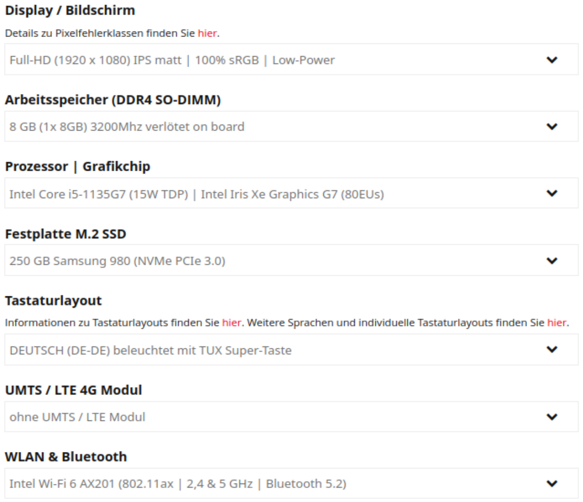
Language Levels for the Universal Variability Language: An Extension Mechanism and Conversion Strategies

Stefan Vill | March 09, 2023

Software Engineering
Programming Languages

universität
uulm

Software Product Line


Display / Bildschirm

Details zu Pixelfehlerklassen finden Sie hier.

Full-HD (1920 x 1080) IPS matt | 100% sRGB | Low-Power

Arbeitsspeicher (DDR4 SO-DIMM)

8 GB (1x 8GB) 3200Mhz verlötet on board

Prozessor | Grafikchip

Intel Core i5-1135G7 (15W TDP) | Intel Iris Xe Graphics G7 (80EUs)

Festplatte M.2 SSD

250 GB Samsung 980 (NVMe PCIe 3.0)

Tastaturlayout

Informationen zu Tastaturlayouts finden Sie hier. Weitere Sprachen und individuelle Tastaturlayouts finden Sie hier.

DEUTSCH (DE-DE) beleuchtet mit TUX Super-Taste

UMTS / LTE 4G Modul

ohne UMTS / LTE Modul

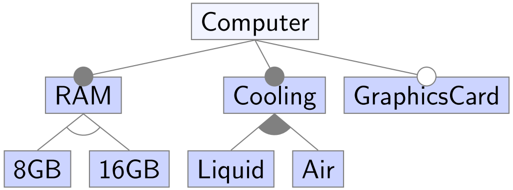
WLAN & Bluetooth

Intel Wi-Fi 6 AX201 (802.11ax | 2,4 & 5 GHz | Bluetooth 5.2)

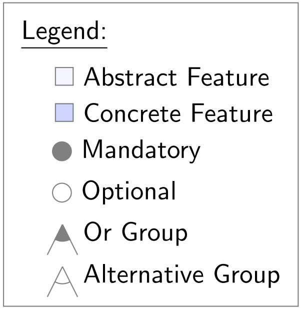
Feature Modelling

Natural Language?	"Every notebook needs either 8 GB of RAM or 16 GB ..."
Enumerating?	$\{\{\mathit{FullHD}, 8\mathit{GBRAM}, ...\}, \{\mathit{FullHD}, 16\mathit{GBRAM}, ...\}, ...\}$
Propositional Formula?	$...(8\mathit{GBRAM} \vee 16\mathit{GBRAM}) \wedge \neg(8\mathit{GBRAM} \wedge 16\mathit{GBRAM})...$

