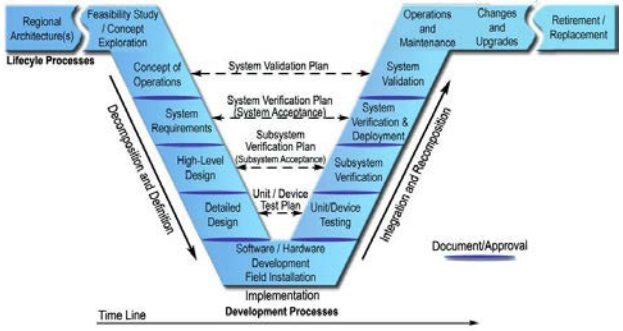
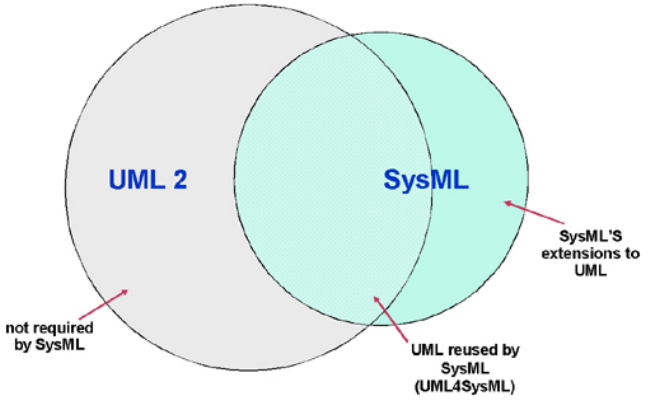
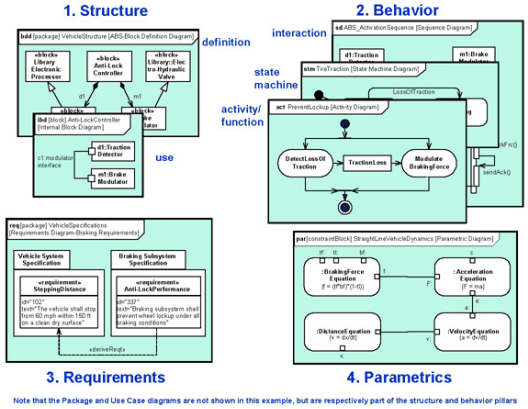
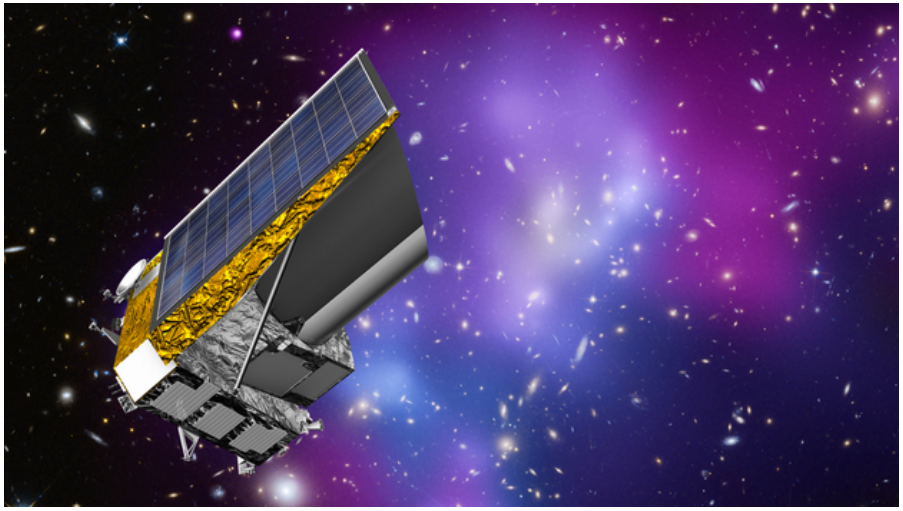






Trends in MBSE and SysML on Euclid



08/12/2016
Harold Metselaar (TEC-SWM)
Jose Lorenzo Alvarez (SCI-PUP)

MBSE Intro

- ✓ **MBSE** What everybody talks about but nobody is doing.
- ✓ **Model-based systems engineering** is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities throughout the System Development Life Cycle (SDLC).
- ✓ Supports **multi-disciplinary** aspect of SE.
- ✓ **Systems** that include hardware, software, data, personnel, procedures, and facilities.

Trends in MBSE - I

- ✓ (Elements of) MBSE **more and more applied**
  - ❑ Missions, studies, partner agencies, primes
- ✓ First steps **MBSE review**
- ✓ Document **generation from models**
  - ❑ Requirement Specification
  - ❑ DDF
  - ❑ ICDs
- ✓ **Roadmap MBSE** (ESA + primes)
- ✓ SECESA 2016
  - ❑ JAXA, cloud platform for space mission concurrent design
  - ❑ Web based systems engineering collaboration tool (Valispace)

Trends in MBSE - II

THALES

- ✓ CDF (phase 0/A/B1)
- ✓ Integrated Design Model (focus on physical concerns)

- ✓ **Trend:** extend scope of this environment

  - ❑ Functional and operational analysis
  - ❑ Functional Allocation trade-off
  - ❑ CAPELLA
  - ❑ Link functional view to physical view of the system
  - ❑ Information stored in different tools
    - Web 2.0 / Semantic Web

Trends in MBSE - III

AIRBUS (KNS Digital Engineering – Status and Challenges)

- ✓ **Trend:** E2E Product Lifecycle Management
  - ❑ Integrated digital flow
  - ❑ Multi-disciplinary integration
  - ❑ Mechanical, Electrical, Thermal, SW, …
  - ❑ SRDB (core engineering DB, E-TM-10-23/25)
  - ❑ SE view on data
  - ❑ Updating lifecycle

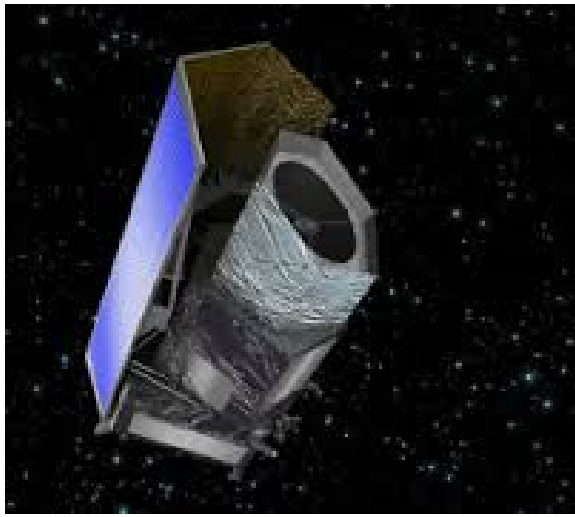
Trends in MBSE - IV

Summary

- ✓ Gain of momentum
- ✓ Full lifecycle
- ✓ Adding more disciplines
- ✓ Better integration
- ✓ Focus on data exchange
- ✓ New solutions Cloud/Web based

Euclid Intro

- ✓ **What?**
  - ❑ Cosmology beyond the Planck mission:
    - Dark Matter distribution
    - Dark Energy nature
- ✓ **How?**
  - ❑ Baryonic Acoustic Oscillations (BAO)
  - ❑ Weak Gravitational Lensing
- ✓ **The Mission:**
  - ❑ Large Sky Survey: 15,000 deg2
  - ❑ Visible imaging (VIS)/Near-Infrared (NISP)
  - ❑ L2 Orbit
  - ❑ 5-6 year



Business case MBSE on Euclid with SysML?

