

# **GETTING THE MOST OUT OF TEAMWORK CLOUD (MAGIC COLLABORATION STUDIO)**

**Tomas Vileiniskis**

CATIA Systems Modeling Application Senior Manager  
Teamwork Cloud Product Manager



## Tomas Vileiniškis

### Teamwork Cloud Product Manager

- 18 years of experience in Computer and Software Engineering
- Leading Teamwork Cloud development for more than 7 years
- 3DS representative at OMG SysML v2 API and Services RTF
- Active member of OMG RAS 3.0 (Reusable Assets) RFC

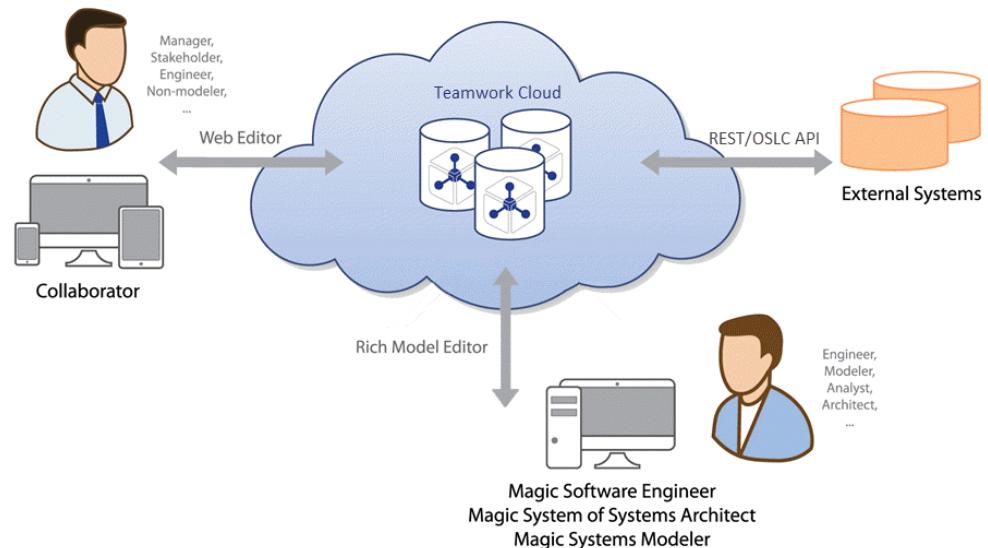
# WHAT IS MAGIC COLLABORATION STUDIO?

- Magic Collaboration Studio is CATIA No Magic's bundle of products for collaborative model development
  - Teamwork Cloud
  - Cameo Collaborator



## Teamwork Cloud

**CAMEO<sup>®</sup>**  
COLLABORATOR



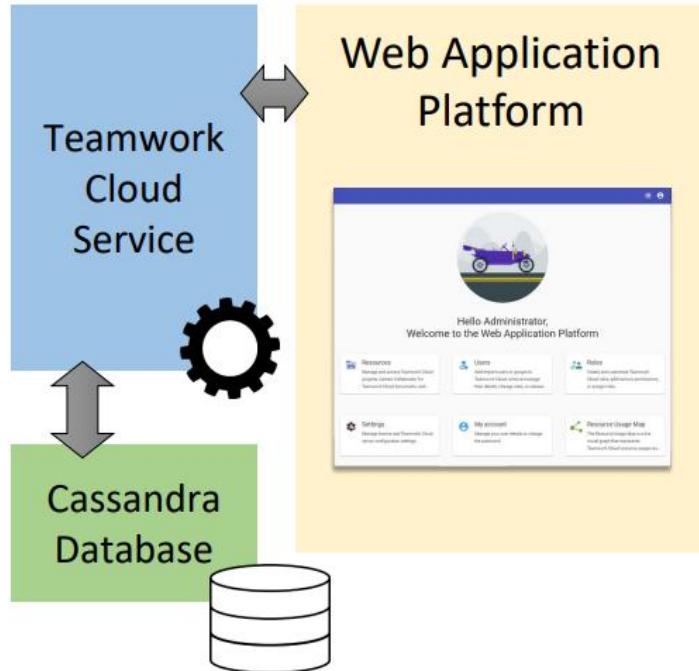
# OUTLINE

- Introduction
- Key Model Management Principles
- APIs and Services
- Recent feature highlights

# INTRODUCTION

# WHAT IS TWC?

- Teamwork Cloud is CATIA No Magic's repository for collaborative model development and model version storage.
- As the name suggests, this product is Cloud-ready, and it can be deployed on a local hardware (VM) or on a hosted server in the Cloud.
- Teamwork Cloud can be accessed both by Cameo modeler and web interface
- Teamwork Cloud also exposes its data via REST and OSLC APIs



# WHAT CAN I DO WITH TWC?

- Collaborative modeling (internal and external)
- Model versioning
- Model history
- Model branching and merging
- Access control
- Model and repository analysis
- Model exchange

# INTERACTING WITH TWC



Hello Administrator,  
Welcome to the Web Application Platform



## Resources

Manage and access Teamwork Cloud projects, Cameo Collaborator for Teamwork Cloud documents, and OSLC resources.



## Users

Add/import users or groups to Teamwork Cloud, remove/manage their details, change roles, or release locked elements.



## Roles

Create and customize Teamwork Cloud roles, add/remove permissions, or assign roles.



## Settings

Manage license and Teamwork Cloud server configuration settings.



## Resource Usage Map

The Resource Usage Map is a live visual graph that represents Teamwork Cloud resource usages as well as identifies potential problem areas.



## My account

Manage your user details or change the password.

# KEY MODEL ORGANIZATION AND MANAGEMENT PRINCIPLES



# RESOURCE ORGANIZATION

- Resource types

- Cameo models



- Collaborator documents

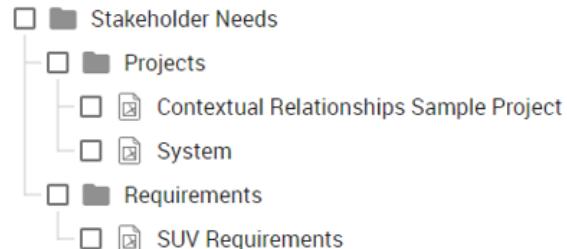


- OSLC documents



- Resources are grouped into categories

- Categories can be nested



Name	Last modified	Branch
Stakeholder Needs	Monday, April 19, 2021 3:00:06 PM	trunk
Projects		
Contextual Relationships Sample Project	Monday, April 19, 2021 3:00:06 PM	trunk
Requirements	Thursday, April 8, 2021 12:19:47 PM	trunk
SUV Requirements		

# ACCESS CONTROL

- Teamwork Cloud uses Role Based Access Control (RBAC) mechanism to handle access to resources and categories in the repository
- A predefined set of Roles comes with installation by default
- Custom Roles can be created containing custom subset of Permissions
- Roles can be global or custom-scoped
- An end user gets assigned a Role and a Scope (in case of a scoped permission)
- Data Markings as an additional security layer on top



## Resource Manager

Predefined role

Resource-specific role. Users who are assigned to this role have full permission to manage selected resource, including the ability to grant permissions to other users.