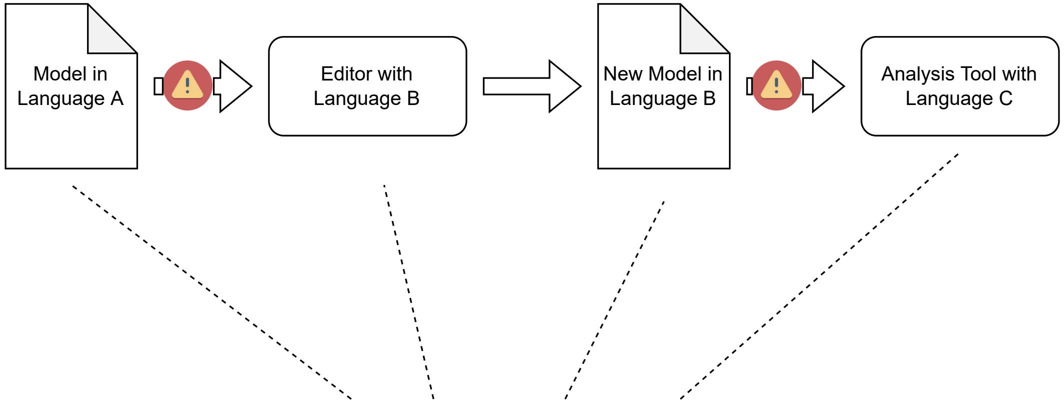
Feature Diagrams



GraphicsCard ⇒ Liquid



Feature Diagrams



One Common Language?

Current Version of UVL

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

```
namespace computer_model

imports
    cpu_model

features
  Computer
    mandatory
      RAM
        or
          RAM8
          RAM16
      cpu_model.CPU
    optional
      SATA-Devices {abstract}
        [1..2]
          DVD-drive {power 10}
          Card-reader {power 7}
          Blu-ray-drive {power 15}
    alternative
      strong_PSU
      weak_PSU

constraints
  Blu-ray-drive => strong_PSU
```

Current Version of UVL

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

```
namespace computer_model

imports
    cpu_model

features
  Computer
    mandatory
      RAM
        or
          RAM8
          RAM16
      cpu_model.CPU
    optional
      SATA-Devices {abstract}
        [1..2]
          DVD-drive {power 10}
          Card-reader {power 7}
          Blu-ray-drive {power 15}
    alternative
      strong_PSU
      weak_PSU

constraints
  Blu-ray-drive => strong_PSU
```

Current Version of UVL

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

```
namespace computer_model

imports
    cpu_model

features
  Computer
    mandatory
      RAM
        or
          RAM8
          RAM16
      cpu_model.CPU
    optional
      SATA-Devices {abstract}
        [1..2]
          DVD-drive {power 10}
          Card-reader {power 7}
          Blu-ray-drive {power 15}
    alternative
      strong_PSU
      weak_PSU

constraints
  Blu-ray-drive => strong_PSU
```

Current Version of UVL

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

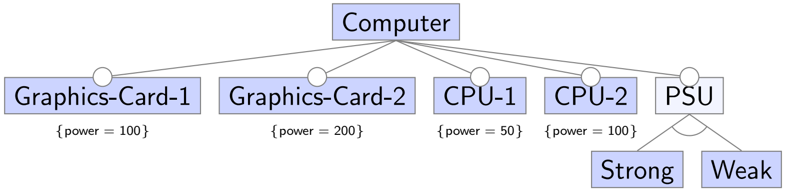
```
namespace computer_model

imports
    cpu_model

features
  Computer
    mandatory
      RAM
        or
          RAM8
          RAM16
      cpu_model.CPU
    optional
      SATA-Devices {abstract}
        [1..2]
          DVD-drive {power 10}
          Card-reader {power 7}
          Blu-ray-drive {power 15}
    alternative
      strong_PSU
      weak_PSU

constraints
  Blu-ray-drive => strong_PSU
```

Current Version of UVL

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

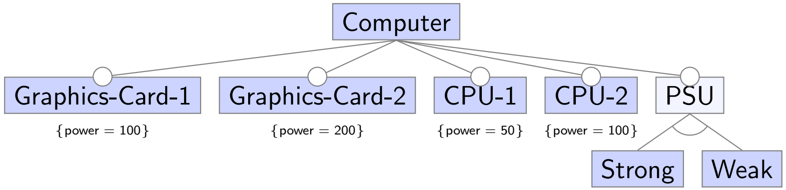
```
namespace computer_model

imports
  cpu_model

features
  Computer
    mandatory
      RAM
        or
          RAM8
          RAM16
      cpu_model.CPU
    optional
      SATA-Devices {abstract}
        [1..2]
          DVD-drive {power 10}
          Card-reader {power 7}
          Blu-ray-drive {power 15}
    alternative
      strong_PSU
      weak_PSU

constraints
  Blu-ray-drive => strong_PSU
```

