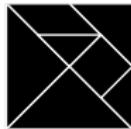


Software Product Lines in Clafer



Kacper Bak

Generative Software Development Lab
University of Waterloo

PhD Seminar. December 14, 2010

Acknowledgments

Krzysztof Czarnecki (UW)

Michał Antkiewicz (UW)

Andrzej Wasowski (ITU)

Software Intensive



Products Come in



Many Variants



iPod shuffle

iPod nano

iPod classic

iPod touch

iPhone

Software Product Line

Software Product Line

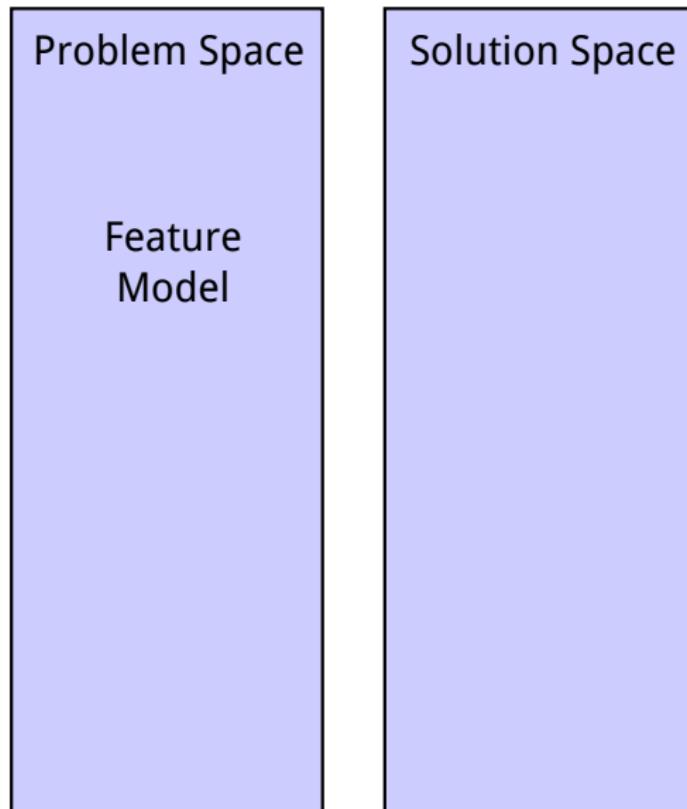
Problem Space

Software Product Line

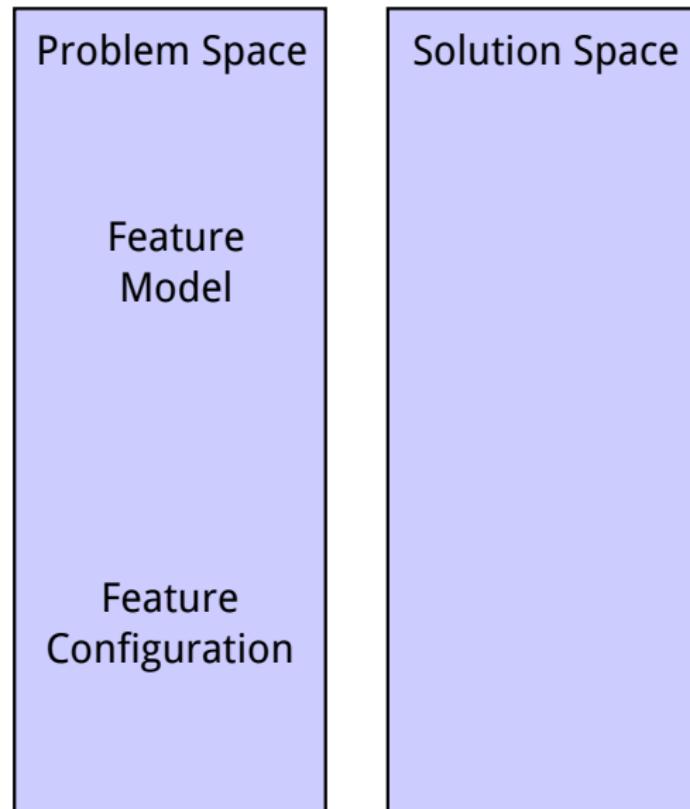
Problem Space

Solution Space

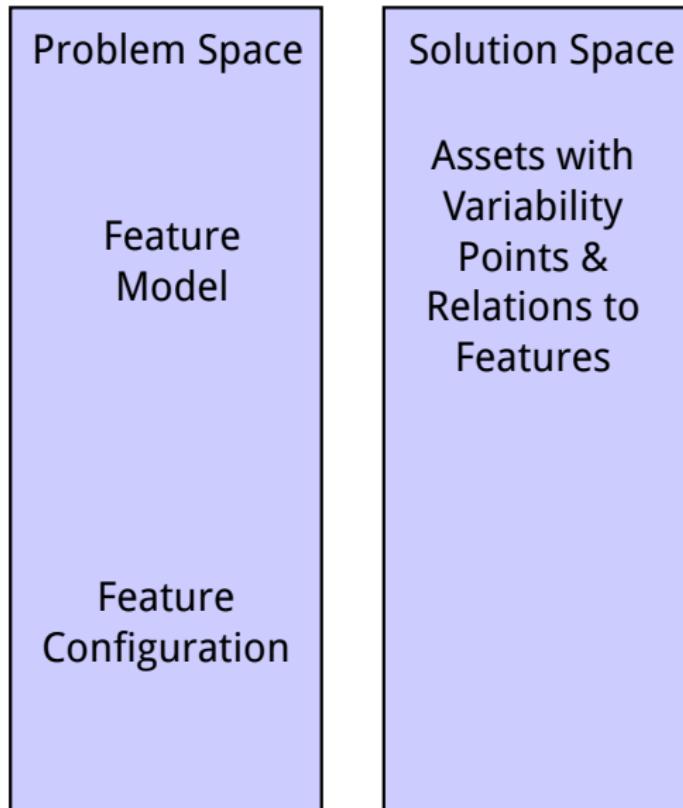
Software Product Line



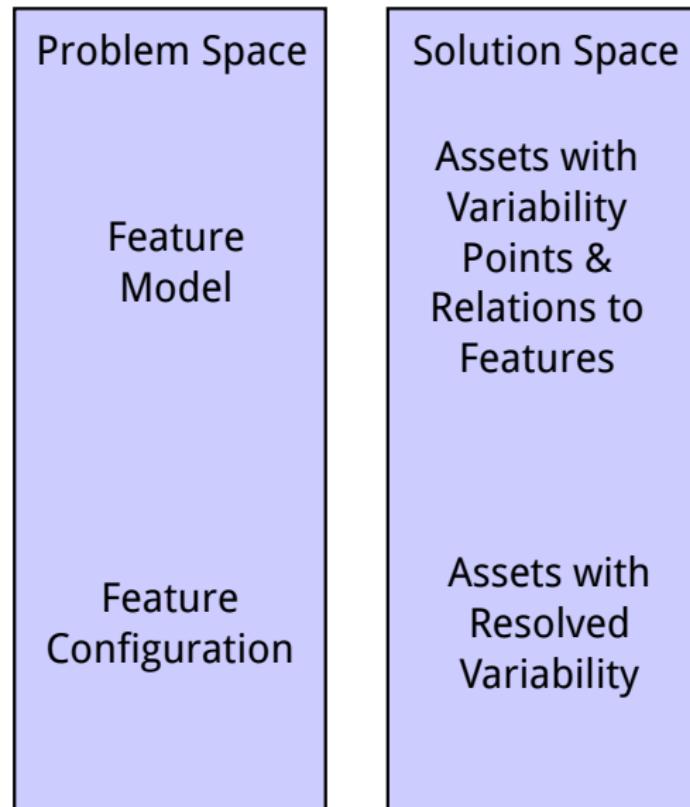
Software Product Line



Software Product Line



Software Product Line



Software Product Line

	Problem Space	Solution Space
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features
	Feature Configuration	Assets with Resolved Variability

Software Product Line

	Problem Space	Solution Space
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features
Application Engineering	Feature Configuration	Assets with Resolved Variability