## Permissions

Administer Resources  
Edit Resource Properties  
Edit Resources  
List All Users  
Manage Model Permissions  
Manage Owned Resource Access Right  
Read Resources  
Remove Resource

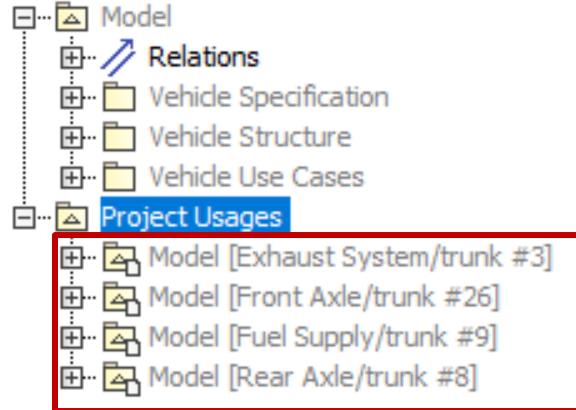
# MODEL VERSIONING

- When a user modifies a project, the user will then “Commit” the changes to Teamwork Cloud
- Each commit creates a new version of the project
- A version will be assigned a unique consecutive version number
- Significant model versions driving model lifecycle state changes can be marked by tags
- Optimistic and pessimistic locking modes

The image shows two overlapping windows from the Teamwork Cloud interface. The top window is the 'History Browser' titled 'History' with a sub-section 'History Browser'. It displays a message: 'In order to open a specific project version, select a node with a corresponding version number in the Version tree and click Open.' Below this is a table titled 'Water Distribution System V6 [trunk]' showing project versions, authors, dates, and comments. The table includes columns for 'Project Version', 'Author', 'Date', and 'Comment'. The 'Comment' column contains truncated text entries. The bottom window is a 'Commit Project to the Server' dialog box. It has a title bar with a checkmark icon and the text 'Commit Project to the Server'. Below the title is a sub-section 'Commit Project to the Server' with the sub-instruction 'Enter a comment and commit changes to the server. Additionally, you may add special tags for the project you are committing and review local changes. Click More to see additional tabs.' A central text area labeled 'Comment:' contains the text 'Fixed up connections in the power system module'. Below this is a checkbox labeled 'Keep locks' with a help icon. At the bottom of the dialog are buttons for 'Edit', 'Add', and 'Remove'. A tab bar at the bottom of the dialog shows 'Version Tags' (selected) and 'Local Changes'. A sub-section 'Add special tags for this version:' contains a list box with the word 'Major' selected. To the right of the list box are checkboxes for 'Create major version' and 'Create minor version'. At the very bottom of the dialog are buttons for 'Less', 'Commit', 'Cancel', and 'Help'. To the right of the dialog is a vertical toolbar with buttons for 'Set as Latest', 'Compare', and 'Properties'. At the bottom right are buttons for 'Open', 'Cancel', and 'Help'.

# MODEL COMPOSITION

- When modeling complex large systems, model composition plays a crucial role
- TWC project can be decomposed to consist of other projects (Used Projects)
- Libraries, Sub-systems, Requirements etc.
- Master project can reference (use) different versions and branches of Used Projects
- Utilities available for efficient Used Project configuration and version handling



# APIs and Services

# WHAT KIND OF APIs?

- Administration
- Repository Management
- Model Management
- Model Decomposition and DSL extensions
- Webhooks
- **SysML v2 standard API and Services**

## Model Management

<b>GET</b>	<code>/osmc/resources/{resourceId}/artifacts</code>	List artifacts in the latest revision of the master branch.
<b>POST</b>	<code>/osmc/resources/{resourceId}/artifacts</code>	Upload a blob into a particular resource.
<b>GET</b>	<code>/osmc/resources/{resourceId}/artifacts/{artifact}</code>	Get the artifact information in the latest revision of the master branch.
<b>GET</b>	<code>/osmc/resources/{resourceId}/branches/{branchId}/artifacts</code>	List artifacts in the latest revision of a particular branch.

## Repository Management

<b>GET</b>	<code>/osmc/resources</code>	List all projects.
<b>DELETE</b>	<code>/osmc/resources/{resourceId}</code>	Delete a project.
<b>GET</b>	<code>/osmc/resources/{resourceId}</code>	Get information about the resource (the project).
<b>PATCH</b>	<code>/osmc/resources/{resourceId}</code>	Change workspace or replace the metadata of the resource.
<b>GET</b>	<code>/osmc</code>	<b>Query Services</b>

## Query Services

<b>GET</b>	/api/projects/{projectId}/queries/{queryId}
<b>PUT</b>	/api/projects/{projectId}/queries/{queryId}
<b>DELETE</b>	/api/projects/{projectId}/queries/{queryId}
<b>POST</b>	/api/projects/{projectId}/query-results
<b>GET</b>	/api/projects/{projectId}/queries

# WEBHOOKS

1 Webhook 2 Resource scope

## Create new webhook

Webhook title \*  
Listen to HSUV model events

Webhook scope  
Model element

Enabled  
When enabled, upon creation of webhook it  
will be listening for new events

Webhook URL \*  
https://sendeventsphere.web...

