28th Annual INCOSE international symposium

Washington, DC, USA
July 7 - 12, 2018

An MBSE Tool to Support Architecture Design for Spacecraft Electrical Power System

Zhigang Liu[1,*], Mengfei Yang[1,*], Qi Chen[1],Qing Du[1],Ning Xia[1],Jinzhi Lu[1,*]
1. Beijing Institute of Spacecraft System Engineering
2. KTH Royal Institute of Technology

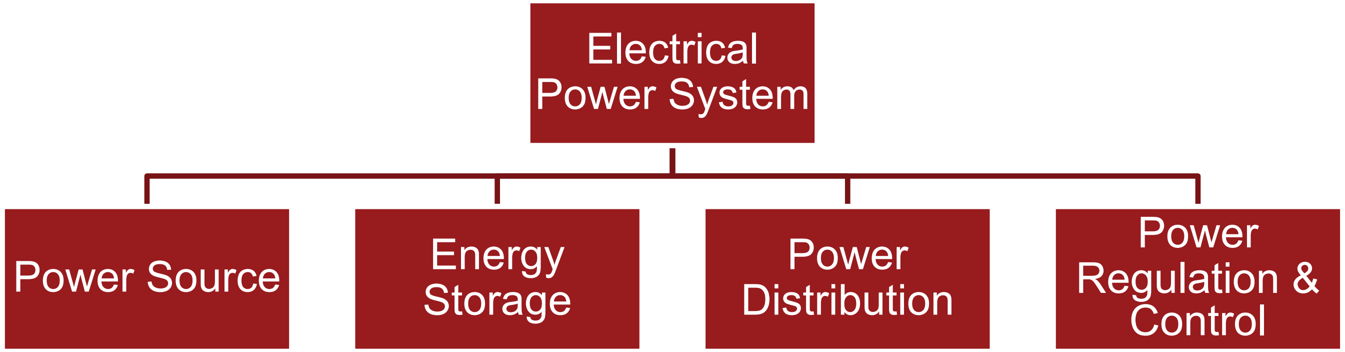
www.incose.org/symp2018

CONTENTS

1.Introduction

About Spacecraft Electrical power system(EPS)

- Electrical power systems (EPS) is used to generate, store, supply, control and distribute electrical powers for spacecraft.

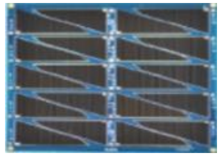


Solar Photovoltaic
Battery
Fuel Cell
Nuclear

Batteries
Flywheels

Bus control
Bus protect
Switch control
…

Energy Storage Control:
Shunt regulator
Main error amplifier

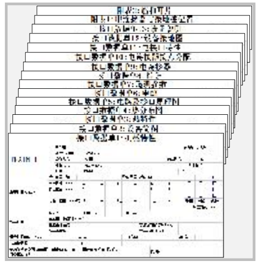



EPS has systemic and global characteristics in spacecraft. Studying design method and tool of EPS is of great significance for improving the design capability of spacecraft system.

Scope

- In this paper, the ***system architecture*** refers to the physical system structures of EPS, but our tool can formalize other views, such as ***requirement*** and ***function***.
- We adopted our tool to support system architecture modeling using current DSML, but based on the proposed meta-meta models, other languages can also be supported.

Challenge



《Interface Data Sheet》
《Equipment Design Report》
《Acceptance Report》

Design domain

Preliminary Design

√ Select and size power source/energy storage
√ Identify power regulation and control
√ power distribution design

Detail Design

√ Interface document design
√ Physical pin design
√ Harness design

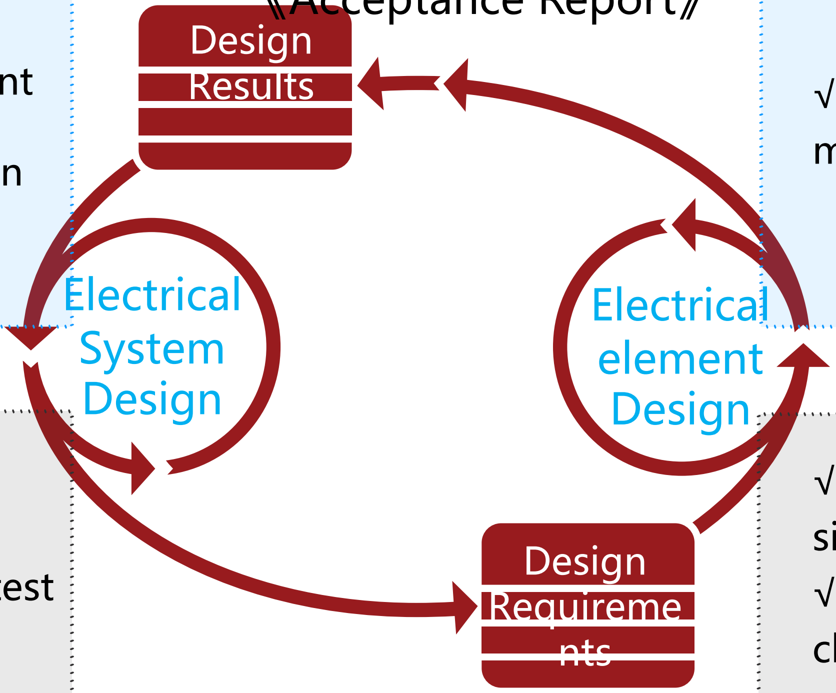
Design domain

Schematic Design

√ schematic modeling

PCB Design

√ 2d layout and wiring
√ 3d assemble



√ Qualitative analysis
√ Desin review
√ Experiment vadlication

Preliminary Validation

√ Semi-physical test
√ Physical prototype test

Detail Validation

V&V domain

√ schematic simulation
√ digital checking

Schematic Validation

√ SI PI simulation
√ thermal analysis
√ assemble simulation
√ EMC simulation

PCB Validateion

V&V domain

《Technical Requirements》
《System Design Report》

Challenge

Compared with element-level designs, system-level design lacks effective methods and tools to overcome several challenges:

- Promote efficiency of document-based design
- Integrate verification of proposed systems
- Enable consistency and coordination of the loop from design to verification
- Improve communications between the stakeholders in system and the element levels.

Objective

The goal in this paper is to adopt Model-based Systems Engineering to formalize, design and analyze EPS architectures and enhance management of the relevant technical resources, e.g., models and data.

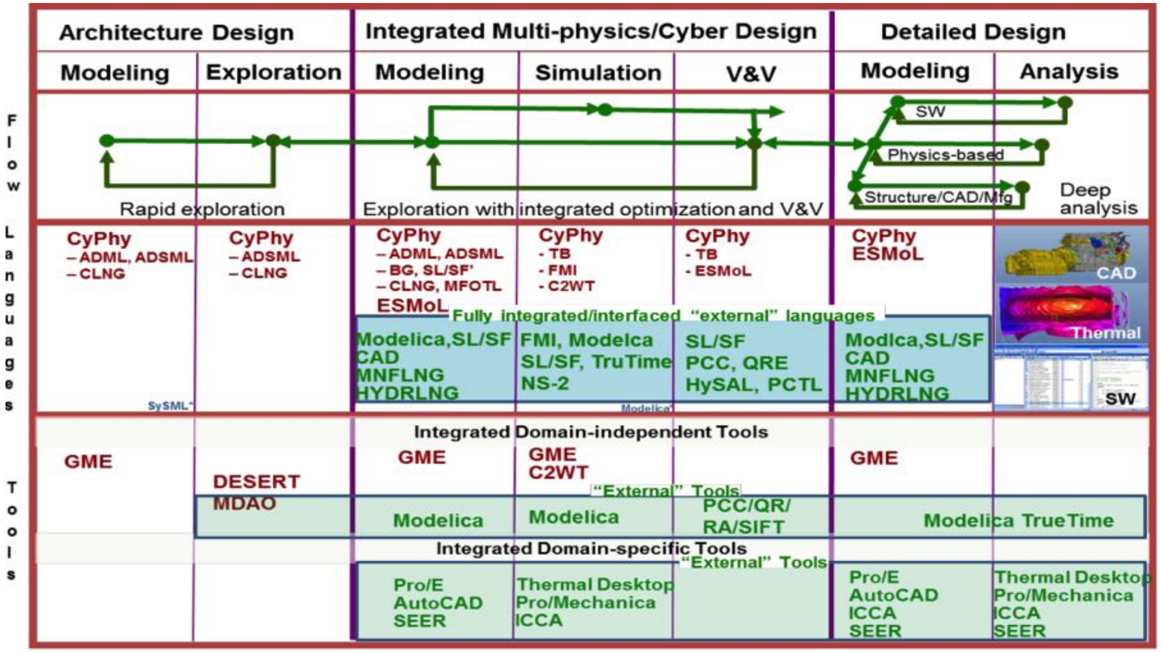
- Objective 1-support for designing EPS architectures in a systematic and structural way
- Objective 2-support for communicating the related information of EPS architectures in a concise and graphical way
- Objective 3-support for implementing V&V of related architecture concepts automatically