Clafer

Modeling language for Software Product Lines
Analyses

Concept modeling

Clafer

Modeling language for Software Product Lines
Analyses
Concept modeling

Clafer Goals

Minimalistic language

Concise notation for feature and meta-models

Mixing feature and meta-models

Uniform semantics

Clafer Goals

Minimalistic language

Concise notation for feature and meta-models

Mixing feature and meta-models

Uniform semantics

Clafer Goals

Minimalistic language

Concise notation for feature and meta-models

Mixing feature and meta-models

Uniform semantics

Clafer Goals

Minimalistic language

Concise notation for feature and meta-models

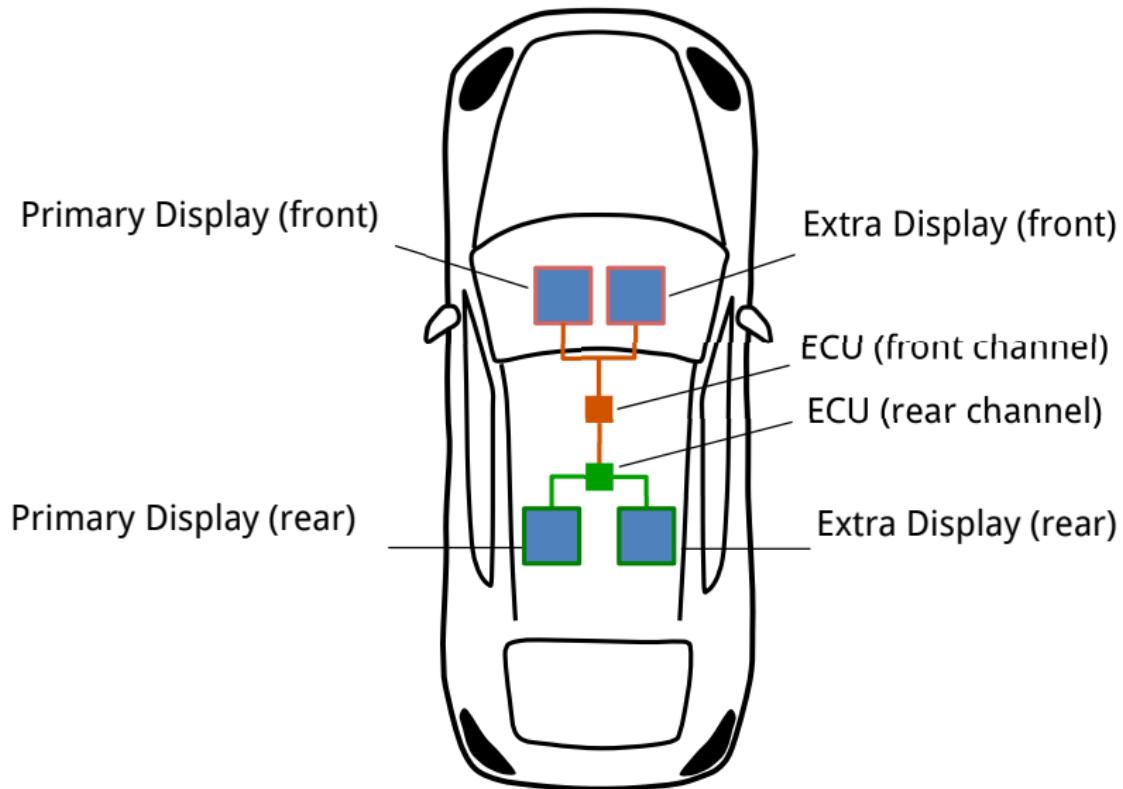
Mixing feature and meta-models

Uniform semantics

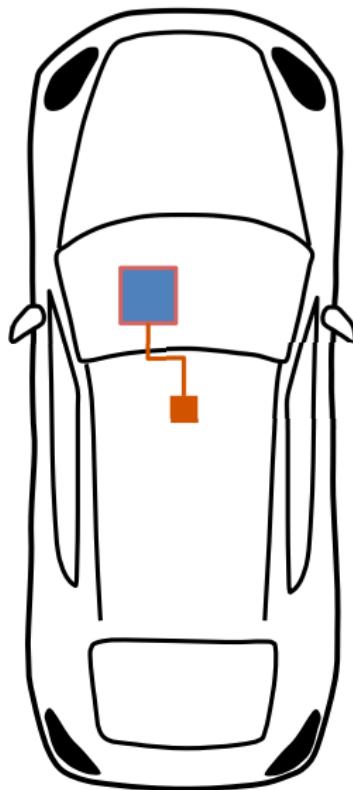
SPL: Telematics System



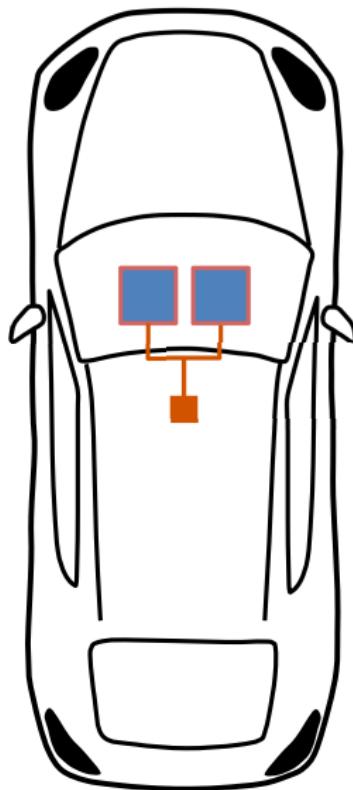
Telematics System: Elements



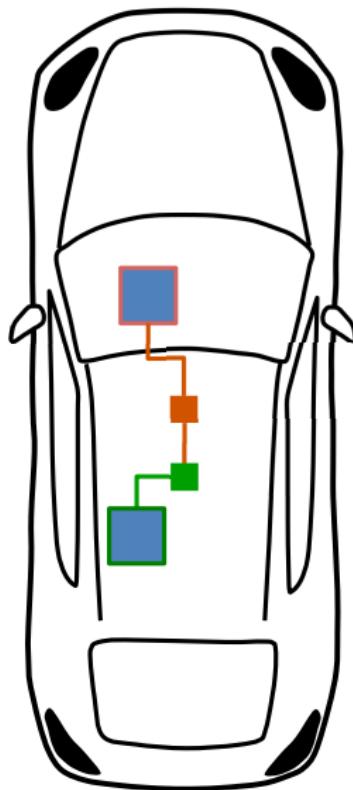
Telematics System: Configurations



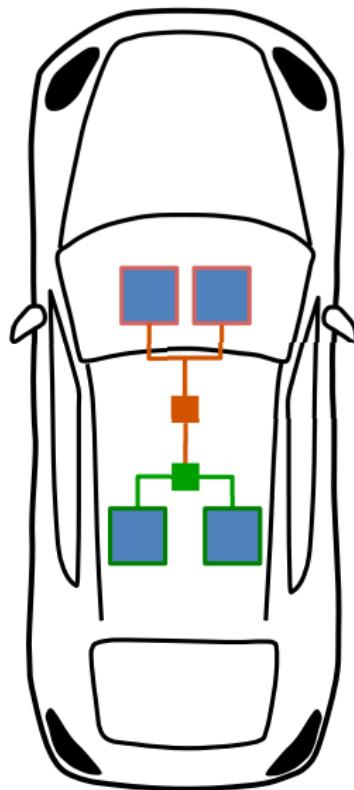
Telematics System: Configurations



Telematics System: Configurations



Telematics System: Configurations

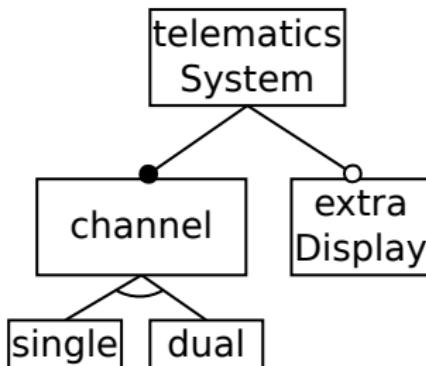


Telematics in Clafer

Feature Model

	Problem Space	Solution Space	
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features	
Application Engineering	Feature Configuration	Assets with Resolved Variability	

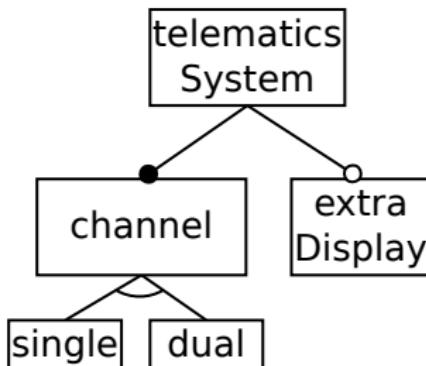
Feature Model



Configuration Semantics: Set of Configurations

- { telematicsSystem, channel, **single** },
- { telematicsSystem, channel, **single**, extraDisplay },
- { telematicsSystem, channel, **dual** },
- { telematicsSystem, channel, **dual**, extraDisplay } }

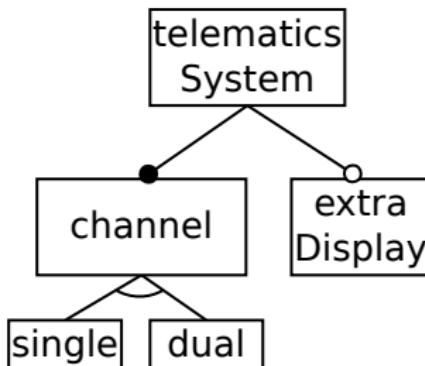
Feature Model



Configuration Semantics: Set of Configurations

- { telematicsSystem, channel, **single** },
- { telematicsSystem, channel, **single**, extraDisplay },
- { telematicsSystem, channel, **dual** },
- { telematicsSystem, channel, **dual**, extraDisplay } }

Feature Model



Configuration Semantics: Set of Configurations

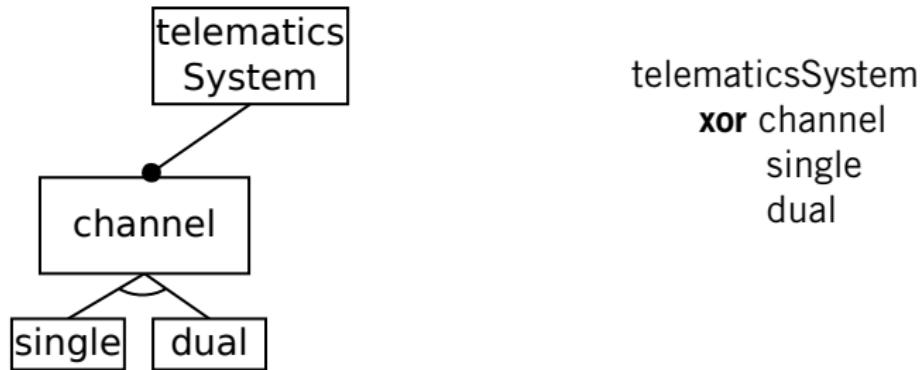
- { telematicsSystem, channel, **single** },
- { telematicsSystem, channel, **single**, **extraDisplay** },
- { telematicsSystem, channel, **dual** },
- { telematicsSystem, channel, **dual**, **extraDisplay** } }

