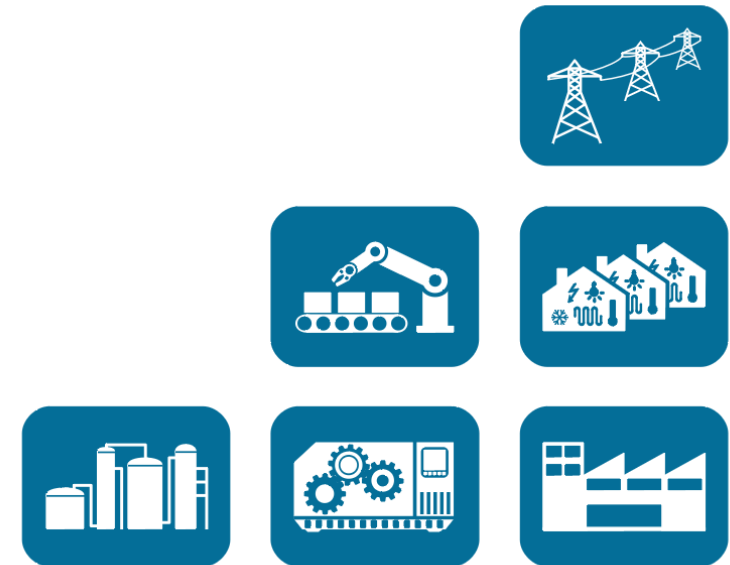


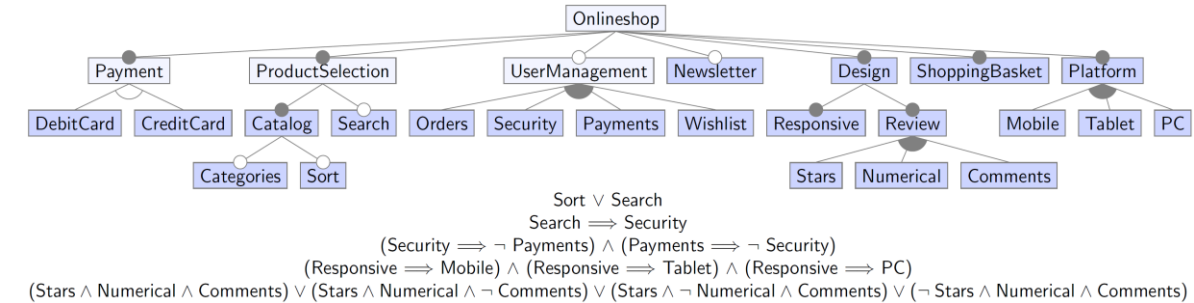


Tutorial on the Universal Variability Language – Session 3



Variability Modeling Approaches

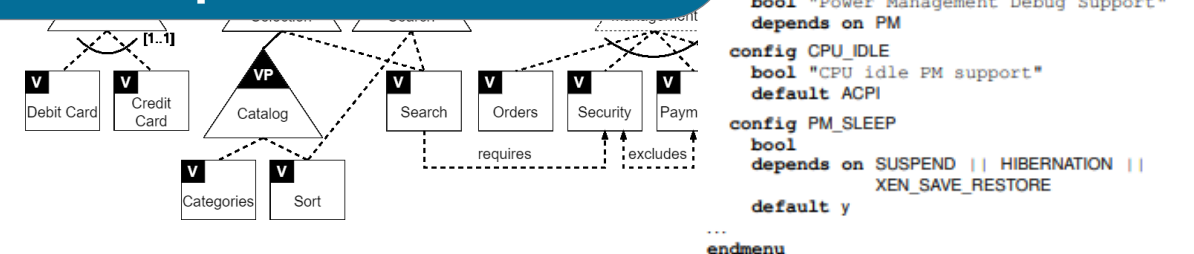
- Feature modeling (FeatureIDE, UVL, Clafer,...)
- Decision modeling (DOPLER, IVML,...)
- Orthogonal Variability modelling (OVM)
- UML-based variability modelling
- Delta-oriented
- Textual variability
- Common Variability
- Kconfig (Linux)
- Component Definition Language (eCos)



