VARiability In safety-critical Embedded Systems

CVL – Common Variability Language or Chaos, Vanity and Limitations?

by Øystein Haugen, SINTEF
Overview of the talk

• Short bio of Øystein Haugen
• What is CVL?
• Chaos
• Vanity
• Limitations
• The Future of CVL
Øystein Haugen – who am I?

University and Research Inst.
- 80-81: UiO, Research assistant for Kristen Nygaard
- 81-84: Norwegian Computing Center
  - Simula-machine
- 97: Practitioners’ verification of SDL systems (Dr. scient.)
- 98-03: Ifi, UiO as Part time Associate Professor
- 04-07: Associate Professor at Ifi (100%)
- 07- : Senior Researcher SINTEF
  - Projects on modeling languages e.g. for variability, train control and pay rolls
- 07- : Assoc. Professor at Ifi (20%)
- 10: General Chair MODELS 2010

Industry and Standardization
- 84-88: SimTech, typographical applications
- 88-90: ABB Technology
  - SDL, prototype SDL tool, ATC
- 91-97: Independent Consultant
- 96-00: Rapporteur ITU for MSC
- 97-03: Ericsson, NorARC
- 99-11: OMG wrt. UML 2.0
  - Responsible for UML 2.x chapter on Interactions
- 09 - : OMG CVL – Common Variability Language
  - Coordinates joint submission team
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>Framework/Configuration</th>
<th>Union-of-all-systems</th>
<th>Domain Specific Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>How?</strong></td>
<td>By mechanisms of a general language</td>
<td>As annotations to a language</td>
</tr>
<tr>
<td><strong>Constructs</strong></td>
<td>Function, Type, Inheritance, Template, Plugin</td>
<td>Pragma, Stereotype</td>
</tr>
<tr>
<td><strong>unforeseen modeling needs</strong></td>
<td>Just enhance the final model</td>
<td>Enhance the product line model</td>
</tr>
</tbody>
</table>
CVL - Common Variability Language

Formal Analysis

Base: DSL e.g. TCL

CVL Model

CVL and its tool

apply

validate

unify

apply
Common Variability Language (CVL)

- **Generic & Standardized**
- **Focused on a domain**

Variability model

- Specification in CVL of base model variabilities

Base model

- Product line model in any MOF-compliant language

Execute CVL

- Selection of a set of choices in the variation model
- Product models fully described in the base language.
- All regular base language tools can be applied to these models

Resolved models

- All regular base language tools can be applied to these models
CVL Architecture

Configurable Units

Variability Realization

Variation Points

Variability Abstraction

VSpecs

Constraints

Resolutions

Base Model
Variation Points over base model

CVL variation points

SYSML (base model) elements

- **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
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
VSpec trees and binding

VSpec tree
- choice
- variable
- constraint

Variation points

VSpec trees and binding

HighSpeed & threshold > 100 → EmergencyPower

Printer

HighSpeed

EmergencyPower

threshold:int

:ObjectExistence
:SlotValueAssignment
:ObjectExistence

MainPower

Operations

inputSection

highSpeedConnector

Operations

mainSupply:MainPower

Operations

emgSupply:EmgPower

Operations

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

VSeps (Variation Specifications) describe the abstract variability

- Every Variation Point is bound to exactly one VSpec
- VSpecs come in different kinds:
  - Choice
  - Variable
  - Constraint
  - VClassifier
  - CVSpec
Constraints in CVL

- CVL include a basic language for expressing constraints on the VSpec tree
  - Propositional logic is supported
- CVL also has the opportunity to let you apply other constraint languages like OCL
Resolution

HighSpeed & threshold>100 → EmergencyPower

Resolution model
- False
- True
- 90

Variation points


:SlotValueAssignment

:ObjectExistence

:ObjectExistence

MainSupply:MainPower

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment

:SlotValueAssignment
Variability Resolution in CVL

- VSpecResolution elements refer to VSpecs
- The set of valid Resolutions is restricted by the constraints
- Represent information necessary to materialize product models
  - Actual yes/no decisions on Choices
  - Actual values to Variables
  - Instances of VClassifiers
  - Configurations of CVSpecs/Configurable Units
Materialization

Resolution model
- False
- True
- 90

Variation points

Materialization

Threshold = 90
Configurable Unit and VInterface

- Power
- Acme
- Texas
- ObjectExistence
- PrinterPackage
- Printer_CU
- Printer_Int
- Pr_Cv
- CU
- VInterface
- CVSpec
- Binding
- Base model reference

- AcmePower
- TexasPower
CVL – more information

- Please look at [http://variabilitymodeling.org](http://variabilitymodeling.org)
- There you will find e.g.
  - The Revised Submission of CVL
  - Tutorial slides from SPLC tutorial on CVL
  - Links to supplementary, and historical material
CHAO?

HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)

SITUATION:
THERE ARE 14 COMPETING STANDARDS.

14?! RIDICULOUS!
WE NEED TO DEVELOP
ONE UNIVERSAL STANDARD
THAT COVERS EVERYONE’S
USE CASES.

YEAH!

