



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

Configuration Management Challenges in Multi-Team Collaboration Using Linked Models

David Hetherington, Mark Petrotta, Tomas Vileiniškis



Hello.



David Hetherington

Principal
System Strategy, Inc



Mark Petrotta

Principal
System Strategy, Inc



Tomas Vileiniškis

Product Manager, Teamwork Cloud
Dassault Systèmes

Today's Agenda



David Hetherington

The Business and
Organizational Challenges



Mark Petrotta

“Real-Life Graph Theory”
System Strategy Inc
Experiences with Large,
Multi-Organization, Model
Exchanges



Tomas Vileiniškis

How the Tool Suppliers
Can Help

The Business and Organizational Challenges

*“The UUIDs seemed
like a really good
idea at the time...”*

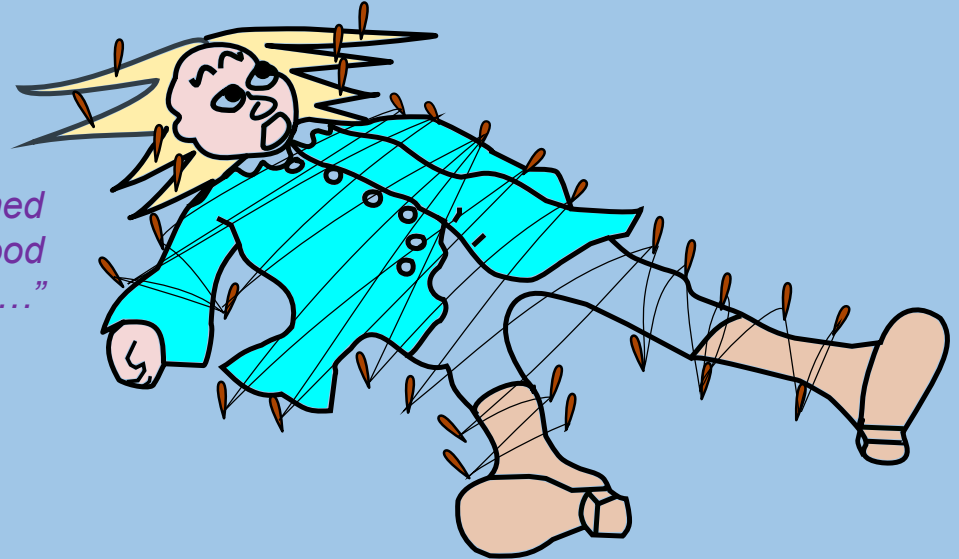
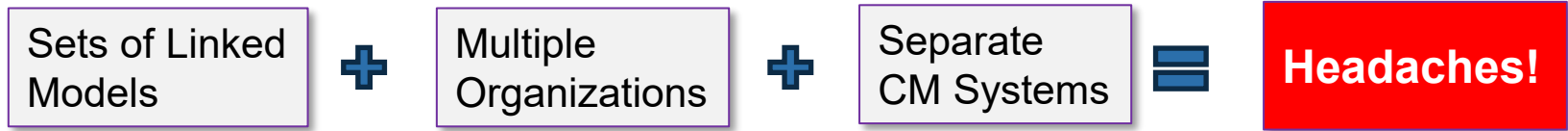


Image Copyright © 2022 Asatte Press, Inc

Summary



If You Have Ever Struggled with a Missing Linked Spreadsheet....

The screenshot shows a Microsoft Excel spreadsheet with the following data:

Product	Price	Utility
ABC Inc	\$25.00	0.7
Good Stuff Ltd	\$27.95	0.71
Shady Deals	\$18.95	0.65
We Copy Anything, Inc	\$12.00	0.2

A warning dialog box is displayed over the spreadsheet. The dialog box has a yellow warning icon and the following text:

Microsoft Excel

This workbook contains links to one or more external sources that could be unsafe.
If you trust the links, update them to get the latest data. Otherwise, you can keep working with the data you have.

The dialog box has three buttons: Update, Don't Update, and Help.