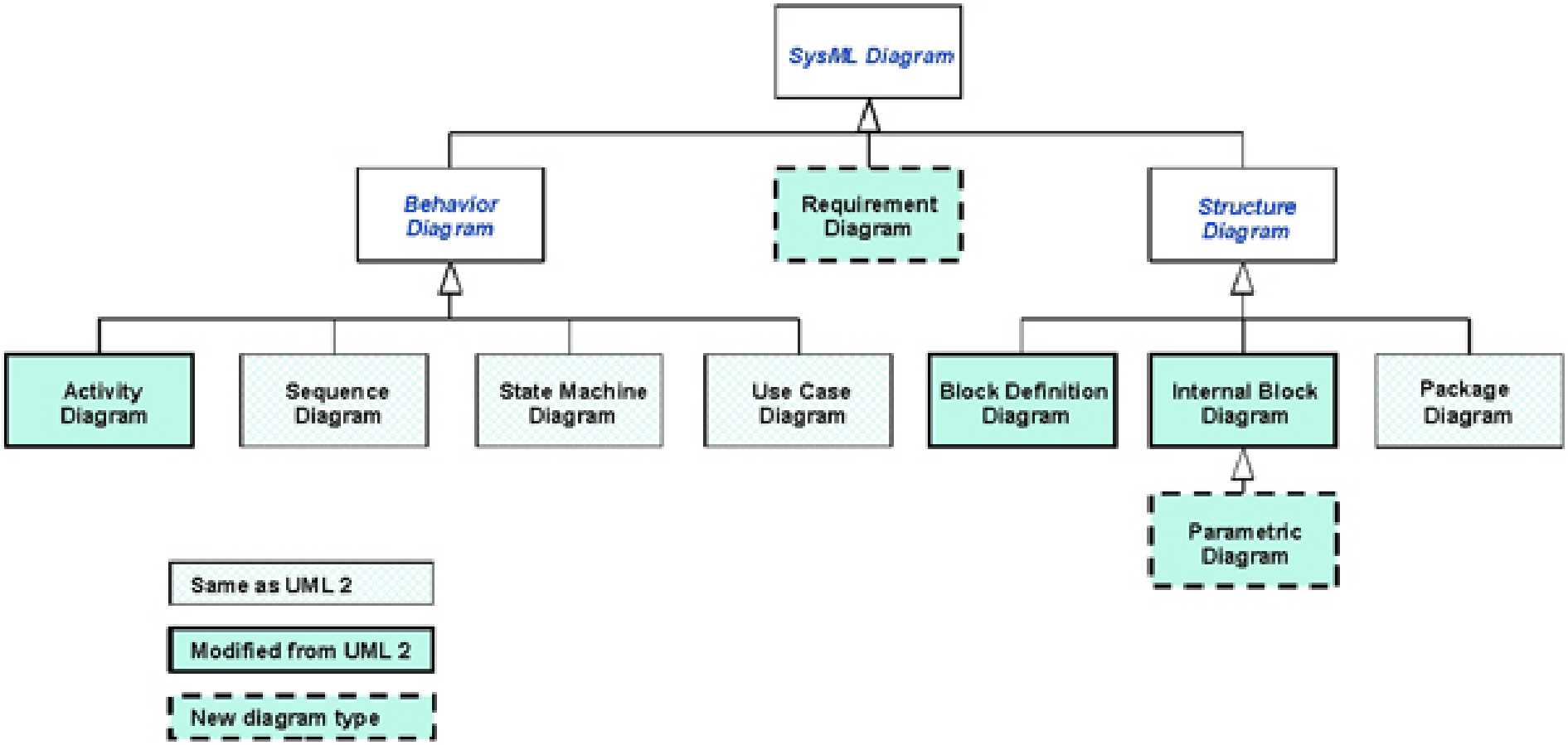
Model Based System Engineering (MBSE) approach for Euclid:

- ✓ **Complex** system and requirement interaction
- ✓ **Many actors** in the system: ESA, Industry, NASA, large/distributed Euclid Consortium
- ✓ Need to **manage information exchange** and control efficiently and coherently

⇨ Decided to implement a Model Based System Engineering (MBSE) collaborative approach with Euclid Consortium.

⇨ Decided to use **SysML**

SysML diagram types

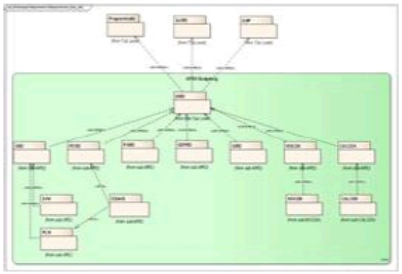


MBSE on Euclid with SysML

«allocate» Model Components

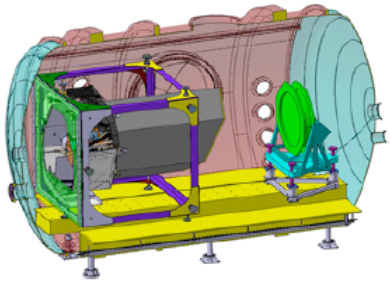
Actor

Actor package describe the different structure and key people contributing to the Euclid Mission.
Status: preliminary



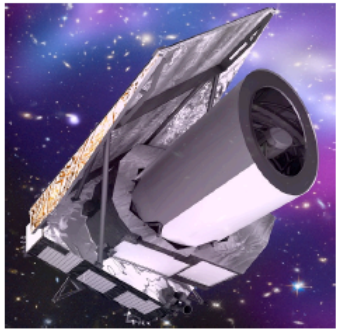
Requirement

Requirement package contains the requirement specification flow down from top level Science Requirements to implementation.
Status: Advanced



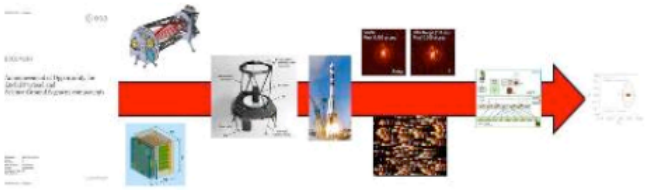
Verification

Verification package contains the test cases and verification approach description that allow verifying that current mission implementation meet the expected needs.
Status: empty



