Rubber To Road

Practicing MBSE as an Empowered SE

Bjorn Cole
Systems Engineer,
Jet Propulsion Laboratory /
California Institute of Technology
Chair, Space Systems WG
bjorn.cole@jpl.nasa.gov

INCOSE
International Council on Systems Engineering

ACT ONE: THE PROBLEM

JPL-identified problems in SE

1. Mission complexity is growing faster than our ability to manage it

   …increasing mission risk from inadequate specification & incomplete verification

2. System design emerges from the pieces, not from an architecture

   …resulting in systems which are brittle, difficult to test, and complex and expensive to operate.

3. Knowledge and investment are lost at project lifecycle phase boundaries

   …increasing development cost and risk of late discovery of design problems.

JPL IOM-3100-09-040, "The Need for an Integrated Model-Centric Engineering Environment at JPL", 24 Oct 2009, Todd Bayer

JPL-identified problems in SE

1. Knowledge and investment are lost between projects

   …increasing cost and risk; damping the potential for true product lines

2. Technical and programmatic sides of projects are poorly coupled

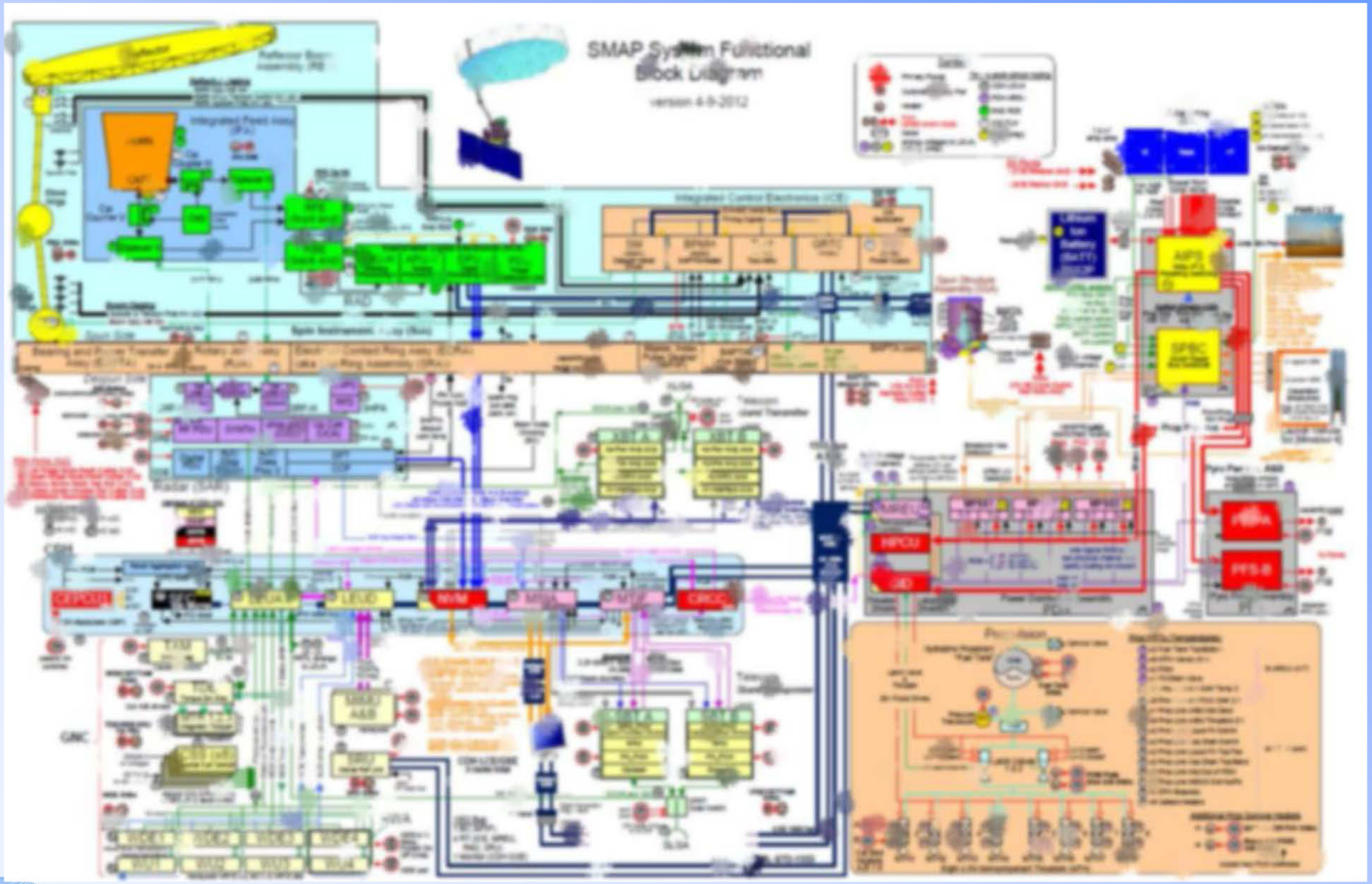
   …hampering effective project decision-making; increasing development risk.

JPL IOM-3100-09-040, "The Need for an Integrated Model-Centric Engineering Environment at JPL", 24 Oct 2009, Todd Bayer

A finer point on it

- Our products are limited
  - Dead, dumb, flat documents
- The computing work metaphor of the desktop is *totally wrong* for systems engineering!
  - Gotta get out there, be live on stage, work together
- In other words, we are succeeding in spite of our infrastructure, not because of it

The world through a drinking straw



INCOSE
International Council on Systems Engineering

The world through a drinking straw

Database of Architectures					
System Trades	**Units**	**Arch 1a Solar**	**Arch 1a**	**Arch 2a**	**Arch 2b**
System 1 Architecture	**-**	**-**	**-**	**-**	**-**
SC_Analogue1	-	Juno	Juno	Juno	Juno
Leveraging Type	-	T	T	TRDTeE	TRDTeE
SC_PropStages	-	Placeholder	Placeholder	Placeholder	Placeholder
SC_Lander1	-	Placeholder	Placeholder	Placeholder	Placeholder
SC_SampleReturns	-	Placeholder	Placeholder	Placeholder	Placeholder
SC_AdtlEls	-	Placeholder	Placeholder	Placeholder	Placeholder
Science Payload	**-**	**-**	**-**	**-**	**-**
Payload_Enc	-	1a	1a	2a	2b
Payload_Return	-	No Instruments	No Instruments	No Instruments	No Instruments
Payload_Lander	-	No Instruments	No Instruments	No Instruments	No Instruments
Mission Design & Navigation	**-**	**-**	**-**	**-**	**-**
Trajectory Architecture					
Configuration Architecture					
Project Risk					
Use JPL or NASA Mass Margin					
Additional NASA Mass Margin					
JPL Mass Margin					
RMA Custom					
Dynamic Trajectory Type					
Dynamic Trajectory Selection					

Database of Architectures					
System Trades	**Units**	**Arch 2c**	**Arch 2d**	**Arch 2e**	**Arch 2f**
System 1 Architecture	**-**	**-**	**-**	**-**	**-**
SC_Analogue1	-	TSSM	Juno	Juno	Juno
Leveraging Type	-	TRDTeE	TRDTeE	TRDTeE	TRDTeE
SC_PropStages	-	Placeholder	Placeholder	Placeholder	Placeholder
SC_Lander1	-	Placeholder	Placeholder	ard Seismic Land	Placeholder
SC_SampleReturns	-	Placeholder	Placeholder	Placeholder	Placeholder
SC_AdtlEls	-	Placeholder	Placeholder	Placeholder	Placeholder
Science Payload	**-**	**-**	**-**	**-**	**-**
Payload_Enc	-	2c	2d	2a	2f
Payload_Return	-	No Instruments	No Instruments	No Instruments	No Instruments
Payload_Lander	-	No Instruments	No Instruments	2e	No Instruments
Mission Design & Navigation	**-**	**-**	**-**	**-**	**-**
Trajectory Architecture	-	Simple Enc. Orbite	imple Enc. Orbite	Enc. Landing	imple Enc. Orbite
Configuration Architecture	-	Single Element	Single Element	Hard Lander	Single Element
Project Risk	**-**	**-**	**-**	**-**	**-**
Use JPL or NASA Mass Margins	%	NASA	NASA	NASA	NASA
Additional NASA Mass Margin	%	0	0	0	0
JPL Mass Margin	%	0	0	0	0
RMA Custom	**-**	**-**	**-**	**-**	**-**
Dynamic Trajectory Type	-	BasicOrb	BasicOrb	BasicOrb	BasicOrb
Dynamic Trajectory Selection	-	910	910	910	910

INCOSE
International Council on Systems Engineering

Desktop is a metaphor for work

INCOSE
International Council on Systems Engineering

But: Is the metaphor appropriate?

INCOSE
International Council on Systems Engineering

There's a lot to look after all at once

INCOSE
International Council on Systems Engineering

And we may only do the job once

INCOSE
International Council on Systems Engineering

We aren't the first to look at this

But you do have to be a little careful

So what are we going to do about it?

INCOSE
International Council on Systems Engineering

ACT TWO: MBSE – THE CLAIMS

Motivation for MBSE

- Development of modern systems (aircrafts, spacecrafts, missions) involves complex interactions and interfaces between constituent systems and system designers
  - Managing and maintaining a consistent understanding of the current system concept or design can be laborious and error prone using traditional systems engineering approaches
  - Facilitate communication between the various disciplines (domains of expertise) which use different terminology and rely on different analytical tools and approaches
- End-to-end architectural analysis from the system level, multi-discipline view to the discipline specific analysis
  - Integrate the perspectives and concerns of the various disciplines into the overall system
  - Coherently communicate concepts and constraints between system-level and discipline specific analysis
- These challenges manifest as risk to cost and schedule

Source: Kevin Somervill, LaRC

Potential in MBSE

- Model Based Systems Engineering (MBSE) is considered an opportunity to improve current processes
  - Capture system “views” relative to stakeholder perspectives
  - Ensure a consistent view of the developing system as close to real-time as possible
  - Reduce time to integrate disparate subsystem concepts to “close” the design within project constraints
  - Corresponding reduction in cost (and schedule) of development

- To better understand these opportunities:
  - There have been several investigations in MBSE for systems model development to support project activities as well as process and tool evaluation

Source: Kevin Somervill, LaRC

MBSE Benefits

- **Improved quality**
  - Early identification of requirements issues
  - Enhanced system design integrity
  - Reduction in unintended behaviors/outcomes
  - Improved specification of allocated req’ts to HW/SW
  - Fewer errors during I&T
  - More rigorous requirements traceability
- **Increased productivity**
  - Improved impact analysis of requirements changes
  - Increased trade space
  - Model reuse to support design/technology evolution
  - Auto-generation of documentation
- **Reduced risk**
  - Improved Improved cost estimates cost estimates
  - Early/on-going requirements validation & design verification

INCOSE
International Council on Systems Engineering

Source: Sanford Friedenthal, INCOSE MBSE Lead

Benefits of MBSE in an Integrated Environment

- Single data environment ensures completeness & consistency of design data
- Rich database permits multi-user input and immediate synchronization, improving efficiency and productivity
- Use of a single data environment results in data availability throughout program life-cycles
- Traceability through model elements enables efficient change / impact analysis enabling a more adaptable system
- Robust query engine allows rapid assessment of the integrated database, finding anomalies early, preventing rework

Source: Christi A. Gau Pagnanelli, Boeing

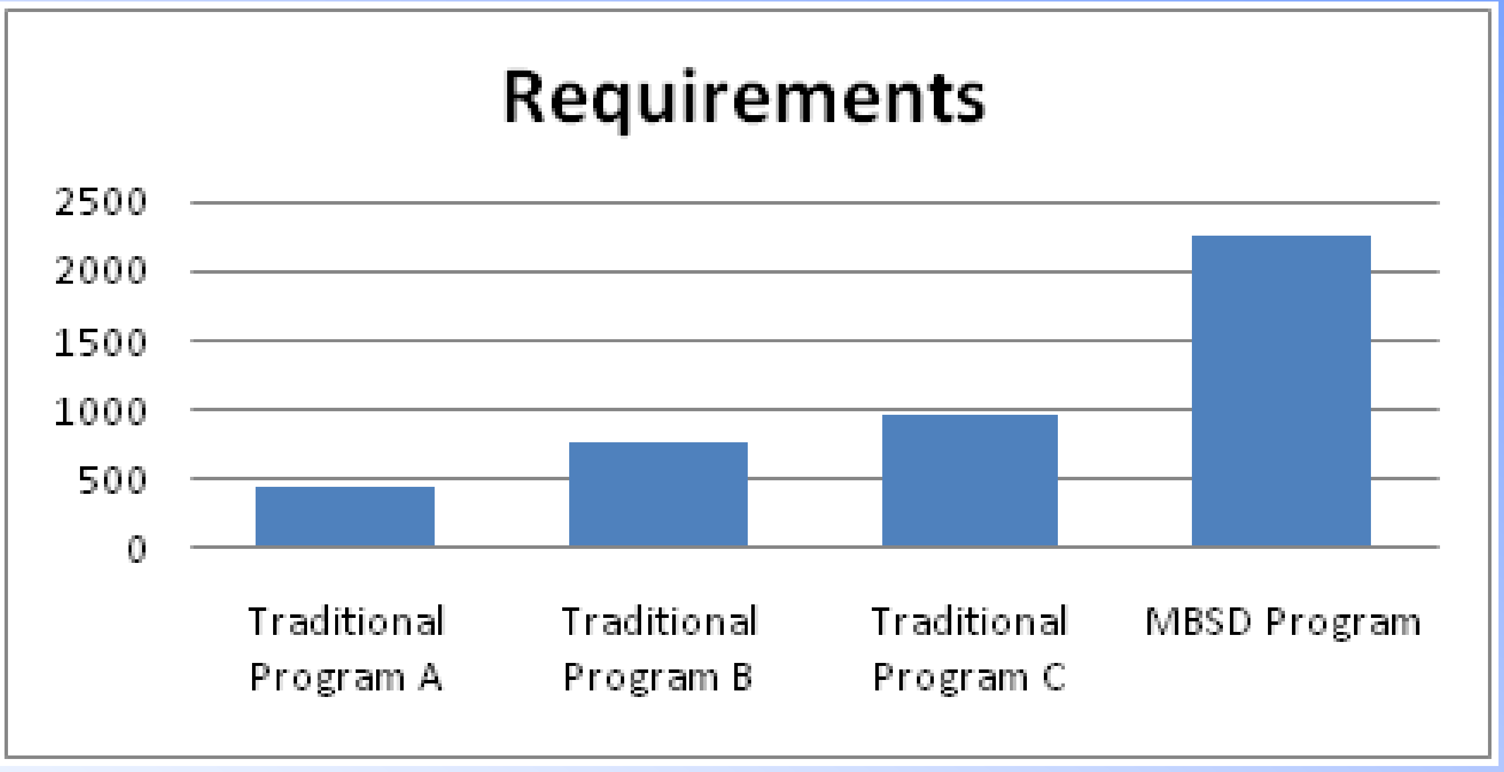
MBSE Return on Investment:
A Case Study from LM Aeronautics

- The set of slides that follow show the data collected from four real aerospace and defense programs at Lockheed Martin.
- Three of the programs used the traditional document-based engineering approach. One program used the model-based engineering approach.
- The actual names of the programs have been omitted to protect the integrity of these ongoing programs.
- *Note: these slides use the acronym "MBSD" (Model-Based Systems Development). Though the industry has not come to consensus on a set of acronyms, "MBSD" is meant to convey a set of model-based activities that span all engineering domains, while "MBSE" is generally used to refer to model-based activities within the systems engineering domain only.*

Source: Lenny Delligatti, Lockheed Martin

MBSE Return on Investment: *Requirements Comparison*

- This figure compares the number of requirements for these four aeronautics programs.
- The program that used the model-based approach had more than twice as many requirements as the next largest program.

