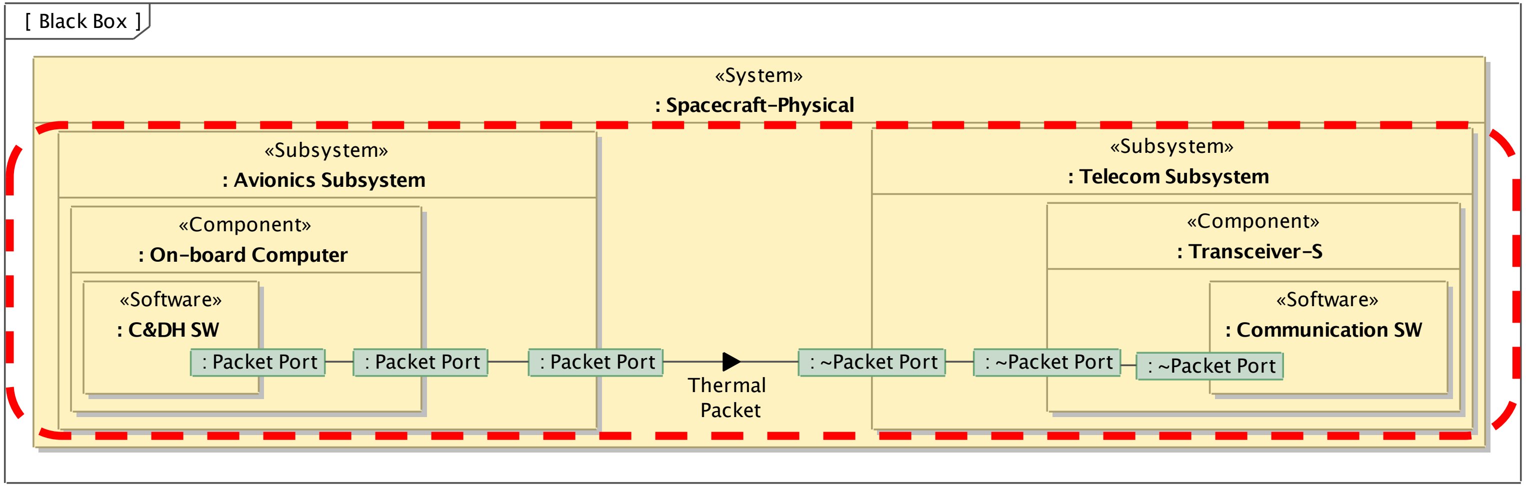
- Space data systems are composed of a flight segment and a ground segment.
- End-to-End Information Systems track data within each segment, and between the segments as an integrated flow.
- Systems include both mission-specific and shared multi-mission resources.
- So, how to accurately describe, model and characterize these system and their interfaces?

