International Council on Systems Engineering
A better world through a systems approach

Next Generation MBPLE with SysML v2: Feature Modeling, Variability Modeling and API Potentials

Presenter: Tim Weilkiens – oose
Presenter: Marco Forlingieri – PTC Inc.
Co-Author: Vince Molnár – Budapest University of Technology and Economics

INCOSE International Symposium 2025 | Ottawa, Canada

TIM WEILKIENS

Tim is a consultant and executive board member of oose. He has more than 20 years experience in the field of MBSE. He is one of the core developers of SysML v1 and SysML v2. Tim published many books about modeling and he is author of the MBSE methodology SYSMOD. With VAMOS he published a tool-independent approach for modeling variants with SysML. He is author of the book MBPLE: The Feature-Based Path to Product Lines Success, published by Wiley.

oose.

MARCO FORLINGIERI

Marco senior director for Product Line Engineering (PLE) at PTC, leads the biggest PLE expert´s pool in the world. He has several years of experience in the field of MBSE and PLE mainly within aerospace, defense, automotive and railway industries in Europe, China and North America. He chairs the INCOSE PLE Working Group and is the leading author of the book MBPLE: The Feature-Based Path to Product Lines Success, published by Wiley.

Introduction

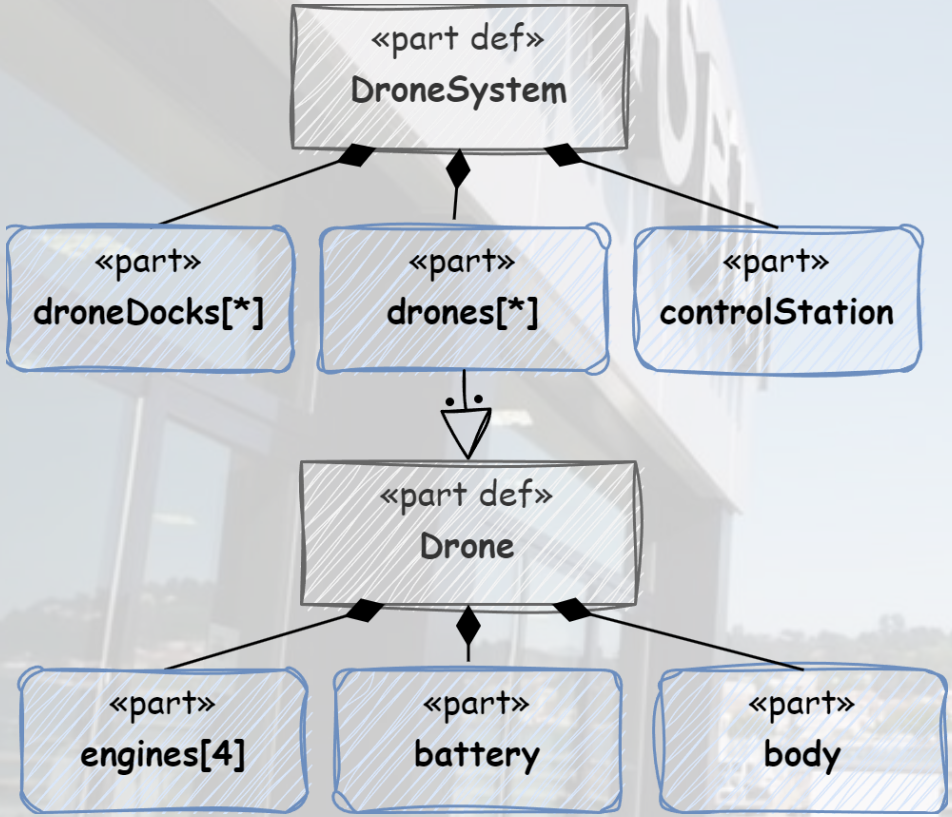

Riproduci (k)

MBPLE and SysMLv1 =

MBPLE and SysMLv2 =

Variability Modeling with SysML v2

First version of the DroneSystem



Change Requests!

- ✓ *Different number of engines*
- ✓ *Different kinds of batteries*
- ✓ *Different payload attachments*
- ✓ …

Variant Modeling Methods

INCOSE International Symposium

Session Track 4: Product Line Engineering

Two Variant Modeling Methods for MBPLE at Airbus

Marco Forlingieri, Tim Weilkiens

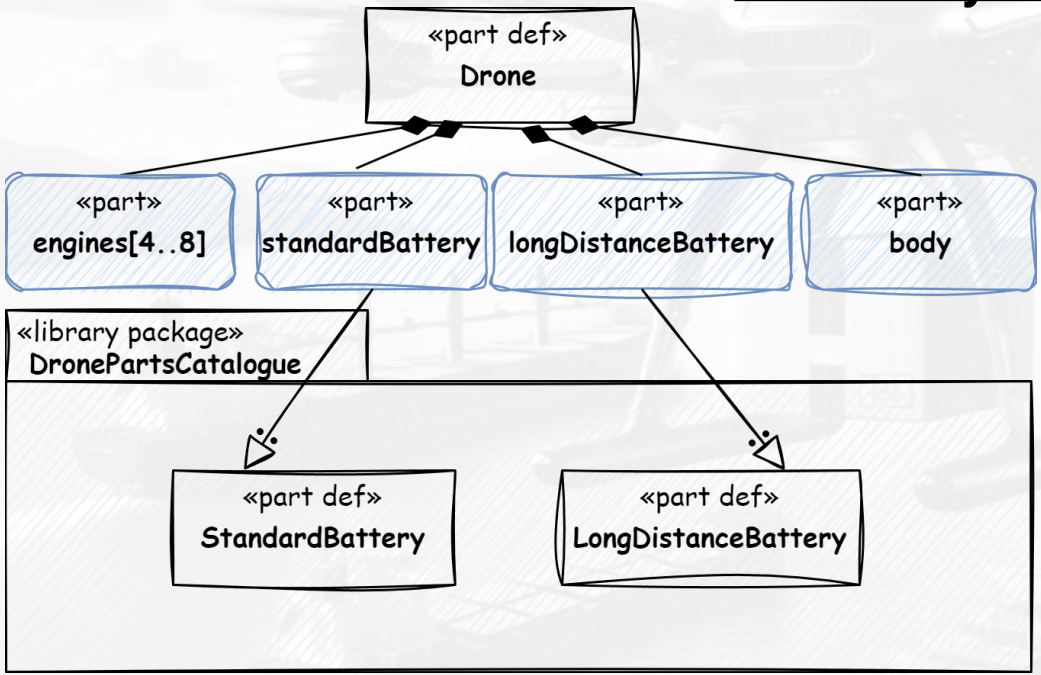
First published: 26 September 2022 | https://doi.org/10.1002/iis2.12984 | Citations: 10

Read the full text > PDF TOOLS SHARE

Abstract

Product Line Engineering (PLE) is a fundamental component of systems engineering. As postulated in INCOSE's Vision 2025, Model-Based Systems Engineering (MBSE) is becoming the norm for systems engineering. Thus, we need to integrate PLE into MBSE. An important aspect of this integration is the definition of model-based methods. This paper defines two possible approaches for variant modeling with SysML applied at Airbus and provides insights into their application in practice.