2.Related Work

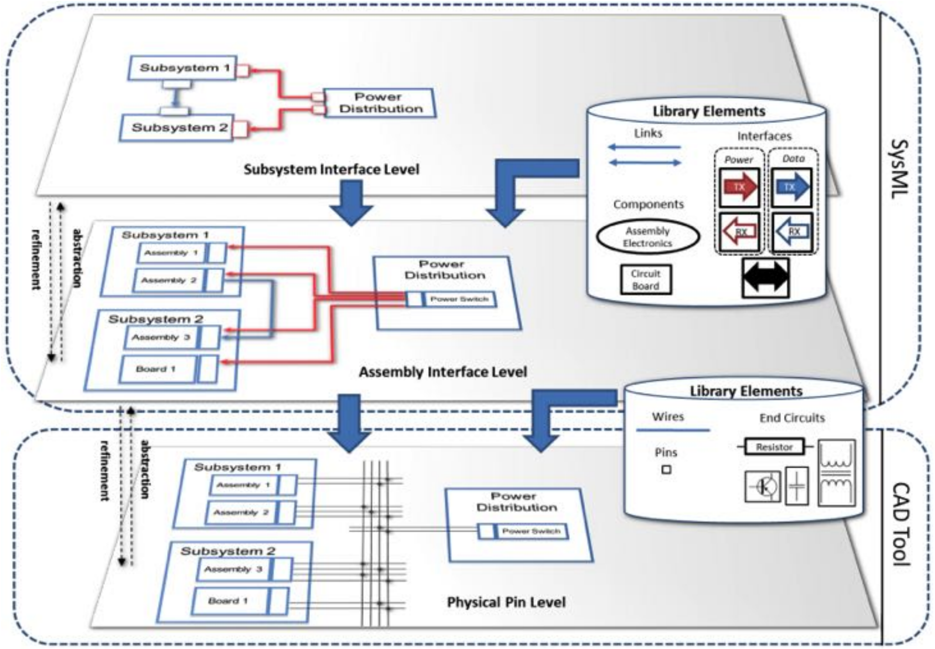
Architectual Modeling

- **DARPA :** In AVM META program, researchers developed the META language, called CyPhy as a model integration language. Cyphy included a sublanguage named ADML (Architecture Design Modeling Language) representing hierarchical component architectures and typed interfaces.

- JPL : The main work included the extension of SysML with domain concepts to design DML, a DSM language formalizing and designing specifications and architectures of the electrical flight system.



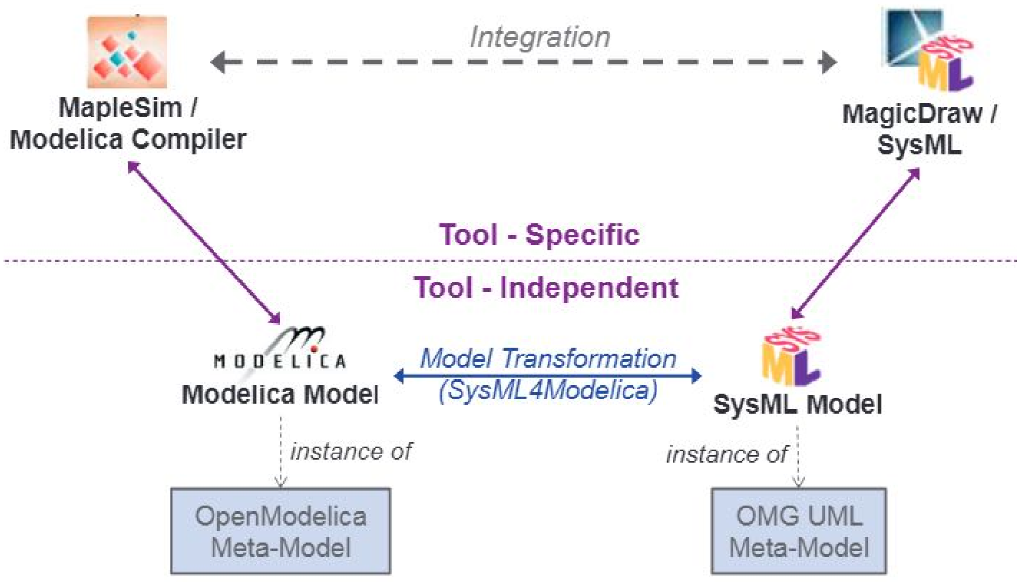
Source:META-X DESIGN FLOW TOOLS Final Report



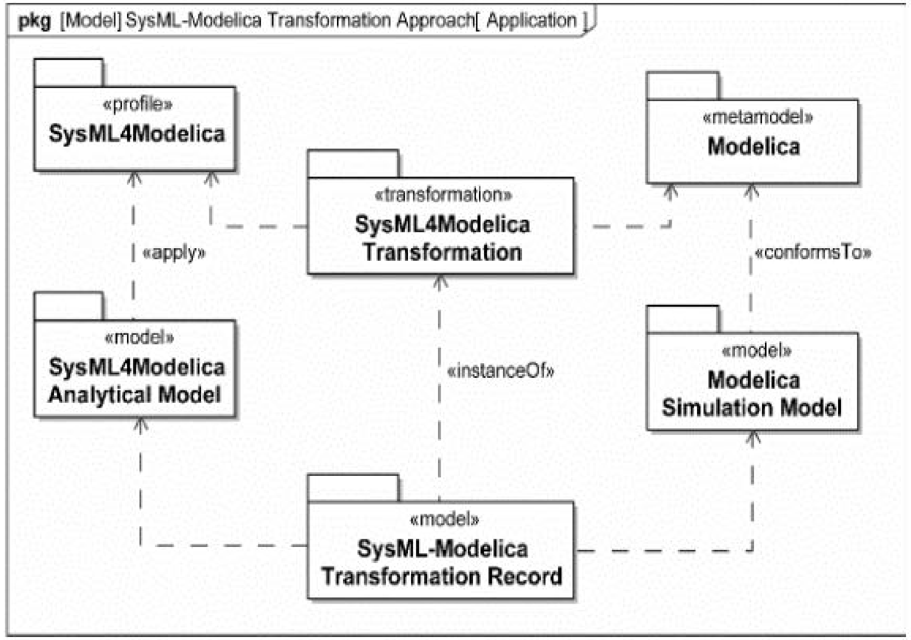
Source:Specification and Design of Electrical Flight System Architectures with SysML

Integrating Descriptive Model with Analytical model

- Integrated DML implementations with V&V using Modelica that a model transformer was developed between a Modelica tool, MapleSim and a SysML modeling tool, MagicDraw (Schamai et al. 2012).
- OMG™ extended SysML and allowed for using Modelica language concepts directly in SysML by SysML4Modelica profile.