End to End Information Flow



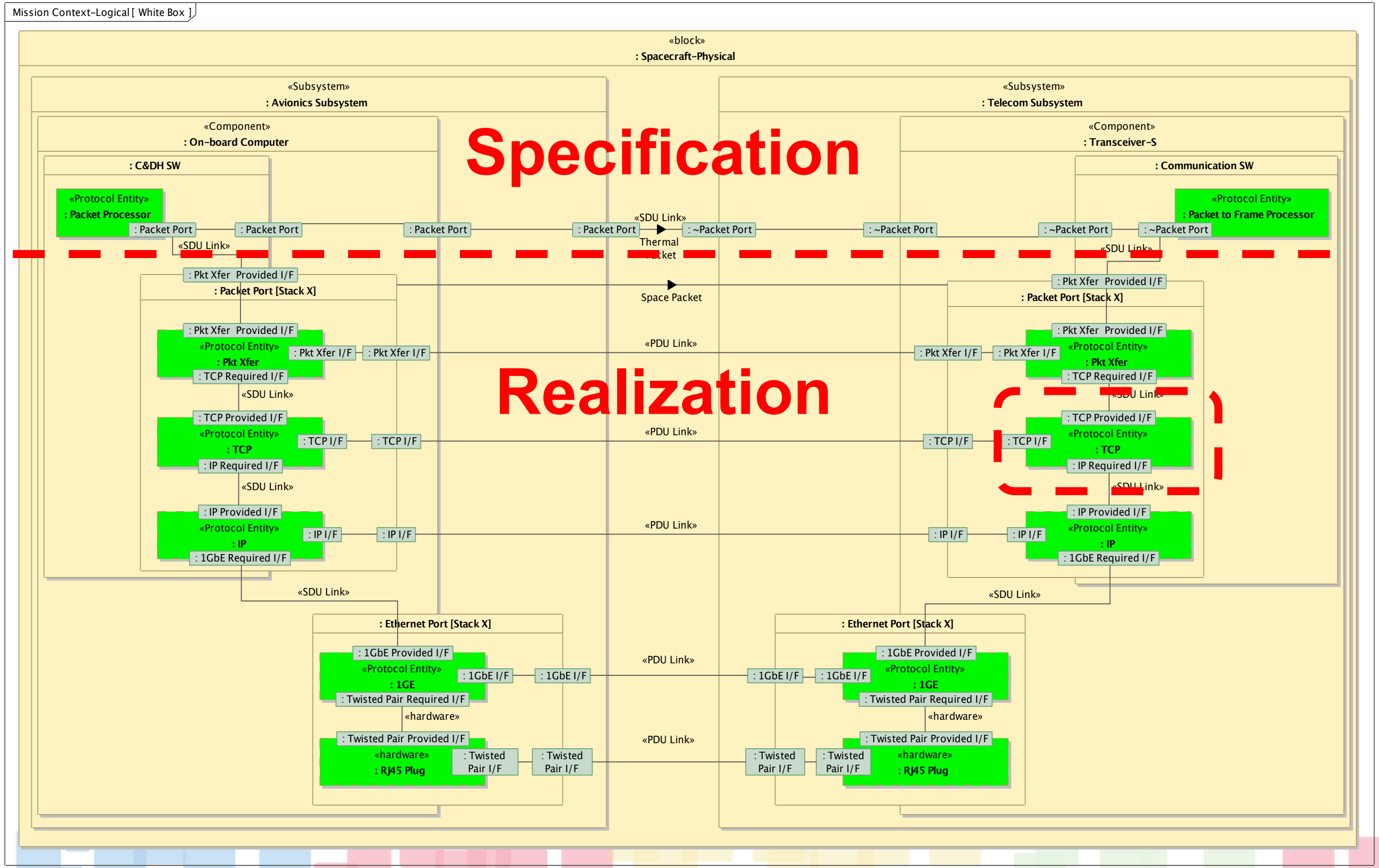
- End to End view of whole context
- Major physical element
- Coarse-grained decomposition
- Elides organizations, ownership and operational details
- Does not show interface or protocol details yet…

Subsystem Interface Specification



- Shows Avionics and Telecom subsystems from End-to-End view
  - Shows decomposition, component and hardware-software relationships
- Does not show interface details or protocol stacks

Subsystem Interface Realization



26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Details of protocol stack
- Ownership split among components between IP and Ethernet layers
- Interface Binding Signature

