Executable SysML Models for the Thirty Meter Telescope Analysis

Nerijus Jankevičius, MODELS 2016

Speaker

- **Nerijus Jankevičius**, nerijus@nomagic.com
- *Product Manager @ No Magic Europe*
- *Since 1997*

- Leads the development of MBSE tools and solutions

- Consulting companies such as NASA/JPL, ESO, BAE Systems, Kongsberg Defense and Aerospace, Nokia, Bernafon, GE Transportation, Bombardier Transportation, Pratt & Whitney, MITRE and others.

- OMG member since 2004
- INCOSE member since 2007
- UML and SysML Revision Task Force member

Meet No Magic

4000+ trainings

1000 000+ installations

10 000 companies

in **90** countries

magicdraw

No Magic MBSE Ecosystem

They Trust Us

LOCKHEED MARTIN

NASA

JPL

NOKIA

Pfizer

Ford

U.S.ARMY

SIEMENS

Microsoft

NORTHROP GRUMMAN

Abbott

BMW

AIRBUS

Raytheon

GENERAL DYNAMICS

GE

GENERAL ATOMICS

BAE SYSTEMS

INSPIRED WORK

esa

Pratt & Whitney

A United Technologies Company

KONGSBERG

Sandia National Laboratories

BOMBARDIER

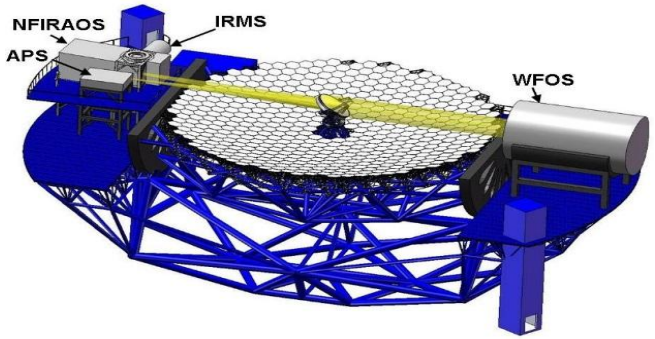
ThyssenKrupp

BOSCH

Thirty Meter Telescope (TMT)

- Developed by TMT International Observatory (TIO)
  - NASA JPL participates in several subsystems of TMT
  - No Magic is a MBSE solution provider

Alignment and Phasing System (APS)



Using MBSE for

- Defining requirements
- System decomposition
- Interfaces and relationships between subsystems
- Functions and behaviors of components
- Operational scenarios (use cases)

Typical Analysis Activities

- Analyze associated scenarios
- Automatically verify system requirements are met
- Develop/refine timing requirements for algorithms, internal and external interface commands
- Evaluate design alternatives
- Help decision making

MagicDraw + Cameo Simulation Toolkit

- Model execution framework and infrastructure
  - Model debugging and animation environment
  - Pluggable engines, languages and evaluators
  - User Interface prototyping support
  - Model driven configs and test cases

- The standard based model execution of:
  - Activities (OMG fUML standard)
  - Composite structures (OMG PSCS)
  - Statemachines (W3C SCXML standard)
  - Actions/scripts (ALF, JSR223)
  - Parametrics (OMG SysML standard)
    - Matlab, Modelica, Mathematica, Maple, FMI
  - Sequence diagrams (OMG UML Testing Profile)

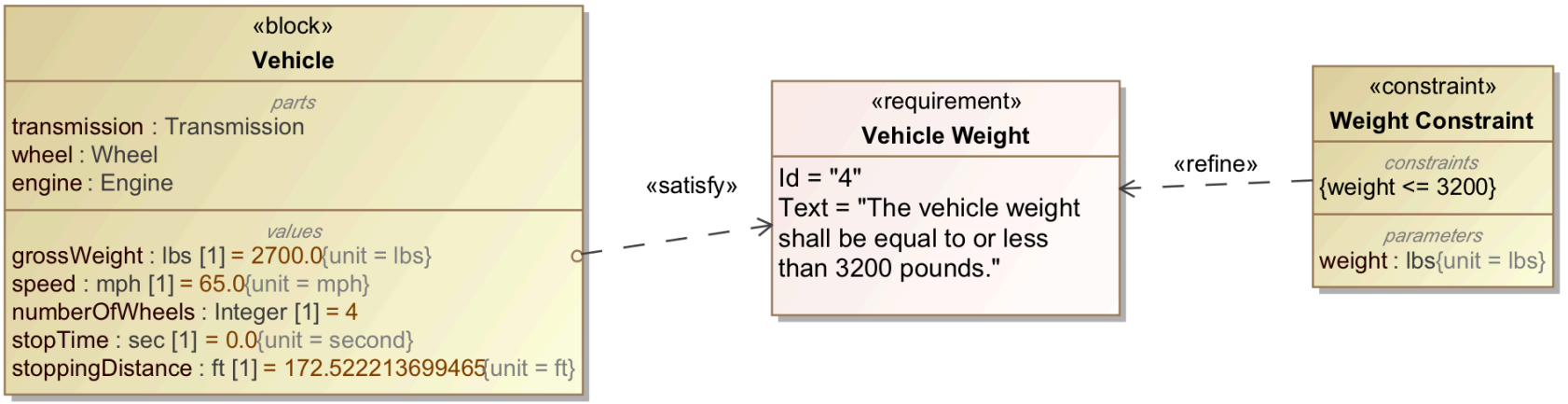
OBJECT MANAGEMENT GROUP

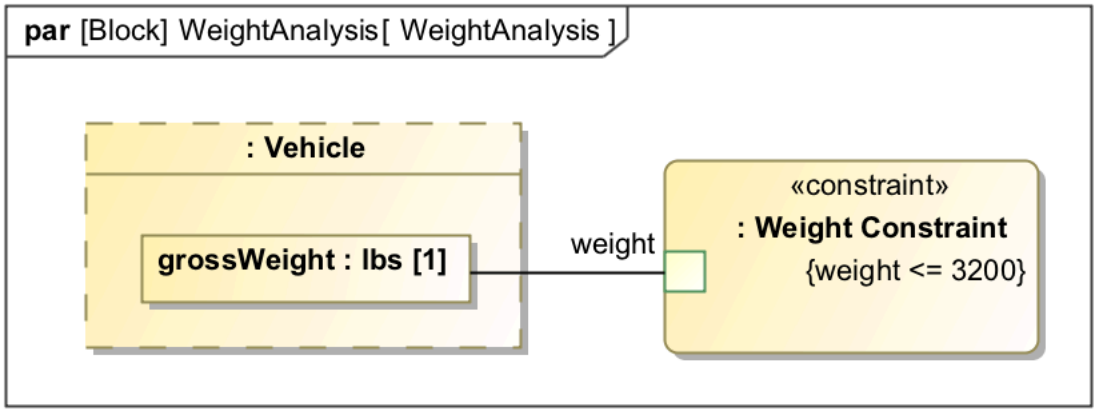
Executable System Engineering Method (ESEM)

Use vanilla UML/SysML - no extensions or markups

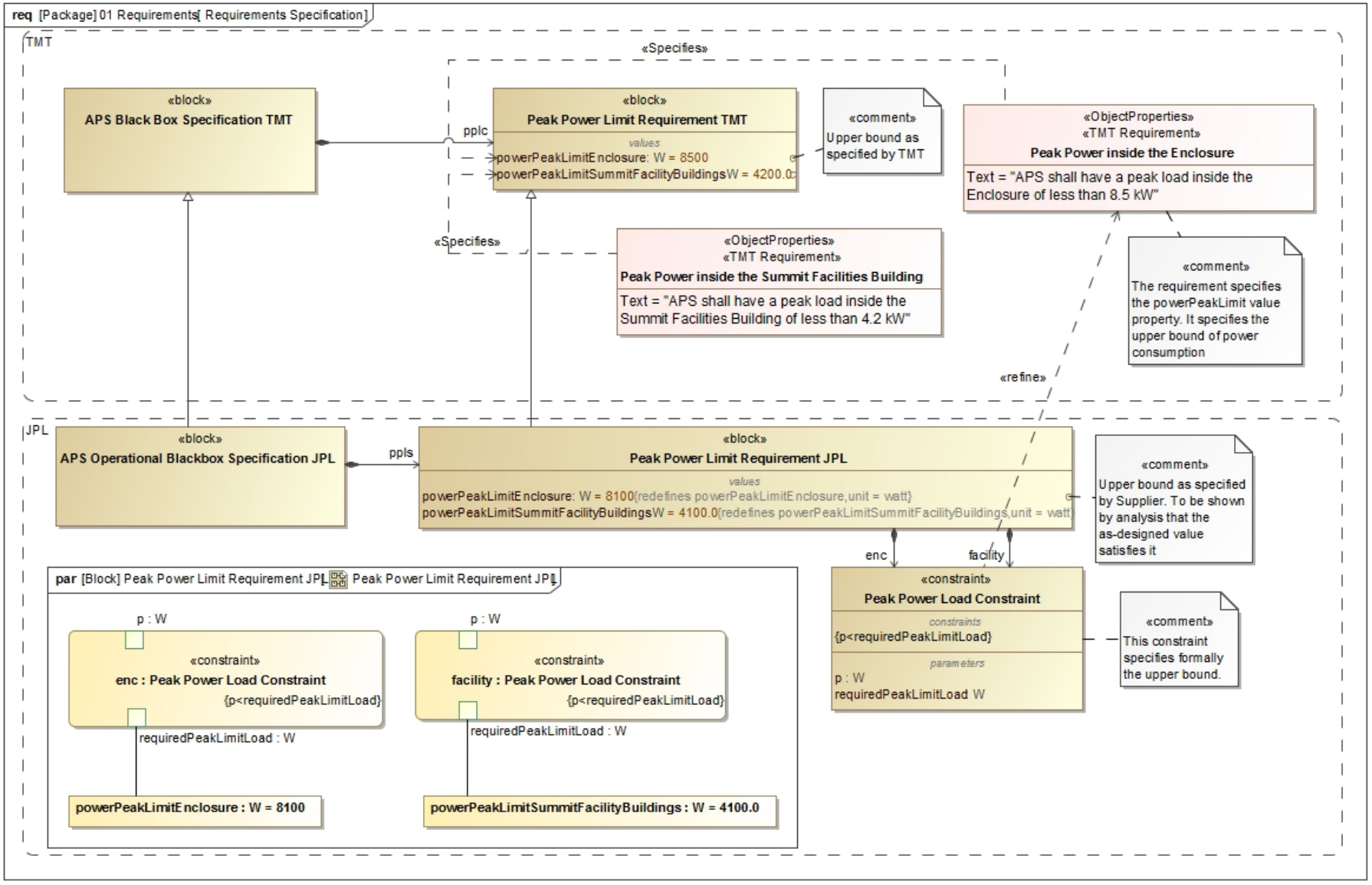
Step 1: Formalize Requirements
Step 2: Specify Design
Step 3: Characterize Components
Step 4: Specify Analysis Context
Step 5: Specify Operational Scenarios
Step 6: Specify Analysis Configurations
Step 7: Run Analysis