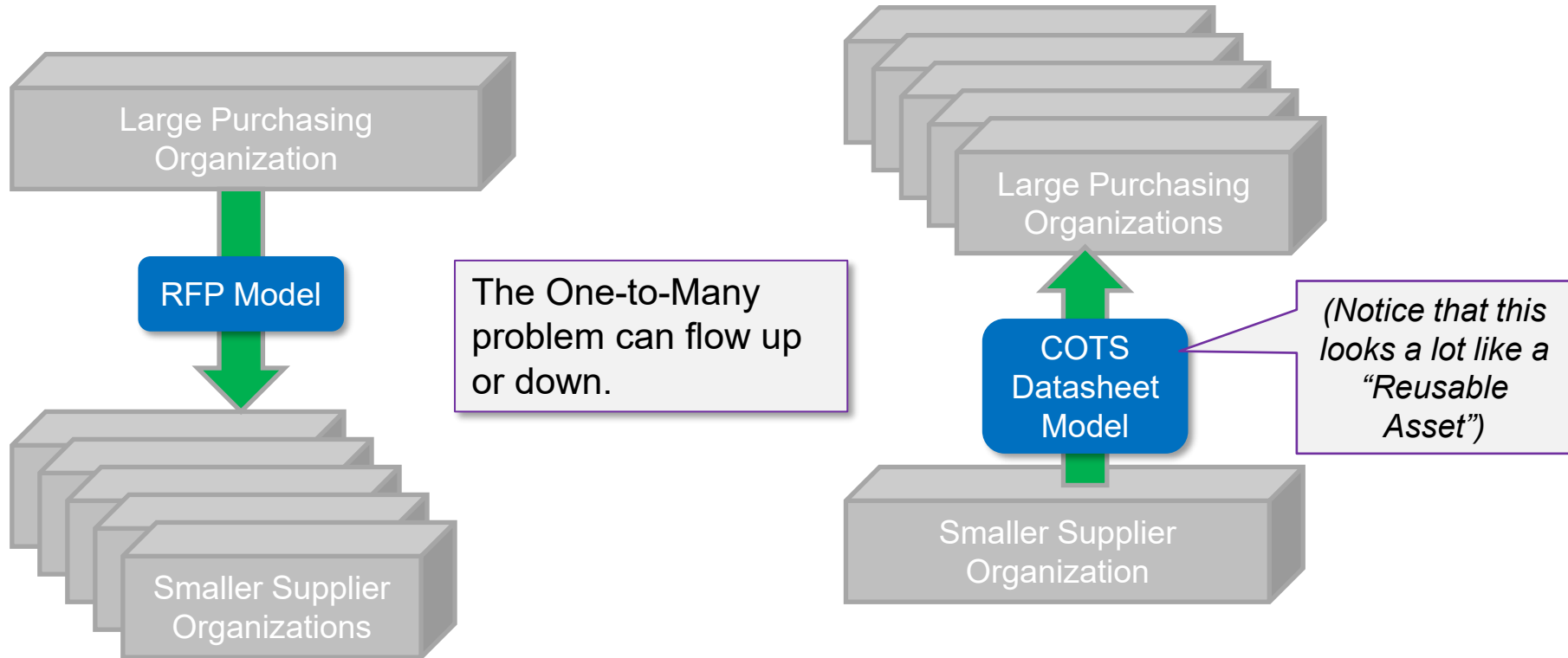
Insight of Daniel Hettema – 2025-03-20

Comment to the OMG SysML V2 Summit:

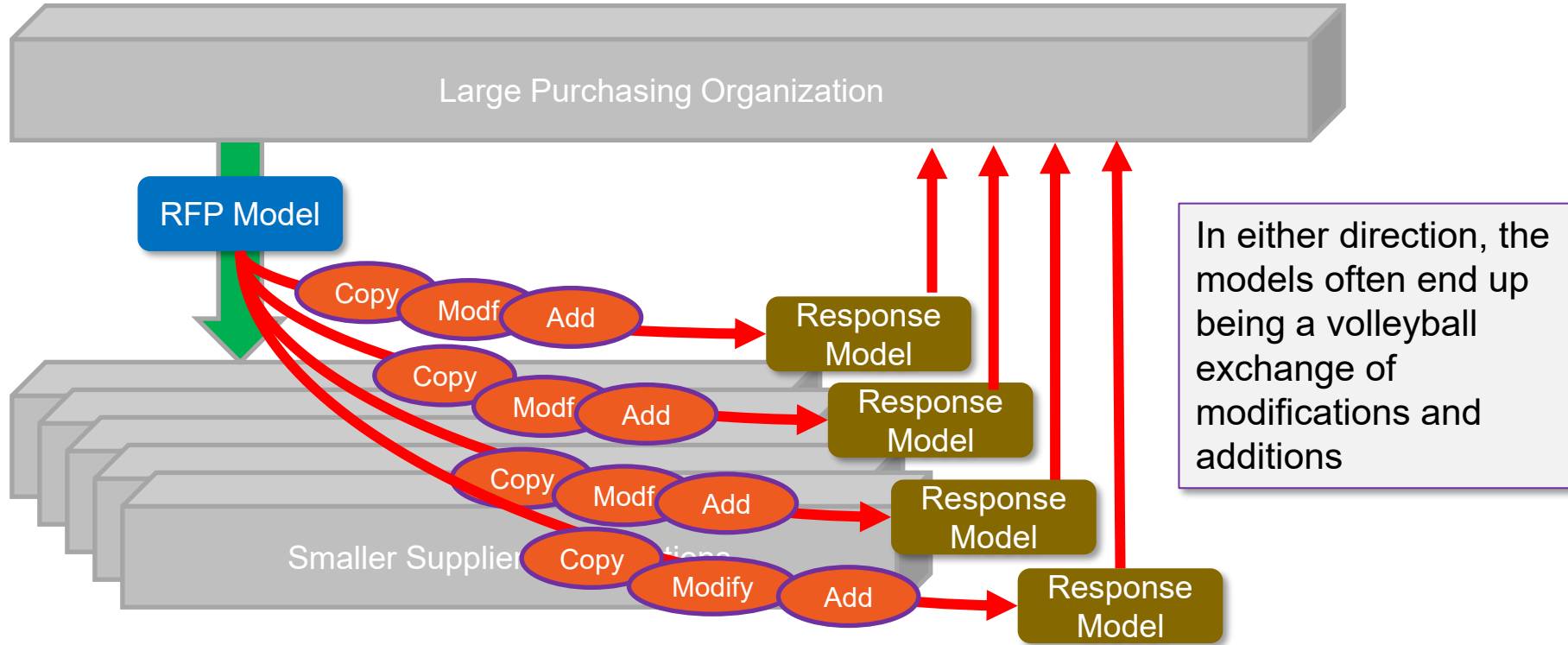
"Of course, we all dream of a fully integrated digital thread. The scale of the Department of Defense is simply too big for that. We are going to be shipping models around for the foreseeable future"



The One-to-Many Problem



The Response Problem



And Just for Fun

Each organization has one or more configuration management systems.