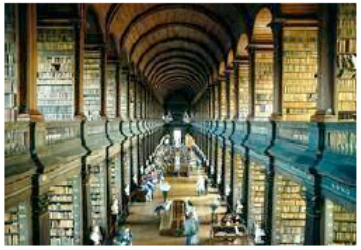
Architecture

Architecture package describe the design architecture of the mission, including the Mission product tree, the Mission environment and the Mission interface.
Status: advanced



Lifecycle

Lifecycle package contains description of the different steps in the life of the Eudid Mission from selection to scientifical analysis of the processed data.
Status: empty

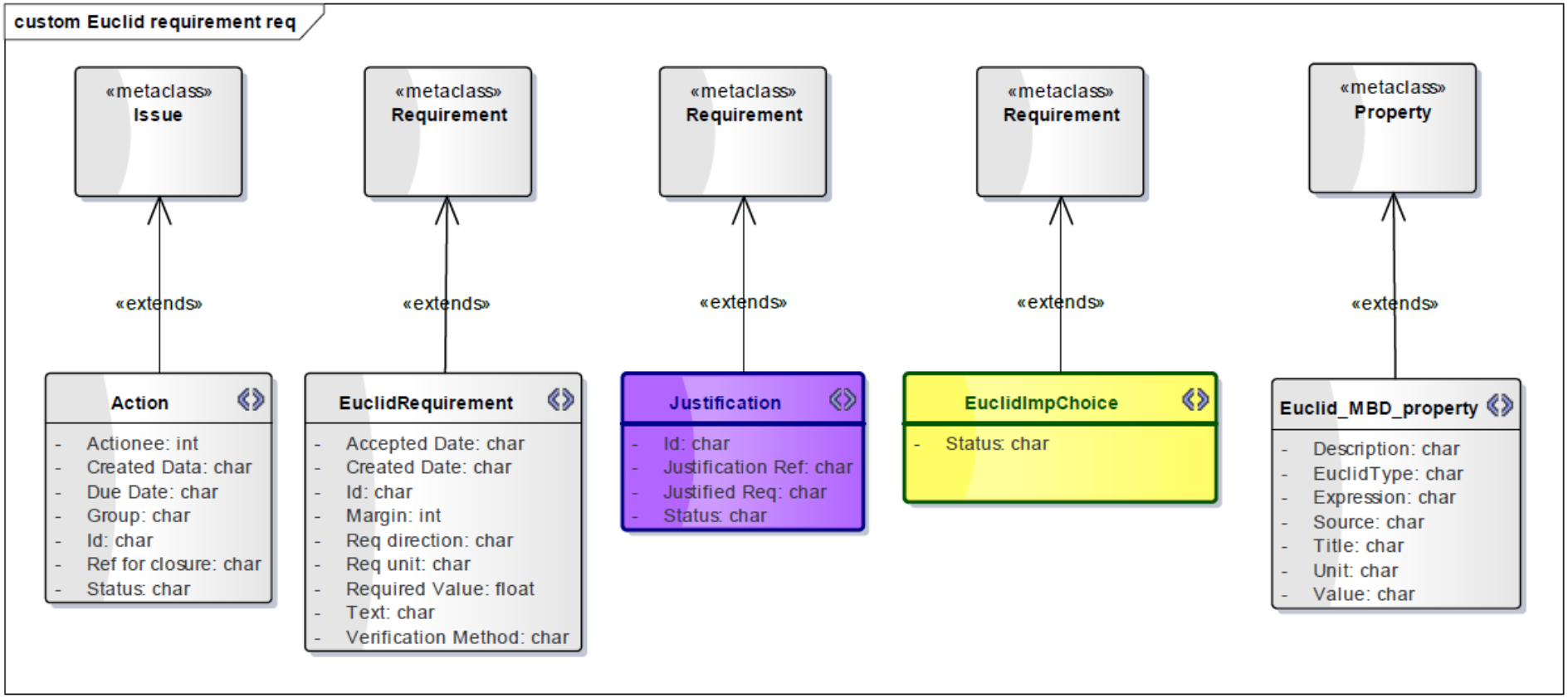


Model Library

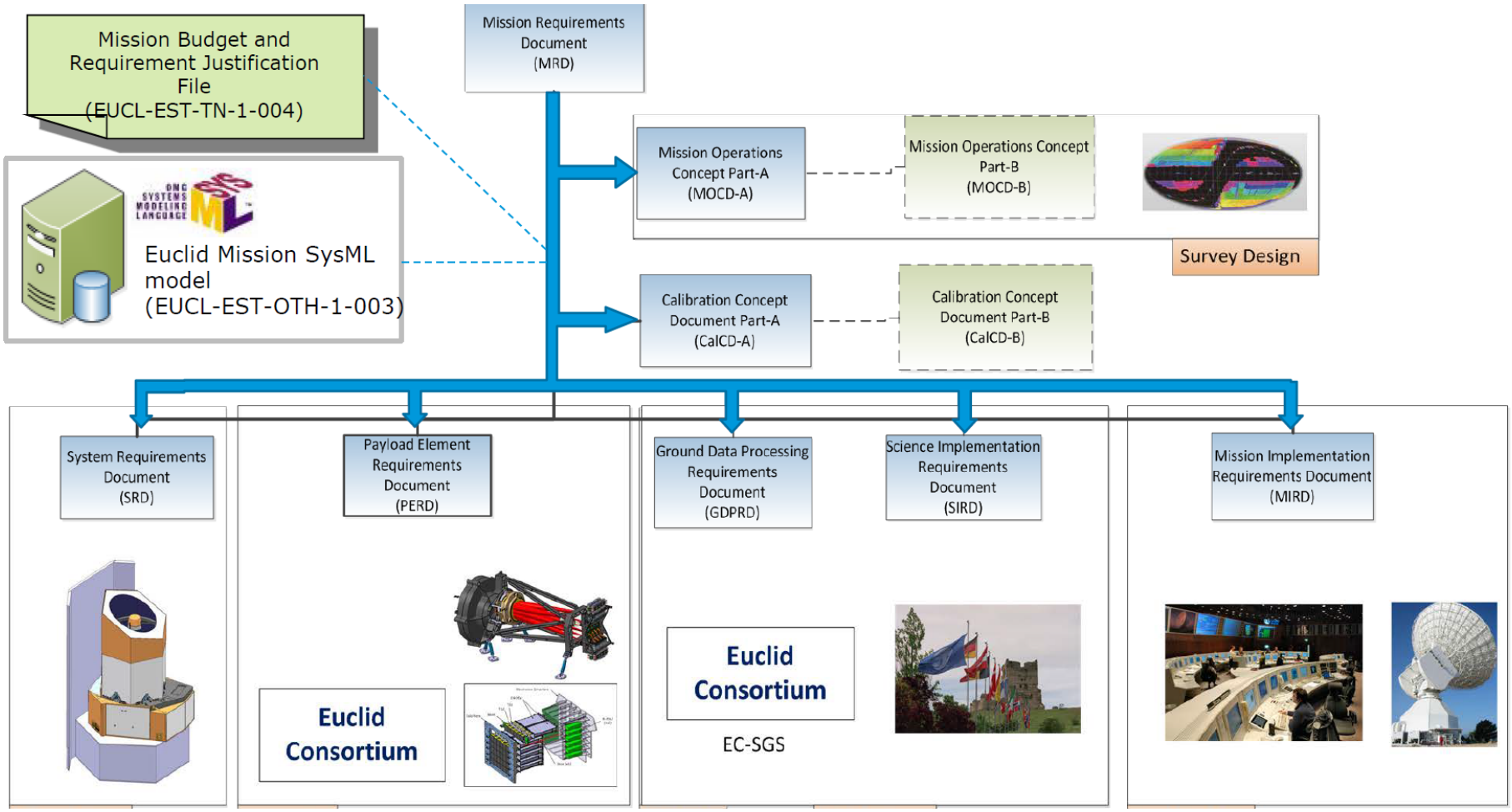
Model Library contains reference definition, profiles and stereotypes used in the model.
Status: advanced