Direct Approach in a Nutshell

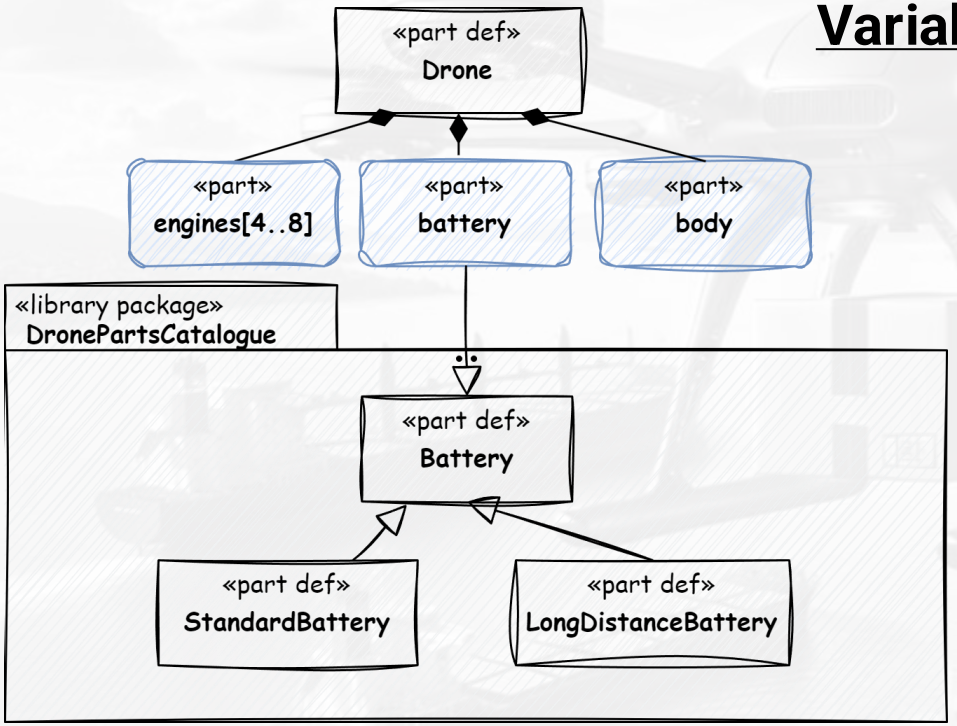


Variability: Different kinds of batteries

Just model it!

A drone can have a standard battery or a long distance battery.

Clean Approach in a Nutshell



Variability: Different kinds of batteries

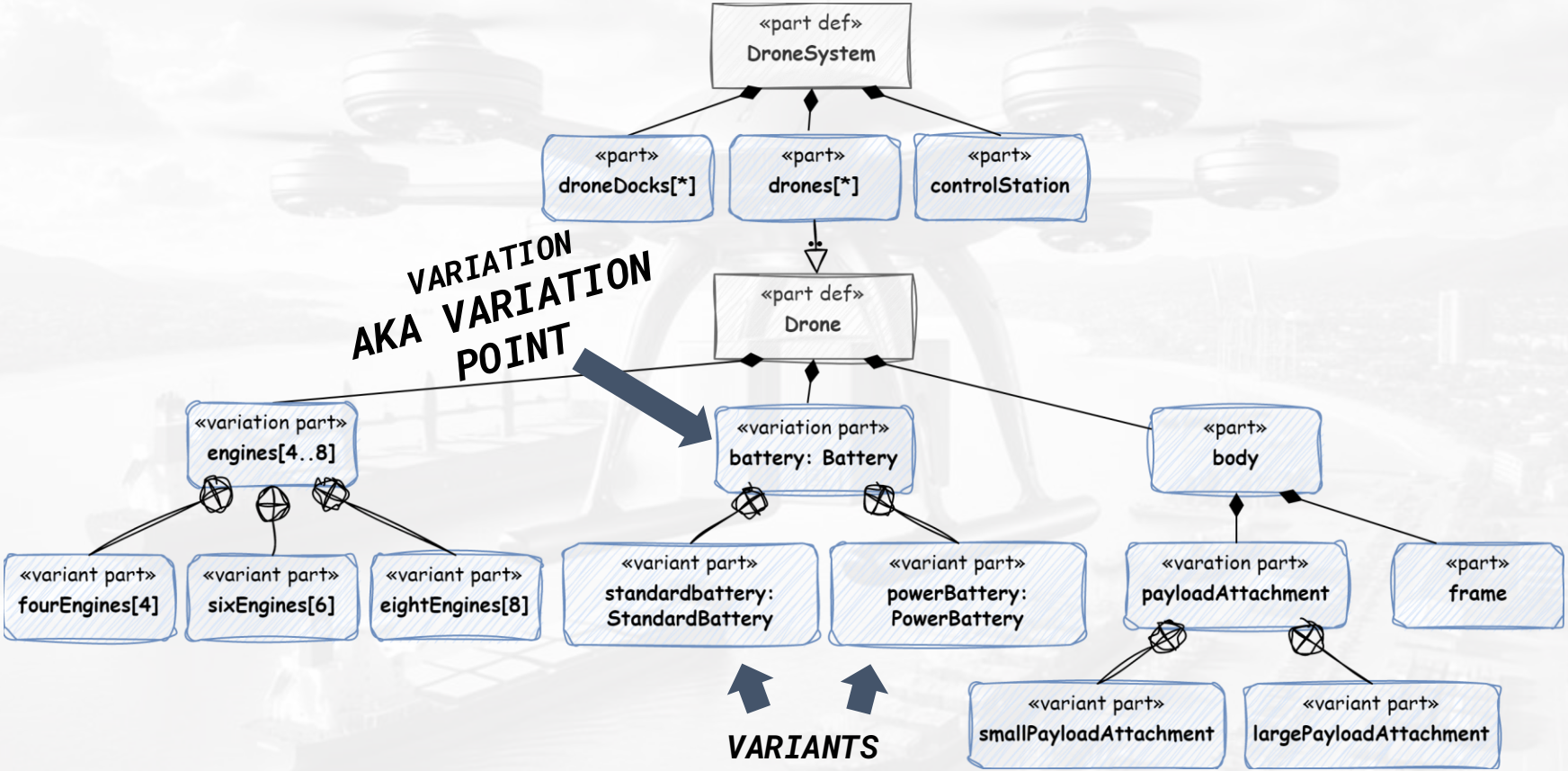
1. Identify the commonality!