Feature Attribute Constraints



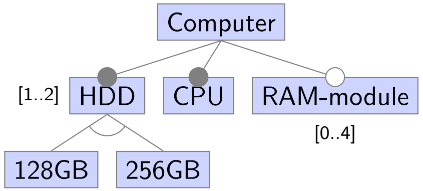
$$\begin{aligned} & \text{Graphics-Card-1.power} + \text{Graphics-Card-2.power} \\ + \quad & \text{CPU-1.power} + \text{CPU-2.power} > 230 \Rightarrow \text{Strong} \end{aligned}$$

Aggregate Function



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

Feature Cardinality



Why Language Levels?

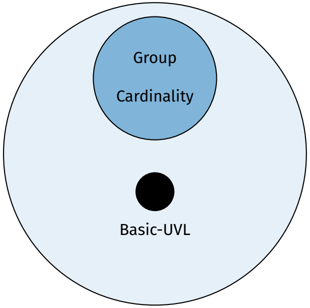
- Simple UVL $\Rightarrow$ Limited Use Cases
- Complex UVL $\Rightarrow$ Complex Tool Integration
  - UVL supports Group Cardinality $\Rightarrow$ Tool must handle Group Cardinality
  - UVL supports Attribute Constraints $\Rightarrow$ Tool must handle Attribute Constraint
  - ...

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

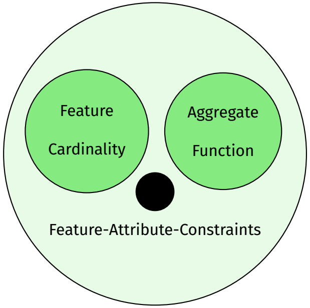
Language Levels

- 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



SAT-level

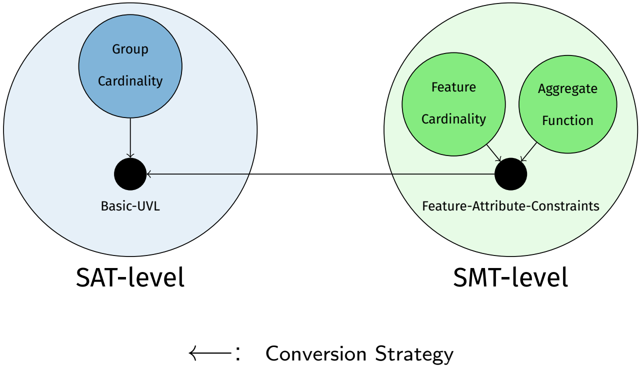


SMT-level

SAT: e.g. $A \wedge B$ satisfiable?

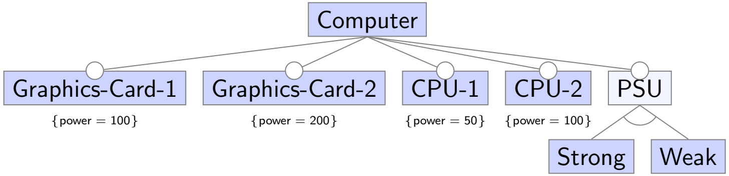
SMT: e.g. $A + B < 7$ satisfiable?

Conversion Strategy Architecture



- 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



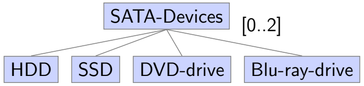
$$sum(\text{power}) > 230 \Rightarrow \text{Strong}$$

$$\Downarrow$$

SMT-Level

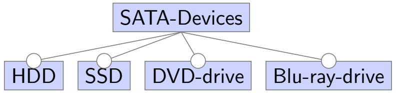
$$\text{Graphics-Card-1.power} + \text{Graphics-Card-2.power} + \text{CPU-1.power} + \text{CPU-2.power} > 230 \Rightarrow \text{Strong}$$

