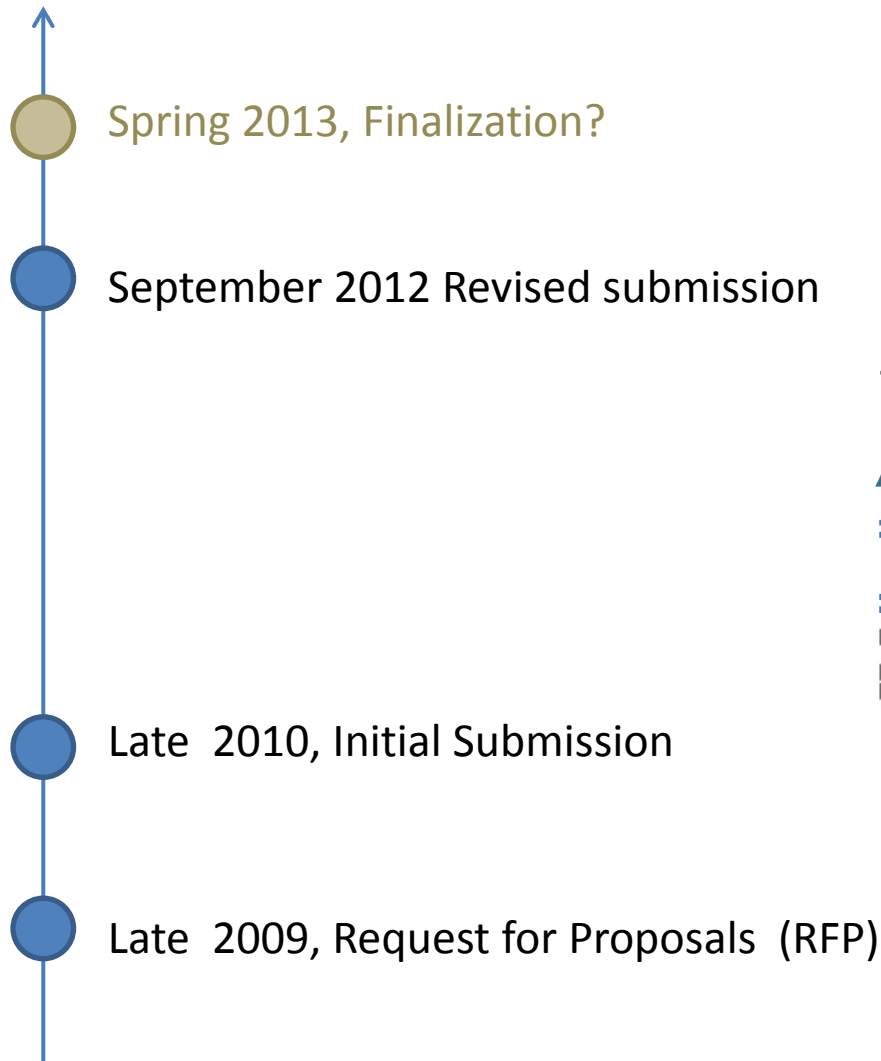


CVL Mini-Tutorial

Øystein Haugen
(SINTEF)

Andrzej Wąsowski
(IT University of Copenhagen)