A drone has a battery.

2. Specialize the variants!

It can be a StandardBattery or a LongDistanceBattery.

SysML v2 Variations and Variants

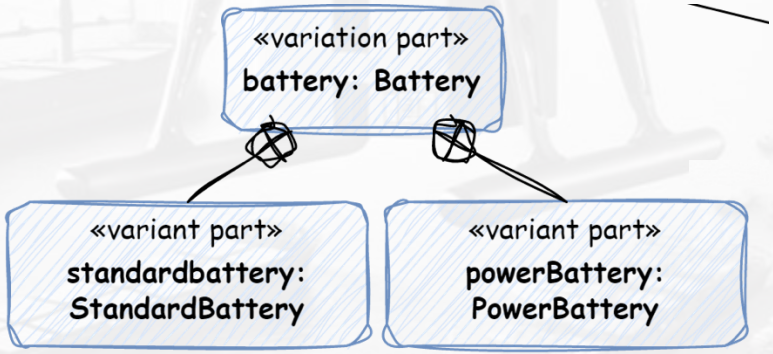


SysML v2 Variations and Variants

Variants are not individual model elements. In the background, it is a VariantMembership relationship between the variation and a variant.

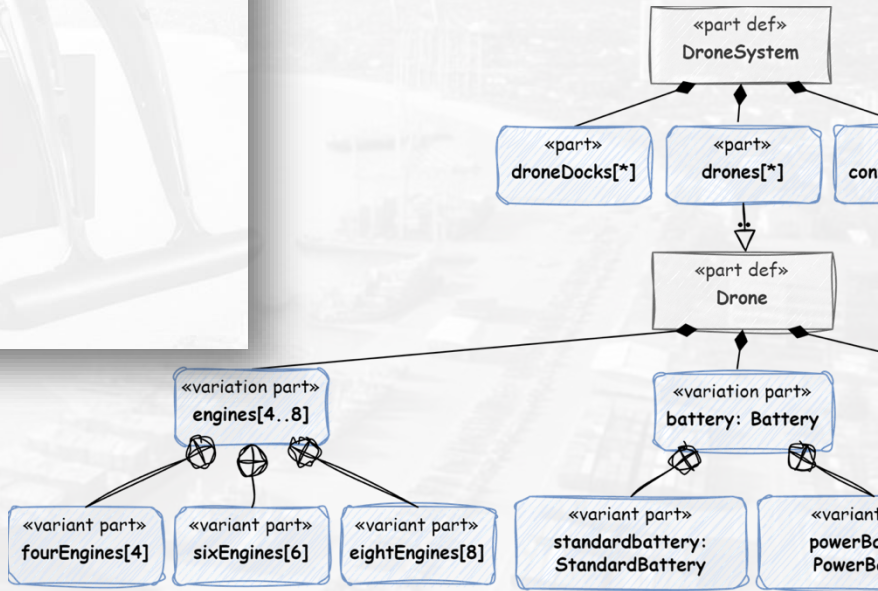
Being a variation is a Boolean characteristic of a definition/usage element that can be switched on or off.

.



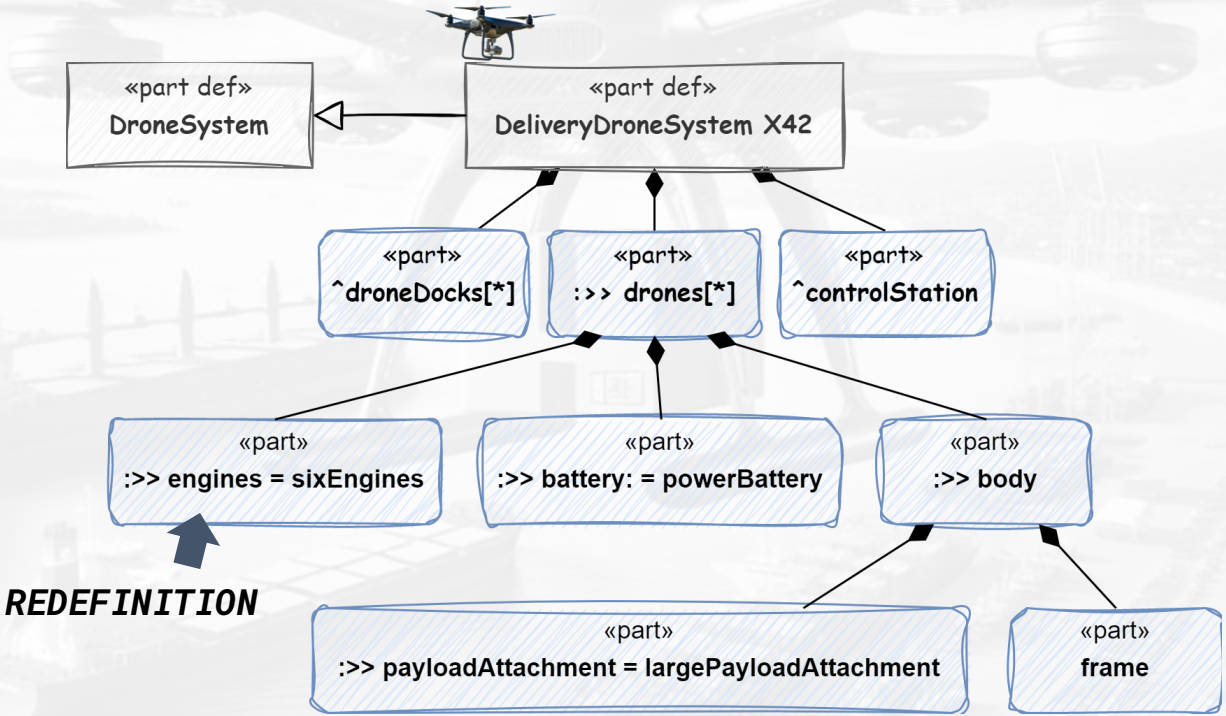
Variations/Variants - Textual Notation

```
part def Drone {
  variation part battery: Battery {
    variant part standardBattery: StandardBattery;
    variant part powerBattery: PowerBattery;
  }
  variation part engines[4..8] {
    variant part fourEngines[4];
    variant part sixEngines[6];
    variant part eightEngines[8];
  }
  […]
}
```