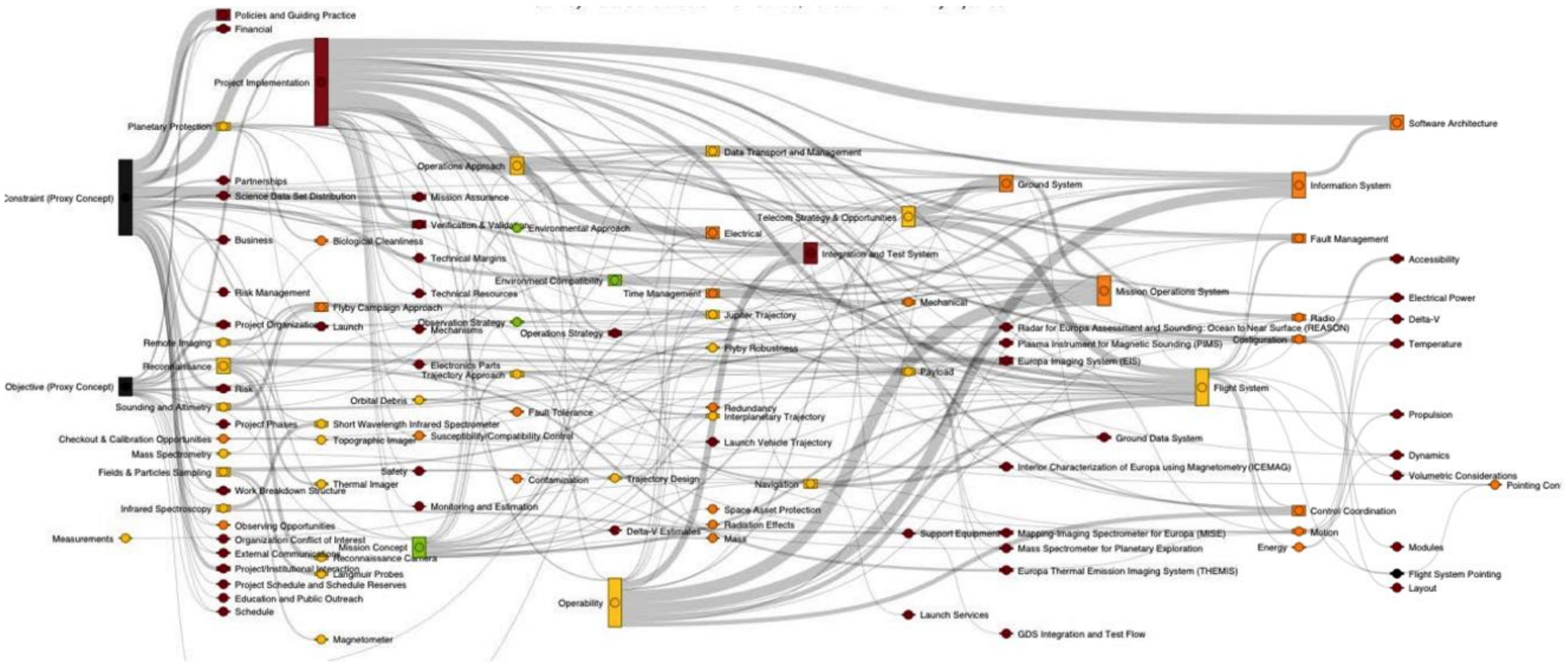
Source:Integrating Analytical Models with Descriptive System Models: Implementation of the OMG SyML Standard for the Tool-specific Case of MapleSim and MagicDraw



Source:SysML-Modelica Transformation Version 1.0

MBSE-driven visualization

MBSE-driven visualization is provided to transform related system models and related data to a virtualized way for different stakeholders (Jackson & Wilkerson 2016).



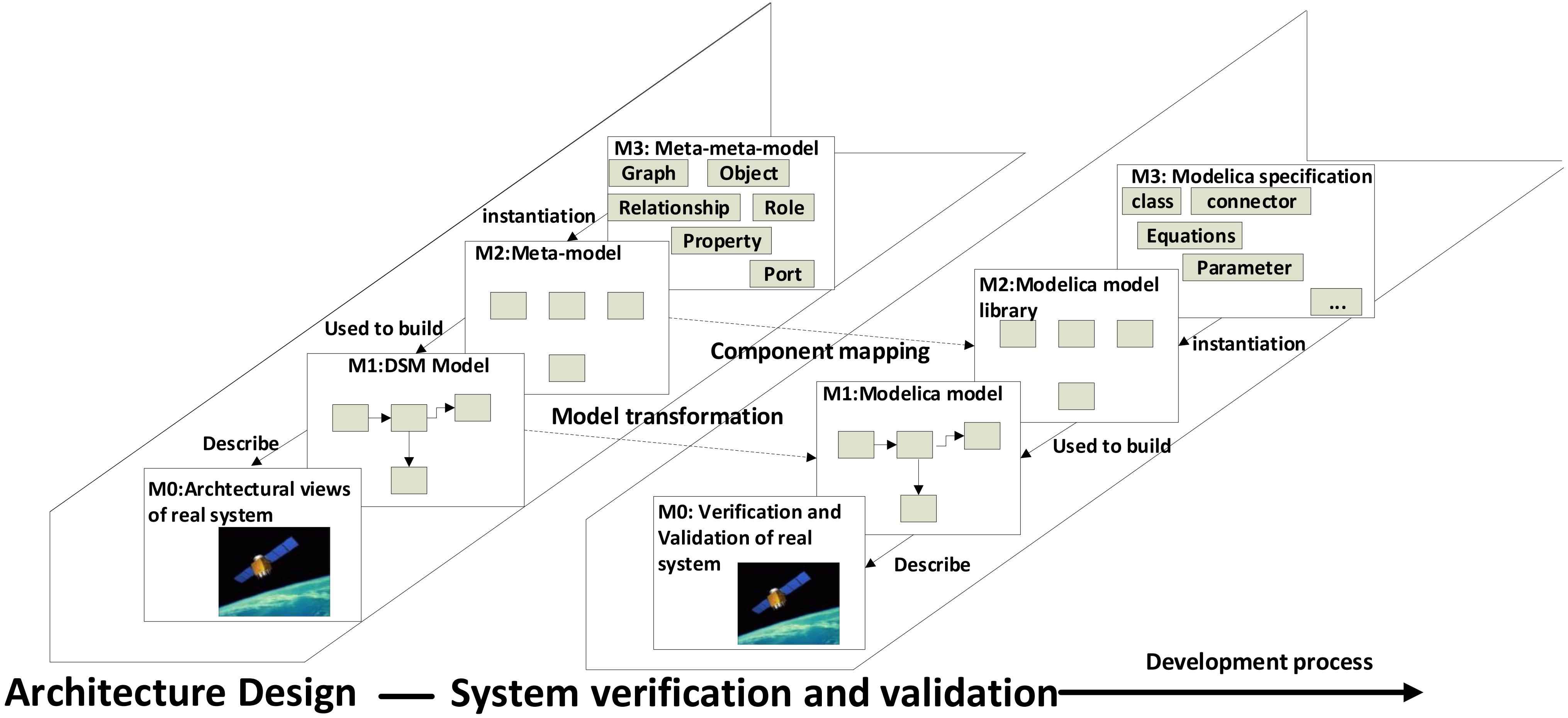
Source:MBSE-driven visualization of requirements allocation and traceability

SUMMARY

First of all, SysML was used to define the reference architecture models and then a model transformer is used to generate Modelica and Simulink models from SysML architecture models automatically. These models are used to verify the system performance of architecture models automatically without manual modeling processes. These authors put forward a framework for integrating system architecture models of SysML or DSMLs with the corresponding simulation models. Usually, a developed model transformation is used to support automated generations of simulation models to promote consistency of the whole design process.