Event  
Commit

Protect webhook URL with basic  
authentication

Allows you to add a username and password  
to secure your webhook URL

CANCEL

NEXT



Event Data



## Webhooks

Webhooks let you listen for model change events in real time and easily pass data to use in services and integrations

<input type="checkbox"/> Title ↑	URL	Sending event	Status	
<input type="checkbox"/> Listen to HSUV model changes	https://sendeventsphere.webhook	Commit	Enabled	

3 Element scope

## Select element scope

<input checked="" type="checkbox"/> Model
>  Relations
<input checked="" type="checkbox"/> HSUVModel
>  Relations
>  Explanations
>  HSUV Analysis
>  HSUV Behavior
>  HSUV Instance Values
>  HSUV Requirements
<input type="checkbox"/> Select recursively

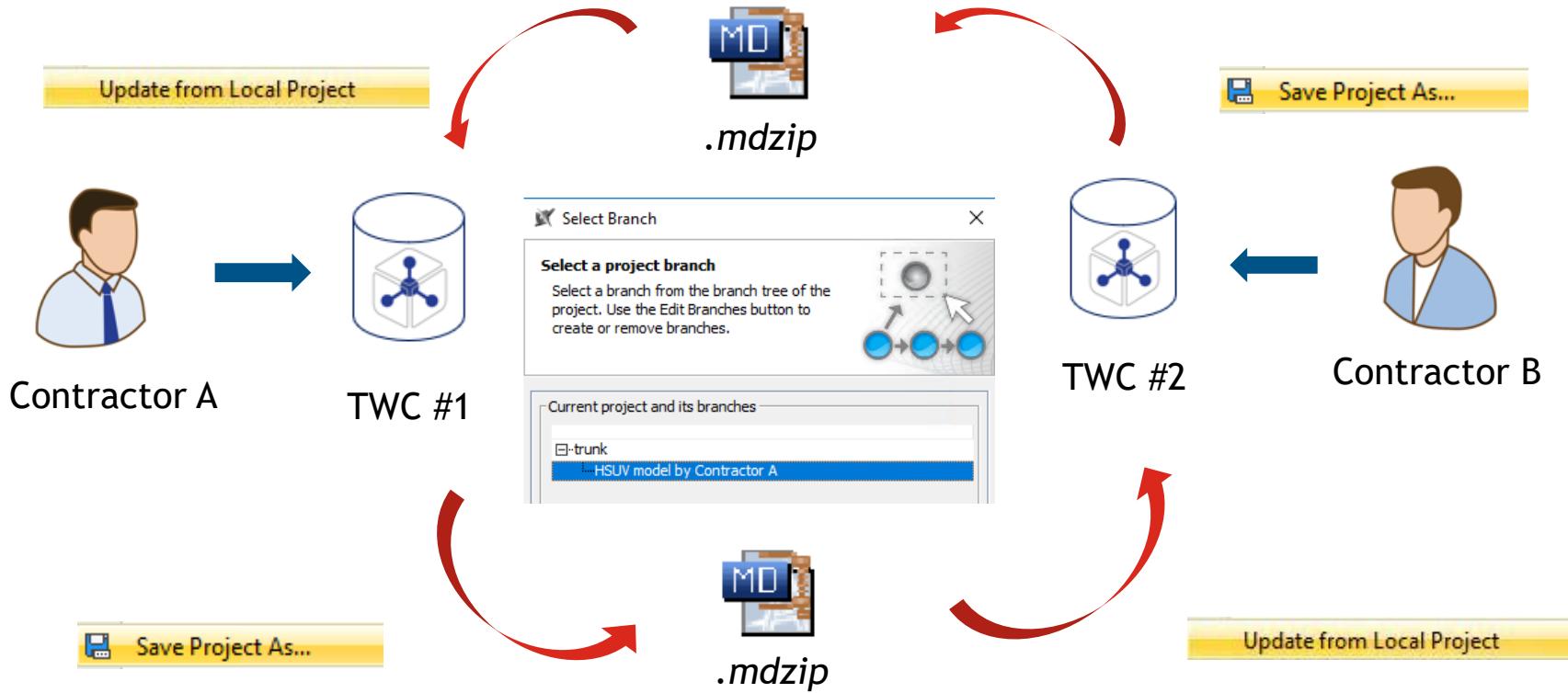
CI/CD Pipeline automation

CANCEL

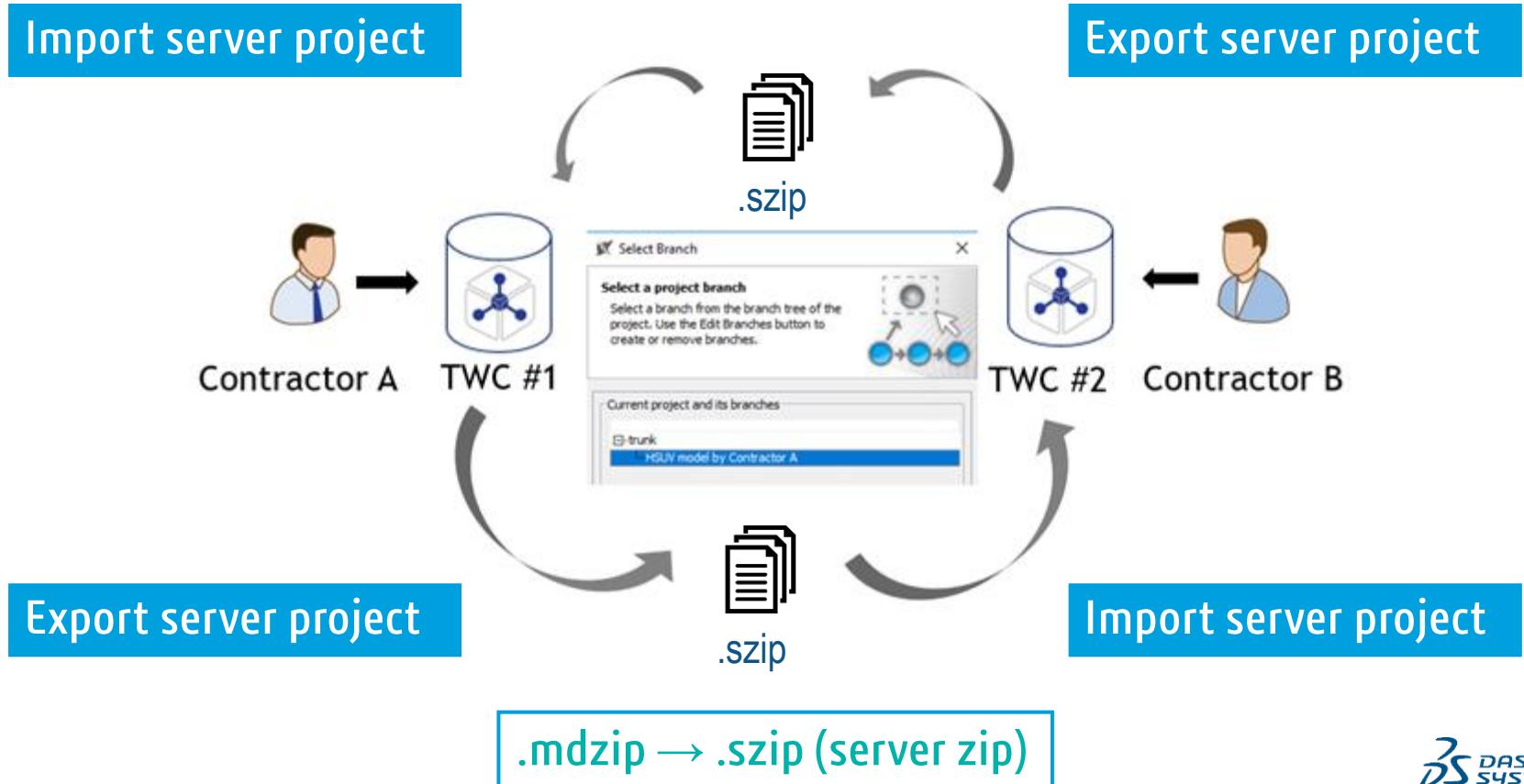
CREATE

# MODEL EXCHANGE

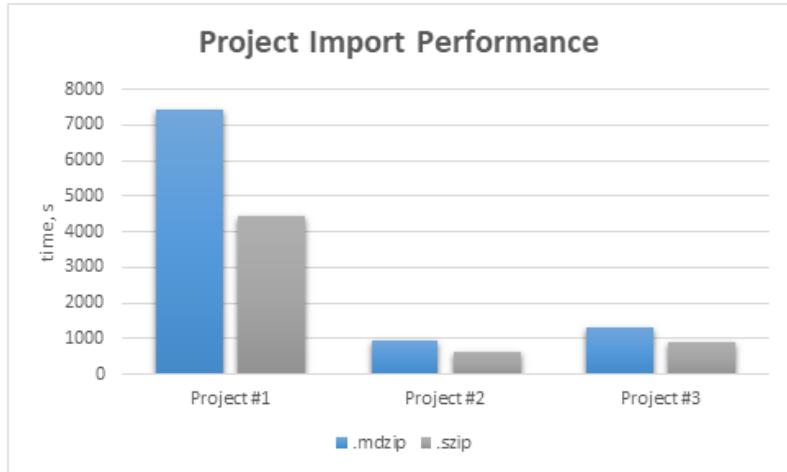
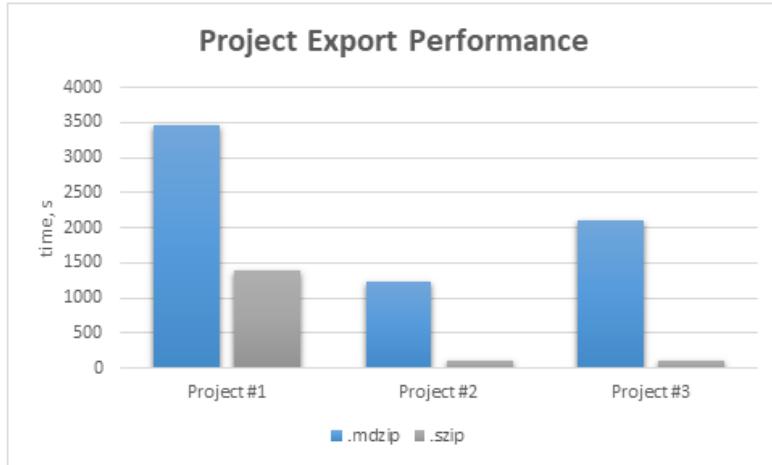
# CLASSICAL MODEL EXCHANGE VIA .MDZIP



# REWORKED DISCONNECTED TEAM COLLABORATION



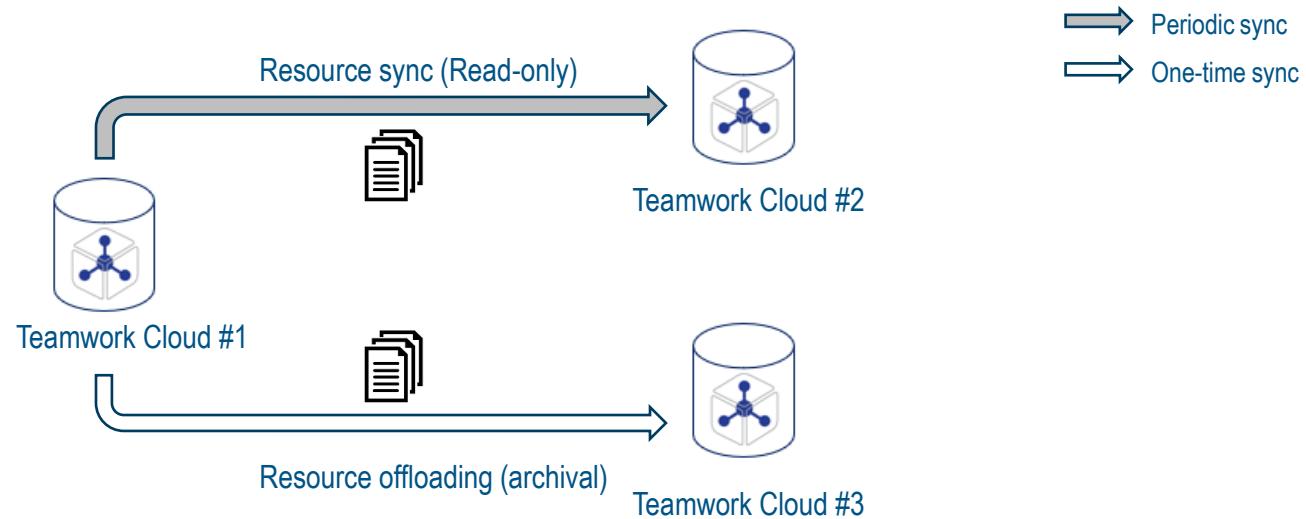