CVL in OMG



JOINT SUBMISSION TEAM initiators from the MoSiS project



tool vendors



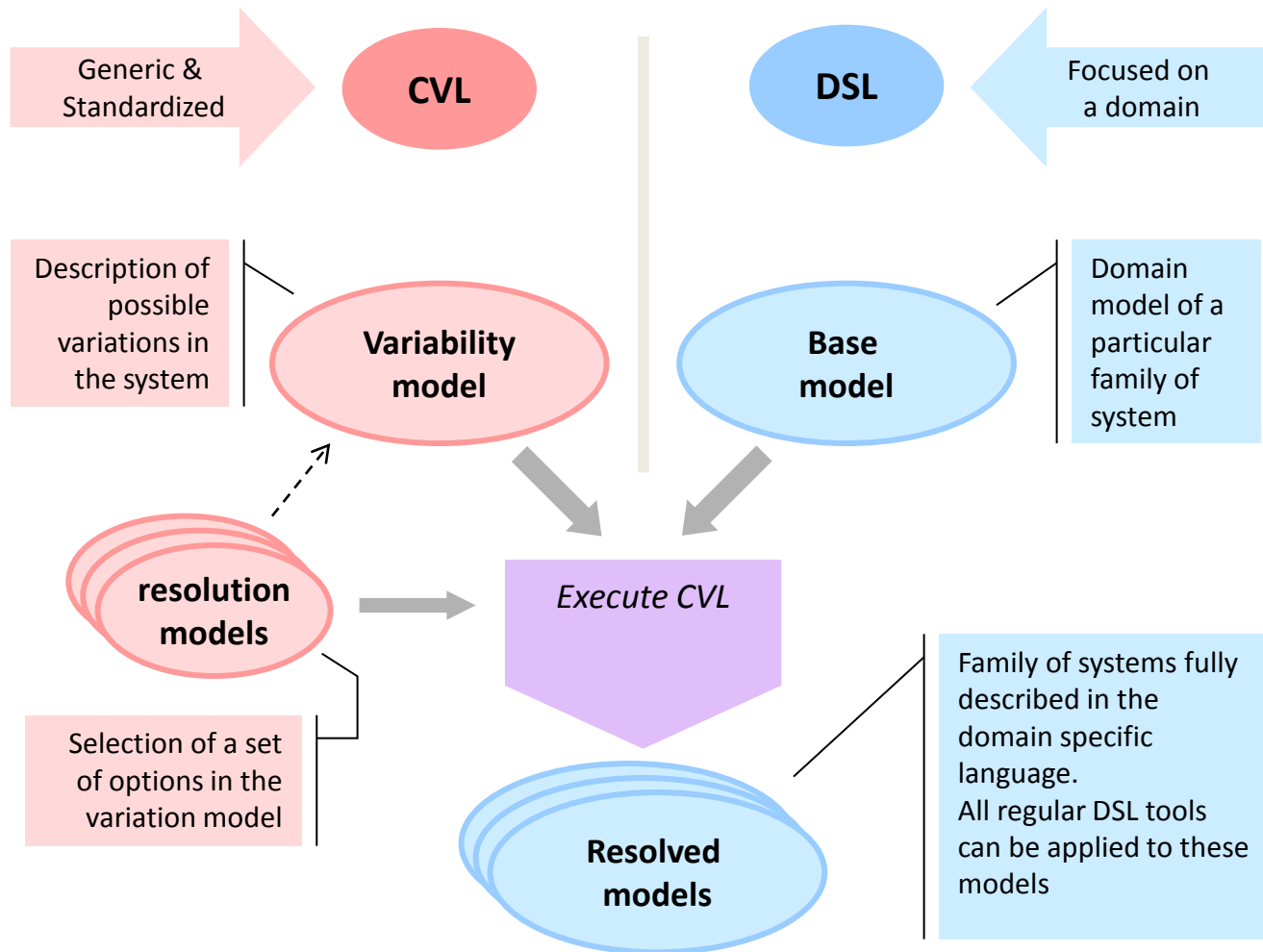
users & consultants



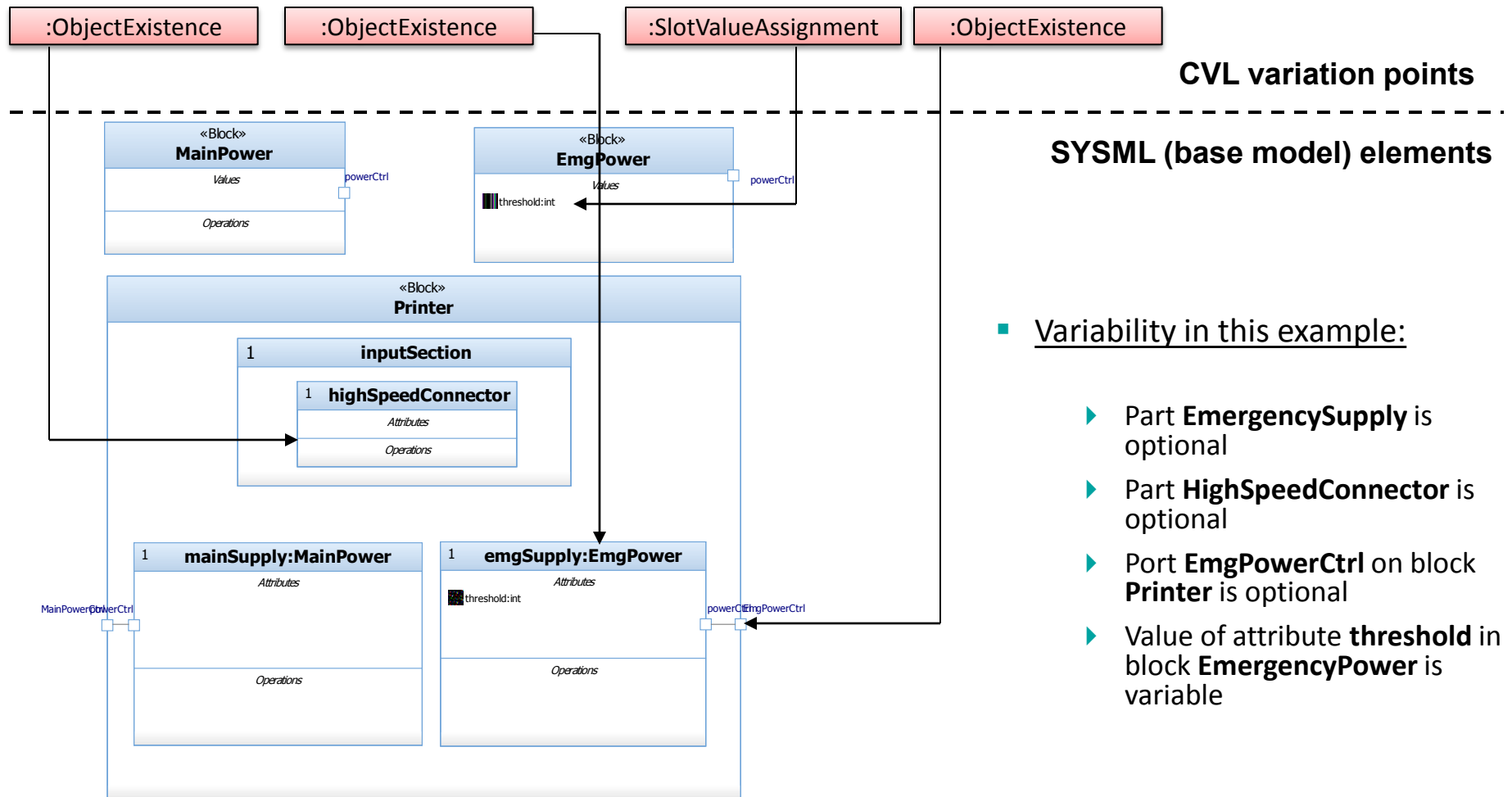
research institutes and universities



Common Variability Language (CVL)

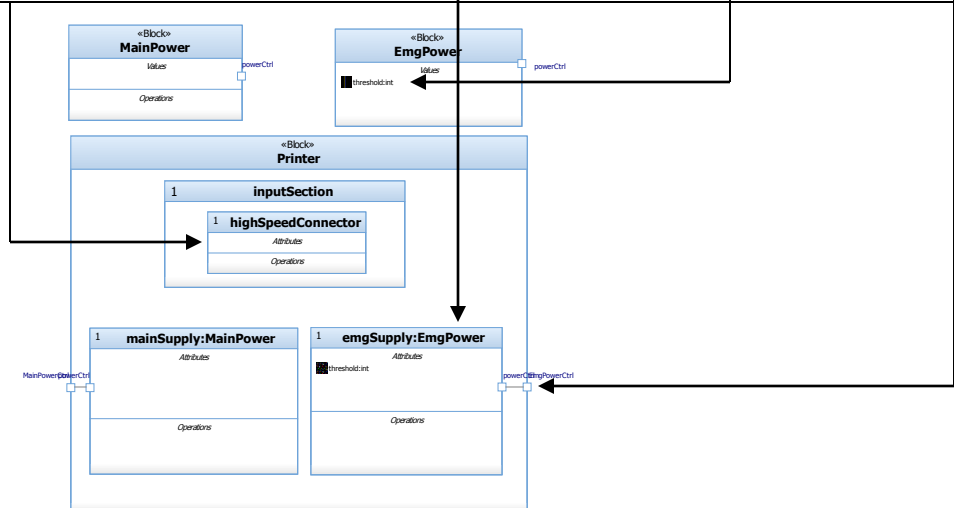
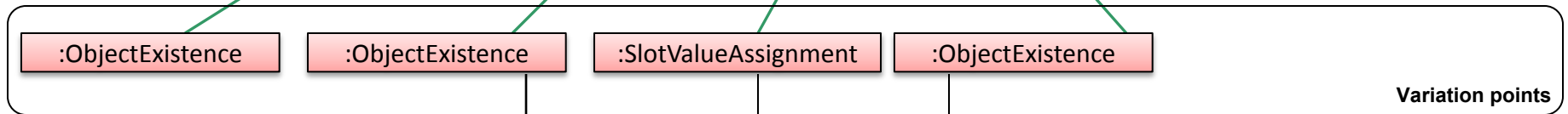
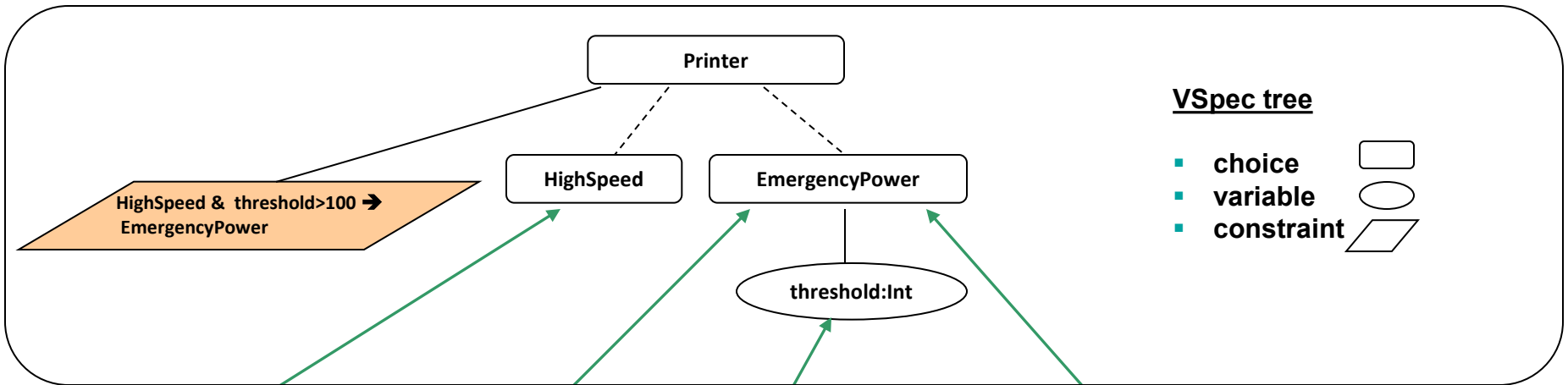


