SPL re-engineering

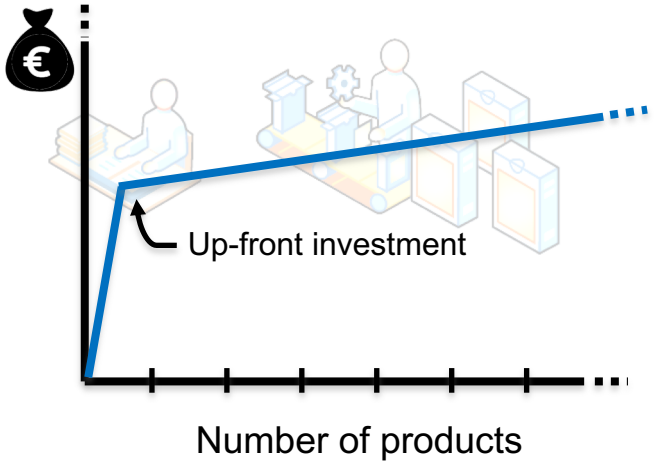
Tewfik Ziadi

Software Product Line Engineering

- SPLE has matured and is widely used in production. Many approaches for this **top-down** vision:
  - **Variability design and manipulation** [Benavides et al. 06, Schobbens 06]
  - **Reusable assets implementation**
    - Code level [Apel et al. 13, Batory et al., 08]
    - Models [Haugen et al. 13].
  - **Product derivation** [Appel et al. 08, Batory, 09]
- Many **existing supporting tools**: FeatureIDE, FeatureHouse, CVL,pure::variant, Gears, etc.

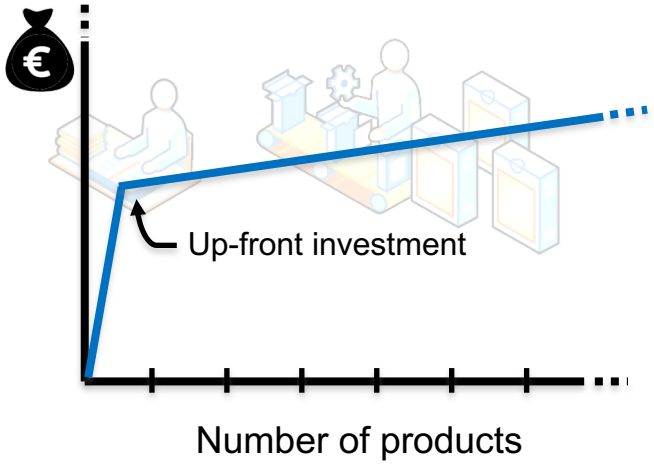
Software Product Line Adoption

- Adopting an SPL approach and designing variability is still a **major challenge** for the following reasons:
  - High **up-front investment**.



Software Product Line Adoption

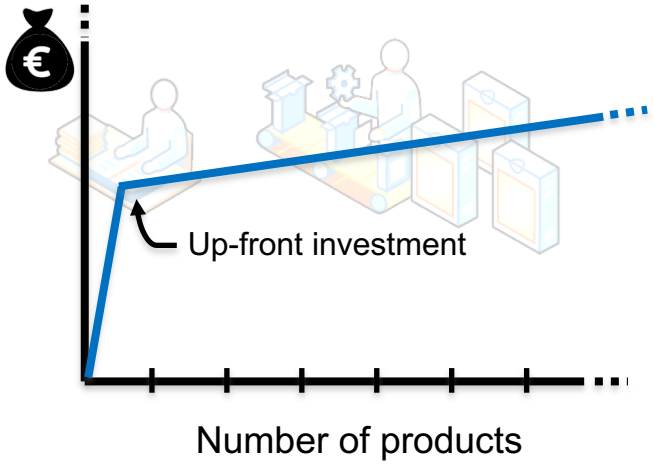
- Adopting an SPL approach and designing variability is still a **major challenge** for the following reasons:
  - High **up-front investment**.



  - Need a **complete understanding of the variability** to **anticipate** all possible variations.

Software Product Line Adoption

- Adopting an SPL approach and designing variability is still a **major challenge** for the following reasons:
  - High **up-front investment**.



  - Need a **complete understanding of the variability** to **anticipate** all possible variations.

➔ **Only possible in mature domains**

Software Product Line Adoption

- It is very difficult (impossible in some cases) for industrial companies to adopt SPL from the beginning.

- **50**% of industrial companies have **existing products (legacy)** before adopting SPL practices.

- SPL adoption from legacy application using **two scenarios**:
  - **Scenario 1**: From a set of variants (created using **clown-and-own**) to SPL
  - **Scenario 2**: From a **single legacy application** to SPL

Berger *et al.* A Survey of Variability Modeling in Industrial Practice, VaMoS '13

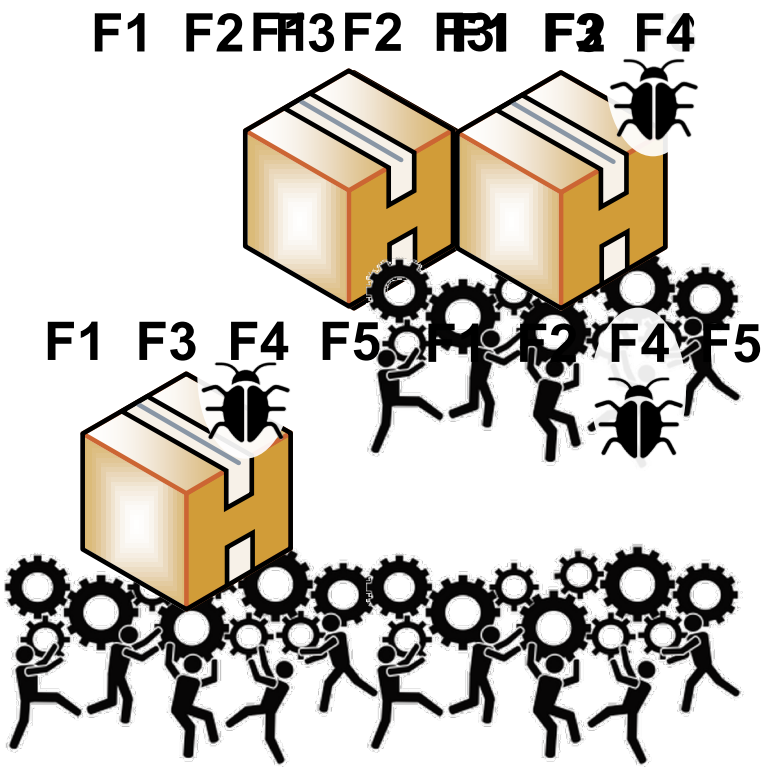
Extractive SPL adoption

- In the rest:
  - **Scenario 1 (LIP6)**: BUT4Reuse. a generic and extension framework for SPL extraction.
    - The principals of the framework
    - Concrete examples to extract SPL from model variants.
  - **Scenario 2 (Mobioos - RedFabriQ)**:
    - The Variability Engine in Mobioos
    - Demo by Karim Ghalab

Software Product Line Adoption: Scenario 1

Clone-and-own:

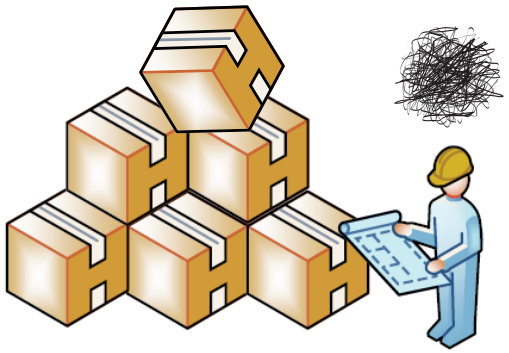
Opportunistic reuse



Software Product Line Adoption

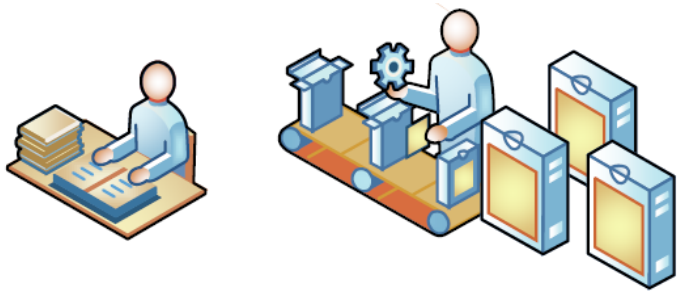
Clone-and-own:

Opportunistic reuse



SPL:

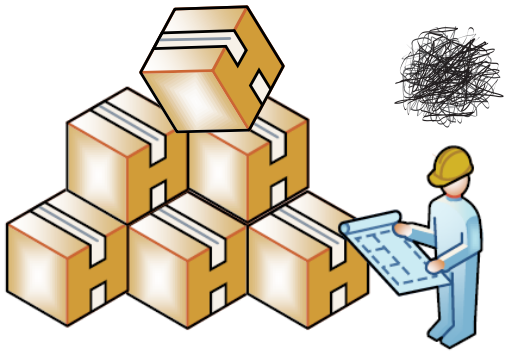
Systematic reuse



Software Product Line Adoption

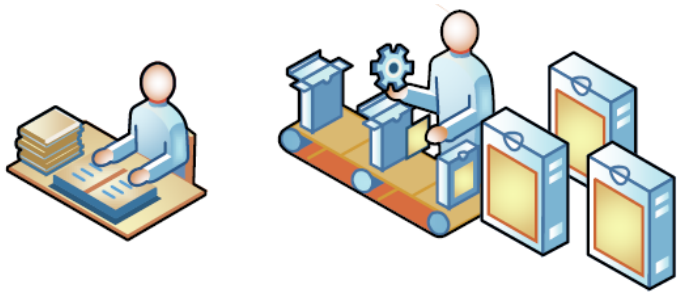
Clone-and-own:

Opportunistic reuse



SPL:

Systematic reuse



Extractive SPL adoption:
SPL Re-egineering

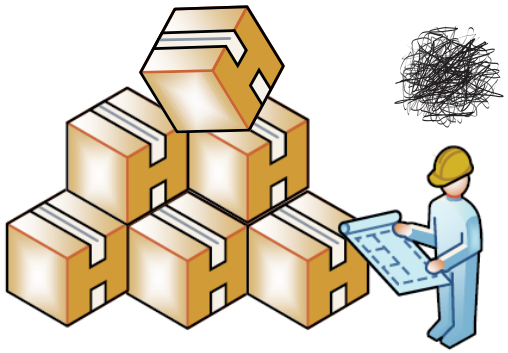
Software Product Line Adoption

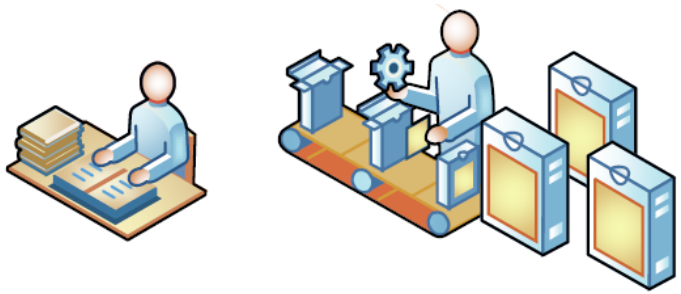
Clone-and-own:

Opportunistic reuse

SPL:

Systematic reuse

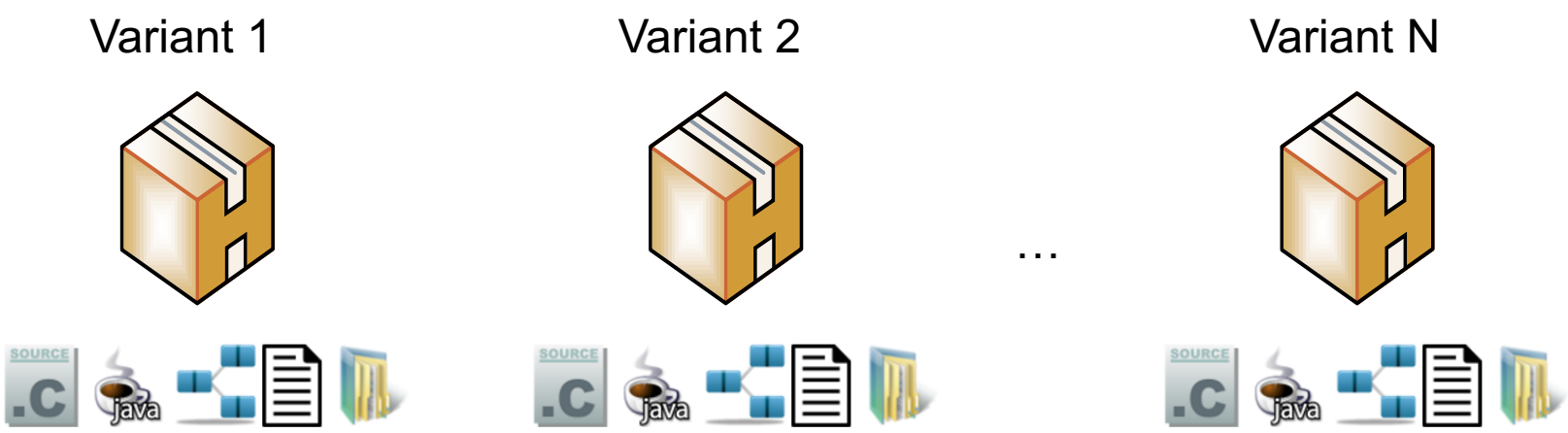




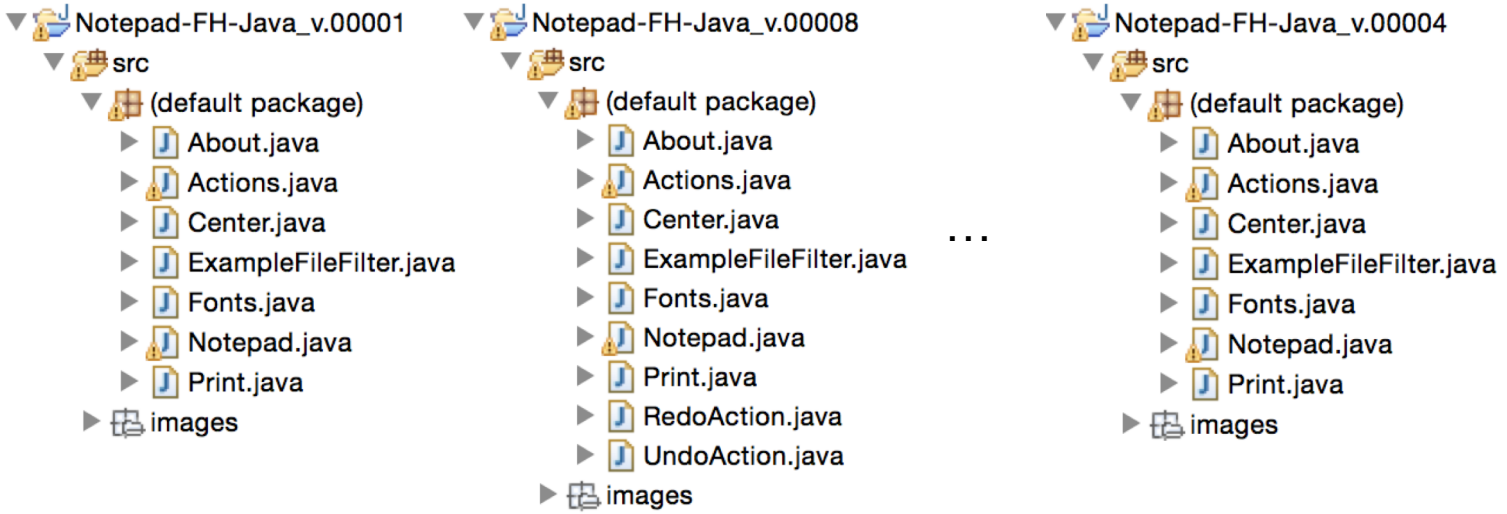
Extractive SPL adoption

Extractive SPL adoption

Static information:
code source, models, requirements..



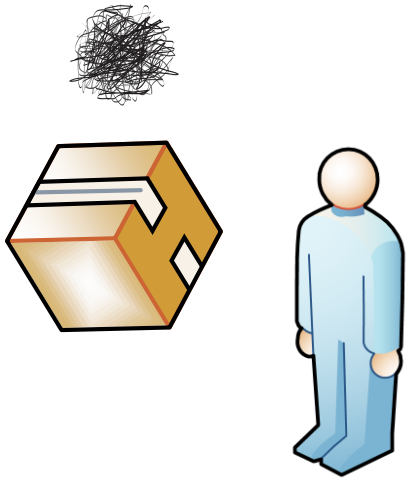
Example of static information



Software Product Line Adoption

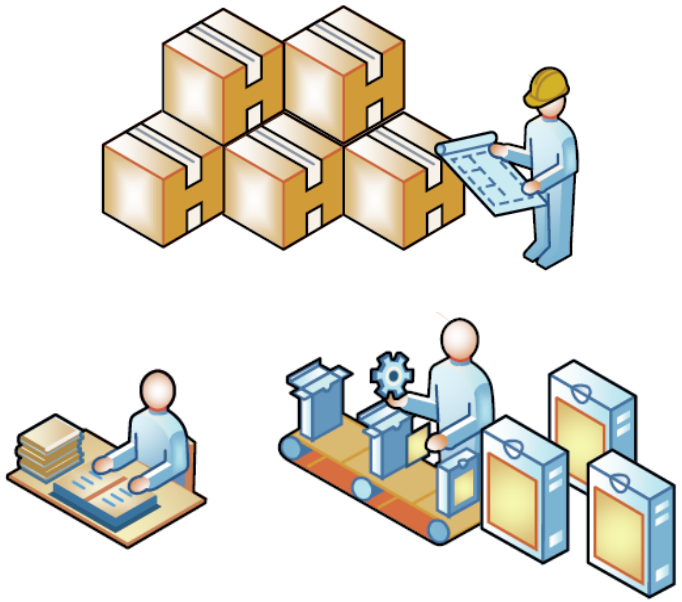
Clone-and-own:

Opportunistic reuse



SPL:

Systematic reuse

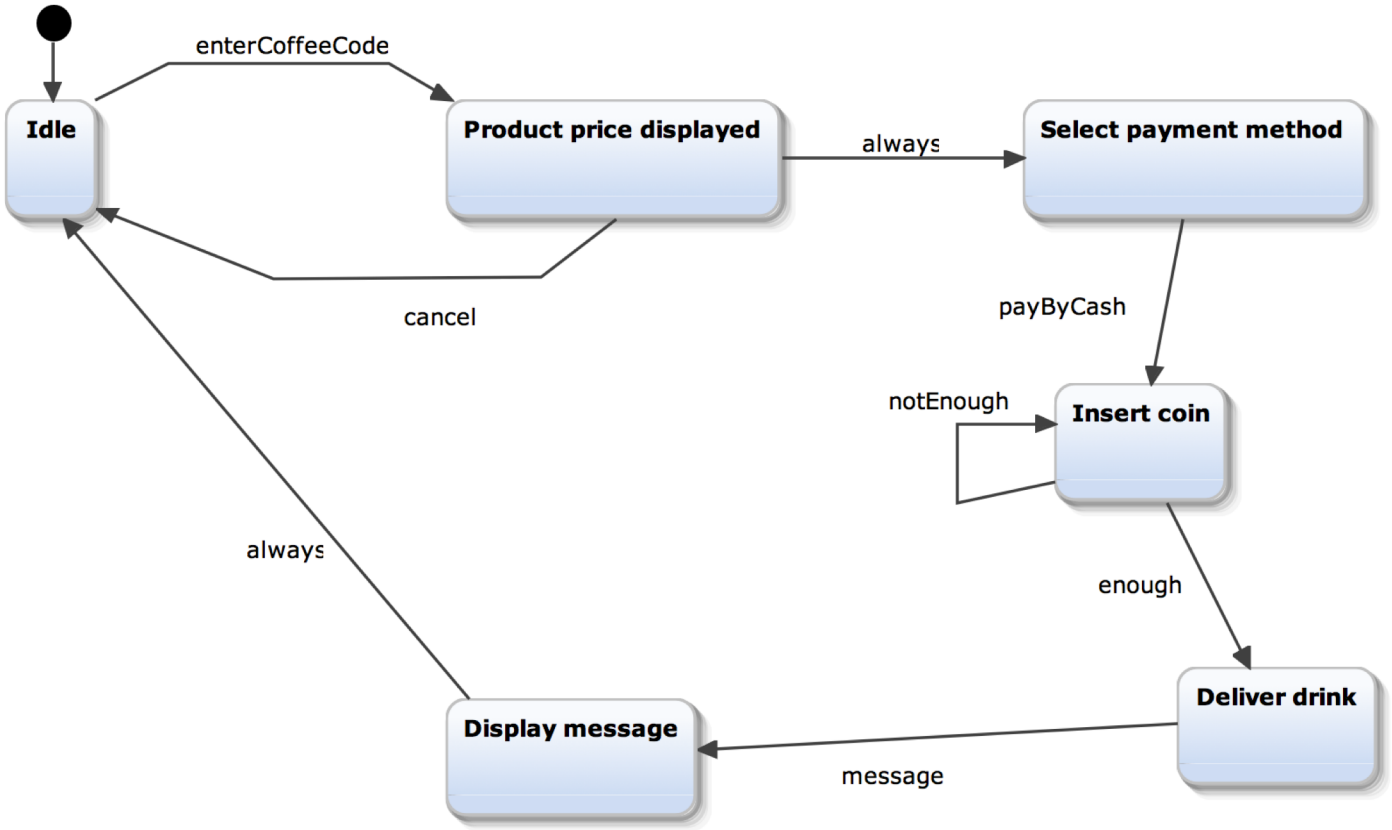


Extractive SPL adoption:
SPLRe-egineering

Krueger *et al.* Easing the transition to software mass customization, PFE '01

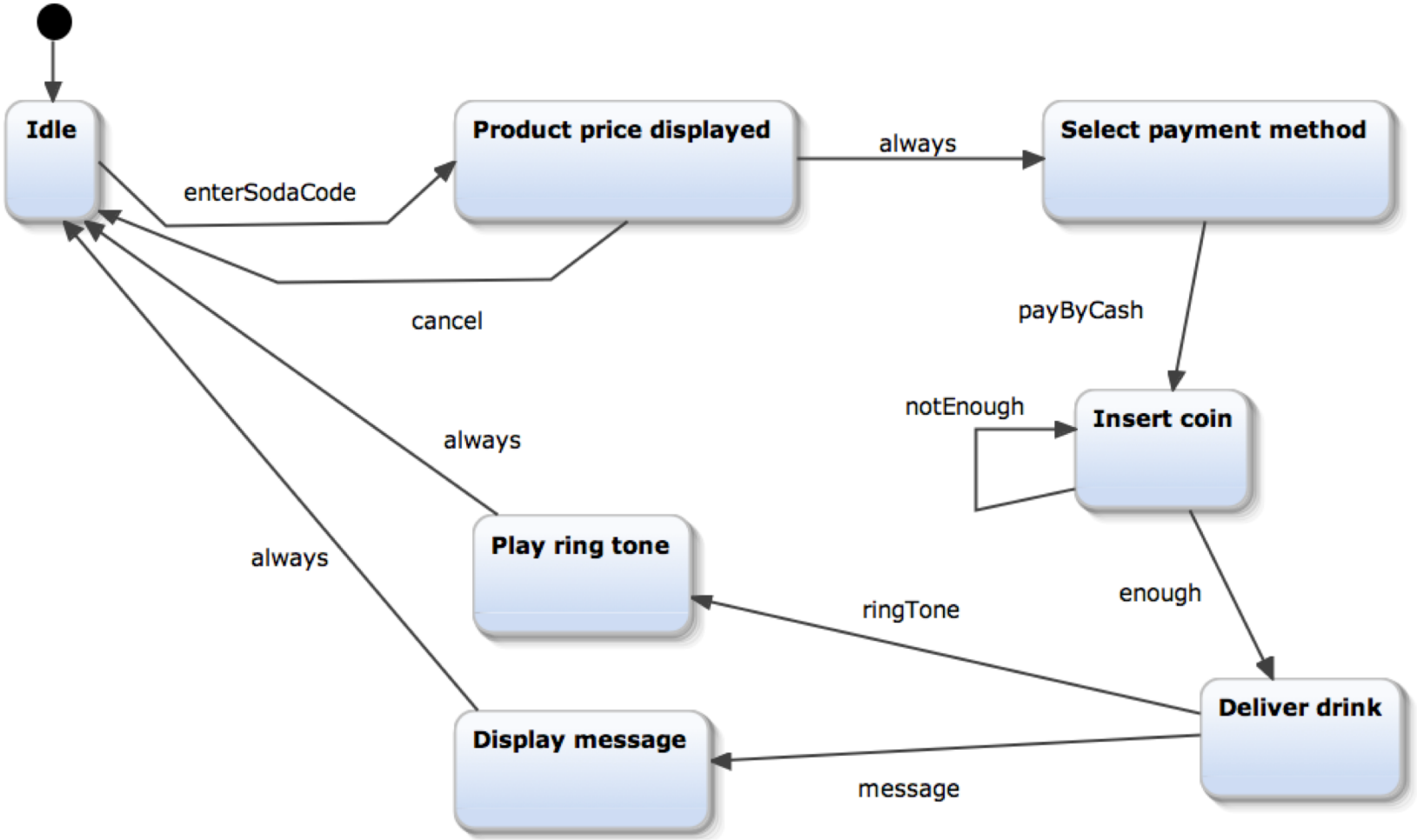
Motivations

- Vending Machine: Coffee, Cash



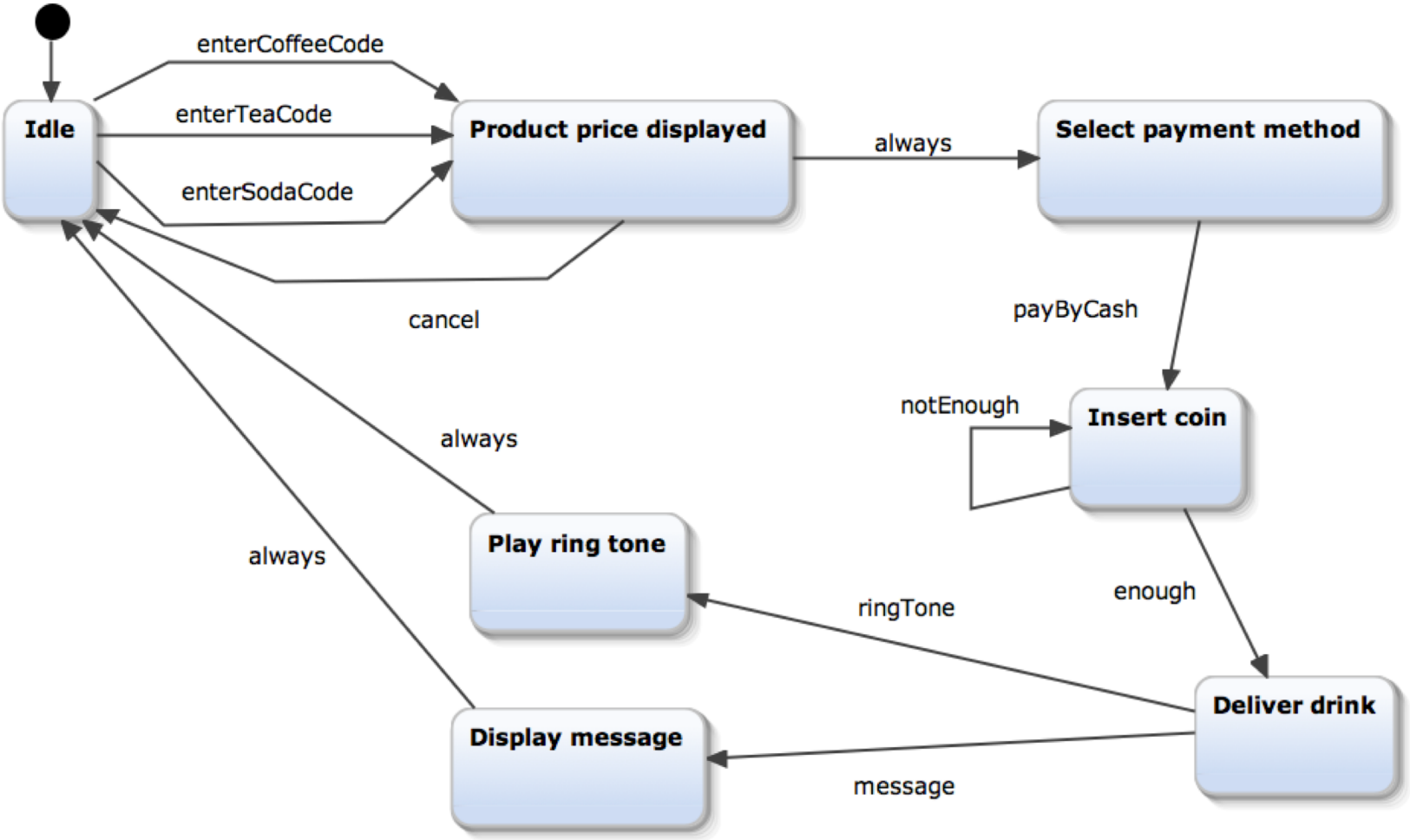
Motivations

- Vending Machine: Soda, Cash, PlayRing

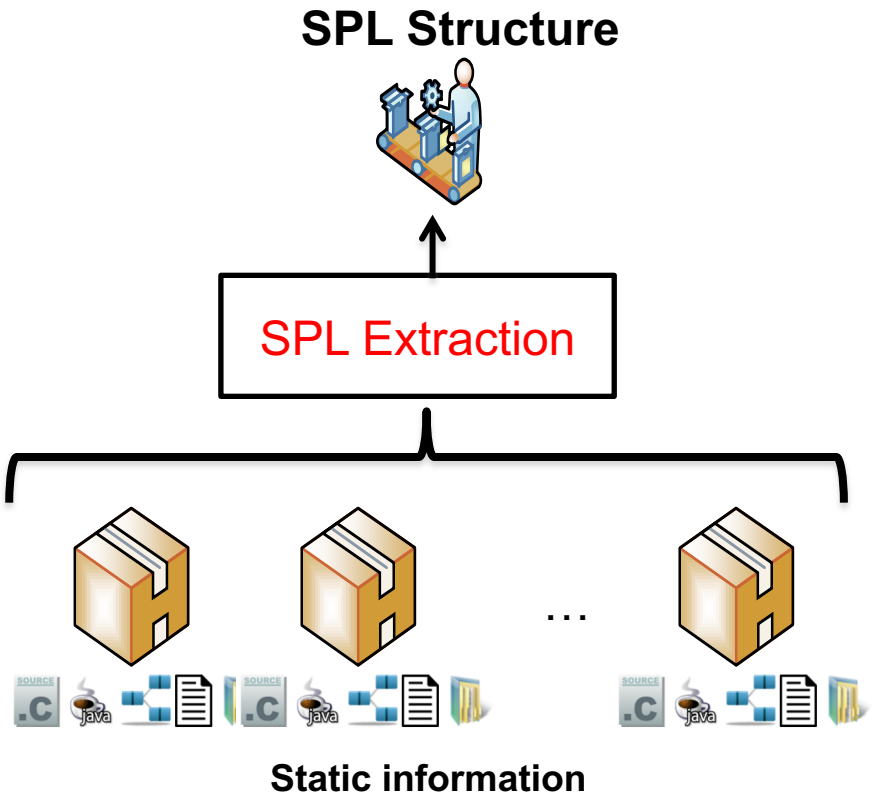


Motivations

- Vending Machine: Coffee, Tea, Soda, Cash, PlayRing



Extractive SPL adoption



Extractive SPL adoption

Genericity, Extensibility, Unification

Benchmarking

Contributions addressed in the same **conceptual** and **technical** framework

Bottom-Up Technologies for Reuse

BUT4Reuse
Bottom-Up Technologies for Reuse