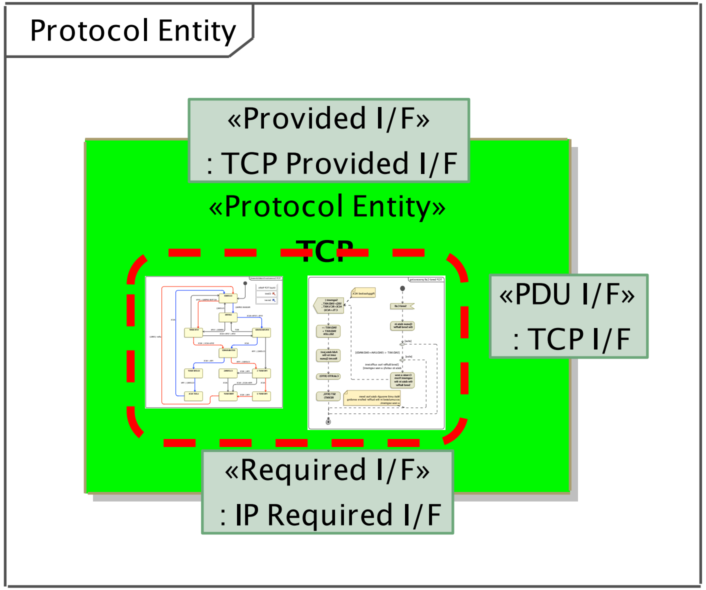
Typical Communication Interface Layers

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

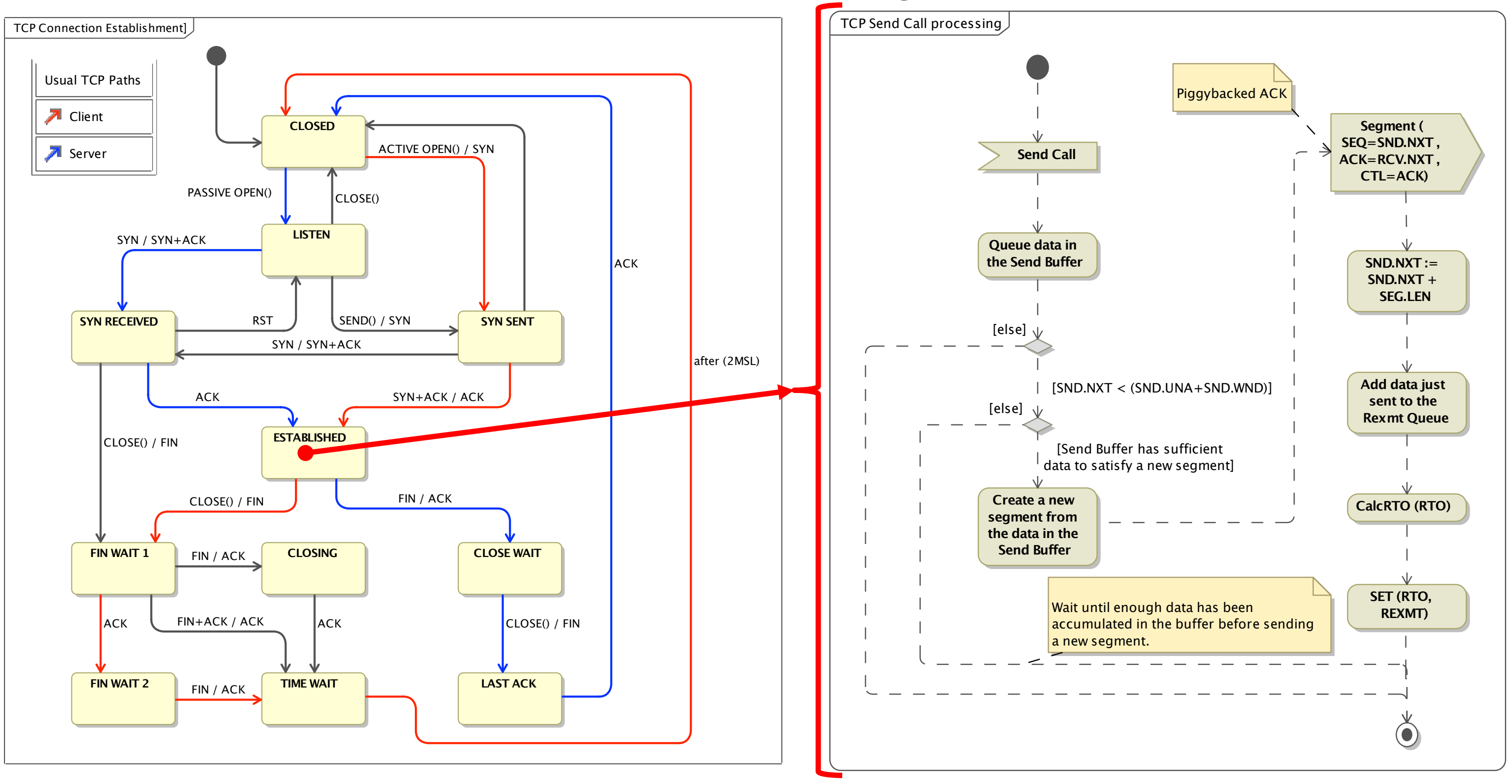
- **Application layer:** packet transfer protocol, manages exchange of packet data between applications.
- **Transport layer:** Transmission Control Protocol (TCP), provides end-to-end, once only, in order, complete delivery of data.
- **Network layer:** Internet Protocol (IP), provides network layer routing over any number of intermediate network nodes.
- **Data link layer:** 1 Gb Ethernet, provides data link layer services that may involve a fabric of switches and hubs.
- **Physical layer:** twisted pair cable (Cat-5) and RJ-45 plug terminations.