Step 1: Formalize Requirements
Satisfy, refine, bind

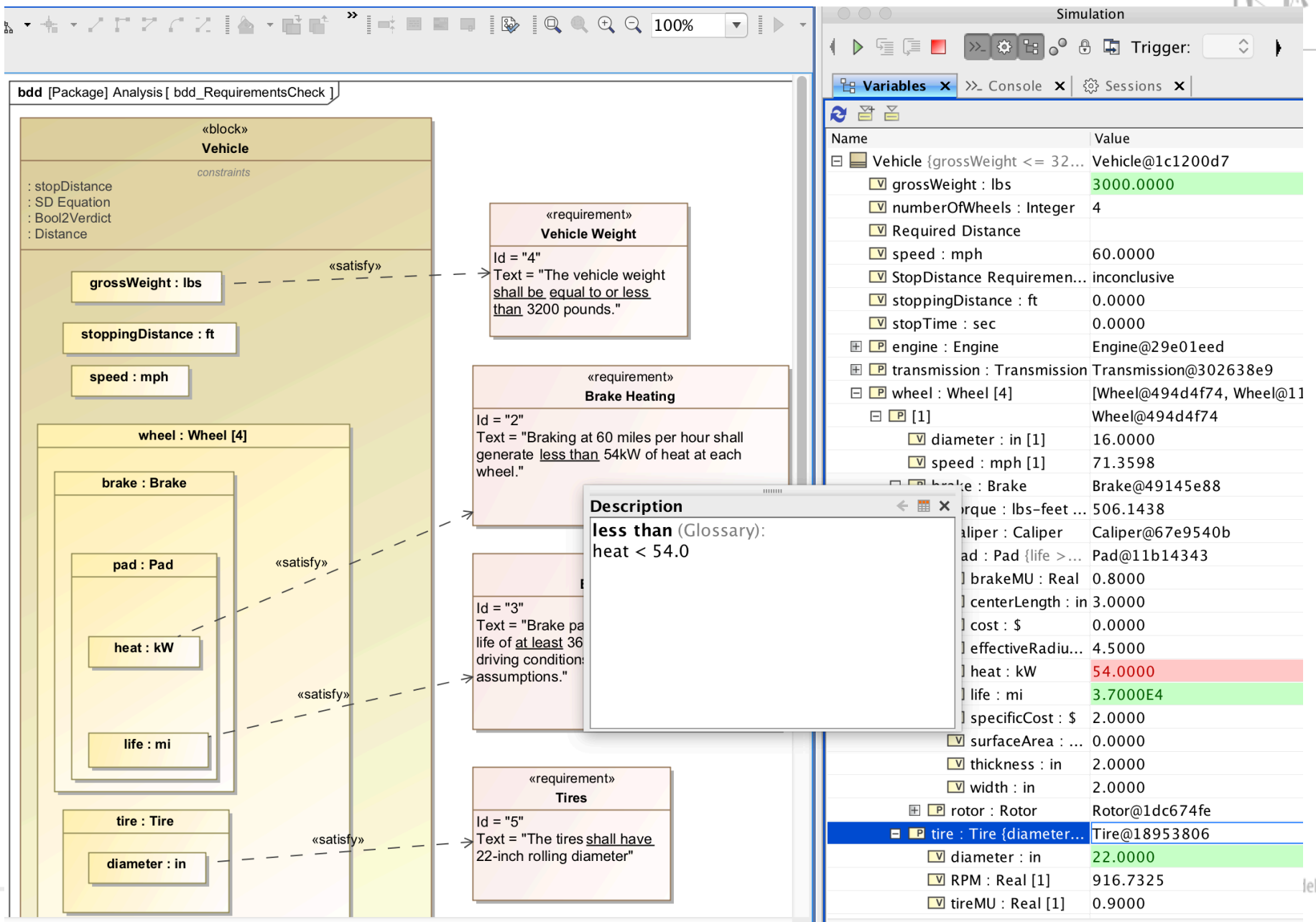




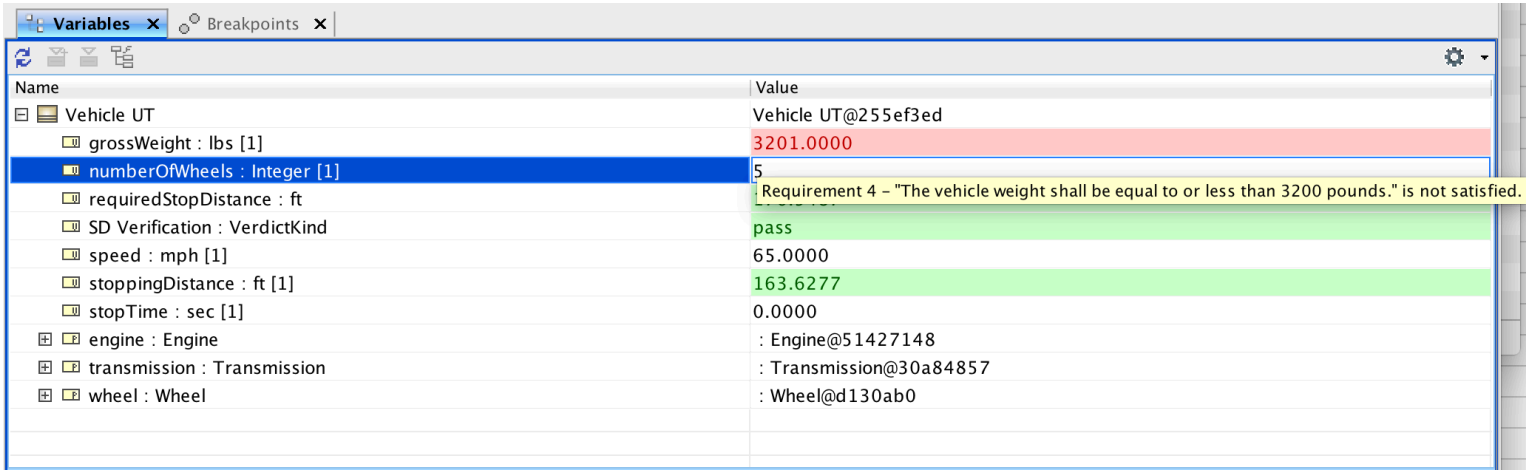
Step 1: Formalize Requirements



Requirements verification



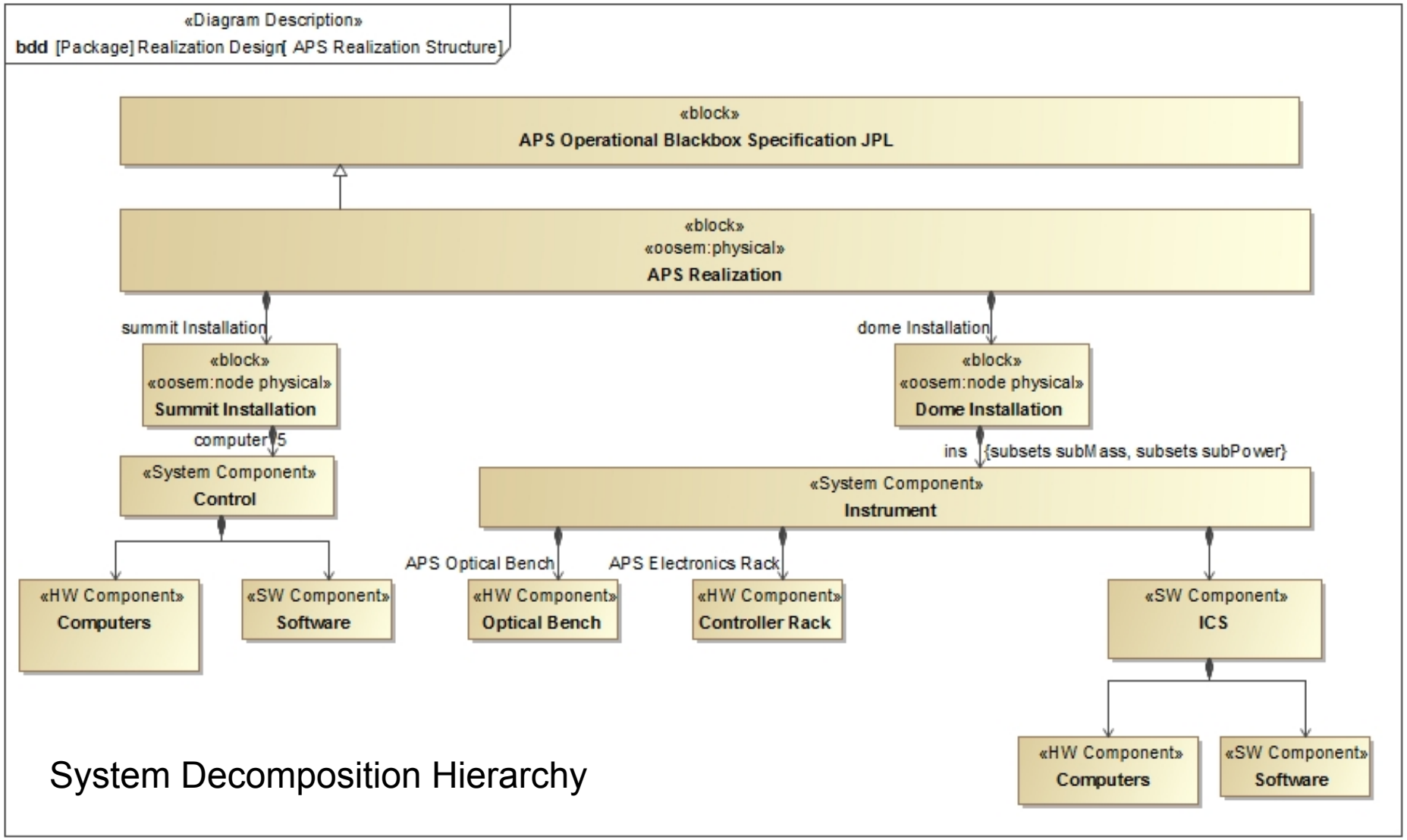
Verification results



#	Name	Gross Weight : Lbs	Speed : Mph	Number Of Wheels : Integer	Stopping Distance : Ft	Required Stop Distance : Ft	SD Verification : Verdict Kind	: Weight Constraint	: SD Constraint
1	vh1	2800.0	65.0	4	178.9119253179637	176.54869701553358	fail	pass	fail
2	vh2	3300.0	65.0	5	168.68838672836577	176.54869701553358	pass	fail	pass
3		3200.0	65.0	4	204.4707717919585	176.54869701553358	fail	pass	fail
4		2700.0	65.0	4	172.522213699465	176.54869701553358	pass	pass	pass
5		2700.0	65.0	4	172.522213699465	176.54869701553358	pass	pass	pass
6	vehicle UT	3500.0	65.0	4	223.6399066474546	176.54869701553358	fail	fail	fail
7	vehicle UT1	2700.0	65.0	4	172.522213699465	176.54869701553358		pass	pass