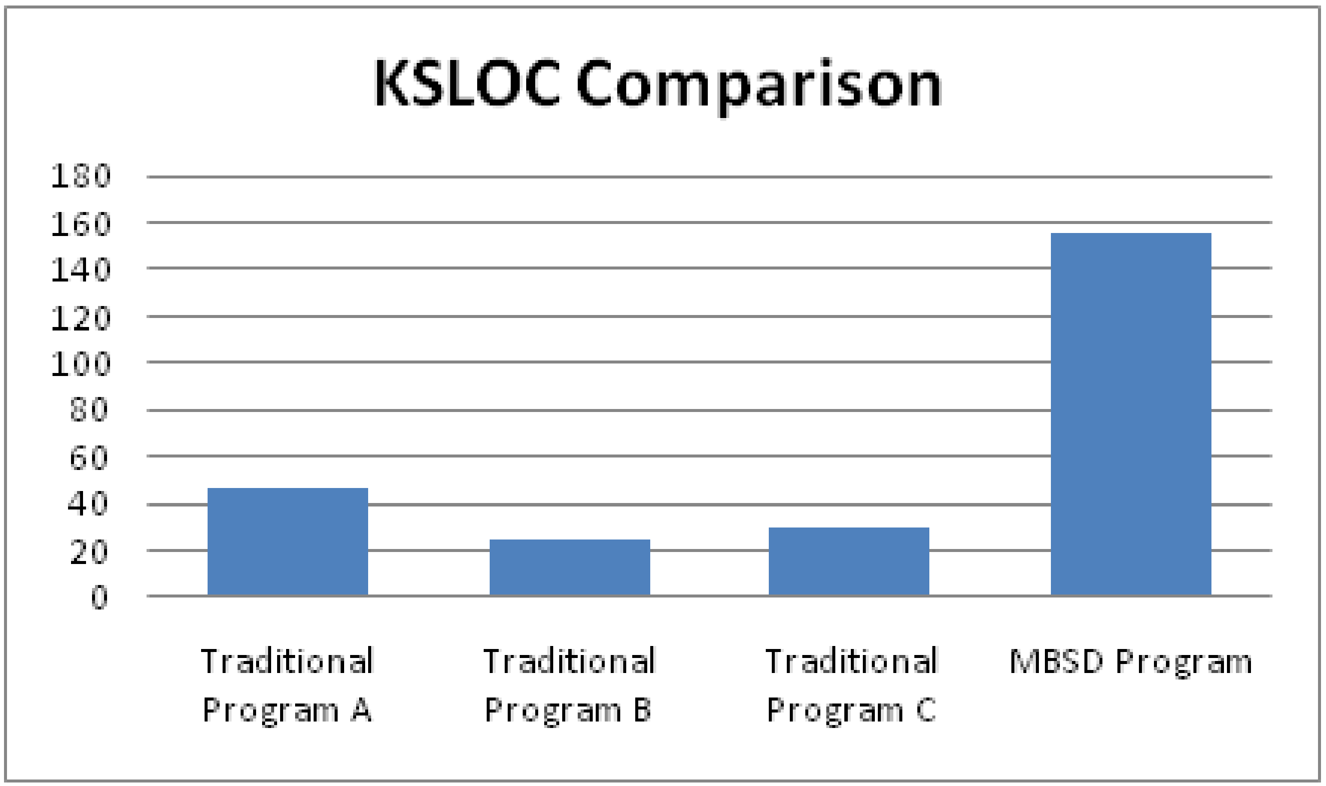


INCOSE
International Council on Systems Engineering

Source: Lenny Delligatti, Lockheed Martin

MBSE Return on Investment:
KSLOC Comparison

- This figure compares KSLOC for these four programs as another metric to contrast system size.
- The program that used the model-based approach developed a significantly larger system than the other three programs.

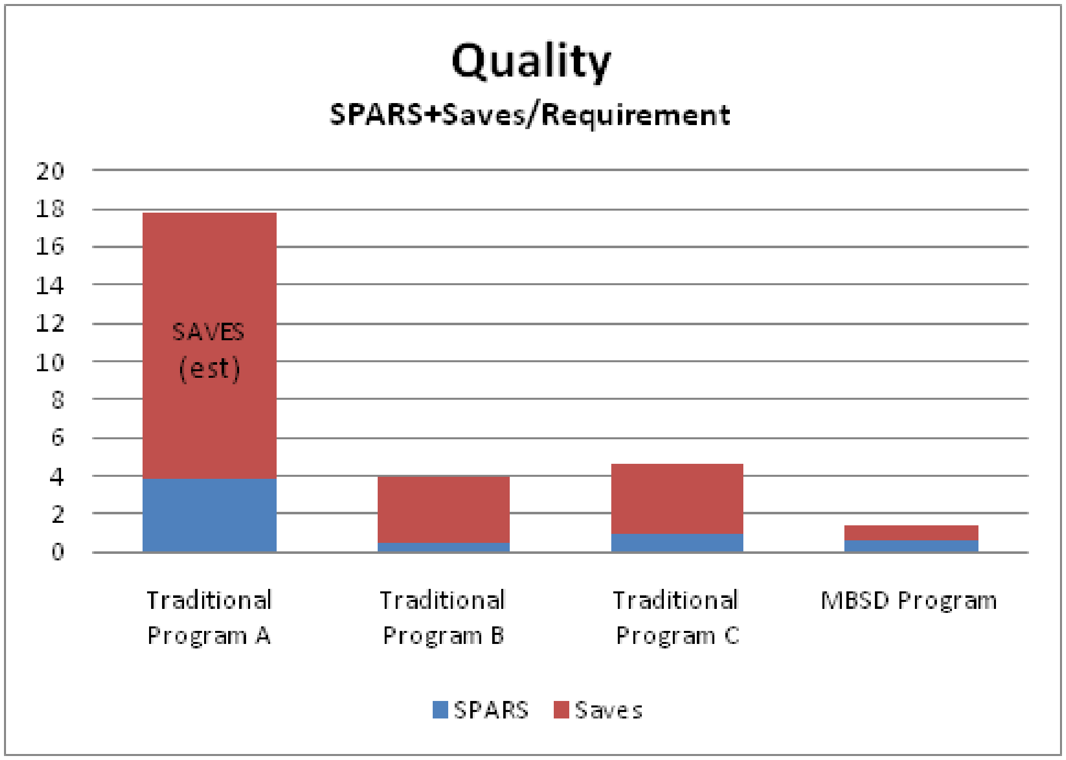


Source: Lenny Delligatti, Lockheed Martin

MBSE Return on Investment: *Quality Comparison*

- This figure compares the number of *defects per requirement* for these four programs.

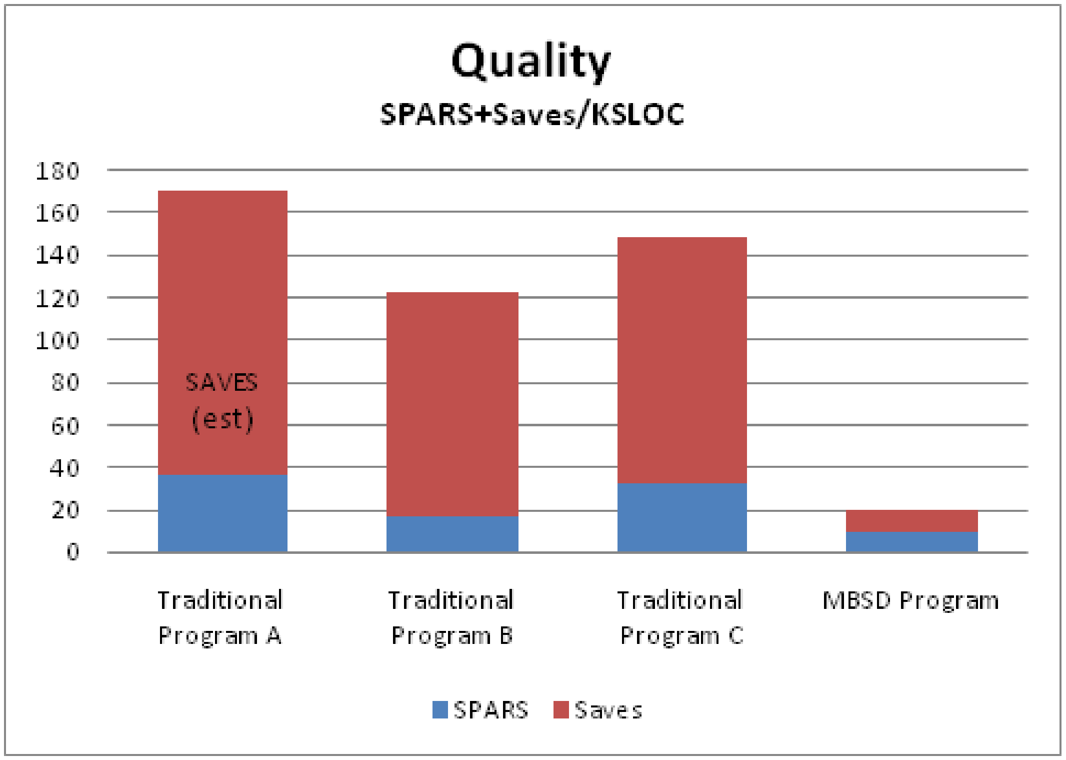
- *Note:*
- *"SPAR" refers to a defect caught after release.*
- *"Save" refers to a defect caught before release.*



Source: Lenny Delligatti, Lockheed Martin

MBSE Return on Investment:
Quality Comparison

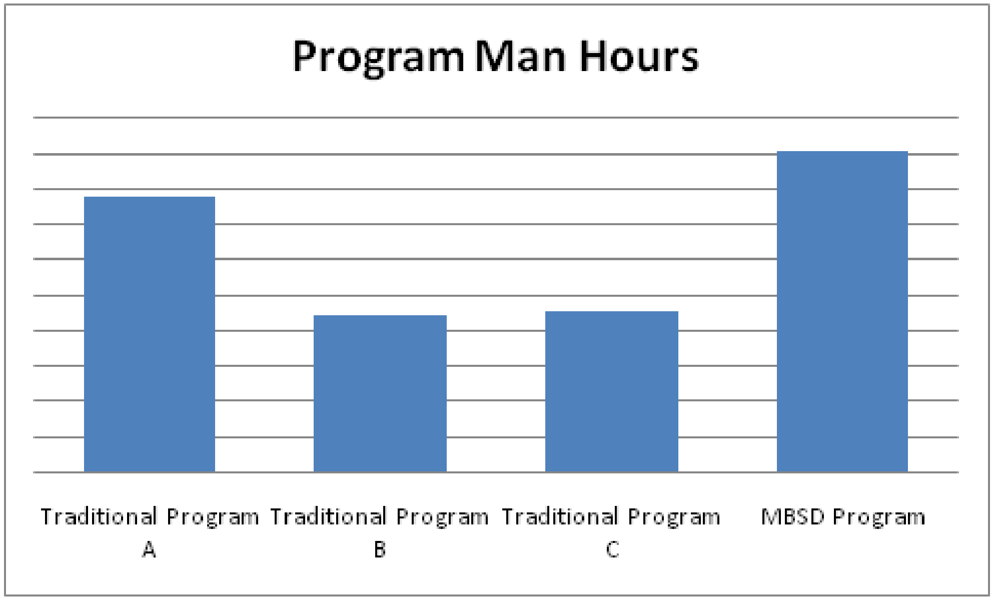
- This figure compares the number of *defects per KSLOC* for these four programs.
- Though the MBSD-produced system was significantly larger than the other three, its defect density was greatly reduced.



Source: Lenny Delligatti, Lockheed Martin

MBSE Return on Investment: *Program Cost in Man-Hours*

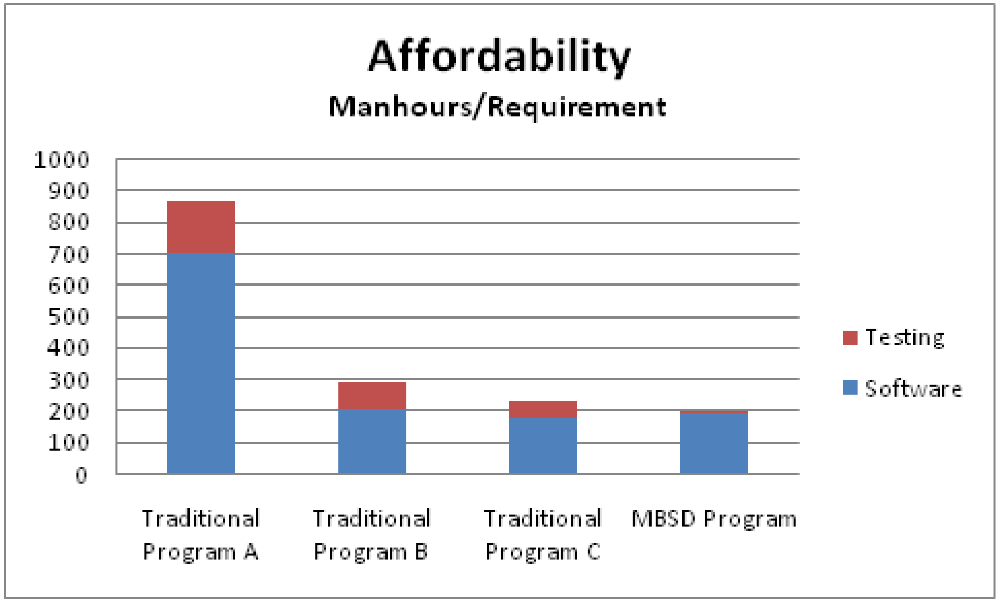
- This figure compares the relative cost of these four programs in man-hours. (Actual values have been omitted for proprietary reasons.)
- As expected, the *total* number of man-hours for the MBSD-produced system was greater than the other three, correlated to system size.



Source: Lenny Delligatti, Lockheed Martin

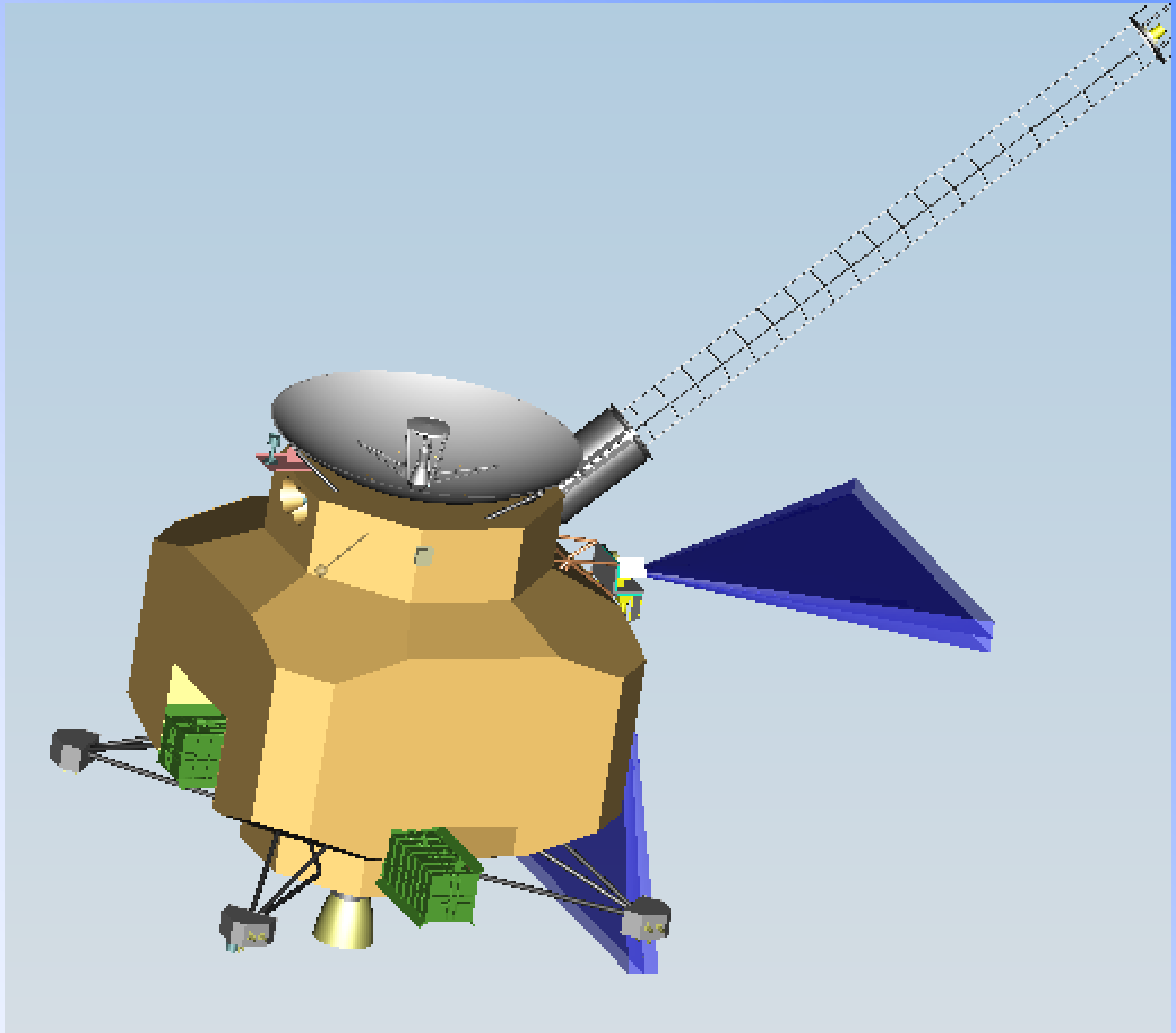
MBSE Return on Investment: *Affordability*

- When normalized for system size, however, the MBSD-produced system was developed at a significantly reduced cost.
- Program C—the next closest in affordability—was 10% more costly than the MBSD-produced system.