Protocol Entity

- Every protocol entity at layer (N) has three ports:
  - the interface that provides services to the upper (N+1) layer
  - the interface that requires services of the lower (N-1) layer
  - the interface with the peer protocol entity at the same layer
- There may also be a management interface, which can be in-line or separate



Protocol Entity Behavior



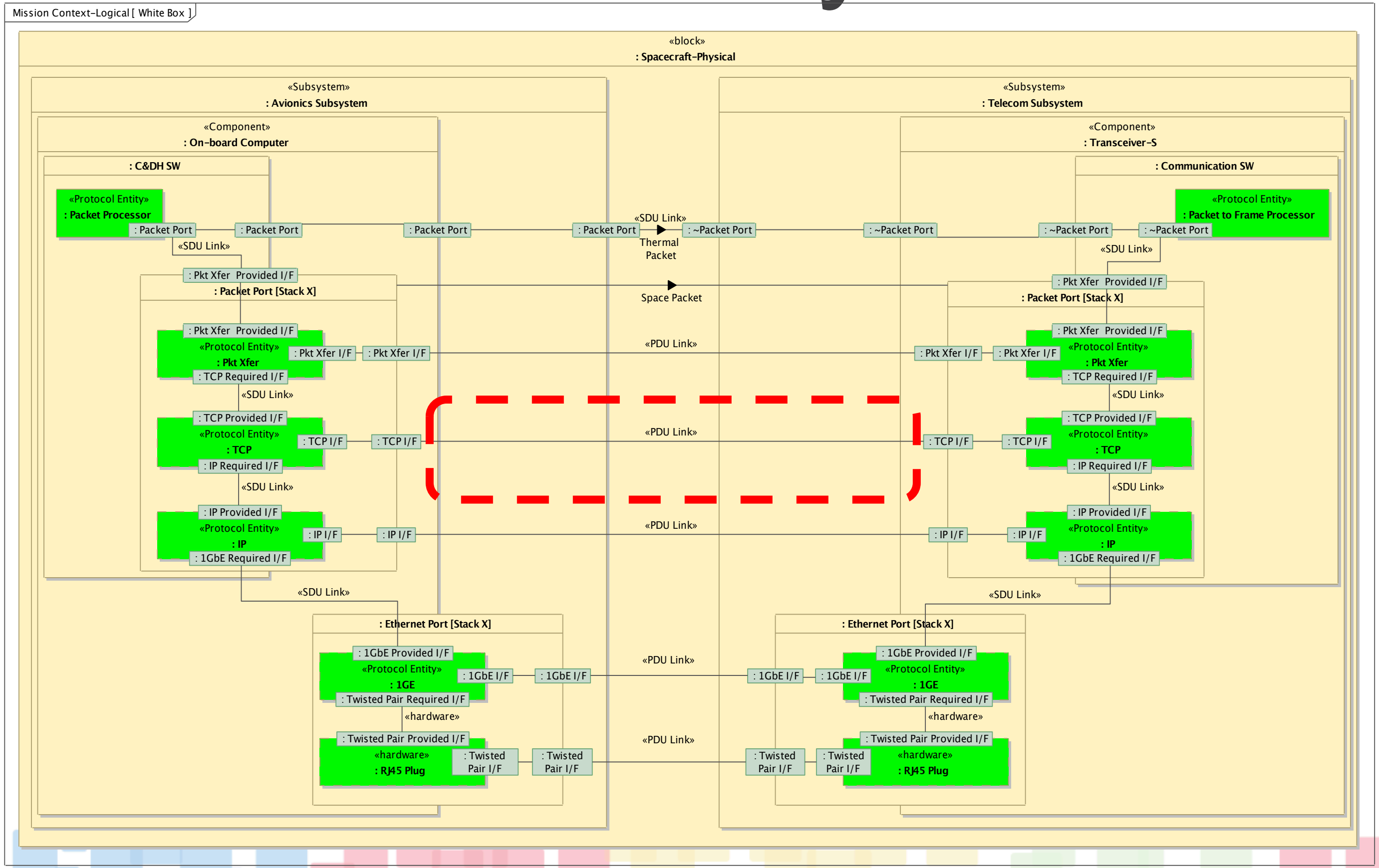
- Activity on right is one of many contained in the states on the left