Product Model - Let's derive a product

The **DeliveryDroneSystem X42** has
six engines, a power battery, and an attachment for large payloads.



ISO/IEC 26580

Shared Assets Supersets

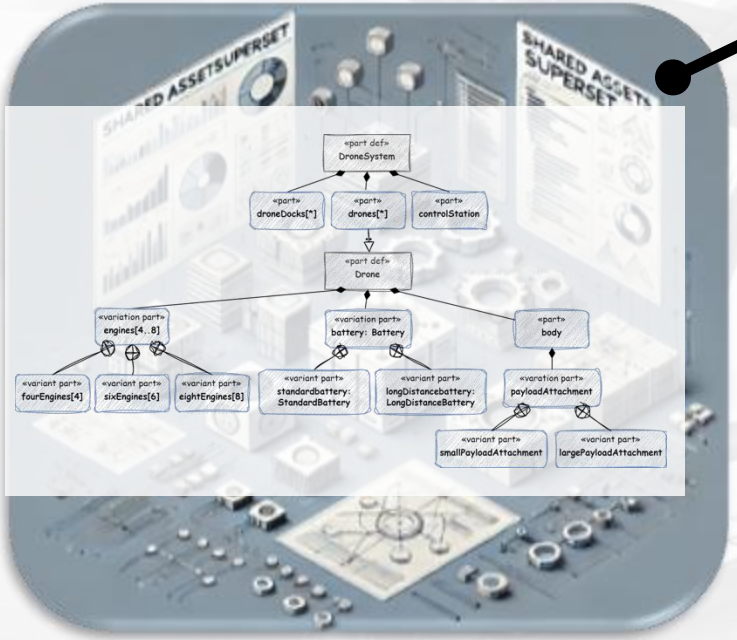
Feature Model: Feature Tree & Feature Configurations