Source: Lenny Delligatti, Lockheed Martin

What is the SE *practice* like now?



ACT THREE: MBSE – THE PRACTICE

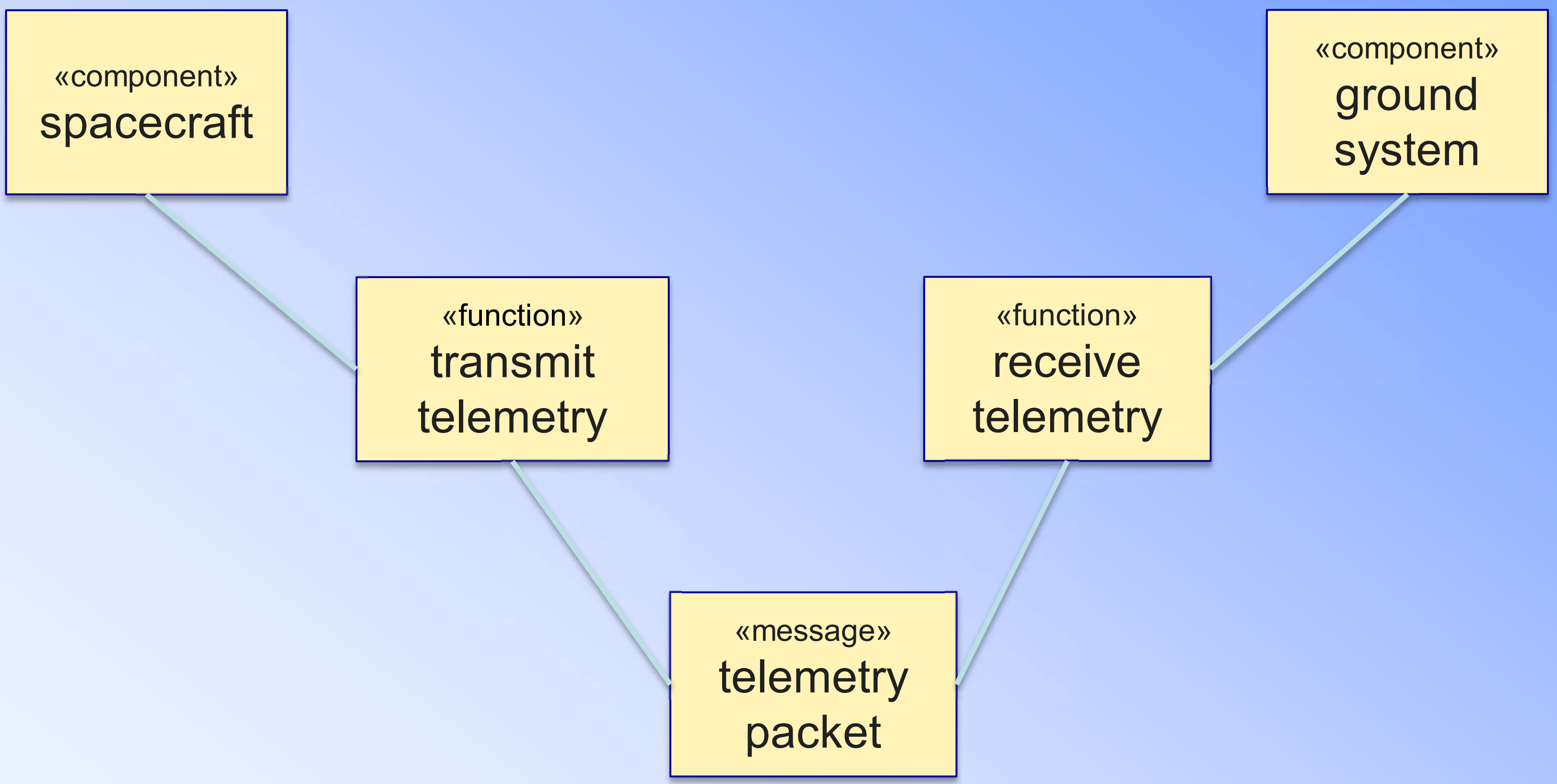
Applying Uniform Reasoning

Making Distinctions Explicit

- Rather than merely hinting at distinctions with shapes or colors, we could devise a set of types or classes to be applied to model elements
- The set of types is application-dependent
  - Systems engineers talk about different things from chefs
  - The distinctions are whatever matters for your application
  - Is red wine a different *type* from white, or is is merely a *property* of wine?
    - It depends on what you want to say about wine
- What kinds of things do systems engineers talk about?
  - Component, Interface, Function, Requirement, Work Package, Product, Process, Objective, Message, etc.
- Let’s apply some classes to our model
- For now, every element has
  - one type, denoted like this: «type»
  - one name, which identifies an individual of that type

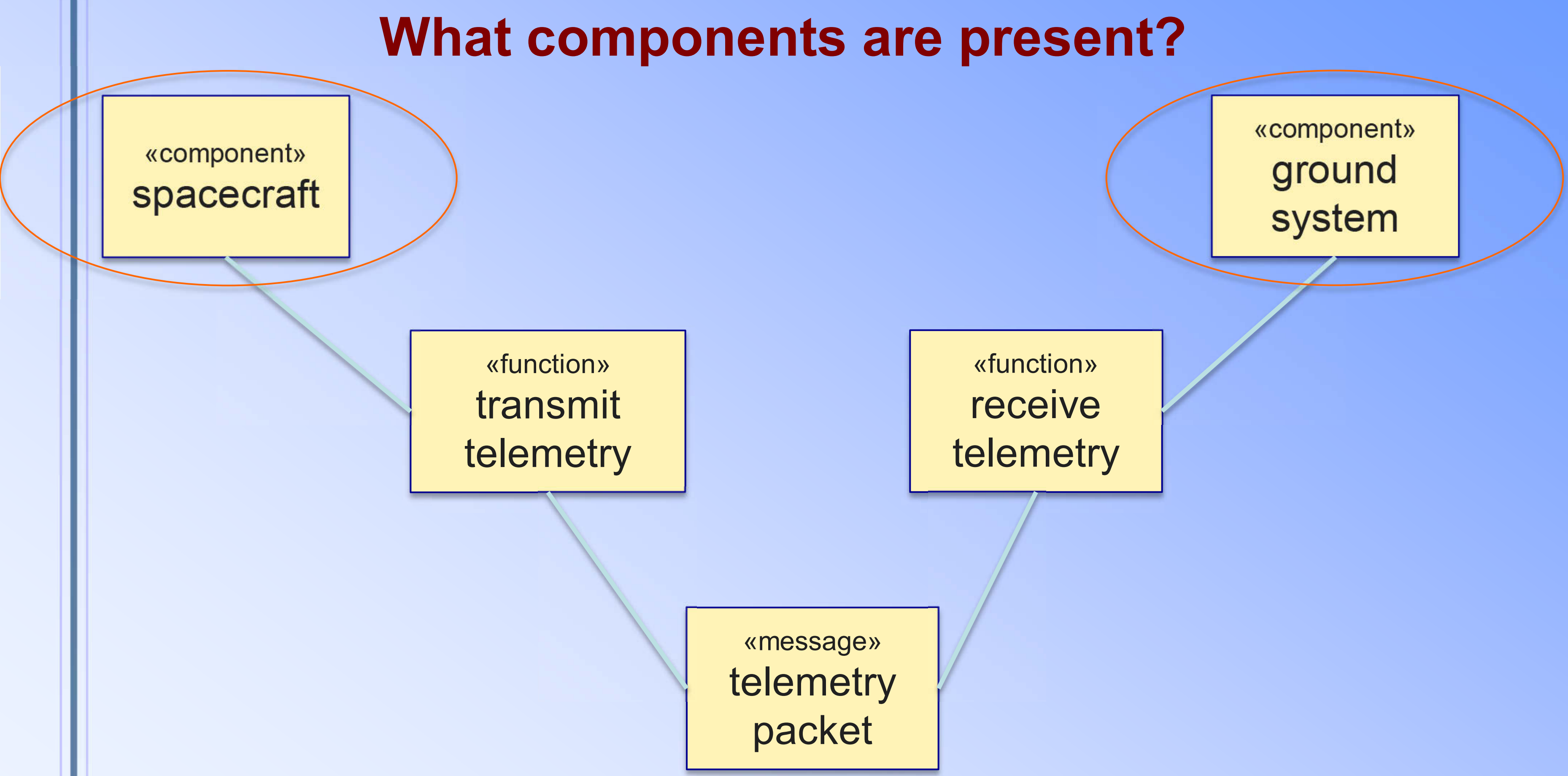
Source: Steve Jenkins, Dan Dvorak, JPL

Model With Typed Elements



Source: Steve Jenkins, Dan Dvorak, JPL

Answering Questions

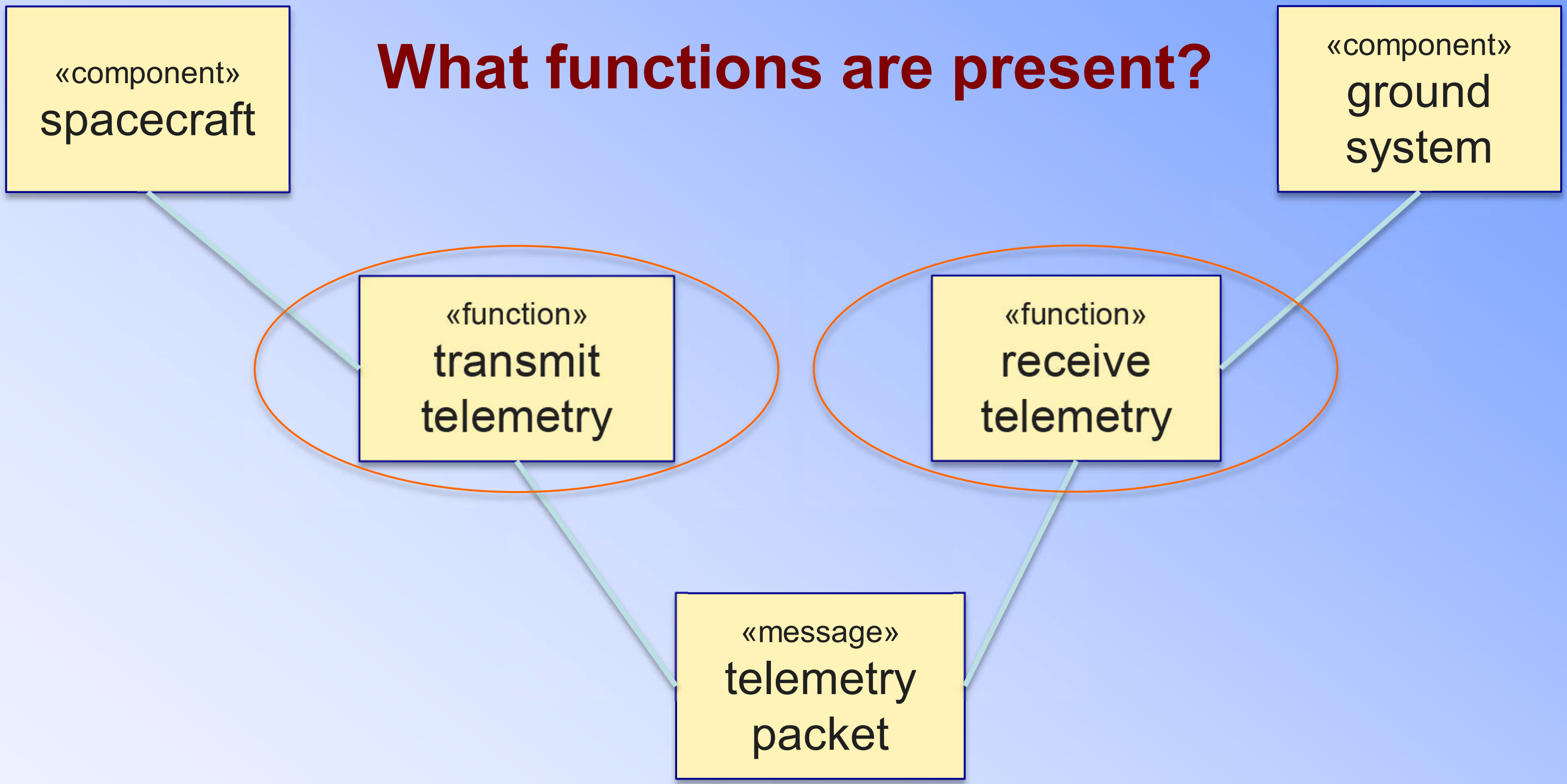


INCOSE
International Council on Systems Engineering

Source: Steve Jenkins, Dan Dvorak, JPL

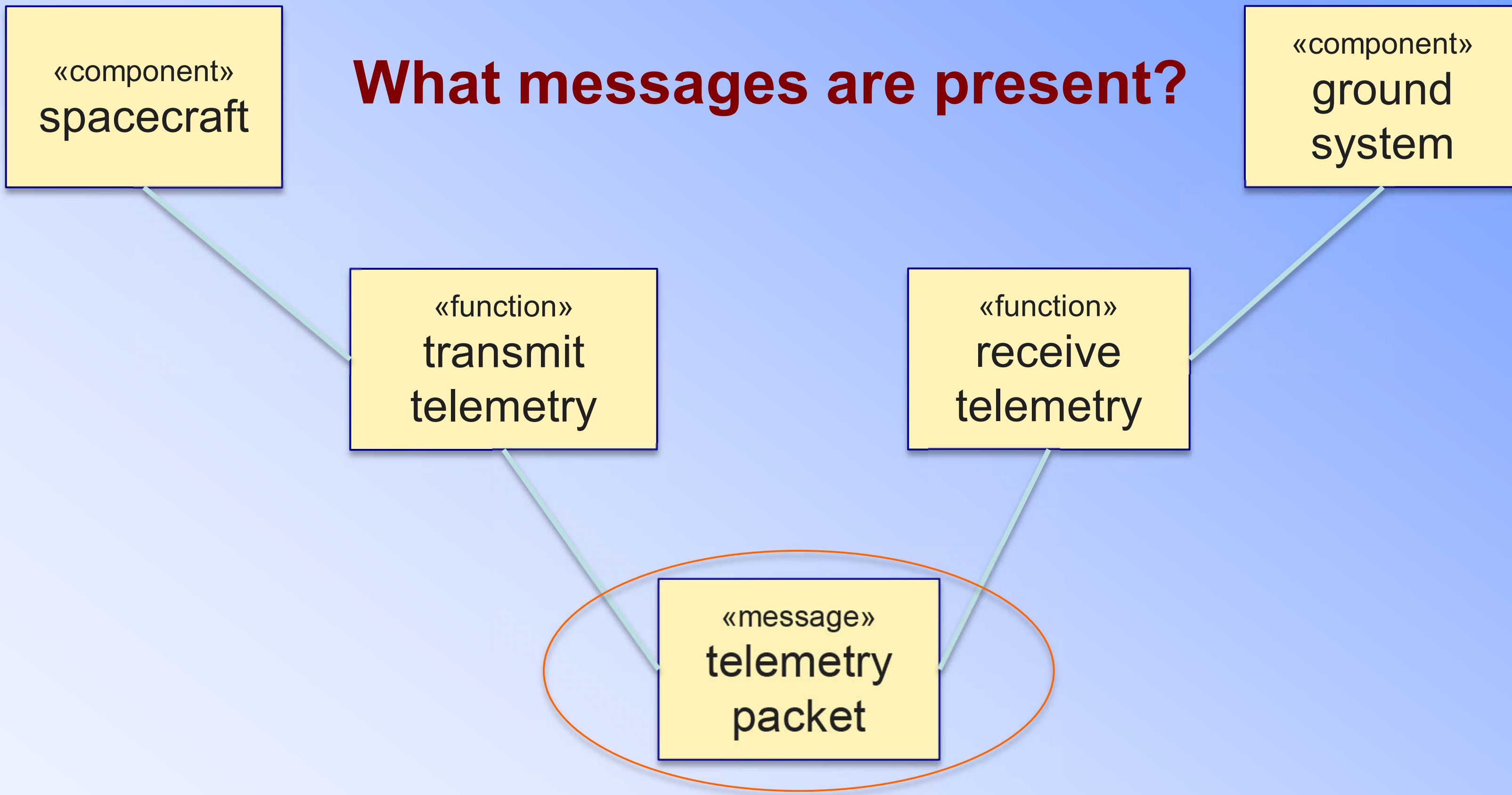
Answering Questions

What functions are present?