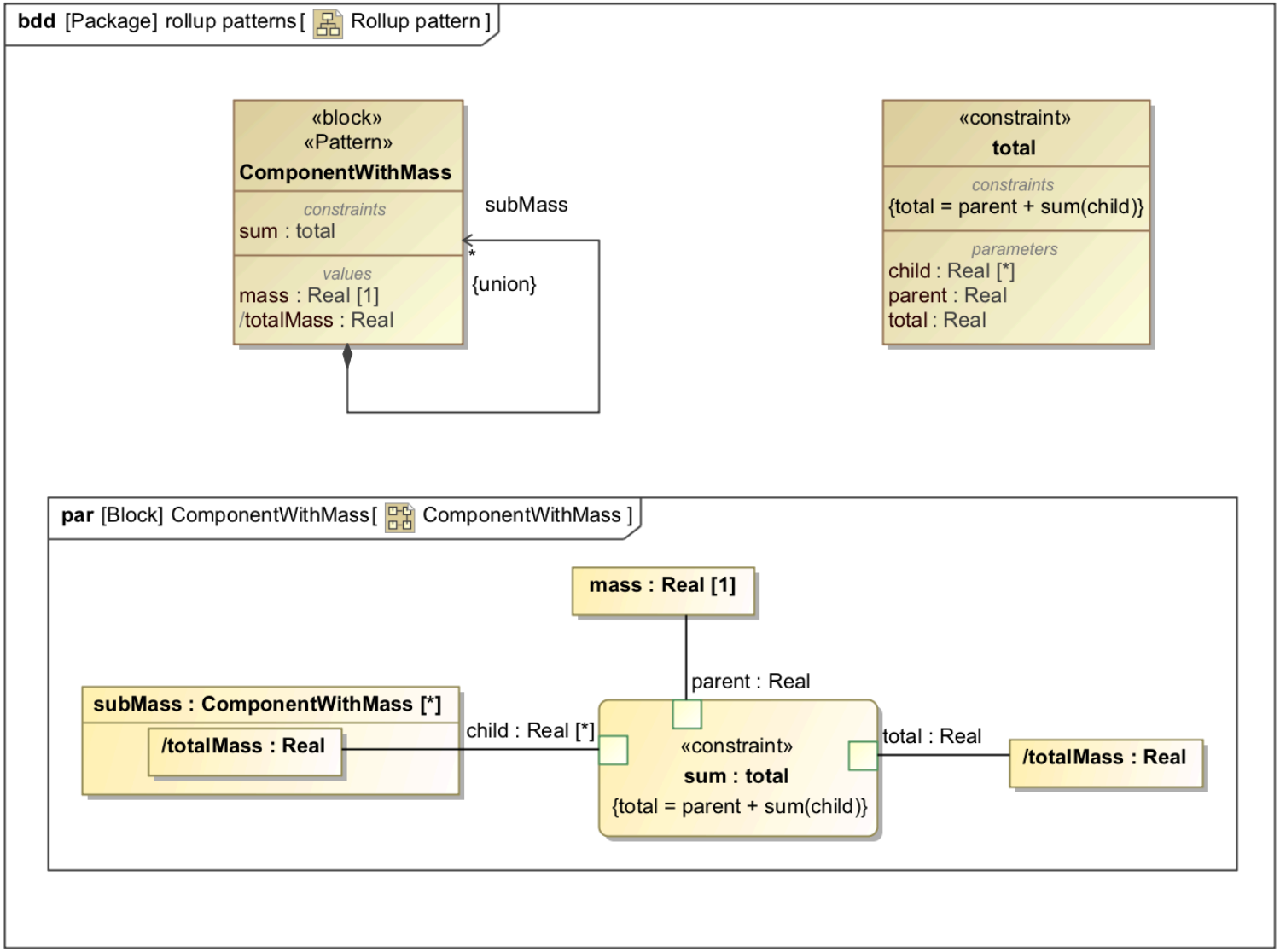
2	Gross Weight : Lbs	Speed : Mph	Number Of Wheels : Integer	Stopping Distance : Ft	Required Stop Distance : Ft	SD Verification : Verdict Kind	: Weight Constraint	: SD Constraint
3	2800	65	4	178.9119253	176.548697	fail	pass	fail
4	3300	65	5	168.6883867	176.548697	pass	fail	pass
5	3200	65	4	204.4707718	176.548697	fail	pass	fail
6	2700	65	4	172.5222137	176.548697	pass	pass	pass
7	2700	65	4	172.5222137	176.548697	pass	pass	pass
8	3500	65	4	223.6399066	176.548697	fail	fail	fail
9	2700	65	4	172.5222137	176.548697		pass	pass

Step 2: Decomposition

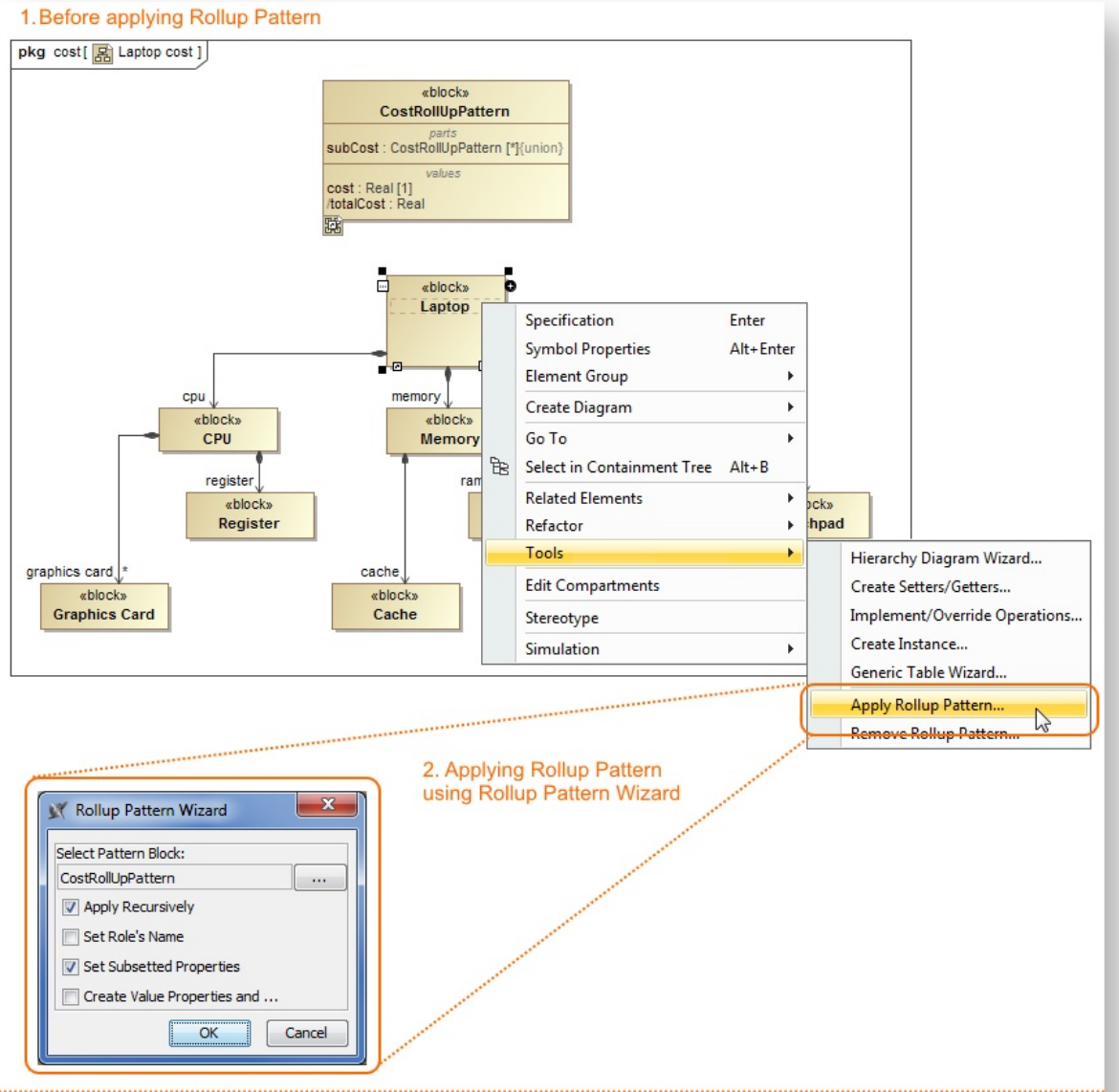


System Decomposition Hierarchy

Step 3: Characterize Components (parametric rollup pattern)



Automation


1. Before applying Rollup Pattern
pkg cost [Laptop cost]
«block»
CostRollUpPattern
parts
subCost : CostRollUpPattern [*]{union}
values
cost : Real [1]
/totalCost : Real
«block»
Laptop
cpu
«block»
CPU
register
«block»
Register
memory
«block»
Memory
graphics card *
«block»
Graphics Card
cache
«block»
Cache
Specification Enter
Symbol Properties Alt+Enter
Element Group
Create Diagram
Go To
Select in Containment Tree Alt+B
Related Elements
Refactor
Tools
Edit Compartments
Stereotype
Simulation
Hierarchy Diagram Wizard...
Create Setters/Getters...
Implement/Override Operations...
Create Instance...
Generic Table Wizard...
Apply Rollup Pattern...
Remove Rollup Pattern...
2. Applying Rollup Pattern
using Rollup Pattern Wizard
Rollup Pattern Wizard
Select Pattern Block:
CostRollUpPattern
Apply Recursively
Set Role's Name
Set Subsetted Properties
Create Value Properties and ...
OK
Cancel

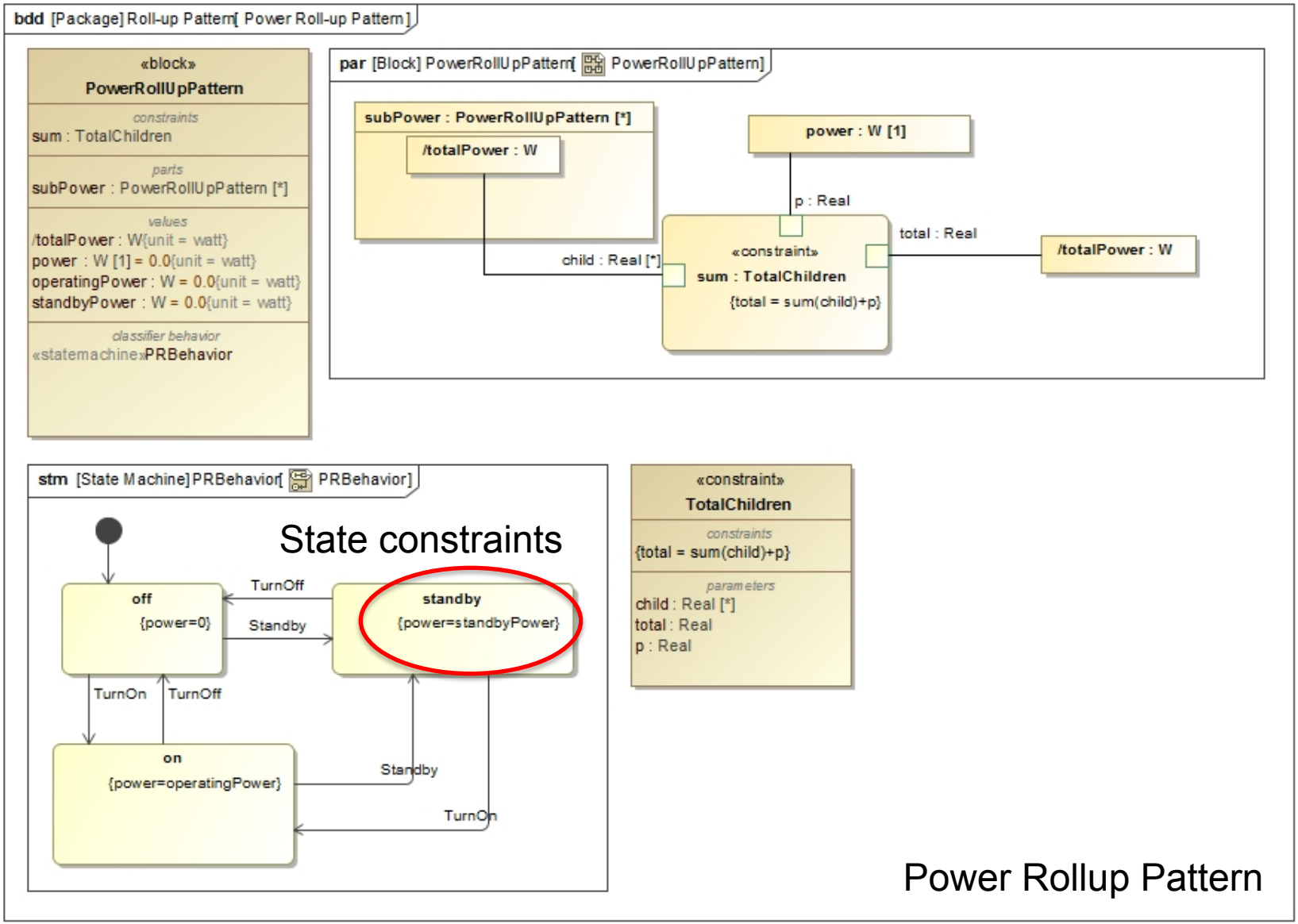
Configuration (composite instance table)