Feature Model

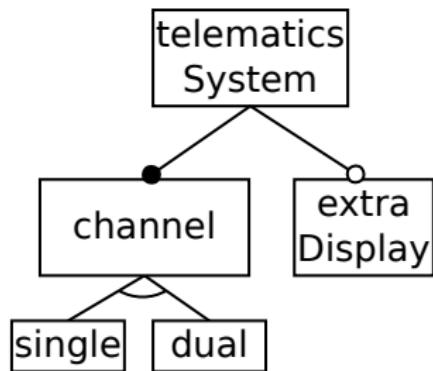
telematics
System

telematicsSystem

Feature Model



Feature Model



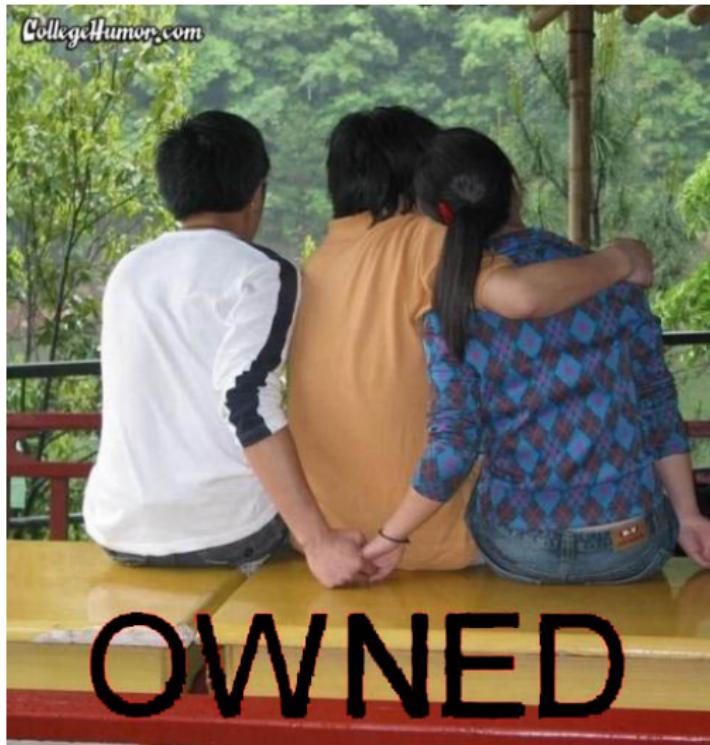
telematicsSystem
xor channel
single
dual

extraDisplay ?

Assets

	Problem Space	Solution Space
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features
Application Engineering	Feature Configuration	Assets with Resolved Variability

Options

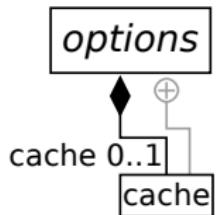


Display Options

options

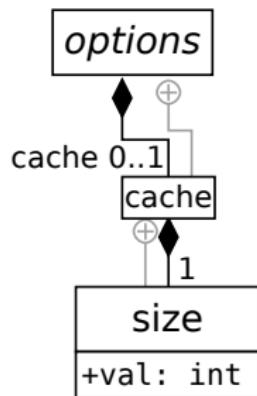
abstract options

Display Options



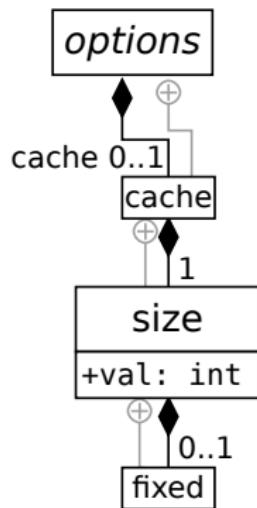
abstract options
cache ?

Display Options



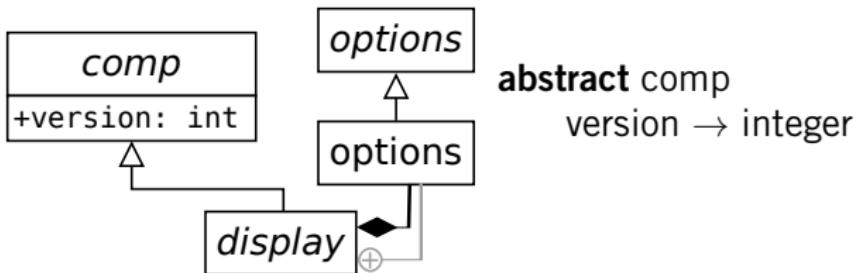
abstract options
cache ?
size → integer

Display Options

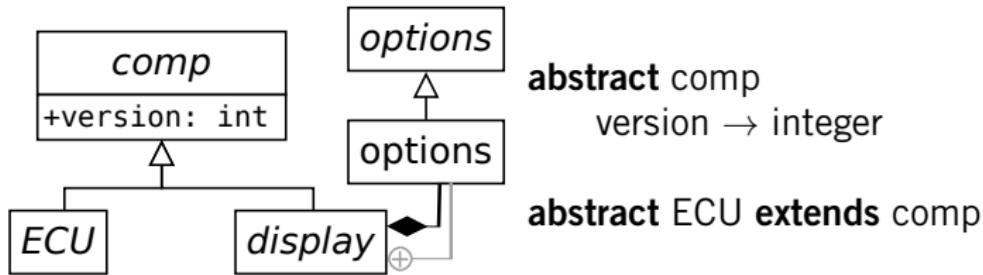


abstract options
cache ?
size → integer
fixed ?