# .MDZIP VS .SZIP PERFORMANCE



Project #1 (1M elements, 57 Used Projects), Project #2 (2M elements, 7 Used Projects), Project #3 (2M+ elements, 3 Used Projects)

# CROSS-CLUSTER SYNC

# CROSS-CLUSTER RESOURCE SYNC



Synchronize/move resources across multi-site TWC deployments

# CROSS-CLUSTER RESOURCE SYNC (II)

## Remote repository management

Synchronize resources from another instance of Teamwork Cloud

Category/Resource from Source ↑	Target category	Source address	Synchronization mode	Status	Synchronized	Actions
<input type="checkbox"/> Vehicle Electrical System	Supplier A Models	twc-a-supplier	On every new commit	Partially synchronized	Sep 24, 2021, 11:23:06 AM	
<input type="checkbox"/> Climate Control System	From UK Site	uk-twc-site	majorVersion	Synchronized	Sep 20, 2021, 2:33:09 PM	
<input type="checkbox"/> Temperature Regulation Loop	Supplier B Models	twc-b-supplier	Manual	Synchronized	Sep 24, 2021, 10:11:05 AM	

1 Login to source 2 Select resources 3 Select target 4 Mode

**Log in to the source server**

Log in to the Teamwork Cloud server (source server) from which you want to synchronize resources and/or categories. Use the source server credentials to log in.