Example Conversion - Group Cardinality



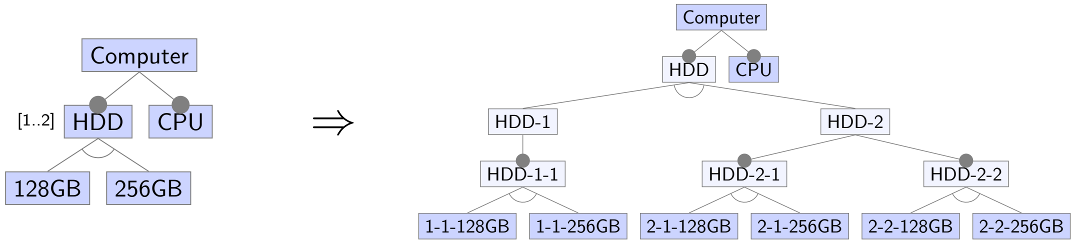
$\Longrightarrow$

SAT-Level



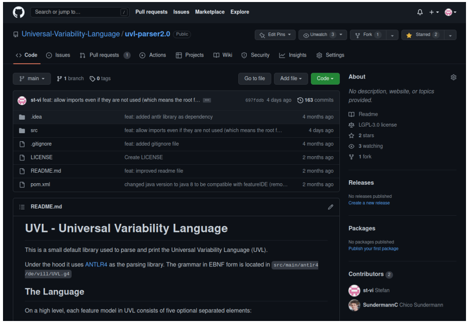
$$
\begin{array}{ll}
(& \neg\text{HDD} \land \neg\text{SSD} \land \neg\text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \text{HDD} \land \neg\text{SSD} \land \neg\text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \neg\text{HDD} \land \text{SSD} \land \neg\text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \neg\text{HDD} \land \neg\text{SSD} \land \text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \neg\text{HDD} \land \neg\text{SSD} \land \neg\text{DVD-drive} \land \text{Blu-ray-drive} \\
\lor & \text{HDD} \land \text{SSD} \land \neg\text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \text{HDD} \land \neg\text{SSD} \land \text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \text{HDD} \land \neg\text{SSD} \land \neg\text{DVD-drive} \land \text{Blu-ray-drive} \\
\lor & \neg\text{HDD} \land \text{SSD} \land \text{DVD-drive} \land \neg\text{Blu-ray-drive} \\
\lor & \neg\text{HDD} \land \text{SSD} \land \neg\text{DVD-drive} \land \text{Blu-ray-drive} \\
\lor & \neg\text{HDD} \land \neg\text{SSD} \land \text{DVD-drive} \land \text{Blu-ray-drive})
\end{array}
$$

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



st-vi feat: allow imports even if they are not used (which means the root f… 697fddb 4 days ago 163 commits

.idea	feat: added antlr library as dependency	4 months ago
src	feat: allow imports even if they are not used (which means the root f…	4 days ago
.gitignore	feat: added gitignore file	4 months ago
LICENSE	Create LICENSE	2 months ago
README.md	feat: improved readme file	2 months ago
pom.xml	changed java version to java 8 to be compatible with featureIDE (remo…	2 months ago

README.md

UVL - Universal Variability Language

This is a small default library used to parse and print the Universal Variability Language (UVL).

Under the hood it uses ANTLR4 as the parsing library. The grammar in EBNF form is located in `src/main/antlr4/de/vill/UVL.g4`

The Language

On a high level, each feature model in UVL consists of five optional separated elements:

About

No description, website, or topics provided.

Readme
LGPL-3.0 license
2 stars
3 watching
1 fork

Releases

No releases published
Create a new release

Packages

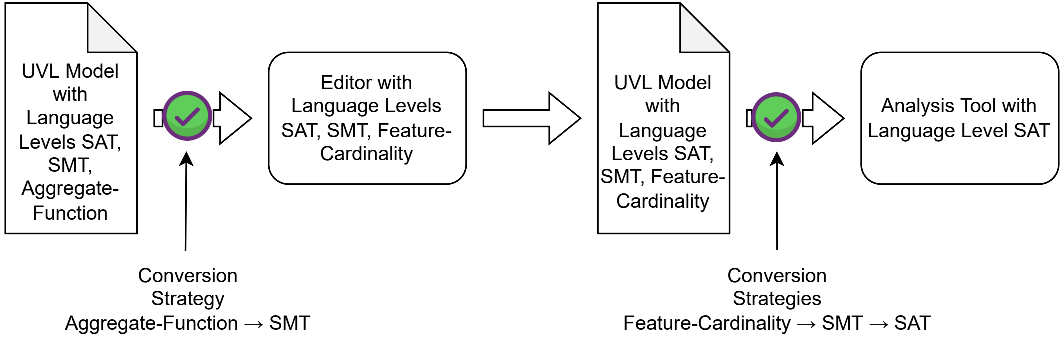
No packages published
Publish your first package

