

Language Levels for the Universal Variability Language: An Extension Mechanism and Conversion Strategies

Stefan Vill, Chico Sundermann, Thomas Thüm | FOSD 23, March 28, 2023





Feature Diagrams



Feature Diagrams



- Cross-Tree Constraints
- Feature Attributes
- Decomposition

1	namespace computer_model
2	
3	Imports
4	cpu_model
5	
6	features
7	Computer
8	mandatory
9	RAM
10	or
11	RAM8
12	RAM16
13	cpu_model.CPU
14	optional
15	SATA—Devices {abstract}
16	[12]
17	DVD-drive {power 10}
18	Card—reader {power 7}
19	Blu-ray-drive {power 15}
20	alternative
21	strong_PSU
22	weak_PSU
23	
24	constraints
25	Blu-ray-drive => strong_PSU
-	· · · · · · · · · · · · · · · · · · ·

- Cross-Tree Constraints
- Feature Attributes
- Decomposition

```
1 namespace computer_model
 2
 3
   imports
 4
        cpu model
 5
 6
   features
 7
     Computer
 8
        mandatory
 9
          RAM
10
            or
              RAM8
11
12
              RAM16
13
          cpu_model.CPU
14
        optional
          SATA-Devices {abstract}
15
16
            [1.2]
              DVD-drive {power 10}
17
18
              Card-reader {power 7}
              Blu-ray-drive {power 15}
19
20
        alternative
21
          strong PSU
          weak PSU
22
23
24
   constraints
25
     Blu-ray-drive => strong_PSU
```

- Cross-Tree Constraints
- Feature Attributes
- Decomposition

1	namespace computer_model
2	
3	imports
4	cpu_model
5	
6	features
7	Computer
8	mandatory
9	RAM
10	or
11	RAM8
12	RAM16
13	cpu_model.CPU
14	optional
15	SATA-Devices {abstract}
16	[12]
17	DVD—drive {power 10}
18	Card—reader {power 7}
19	Blu-ray-drive {power 15}
20	alternative
21	strong_PSU
22	weak_PSU
23	
24	constraints
25	Blu-ray-drive => strong_PSU

- Cross-Tree Constraints
- Feature Attributes
- Decomposition

1	namespace computer_model
2	
3	imports
4	cpu_model
5	
6	features
7	Computer
8	mandatory
9	RAM
10	or
11	RAM8
12	RAM16
13	cpu_model.CPU
14	optional
15	SATA-Devices {abstract}
16	[12]
17	DVD-drive {power 10}
18	Card—reader {power 7}
19	Blu-ray-drive {power 15}
20	alternative
21	strong_PSU
22	weak_PSU
23	
24	constraints
25	Blu-ray-drive => strong_PSU

- Groups
- Cross-Tree Constraints
- Feature Attributes
- Decomposition

1	namespace computer_model
2	
3	imports
4	cpu_model
5	
6	features
7	Computer
8	mandatory
9	RAM
10	or
11	RAM8
12	RAM16
13	cpu_model.CPU
14	optional
15	SATA—Devices {abstract}
16	[12]
17	DVD—drive {power 10}
18	Card-reader {power 7}
19	Blu-ray-drive {power 15}
20	alternative
21	strong_PSU
22	weak_PSU
23	
24	constraints
25	Blu-ray-drive => strong_PSU

Feature Attribute Constraints



 $\begin{array}{l} {\sf Graphics}{\sf -Card-1.power} + {\sf Graphics}{\sf -Card-2.power} \\ + {\sf CPU-1.power} + {\sf CPU-2.power} > 230 \Rightarrow {\sf Strong} \end{array}$

Aggregate Function



Feature Cardinality



Why Language Levels?

- Simple UVL \Rightarrow Limited Use Cases
- Complex UVL \Rightarrow Complex Tool Integration
 - UVL supports Group Cardinality \Rightarrow Tool must handle Group Cardinality
 - UVL supports Attribute Constraints \Rightarrow Tool must handle Attribute Constraint
 - ...

Solution: Different levels of UVL \Rightarrow Tools can integrate UVL partially

Language Levels

- A Language Level encapsulates optional UVL features
- Different types of Language Levels (major and minor) based on idea from Thüm et al.
 - Major: Based on solving techniques
 - Minor: Based on use-cases and assigned to major level
- Adopt popular language features from other feature-modelling languages and tools

Language Level Overview





SMT: e.g. A + B < 7 satisfiable?

Conversion Strategy Architecture



 \leftarrow : Conversion Strategy

- One conversion strategy per language level
- Transitive conversion from every level to Basic-UVL possible
- Tools can use UVL models with language levels they do not support

Example Conversion - Aggregate Function



 $sum(\texttt{power}) > 230 \Rightarrow \texttt{Strong}$



Example Conversion - Group Cardinality



 \Rightarrow



 \neg HDD $\land \neg$ SSD $\land \neg$ DVD-drive $\land \neg$ Blu-ray-drive $HDD \land \neg SSD \land \neg DVD$ -drive $\land \neg Blu$ -ray-drive \vee \neg HDD \land SSD $\land \neg$ DVD-drive $\land \neg$ Blu-rav-drive \vee \neg HDD $\land \neg$ SSD \land DVD-drive $\land \neg$ Blu-ray-drive \vee \neg HDD $\land \neg$ SSD $\land \neg$ DVD-drive \land Blu-rav-drive V HDD \land SSD $\land \neg$ DVD-drive $\land \neg$ Blu-ray-drive \vee $HDD \land \neg SSD \land DVD$ -drive $\land \neg Blu$ -rav-drive \vee $HDD \land \neg SSD \land \neg DVD$ -drive $\land Blu$ -ray-drive V \neg HDD \land SSD \land DVD-drive $\land \neg$ Blu-rav-drive V \neg HDD \land SSD $\land \neg$ DVD-drive \land Blu-rav-drive V \vee \neg HDD $\land \neg$ SSD \land DVD-drive \land Blu-rav-drive)

Example Conversion - Feature Cardinality



Implementation

- Java library "uvl-parser2.0" supporting this UVL draft
- Parsing, Printing, Automatic transitive conversion
- Published on GitHub under LGPL-3.0 license
- Already used by FeatureIDE and TraVarT





https://github.com/Universal-Variability-Language/uvlparser2.0

Conclusion



New use case? \Rightarrow New language level + one conversion strategy