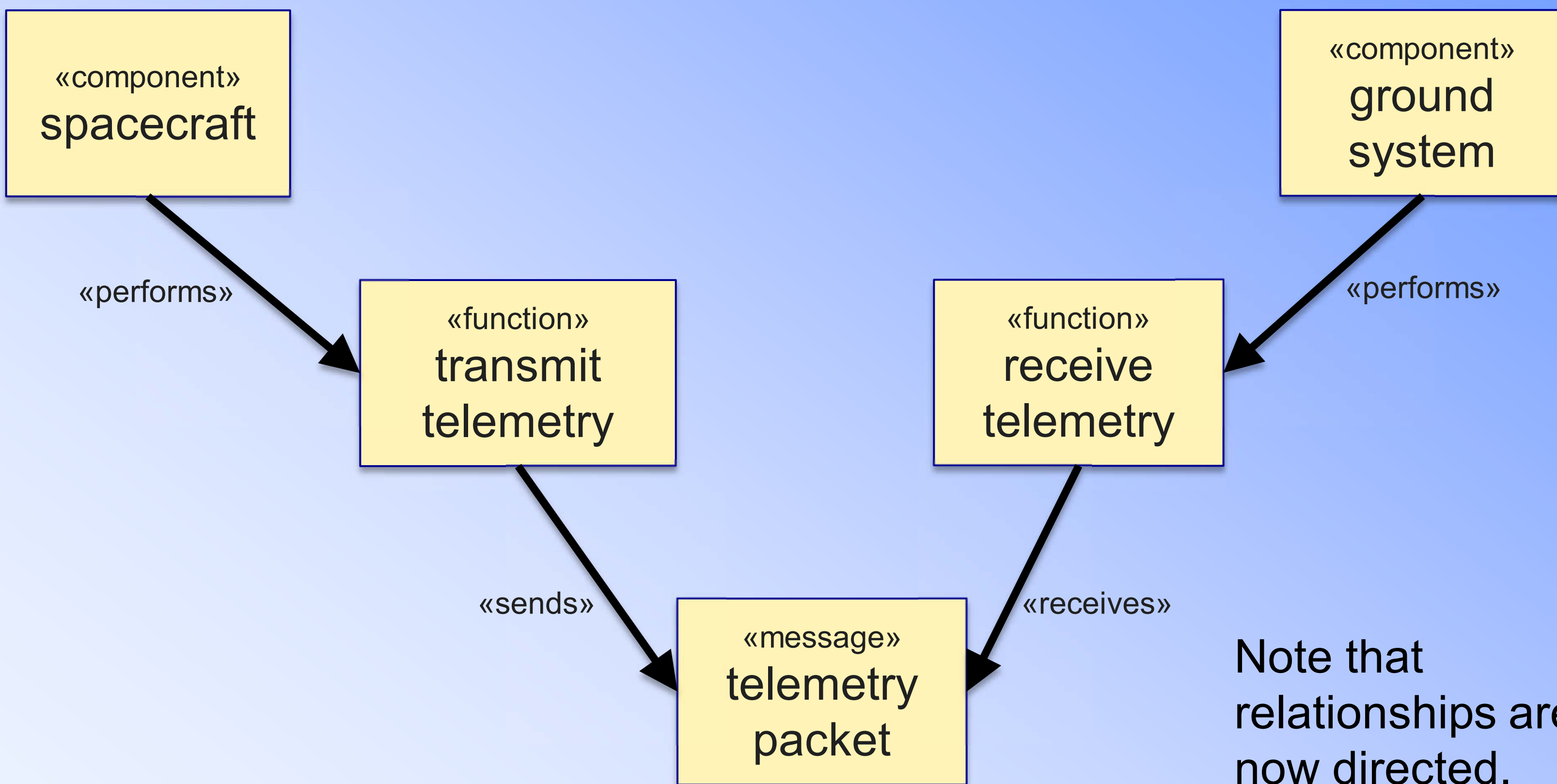
Source: Steve Jenkins, Dan Dvorak, JPL

Answering Questions



Source: Steve Jenkins, Dan Dvorak, JPL

Add Typed Relationships

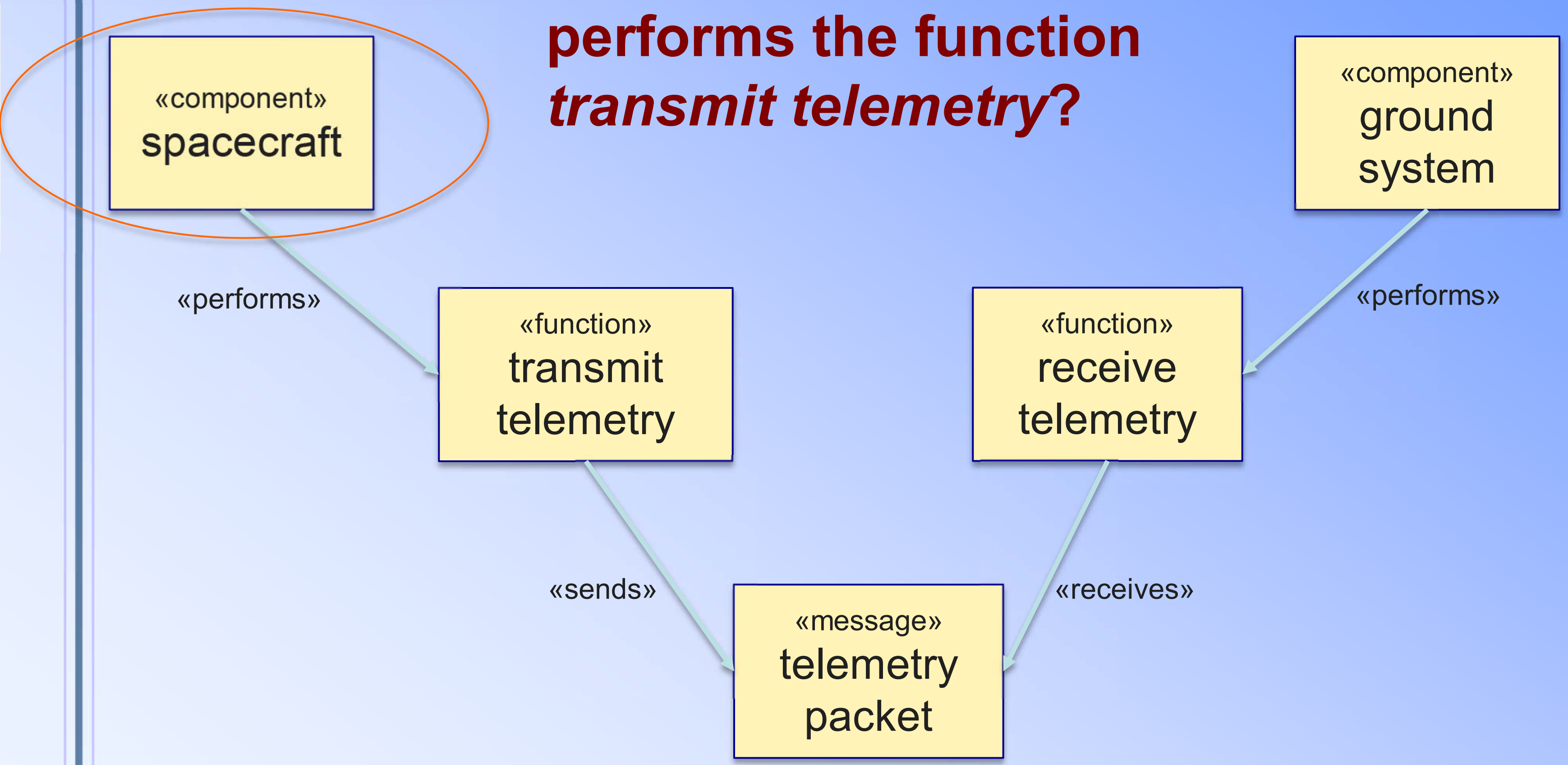


Note that relationships are now directed.

Source: Steve Jenkins, Dan Dvorak, JPL

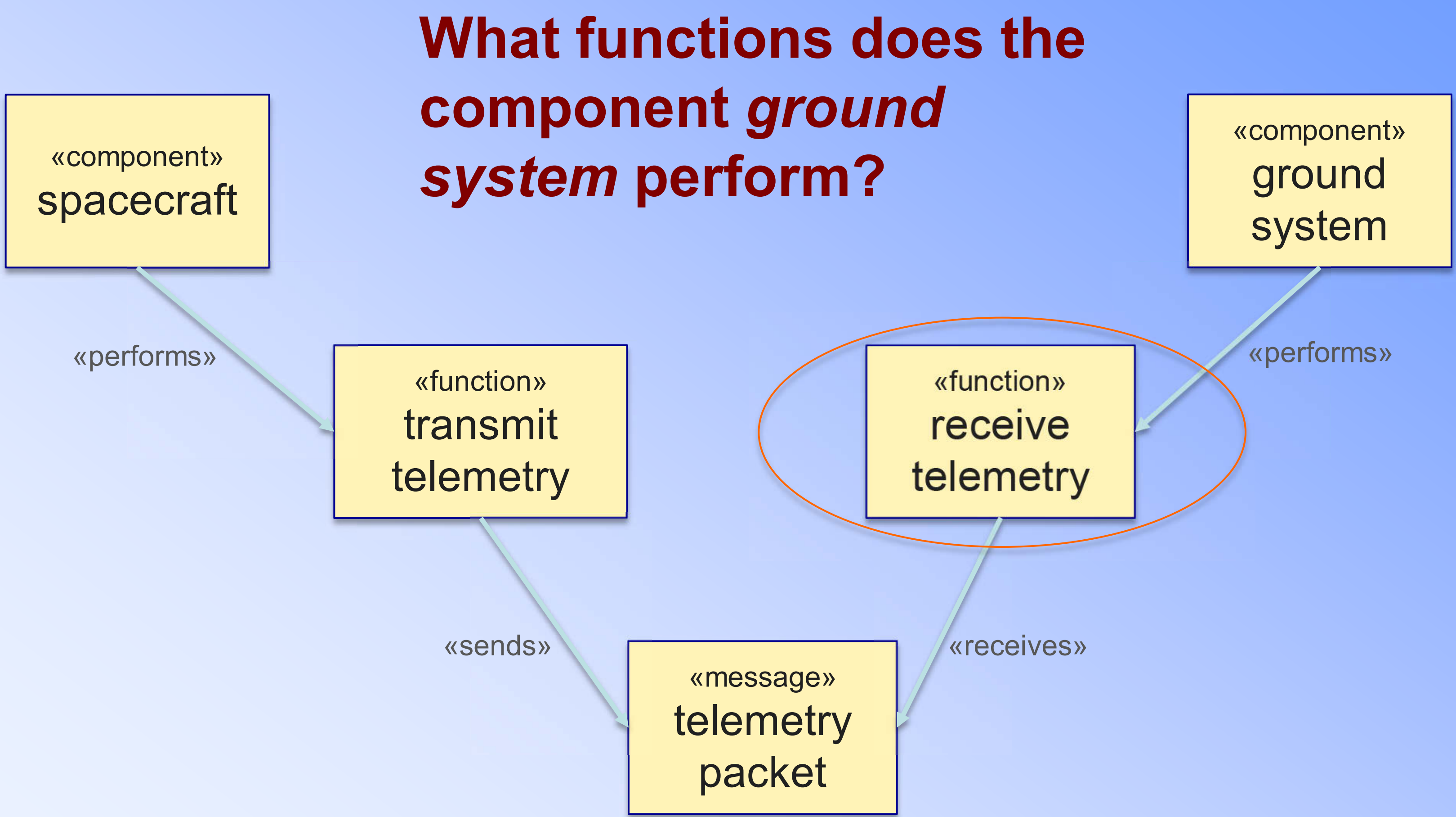
More Questions and Answers

What component performs the function *transmit telemetry*?



Source: Steve Jenkins, Dan Dvorak, JPL

More Questions and Answers

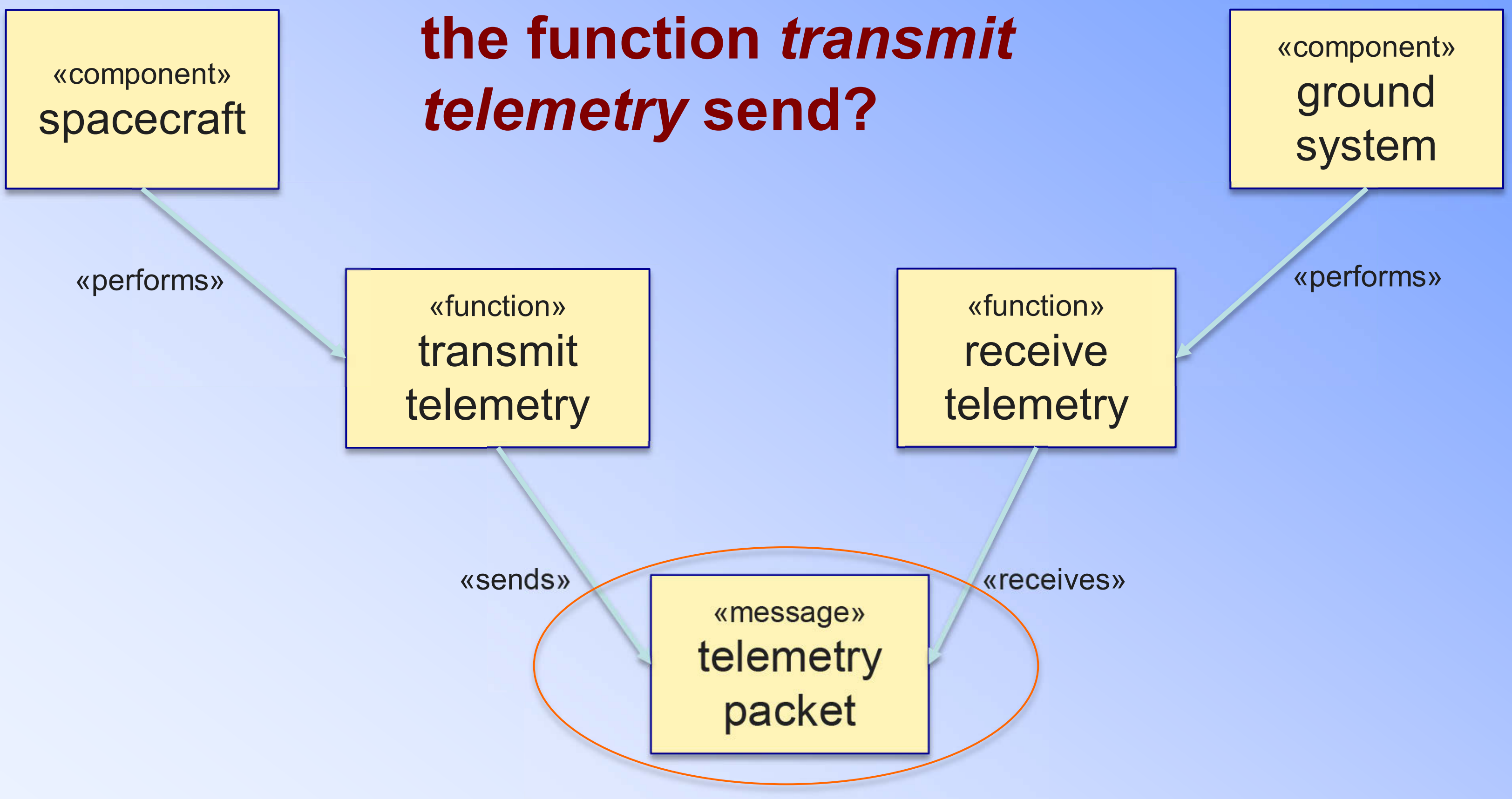


INCOSE
International Council on Systems Engineering

Source: Steve Jenkins, Dan Dvorak, JPL

More Questions and Answers

What messages does the function *transmit telemetry* send?