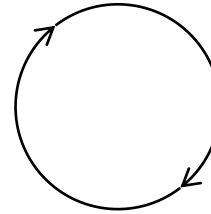
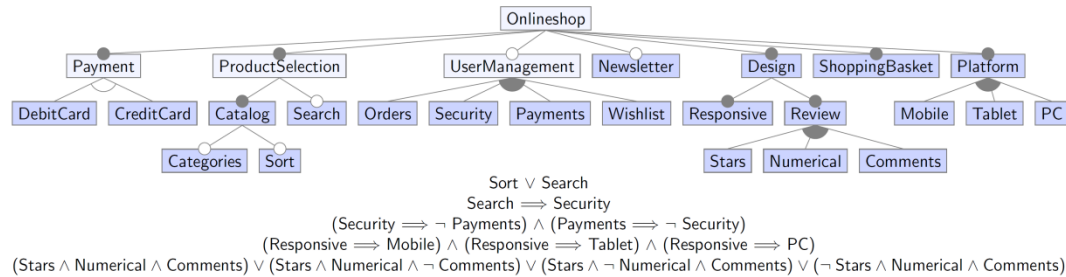
ID	Question	Type	Range	Card.	Constraint/Rule	Visibility
Payment	Which payment methods should be supported?	Enum	DebitCard CreditCard	1:1		
Search	Should a search function be supported?	Boolean	true false		if (Search) { UserManagement = Security } if (!Search) { Sort = true }	
Categories	Split products into categories?	Boolean	true false		if (!Sort) { Search = true }	
					if (Security) { disAllow(Payments) } if (!Security) { allow(Payments) } if (Payments) { disAllow(Security) } if (!Payments) { allow(Security) }	
				1:4		
				2:3		

How can researchers and practitioners better understand the strengths and weaknesses of an approach?

How can we support researchers and practitioners in picking the right approach for their specific use case?