Often the CM systems are not linked even within organizations.

The CM systems are rarely linked between systems.

Models are not single, one-time deliveries. Rather there are major versions, minor versions, hot fixes, etc...

The updates need to move smoothly between disconnected CM systems.

The models are not single models, but rather clusters of linked models.

In the volleyball exchange, different teams will add links to other models.

Often, multiple teams will add links to the same other models, unaware of each other's links.

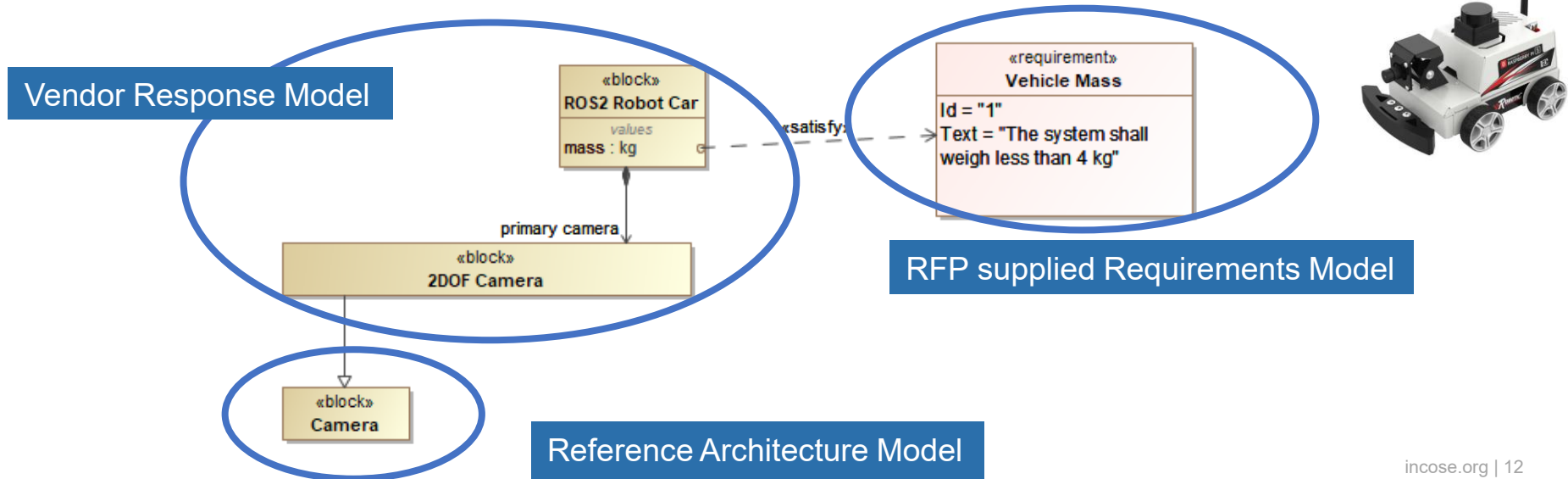
“Real-Life Graph Theory”