Contributors 2

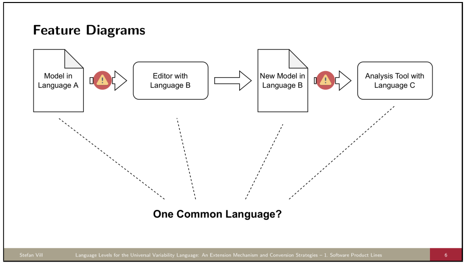
st-vi Stefan
SundermannC Chico Sundermann

https://github.com/Universal-Variability-Language/uvl-parser2.0

Conclusion

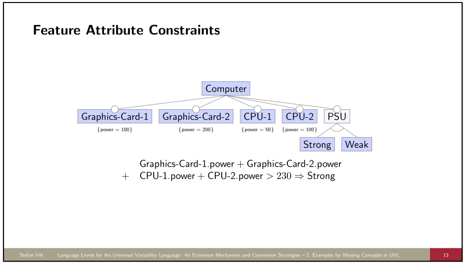

UVL Model with Language Levels SAT, SMT, Aggregate-Function
Editor with Language Levels SAT, SMT, Feature-Cardinality
UVL Model with Language Levels SAT, SMT, Feature-Cardinality
Analysis Tool with Language Level SAT
Conversion Strategy Aggregate-Function → SMT
Conversion Strategies Feature-Cardinality → SMT → SAT

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



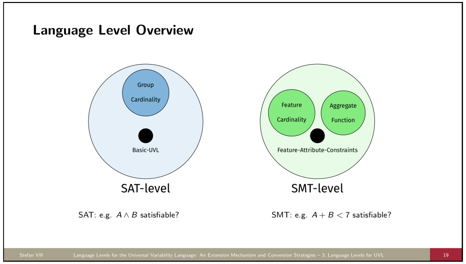
Feature Diagrams

One Common Language?



Feature Attribute Constraints

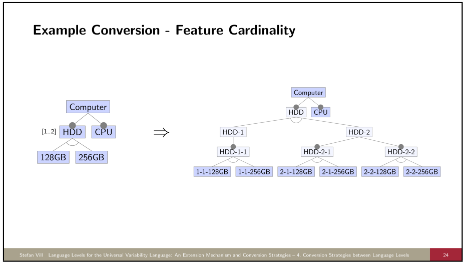
$$\begin{aligned} &\text{Graphics-Card-1.power} + \text{Graphics-Card-2.power} \\ +\ &\text{CPU-1.power} + \text{CPU-2.power} > 230 \Rightarrow \text{Strong} \end{aligned}$$



Language Level Overview

SAT: e.g. $A \land B$ satisfiable?

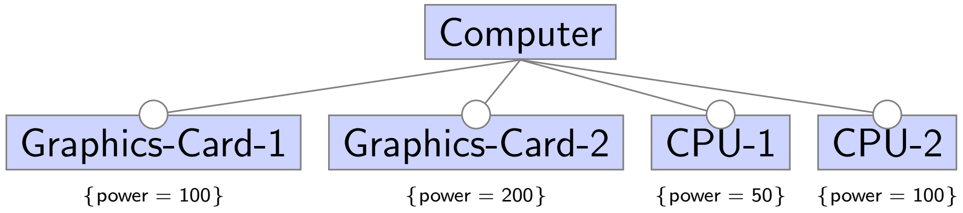
SMT: e.g. $A + B < 7$ satisfiable?