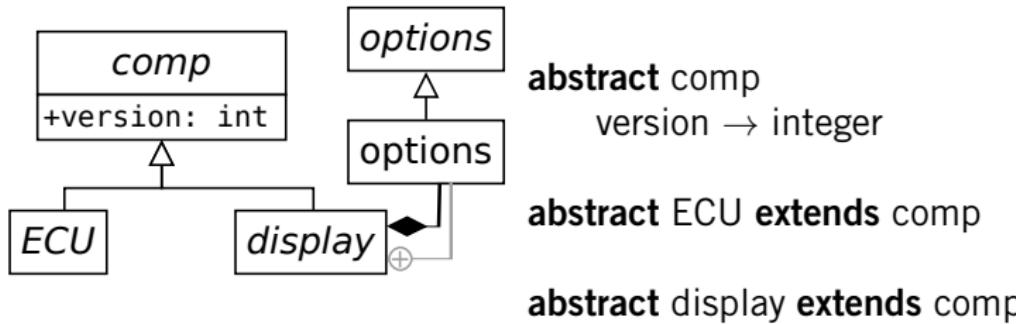
Mixed: Meta-Model with Options



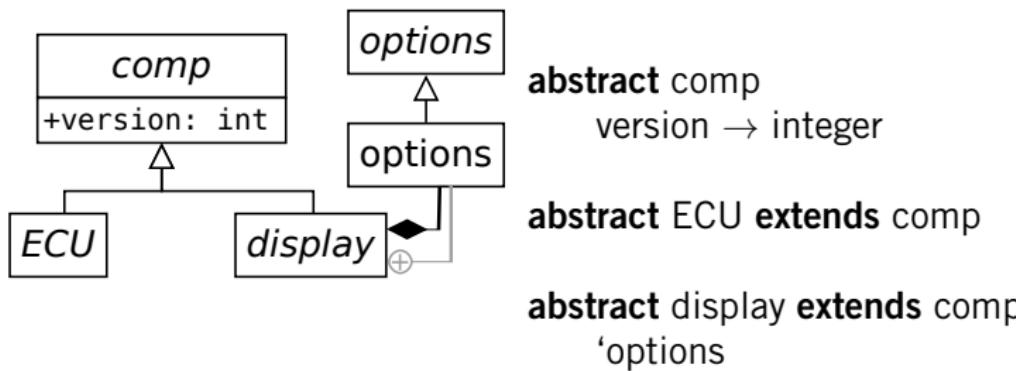
Mixed: Meta-Model with Options



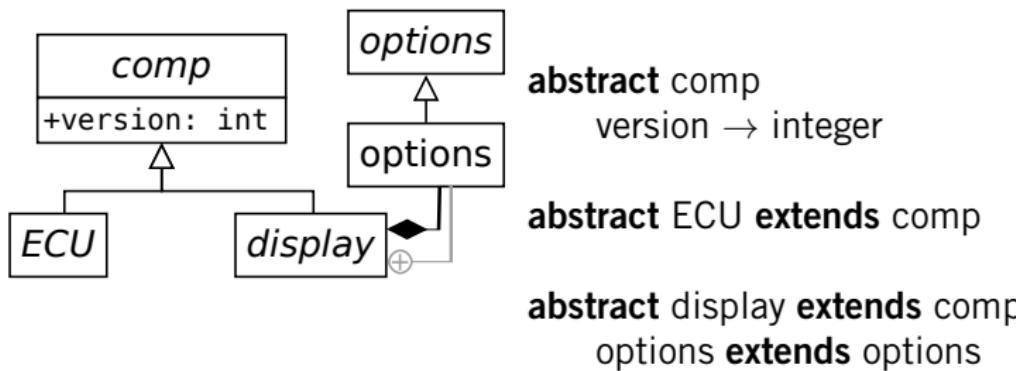
Mixed: Meta-Model with Options



Mixed: Meta-Model with Options



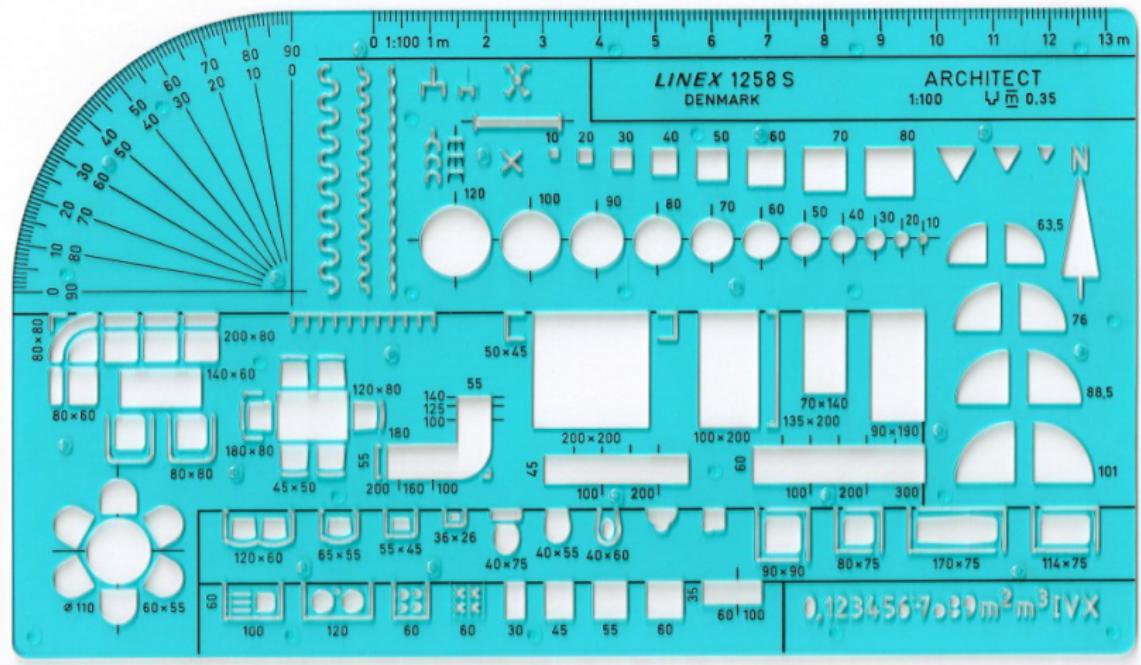
Quotation is a Syntactic Sugar



Instance sharing = references

Type sharing = quotations

Architectural Template



Specialized: Architectural Template

frontECU

frontECU : ECU

Specialized: Architectural Template

frontECU

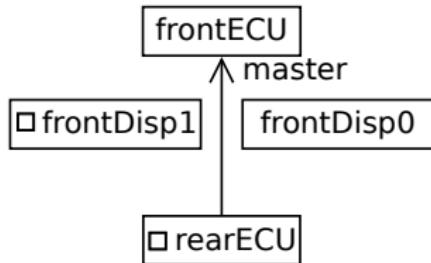
frontECU : ECU

□ frontDisp1

frontDisp0

frontDisp : display 1..2

Specialized: Architectural Template

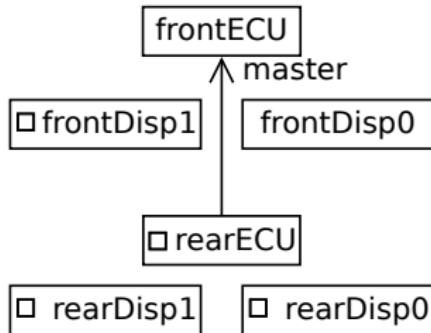


frontECU : ECU

frontDisp : display 1..2

rearECU : ECU ?
master → frontECU

Specialized: Architectural Template



frontECU : ECU

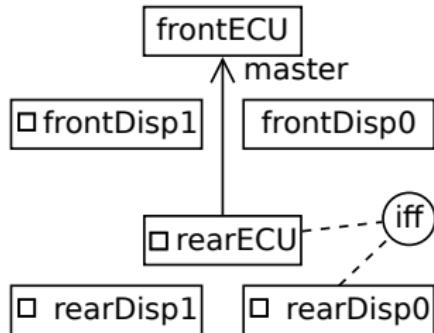
frontDisp : display 1..2

rearECU : ECU ?
master → frontECU

rearDisp : display 0..2

[**no** cache]

Specialized: Architectural Template



frontECU : ECU

frontDisp : display 1..2

rearECU : ECU ?
master → frontECU

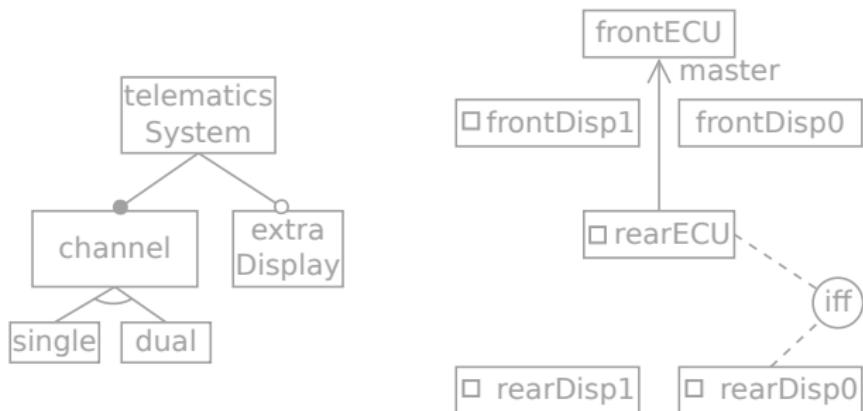
rearDisp : display 0..2

[**no** cache]
[rearECU ⇔ rearDisp]

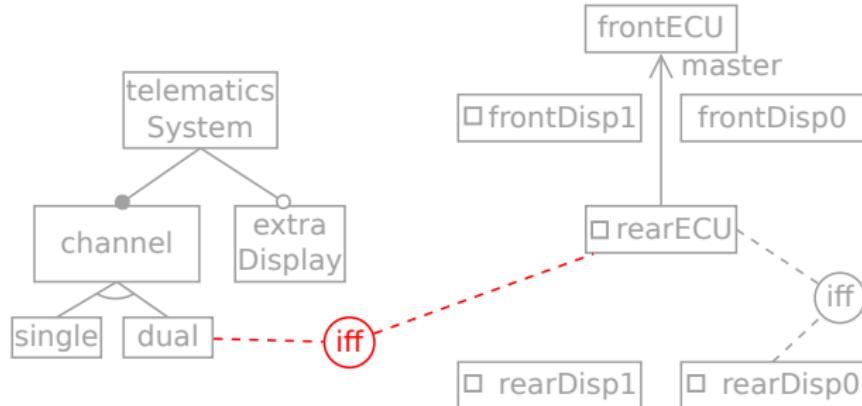
Relations to Features

	Problem Space	Solution Space
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features
Application Engineering	Feature Configuration	Assets with Resolved Variability

Coupled: Mapping Constraints

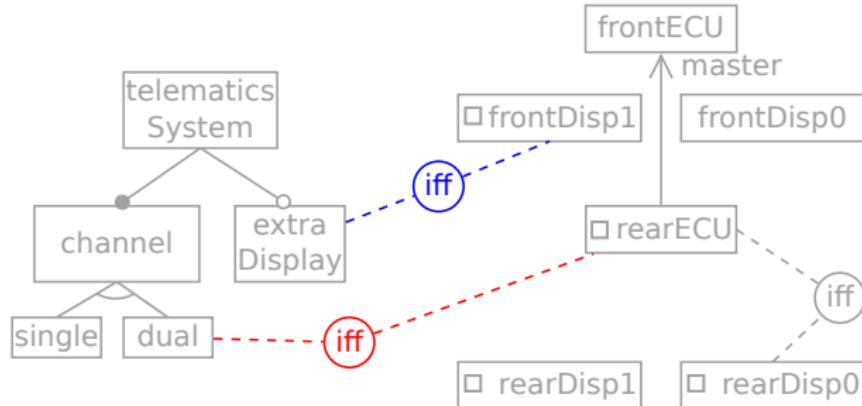


Coupled: Mapping Constraints



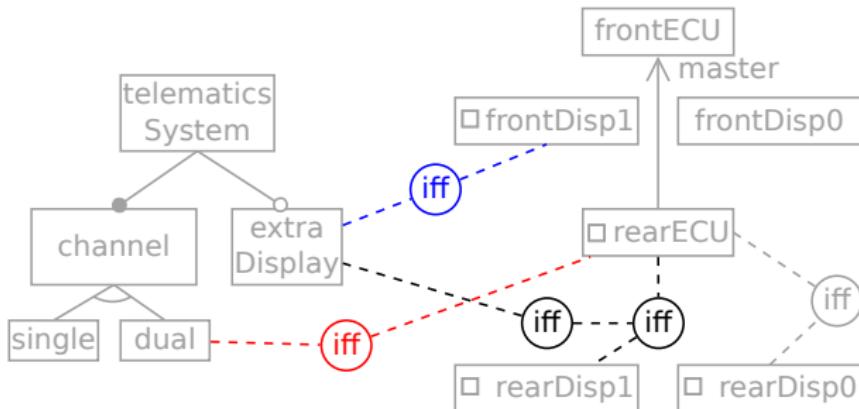
[dual \Leftrightarrow rearECU]

