Second Tutorial on the Universal Variability Language

Sebastian Krieter, **Kevin Feichtinger**, José A. Galindo, David Benavides, Rick Rabiser, Chico Sundermann, **Thomas Thüm** | 28.08.2023

universität ulm

JOHANNES KEPLER UNIVERSITÄT LINZ

University of Seville

General Introduction

Software and Materials

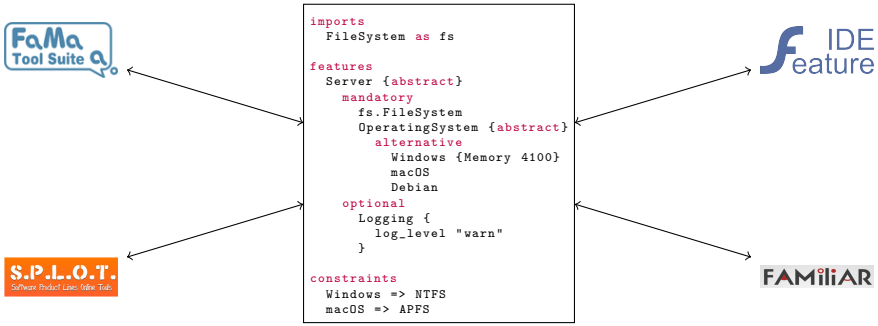
https://github.com/
Universal-Variability-Language/
tutorial-splc-2023

- All **links to the required software and material for the day** in this repository
- **Start downloading now** ⇒ Less idle time later
- QR code also on handout

MODEVAR Initiative

- Unify community behind single variability language (`https://modevar.github.io/`)
- Workshops at SPLC'19,'20,'21,'22 and VaMoS'20, VaMoS'23
- 29 submissions so far

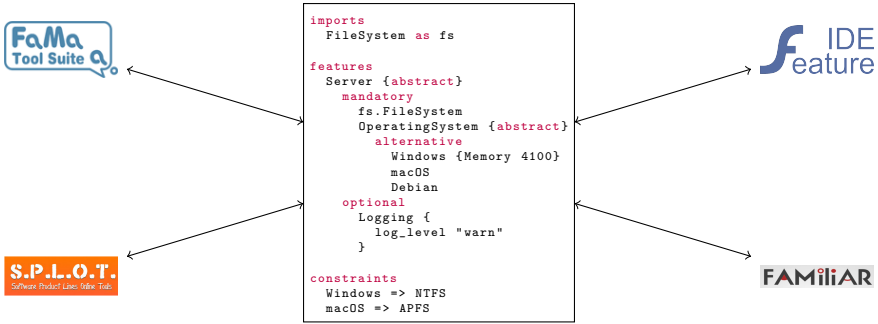
The Goal



MODEVAR Initiative

- Unify community behind single variability language (`https://modevar.github.io/`)
- Workshops at SPLC'19,'20,'21,'22 and VaMoS'20, VaMoS'23
- 29 submissions so far

Universal **V**ariability **L**anguage?



The Vision with UVL

- Community effort
- Unified language for variability models

```
features
  UVL {abstract}
    mandatory
      Structure
        mandatory
          FeatureGroups
            alternative
              "Or"
              "Alternative"
              "Mandatory"
              "Optional"
          "Propositional Constraints"
        optional
          "Feature Attributes"
          "Model Composition"
      "Design Goals"
        mandatory
          "Human Readability"
          Exchange
          "Teaching and Learning"

constraints
  "Human Readability" => "Model Composition"
  Exchange => "Feature Attributes"
```

The Vision with UVL

- Community effort
- Unified language for variability models
- Simplify exchange
- Convenient integration into existing tools
- Broad tool support
  - Parsing
  - Editing
  - Transforming
  - Analyzing

```
features
  UVL {abstract}
    mandatory
      Structure
        mandatory
          FeatureGroups
            alternative
              "Or"
              "Alternative"
              "Mandatory"
              "Optional"
          "Propositional Constraints"
        optional
          "Feature Attributes"
          "Model Composition"
      "Design Goals"
        mandatory
          "Human Readability"
          Exchange
          "Teaching and Learning"

constraints
  "Human Readability" => "Model Composition"
  Exchange => "Feature Attributes"
```


Introduce yourself in 12 seconds!