Requirements - I

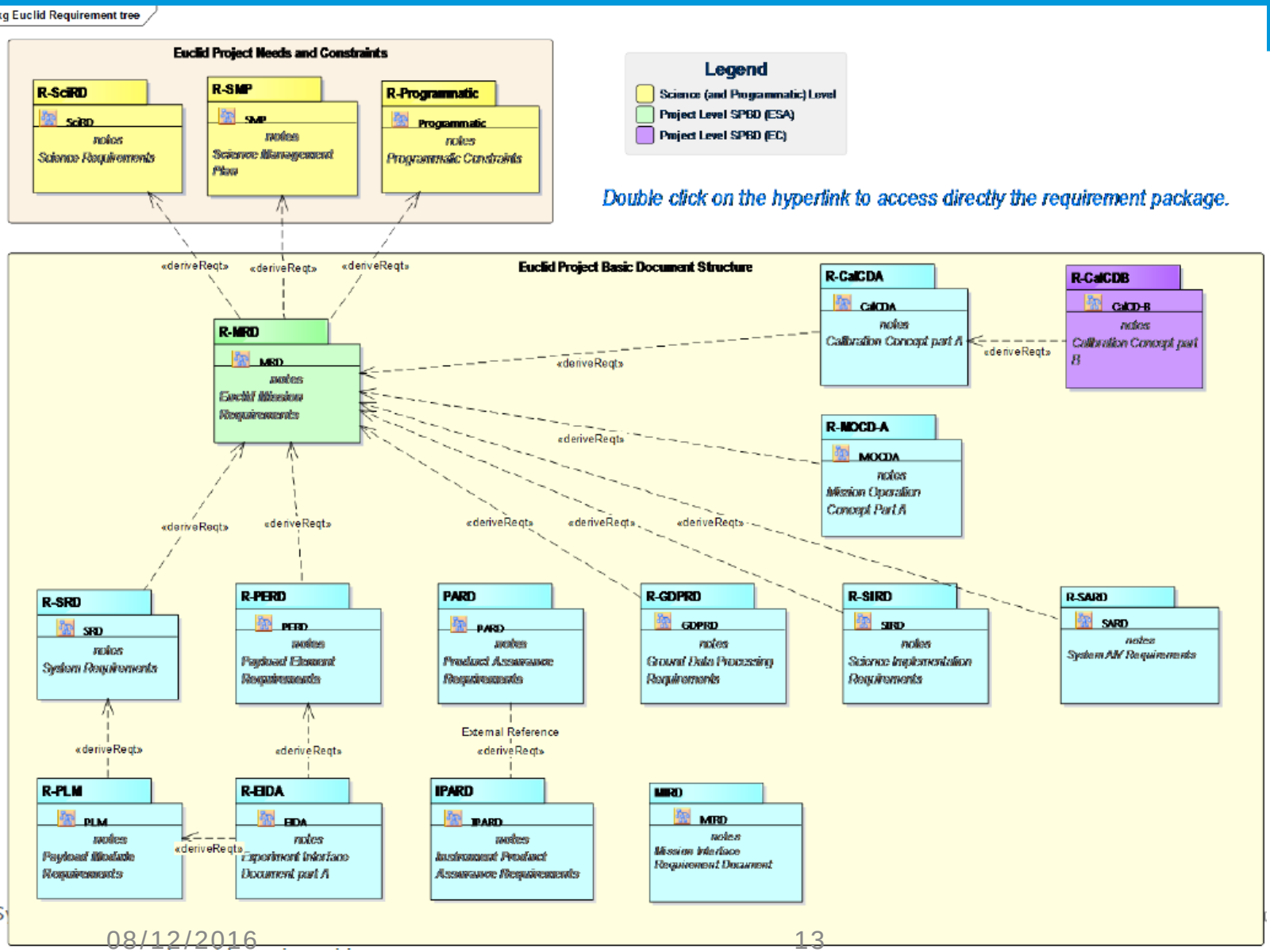
Euclid specific SysML **extension**



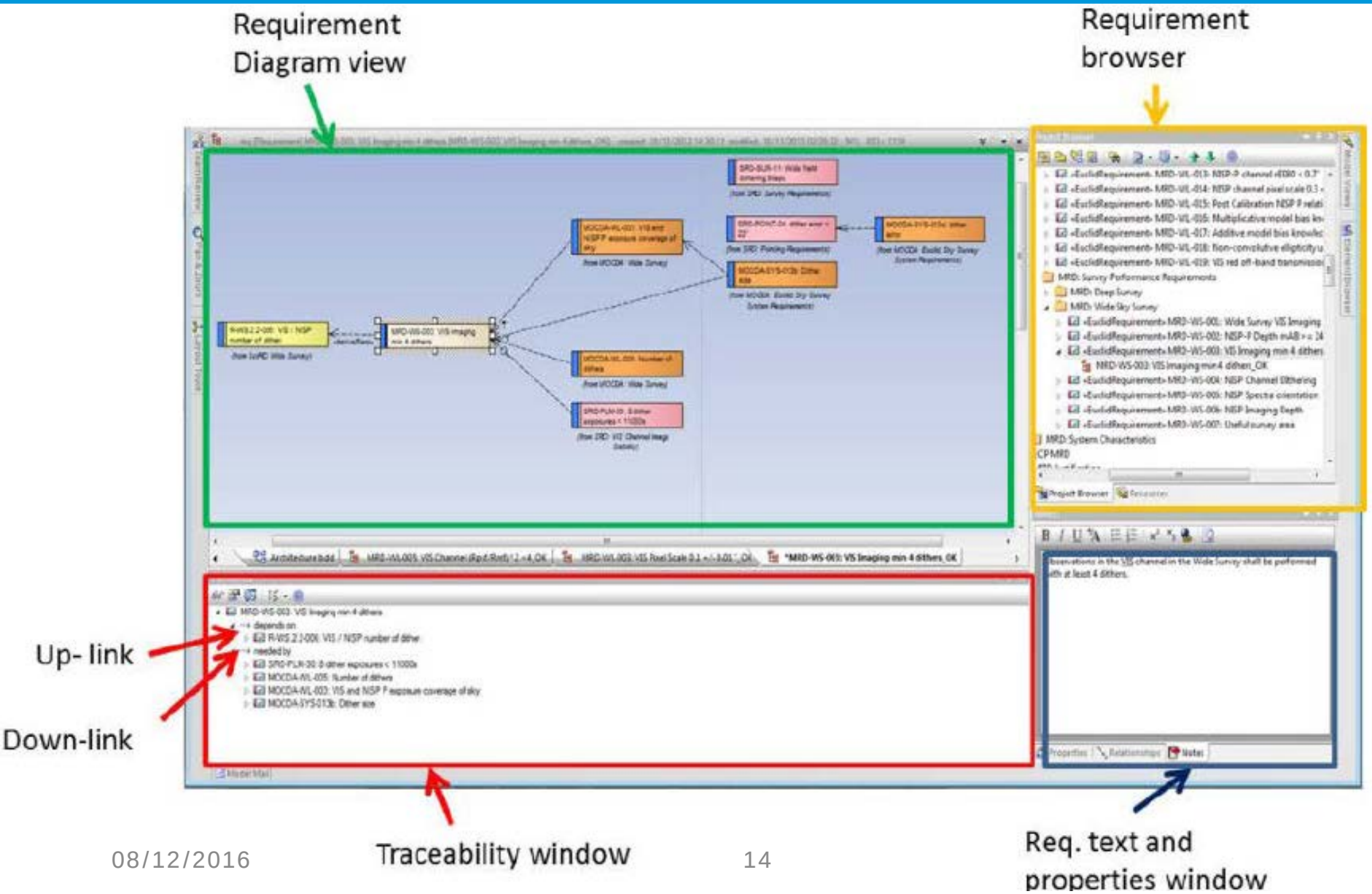
Requirements - II



Requirements - III



Requirements - IV

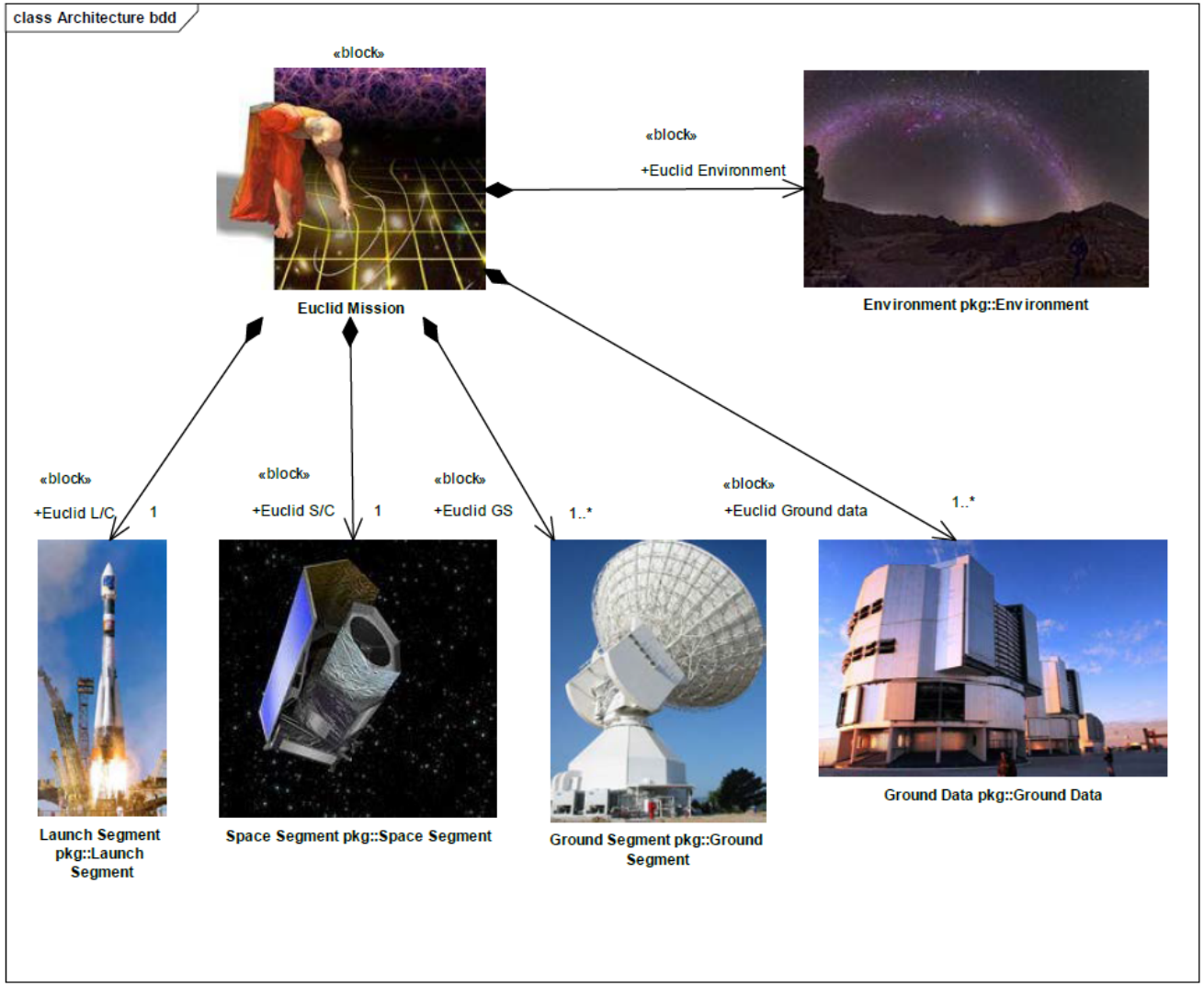


Requirements - V

Benefits:

- ✓ **Part of model**!
- ✓ Full justification logic track
- ✓ **Traceability and control**
- ✓ Adequate change **impact assessment**
- ✓ Requirement document **generation**

Architecture modelling - I



High level decomp.