Variability Model Transformations

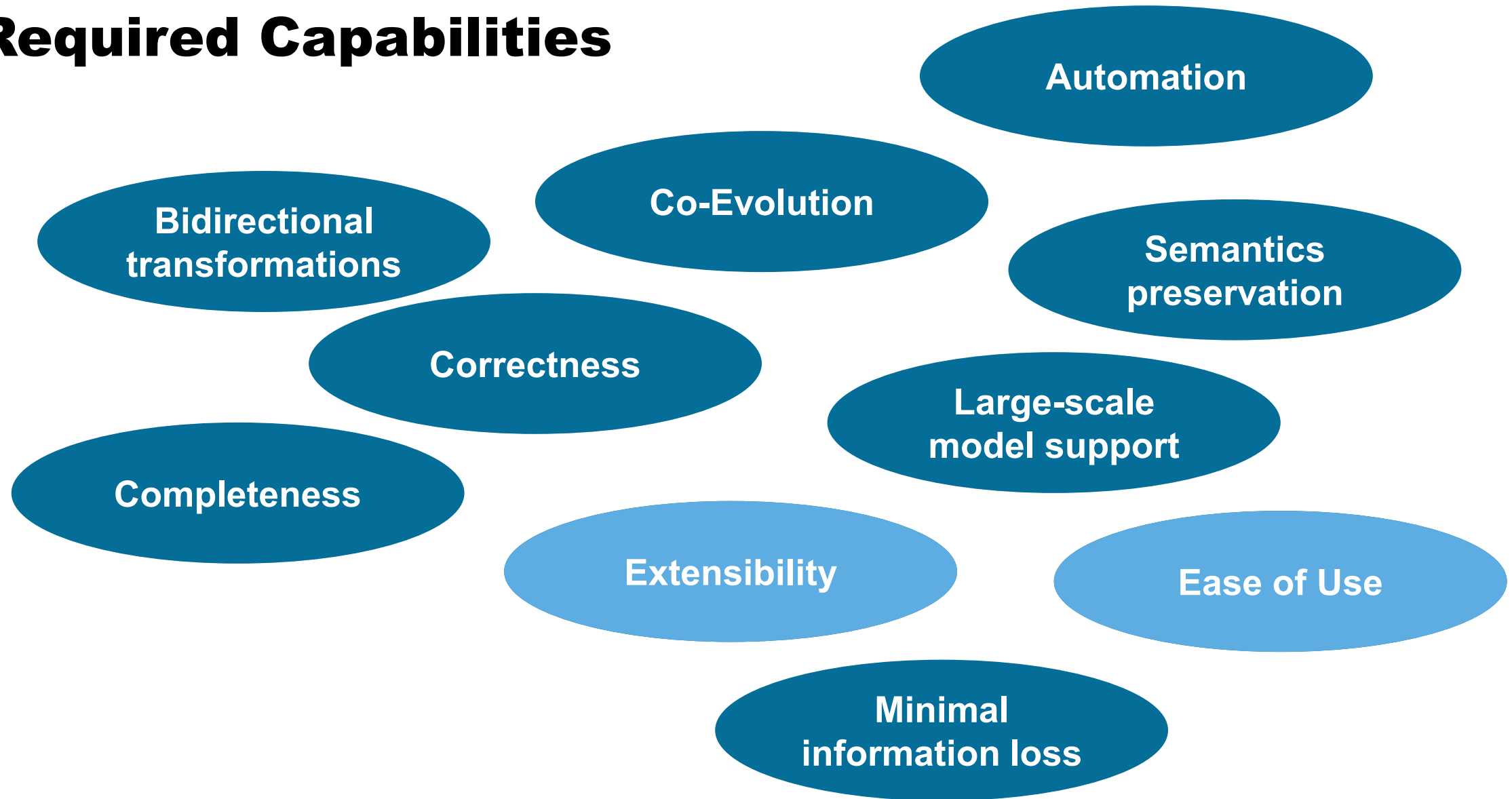


ID	Question	Type	Range	Card.	Constraint/Rule	Visibility
Payment	Which payment methods should be supported?	Enum	DebitCard CreditCard	1:1		
Search	Should a search function be supported?	Boolean	true false		if (Search) { UserManagement = Security }	
Categories	Split products into categories?	Boolean	true false		if (!Search) { Sort = true }	
Sort	Should products be sortable?	Boolean	true false		if (!Sort) { Search = true }	
UserManagement	Which user management options should be added?	Enum	Orders Security Payments Wishlist	1:4	if (Security) { disAllow(Payments) }	
Newsletter_percent	How many percent off?	Number	0 - 100		if (!Security) { allow(Payments) }	
Newsletter_mail	From which mail should the newsletter be sent?	String			if (Payments) { disAllow(Security) }	
Review	Which review techniques should be supported?	Enum	Stars Numerical Comments	2:3	if (!Payments) { allow(Security) }	



- (1) Switching to a different approach without losing invested modeling efforts
- (2) Experimenting with different approaches before selecting one
- (3) Integrating tools of other approaches, e.g., for analysis
- (4) Transforming custom-developed variability artifacts into well-knowns

Required Capabilities



WebTraVarT



TraVarT Online

Convert model file

Please only upload allowed file format

Files will be deleted once you leave the site.

Upload model...

Drop model file here.

WikiMatrix.xml

TraVarT Online

Files will be deleted once you leave the site. Applies also to generated files.

Upload model...

Drop model file here.

WikiMatrix.xml

Original Model	Target Model			
Feature Model->Decision Model	Features	Constraints	Decisions	Rules
			26	76

wikimatrix.csv (5 Kb)