Example Conversion - Feature Cardinality

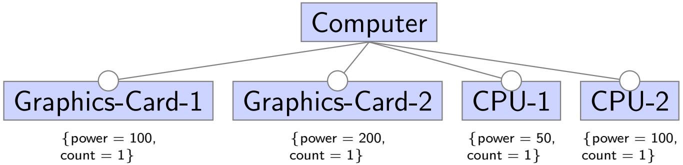
Backup-Slides

Example Conversion - Average Aggregate Before



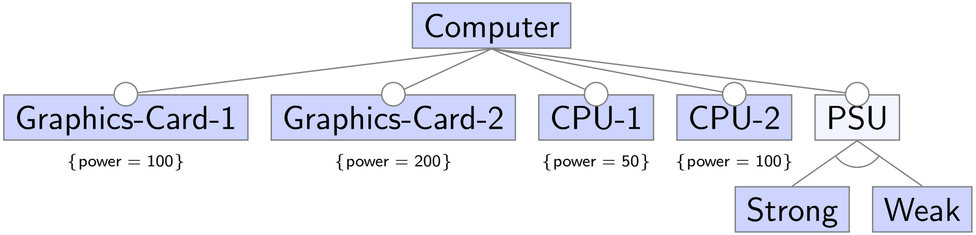
$$avg(\text{power}) < 100$$

Example Conversion - Average Aggregate After



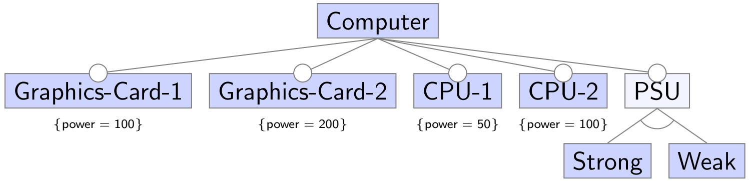
$$\begin{aligned} & ((\text{Graphics-Card-1.power} + \text{Graphics-Card-2.power} + \text{CPU-1.power} + \text{CPU-2.power}) \\ / \quad & (\text{Graphics-Card-1.count} + \text{Graphics-Card-2.count} + \text{CPU-1.count} + \text{CPU-2.count})) \\ & < 100 \end{aligned}$$

Example Conversion - Feature Attribute Constraint Before



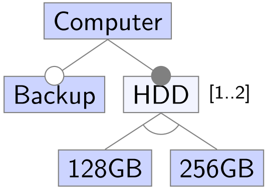
$$\begin{aligned} & \text{Graphics-Card-1.power} + \text{Graphics-Card-2.power} \\ + \quad & \text{CPU-1.power} + \text{CPU-2.power} \\ + \quad & \text{Integrated.power} + \text{BluRay.power} > 230 \Rightarrow \text{Strong} \end{aligned}$$

Example Conversion - Feature Attribute Constraint After



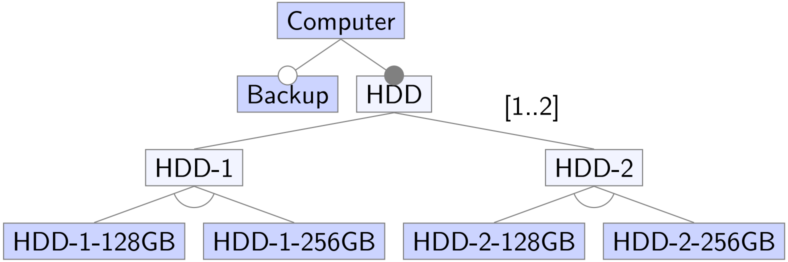
$$
\begin{aligned}
&((\neg \text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \neg \text{CPU-1} \land \text{CPU-2})\\
\lor\ &(\neg \text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \text{CPU-1} \land \neg \text{CPU-2})\\
\lor\ &(\neg \text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \text{CPU-1} \land \text{CPU-2})\\
\lor\ &(\text{Graphics-Card-1} \land \neg \text{Graphics-Card-2} \land \text{CPU-1} \land \text{CPU-2})\\
\lor\ &(\text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \neg \text{CPU-1} \land \neg \text{CPU-2})\\
\lor\ &(\text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \neg \text{CPU-1} \land \text{CPU-2})\\
\lor\ &(\text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \text{CPU-1} \land \neg \text{CPU-2})\\
\lor\ &(\text{Graphics-Card-1} \land \text{Graphics-Card-2} \land \text{CPU-1} \land \text{CPU-2}))\\
&\Rightarrow \text{Strong}
\end{aligned}
$$