System Strategy Inc

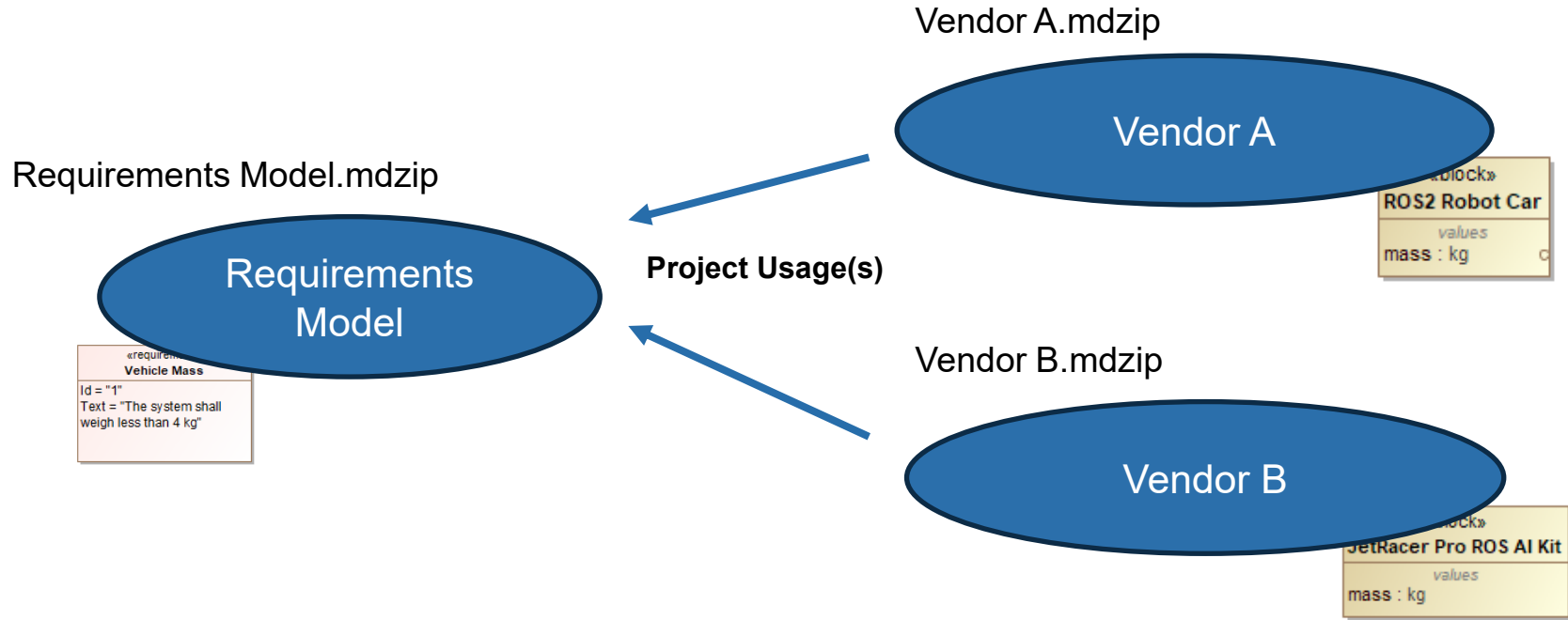
Experiences with Large, Multi-Organization, Model Exchanges

Models as Units of Configuration

MBSE has us thinking about definition, usage, and traceability without regard to *model units* or *configuration management*



Thinking about Units of Configuration



Models as units of configuration ==
Define once, Use many times at model level

Configuration Management of Libraries

Previous examples are “sneaker-net”

Problems:

- No specification of how to get / distribute mdzip files

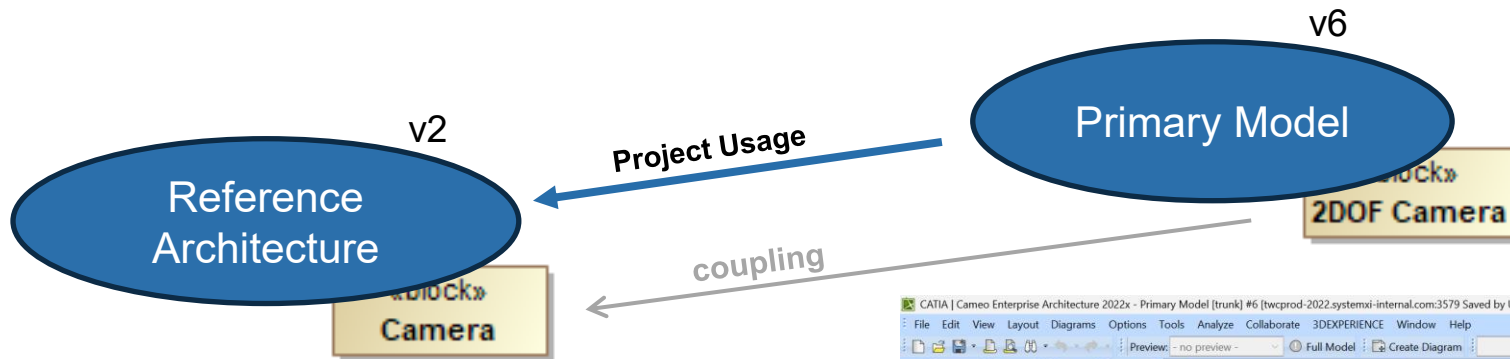
- No version control of libraries

- No change management processes

→ Configuration management of libraries

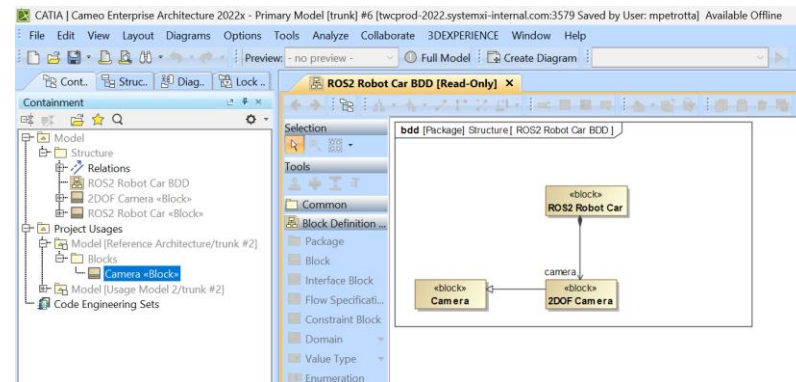
Basic Configuration Managed Library

Model Network
[PM@6 -> RA@2]