Variation Points over base model

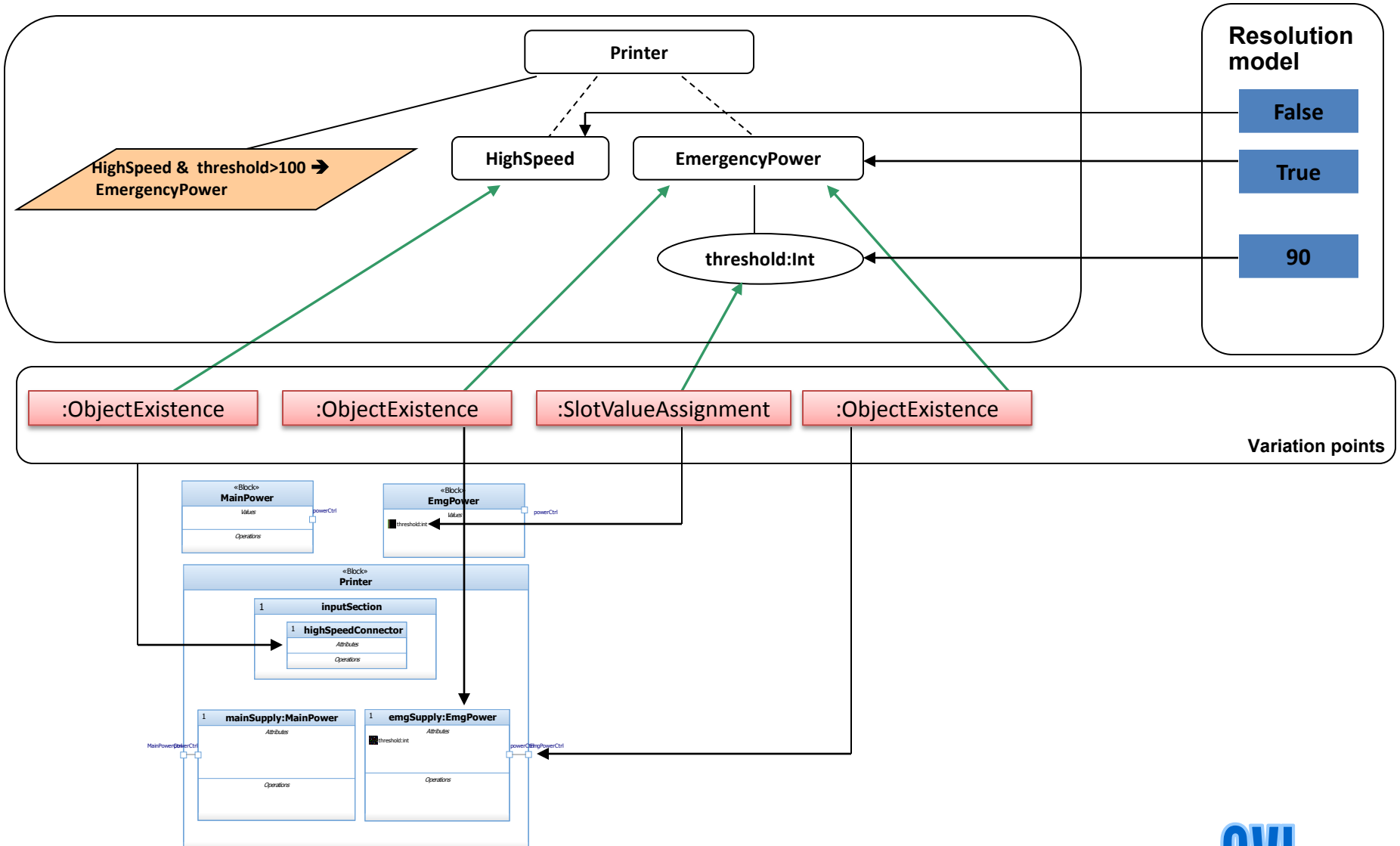


- Variability in this example:
 - ▶ Part **EmergencySupply** is optional
 - ▶ Part **HighSpeedConnector** is optional
 - ▶ Port **EmgPowerCtrl** on block **Printer** is optional
 - ▶ Value of attribute **threshold** in block **EmergencyPower** is variable