[CHARACTER 1] [CHARACTER 2]

[SOON]:

SITUATION:
THERE ARE 15 COMPETING STANDARDS.
Why standardize?
Why standardize a language?

- **Common terms and interpretation**
  - across persons, teams, companies and cultures
    - Experience SISU project
      - Very large SDL specification ported from Alcatel to Kongsberg
    - Experience MSC
      - We have a Korean translation of MSC 2000
  - across computers! portability
    - Experience Simula
      - We ported the exact same code on at least 5 machines without changing a single line of code around 1980

- **Common teaching material**
- **Common libraries**
- **Common and open reviewing process**
OMG is valuable from many different perspectives:

- For the small tool vendors,
  - OMG may provide standards that let the small SMEs play on the same field as the big ones through interfacing through the same technology.
- For the large industry,
  - OMG may provide means to establish the fundaments of multiple vendors of tooling.
- For the academic,
  - OMG is a challenge because you have to defend your ideas in a larger context, and
- for the industrial researcher,
  - OMG is a vehicle for disseminating industrial results to a wider community in a form that many companies can attach to.

-Dr. Øystein Haugen, Senior Researcher, SINTEF
Why make something special/proprietary?

- Nokia N95
- iPhone
- HTC Wildfire
CVL in OMG

Late 2009, Request for Proposals (RFP)

Late 2010, Initial Submission

September 2012 Revised submission

Spring 2013, Finalization?

JOINT SUBMISSION TEAM

initiators from the MoSiS project

tool vendors

users & consultants

research institutes and universities

SINTEF
technalia

IBM

datего

pure-systems

TATA

University of Waterloo

Inria

CEA

List

IT UNIVERSITY OF COPENHAGEN

Fraunhofer

FOKUS

COMMON
VARABILITY
LANGUAGE
In Variability Modeling there has been
- focus on notation
- focus on the abstraction layer (feature models)
- few attempts to standardize formally

but for some reason now things start to happen
- Separate language (variability language as such)
  - CVL (our baby)
- Amalgamated approaches (variability combined into another language)
  - AUTOSAR
  - Matlab Simulink
  - SysML
Success is not guaranteed

- **Risk 0: Nobody cares about your proposed standard**
  - This is why I prepared OMG over many years (since 2005)
- **Risk 1: You meet strong opposition**
  - This once happened when we wanted OO into SDL (1989-1992)
  - This happened again when we wanted UML more precise (1999)
- **Risk 2: You run out of time and money**
  - Research projects are seldom more than 4 years
  - Standardization may easily take longer
- **Risk 3: You achieve a standard, but nobody makes tools**
  - This has happened to many standards
- **Risk 4: There is a standard, there are tools, but no users**
  - May be your compromises were just not good enough ....
Some assets of CVL

• CVL emerges from a series of European research projects:
  – CAFE, Families, MoSiS, VERDE, CESAR, VARIES, ...
• CVL has a true international consortium
  – Europe, North-America, India
• CVL has a fine mixture of competence
  – Tool vendors (Atego, pure-systems, IBM)
  – Serious users (Thales, TCS)
  – Renowned scientists from research institute and universities
Limitations?

• For the academic:
  – Cannot invent new things all the time
  – Needs to have the totality in mind
    • not only a very limited part of the domain/language
    • no assumptions of simplification
  – Little personal credit for authoring a standard

• For the industrial researcher
  – The tool prototype is lagging behind the language
    • CVL 1 Tool is supporting something similar to OMG CVL but ...
    • ATL and MOFScript were not quite supporting the OMG standard
  – The project terminates with no standard in sight
• For the tool vendor:
  – The compromises of the standardization increases the need to modify the tool they already have
  – They always want the simplest solution that they believe their customers are asking for
    • The solution perceived as simplest may also be the most special
• For the industrial early adopter:
  – Tool support is lacking!
    • As an Ericsson representative I experienced that with SDL, MSC and UML
    • The more we are asking for of the standard, the more the tooling is lagging
BigLever (Krueger) has one US patent and a couple of patent applications

Krueger claims that any tool that supports CVL may infringe upon these patents

Krueger does not want to be more specific than this and says that the information is in the patent papers
The BigLever patents

  - **Abstract:** A system and method for the mass customization of software includes a software production line infrastructure, development environment, and actuator. The infrastructure organizes the software production line using a feature declarations component, a product definitions component, and automatons component and a partition composition component. The development environment browses, displays, organizes, edits, and maintains the infrastructure. The actuator actuates the software production line to produce custom software products.

- **US Patent Application, 12/273,352, Filed Nov 18, 2008.** This contains claims that are still pending from the original patent listed above.

What happens now with CVL?

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
- CVL was presented to the OMG Architectural Board in December 2012
  - The AB will vote on adoption by e-mail once some technicalities of the metamodel has been cleared
- When technically adopted the FTF phase starts
  - FTF = Finalization Task Force
  - Tool vendors will implement CVL tooling and find problems
- There will also be a vote by the OMG Business Committee on the business aspects of the standard.
  - They will also consider the patent infringement possibilities
- 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/
The Future of CVL

- Common or Chaos?
- Variability Language or a Variety of languages?
- Limitations or leverage of the community?