Second Tutorial on the Universal Variability Language

Handout

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

August 27, 2023

1 Material

During the tutorial, we need different software which may require some time to download and install. To have a smoother experience during the tutorial, it would be great to start preparing the respective tools during the introduction. The QR code leads to GitHub repository[a] that provides links to find the required software.

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

2 Sessions

2.1 Session 1: Textual Editing with UVLS

UVLS is an LSP that enables syntax highlighting, autocompletion, semantic analyses, and configuration for UVL models. In the tutorial, we will use the LSP in VSCode.

How to get started

1. **Install z3**: Find instructions at[1].
2. **Install VSCode**
3. **Install UVLS** from internal marketplace
   a) Alternative: Extension Tab > ... > Install from VSIX [2].

[1] https://github.com/Universal-Variability-Language/uvl-lsp
[2] https://marketplace.visualstudio.com/items?itemName=caradhras.uvls-code

b) Arch: Install code-marketplace (AUR) to enable extensions in VSCode

4. **Accept the automatic download** for the LSP.
5. You can now edit .uvl files with the editing support of UVLS.

2.2 Session 2: Modeling UVL with FeatureIDE

FeatureIDE is a widely used environment integrated in Eclipse for feature-oriented software development. With FeatureIDE, UVLS can be used for various parts of product-line engineering.

Instead of using the UVL model created in the first session of the tutorial, you can also continue with this UVL model[3].

How to get started

1. **Download pre-packaged Eclipse** with FeatureIDE (v.3.10.0)[4]
2. **Extract Eclipse** and run it.
3. Create a new feature modeling project via New → FeatureIDE Project
4. Add your UVL model to this project

2.3 Session 3: Transforming UVL models with TraVarT

TraVarT enables the transformation between different variability artifacts. In the latest version, TraVarT provides a plugin infrastrucutre, which enables supporting different types of artifacts at the same time and an extension with your own custom approach.

- **Download TraVarT** [5]

[3] `https://github.com/Universal-Variability-Language/tutorial-splc-2023/blob/main/models/icecream-shared.uvl`
[4] `https://featureide.github.io/`
[5] https://github.com/SECPS/TraVarT