A joint collaboration with university of Luxembourg
J. Martinez's PhD, 2016

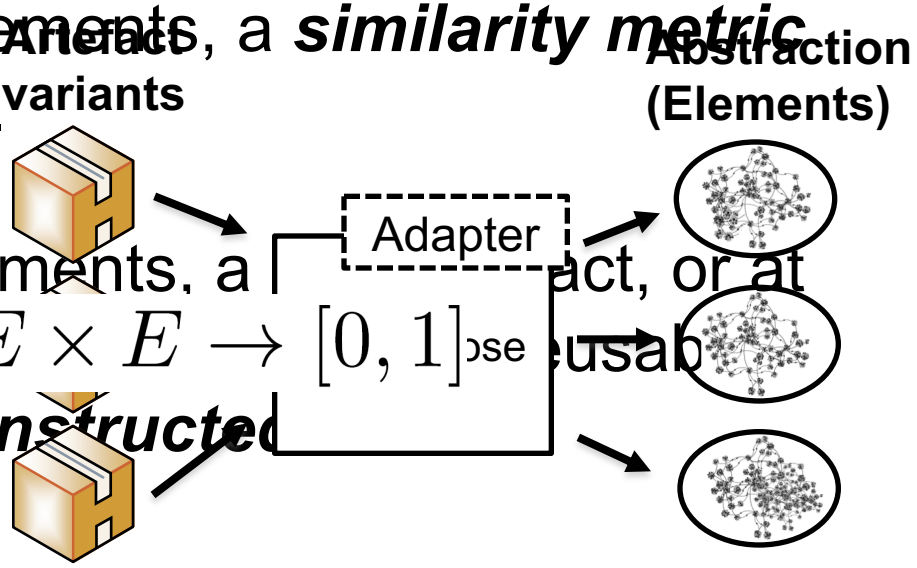
A generic and extensible framework

Generic: Intermediate representation based on **five principles**

1) A typical software artefact can be decomposed into a set of ***elements.***

2) Given a pair of elements, a ***similarity metric*** can be computed.

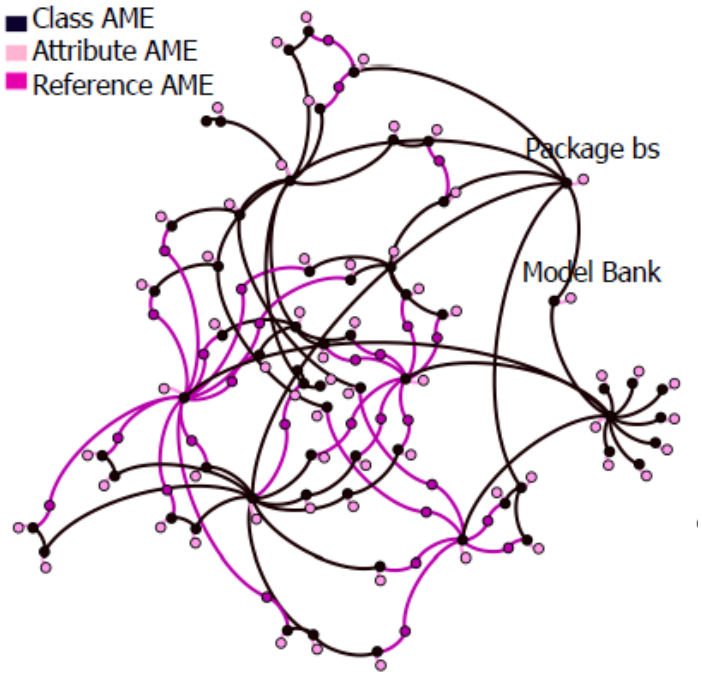
3) Given a set of elements, a [illegible]act, or at least a part ($\sigma : E \times E \rightarrow [0,1]$ [illegible]usable asset), can be ***constructed***



A generic and extensible framework

Generic: Intermediate representation based on **five principles**

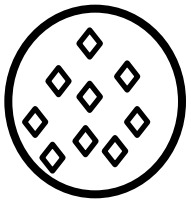
4) For a set of elements, we can identify the different **dependencies**.