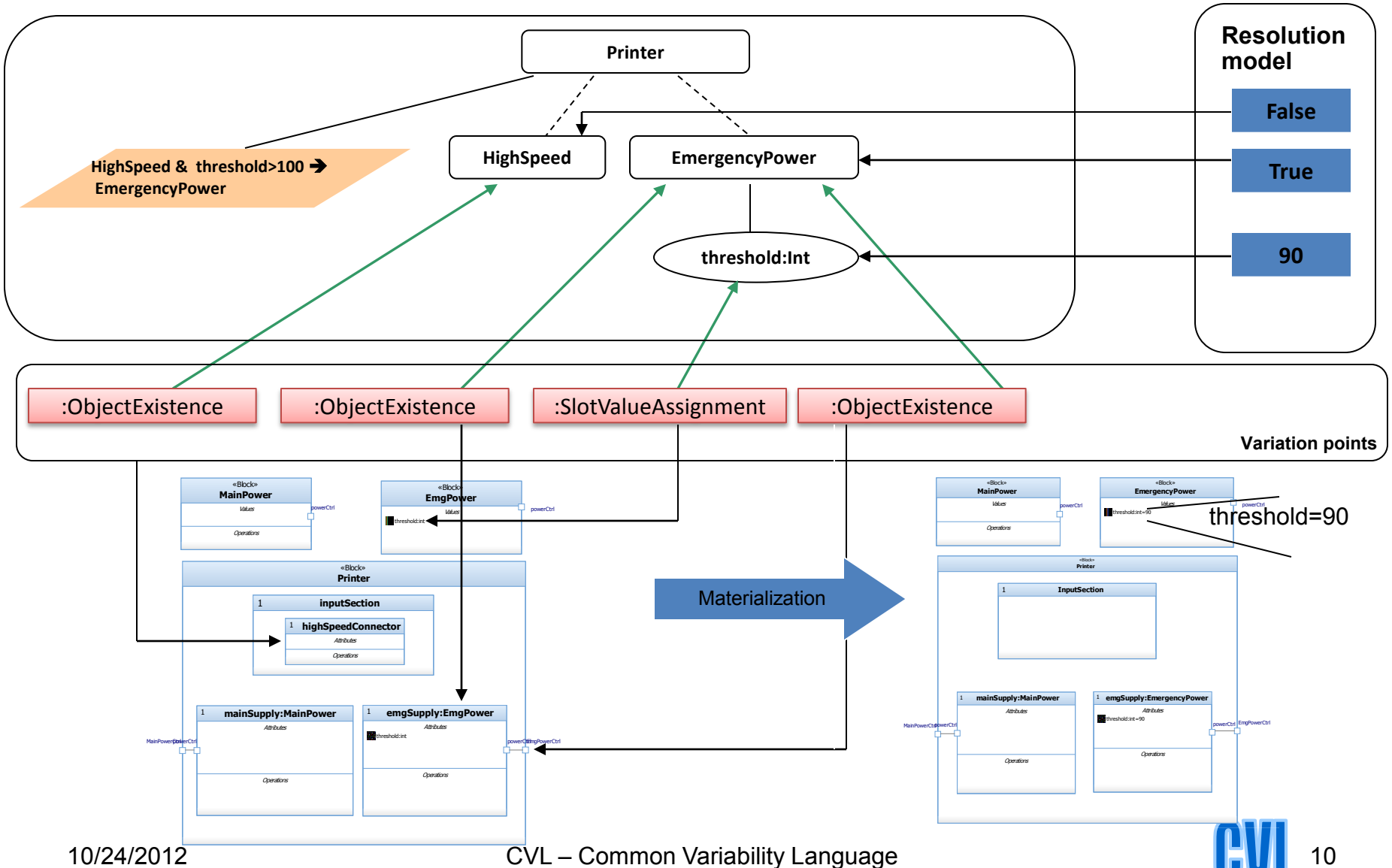
VSpec trees and binding



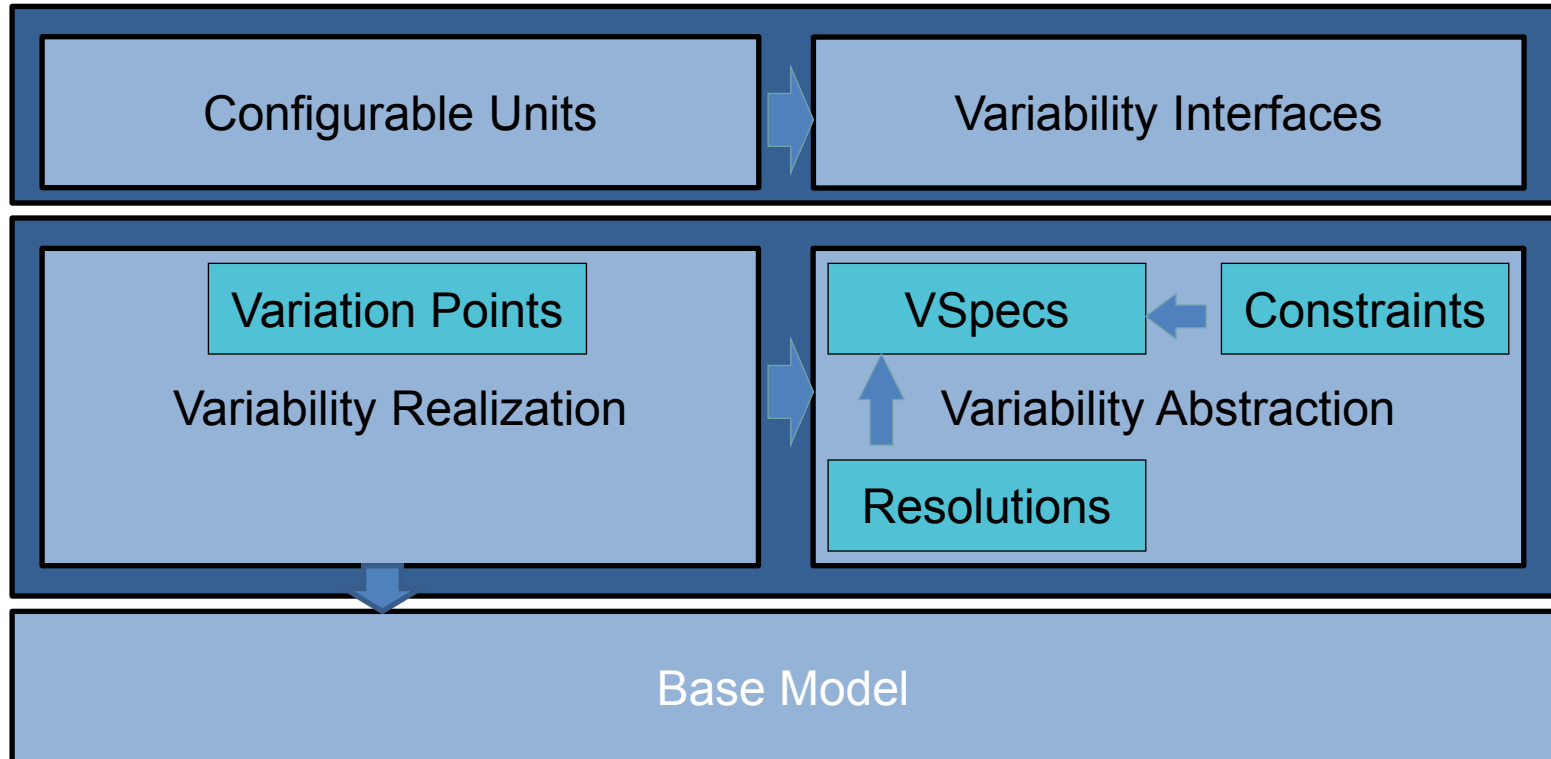
Resolution



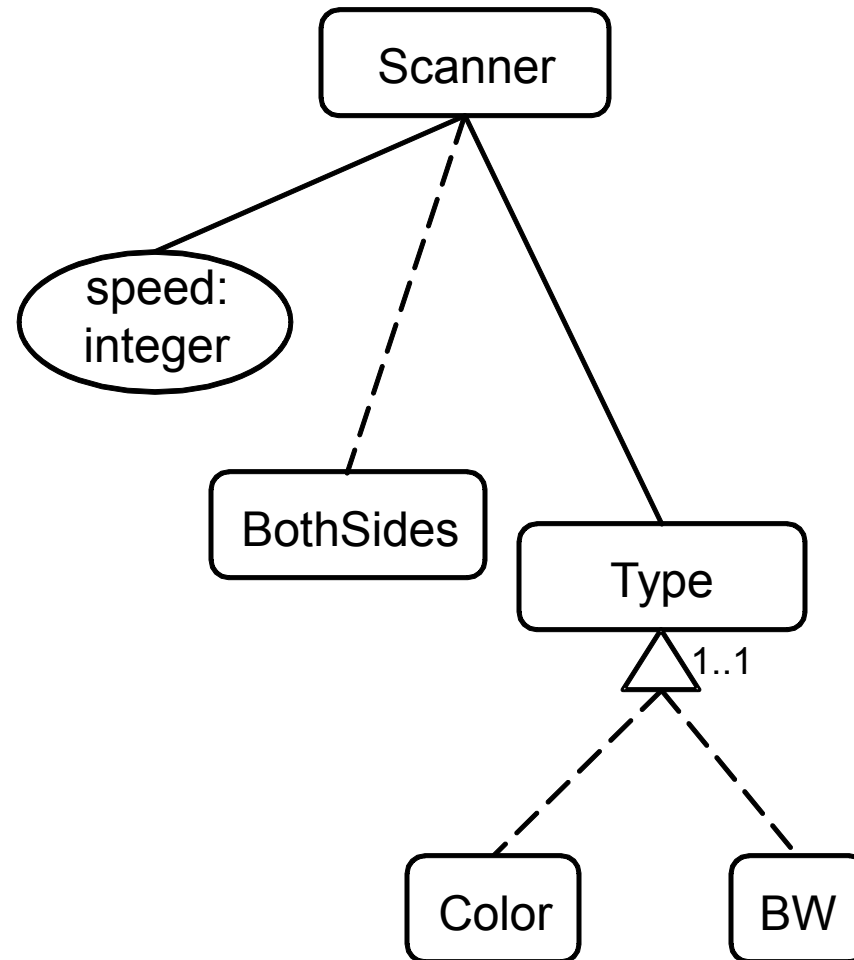
Materialization



CVL Architecture

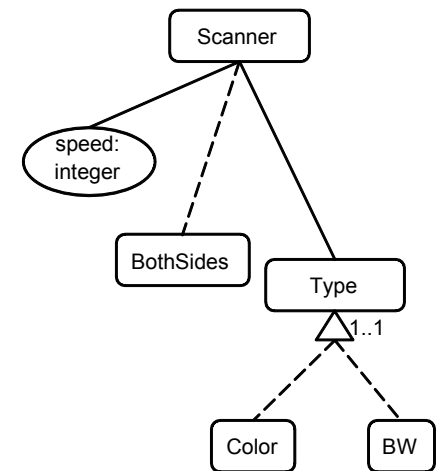


Variability Specification (Vspec)



Variability Specification (Vspec)

- Abstract variability specifiers
- Similar to features in feature modeling
- Essentially **decisions** like in decision modeling
- Example: “GPS” is a feature of a camera, but it is also a choice, which can be decided yes/no
- VSpecs can be used for feature modeling
- Variation points are bound to VSpecs, giving semantics to VSpecs.
- VSpecs are arranged in trees,
- Parent-child relations organize the resolution space in usual way, like in FMs.

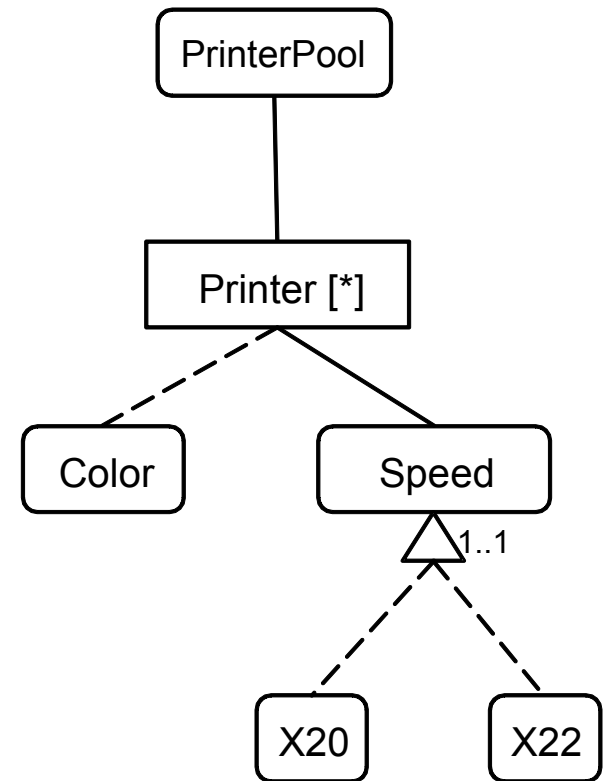


VSpec Types (continued)