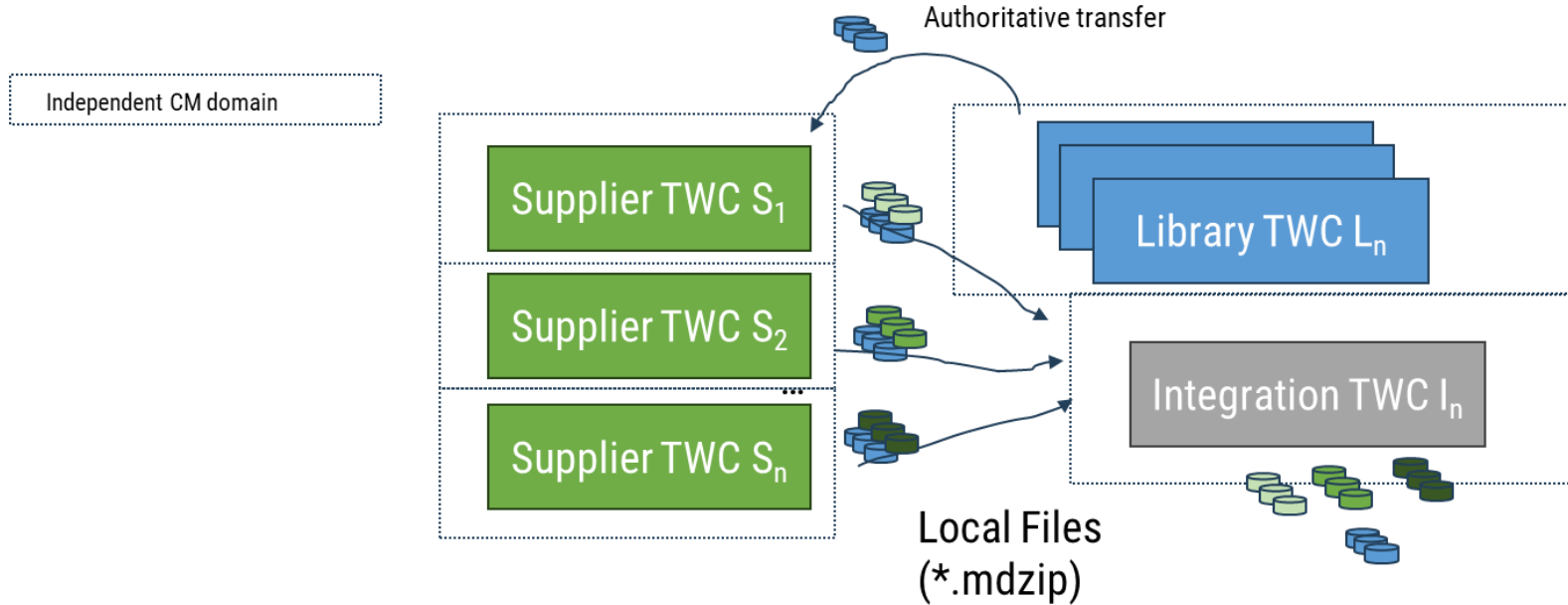


What happens if 'Reference Architecture' deletes "Camera"?

What happens if 'Reference Architecture' adds new elements?



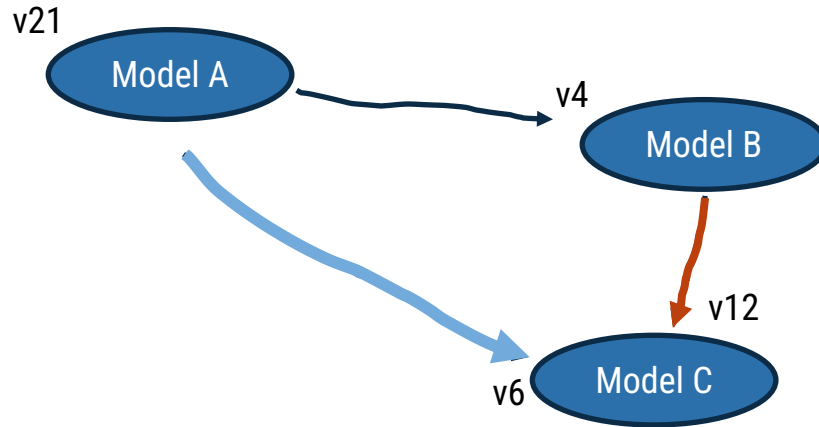
Challenge: Distributed CM at Scale



- *How are changes to L_x propagated?*
- *How are multiple configurations of S_x and L_x reconciled in I_x ?*

Example of CM Issues: Diamond Problem

Once loaded, A will only use a single version of C



Model Network =

$(M_x @ \text{Version}(M_x)) \text{ uses } M_y @ \text{Version}(M_y)$

MN = [A@21 -> B@4],
[A@21 -> C@6],
[B@4 -> C@12]

TMN += [A@21 -> B@4 -> C@12]