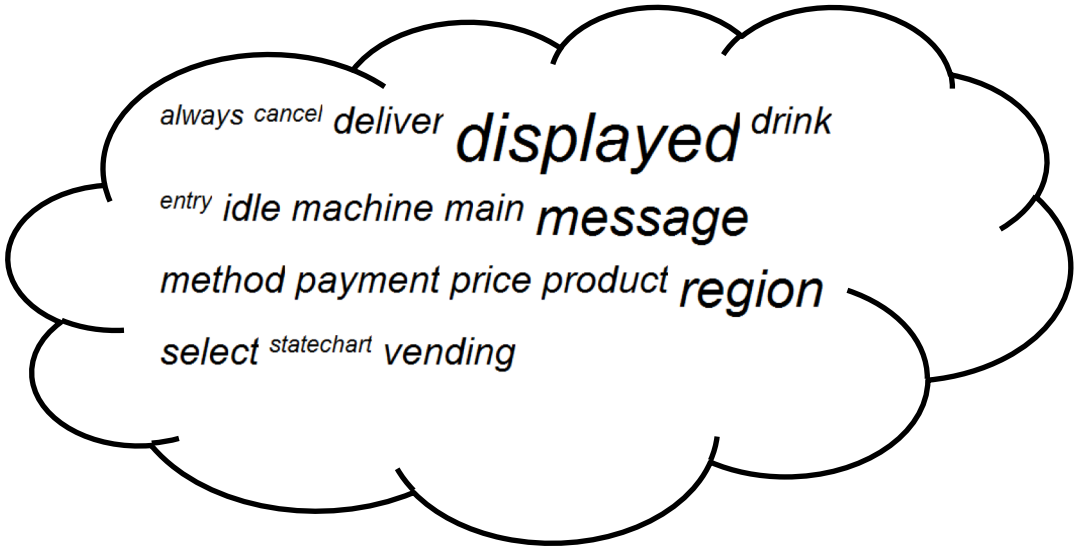


A generic and extensible framework

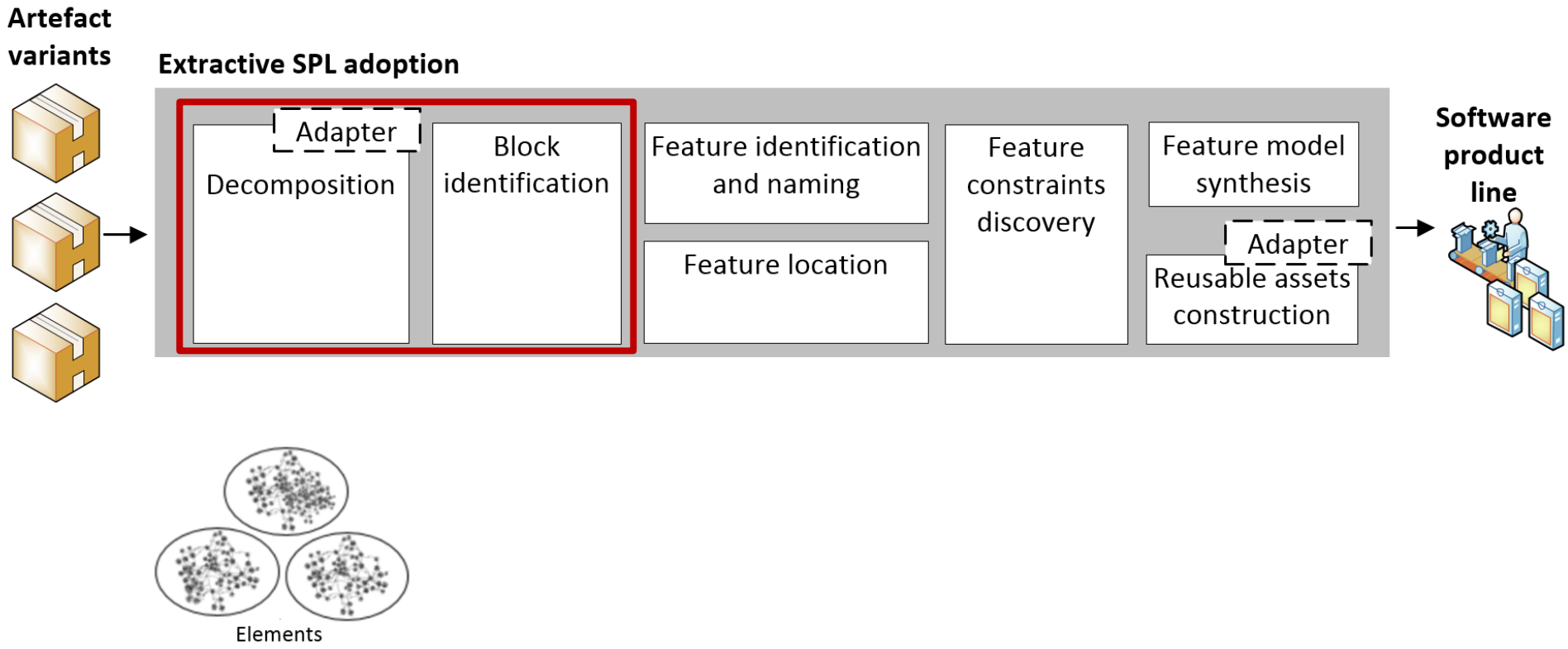
Generic: Intermediate representation based on **five principles**

5) Each set of elements can be characterized by a set of words that identify it.

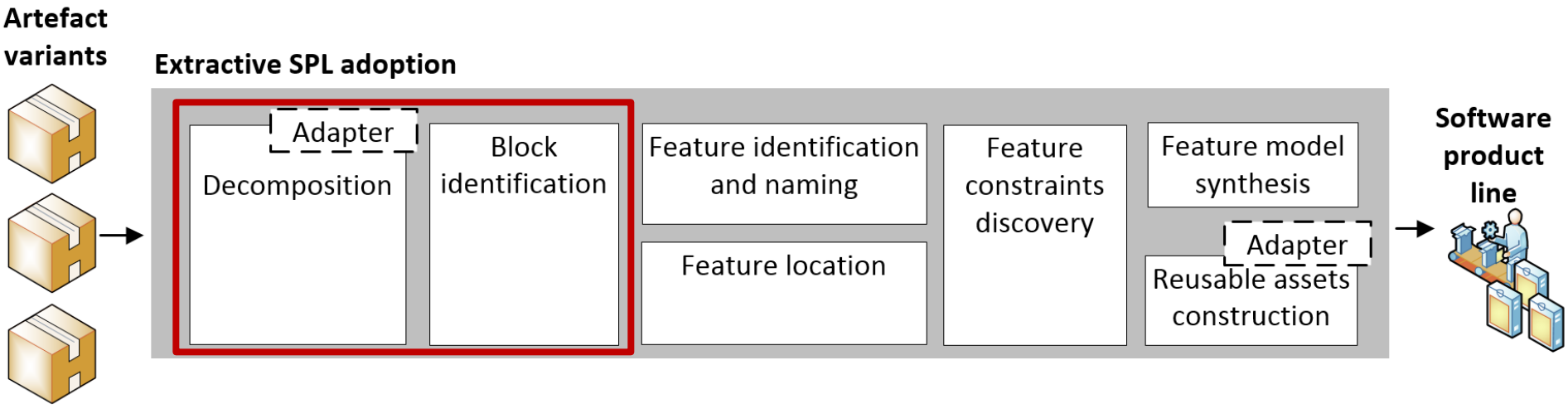


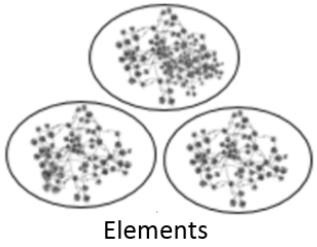


A generic and unified framework



A generic and unified framework

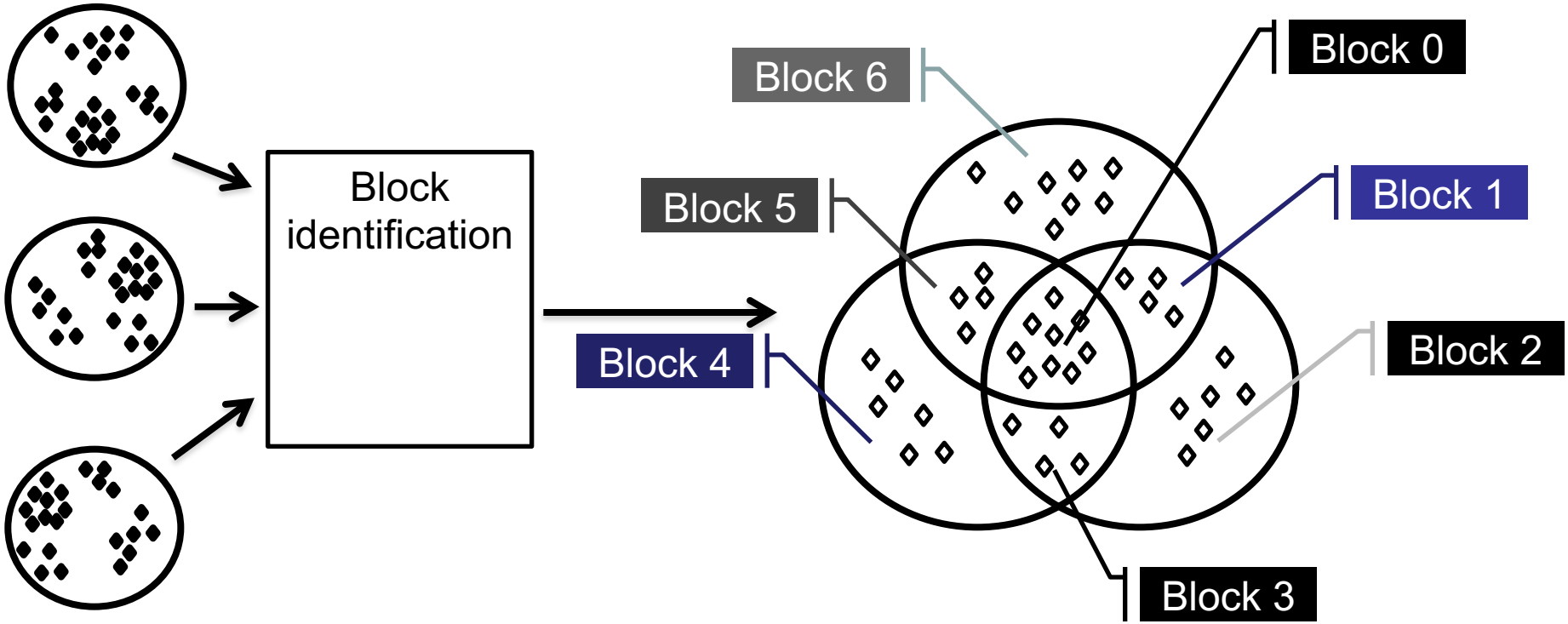




Adapter:

- Specific to each artefact type.
- Deals with the 5 principals

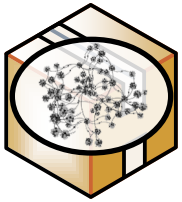
Block identification



Concrete usages of the framework

15 **Adapters**

Artefact types



- Java source code
- C/C++ source code

ExtractorPL, Ziadi *et al.* SAC '14

- EMF models

MoVa2PL, Martinez *et al.* ASE '15

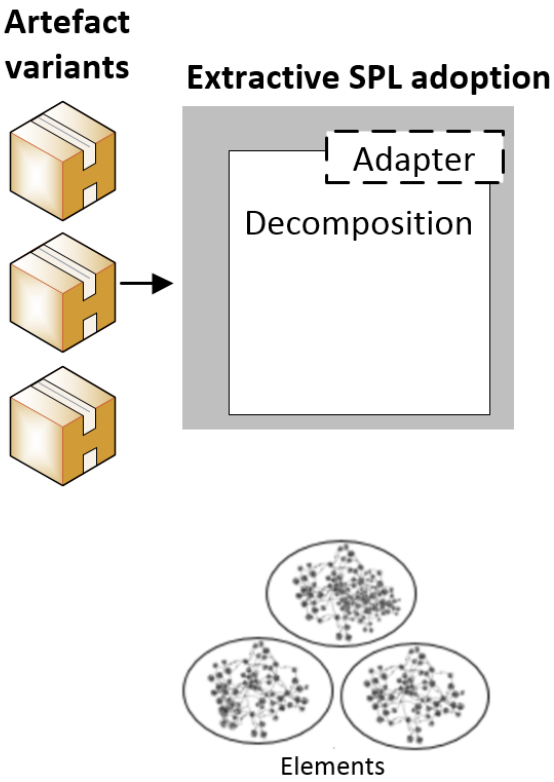
- File structure
- Text lines
- Natural language text
- Requirements
- CSV
- JSON
- Scratch
- Images
- Music
- Graphs
- Eclipse plugin-based system

Martinez *et al.* SPLC'15

- Android

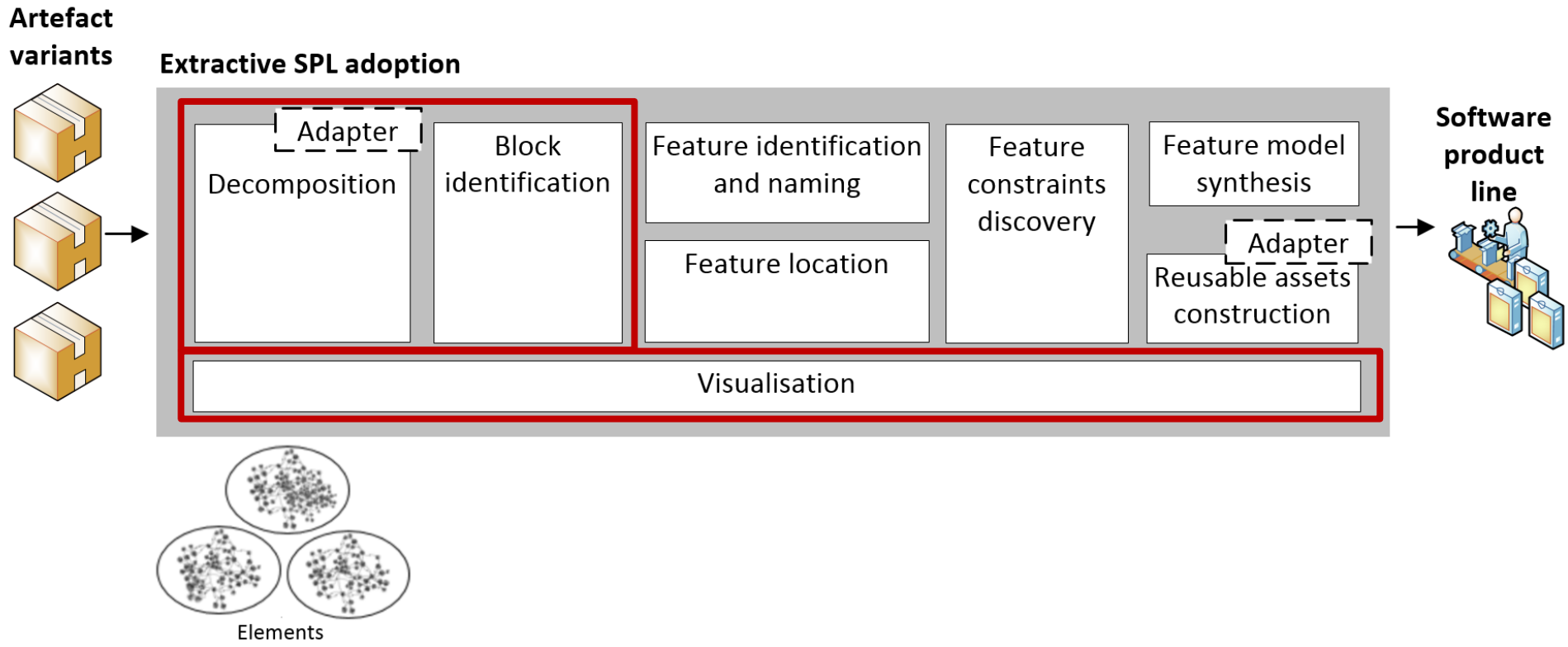
A generic and unified framework

- **Unified**

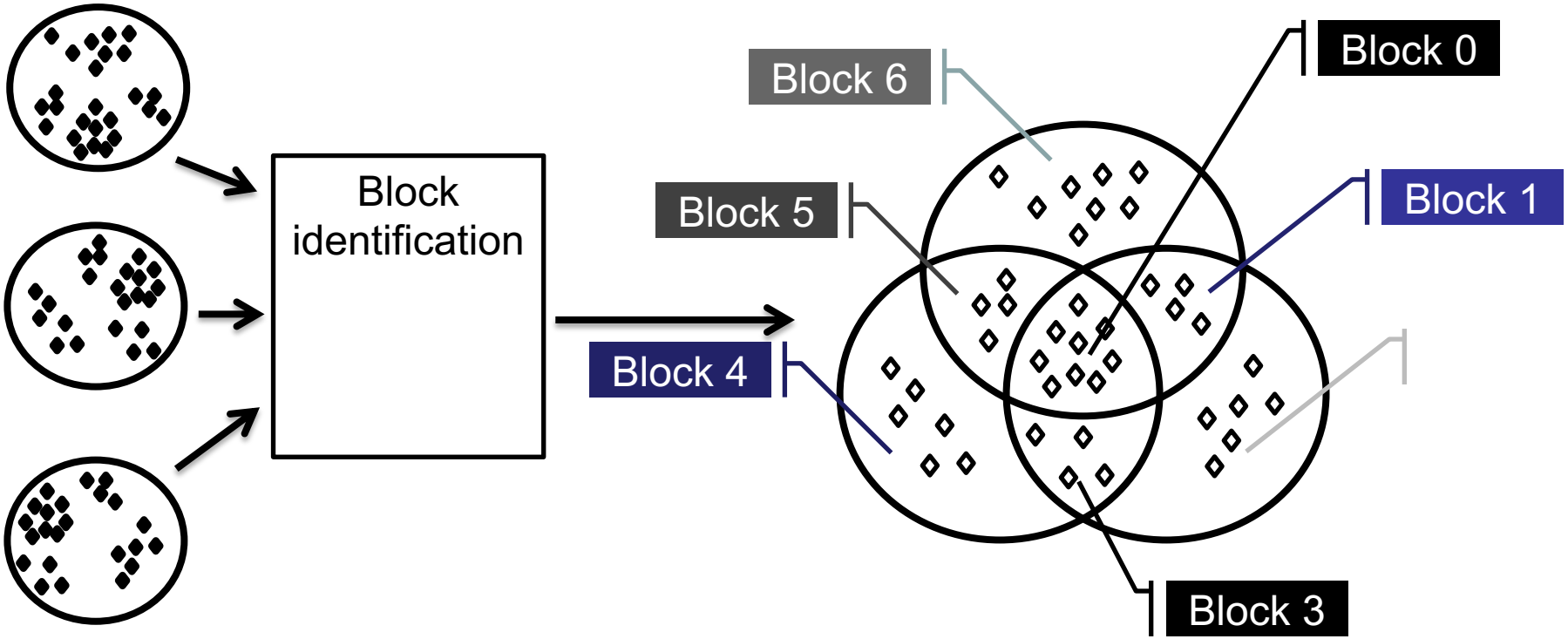


A generic and unified framework

- **Unified**
  - ***Extensible*** for different techniques for each activity
  - Enabling to ***chain activities***
  - Enabling to ***combine techniques***



Block identification



Interdependent elements, Ziadi *et al.* CSMR '12

Techniques

4 Block identification

- Formal Concept Analysis (FCA)
- Interdependent elements
- FCA + structural splitting
- Similar elements

1 Feature identification and naming

- VariClouds, guided or automatic

6 Feature location

- Latent semantic indexing
- Feature-specific block
- Strict feature-specific (SFS)
- SFS + shared term
- SFS + term frequency
- SFS + tf-idf