Coupled: Mapping Constraints



[dual \Leftrightarrow rearECU
extraDisplay \Leftrightarrow #frontDisp = 2

Coupled: Mapping Constraints



[$\text{dual} \Leftrightarrow \text{rearECU}$

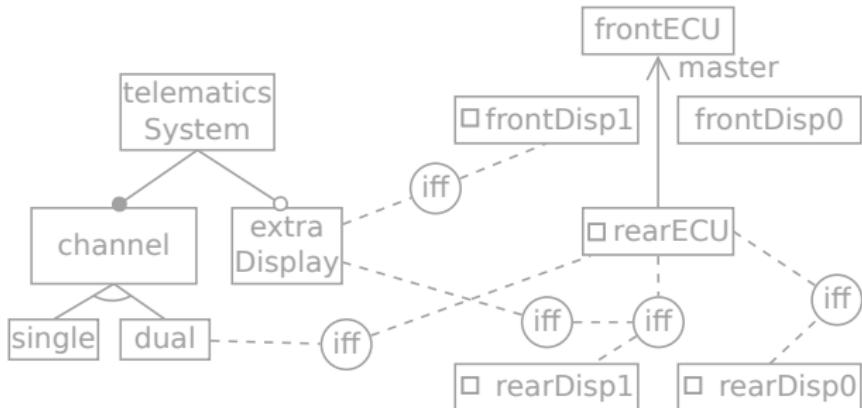
$\text{extraDisplay} \Leftrightarrow \#\text{frontDisp} = 2$

$\text{extraDisplay} \Leftrightarrow (\text{rearECU} \Leftrightarrow \#\text{rearDisp} = 2)$]

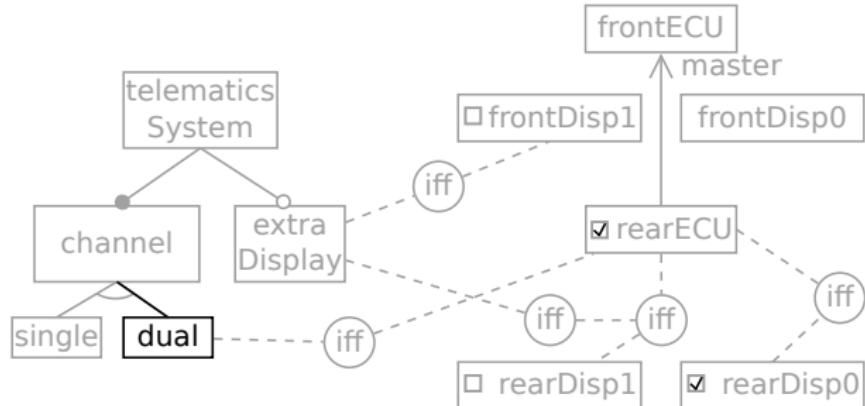
Configuration

	Problem Space	Solution Space	
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features	
Application Engineering	Feature Configuration	Assets with Resolved Variability	

Configured: Constraints

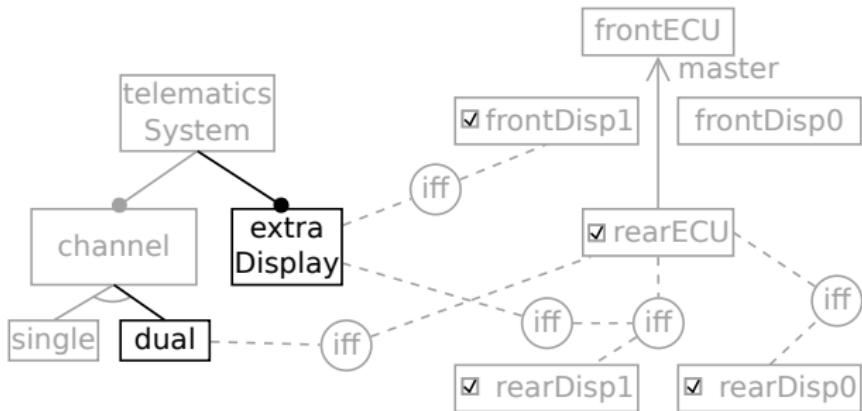


Configured: Constraints



[dual

Configured: Constraints

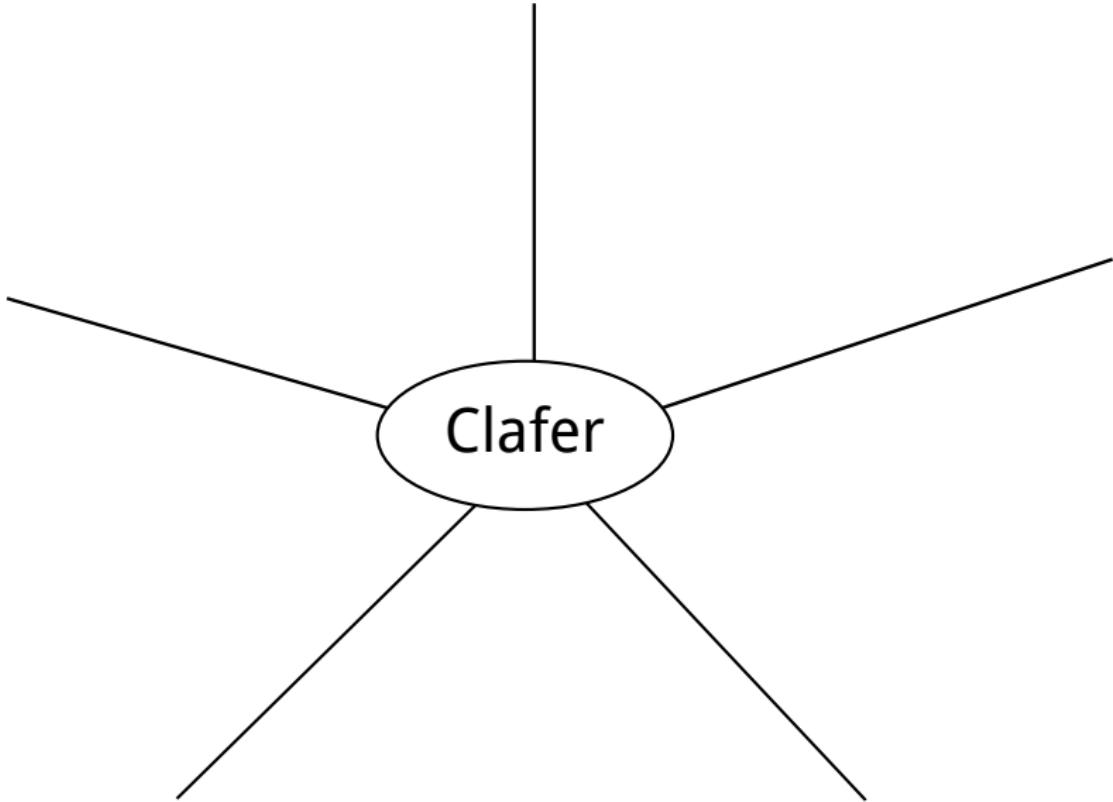


```
[dual  
extraDisplay]  
[ comp.version == 1 ]
```

Resolved Variability

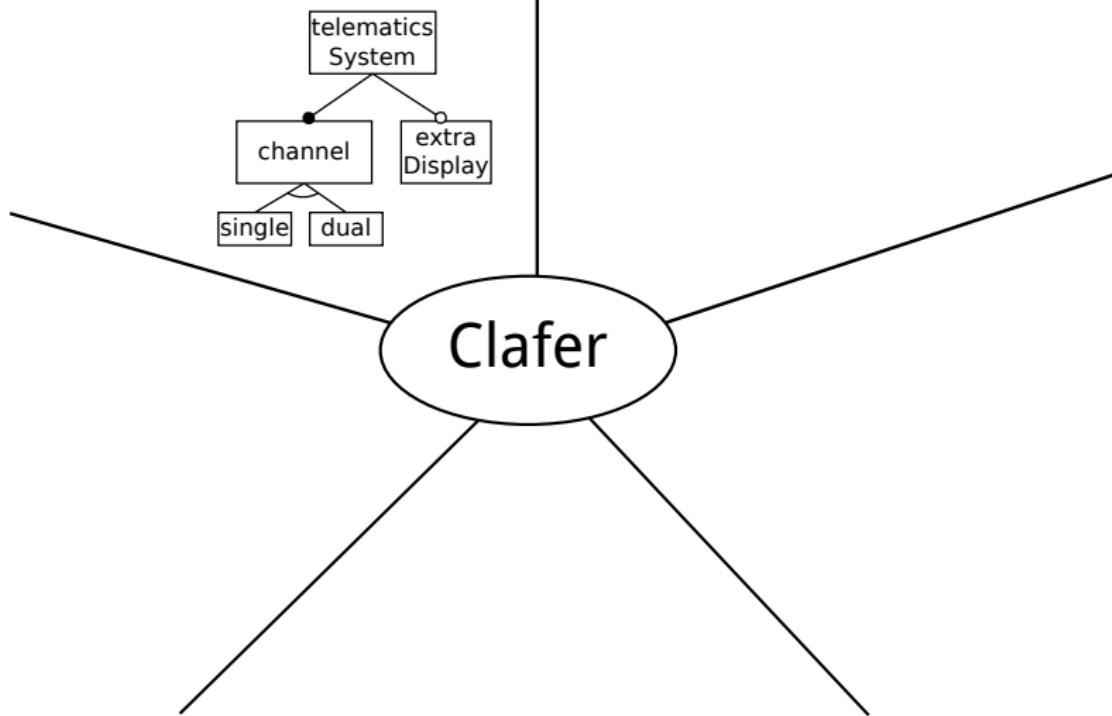
	Problem Space	Solution Space
Domain Engineering	Feature Model	Assets with Variability Points & Relations to Features
Application Engineering	Feature Configuration	Assets with Resolved Variability