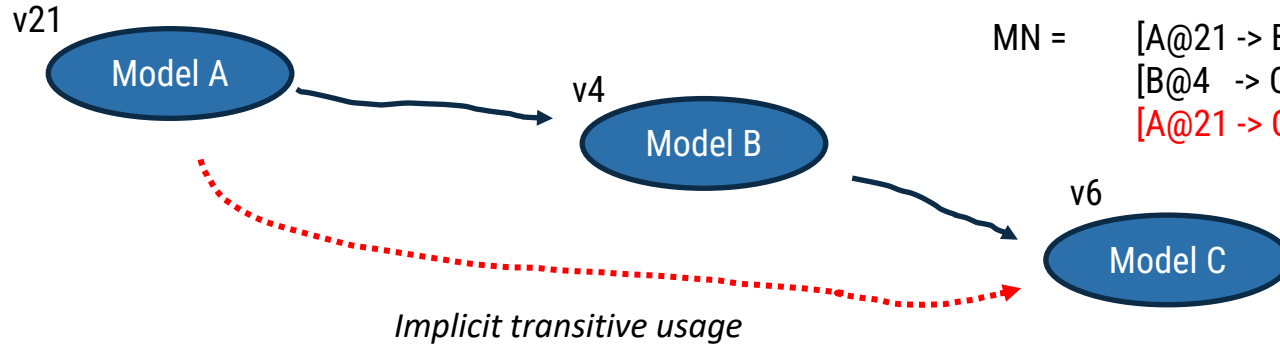
Element Network Error:
B@4 may refer to newer
content in C@12

Which?

- *v6 – principle of shortest path (a->c beats a->b->c)*

Example of CM Issues: Transitivity

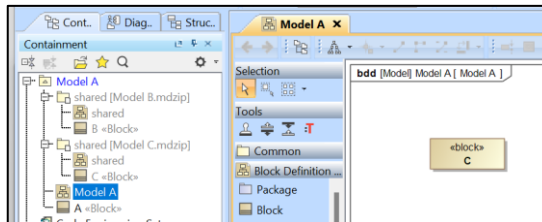
If Model A mounts B, which mounts C:



Model Network =
 $(M_x @ \text{Version}(M_x) \text{ uses } M_y @ \text{Version}(M_y))$

MN = [A@21 -> B@4],
 [B@4 -> C@6],
 [A@21 -> C@6] "Automatic" / Implicit Usage

Tools make it easy to create a transitive usage of C – “automatic usage”

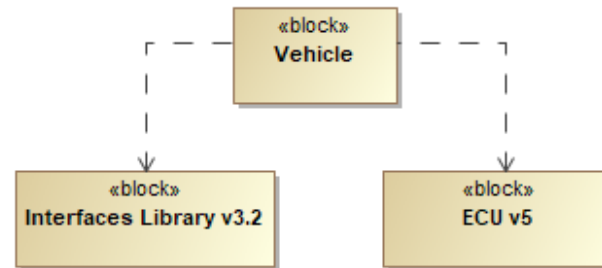
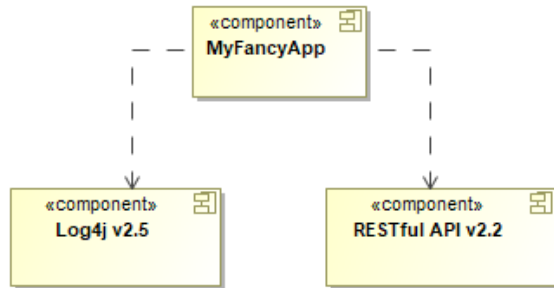


When the number of models ~ >50,
 unintended semi-formalized
 dependencies become cumbersome

How the Tool Suppliers Can Help

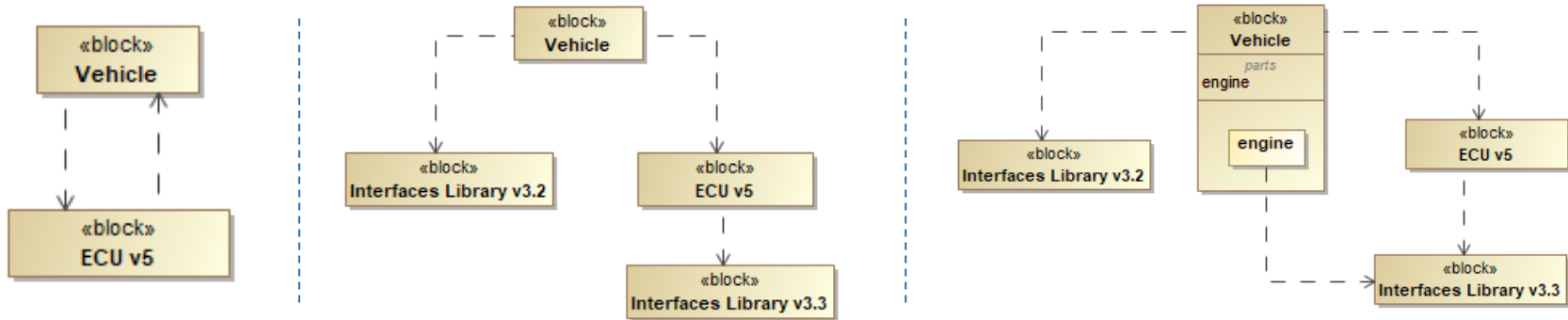
Model Dependency Management

- Linking of software modules and components in software engineering is done on daily basis
- Dedicated tools for module dependency management (e.g. Maven)
- Cyclic Usages, Transitivity, Version Conflicts
- Software component = MBSE model