3 Feature constraints discovery

- Structural constraints
- Association-rules mining
- FCA concept lattices

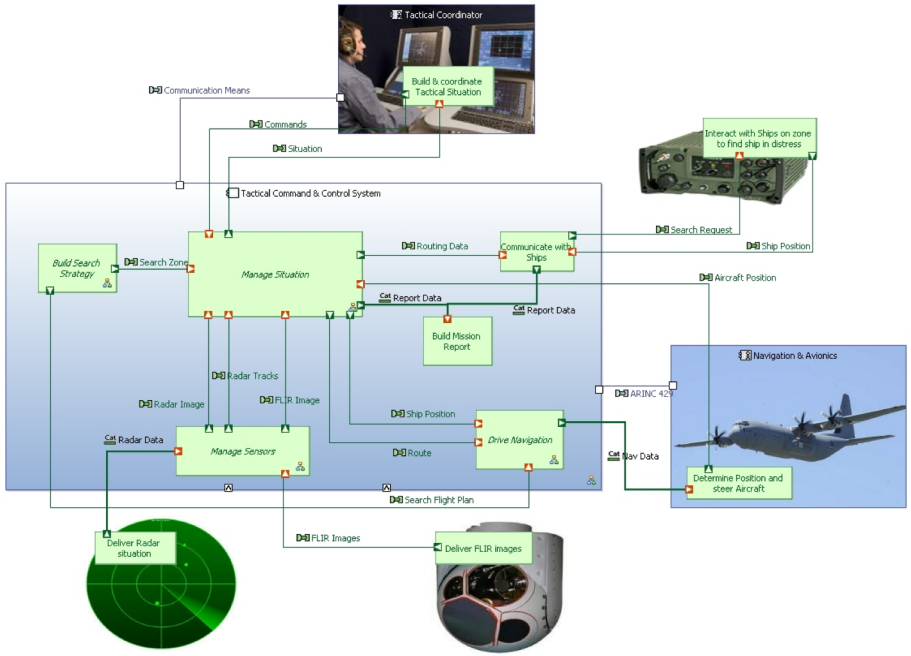
2 Feature model synthesis

- Flat feature diagram
- Alternatives and hierarchy

6 Visualisation paradigms

- Bars and stripes
- Heat-maps
- Graphs
- Pruned concept hierarchy (AOC posset)
- Word clouds
- Feature relations graphs

Models case study



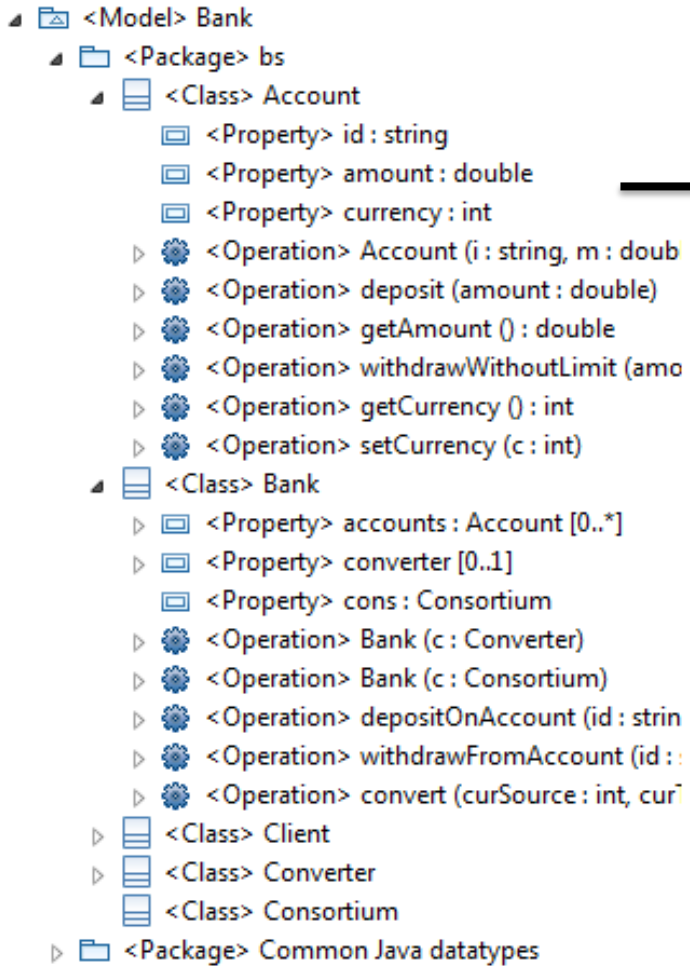
Domain Specific Languages (DSL)

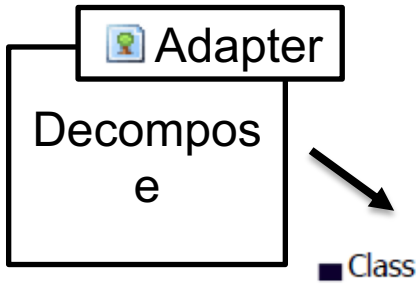
Model variants to PL (MoVa2PL)

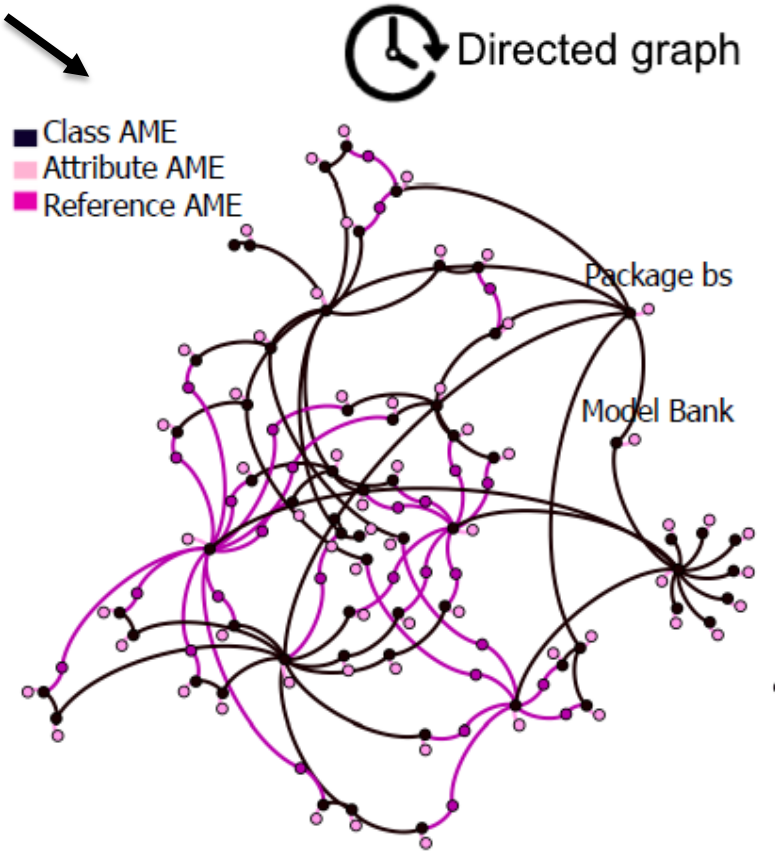
- Adapter
  - Atomic Model Elements (AMEs)