Protocol Entity Behavior (cont.)

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Describes how a protocol entity behaves when receiving a Protocol Data Unit (PDU) from a peer entity
- Describes the exchange(s) of PDUs between peers
- May describe the behavior at the required and provided interfaces, such as start-up, connection establishment, and Service Data Unit (SDU) transformation into PDUs
- Typically involves describing the dynamics of PDU exchanges, including nominal and error conditions

Protocol Entity Interaction

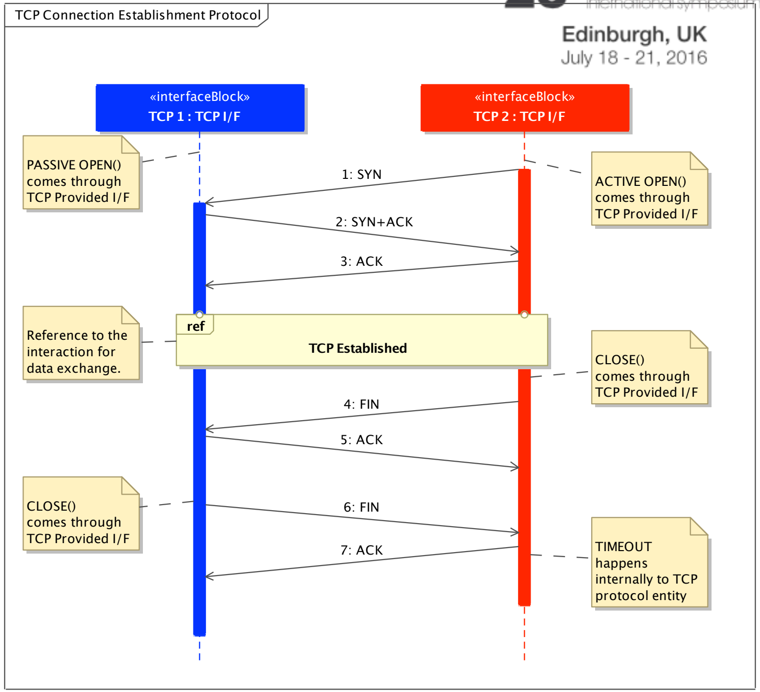


- Constrains the behavior of peer protocol entities

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

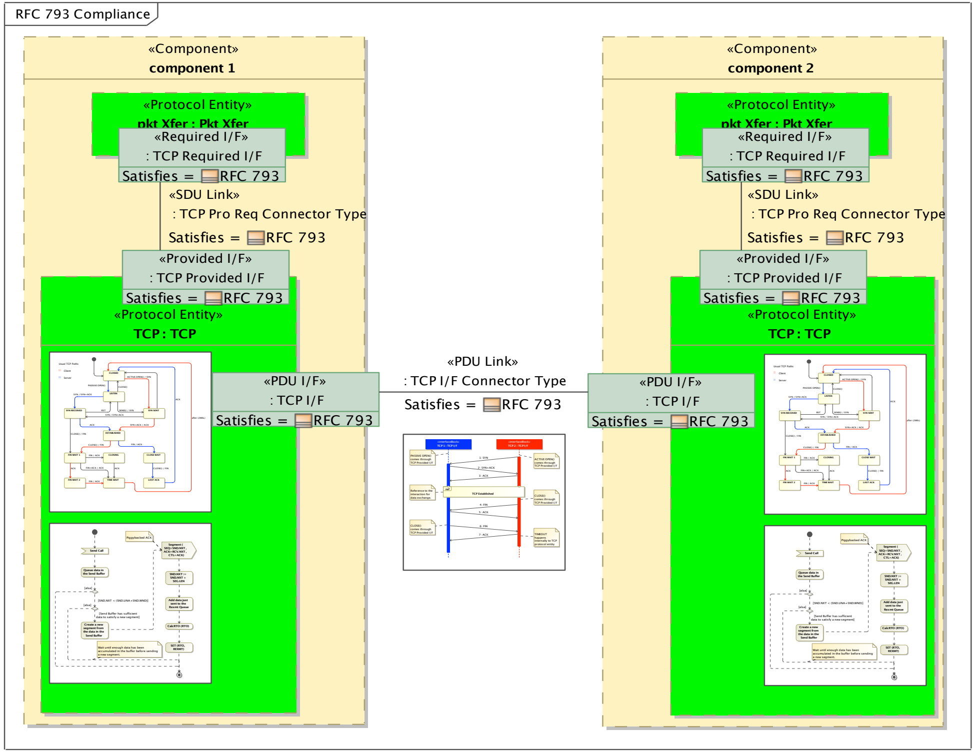
Protocol Entity Interaction (cont.)

- Specifies allowable interactions between peer protocol entities
- Keeps the peer state machines synchronized
- Describes the PDU exchanges you would see on the wire for a single layer



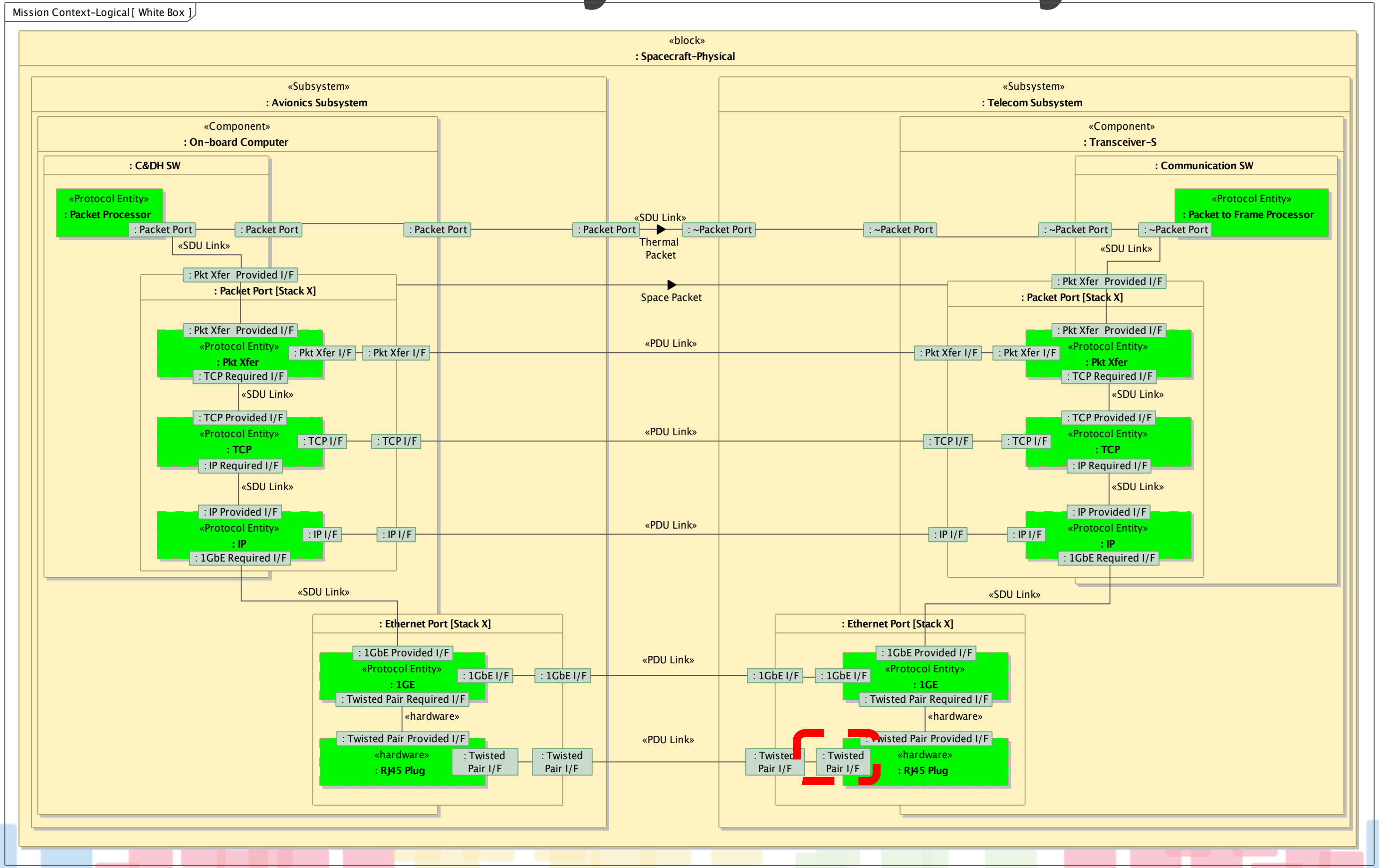
26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