#	Name	mass : kg	totalPower : W	power : W	operatingPower : W	standbyPower : W
1	aps.dome installation_1.bench_1_imported	2678.1	600.0	0.0	0.0	0.0
2	aps_imported	0.0	0.0	0.0	0.0	0.0
3	aps.coordinator_1_imported	0.0	0.0	0.0	0.0	0.0
4	aps.dome installation_1_imported	0.0	600.0	0.0	0.0	0.0
5	aps.dome installation_1.bench_1.apt + te ccd_1_imported	0.0	300.0	300.0	300.0	100.0
6	aps.dome installation_1.bench_1.large motor_1[1]_imported	0.0	0.0	0.0	10.0	10.0
7	aps.dome installation_1.bench_1.large motor_1[2]_imported	0.0	0.0	0.0	10.0	10.0
8	aps.dome installation_1.bench_1.large motor_1[3]_imported	0.0	0.0	0.0	10.0	10.0
9	aps.dome installation_1.bench_1.large motor_1[4]_imported	0.0	0.0	0.0	10.0	10.0
10	aps.dome installation_1.bench_1.large motor_1[5]_imported	0.0	0.0	0.0	10.0	10.0
11	aps.dome installation_1.bench_1.large motor_1[6]_imported	0.0	0.0	0.0	10.0	10.0
12	aps.dome installation_1.bench_1.large motor_1[7]_imported	0.0	0.0	0.0	10.0	10.0
13	aps.dome installation_1.bench_1.large motor_1[8]_imported	0.0	0.0	0.0	10.0	10.0
14	aps.dome installation_1.bench_1.large motor_1[9]_imported	0.0	0.0	0.0	10.0	10.0
15	aps.dome installation_1.bench_1.large motor_1[10]_imported	0.0	0.0	0.0	10.0	10.0
16	aps.dome installation_1.bench_1.lowfs + te ccd_1_imported	0.0	150.0	150.0	150.0	100.0
17	aps.dome installation_1.bench_1.pit + te ccd_1_imported	0.0	150.0	150.0	150.0	100.0
18	aps.dome installation_1.bench_1.sh ccd_1_imported	0.0	0.0	0.0	150.0	100.0
19	aps.dome installation_1.bench_1.small motor_1[1]_imported	0.0	0.0	0.0	10.0	10.0
20	aps.dome installation_1.bench_1.small motor_1[2]_imported	0.0	0.0	0.0	10.0	10.0
21	aps.dome installation_1.bench_1.small motor_1[3]_imported	0.0	0.0	0.0	10.0	10.0
22	aps.dome installation_1.bench_1.small motor_1[4]_imported	0.0	0.0	0.0	10.0	10.0

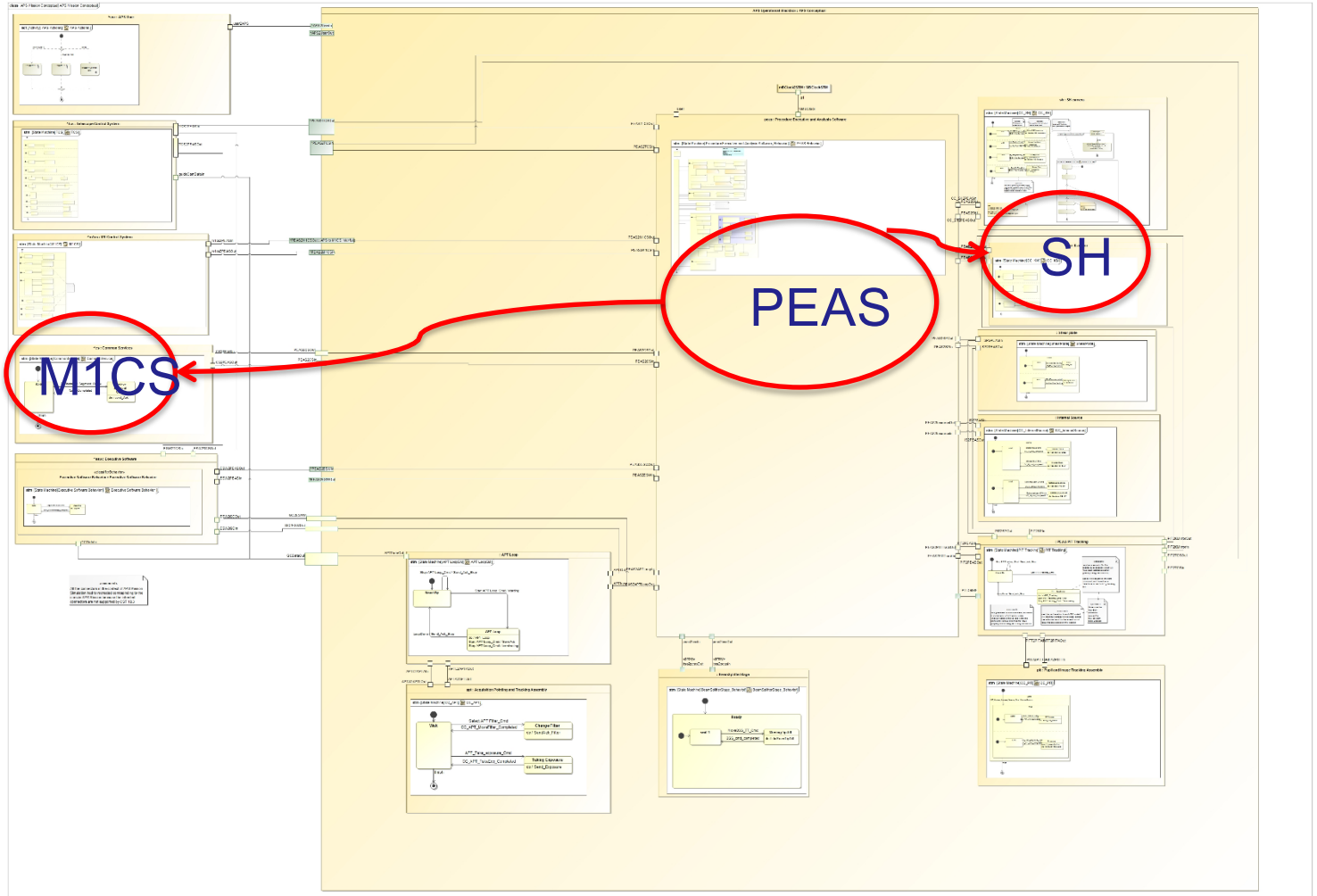
#	Name	Ma : Mass[kilogram]	Me : Mass[kilogram]	Mr : Mass[kilogram]	Margin : Mass[kilogram]
1	spacecraft	130.0 kg	95.0 kg	15.0 kg	35.0 kg
2	telecom	35.0 kg	27.0 kg	5.0 kg	8.0 kg
3	amplifier	10.0 kg	8.0 kg	10.0 kg	2.0 kg
4	antenna	20.0 kg	19.0 kg	20.0 kg	1.0 kg
5	propulsion	80.0 kg	68.0 kg	7.0 kg	12.0 kg
6	tank	44.0 kg	38.0 kg	44.0 kg	6.0 kg
7	thruster	29.0 kg	30.0 kg	29.0 kg	-1.0 kg

Requirement 1 - "Estimated mass shall be less than allocated mass" is not satisfied.