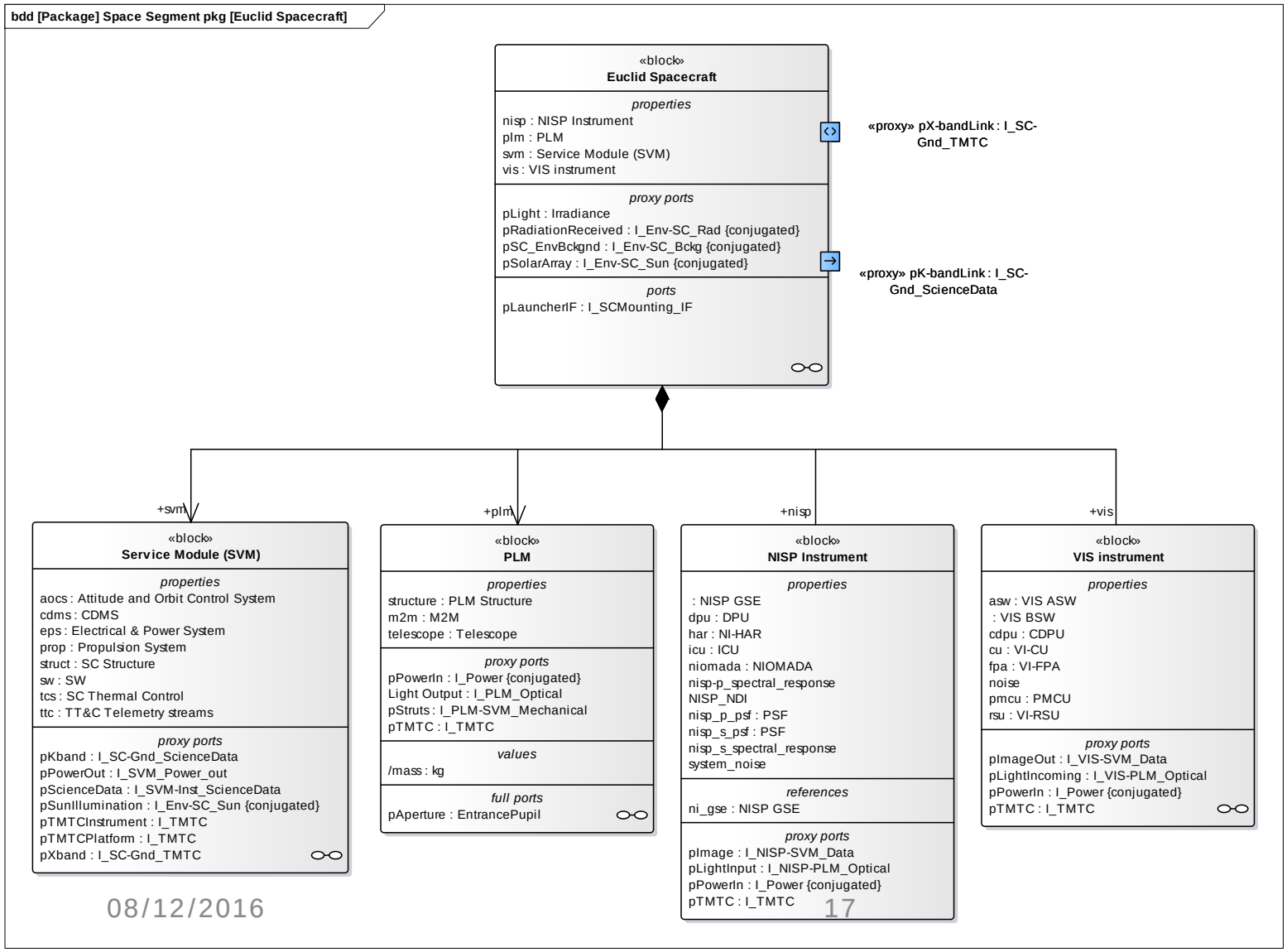
Modelled:

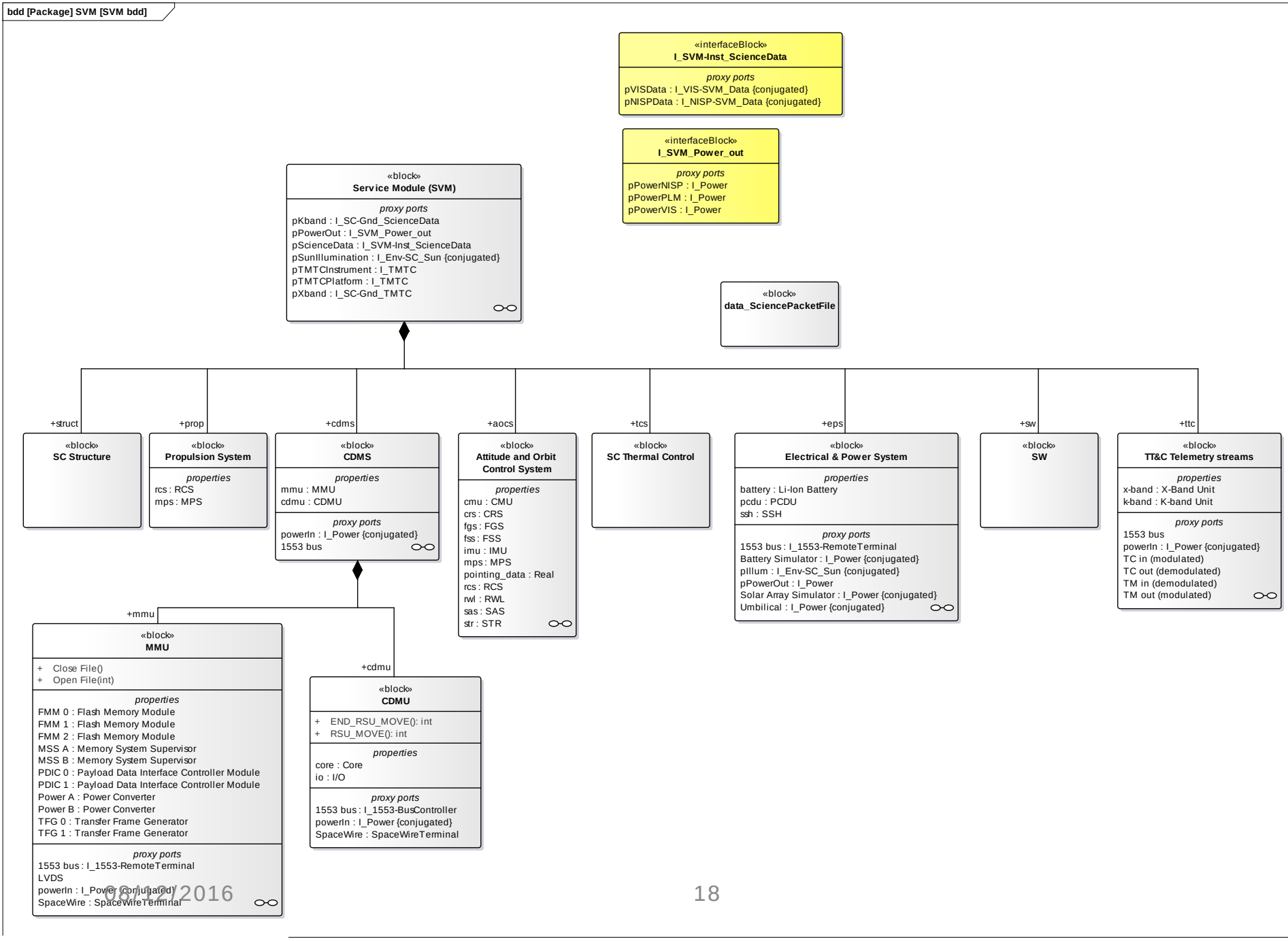
- ✓ Environment
- ✓ Launcher
- ✓ Space Segment
- ✓ Ground Segment

Mostly:

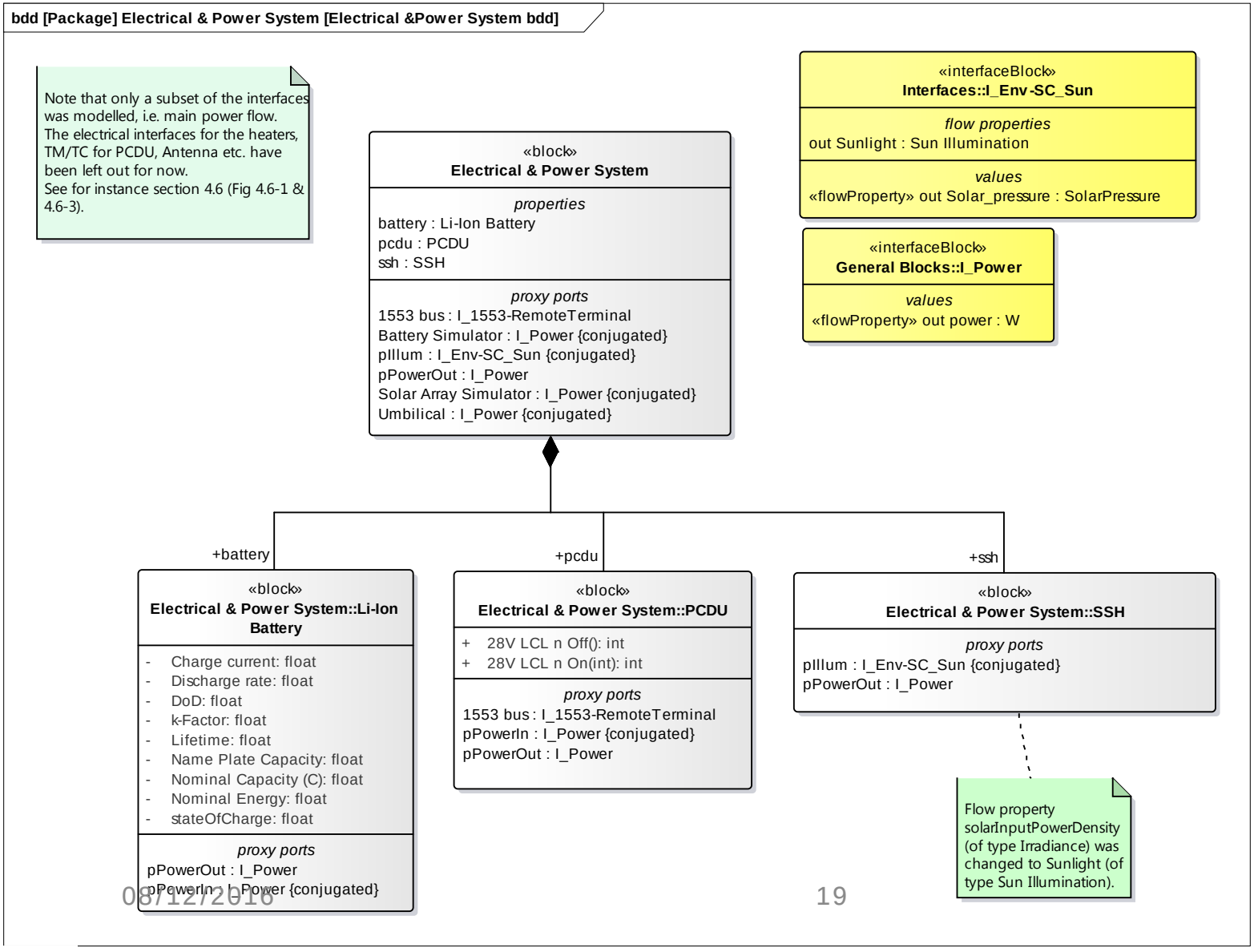
- ✓ Hierarchy
- ✓ Interconnection
- ✓ Characterization