Source Teamwork Cloud address \*

remote-uk-site-twc

Username \*

Syncer

Password \*

•••••

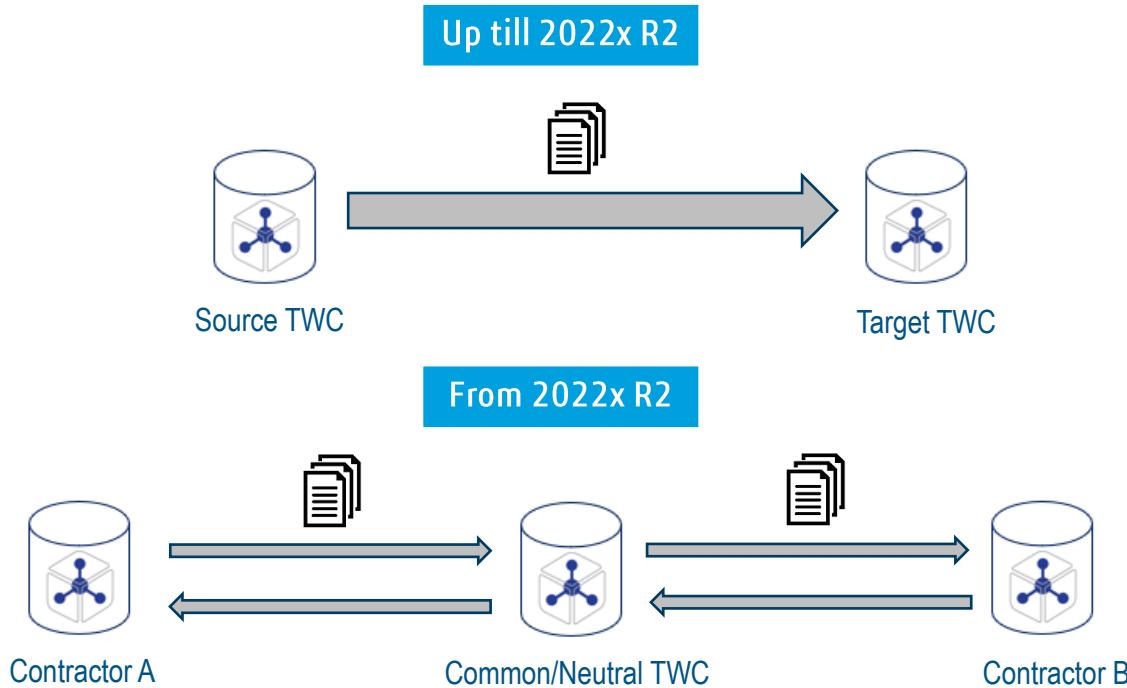
CANCEL NEXT



Source TWC

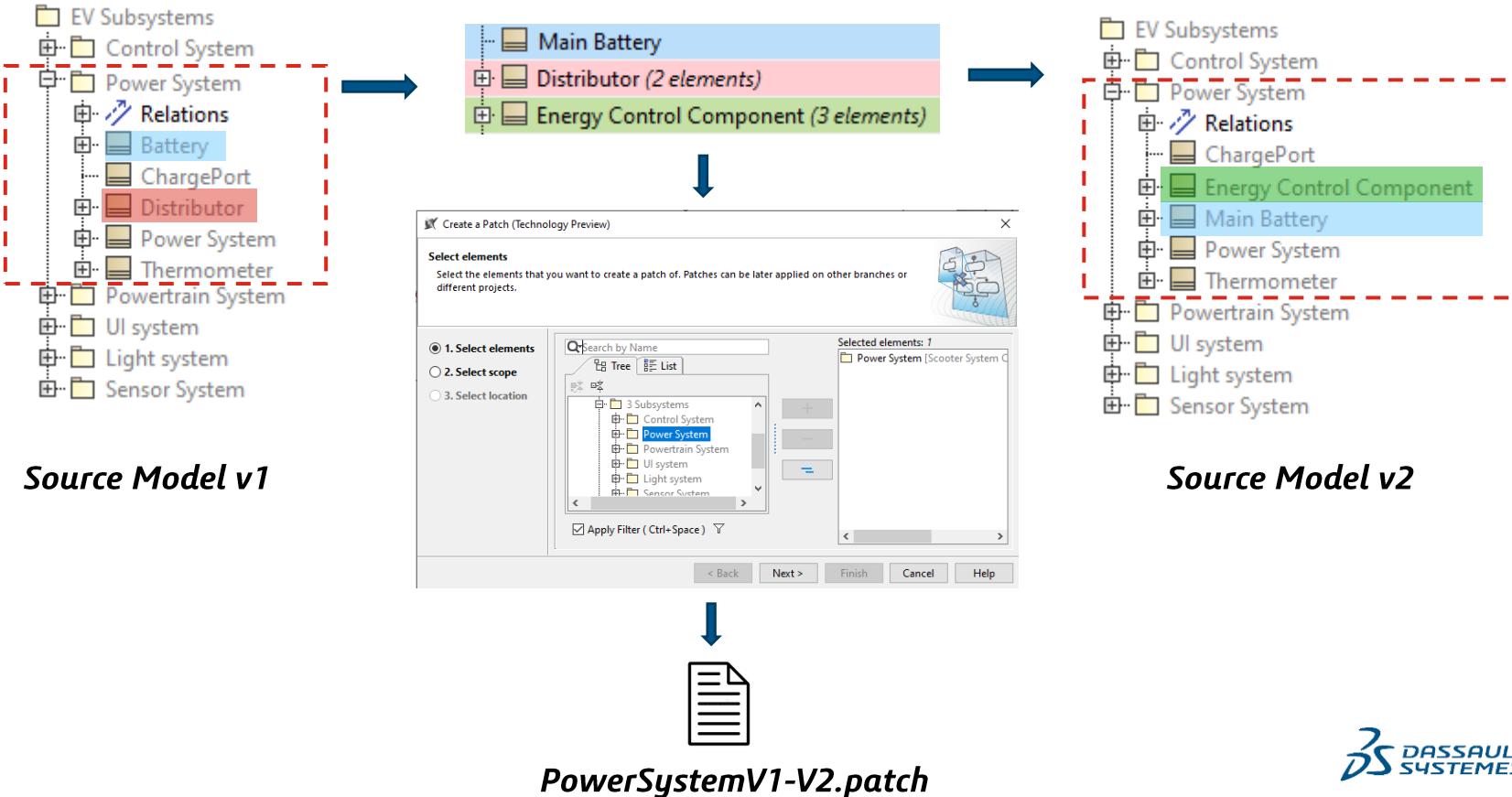
Target TWC