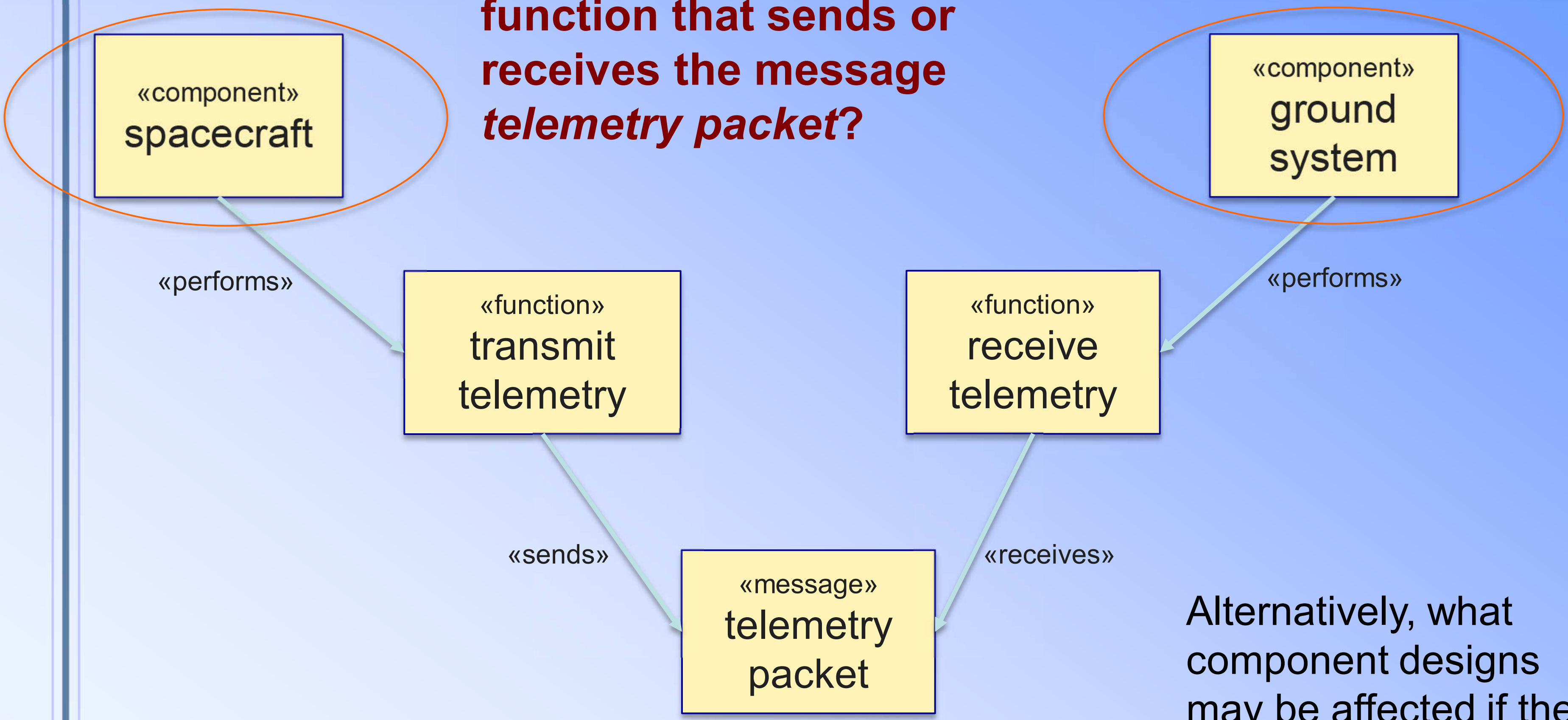


INCOSE
International Council on Systems Engineering

Source: Steve Jenkins, Dan Dvorak, JPL

More Questions and Answers

What components perform a function that sends or receives the message *telemetry packet*?



Alternatively, what component designs may be affected if the definition of *telemetry packet* changes?

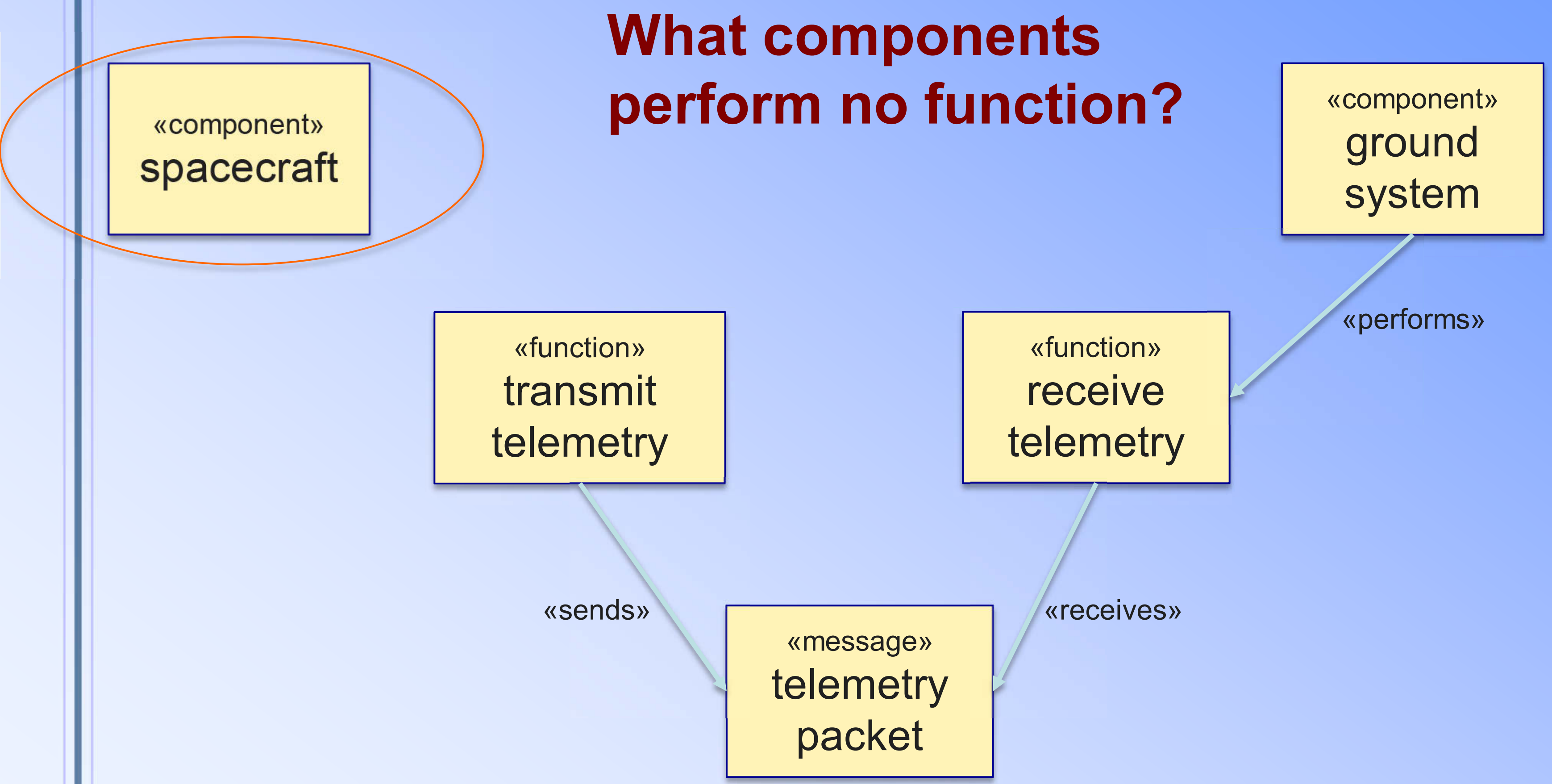
Source: Steve Jenkins, Dan Dvorak, JPL

Reasoning About Models

- We can use models to answer questions
- The questions may be about the system itself
  - What is it?
  - How does it work?
  - Is the performance adequate?
  - What happens if something breaks?
- The questions may be about the model
  - Is it complete?
  - Is it consistent?
  - Does it support required analyses?
- The questions may be about the design artifacts
  - Are all required documents present?
  - Does each document contain all required content?
- We call answering these kinds of questions *reasoning*
  - It doesn’t necessarily mean exotic, artificial intelligence

Source: Steve Jenkins, Dan Dvorak, JPL

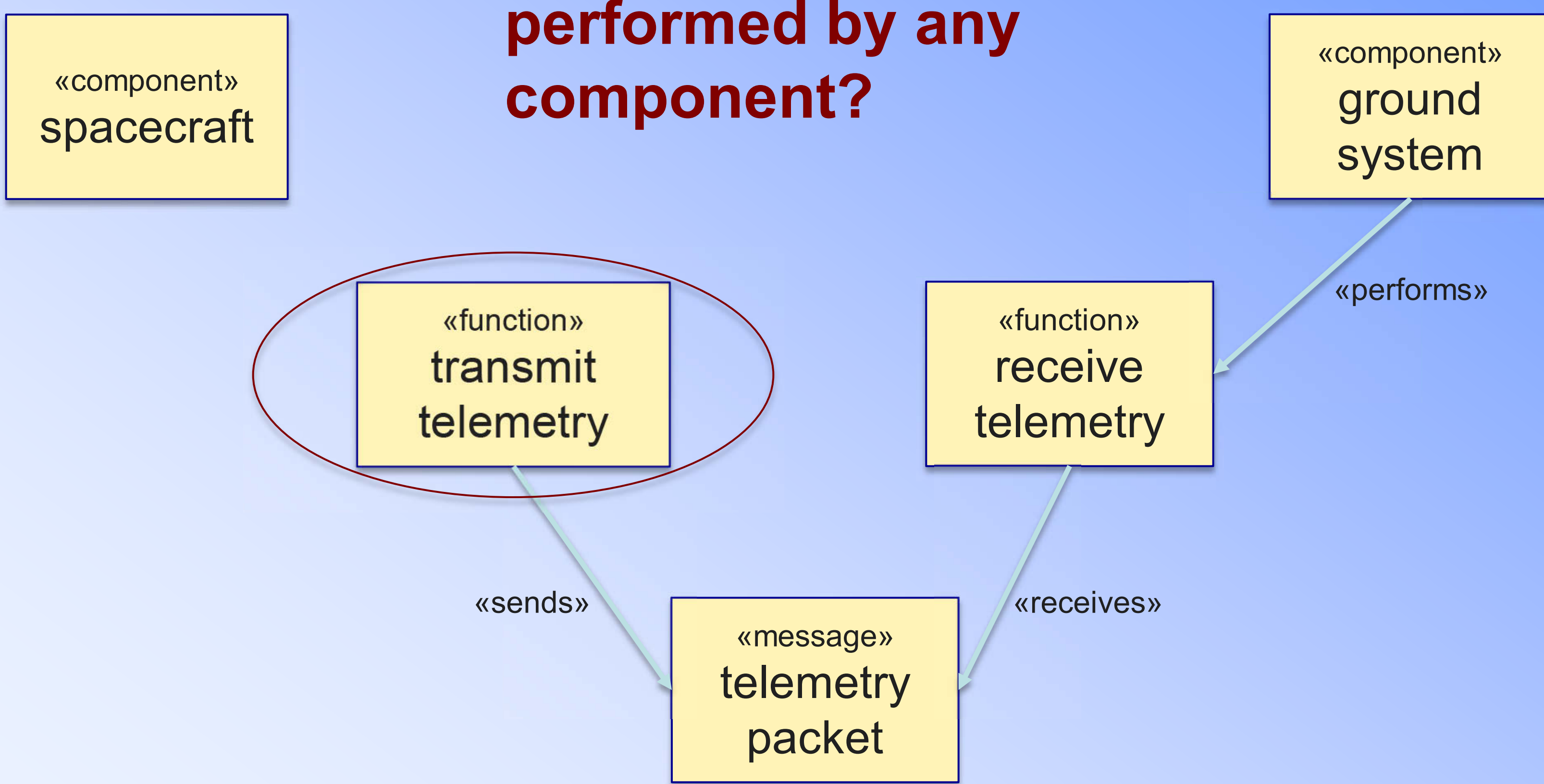
Reasoning About Completeness



Source: Steve Jenkins, Dan Dvorak, JPL

Reasoning About Completeness

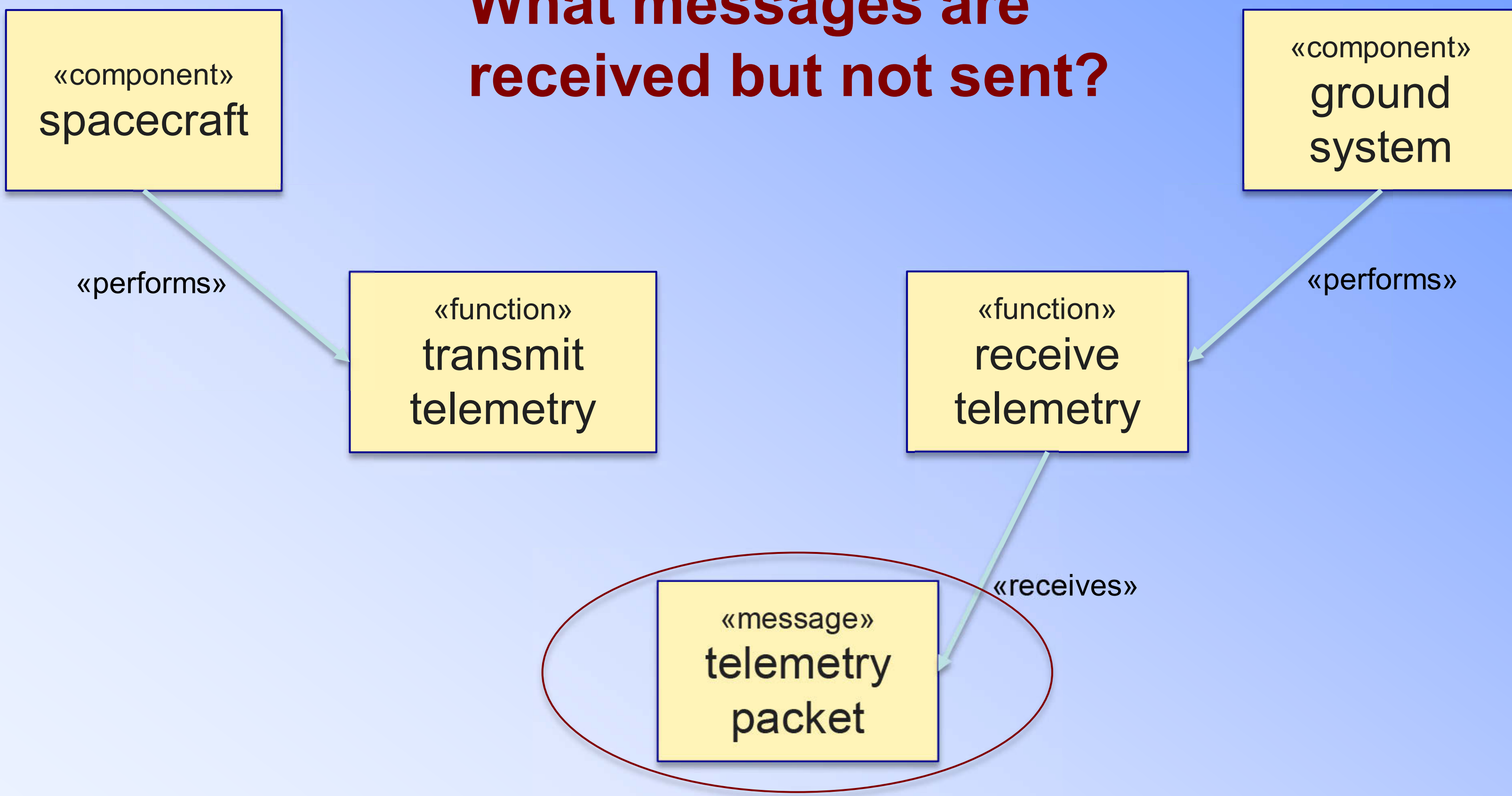
What functions are not performed by any component?