Example Conversion - Feature Cardinality Before

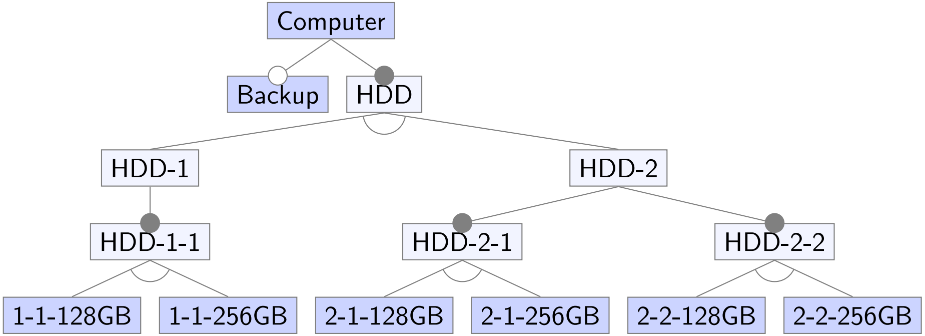


Backup ⇒ 256GB

Example Conversion - Feature Cardinality After - Option 1



Example Conversion - Feature Cardinality After - Option 2



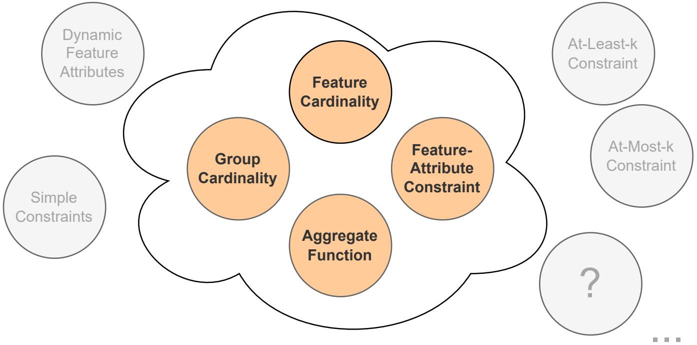
Example Conversion - Feature Cardinality After - Constraints

Before:
Backup $\Rightarrow$ 256GB

After - Option 1:
Backup $\Rightarrow$ (HDD-1-1-256GB $\vee$ HDD-2-1-256GB $\vee$ HDD-2-2-256GB)

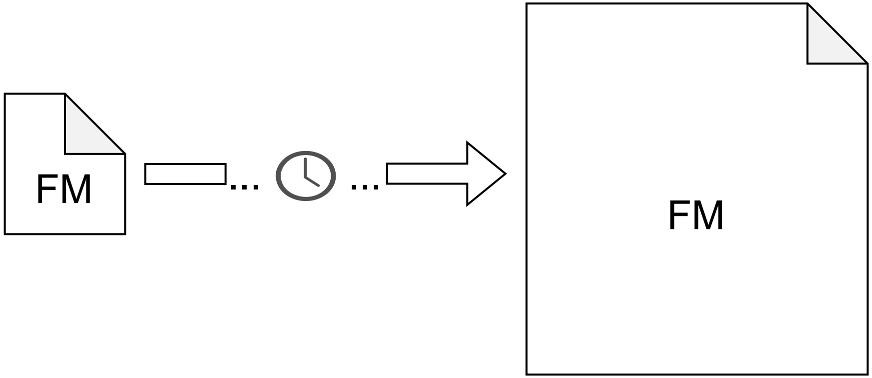
After - Option 2:
HDD-1 $\Rightarrow$ (Backup $\Rightarrow$ HDD-1-1-256GB)
HDD-2 $\Rightarrow$ (Backup $\Rightarrow$ HDD-2-1-256GB)
HDD-2 $\Rightarrow$ (Backup $\Rightarrow$ HDD-2-2-256GB)