- **Choice** requires yes/no decision
- **Variable** is a VSpec allows for providing a value of a specified type
- **VClassifier** allows for creating instances and then providing per-instance resolutions for the VSpecs in its sub-tree
- **Composite VSpecs** – used for modularity; explained later

Tree Structure Semantics

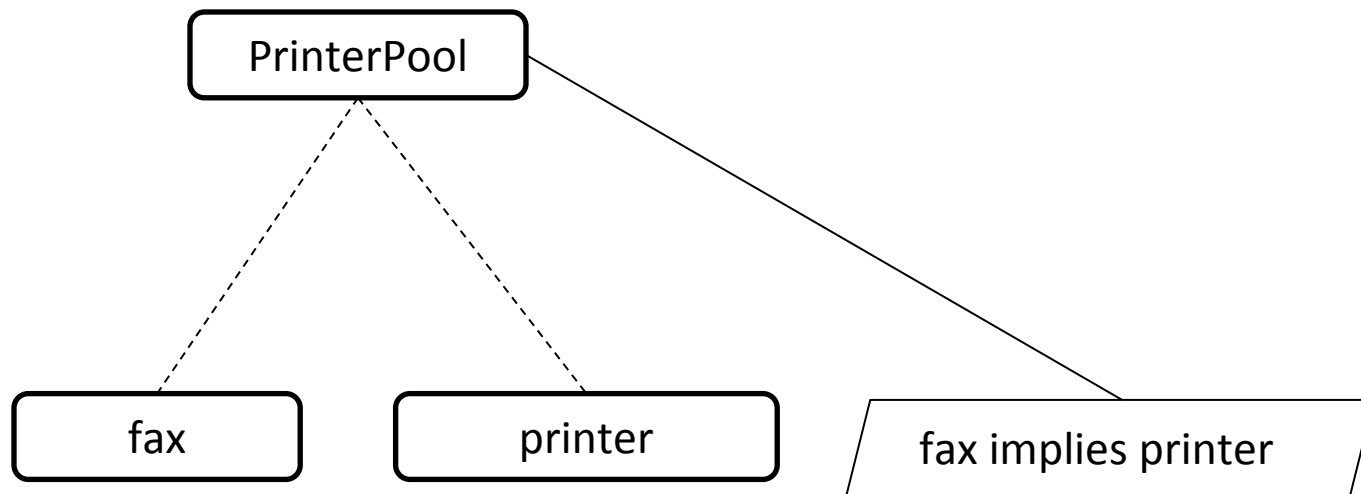
- Configuration semantics akin to cardinality based feature models
- Negative resolution implication
- Positive resolution implication
- Group Multiplicity
- Instance Multiplicity



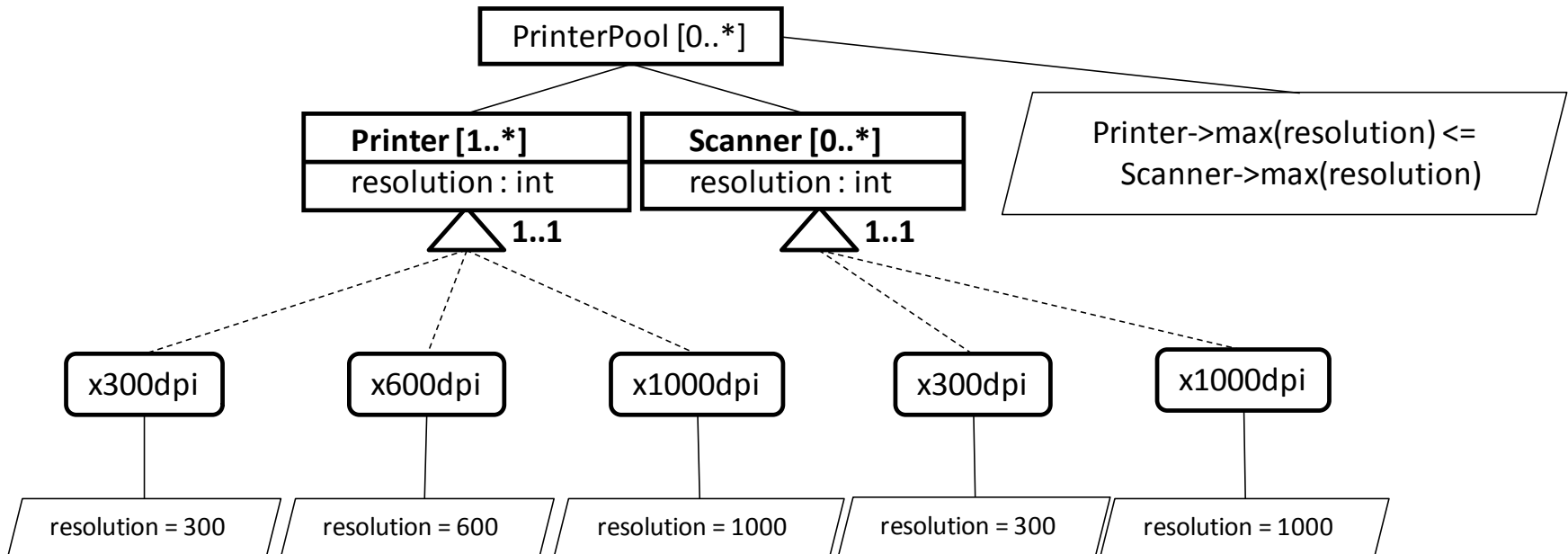
Constraints

- Constraints express intricate relations between VSpecs
- Relations that cannot be captured by trees
- CVL offers basic constraint language
- A sublanguage of OCL
- Other constraint languages, including full OCL, are admitted