Cyber-Physical Systems Lab

GitHub

Example models

Johannes Kepler Universität Linz - LIT CPS Lab

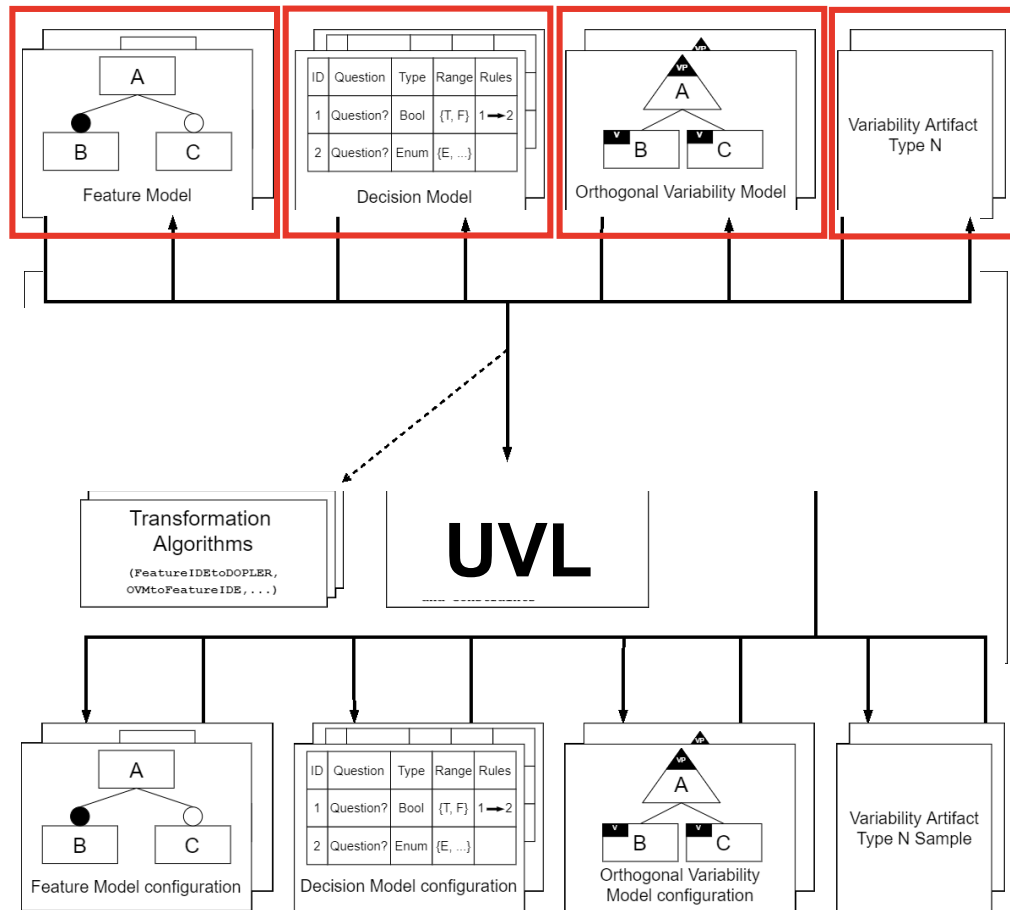
Altenbergerstraße 69

4040 Linz, Österreich

Extensibility?

What if I am not online or can/should not upload my artifacts to third party?

TRAVART: Transforming Variability Artifacts



```
Usage: travart [-hvv] [COMMAND]
TraVarT main command to transform and validate variability artifacts.
-h, --help      Show this help message and exit.
-v, --verbose   Enable verbose log information during execution.
-V, --version   Print version information and exit.

Commands:
transform       Transforms the given variability artifacts into another type.
plugin          Lists the available plugins for the TraVarT command.
```

Github: <https://github.com/SECPS/TraVarT> - Plugin Github: <https://github.com/SECPS> - WebTraVarT: <https://litcps.jku.at/travart/>

Practical Part I



Export TRAVART zip and transform some artifacts

Implementing a TRAVART Plugin

```
24 public interface IPlugin<T> extends ExtensionPoint {
25
26     /**
27      * Returns the transformer of the plugin to transform the variability model.
28      *
29      * @return the transformer of the plugin to transform the variability model.
30      */
31     IModelTransformer<T> getTransformer();
32
33     /**
34      * Returns the reader of the plugin to read the variability model from the file
35      * system.
36      *
37      * @return the reader of the plugin to read the variability model from the file
38      * system.
39      */
40     IReader<T> getReader();
41
42     /**
43      * Returns the statistics of the plugin to get the statistics the variability
44      * model.
45      *
46      * @return the statistics of the plugin to get the statistics the variability
47      * model.
48      */
49     IStatistics<T> getStatistics();
50
51     /**
52      * Returns the writer of the plugin to write the variability model to the file
53      * system.
54      *
55      * @return the writer of the plugin to write the variability model to the file
56      * system.
57      */
58     IWriter<T> getWriter();
```

```
60     /**
61      * Returns the variability model type name.
62      *
63      * @return the name of the variability model type.
64      */
65     String getName();
66
67     /**
68      * Returns the version of the plugin.
69      *
70      * @return the version of the plugin.
71      */
72     String getVersion();
73
74     /**
75      * Returns a unique ID of the plugin, such that it can be identified.
76      *
77      * @return the unique ID of the plugin.
78      */
79     String getId();
80
81     /**
82      * Returns a iterable of file extensions for which this plugin is applicable.
83      *
84      * @return a unmodifiable list of file extensions.
85      */
86     Iterable<String> getSupportedFileExtensions();
87 }
```


Practical Part III



Practical Example FeatureIDE

Future Work

- Make today's elements (cli and plugin extension support) publicly available
 - <https://github.com/SECPS>
- Make further plugins publicly available
- Extend CLI commands
 - Validating and evaluating the created variability artifacts
 - Option for one-way or roundtrip focused transformation
 - Option to log statistical information
 - Option to create different types of artifacts at once

What would you need from an CLI to enable your use case?