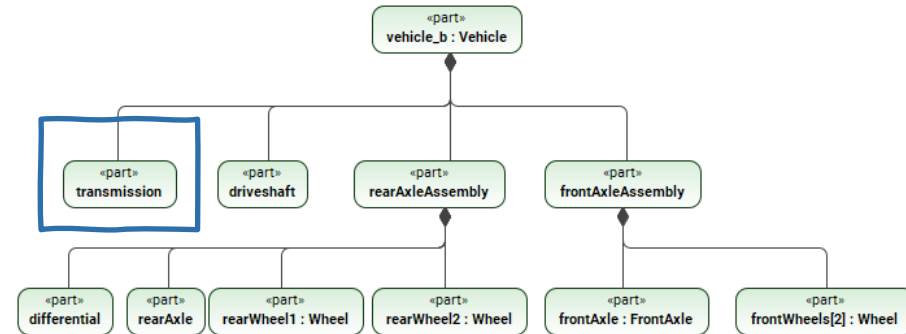
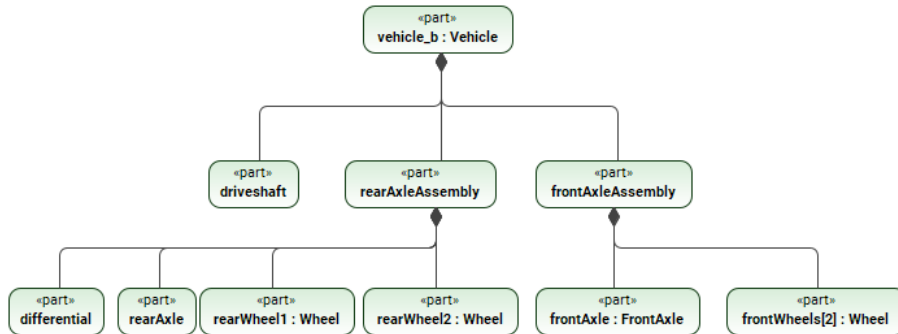
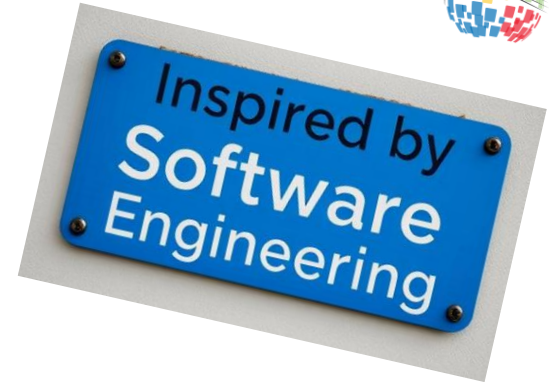
Model Dependency Management

- Do not allow cyclic usages
- Clear conflicting version resolution rules
- Direct element-level references between modules indicate a direct dependency relationship



Model Patching

- Well established principle of making and delivering small code changes in a large code base (e.g. provide a bug fix)
- Could be very well applied to MBSE as well:
 - No need to move huge models back and forth to deliver only small model changes
 - E.g. one element is missing in one single diagram in a spider web of 50 linked models

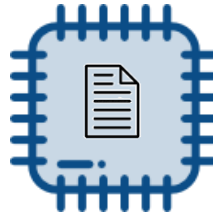
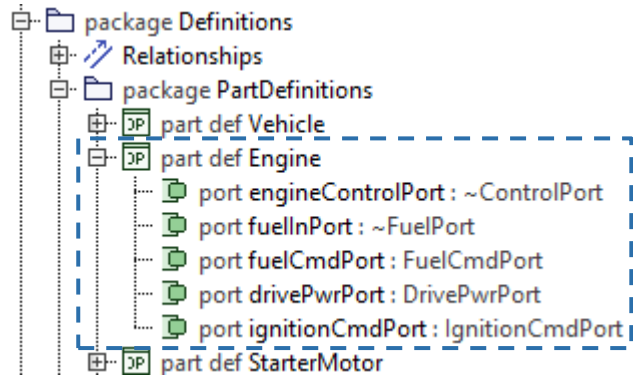


Model Patching

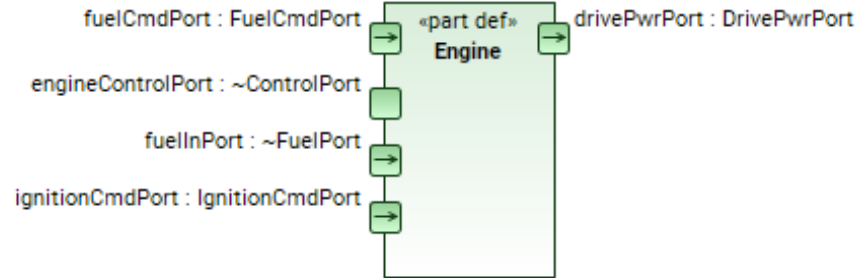
- Moving partial model changes around in poorly decomposed models
- Blackbox-whitebox abstraction collaborative modeling principle
 - Contractor/Supplier receives a blackbox-level model
 - Works on whitebox-level solution model and puts that into a patch
 - Patch gets sent back and applied onto the source model
 - No need to move large models back and forth!