3. Model-based Architecture Design and Automated Verification

3.1.1 Domain-specific Modeling Approach

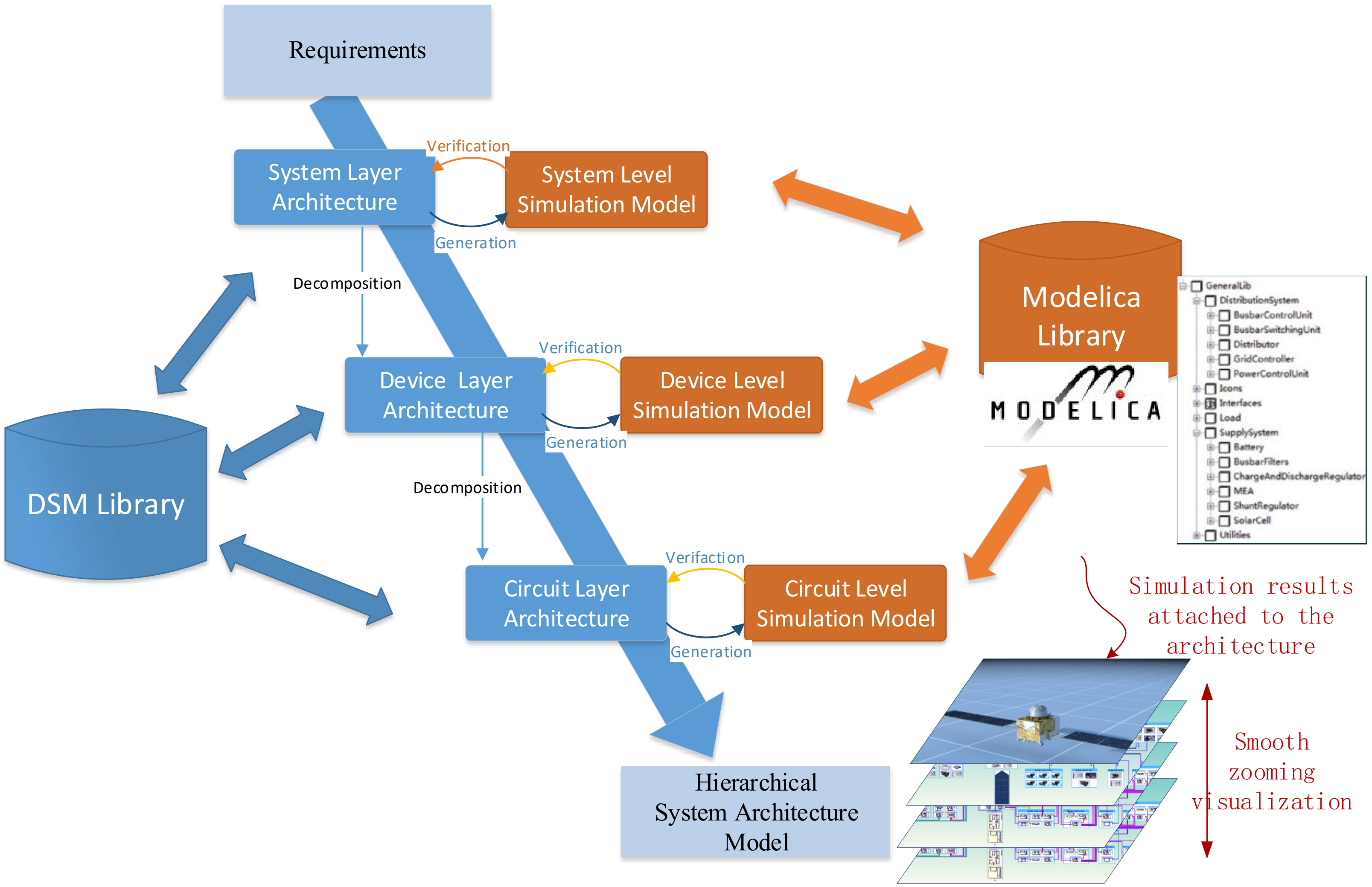


Developers make use of DSM to support architecture design and Modelica language to support system V&V in our tool. In order to support definitions and implementations of DSM and Modelica models, five modeling levels are used to construct model concepts in our tool.

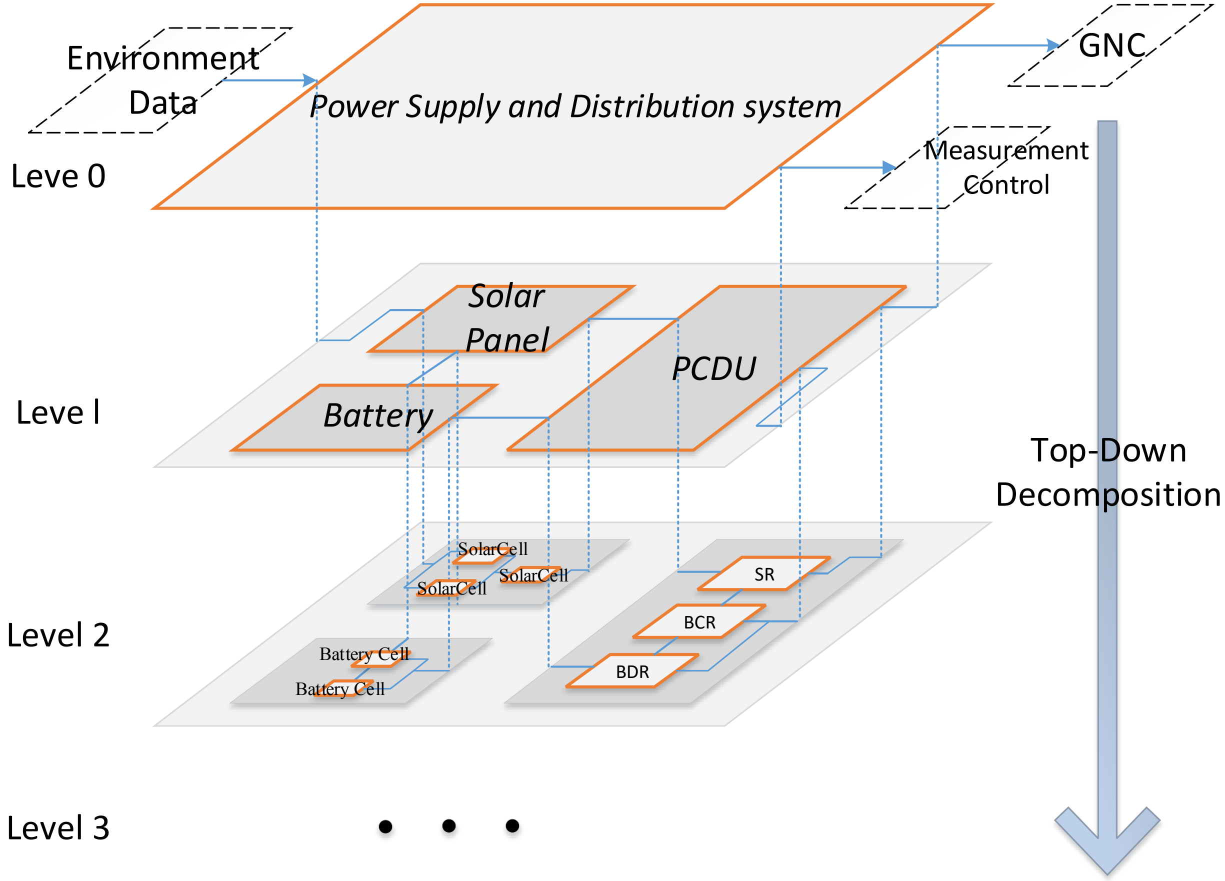
- M3 (Referring to Meta-meta-model and Modelica specification)
- M2 (Referring Meta-models and Modelica library)
- M1 (Referring to DSM and Modelica models)
- M0 (Referring to modeling purposes - views of stakeholders)