PLE Configurator

Product Models

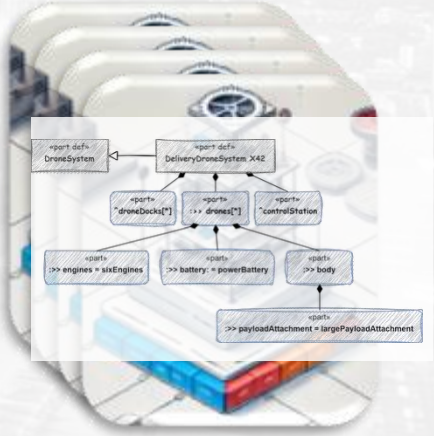
ISO/IEC 26580

Feature Model: Feature Tree & Feature Configurations



Shared Assets Supersets

PLE Configurator

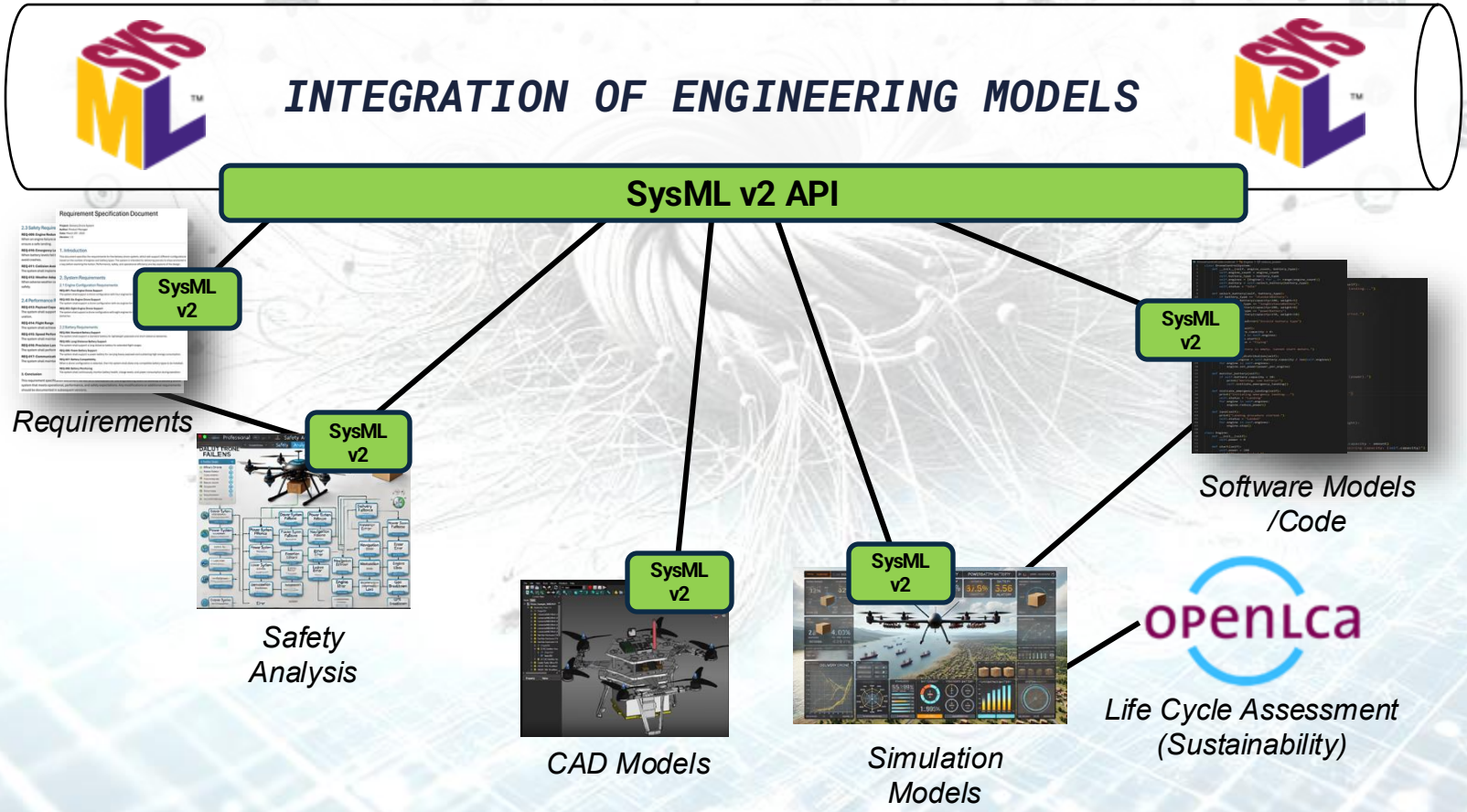