Dynamic rollup pattern

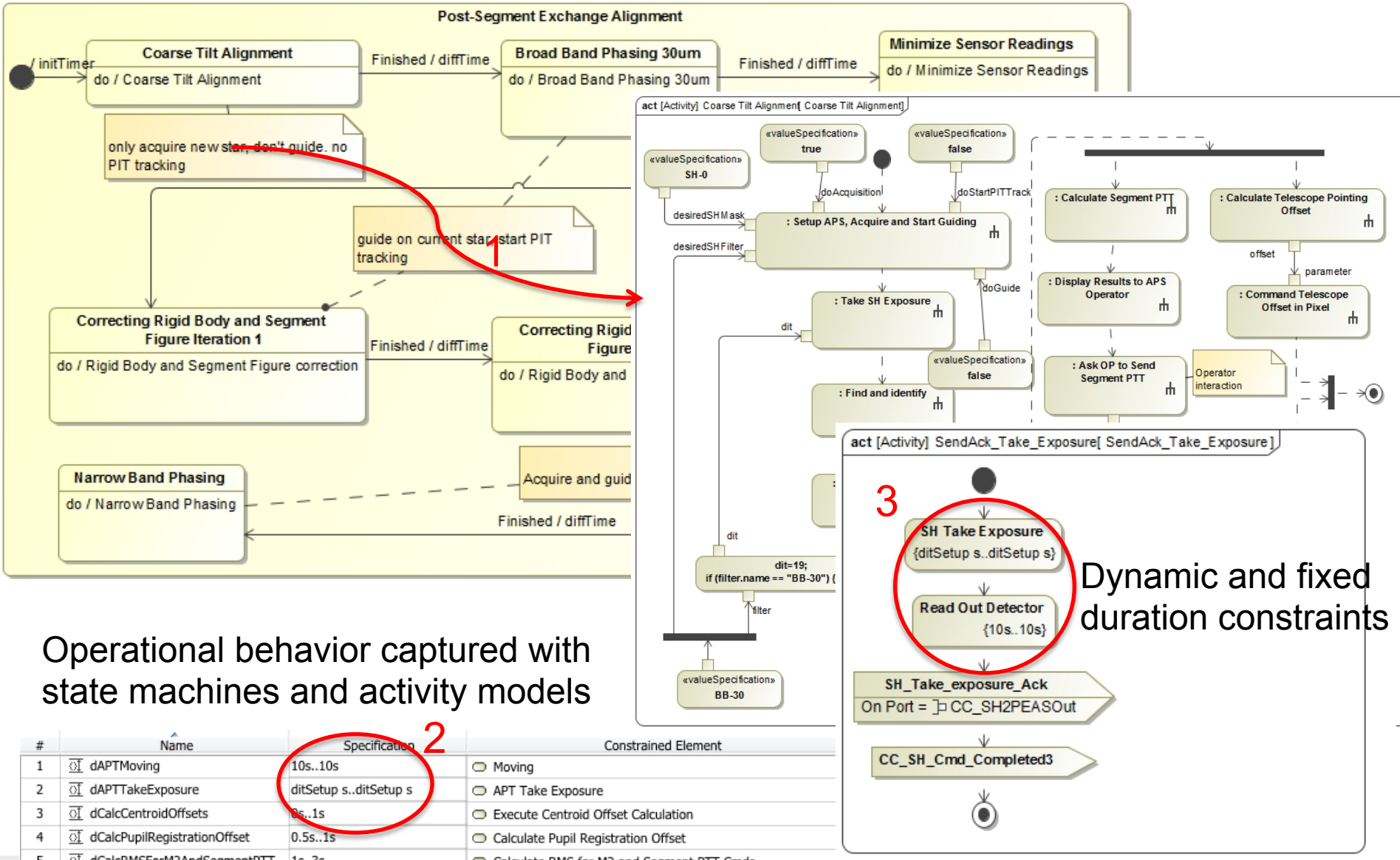


Communication among components



Communication between state machine specified components over ports

Operational behavior



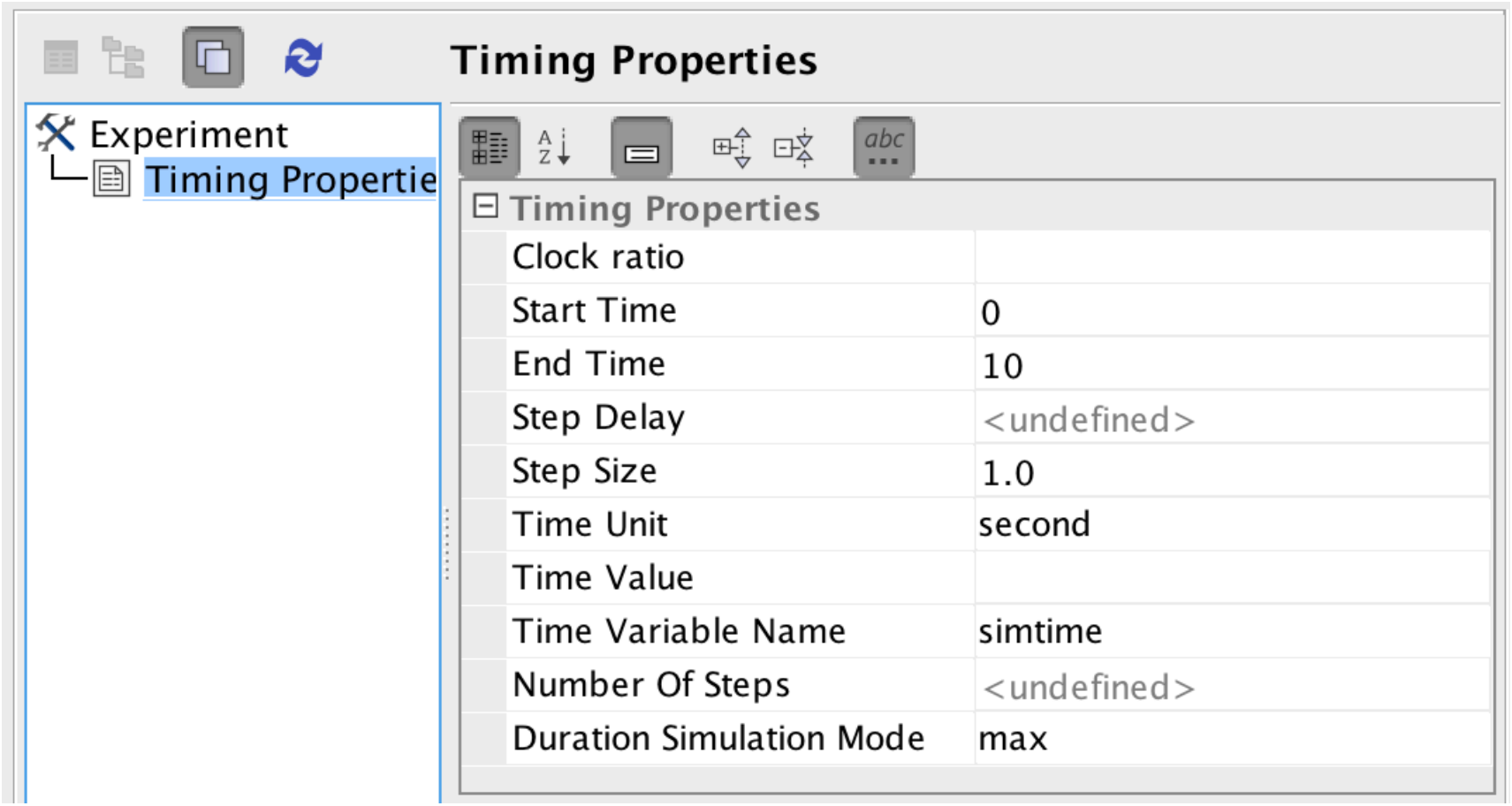
Dynamic and fixed duration constraints

Operational behavior captured with state machines and activity models

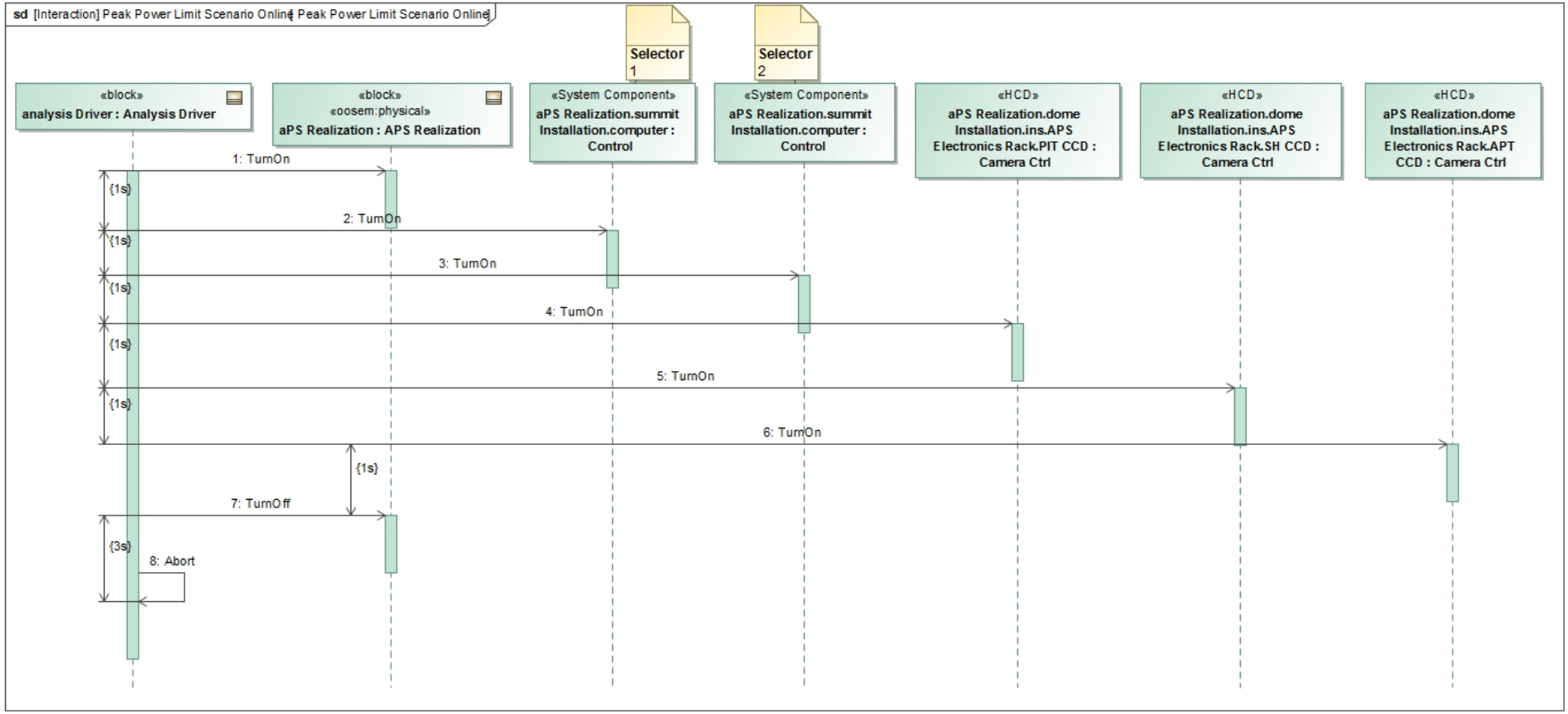
2

#	Name	Specification	Constrained Element
1	dAPTMoving	10s..10s	Moving
2	dAPTTakeExposure	ditSetup s..ditSetup s	APT Take Exposure
3	dCalcCentroidOffsets	0s..1s	Execute Centroid Offset Calculation
4	dCalcPupilRegistrationOffset	0.5s..1s	Calculate Pupil Registration Offset
5	dCalcRMSForM2AndSegmentPTT	1s..3s	Calculate RMS for M2 and Segment PTT Cmds