Behavior Context



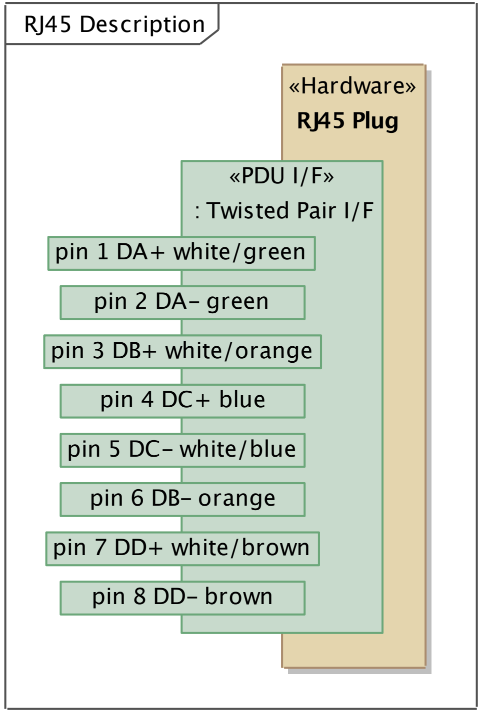
- Shows all behaviors in context
- Shows compliance with RFC 793 spec
  - Note that Pkt Xfer entity also complies with RFC 793 spec at the Required Interface
  - Interface and behavior
- Behavior on vertical SDU links typically implementation specific, i.e. not specified in standards