Product Models

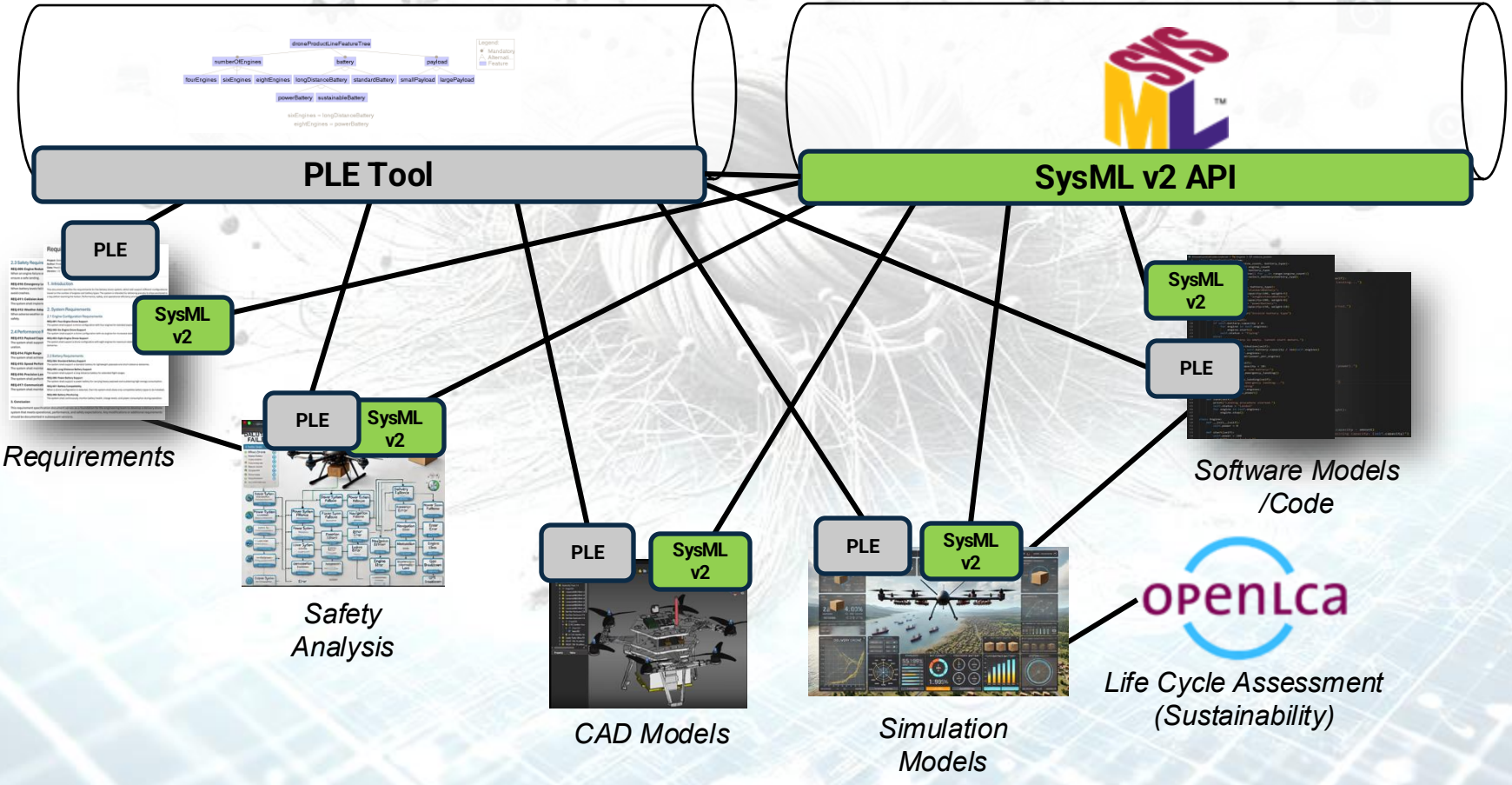
Feature Modeling with SysML v2

Feature Tree with FeatureIDE

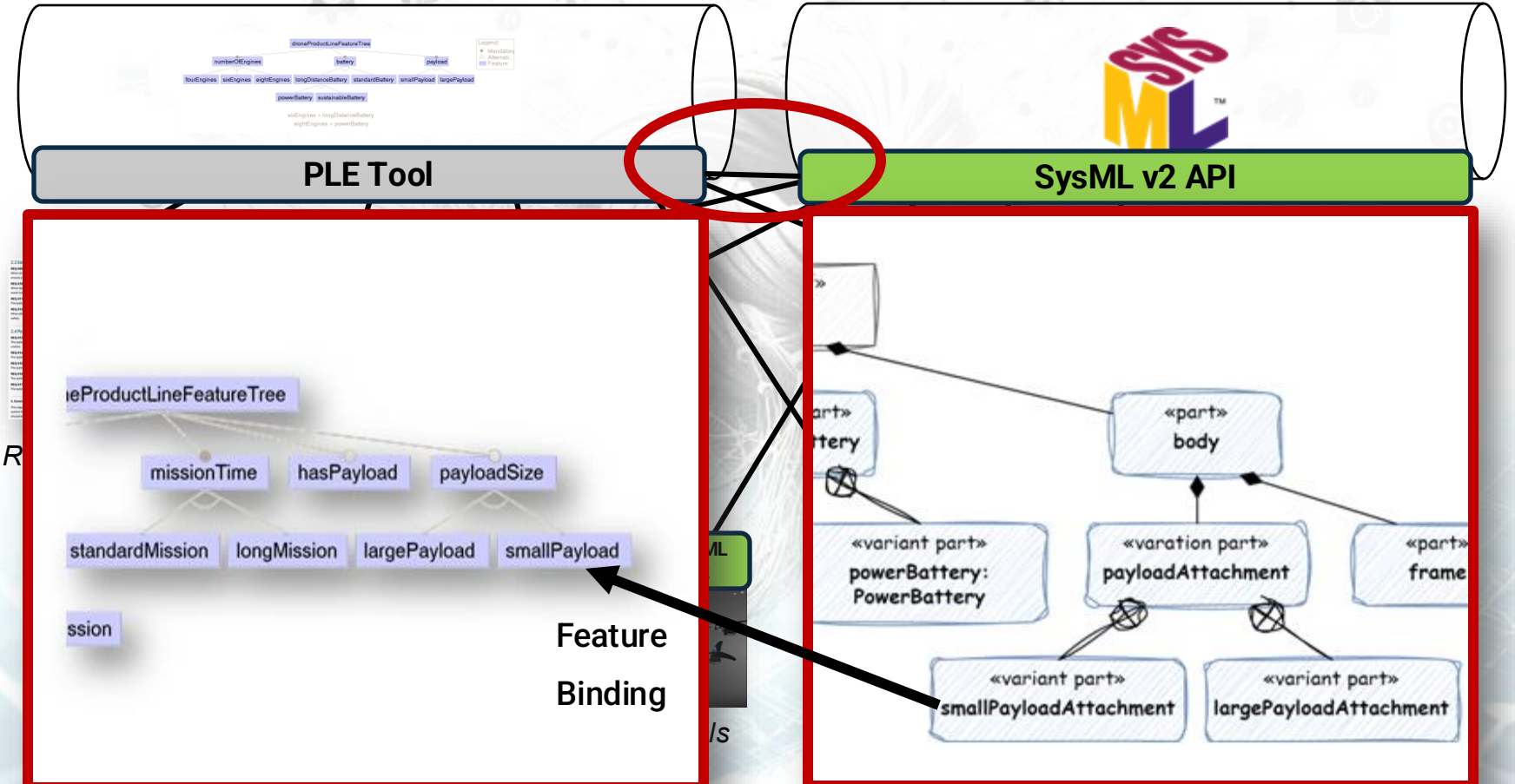
SysML Model as the Backbone of the Digital Thread