M1.5 meta-models templates for each languages

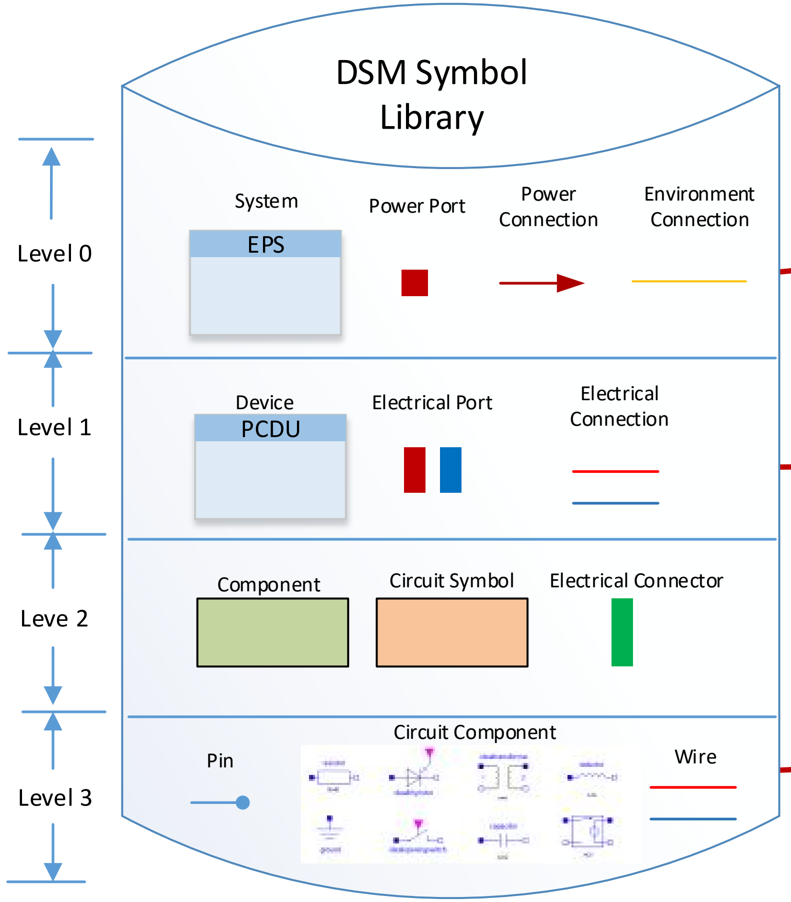
3.1.2 MBSE Workflow during Architecture Design