Introduce yourself in 12 seconds!
What are your
name, affiliation, and main research interests?

What is gonna happen today?

universität ulm

- **Session 1**: UVL Basics
  - Textual editing with our LSP
  - Language levels

What is gonna happen today?

universität ulm

- **Session 1**: UVL Basics
  - Textual editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE

What is gonna happen today?

- **Session 1**: UVL Basics
  - Textual editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE
- **Session 3**: Transforming UVL Models
  - Transformations using TRAVART
  - Various variability model formats

universität ulm

JKU
JOHANNES KEPLER
UNIVERSITÄT LINZ

What is gonna happen today?

- **Session 1**: UVL Basics
  - Textual editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE
- **Session 3**: Transforming UVL Models
  - Transformations using TRAVART
  - Various variability model formats
- (**Session 4**: Analyzing UVL Models with FLAMA)

universität ulm

JOHANNES KEPLER
UNIVERSITÄT LINZ

Almost Starting

https://github.com/
Universal-Variability-Language/
tutorial-splc-2023

- All **links to the required software and material for the day** in this repository
- You will need the following soon:
  1. z3
  2. VSCode
     3. Extension: UVLS
     4. Extension: GraphViz Interactive Viewer

Session 1: UVL Basics

The Universal Variabiliy Language

- **Feature tree**: Parent-child relationships
- **Constraints**: Expressions on features and attributes
- **Attributes**: {`Key Value`}
- Language levels

```
imports
  FileSystem as fs

features
  Server {abstract}
    mandatory
      fs.FileSystem
      OperatingSystem {abstract}
        alternative
          Windows {Memory 4100}
          macOS
          Debian
    optional
      Logging {
        log_level "warn"
      }

constraints
  Windows => NTFS
  macOS => APFS
```

Try UVL yourself with UVLS

https://github.com/
Universal-Variability-Language/
tutorial-splc-2023

- See **How to get started** in Session 1
- Important Steps
  1. Install z3
  2. Download VScode
  3. Install UVLS in VSCode marketplace
  4. Create and edit UVL file together

Session 2: Modeling with UVL

Session 2: Modeling with UVL

https://github.com/
Universal-Variability-Language/
tutorial-splc-2023

- See **How to get started** in *Session 2*
- Important Steps
  1. Download FeatureIDE prepackage (v3.10.0)
  2. Extract
  3. Start eclipse
  4. Create feature-modeling project
  5. Work with your previously created UVL model

Session 3: Transforming UVL Models

Session 4: Analyzing UVL Models with FLAMA

Wrap-Up

Wrap-Up What happened?

- **Session 1**: UVL Basics
  - Textual Editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE
- **Session 3**: Transforming UVL Models
  - Transformations using TRAVART
  - Various variability model formats

TRAVART

Wrap-Up What happened?

- **Session 1**: UVL Basics
  - Textual Editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE
- **Session 3**: Transforming UVL Models
  - Transformations using TRAVART
  - Various variability model formats

TRAVART

Wrap-Up What happened?

- **Session 1**: UVL Basics
  - Textual Editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE
- **Session 3**: Transforming UVL Models
  - Transformations using TRAVART
  - Various variability model formats

TRAVART

Wrap-Up What happened?

- **Session 1**: UVL Basics
  - Textual Editing with our LSP
  - Language levels
- **Session 2**: Modelling with UVL
  - Java parser library
  - Modelling UVL in FeatureIDE
- **Session 3**: Transforming UVL Models
  - Transformations using TRAVART
  - Various variability model formats
- (**Session 4**: Analyzing UVL Models with FLAMA)

TRAVART

Wrap-Up Questions & Feedback

- Community project

https://universal-variability-language.github.io/

Wrap-Up Questions & Feedback

- Community project
- What to tackle in ...

https://universal-variability-language.github.io/

Wrap-Up Questions & Feedback

- Community project
- What to tackle in ...
  - language design?

https://universal-variability-language.github.io/

Wrap-Up Questions & Feedback

- Community project
- What to tackle in ...
  - language design?
  - tool support?

https://universal-variability-language.github.io/

Wrap-Up Questions & Feedback

- Community project
- What to tackle in ...
  - language design?
  - tool support?
  - other directions?

https://universal-variability-language.github.io/