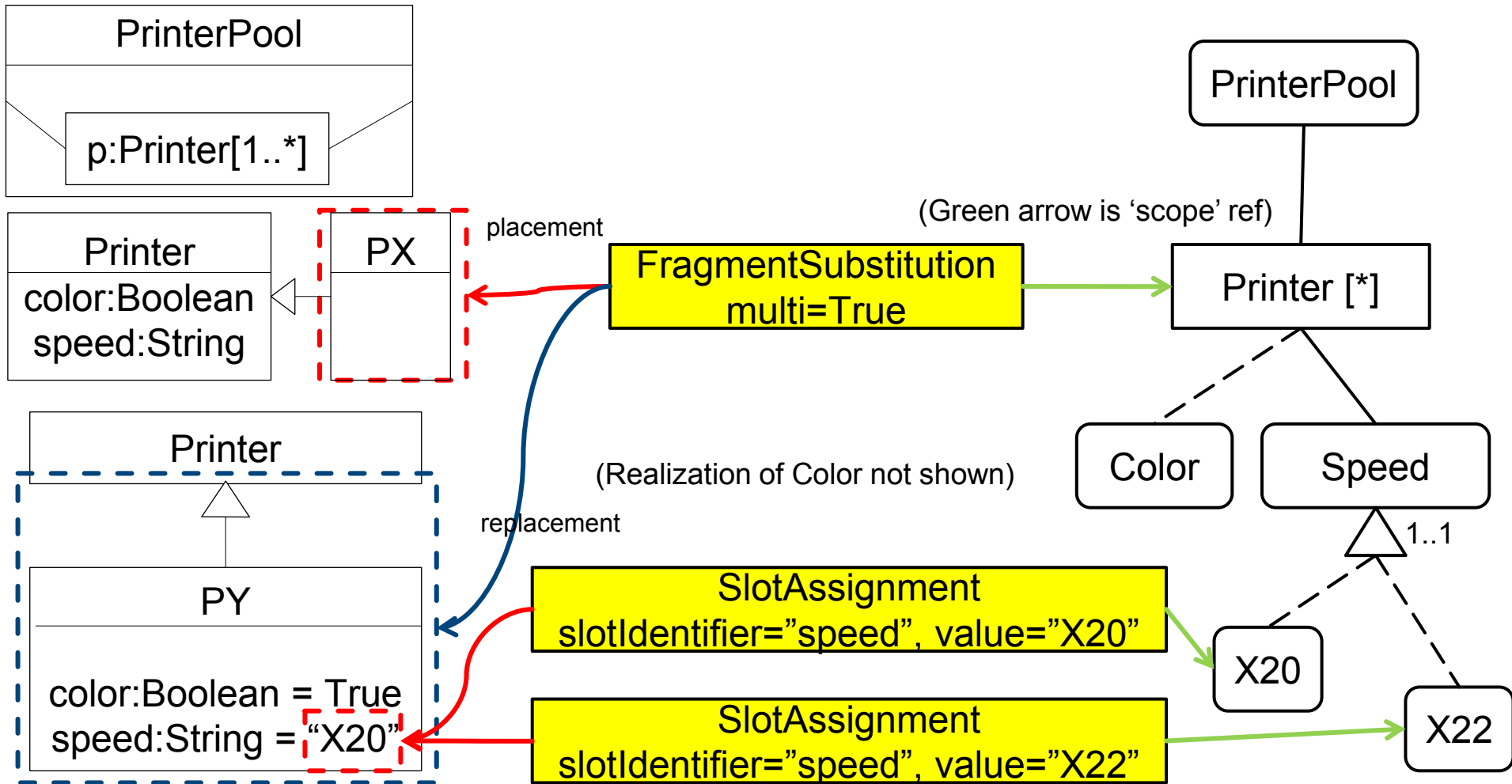
Simple Propositional Constraints



Arithmetic Constraint w/ VClassifier

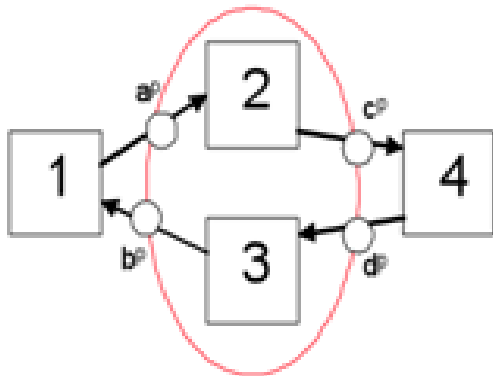


Realization Layer of the PrinterPool

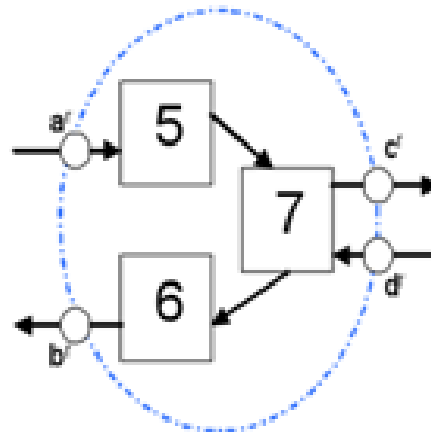


Achieving generality: The Fragment

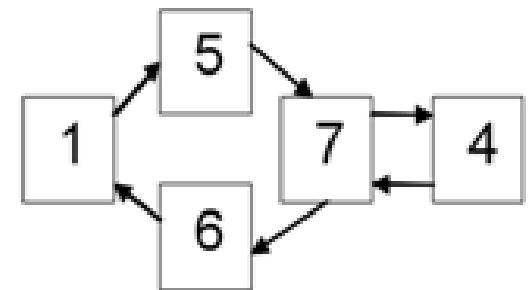
- CVL has two layers: abstraction layer and realization layer
- Substitution is the key concept of realization layer
- The fragments are on the model instances and they are all defined by MOF



Base model plus Placement Fragment



Replacement Fragment

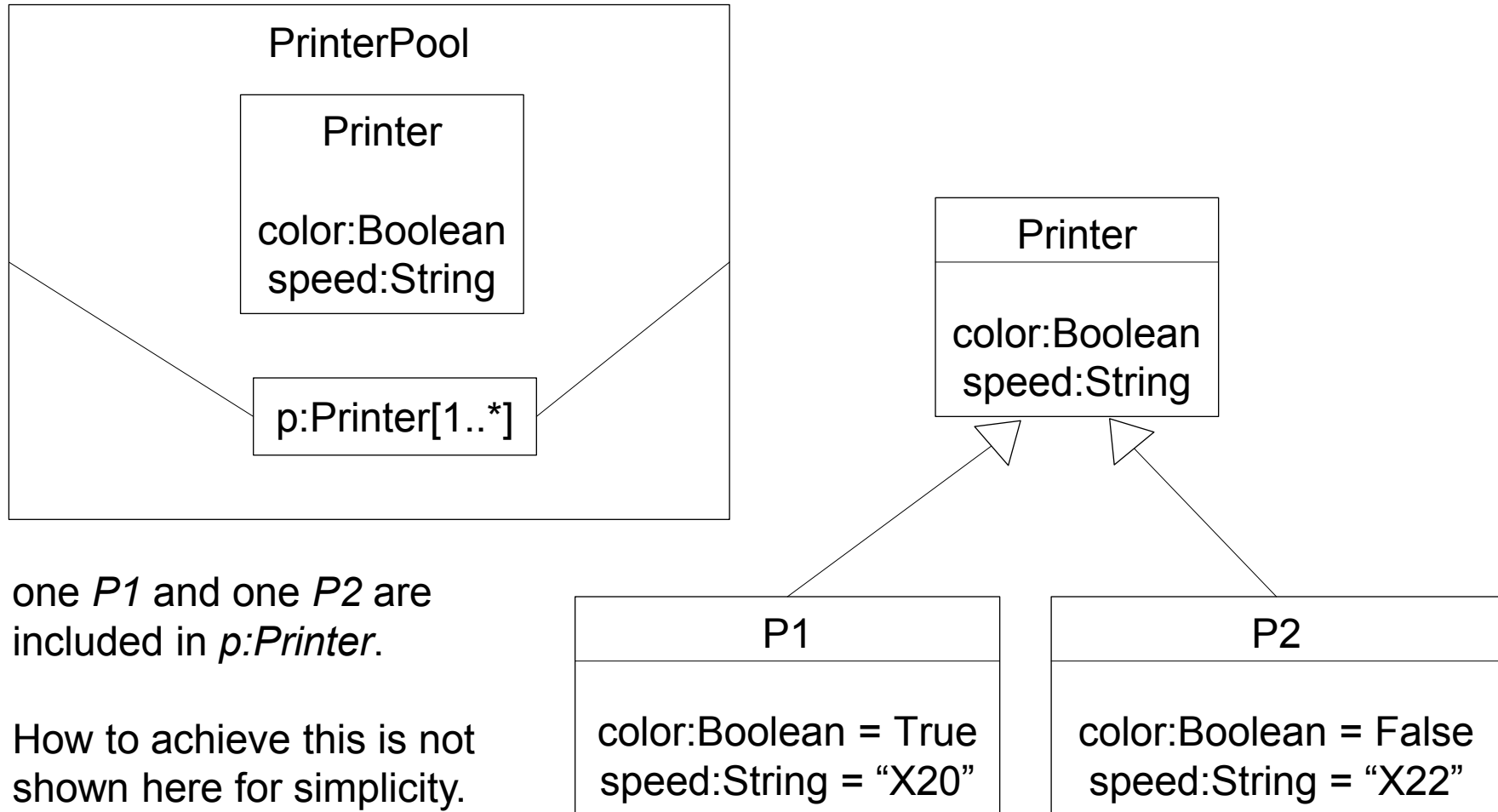


Resolved product model

Variation points in CVL

- Variation Points refer to Base objects
- Variation Points define the base model modifications precisely
- There are different kinds of Variation Points
 - Existence
 - Value assignment
 - Substitution
 - Opaque variation point
 - Configurable Unit