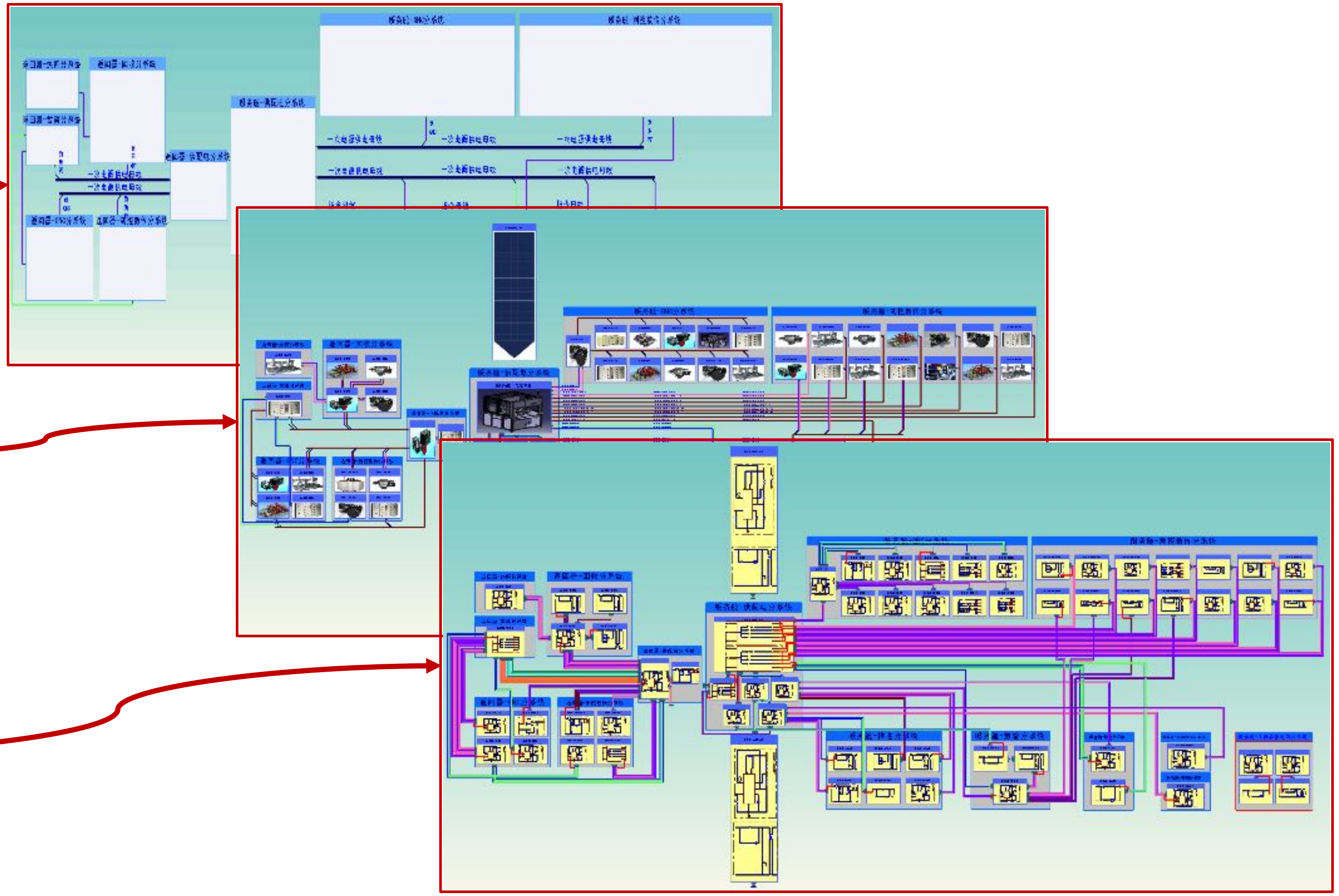


3.2.1 Hierarchical Architecture Design

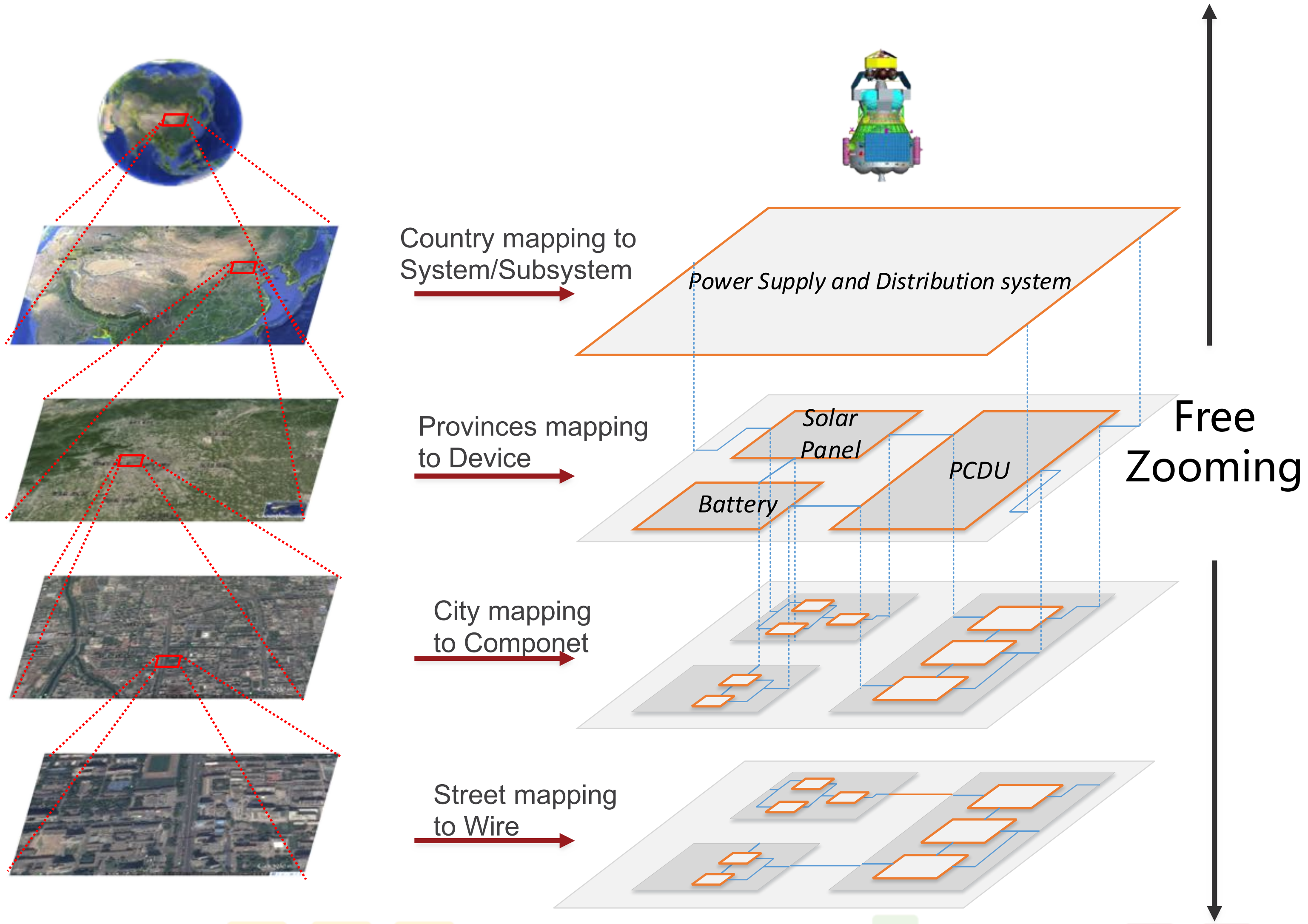


3.2.2 Domain-specific definitions of meta models

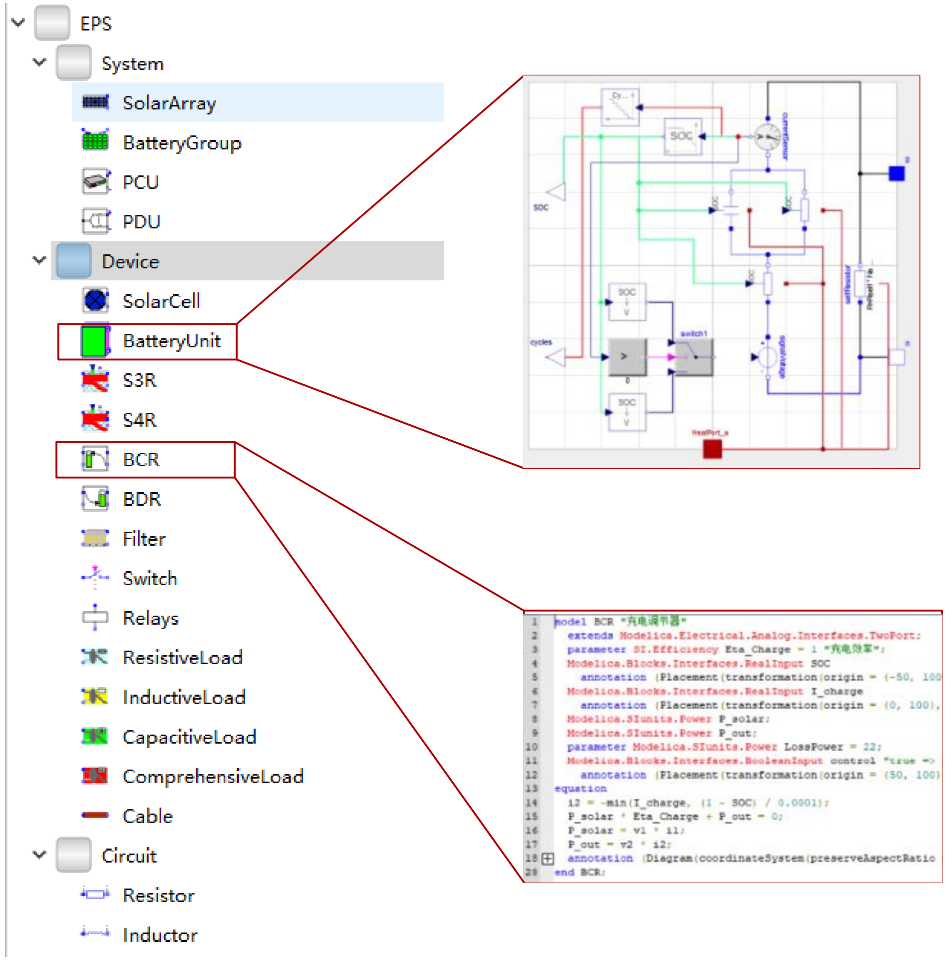




3.3 MBSE-driven Visualization of Architecture



3.4 Architecture-driven Method for Automated V&V



Modelica Model Library based on EPS

In *System Level 0*, high-level systems of spacecraft are built. There are three types of model in this level: *power supply system model*, *distribution system model* and *load system model*. The parameters of these models include *voltage*, *current*, *power*, etc.

In *System Level 1*, according to the functional requirements, the device compositions, control algorithm of the devices, and specify the power flow and key property of the devices are built. For example, Modelica model library in the device level includes models of *solar array*, *battery*, etc.

In *System Level 2*, internal components of the device are built, e.g., *battery cells*.

In *System Level 3*, internal circuits are built.

3.4 Architecture-driven Method for Automated V&V

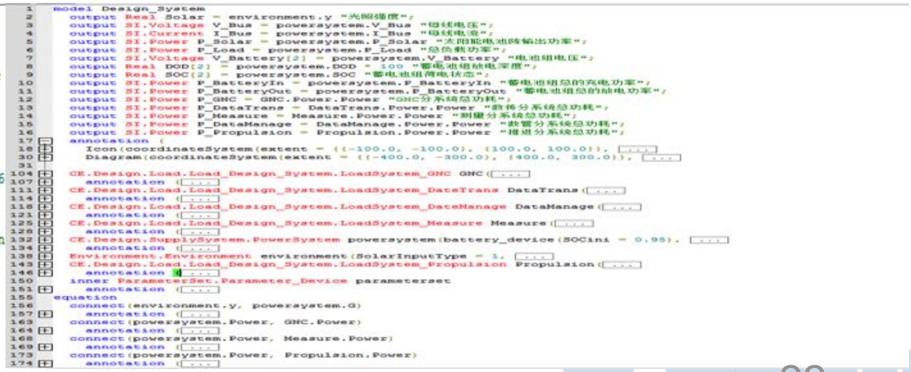
STEP1:Generating blocks of Modelica models in System Level 1

- **Block definition:** Modelica blocks related to devices are generated with the same meta-model name
- **Parameter definition:** Based on properties in the DSM models， parameters in Modelica models are declared
- **Interface definition:** Based on connections in the DSM models, interfaces of Modelica models can be generated
- **Equation definition:** Based on Modelica specification, equations of related domain specific concepts indicated by meta-models and equations of connectors are generated.