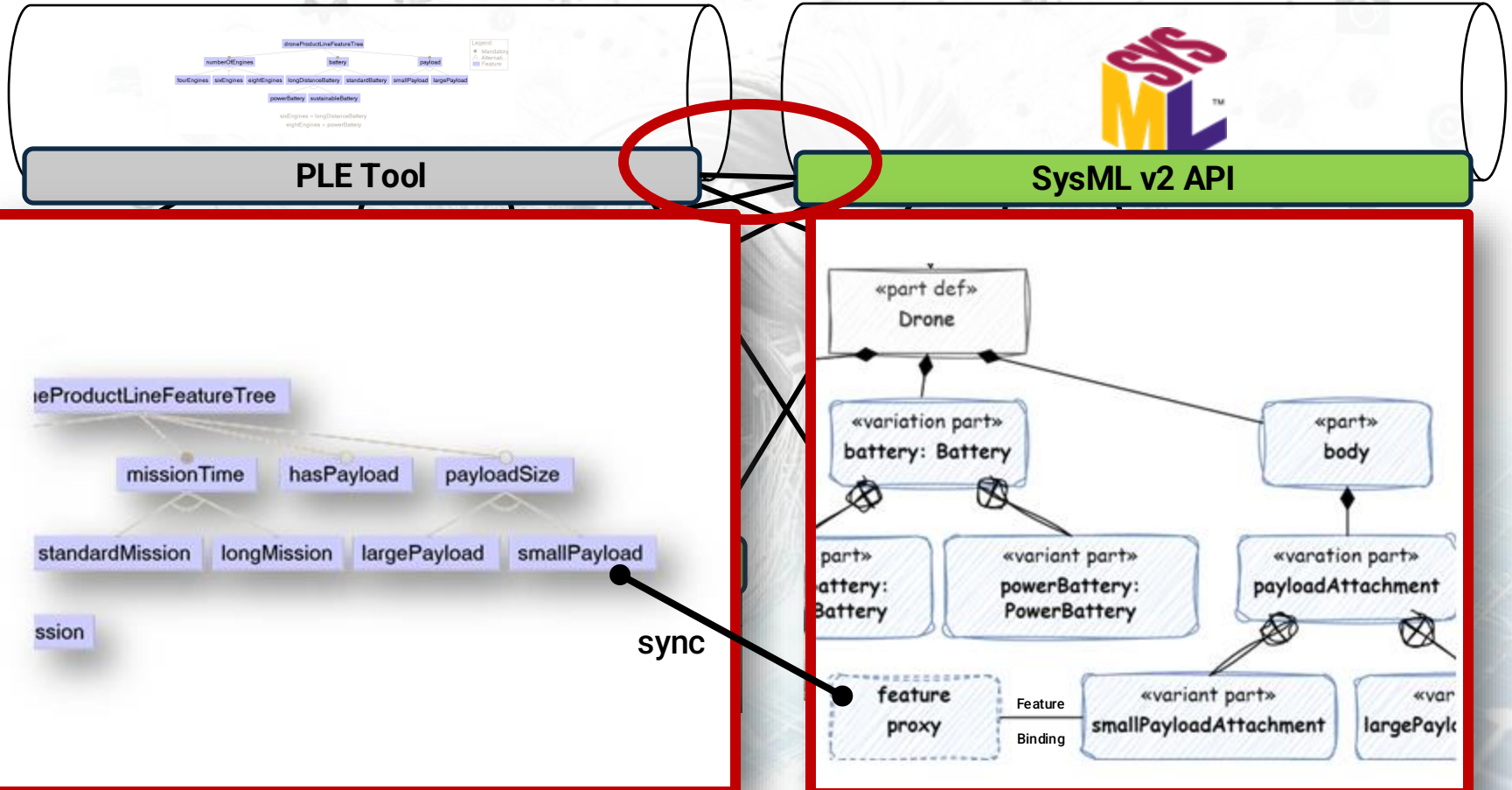
PLE in the Digital Thread



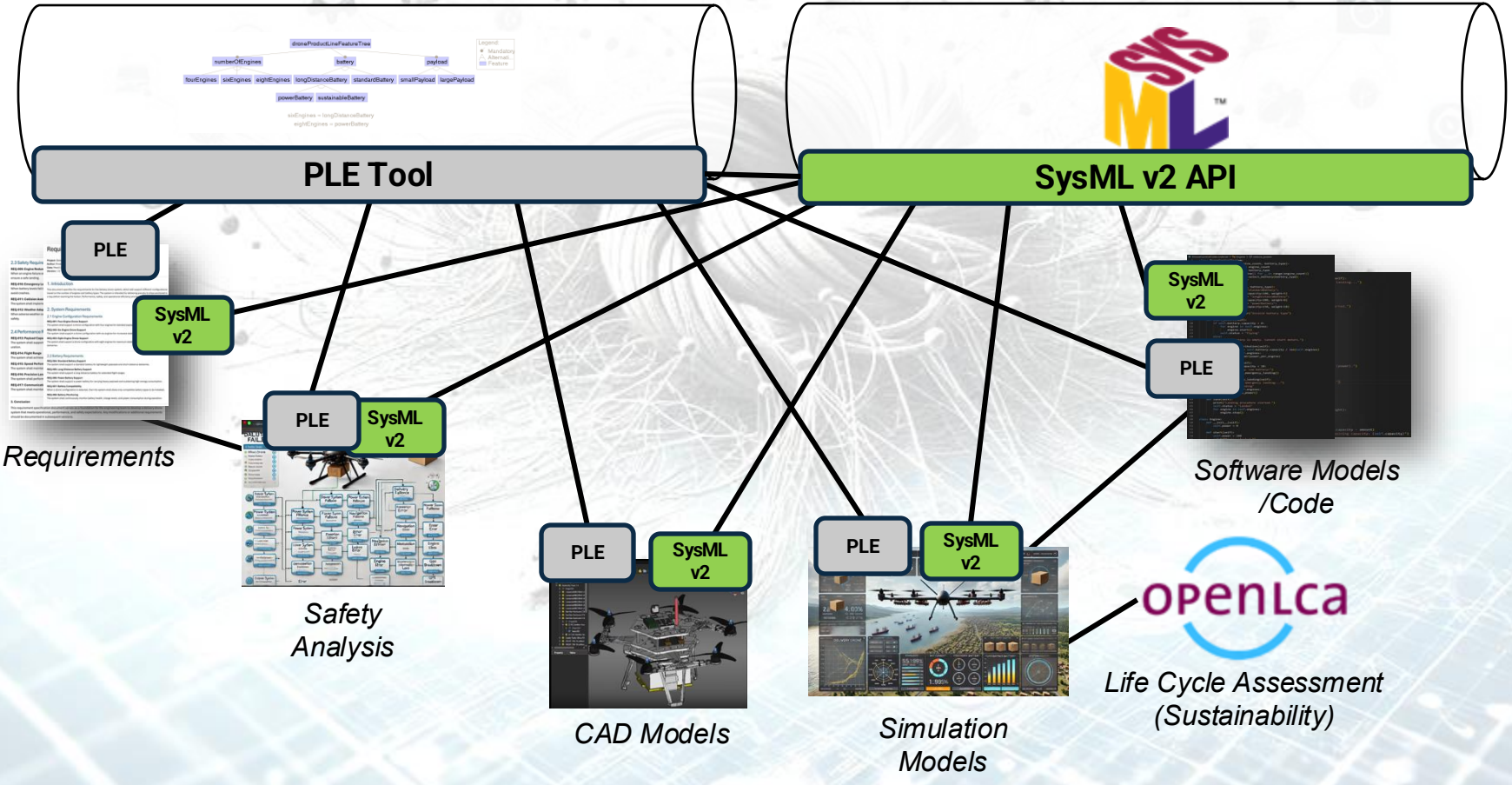
PLE in the Digital Thread



PLE in the Digital Thread

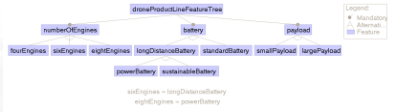


PLE in the Digital Thread



PLE in the Digital Thread

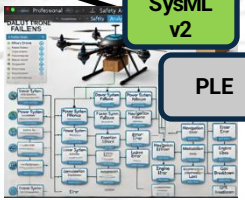
PLE Tool



SysML v2 API

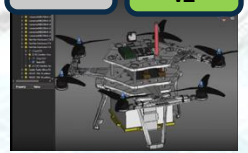
SysML v2 PLE

Requirements



SysML v2 PLE

Safety Analysis



PLE SysML v2

CAD Models

SysML v2 PLE

Simulation Models

SysML v2 PLE

Software Models /Code

openLCA

Life Cycle Assessment (Sustainability)


1. Foundation for Feature and Feature Binding in SysML v2 as an anchor point for Feature Modeling Tools
2. Feature Modeling with SysML v2
SysML v2 – Language Extension

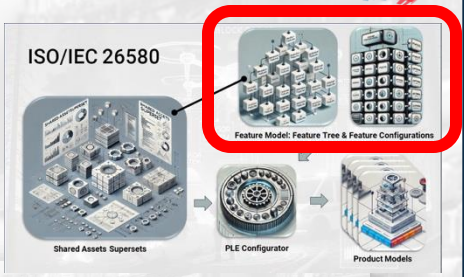
Laboratory: SysML v2 for Feature Modeling

PLEML Library

```
library package FeatureModeling {
  enum def LifeCycleKind :> ScalarValues::Natural {
    doc /* Definition of f
         * The numbers pro
         */
    enum Development = 1;
    enum Production = 2;
    enum Operation = 3;
  }
  occurrence def Feature {
    doc /* A Feature is a
         * product line th
         * in the product line.
         */
    attribute bindingTime : LifeCycleKind default LifeCycleKind::Development {
      doc /* The binding time of the feature with default Development. */
    }
  }
}
```