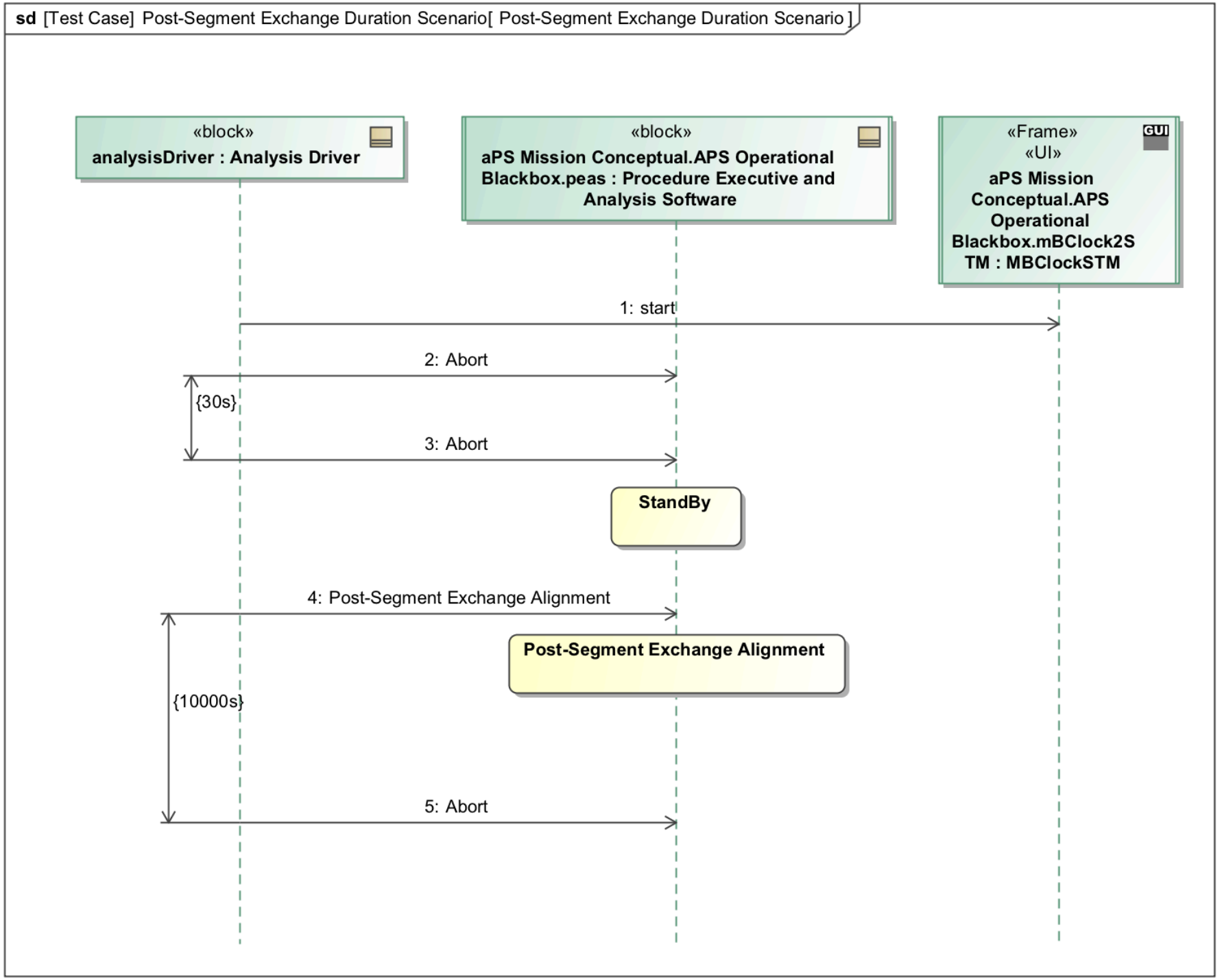
Clock settings



Step 5: Specify Operational Scenarios



Operational Scenario Driver


sd [Test Case] Post-Segment Exchange Duration Scenario[Post-Segment Exchange Duration Scenario]
«block»
analysisDriver : Analysis Driver
«block»
aPS Mission Conceptual.APS Operational Blackbox.peas : Procedure Executive and Analysis Software
«Frame»
«UI»
GUI
aPS Mission Conceptual.APS Operational Blackbox.mBClock2S TM : MBClockSTM
1: start
2: Abort
{30s}
3: Abort
StandBy
4: Post-Segment Exchange Alignment
Post-Segment Exchange Alignment
{10000s}
5: Abort

Step 6: Specify Scenario Configurations

Scenario Condition Pattern

- A decomposition tree of instance specifications representing the state of the scenario
  - Can be presented in tabular form
    - Rows represent the instance specifications (e.g., component)
    - Columns represent values (e.g., operating power) from the instance specifications

Issues

- Hard to keep instance specifications in sync with Block hierarchy
  - Mitigation: tool automation
- Instance specifications cannot be displayed in IBDs
  - Mitigation: use full specialization tree of singleton Blocks for each scenario

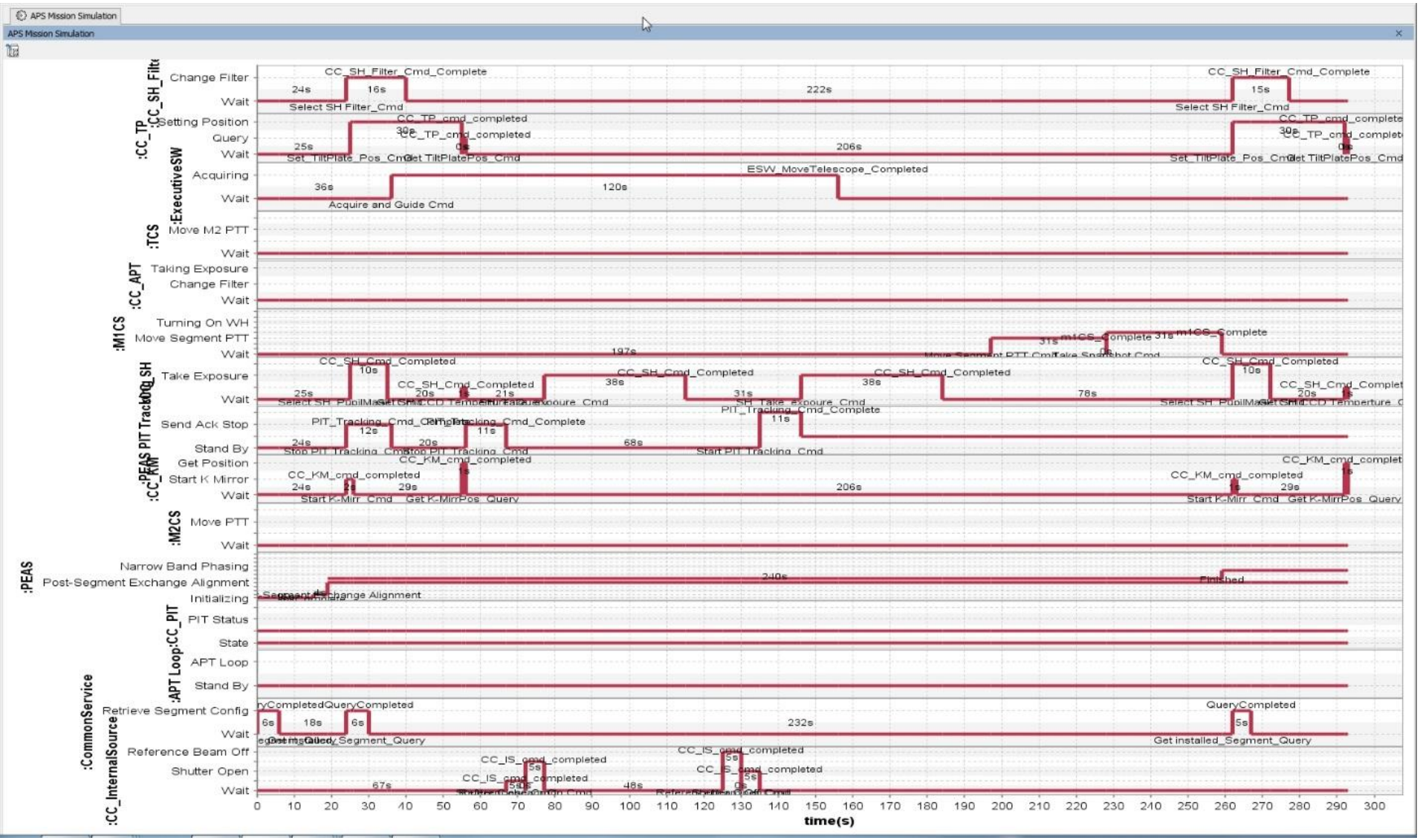
Step 6: Specify Analysis Configurations

#	Name	Classifier	Operating Power : W	Standby Power : W
1	peak Power Limit Scenario Online.aPS Realization	APS Realization	0.0	0.0
2	peak Power Limit Scenario Online.aPS Realization.dome Installation	Dome Installation	0.0	0.0
3	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins	Instrument	0.0	0.0
4	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack	Controller Rack	0.0	0.0
5	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt bs	Motor Ctrl	0.0	0.0
6	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt bs.subMass[1]	MassRollUpPattern		
7	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt bs.subPower[1]	PowerRollUpPattern		
8	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt ccd	Camera Ctrl	150.0	200.0
9	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt ccd.subMass[1]	MassRollUpPattern		
10	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt ccd.subPower[1]	PowerRollUpPattern		
11	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt filter 1	Slide Wheel Ctrl	0.0	0.0
12	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt filter 1.subMass[1]	MassRollUpPattern		
13	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt filter 1.subPower[1]	PowerRollUpPattern		
14	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt filter 2	Slide Wheel Ctrl		
15	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.apt filter 3	Slide Wheel Ctrl		
16	peak Power Limit Scenario Online.aPS Realization.dome Installation.ins.aps electronics rack.pit ccd	Camera Ctrl	150.0	100.0

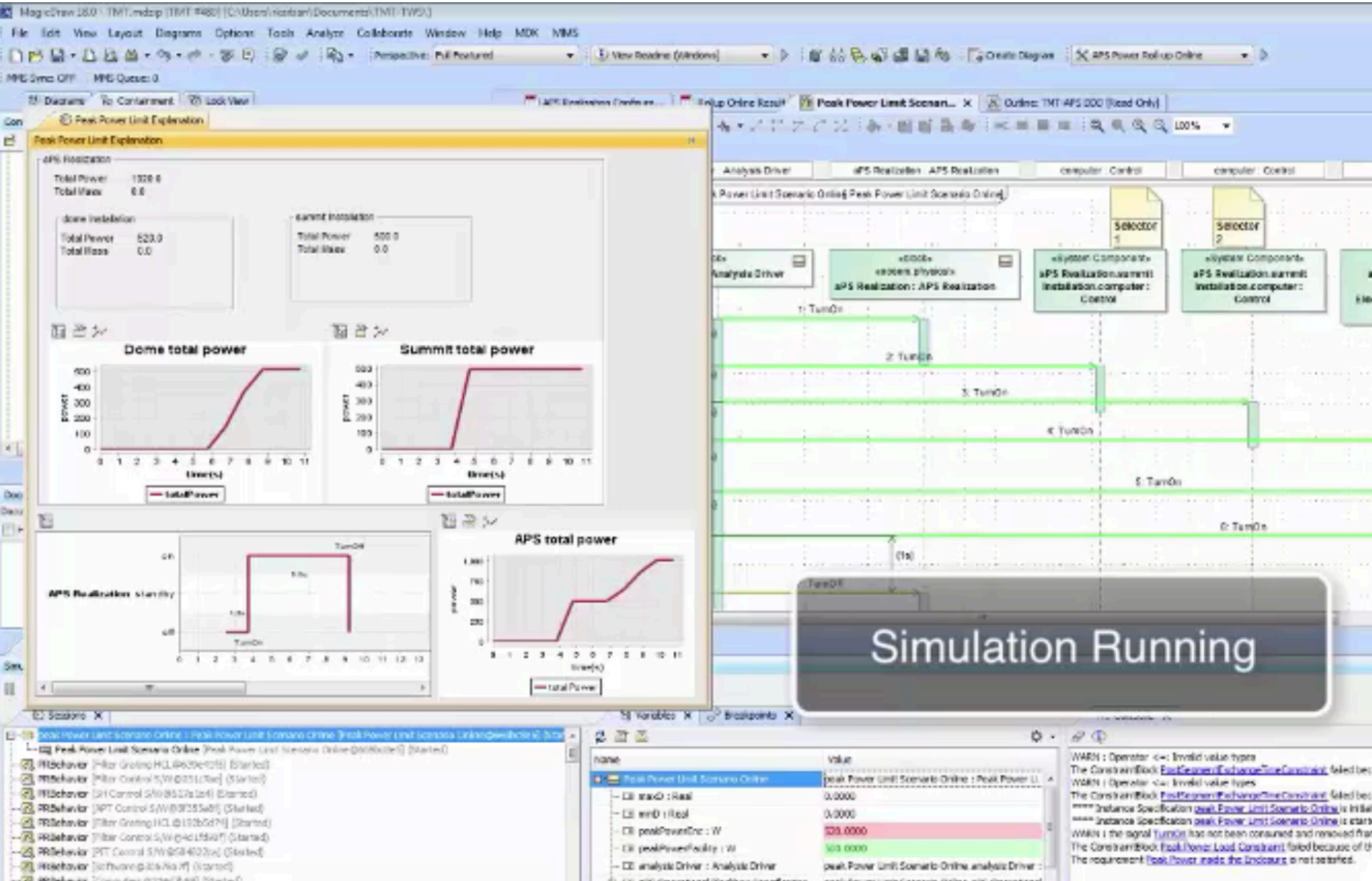
Scenario Initial Condition Pattern

178	peak Power Limit Scenario Online.aPS Realization.summit Installation.subMass[1]	MassRollUpPattern				
179	peak Power Limit Scenario Online.aPS Realization.summit Installation.subPower[1]	PowerRollUpPattern				
180	peak Power Limit Scenario Online.aPS Operational Blackbox Specification JPL.pplc	Peak Power Limit Requirem				8500.0
181	peak Power Limit Scenario Online.aPS Operational Blackbox Specification JPL.ppls	Peak Power Limit Requirem				8100.0
182	peak Power Limit Scenario Online.aPS Realization.pplc	Peak Power Limit Requirem				
183	peak Power Limit Scenario Online.aPS Realization.ppls	Peak Power Limit Requirem				
184	peak Power Limit Scenario Online.aPS Realization.summit Installation.computer1	Control	500.0	100.0	0.0	