STEP2: Generating Modelica models in System Level 0

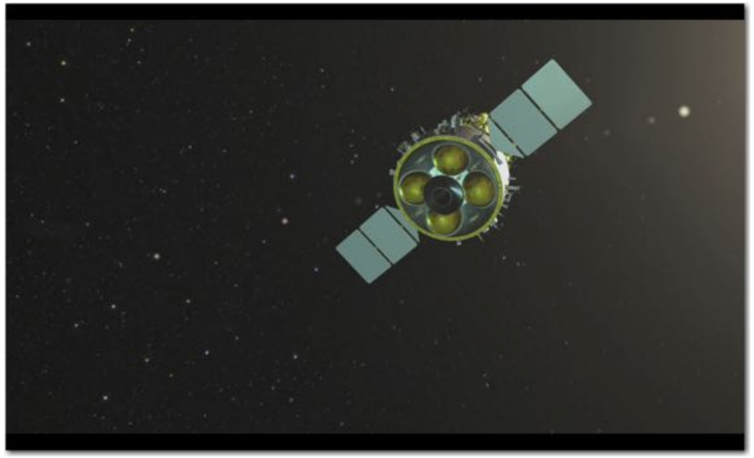
- **External connections of EPS definition:** Define the external interfaces of EPS.
- **Device definition:** Define devices in EPS.
- **Environment definition:** Define environment blocks based on Modelica model library.
- **Connector equation definition:** Define connector equations of connections between devices.

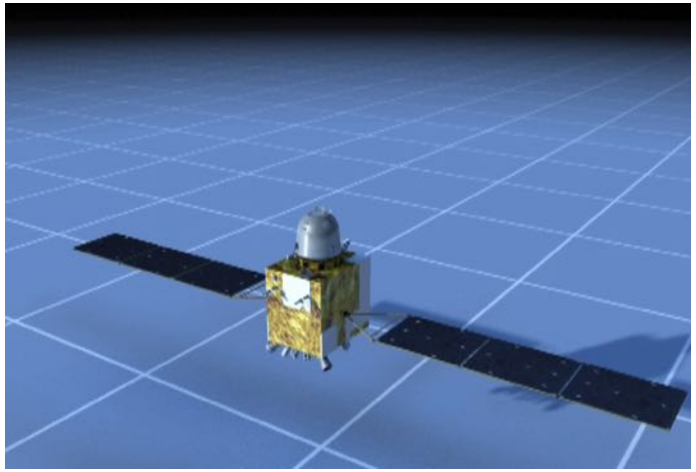
```
model Load1 "负载1模型"

  parameter Modelica.SIunits.Power PowerLoad = 20 "额定功率";
  parameter Modelica.SIunits.Voltage V_rated = 29 "输入额定电压";
  parameter Modelica.SIunits.Voltage Vset = 29 "输出额定电压";

  //电连接器
  Interfaces.Special.X01 X01[2]
    annotation (Placement(transformation(extent = {{-26.18, -26

  //控制开关
  Modelica.Blocks.Interfaces.BooleanInput control "true => p--n
    annotation (Placement(transformation(extent = {{-20, -20},
  annotation (Icon(graphics = {...
equation

end Load1;
```

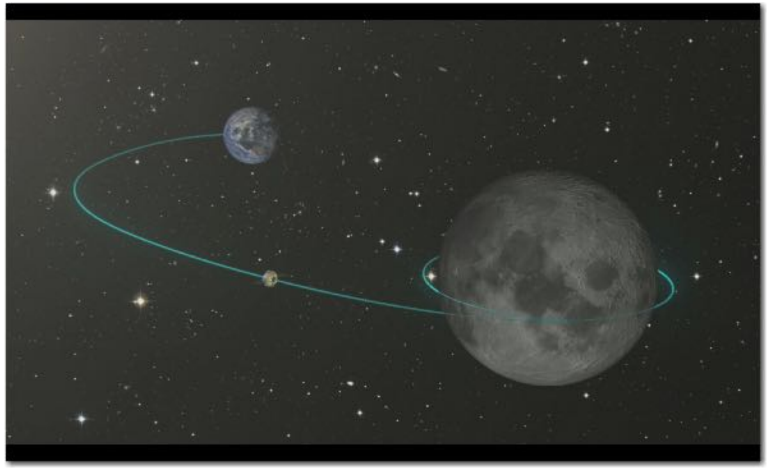


4. Case study and Evaluation

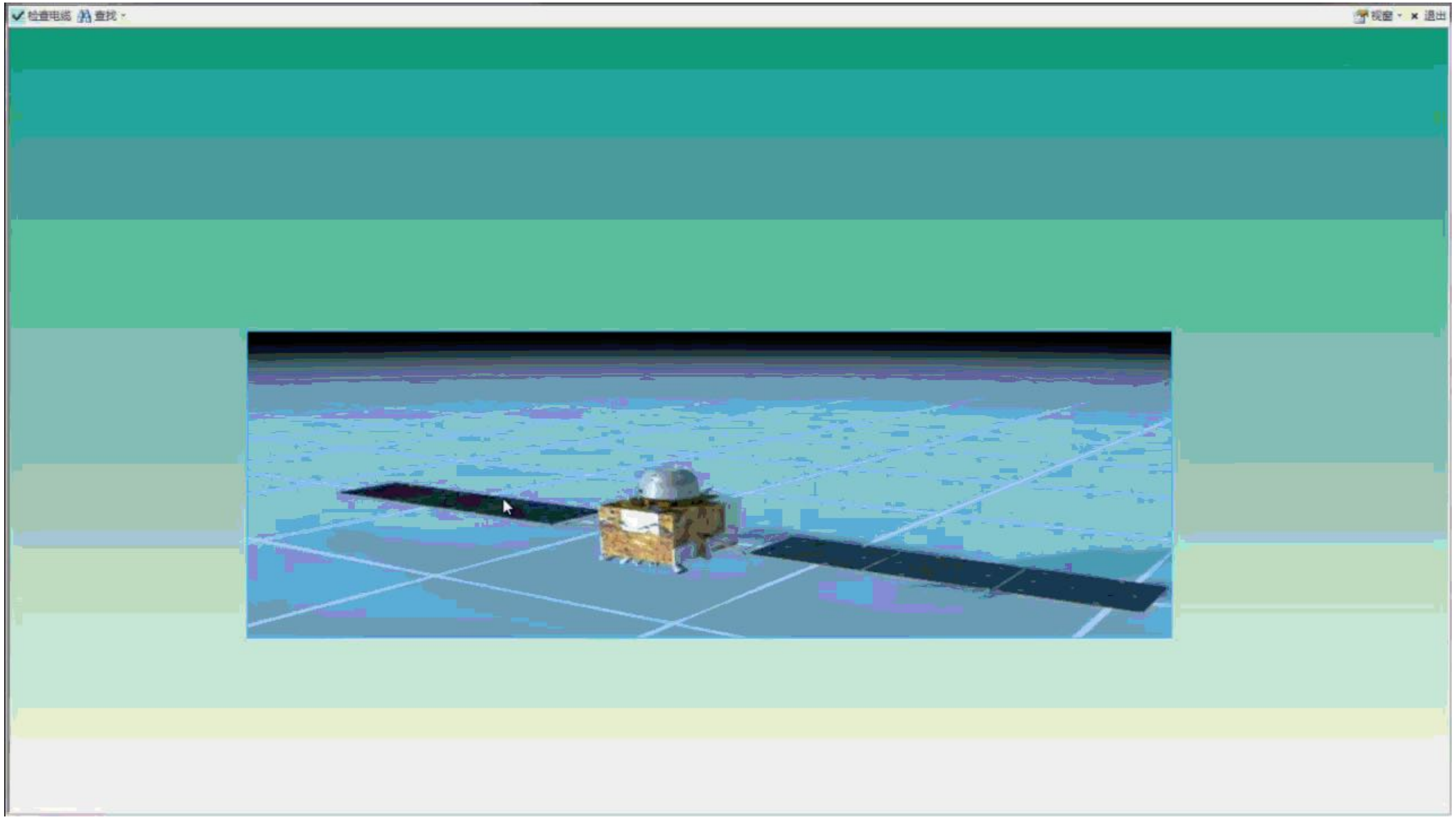
The returning spacecraft composes of service and return modules. The service module includes two busses(+Y and -Y) for two polar arrays. Each bus is connected to a solar array, hydrogen-nickel battery and the related power supply controller on one side of the returning spacecraft. By the main distributor, thetwo buses are integrated and connected with the loads in order to supply electric powers.