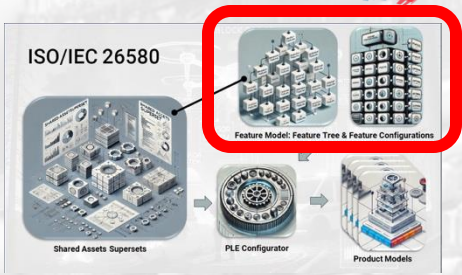
```
constraint def XORConstraint :> FeatureConstraint {
  in occurrence excluded :> constrainedFeatures;
  in occurrence featureConfiguration :>> featureConfiguration;
  if featureConfiguration != null ?
    featureConfiguration.suboccurrences->includes(owningFeature)
      implies (1..size(excluded)-1)->ControlFunctions::forAll {
        in f; not featureConfiguration.suboccurrences->includes(f)
      }
    else
      true
  }
}
```

Feature Representation - Textual



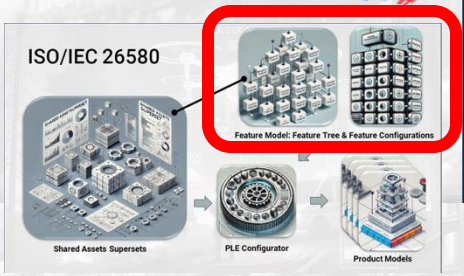

```
package DroneFeatureModel {
  private import FeatureModeling::*;

  #feature def droneProductLineFeatures {
    #feature missionTime {
      attribute :>> bindingTime = LifeCycleKind::Production;
    }
    #feature standardMission :> missionTime;
    #feature longMission :> missionTime;
  }
}
```

Feature Configuration




```
#featureConfiguration deliveryDrone :> droneProductLineFeatures {
    #feature :>> missionKind = (deliveryMission);
    #feature :>> missionTime = (longMission);
    #feature :>> hasPayload;
    #feature :>> payloadSize = largePayload;
}
#featureConfiguration observationDrone :> droneProductLineFeatures {
    #feature :>> missionKind = (forestFireMission);
    #feature :>> missionTime = (longMission);
}
#featureConfiguration consumerDrone :> droneProductLineFeatures {
    #feature :>> missionKind = (securityMission, funMission);
}
```

Feature Binding




```
// Feature Bindings from Shared Assets to the Features
#featureBinding connect droneSystemRequirements.minimumFlightTime::standar
  to droneFeatureModel.droneProductLineFeatures.standardMission;
#featureBinding connect droneSystemRequirements.minimumFlightTime::extende
  to droneFeatureModel.droneProductLineFeatures.longMission;

#featureBinding connect droneProduct.battery::standardBattery
  to droneFeatureModel.droneProductLineFeatures.standardMission;
#featureBinding connect droneProduct.battery::longDistanceBattery
  to droneFeatureModel.droneProductLineFeatures.longMission;
#featureBinding connect droneProduct.battery::longDistanceBattery::longRan
  to droneFeatureModel.droneProductLineFeatures.deliveryMission;
#featureBinding connect droneProduct.battery::longDistanceBattery::longRan
  to droneFeatureModel.droneProductLineFeatures.forestFireMission;

#featureBinding connect droneProduct.body::bodyWithoutPayload
```

What's next!

- Complete the language extension PLEML
- Variation WG for SysML v2.1
  - Optional variation
  - Constraints
  - …

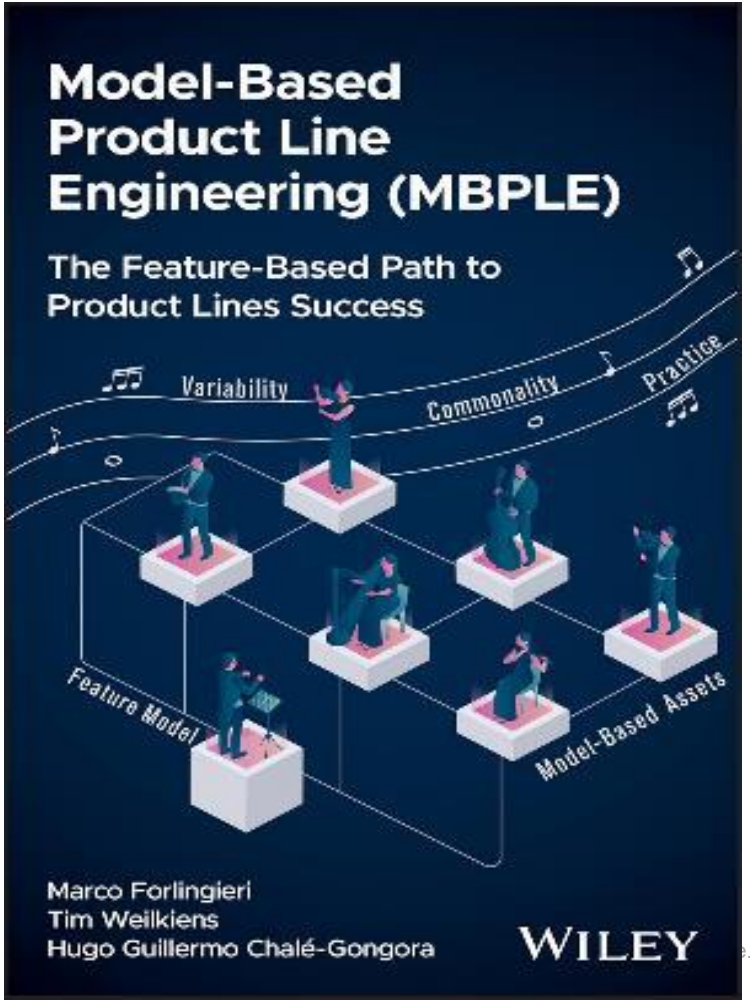
Thank You

Question Time!

Contacts:

mforlingieri@ptc.com

tim.weilkiens@oose.de



LOCKHEED MARTIN ptc

PRODUCT LINE ENGINEERING IN ACTION

Master complexity and deliver at scale

Week of October 20th

Lockheed Martin Facility, Orlando, FL

Stay tuned for more details and registration information.

- Join fellow architects, software, and systems engineers for a PLE User Group to hear expert insights and best practices from industry leaders.