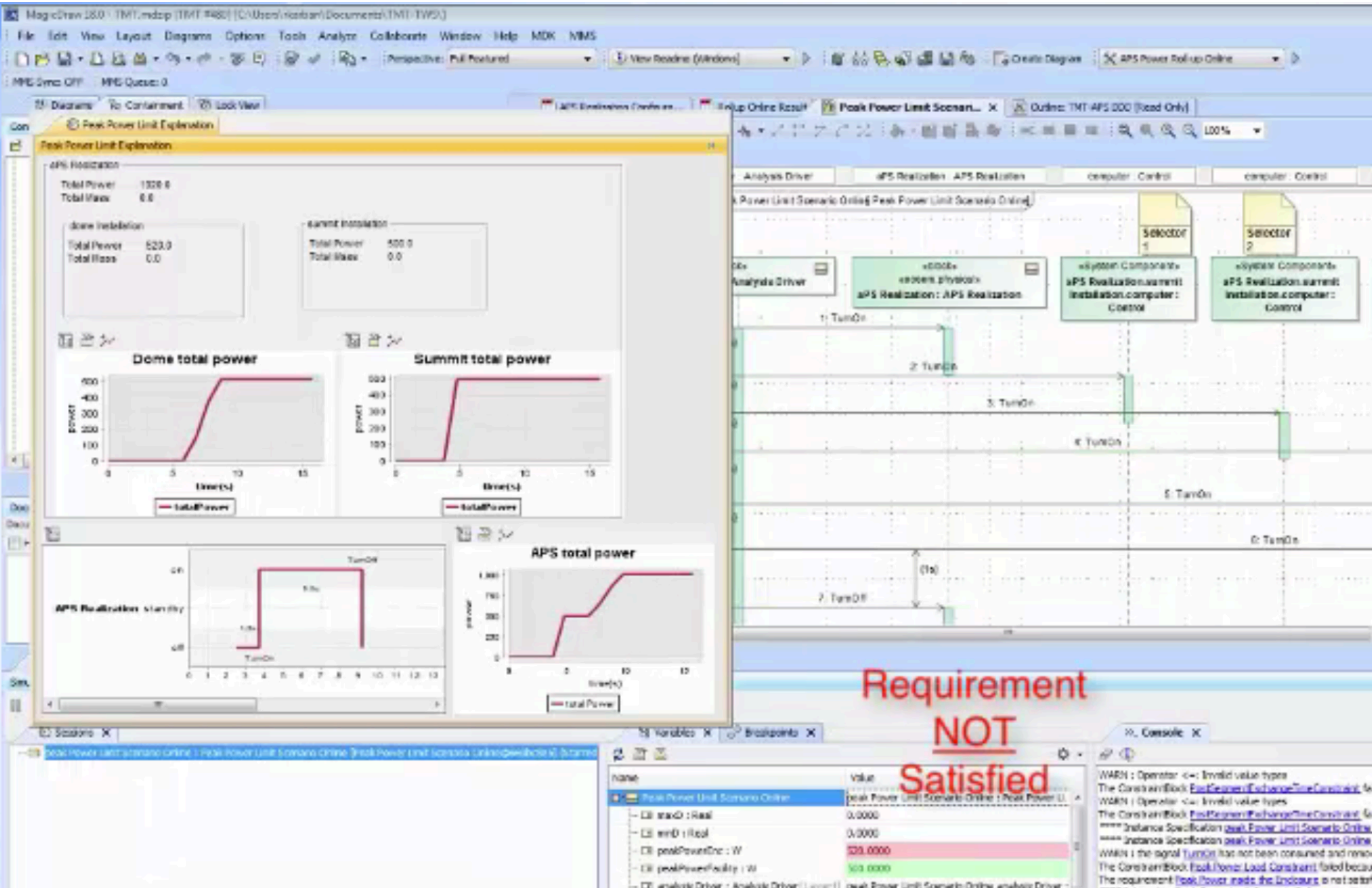
Timeline of component states



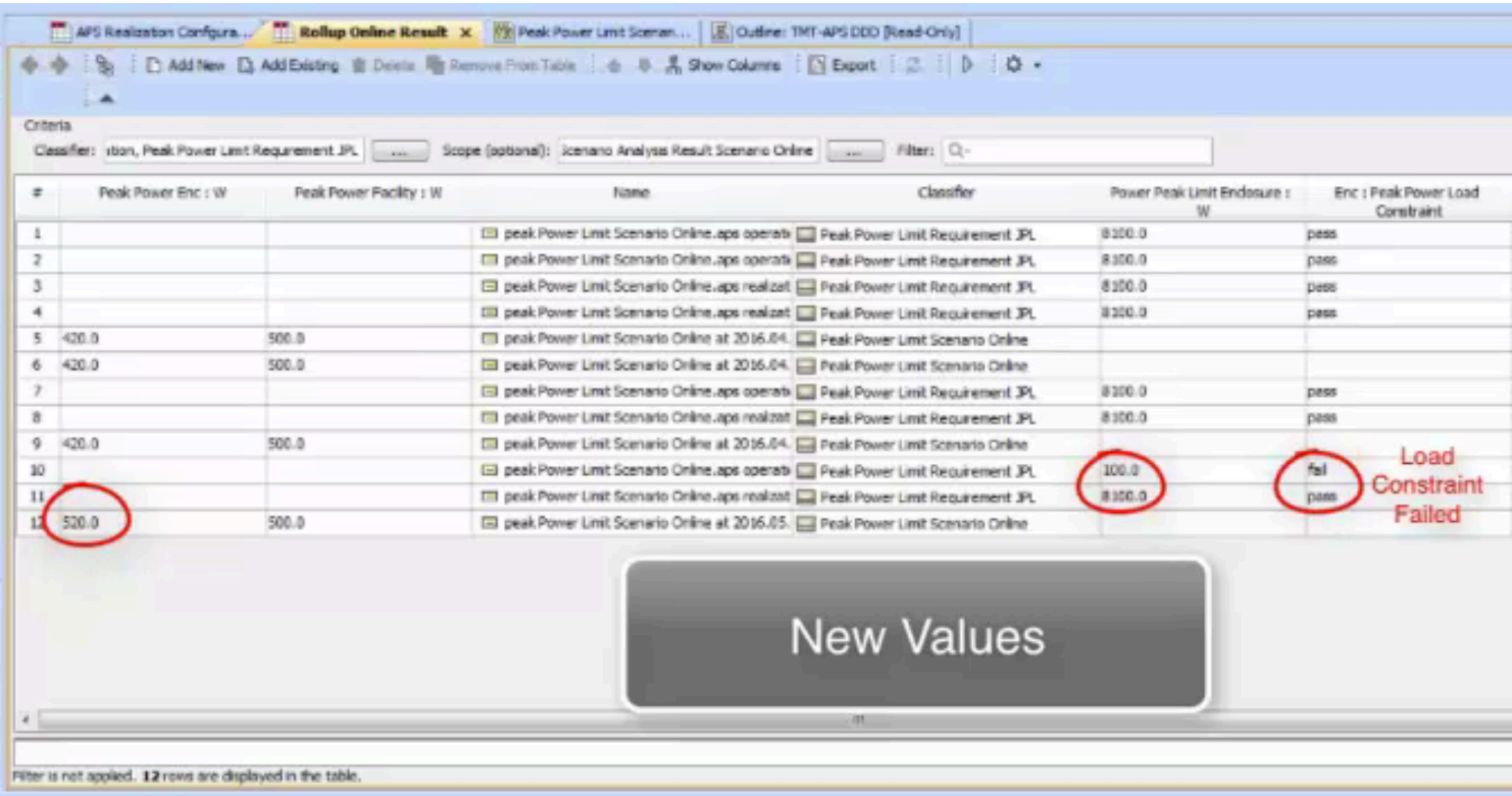
Run !


Dome total power
Summit total power
APS total power
Simulation Running

Verify

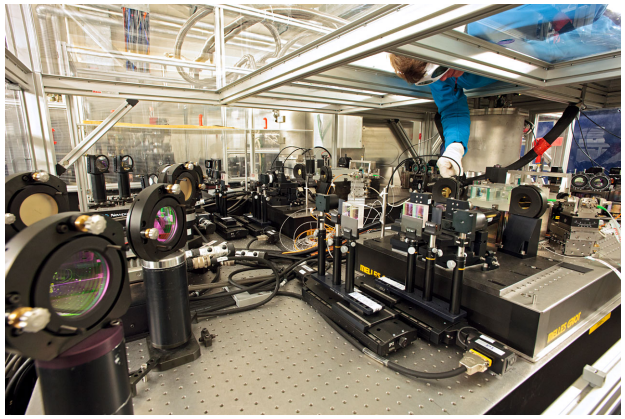

Peak Power Limit Explanation
Dome total power
Summit total power
APS total power
Requirement
NOT
Satisfied