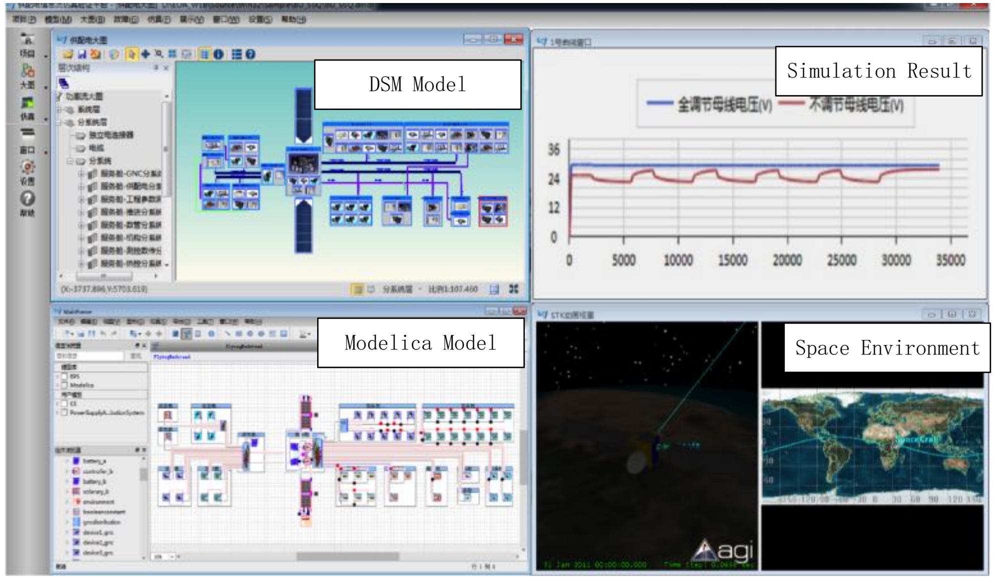


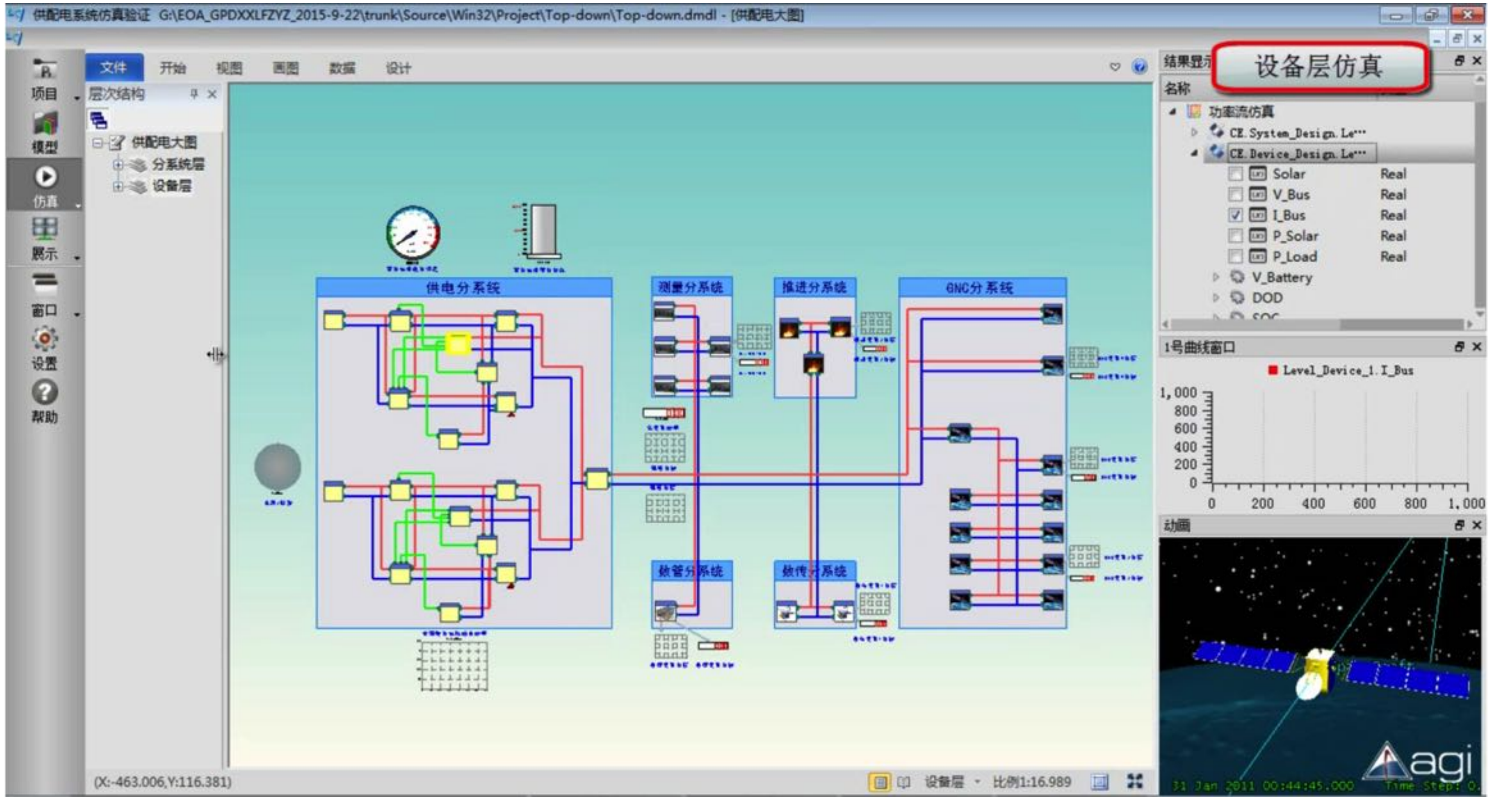


Visualizations of a returning spacecraft



DSM Models and related Modelica models mapping to DSM models of a returning spacecraft





Learns

- Current MBSE tool supports development of architecture models of EPS and visualizations of architectural information. Based on the provided meta-meta models, several meta-models are developed for building the architecture models.
- Compared with existing languages like SysML and UML, such meta-models are more simplified and less time-consumption when engineers start their MBSE industry practices.
- Different stakeholders can make use of the MBSE tool to browse the architectures of EPS which can help stakeholders to understand each other’s views and promote the communications between the whole team.

Limitations

There are several limitations about the MBSE tool:

(1) The links between requirements and architecture models will be done in the future.

(2) The model transformer in current tool is limited. In the future, a code generator editor will be developed in order to transform architecture models to other DSM models, data and code.

(3) In the future, our MBSE tool can support to develop meta-models based on existing languages such as SysML and AADL.

5. Conclusion

In this paper, an MBSE tool has been proposed to support architecture design, MBSE-driven visualization and automated V&V using Modelica language for EPS.

- A DSM technique is adopted by the tool to design related architecture models.
- Moreover, the MBSE-driven visualization in our tool can represent system architecture in multiple views.
- Furthermore, a model transformer is developed to support architecture-driven V&V which can generate related Modelica models from architecture models automatically.
- Finally, through a real case of EPS in spacecraft, our tool indeed promote the effectiveness of system developers when they design architectures.

28th Annual INCOSE international symposium

Washington, DC, USA
July 7 - 12, 2018

www.incose.org/symp2018