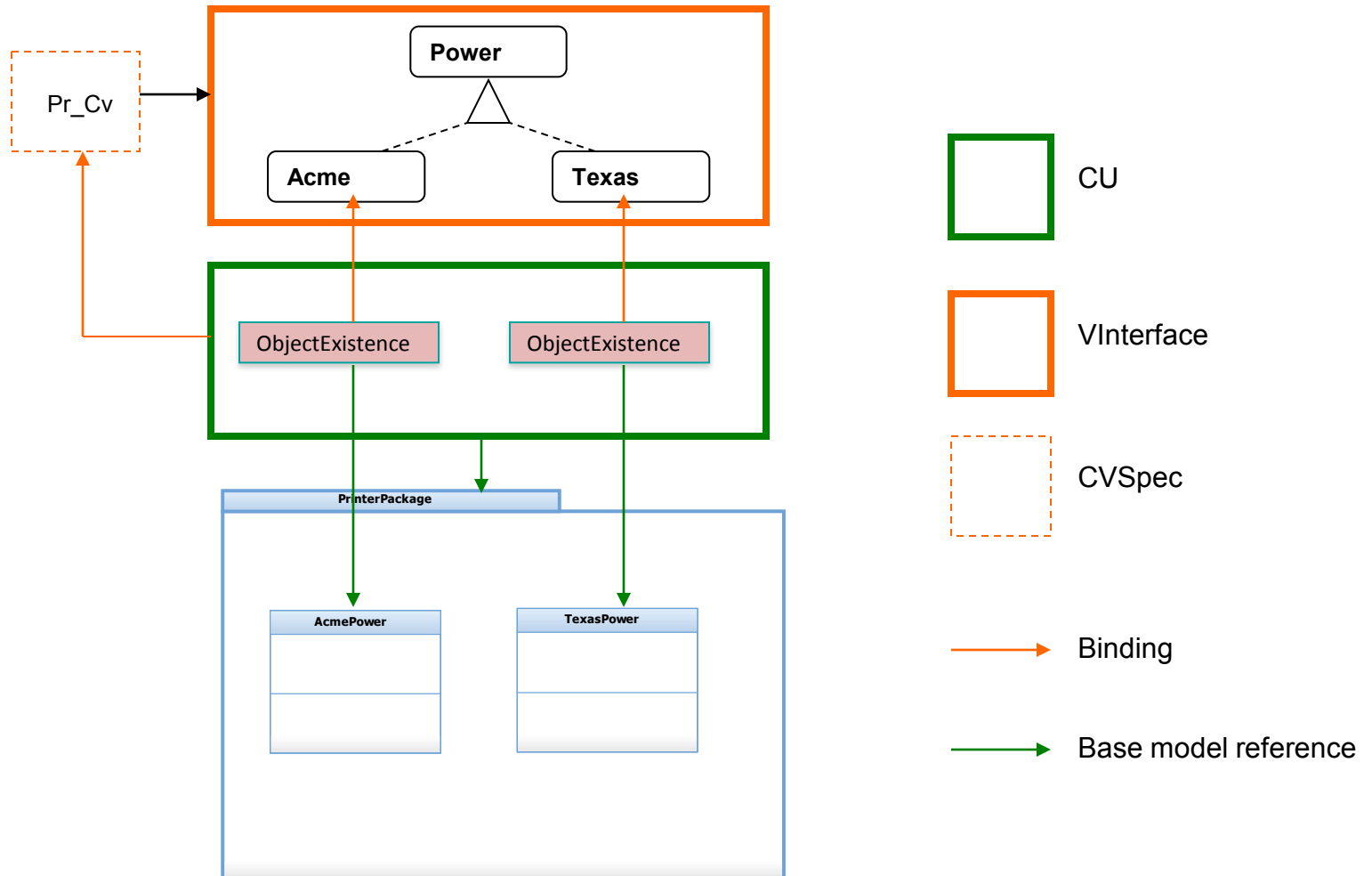
Resulting Product model of PrinterPool



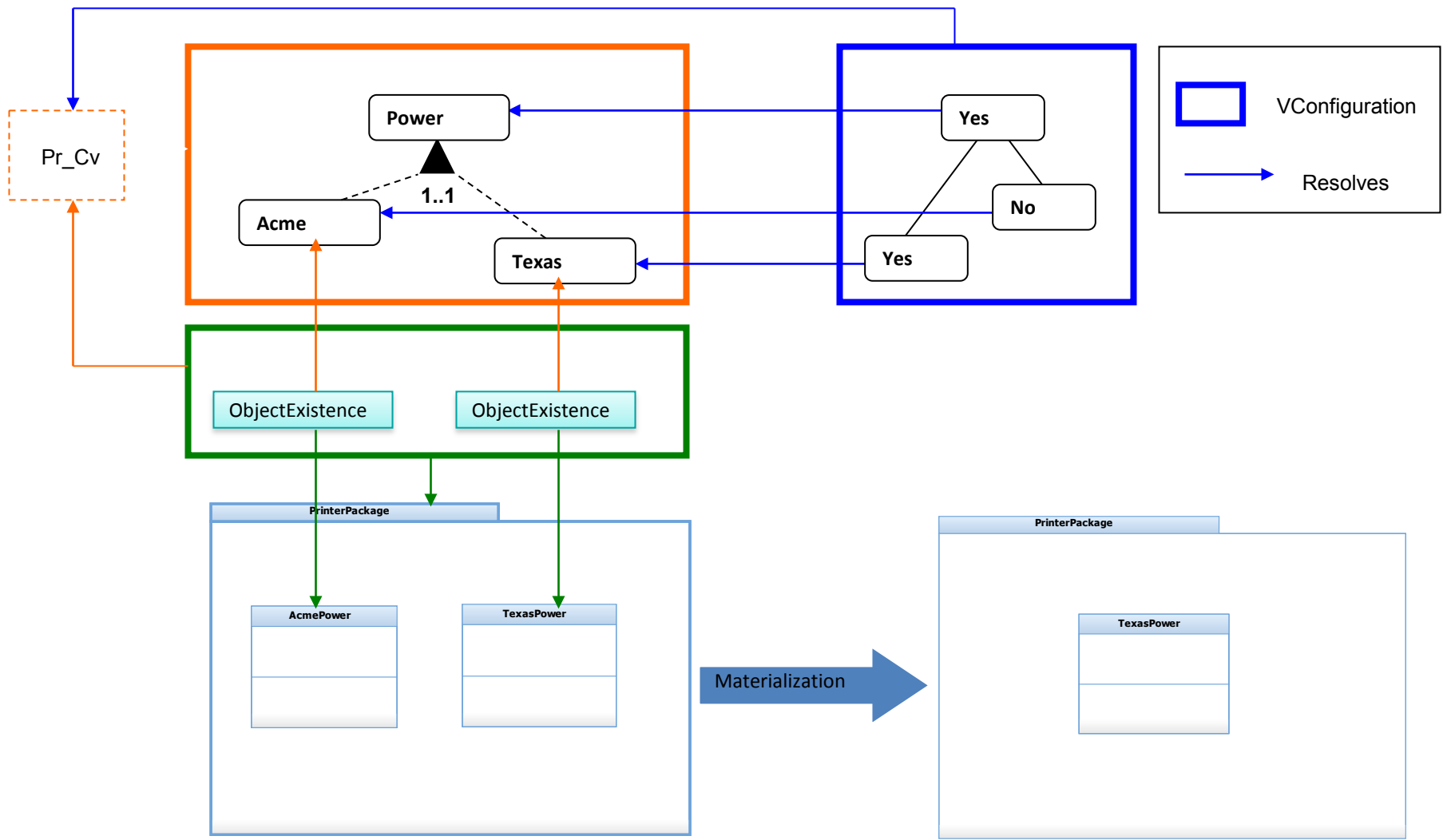
one *P1* and one *P2* are included in *p:Printer*.

How to achieve this is not shown here for simplicity.