Future Work



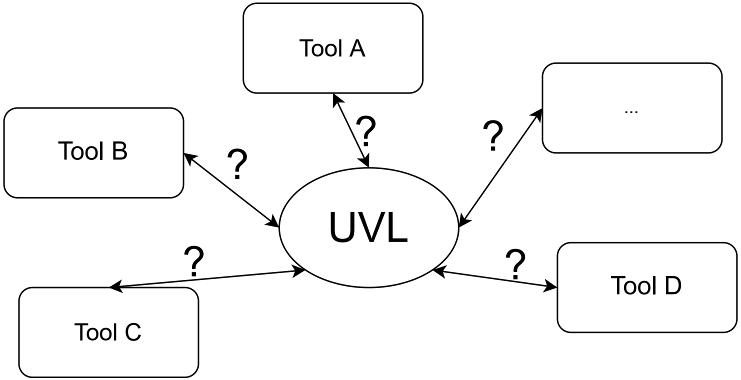
Identify and add new, useful language levels

Future Work



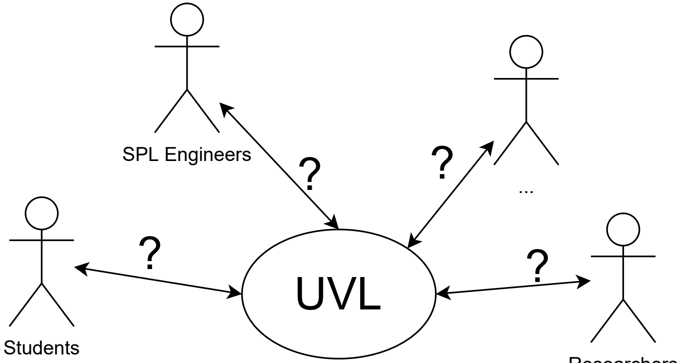
Evaluate (and improve) conversion strategy performance

Future Work



Evaluate tool coverage of UVL

Future Work



Gather community feedback on this UVL draft

Language Level Detection

Explicit enumeration
(exhaustive)

```
include
    SAT-level.group-cardinality
    SMT-level.feature-cardinality

features
    Computer
        optional
            RAM-module cardinality [1..4]
        mandatory
            SATA-Devices
                [0..3]
                    HDD
                    SSD
                    DVD-drive
                    Card-reader
                    Blu-ray-drive
            CPU
```

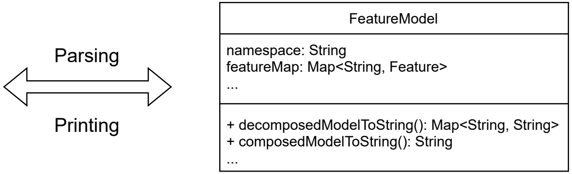
Implicit

```
features
    Computer
        optional
            RAM-module cardinality [1..4]
        mandatory
            SATA-Devices
                [0..3]
                    HDD
                    SSD
                    DVD-drive
                    Card-reader
                    Blu-ray-drive
            CPU
```

Detecting and checking used language levels

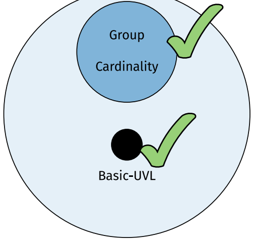
Parsing and Printing

```
include
    SAT-level.group-cardinality
    SMT-level.feature-cardinality

features
    Computer
        optional
            RAM-module cardinality [1..4]
        mandatory
            SATA-Devices
                [0..3]
                    HDD
                    SSD
                    DVD-drive
                    Card-reader
                    Blu-ray-drive
            CPU
```

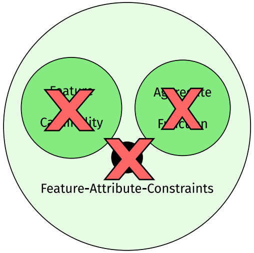

Parsing
Printing
FeatureModel
namespace: String
featureMap: Map<String, Feature>
...
+ decomposedModelToString(): Map<String, String>
+ composedModelToString(): String
...

- Same interface for basic functionality as initial UVL library
- New functionality (e.g. printing composed feature models)
- Further improvements (e.g. less restrictive feature names)

Applying Conversion Strategies



Or



- Set supported or unsupported language levels
- Convert or just remove language levels
- Automatic transitive conversion
- Correct order for conversion strategy application