Architecture modelling - II

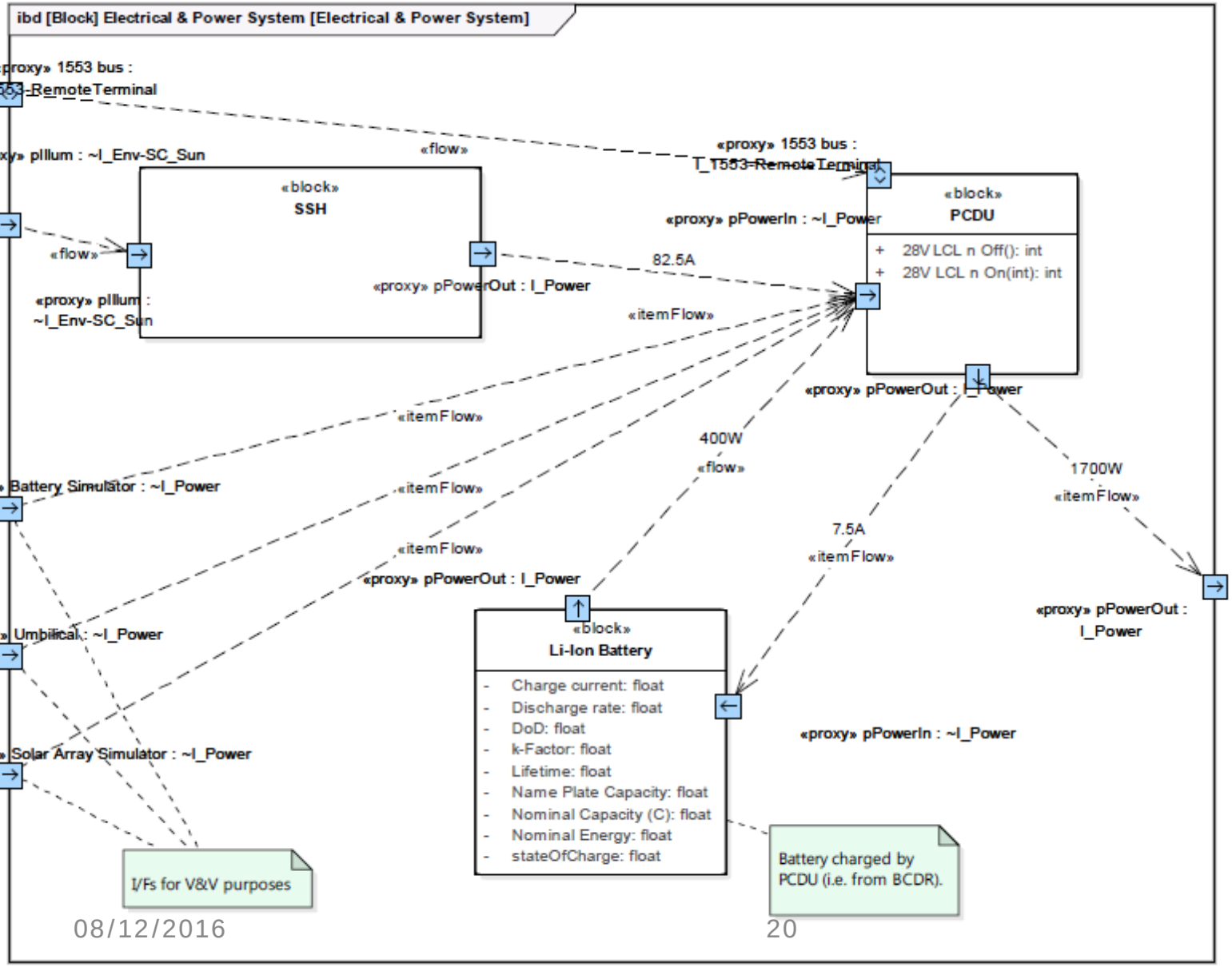



bdd [Package] SVM [SVM bdd]
«interfaceBlock»
I_SVM-Inst_ScienceData
proxy ports
pVISData : I_VIS-SVM_Data {conjugated}
pNISPData : I_NISP-SVM_Data {conjugated}
«interfaceBlock»
I_SVM_Power_out
proxy ports
pPowerNISP : I_Power
pPowerPLM : I_Power
pPowerVIS : I_Power
«block»
Service Module (SVM)
proxy ports
pKband : I_SC-Gnd_ScienceData
pPowerOut : I_SVM_Power_out
pScienceData : I_SVM-Inst_ScienceData
pSunIllumination : I_Env-SC_Sun {conjugated}
pTMTCInstrument : I_TMTC
pTMTCPlatform : I_TMTC
pXband : I_SC-Gnd_TMTC
«block»
data_SciencePacketFile
+struct
+prop
+cdms
+aocs
+tcs
+eps
+sw
+ttc
«block»
SC Structure
«block»
Propulsion System
properties
rcs : RCS
mps : MPS
«block»
CDMS
properties
mmu : MMU
cdmu : CDMU
proxy ports
powerIn : I_Power {conjugated}
1553 bus
«block»
Attitude and Orbit Control System
properties
cmu : CMU
crs : CRS
fgs : FGS
fss : FSS
imu : IMU
mps : MPS
pointing_data : Real
rcs : RCS
rwl : RWL
sas : SAS
str : STR
«block»
SC Thermal Control
«block»
Electrical & Power System
properties
battery : Li-Ion Battery
pcdu : PCDU
ssh : SSH
proxy ports
1553 bus : I_1553-RemoteTerminal
Battery Simulator : I_Power {conjugated}
pIllum : I_Env-SC_Sun {conjugated}
pPowerOut : I_Power
Solar Array Simulator : I_Power {conjugated}
Umbilical : I_Power {conjugated}
«block»
SW
«block»
TT&C Telemetry streams
properties
x-band : X-Band Unit
k-band : K-band Unit
proxy ports
1553 bus
powerIn : I_Power {conjugated}
TC in (modulated)
TC out (demodulated)
TM in (demodulated)
TM out (modulated)
+mmu
+cdmu
«block»
MMU
+ Close File()
+ Open File(int)
properties
FMM 0 : Flash Memory Module
FMM 1 : Flash Memory Module
FMM 2 : Flash Memory Module
MSS A : Memory System Supervisor
MSS B : Memory System Supervisor
PDIC 0 : Payload Data Interface Controller Module
PDIC 1 : Payload Data Interface Controller Module
Power A : Power Converter
Power B : Power Converter
TFG 0 : Transfer Frame Generator
TFG 1 : Transfer Frame Generator
proxy ports
1553 bus : I_1553-RemoteTerminal
LVDS
powerIn : I_Power {conjugated}
SpaceWire : SpaceWireTerminal
«block»
CDMU
+ END_RSU_MOVE(): int
+ RSU_MOVE(): int
properties
core : Core
io : I/O
proxy ports
1553 bus : I_1553-BusController
powerIn : I_Power {conjugated}
SpaceWire : SpaceWireTerminal

Architecture modelling - IV



Architecture modelling - V



Architecture modelling - VI

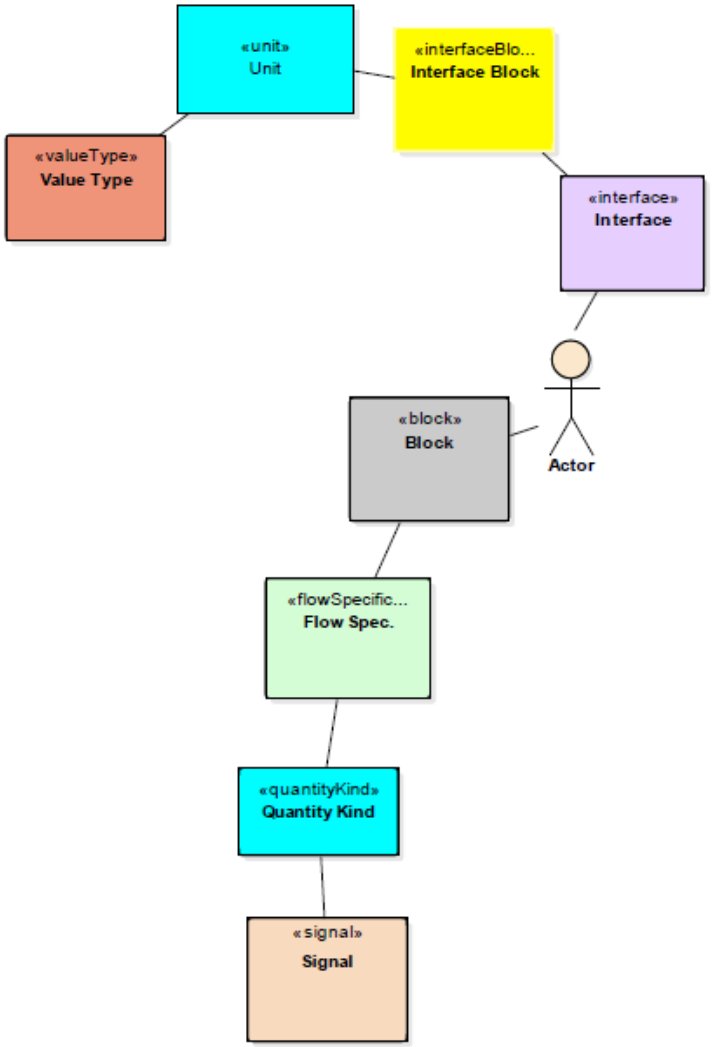
Benefits:

- ✓ Facilitate **communication** among various stakeholders across SDLC
- ✓ Management of **complexity**
- ✓ **Explore** multiple solutions or ideas concurrently with **minimal risk**
- ✓ **Detect errors** and omissions **early** in SDLC

But:

- ✓ Is SysML too **complex?**
- ✓ Is SysML too **generic/flexible?**
- ✓ Is SysML **missing** something?
- ✓ No best practices yet ☹

Questions


«unit»
Unit
«interfaceBlo...
Interface Block
«valueType»
Value Type
«interface»
Interface
«block»
Block
Actor
«flowSpecific...
Flow Spec.
«quantityKind»
Quantity Kind
«signal»
Signal