Interface Physical Layer



26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

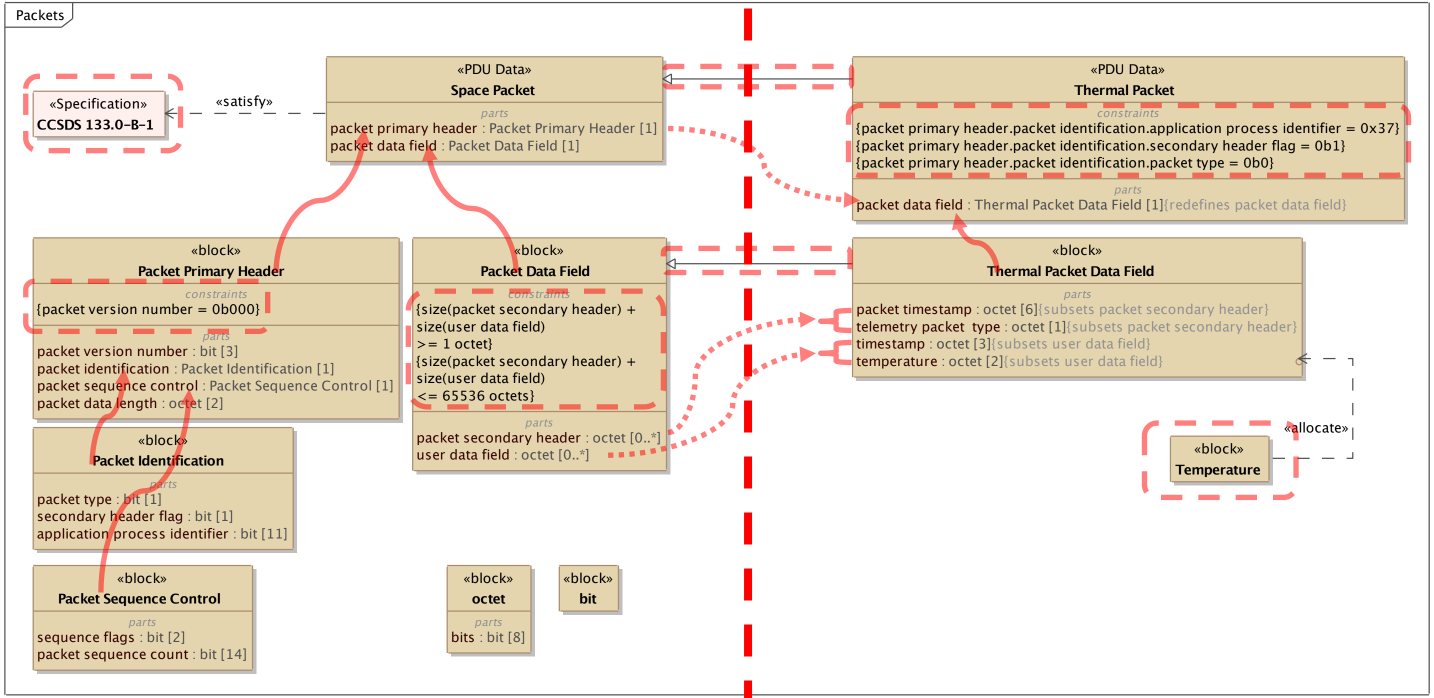
Electrical Connection



- Interfaces may be further decomposed
- Get as specific as your problem demands
- Physical layer interfaces may be governed by constraints based on physical laws
  - Ohm's law, Kirchhoff's law, laws of thermodynamics, etc.
  - Instead of activities and state machines

Data Packet Structure

Generic Space Packet | Specific Thermal Packet



- Shows how packets conform to standard
- Left side is generic space packet
  - Arrows show typing
  - Boxes show constraints
- Right side is specific thermal packet
  - Show adaptations (redefinition, subsetting and new constraints)

Methodology Summary

- Nested ports to capture interface details that a component presents
- Specification of stacks allows components to be explicit about what combinations of protocols, and in what order, they support
- Reference properties for protocol entities in ports allow separation of protocol implementation from all the contexts in which it’s used
- Power of the method depends on rigor of model content, rather than visual presentation on SysML diagrams

Summary

- Layered Interface Modeling Approach provides a means to model complex interfaces
- May be elaborated to include additional details as needed
- Ties to standards, and subsections of standards
- Compliance of data structures, behavior and physical laws
- Layered Approach
  - Specification vs Realization
  - Layers defined by protocol entities that have peer layer interactions and layer to next layer interactions.

Summary (cont.)

- Allows construction of full set of consistent views at desired level of detail
- Development of library components allows re-use of common protocols and standards
- Only model what is needed at any given point

Acknowledgements

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

The trade studies that first defined and used this method were chartered and funded by the NASA Space Communications and Navigation (SCaN) Program Office, with strong management support from Phil Librecht and encouragement to adopt OMG SysML™ modeling from Jim Schier. The authors would also like to acknowledge the excellent leadership and guidance of the trade study tasks provided by the co-leads: Wallace Tai from JPL and Nate Wright from GSFC.