SPLs in Clafer



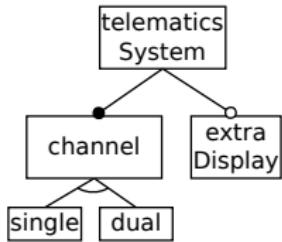
SPLs in Clafer

Feature Model

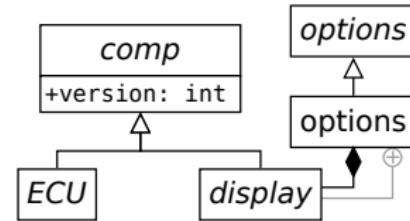


SPLs in Clafer

Feature Model



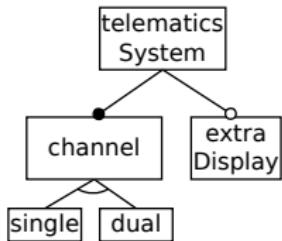
Meta-Model



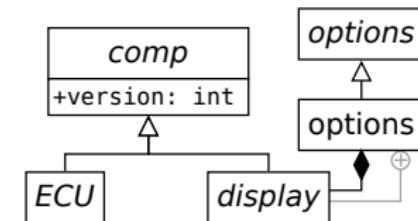
Clafer

SPLs in Clafer

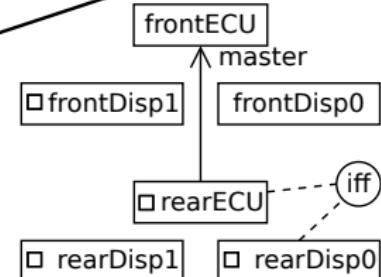
Feature Model



Meta-Model

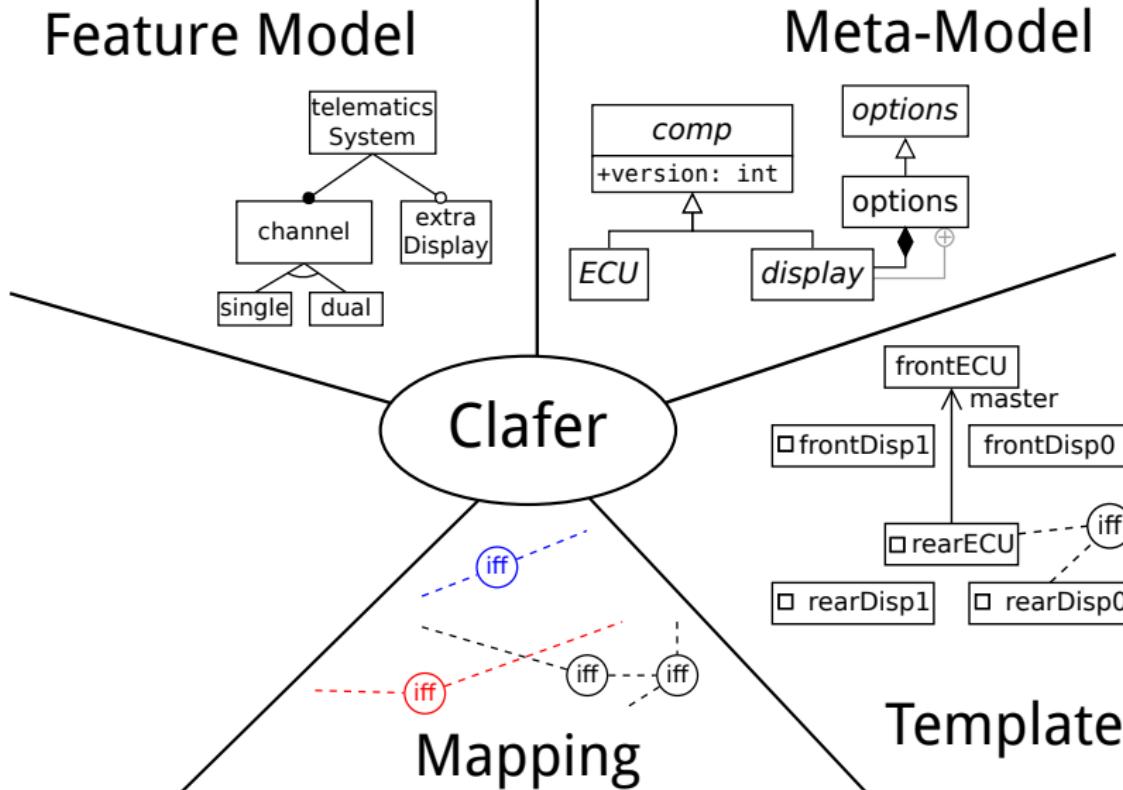


Clafer

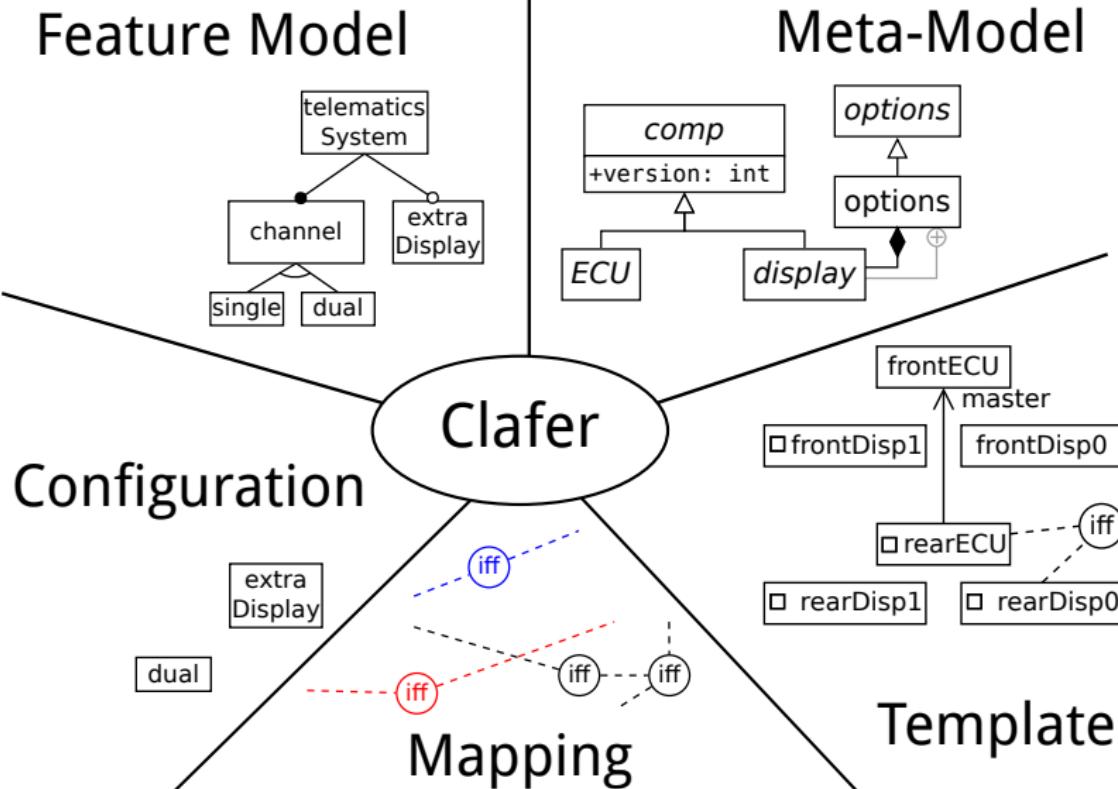


Template

SPLs in Clafer



SPLs in Clafer



Reasoning: Clafer to Alloy

Clafer to Alloy

Structural modeling

Model checking

Alloy: instantiation, testing ideas

Reasoning: Clafer to Alloy

Clafer to Alloy

Structural modeling

Model checking

Alloy: instantiation, testing ideas

Reasoning: Clafer to Alloy

Clafer to Alloy

Structural modeling

Model checking

Alloy: instantiation, testing ideas

Reasoning: Clafer to Alloy

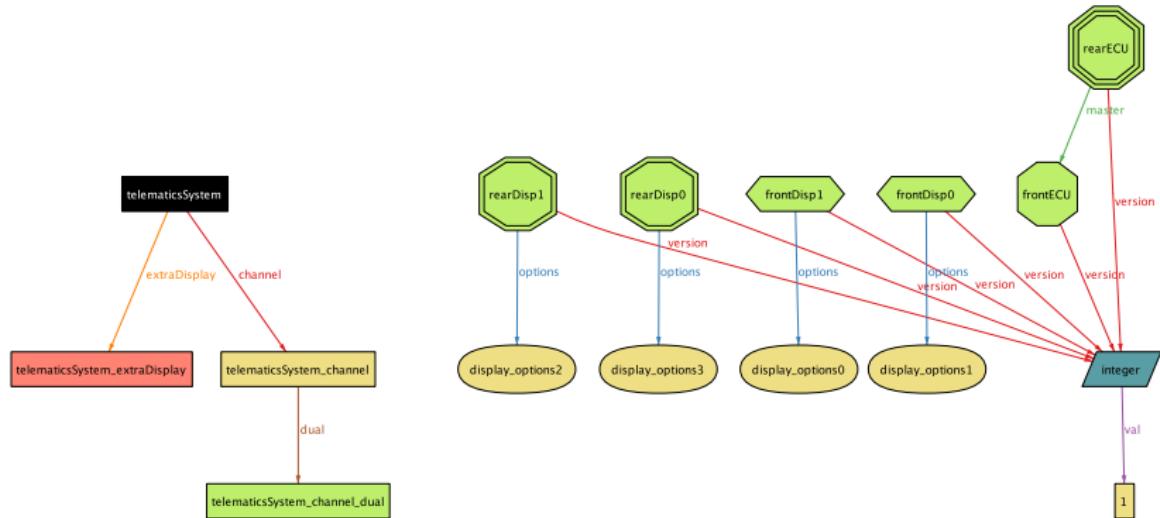
Clafer to Alloy

Structural modeling

Model checking

Alloy: instantiation, testing ideas

Telematics Instance Generated by Alloy



clafer2alloy Translator

Parser, desugarer, semantic analyzer, code generator
Haskell

Available online

Released source code

Documentation

clafer2alloy Translator

Parser, desugarer, semantic analyzer, code generator

Haskell

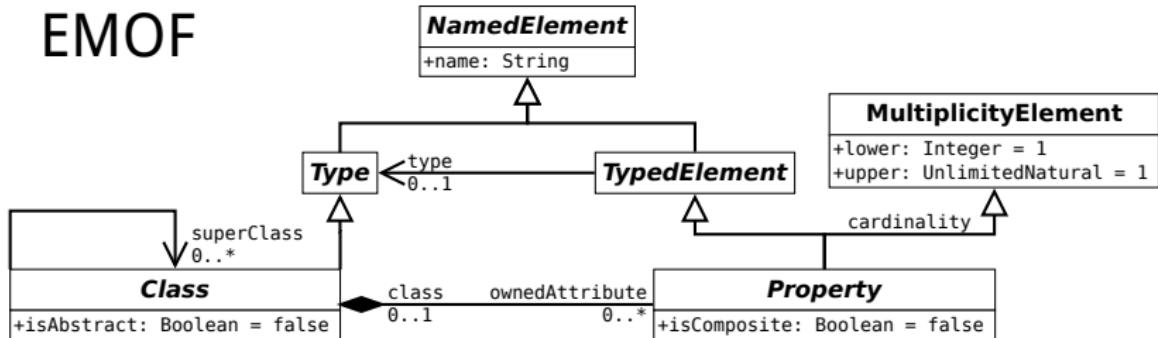
Available online

Released source code

Documentation

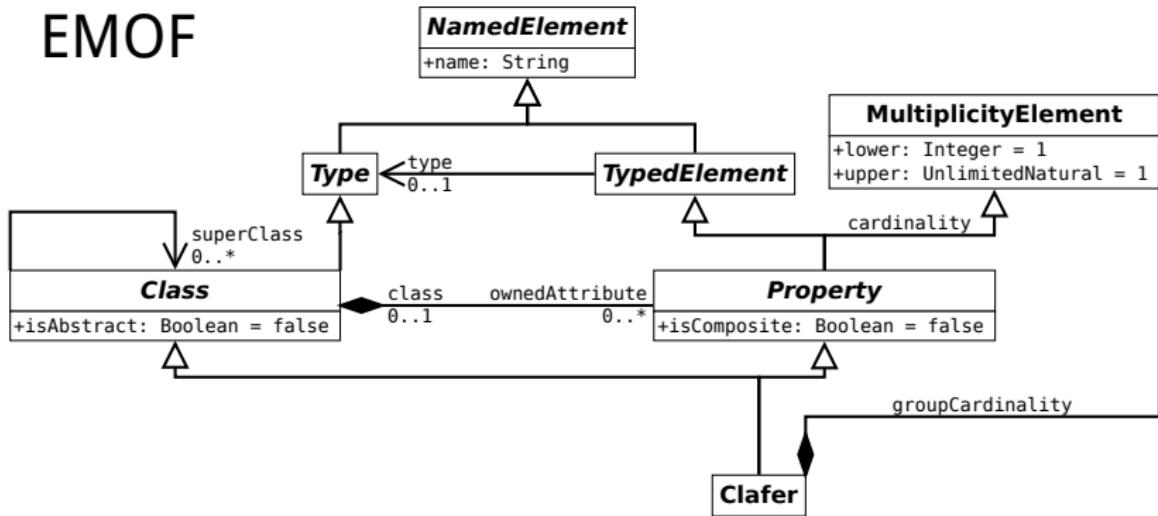
Clafer and EMOF

EMOF



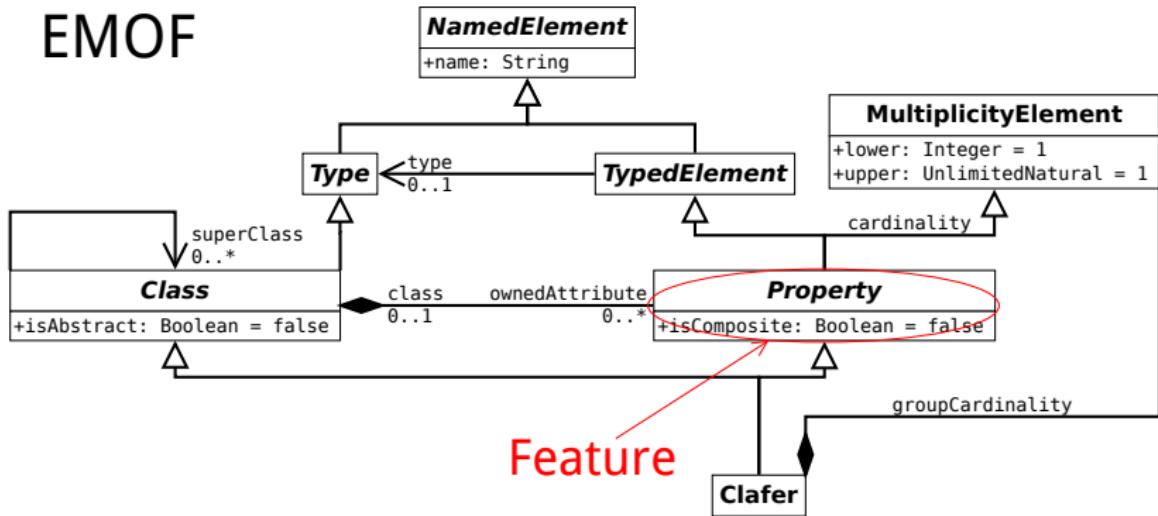
Clafer and EMOF

EMOF



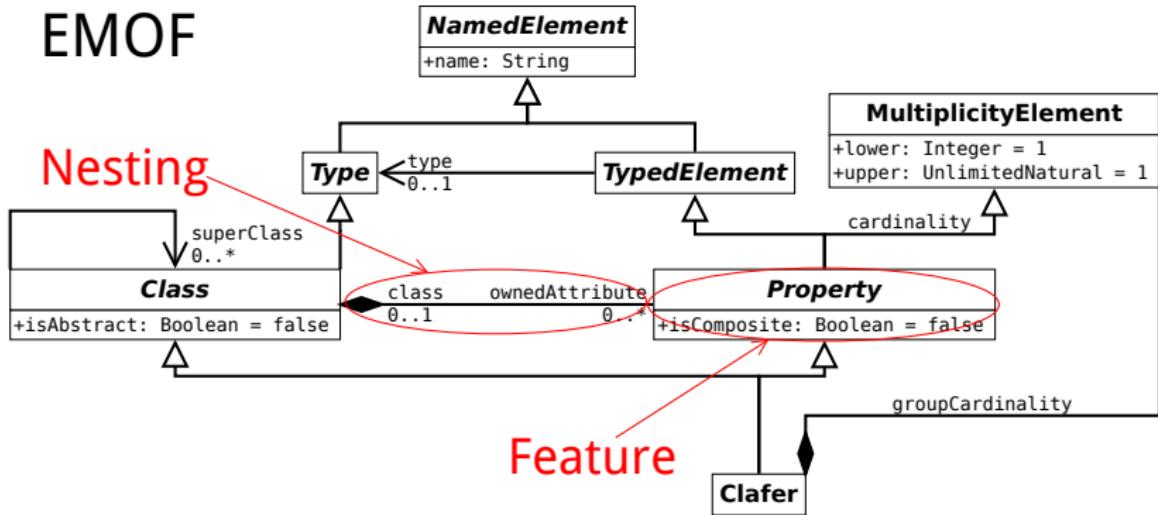
Clafer and EMOF

EMOF

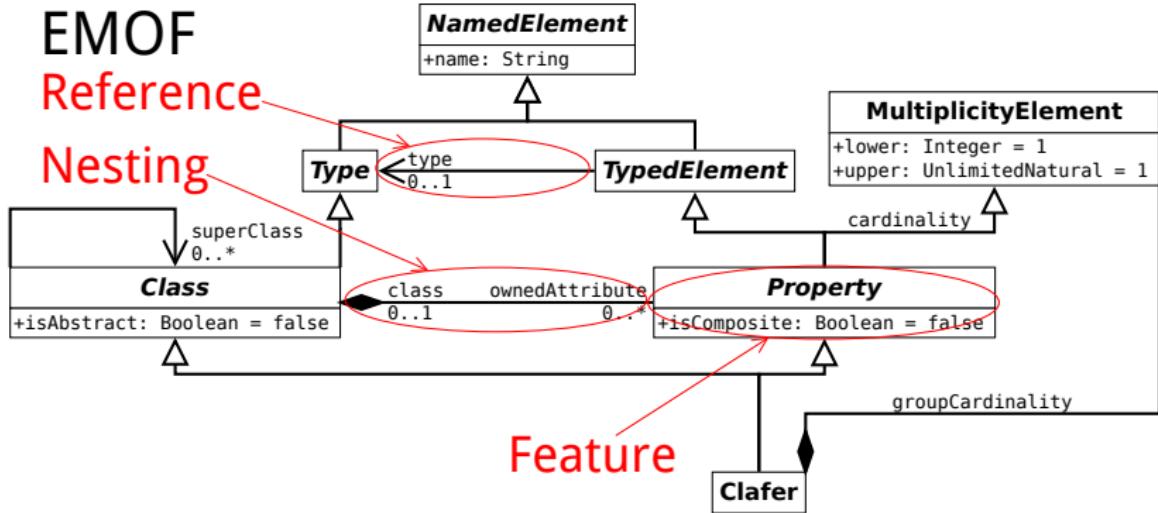