# CROSS-CLUSTER RESOURCE SYNC TOPOLOGIES

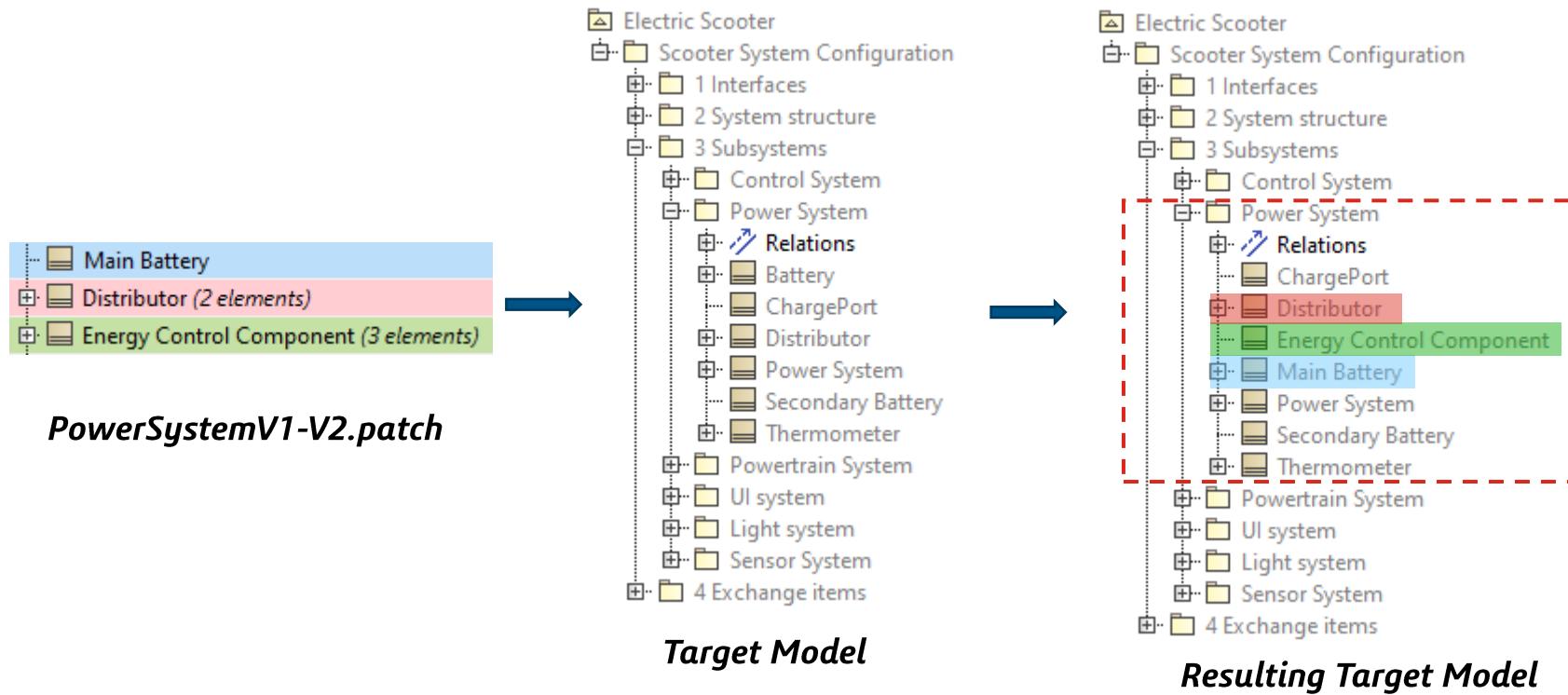


# MODEL PATCHING

# PORTABLE MODEL DIFF & PATCH (EXPORT)

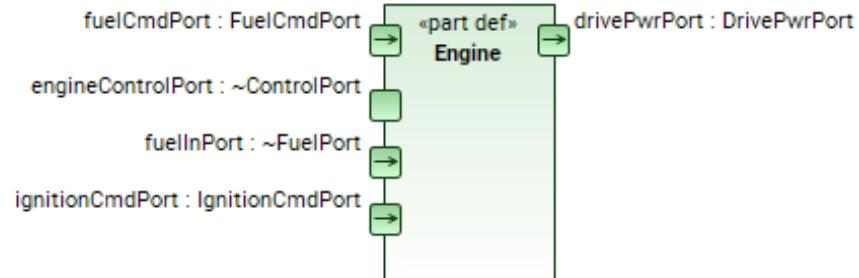
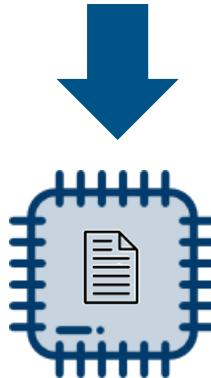
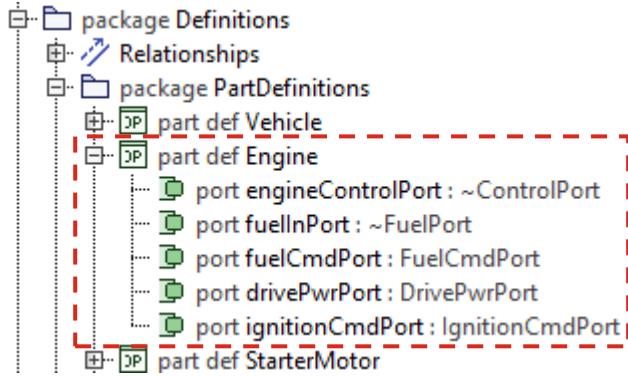