Simulation results table



ESO Applications

PRIMA



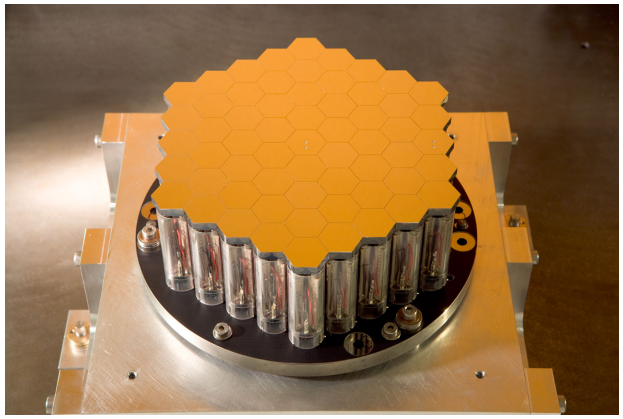
Auto-code generation

10 Components

States: 252

Transitions: 864

APE



Auto-code generation

11 Components

States:432

Transitions:1260

ELT

High level operations

17 Components

States: 34

Transitions:57

Activity elements: 340

JPL Applications

Thirty Meter Telescope APS

16 Components

States: 62

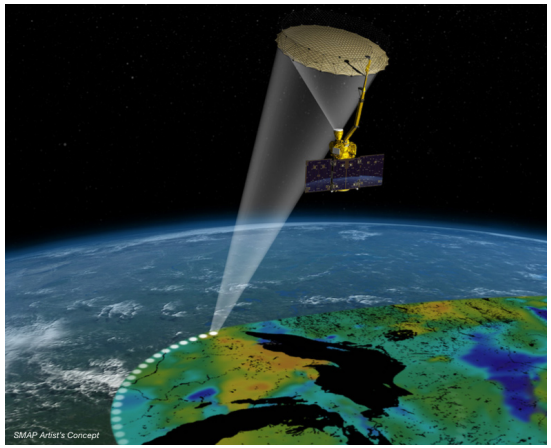
Transitions: 116

Signals: 100

Activity elements: 432

Duration constraints: 64

SMAP satellite



Modeling Fault Protection

40 Components

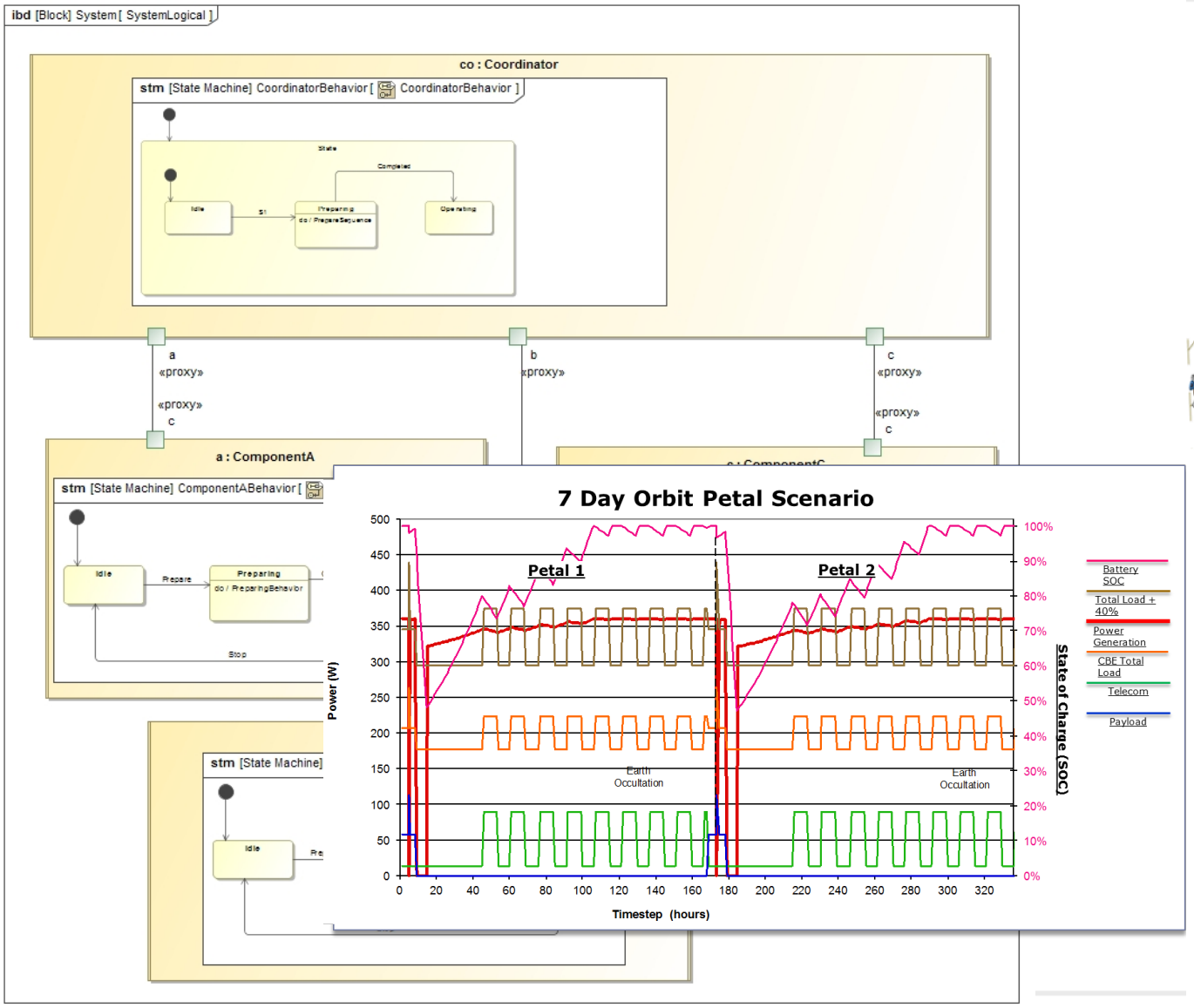
States: 2000

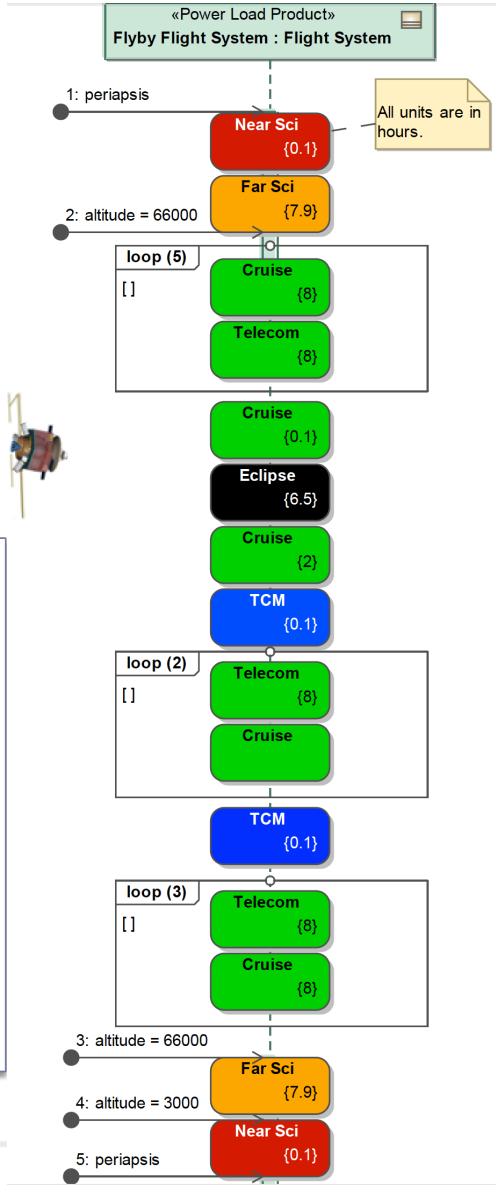
Transitions: 3000

Parallel regions: 600

Signals: 500

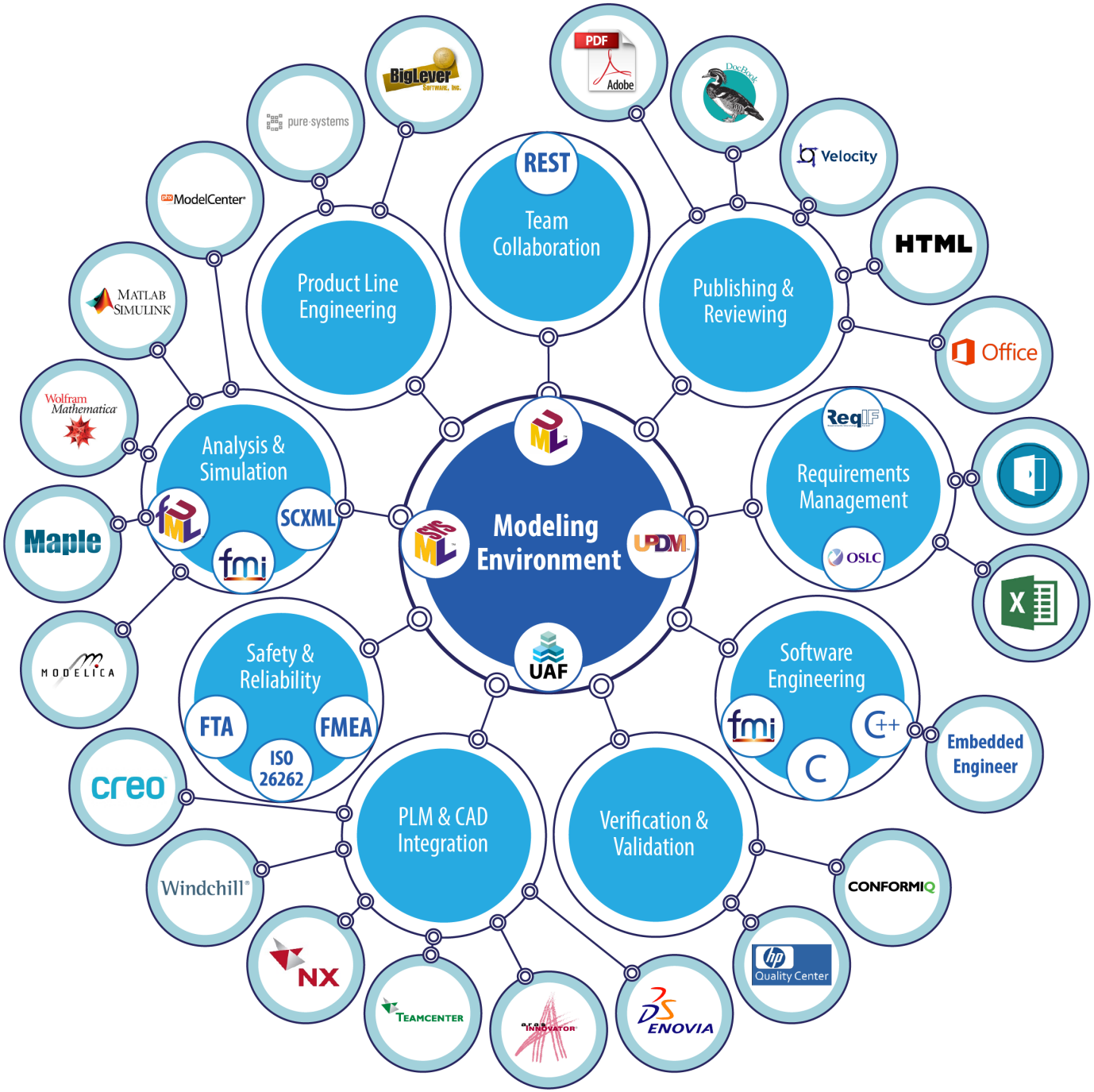
Spacecraft power profile





Summary & Outlook

- Proved to be very useful
- Big interest in other projects at JPL (Europa, MARS2020)
- Working on a Web based UI
- Need better ways to record results (plots)
- Working on analysis solver integration
- Inheriting behavior patterns


Modeling Environment
SysML
UML
UPDM
UAF
Team Collaboration
REST
Product Line Engineering
pure-systems
BigLever Software, Inc.
ModelCenter
Publishing & Reviewing
PDF
Adobe
DocBook
Velocity
HTML
Office
Requirements Management
ReqIF
OSLC
Analysis & Simulation
MATLAB Simulink
Wolfram Mathematica
Maple
MODELICA
SCXML
fmi
Safety & Reliability
FTA
FMEA
ISO 26262
Software Engineering
fmi
C++
C
Embedded Engineer
PLM & CAD Integration
creo
Windchill
NX
Teamcenter
PTC Innovator
Verification & Validation
ENOVIA
hp Quality Center
CONFORMIQ

Thank You !

See more:
https://github.com/Open-MBEE/TMT-SysML-Model

www.nomagic.com/mbse

Contact : nerijus@nomagic.com