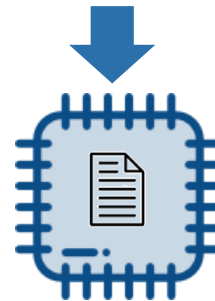
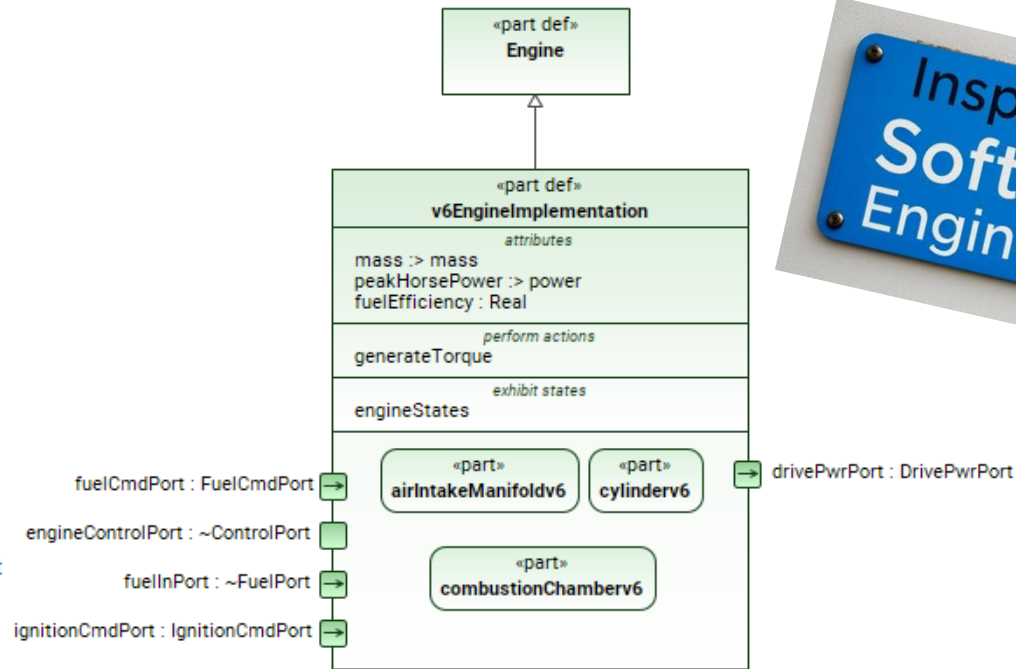
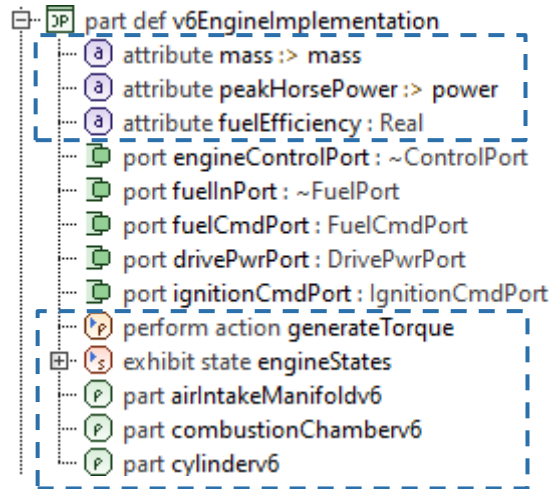
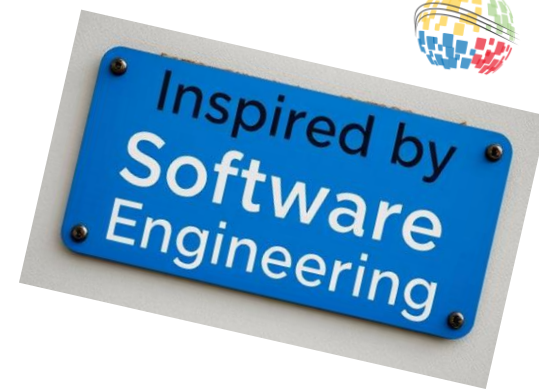
Model Patching



EngineBlackBox.patc
h



Model Patching



EngineImpl1.patc
h

Questions?

David Hetherington

dhetherington@systemxi.com

david_hetherington@ieee.org

Mark Petrotta

mpetrotta@systemxi.com

Tomas Vileiniškis

Tomas.VILEINISKIS@3ds.com

The Dance of Digital Engineering

A symphony of models, vast and wide,
Where countless minds and hands collide.
From vendors, teams, and firms they stream,
To build, refine, pursue a dream.
They share, improve, reshape, revise,
Breaking limits, growing wise.
A living web of code and lines,
Evolving fast in real-time.
A tapestry both vast and bright,
Threads of logic, pure insight.
In digital realms, they craft, they mold—
Engineering's future, bold and bold.



35th Annual **INCOSE** international symposium

hybrid event

Ottawa, Canada
July 26 - 31, 2025