# PORTABLE MODEL DIFF & PATCH (APPLICATION)



# MODEL PATCHING (BB)

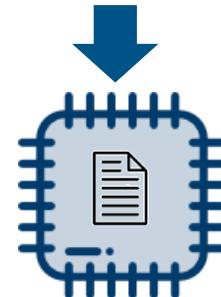
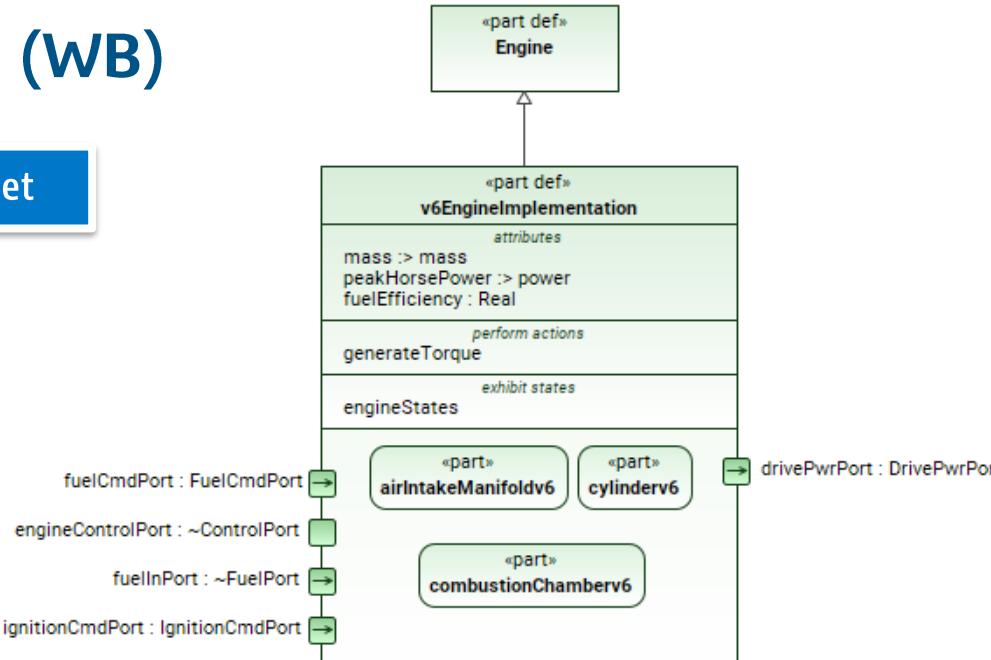
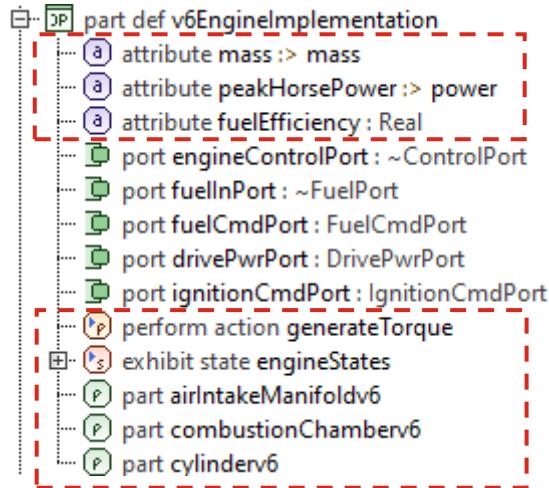
## A Serializable Model Change Set



EngineBlackBox.patch

# MODEL PATCHING (WB)

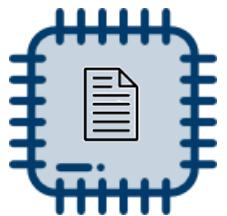
## A Serializable Model Change Set



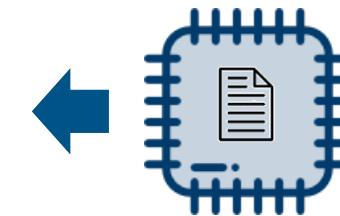
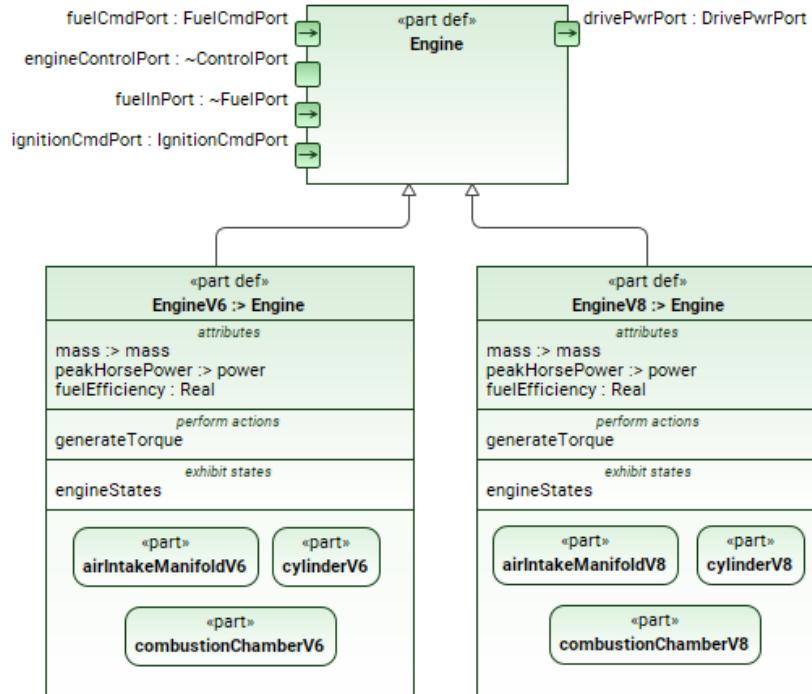
EnginemImpl1.patch