Source: Steve Jenkins, Dan Dvorak, JPL

Reasoning About Completeness

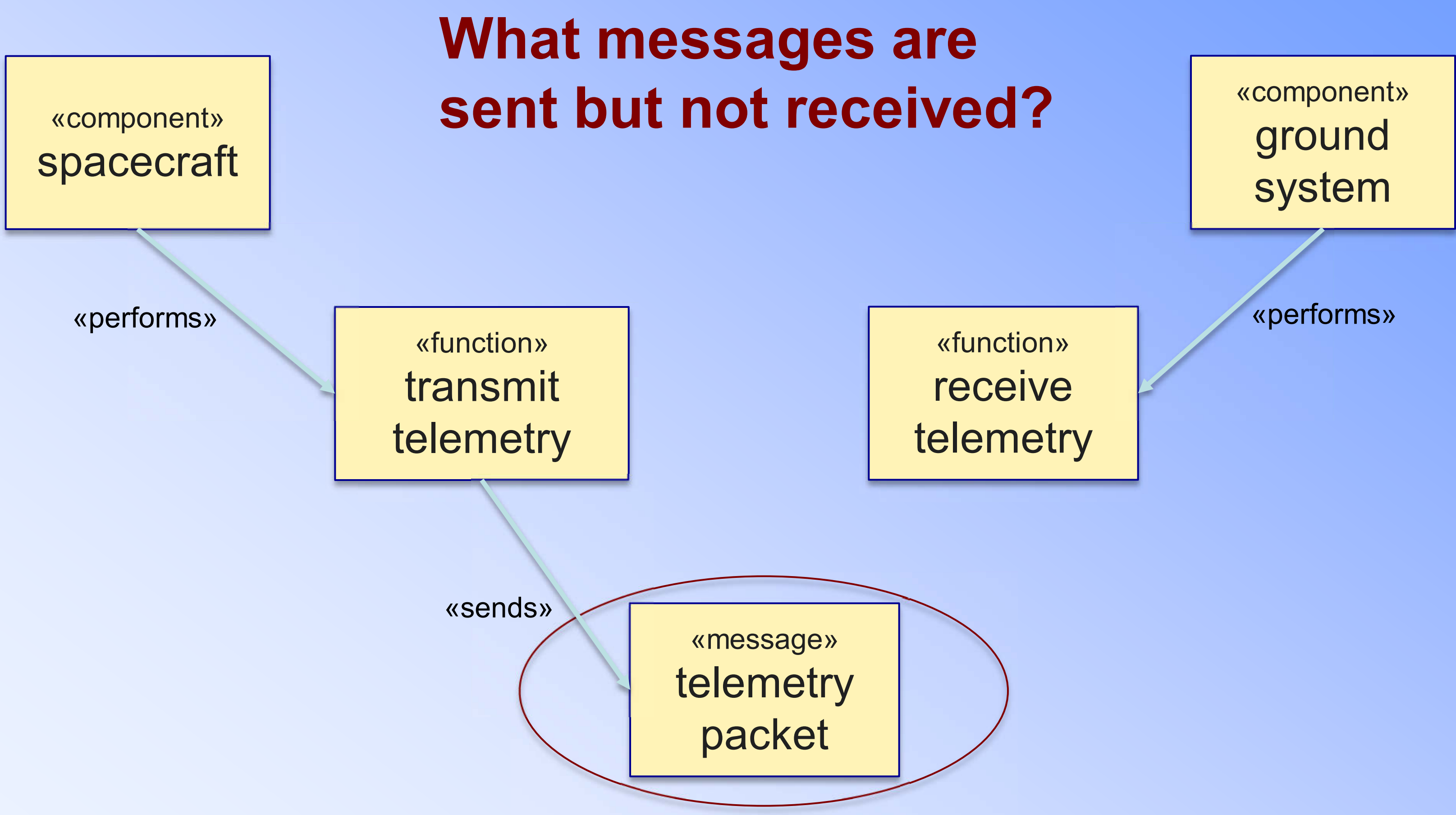
What messages are received but not sent?



INCOSE International Council on Systems Engineering

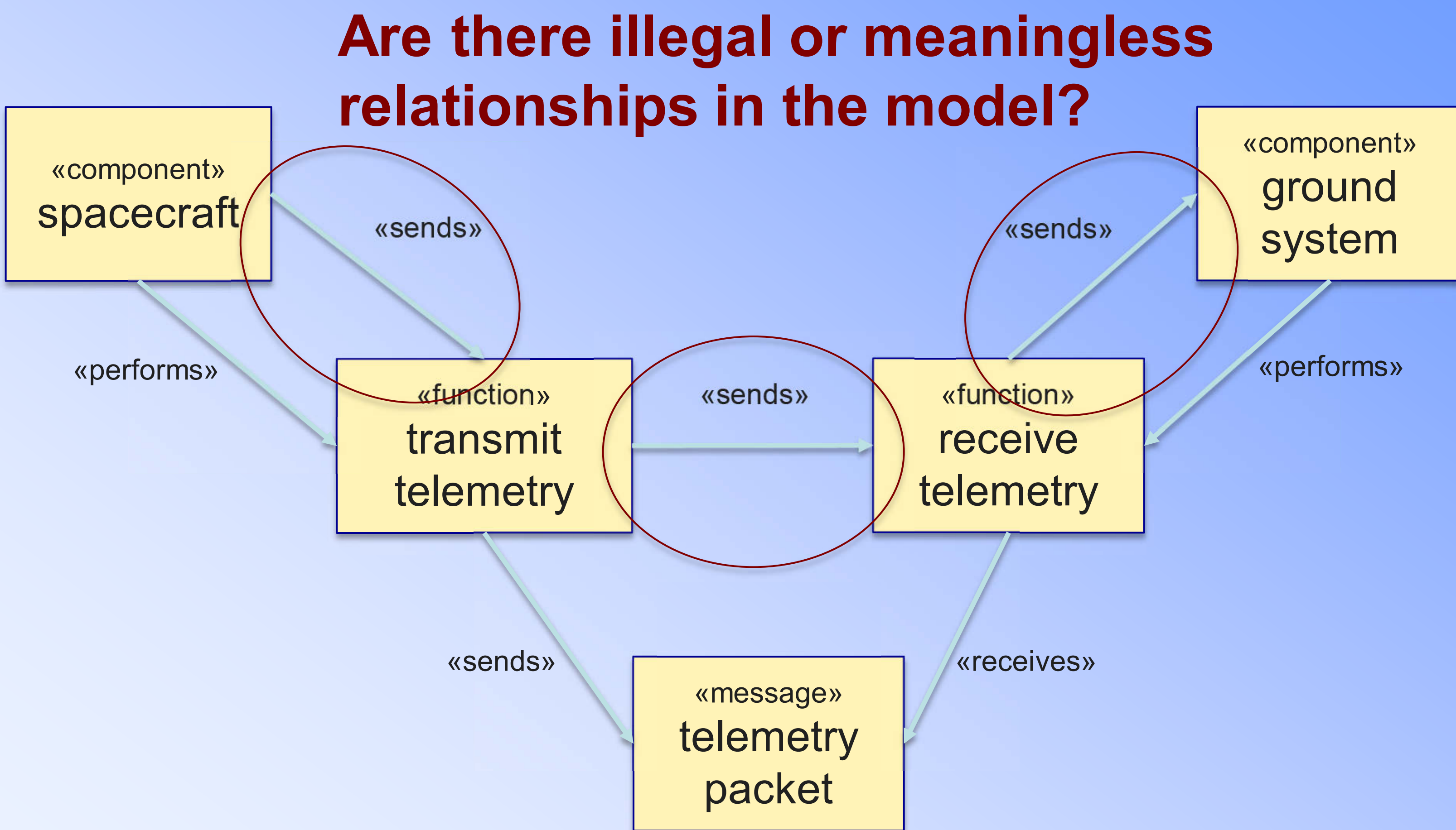
Source: Steve Jenkins, Dan Dvorak, JPL

Reasoning About Completeness



Source: Steve Jenkins, Dan Dvorak, JPL

Reasoning About Consistency



Source: Steve Jenkins, Dan Dvorak, JPL

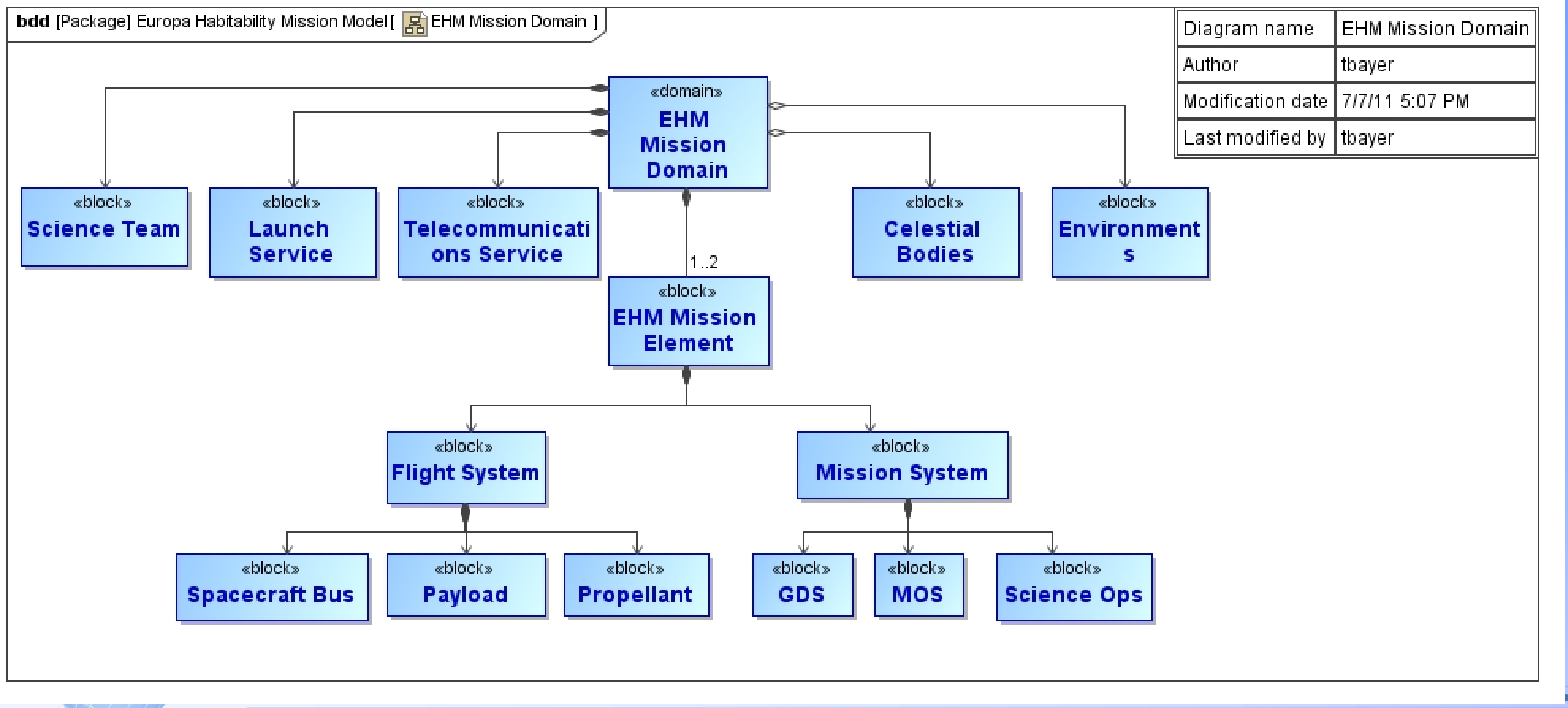
Reasoning About Resources

- Resources come in lots of shapes and sizes
- Mass, power, etc. are numeric
- Volume, FOV are geometric
- Switch count, bus loading, dependency complexity are graph-based ← New analyses enabled by MBSE tools

Viewing the System and its Interactions

Mission Domain

- Top-level view: “front door” to lower-level views



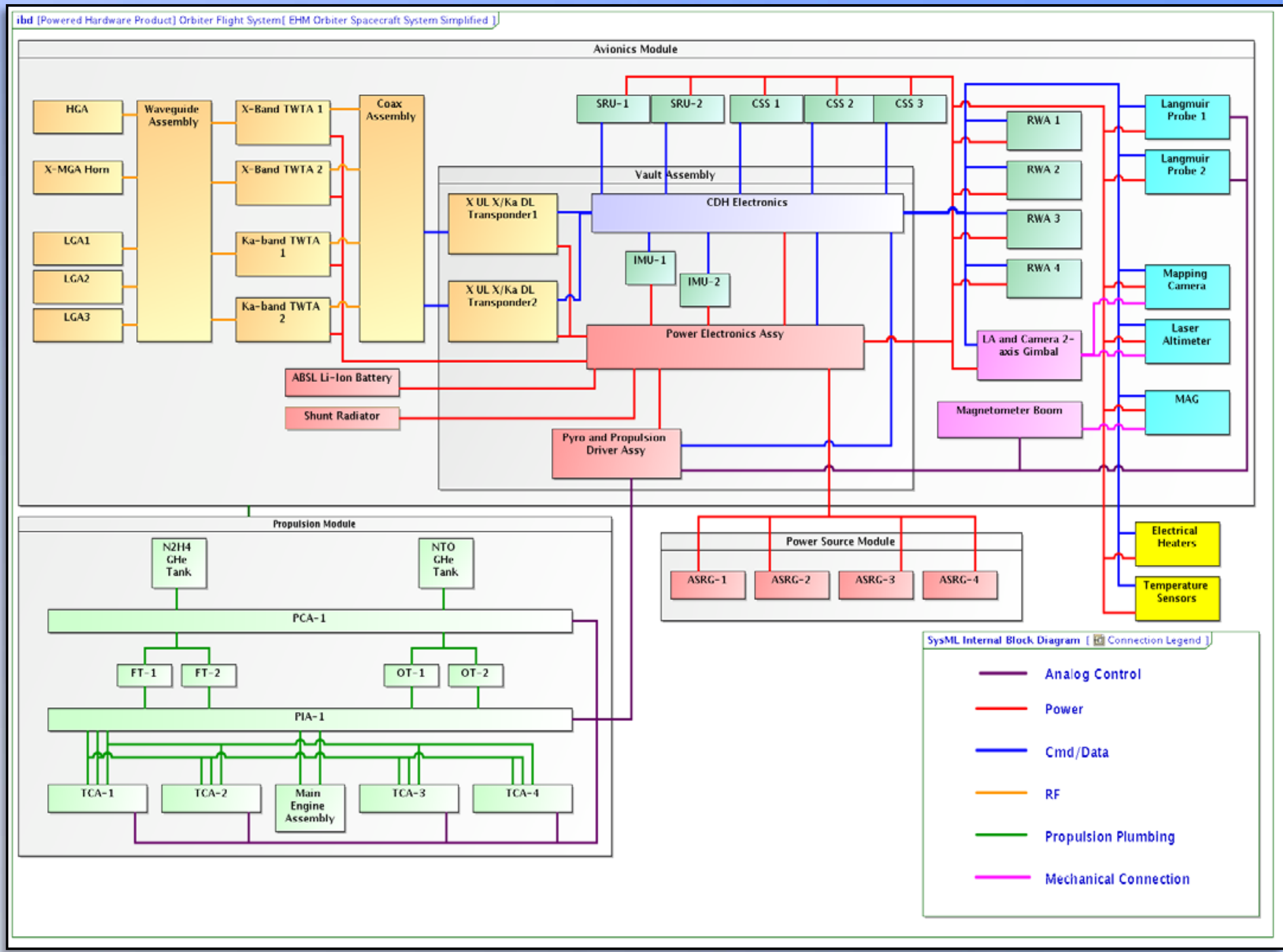
Flight System Deployment (a.k.a. System Block Diagram)

Deployment: a specific arrangement of parts from the product list.