Clafer and EMOF

EMOF



Clafer and EMOF



Analyses



Feature Model and Asset Analyses

Non-trivial analyses

NP-hard problems

Instance finding by combinatorial solvers

This is what we got from Alloy!

Feature Model and Asset Analyses

Non-trivial analyses

NP-hard problems

Instance finding by combinatorial solvers

This is what we got from Alloy!

Configurers

Constraint propagation

Repairing broken configurations

Configuration completion (optimization)

FM Developers

Consistency checking

Finding contradicting constraints

Repair

Element liveness

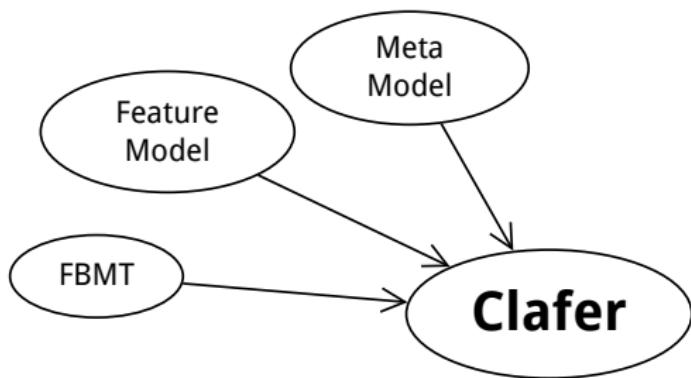
Uniqueness checking

Edit classification (refactoring, refinement, general)

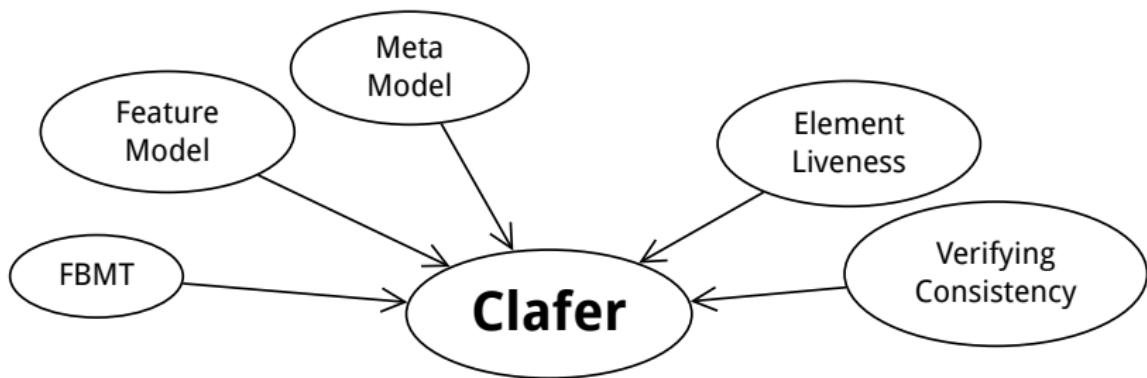
Pivot Language

Clafer

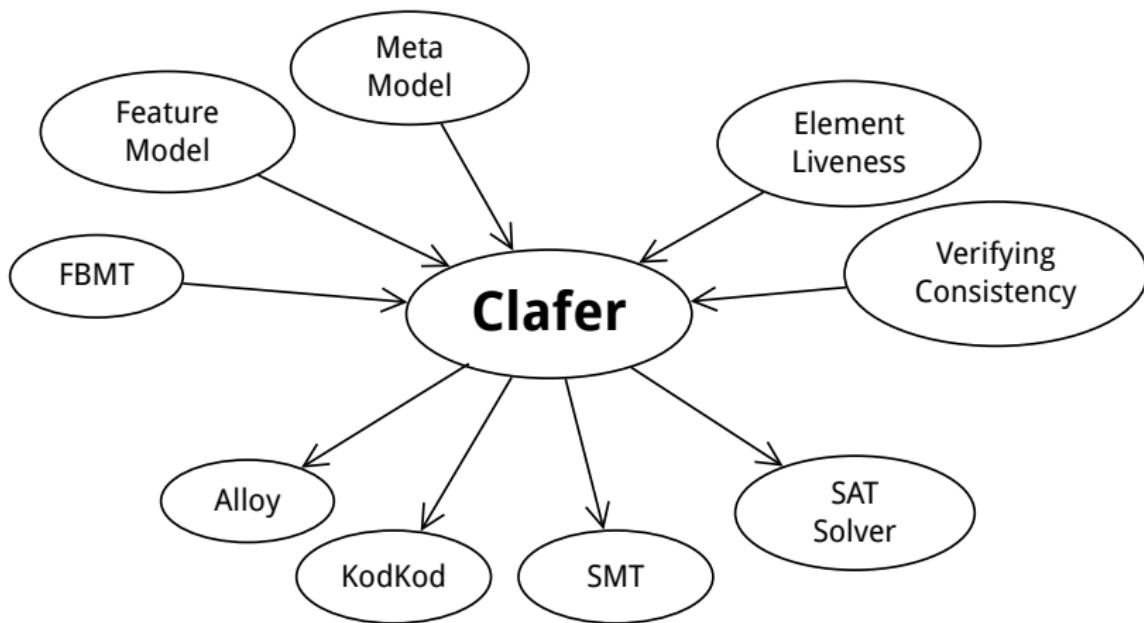
Pivot Language



Pivot Language



Pivot Language



Experimental Evaluation

Feature models

Meta-models

Feature-based model templates

Current Status

Implementation

clafer2alloy translator

Prototype in MPS (Projectional workbench)

Direct translation to KodKod (in progress)

Formalization (in progress)

Clafer in MPS

The screenshot shows the MPS (Modeling Platform) interface with the Clafer editor open. The top menu bar includes File, Edit, Search, View, Go To, Generate, Build, Run, Tools, Version Control, Window, and Help. The title bar indicates the project is 'ca.uwaterloo.gsd.clafer' located at '/home/harv/MPSProjects/ca.uwaterloo.gsd.clafer'. The left sidebar has a 'Project' view showing various Clafer elements like PrimitiveOperator, SetLogOperator, etc., and a 'Logical View' showing concepts like Cardinality, Clafer, Enumeration, Extendable, GroupCardinality, Element, Identifier, Interval, Module, and QuotedClafer. The main workspace displays Clafer code:

```
options
  xor size
    small
    large
  cache ?
    size -> integer
    fixed ?
    [ small && cache == fixed ]

comp
  version -> integer

ECU extends comp

display extends comp
  server -> ECU
  options
    [ version == server . version ]

p7aECU extends ECU
  display 1..2
  [ server = parent ]
  [ ~ options . cache ]

ECU1 extends p7aECU

ECU2 extends p7aECU ?
  master -> ECU1

telematicsSystem
  xor channel
    single
    dual
  extraDisplay ?
  xor size
    small
    large
  [ dual <=> ECU2 ]
  [ extraDisplay <=> # ECU1 . display = 2 ]
  [ extraDisplay <=> (ECU2 <=> # ECU2 . display = 2) ]
  [ large <=> ~ p7aECU . display . options . size . small ]
  [ small <=> ~ p7aECU . display . options . size . large ]

[ dual && extraDisplay && telematicsSystem . size . large && comp . version = 1 ]
```

Concept Modeling

Clafer vs OWL

Domain and knowledge modeling

Clafer seems suitable for non-technical users

OWL is machine oriented

Is modeling in Clafer more intuitive than OWL?

Conclusion

Conclusion

Modeling and analyzing SPLs

Concept modeling

Common infrastructure

Non-trivial analyses

Conclusion

Modeling and analyzing SPLs

Concept modeling

Common infrastructure

Non-trivial analyses

Conclusion

Modeling and analyzing SPLs

Concept modeling

Common infrastructure

Non-trivial analyses

Conclusion

Modeling and analyzing SPLs

Concept modeling

Common infrastructure

Non-trivial analyses

Thanks for listening!

Questions?

gsd.uwaterloo.ca/clafer