# MODEL PATCHING (MULTIPLE WBs)

## A Serializable Model Change Set

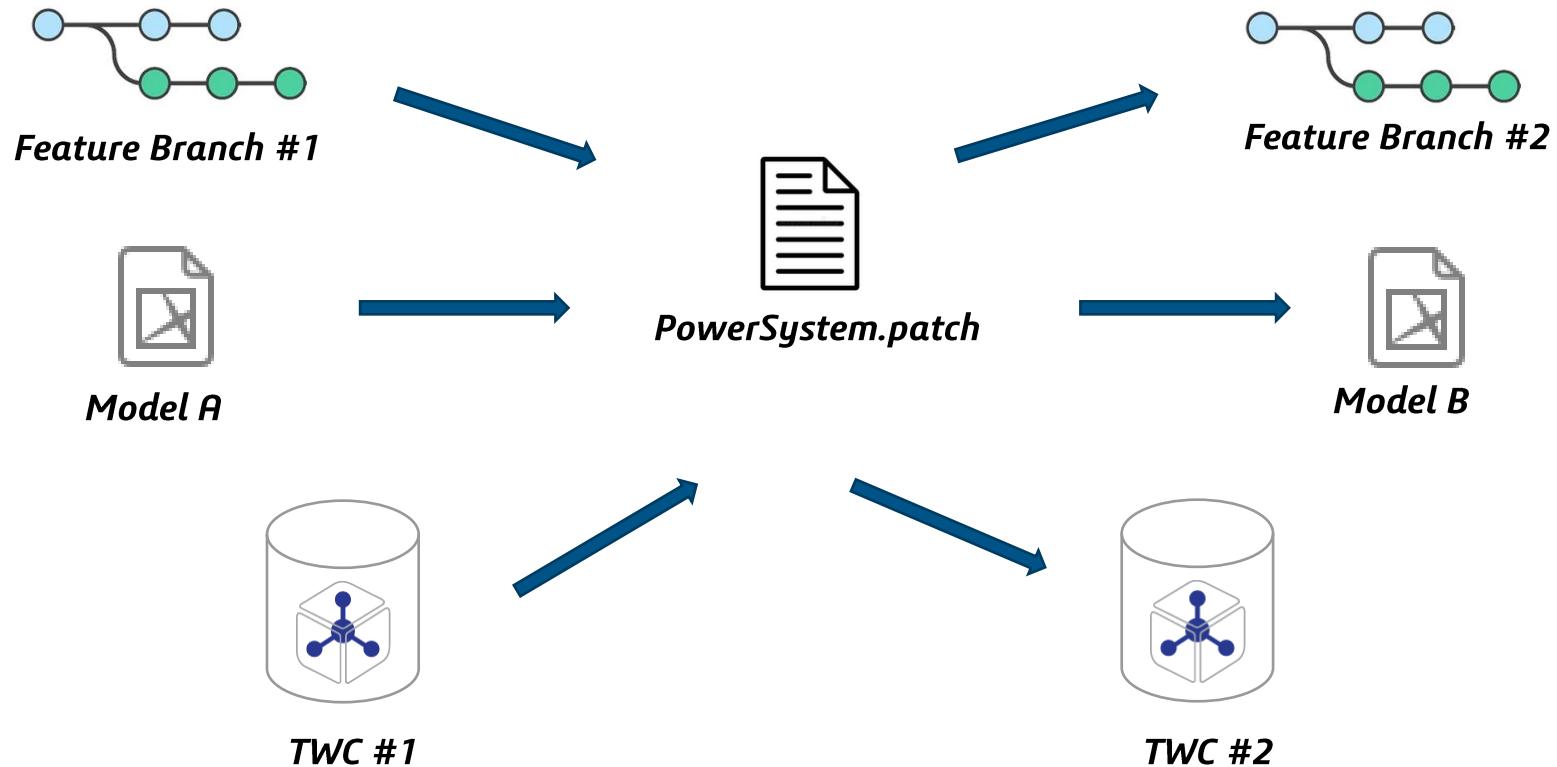


Enginelmpl1.patch



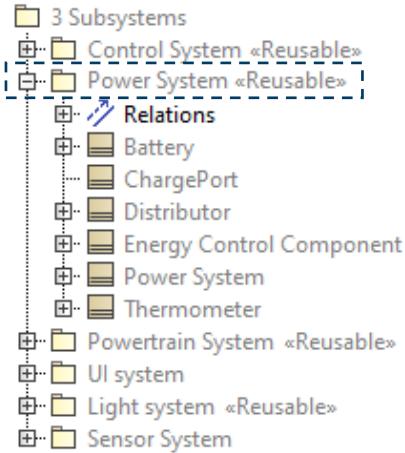
Enginelmpl2.patch

# PORTABLE MODEL DIFF & PATCH

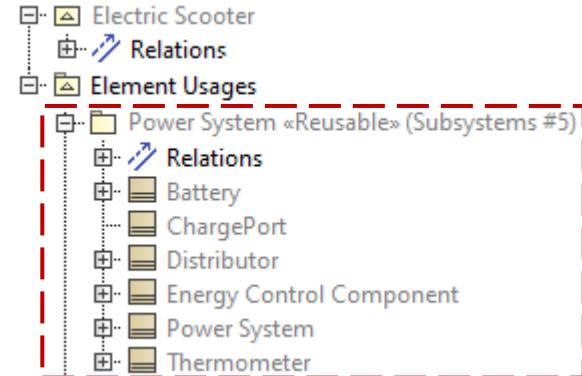
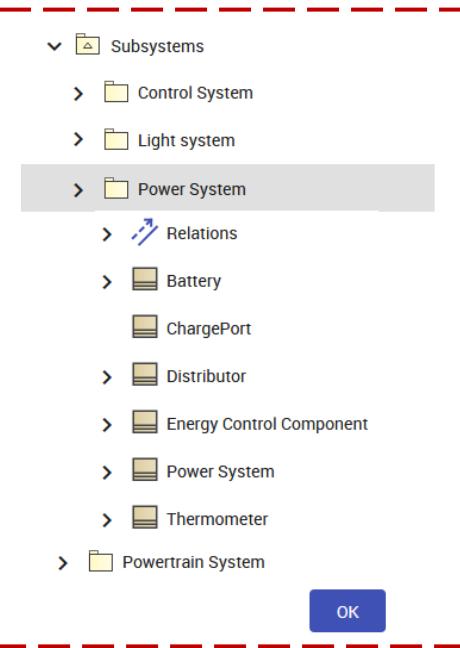


# USED PROJECTS 2.0

# LIGHTWEIGHT ELEMENT REUSE (TECH. PREVIEW)



1. Light model tree browser



1. Select elements to reuse
2. Import the scoped selection only

# LIGHTWEIGHT ELEMENT REUSE (TECH. PREVIEW)

The image shows three overlapping dialog boxes for selecting a project and its version, with a detailed component tree on the right.

**Left Dialog (1. Select project):** A tree view of projects. The 'Used Projects' section is expanded, showing 'UsedP2.0' selected. Other items include 'TWC training', '2021x Project', and 'Remote Project from Su...'. A 'Project description:' text area is present.

**Middle Dialog (2. Select version):** A table of project versions for 'UsedP2.0 [trunk]'. The table has columns: Project Version, Author, and Date. The data is as follows:

Project Version	Author	Date
32	TVS2	Wednesday, Ju...
30	TVS2	Wednesday, Ju...
29	TVS2	Tuesday, Ju...
28	TVS2	Tuesday, Ju...
27	TVS2	Tuesday, Ju...
26	TVS2	Tuesday, Ju...
25	TVS2	Tuesday, Ju...
24	TVS2	Tuesday, Ju...
23	TVS2	Tuesday, Ju...
22	TVS2	Tuesday, Ju...
19	TVS2	Wednesday, Ju...
17	TVS2	Wednesday, Ju...
1	TVS2	Wednesday, Ju...

**Right Dialog (3. Select scope):** A tree view of components under 'UsedP2.0'. The 'Power System' node is selected. Other components include 'Subsystems', 'Relations', 'Battery', 'ChargePort', 'Distributor', 'Energy Control Component', 'Thermometer', and 'Powertrain System'.

