CVL Tutorial

Motivation and Background
What do we mean by “variability”? 

- **Product Line variance**
  - often variants of the same software base
- **Cross-cutting variability**
  - often variability is orthogonal to the software design
  - variability needs are discovered after the first software design
- **The variability designer is not always the software designer**
  - division of labor and of competences
# Common ways to model variability

<table>
<thead>
<tr>
<th>Constructs</th>
<th>Framework/Configuration</th>
<th>Union-of-all-systems</th>
<th>Domain Specific Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td>How?</td>
<td>By mechanisms of a general language</td>
<td>As annotations to a language</td>
<td>By the specific language mechanisms</td>
</tr>
<tr>
<td><strong>unforeseen modeling needs</strong></td>
<td>Just enhance the final model</td>
<td>Enhance the product line model</td>
<td>If not expressible, enhance the language</td>
</tr>
</tbody>
</table>
CVL - Common Variability Language

Formal Analysis

Base: DSL e.g. TCL

CVL Model

Station

Basic Track

Additional Tracks

Main Signal

Manual Switch

Remote Switch

Divert Main Signal

LEFT

RIGHT

validate

apply

unify
Goals for OMG Standardization

• Intuitive ways to describe the product line variability on high abstraction level
  – and means to configure products without being a software specialist
• Automatic means to produce products from product line (cf. MDA)
• Generic ways to describe variability
  – that can work well with product line descriptions in any base language
• Techniques for making generic tools that can work well with the tools for the base languages
Common Variability Language

• CVL = Common Variability Language
• The objective of CVL is to enable the specification of the variability in product line models in order to support seamless product line modeling across the whole product line engineering process.
• CVL is a specification language including a metamodel, semantics and concrete syntax for variability specification.
• Variability specifications shall relate to a base product line model that describes the whole product line and shall comprise:
  – a variability model with the following elements: a model of possible choices and relationships between those choices and the base model
  – resolution models which resolve variability (by a set of choices) and thus define specific product models.
• CVL shall support base models in languages that are defined by means of metamodels, including UML and Domain Specific Languages.