The authoritative statement of the Flight System decomposition

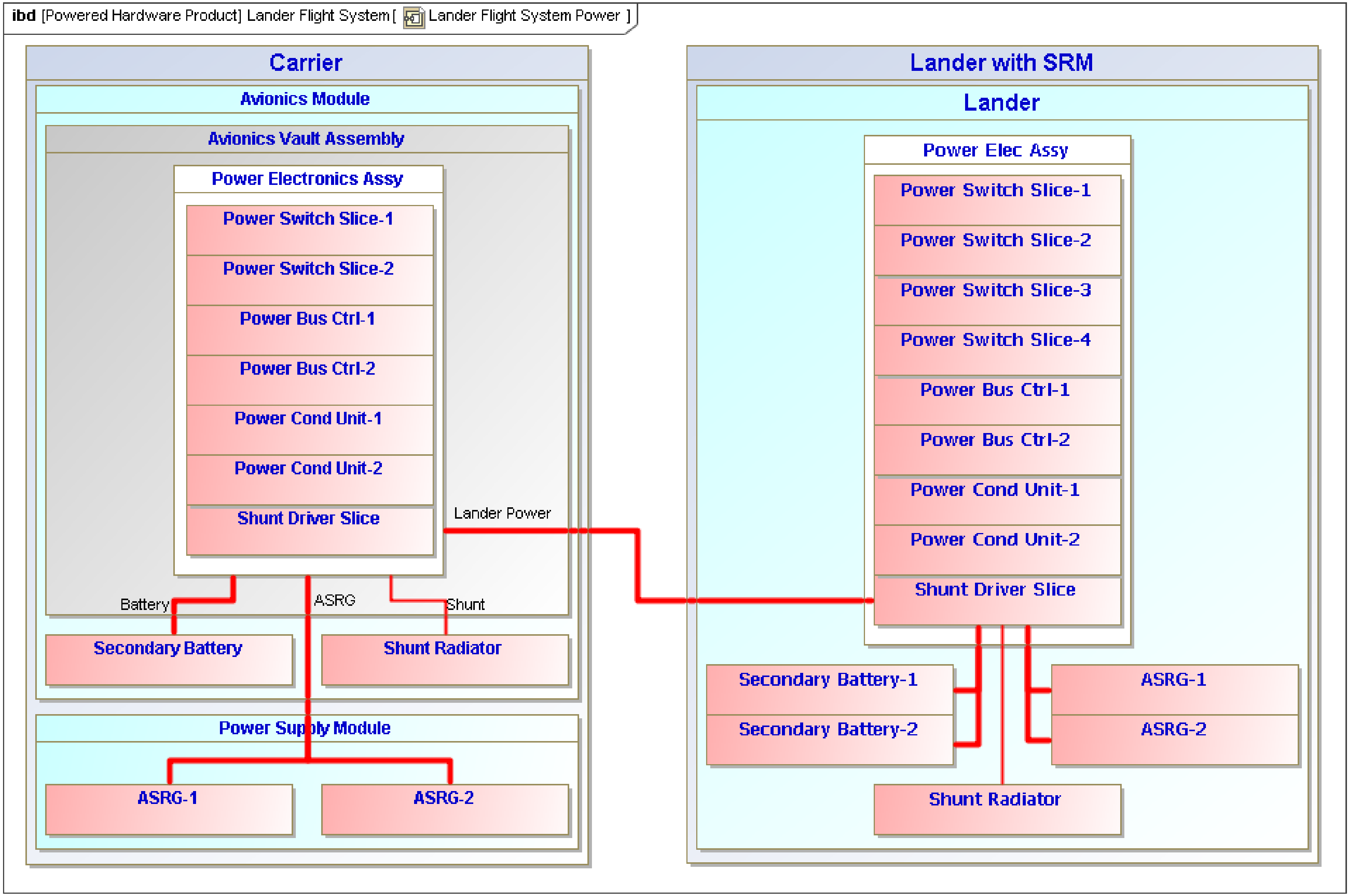
Mass and Power reports are produced *directly* from the model underlying this diagram

Orbiter, Flyby, and Lander

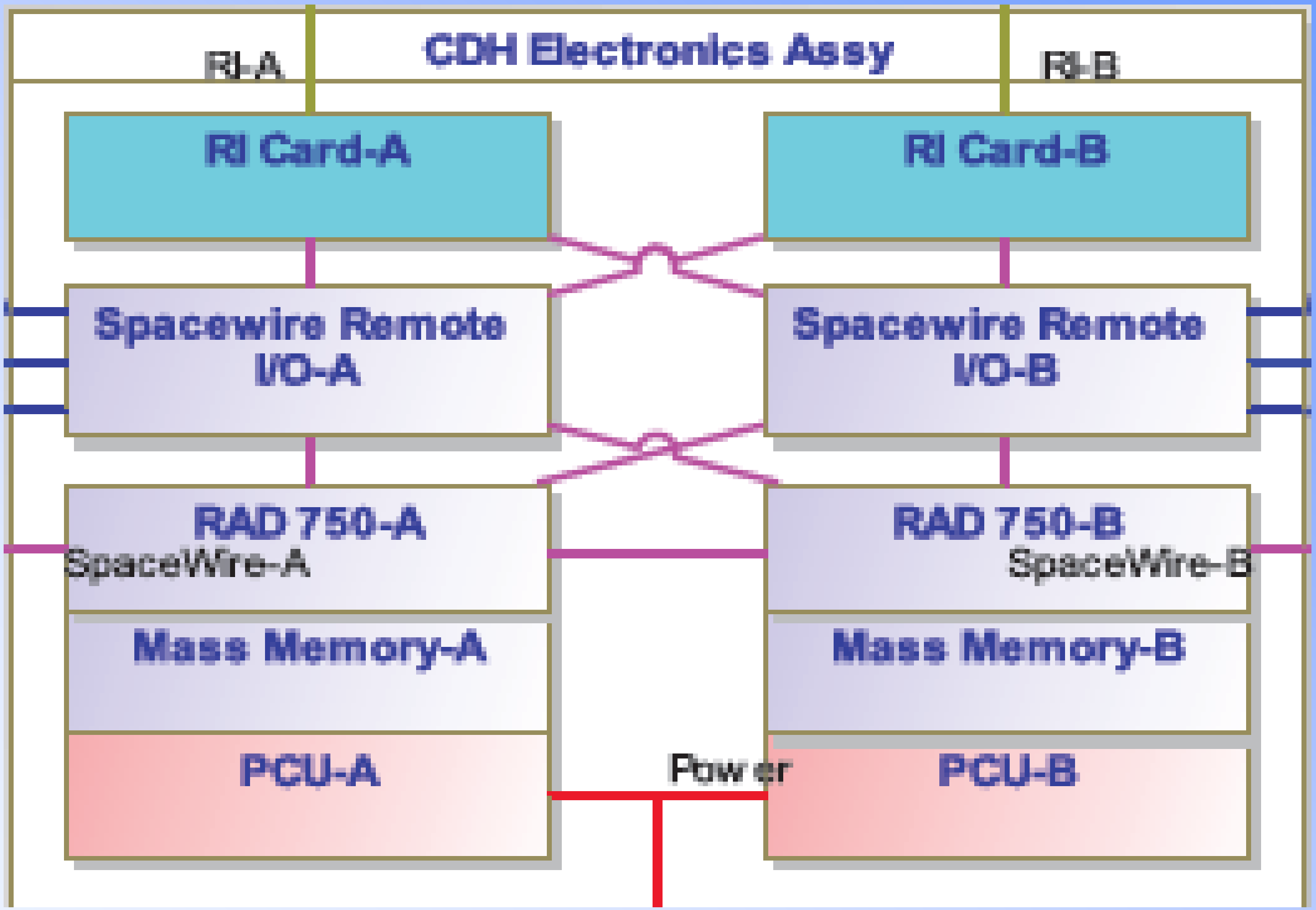


Subsystem Deployment

Example block diagram for power subsystem interconnect



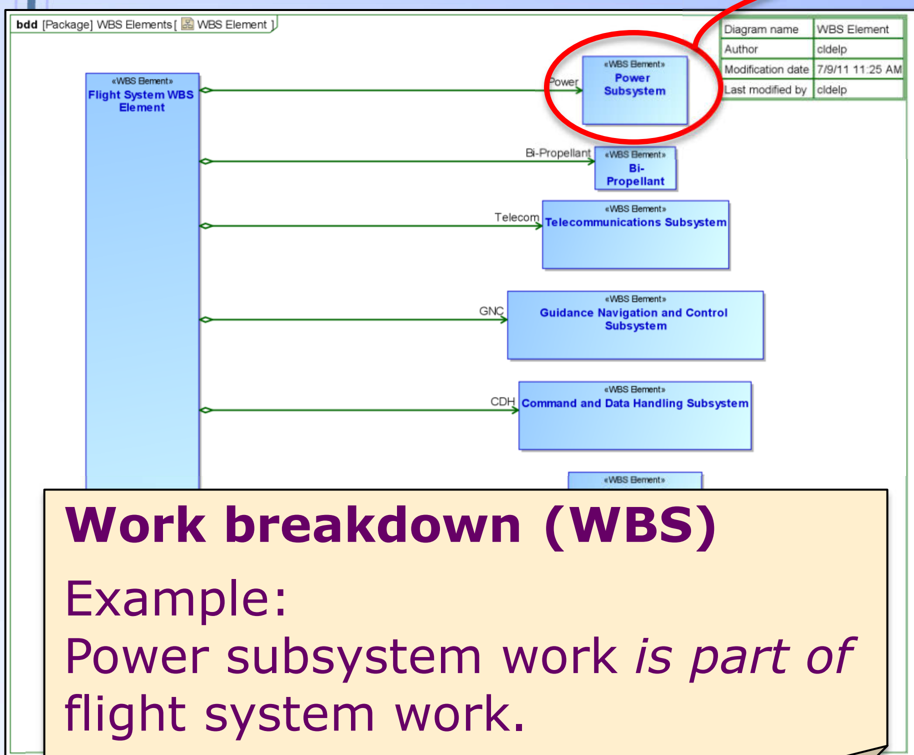
Component Deployment (a.k.a. Box Diagram)



INCOSE International Council on Systems Engineering

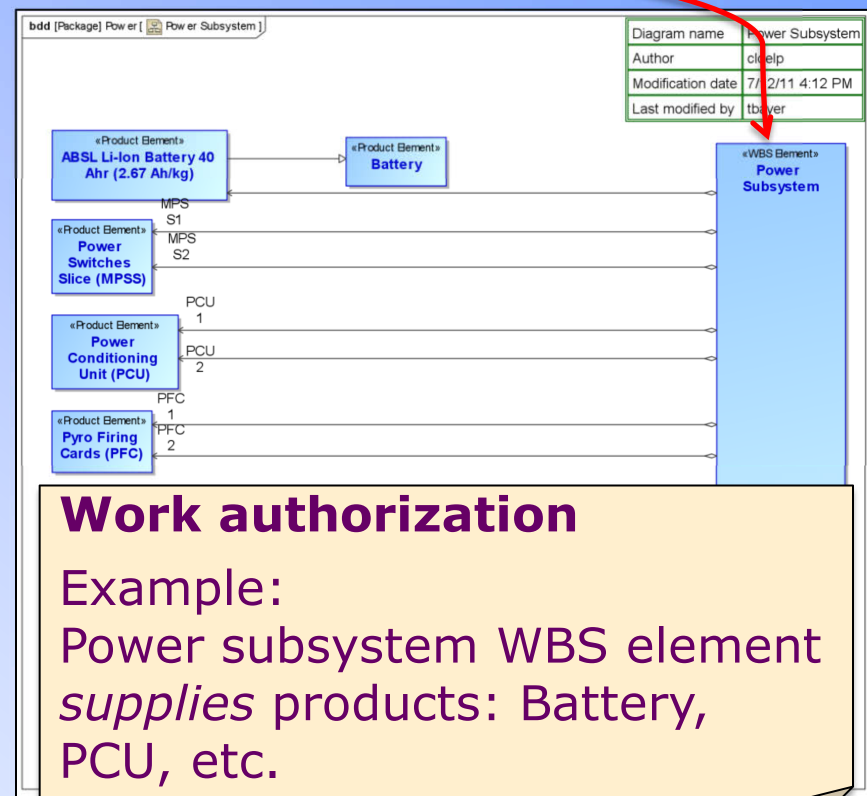
Work Breakdown vs. Product Breakdown

- Subsystems seldom delivered as integrated components
- Better understood as aggregations of convenience, in this case aggregations of delivery responsibility



Work breakdown (WBS)

Example:
Power subsystem work *is part of* flight system work.



Work authorization

Example:
Power subsystem WBS element *supplies* products: Battery, PCU, etc.

Resource Estimation and Tracking

Equipment List and Mass Report

- Collects products from FS Deployment, grouped in 2 ways:
  - by work package (“subsystem”)
  - by physical composition (“assembly”)
- Produced directly from the model
- Enables completeness and correctness checks

	Mass per Unit			Num Of Units	Total Mass			
Products	**CBE (kg)**	**Cont.**	**MEV (kg)**		**CBE (kg)**	**Cont.**	**MEV (kg)**	**WorkPackage**
Flight System	106.4	0.28	136.0					Spacecraft System
Product-1	**2.5**	**0.25**	3.1	2	5	0.25	6.25	Payload System
Product-2	47.5	0.28	60.7	2	95	0.28	121.42	C&DH
Product-3	**1.2**	**0.30**	1.6	1	1.2	0.30	1.56	Payload System
Product-4	**10.4**	**0.25**	13.0	2	20.8	0.25	26	C&DH
Product-5	**25.5**	**0.30**	33.2	1	25.5	0.30	33.15	Power
Product-6	**3.2**	**0.30**	4.2	2	6.4	0.30	8.32	Power

Values illustrative only

Power Consumption Report

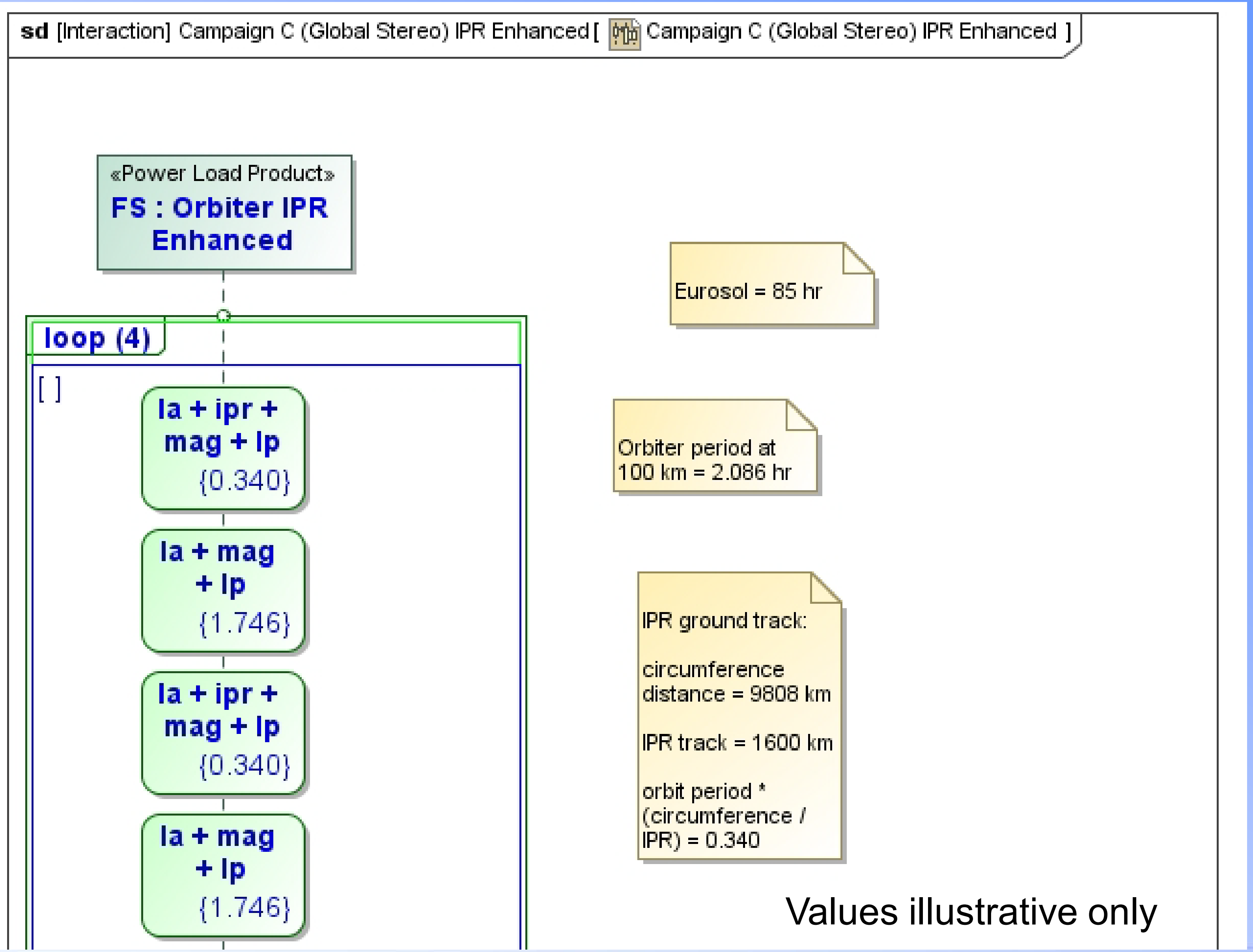
- Similar to mass report
- Can be connected to scenarios for integrated power analysis – separation makes trades easier