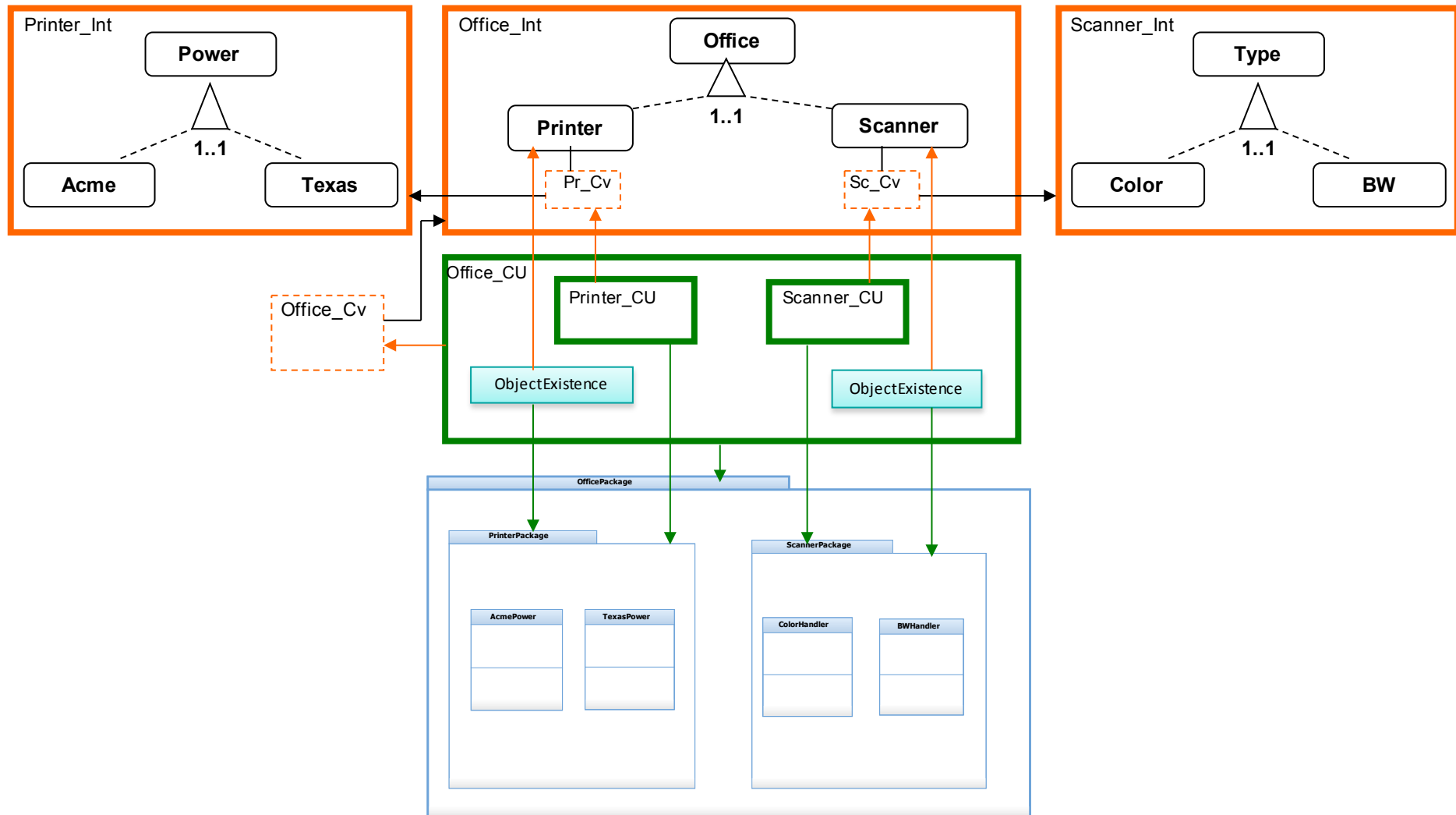
Configurable Unit and VInterface



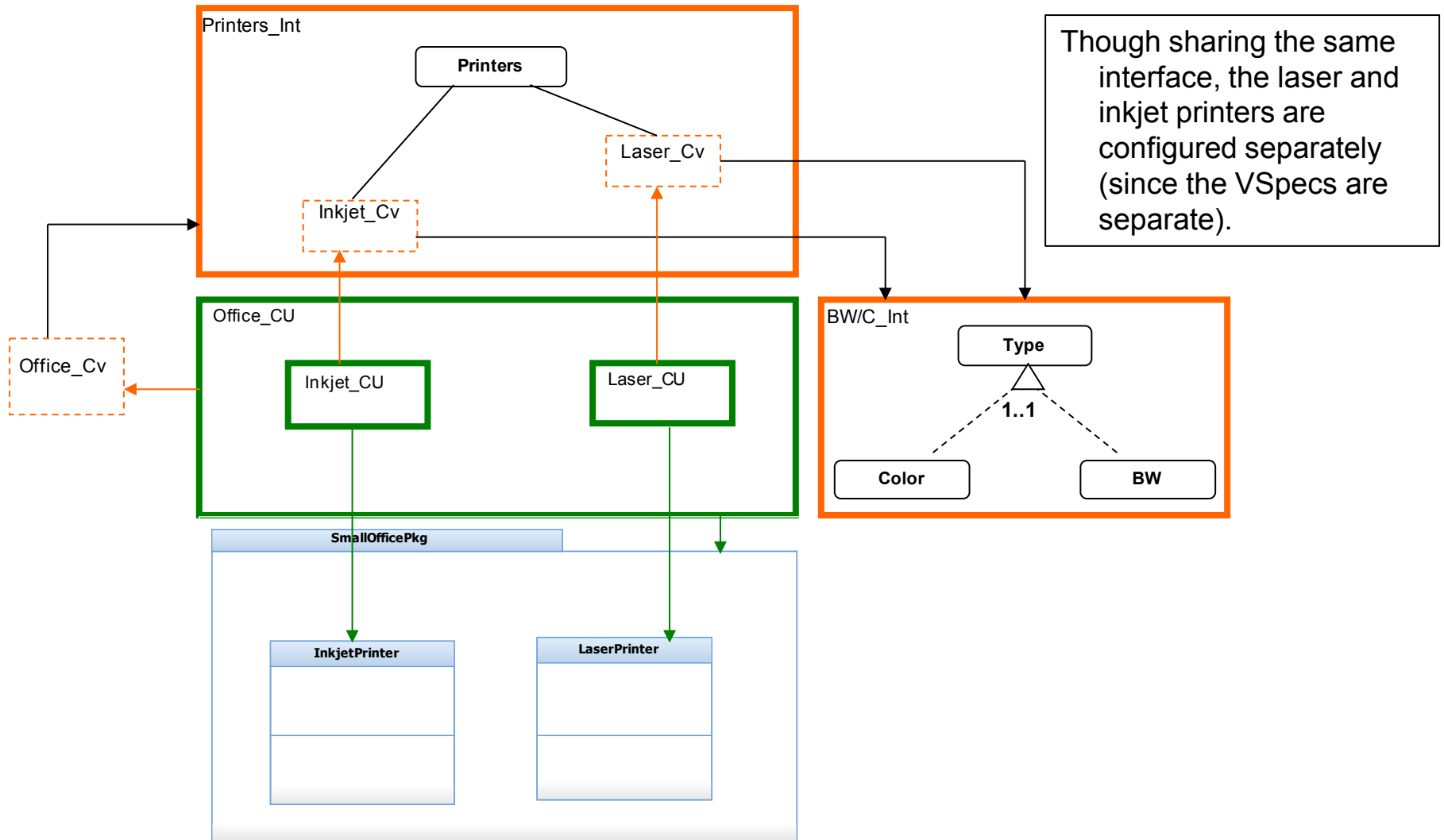
VConfiguration



Composition of Configurable Units



Reusing Variability Interfaces



What happens now with CVL?

Date: August 13, 2012



Common Variability Language (CVL)

OMG Revised Submission

OMG document: ad/2012-08-05

Submitters

IBM
Fraunhofer FOKUS
Thales
Tata Consultancy Services

Supporters

SINTEF
University of Oslo
Tecnalia Research & Innovation
University of Waterloo
IT University of Copenhagen
INRIA
CEA
Atego
Pure-systems

Primary Contact:

Øystein Haugen, SINTEF
oystein.haugen@sintef.no

CVL in the OMG process

- CVL Revised Submission exists
- In 3 months it is possible that CVL will be "adopted technology"
 - This depends on the reviews in OMG
- When adopted the FTF phase starts
 - FTF = Finalization Task Force
 - Tool vendors will implement CVL tooling and find problems
- After successful FTF then CVL will be "available technology"

Potential tool vendors for CVL?

- Research/experimental tools
 - from SINTEF, INRIA
- Internal proprietary tools
 - potentially from TCS
- Commercial tool vendors in consortium
 - IBM, pure-systems, Atego
- Commercial tool vendors outside consortium
 - Big Lever, NoMagic, Sparx,

- Where to find the CVL Revised Submission?
 - <http://variabilitymodeling.org/>