Product	Number of Units	Cruise				
		State	Duration [%]	Steady-State Power CBE [W]	Contingency	Steady-State Power MEV [W]
TWTA (TWTA)	1	Off, On		36.8	0.3	47.8
		Off	20.0%	0	0.3	0
		On	80.0%	46	0.3	59.8
SDST (SDST-A)	1	Standby		4	0.3	5.2
SDST (SDST-B)	1	Standby		4	0.3	5.2
Reaction Wheel (RWA)	1	Low Speed		5	0.3	6.5

Values illustrative only

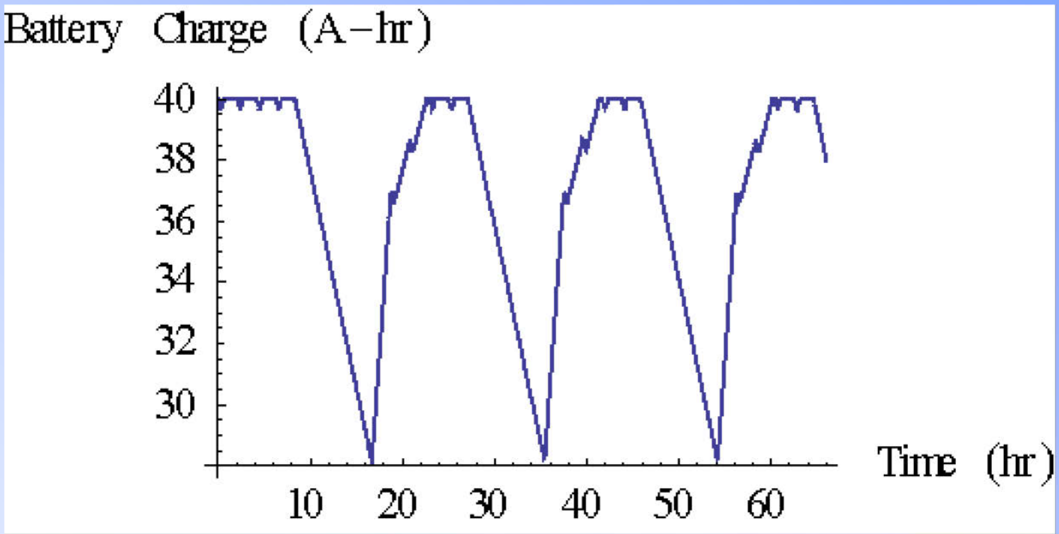
INCOSE
International Council on Systems Engineering

Power Scenario for Science

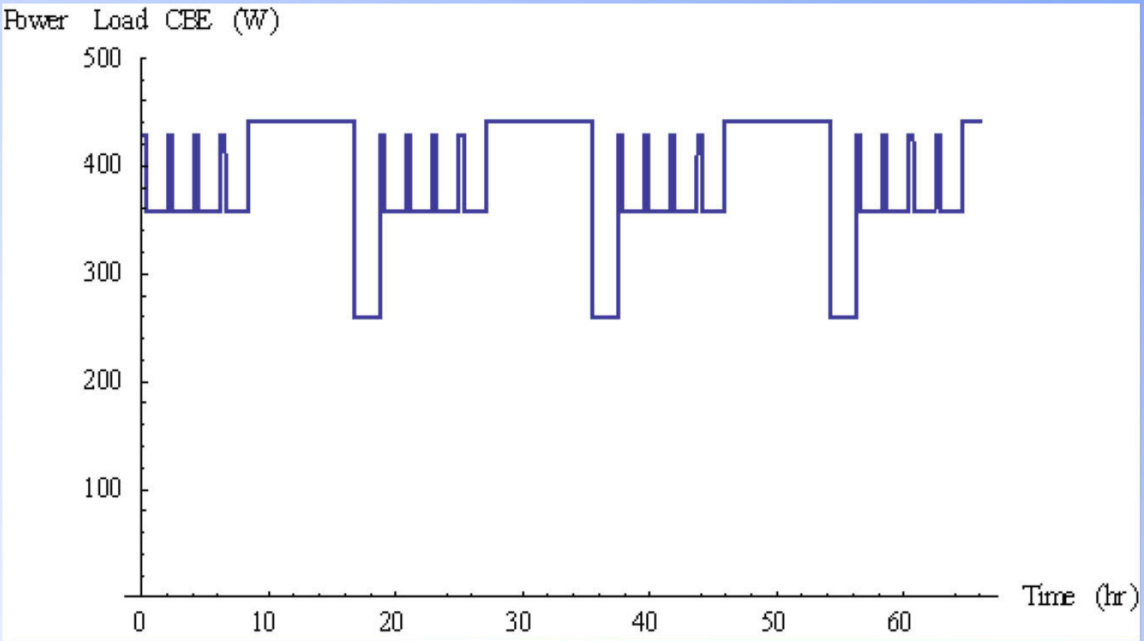


Values illustrative only

Power Results Plotted



Values illustrative only

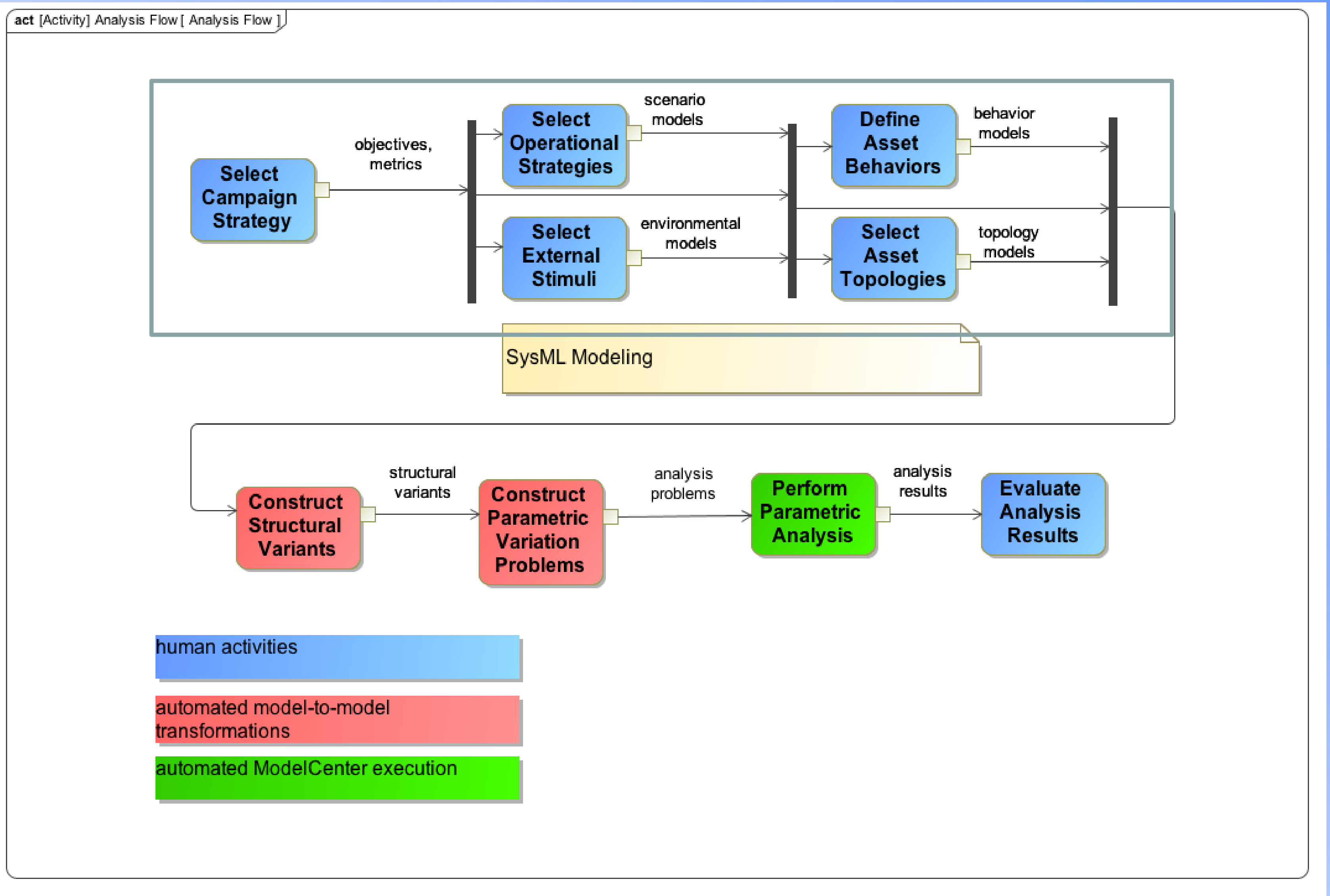


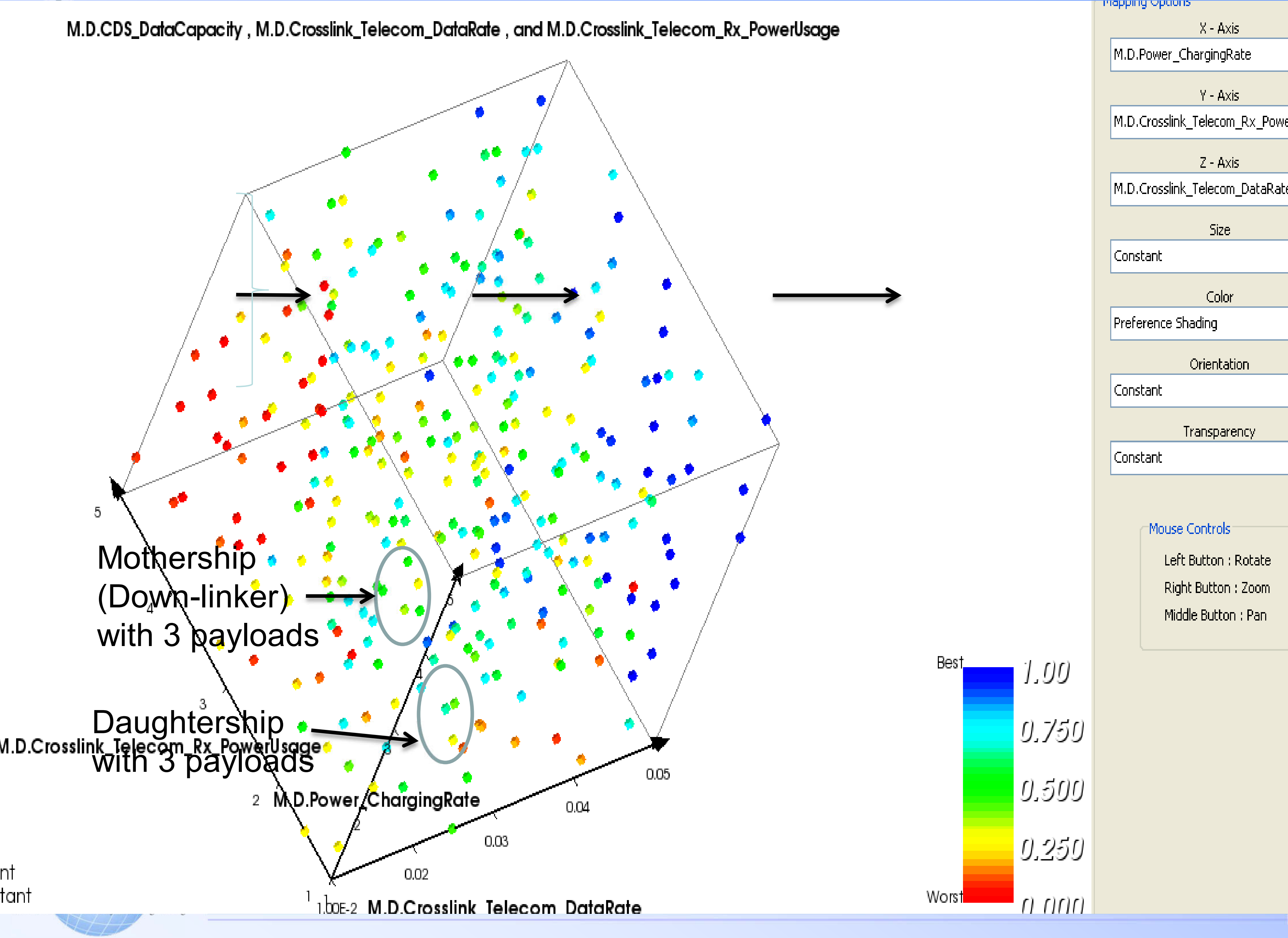
INCOSE
International Council on Systems Engineering

Integrated Analysis

INCOSE
International Council on Systems Engineering

Summary of Model Transformation flow




M.D.CDS_DataCapacity , M.D.Crosslink_Telecom_DataRate , and M.D.Crosslink_Telecom_Rx_PowerUsage
Mothership
(Down-linker)
with 3 payloads
Daughtership
with 3 payloads
M.D.Crosslink_Telecom_Rx_PowerUsage
M.D.Power_ChargingRate
M.D.Crosslink_Telecom_DataRate
1.00E-2
0.02
0.03
0.04
0.05
Best
Worst
1.00
0.750
0.500
0.250
0.000
X - Axis
M.D.Power_ChargingRate
Y - Axis
M.D.Crosslink_Telecom_Rx_Powe
Z - Axis
M.D.Crosslink_Telecom_DataRate
Size
Constant
Color
Preference Shading
Orientation
Constant
Transparency
Constant
Mouse Controls
Left Button : Rotate
Right Button : Zoom
Middle Button : Pan

Summary

Several lessons along the way

- Automated reasoning and analysis require *regular patterns* of modeling
  - Agreements can take a long time to come by
  - Unanimous agreements feel the best, but occasionally a process for decision making with contention is needed (voting, going to higher authority)
- Modeling can get a magical quality
  - Suddenly you're an expert and everyone gets a little uncomfortable about the right way to do things

Several lessons along the way

- Models can give a lot of breathing room in expression
  - Models and diagrams are different
  - Interconnectedness among diagram elements is the real key to trusting that you see the whole by flipping through the parts
- Like any technology, deep tension between institutional and project development
  - Project will take shortcuts and occasionally be short-sighted, but they are the ultimate customer
  - Institutional efforts can quickly become the "ivory tower" or appear unresponsive, but often house the elite expertise

Several lessons along the way

- It's hard, but you have to stay in the room
  - This advice directed at modelers that may not be domain experts
  - You become the translator and should express as much as you can in the foreign language of the domain
  - Being seen to do "real work" builds credibility and trust immeasurably
  - Challenges to the "slop" in domain methods should be mounted, but very strategically

- The payoff is seeing it all come together

Claims looking good so far

- Capture system “views” relative to stakeholder perspectives
- Ensure a consistent view of the developing system as close to real-time as possible
- Reduce time to integrate disparate subsystem concepts to “close” the design within project constraints
- Early identification of requirements issues
- Enhanced system design integrity

Claims looking good so far

- Reduction in unintended behaviors/outcomes
- Improved specification of allocated req'ts to HW/SW
- Fewer errors during I&T
- More rigorous requirements traceability
- Rich database permits multi-user input and immediate synchronization, improving efficiency and productivity
- Use of a single data environment results in data availability throughout program life-cycles

Claims looking good so far

- Traceability through model elements enables efficient change / impact analysis enabling a more adaptable system
- Robust query engine allows rapid assessment of the integrated database, finding anomalies early, preventing rework
- Improved impact analysis of requirements changes
- Increased trade space
- Model reuse to support design/technology evolution
- Auto-generation of documentation

Overall

- Using MBSE to do SE work provides some noticeable advantages
- MBSE workflows are on a different rhythm than the traditional ones
- Modelers tend to desire much more precision than traditional SE's – not always a bad thing but sometimes lose the forest for the trees
- Many of the early promises seem to be panning out
- Many of them also appear to have a lot of wrinkles and "yes, buts"
  - The secret sauce will be in knowing and working with these

INCOSE
International Council on Systems Engineering

And that's the story

INCOSE
International Council on Systems Engineering