  MOF basic elements
    - *Class*, *Attribute* and *Reference*

MoVa2PL

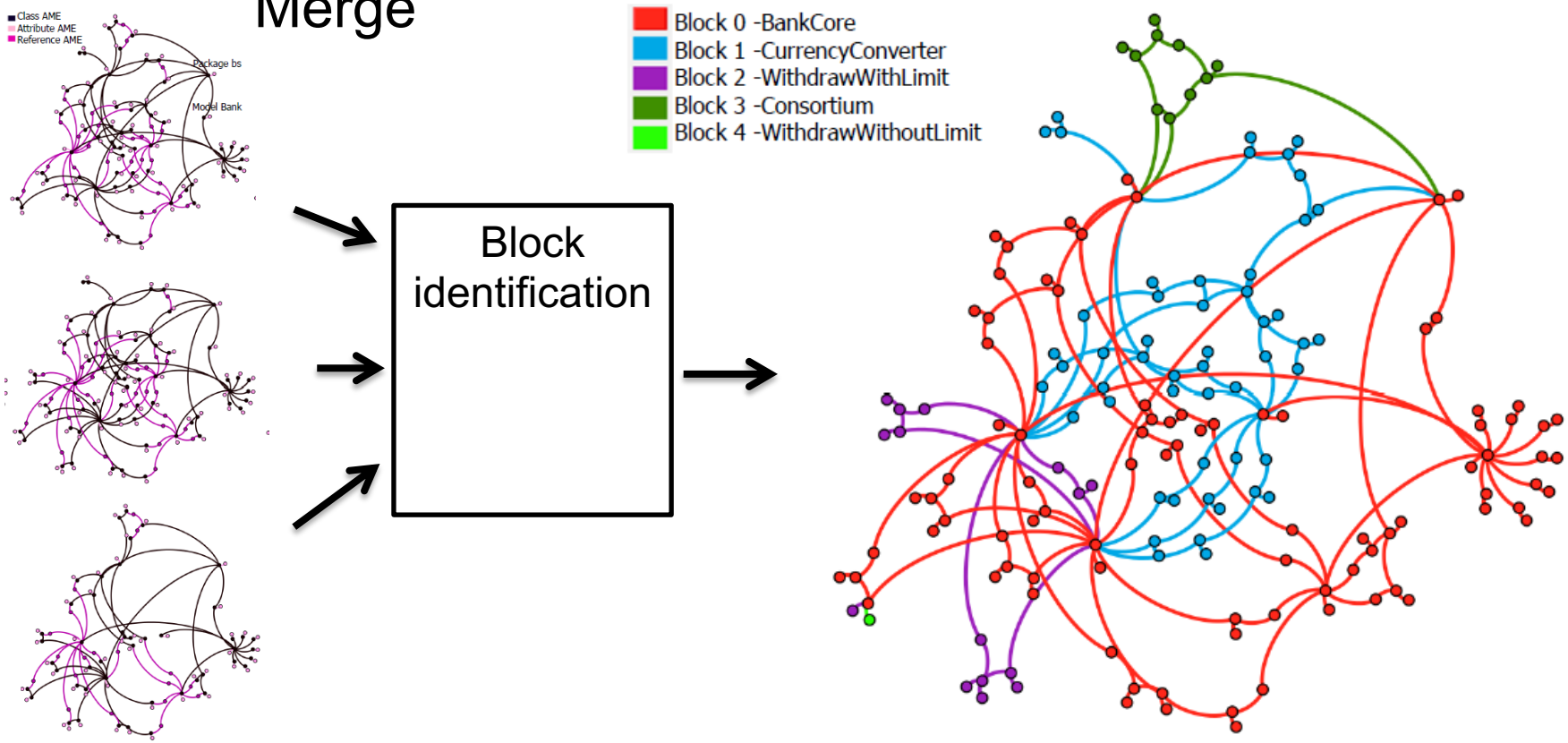

<Model> Bank
<Package> bs
<Class> Account
<Property> id : string
<Property> amount : double
<Property> currency : int
<Operation> Account (i : string, m : doub
<Operation> deposit (amount : double)
<Operation> getAmount () : double
<Operation> withdrawWithoutLimit (amo
<Operation> getCurrency () : int
<Operation> setCurrency (c : int)
<Class> Bank
<Property> accounts : Account [0..*]
<Property> converter [0..1]
<Property> cons : Consortium
<Operation> Bank (c : Converter)
<Operation> Bank (c : Consortium)
<Operation> depositOnAccount (id : strin
<Operation> withdrawFromAccount (id :
<Operation> convert (curSource : int, cur
<Class> Client
<Class> Converter
<Class> Consortium
<Package> Common Java datatypes


Adapter
Decompos
e


Directed graph
Class AME
Attribute AME
Reference AME
Package bs
Model Bank

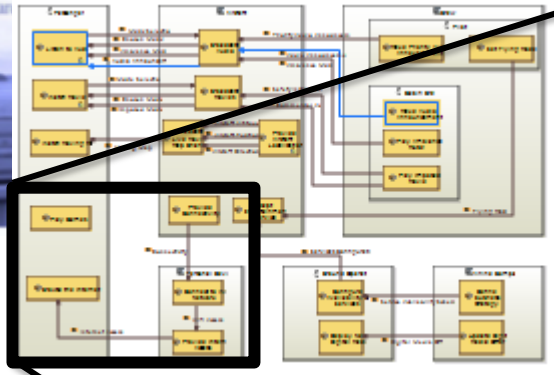
MoVa2PL

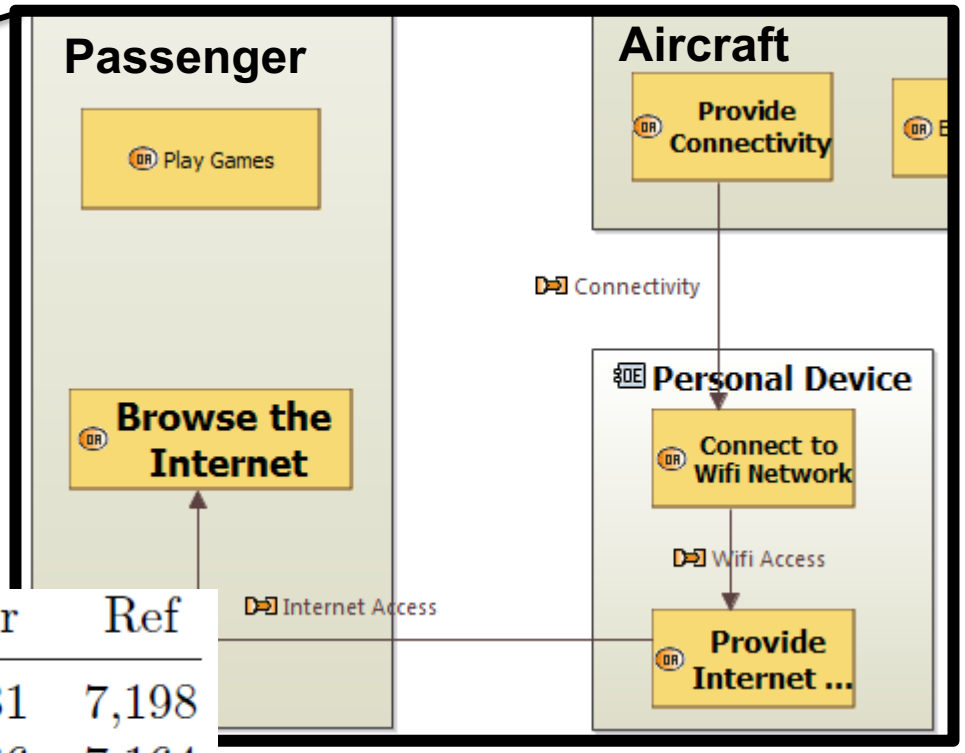
- Similarity metric for Block identification
  - Diff and match policies are flexible with EMF Diff/Merge



In-Flight Entertainment system case study

Systems Engineering DSL: Capella





Variant	AMEs	Class	Attr	Ref
Original	16,624	5,345	4,081	7,198
LowCost1	16,551	5,321	4,066	7,164
LowCost2	16,594	5,335	4,075	7,184

ArgoUML case study

- UML models editor
  - 7 variants
  - Big UML models

Variant	AMEs	Class	Attr	Ref
ActivityDisabled	157,896	51,235	77,707	28,954
CollabDisabled	158,535	51,418	78,046	29,071
DeployDisabled	157,314	51,033	77,450	28,831
Original	159,771	51,820	78,667	29,284
SequenceDisabled	155,231	50,349	76,417	28,465
StateDisabled	156,193	50,699	76,805	28,689
UsecaseDisabled	157,504	51,056	77,547	28,901

Results
In-Flight Entertainment systems

Features associated to Blocks

Block	**AMEs**	Class	Attr	Ref
Block0 -Core	16,521	5,311	4,060	7,150
Block1 -Wi-Fi	73	24	15	34
Block2 -ExteriorVideo	30	10	6	14

Discovered structural constraints

- Wi-Fi *requires* Core
- ExteriorVideo *requires* Core

Model-based SPL extracted

- Base model contains all features
- Derivation mechanism based on subtraction

Results
ArgoUML

Big Blocks identified as features

Block	**AMEs**	Class	Attr	Ref
Block 0 -Core	143,894	46,724	70,696	26,474
Block 1 -UseCase	2,260	760	1,117	383
Block 2 -Sequence	4,509	1,461	2,233	815
Block 3 -Collaboration	1,204	392	604	208
Block 4 -State	3,499	1,095	1,818	586
Block 5 -Deployment	2,457	787	1,217	453
Block 6 -Activity	1,796	559	916	321
Block 7	1	0	0	1
Block 8	1	0	0	1
⋮	⋮	⋮	⋮	⋮
Block 40	4	2	2	0

Bottom-Up Technologies for Reuse

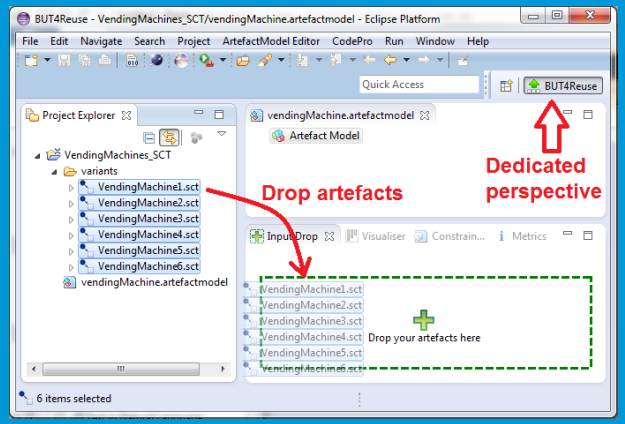
http://but4reuse.github.io

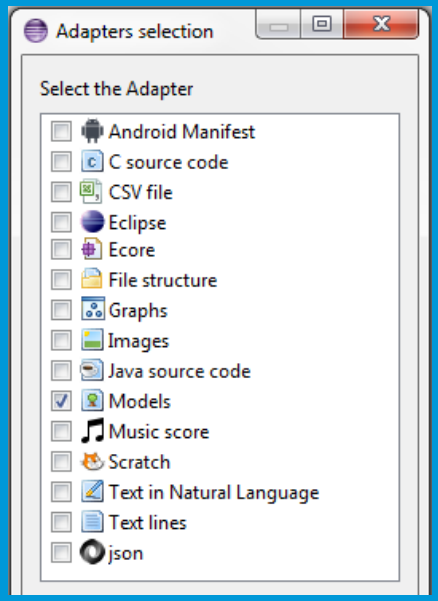
BUILT ON eclipse

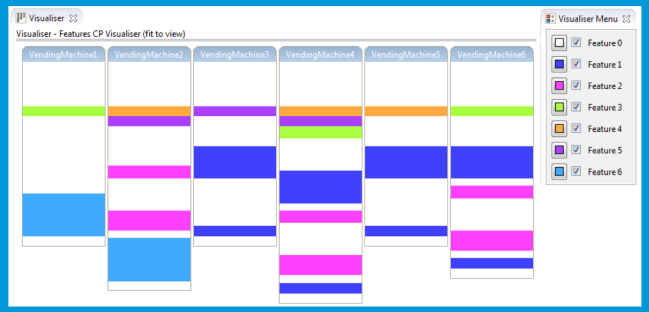
GitHub

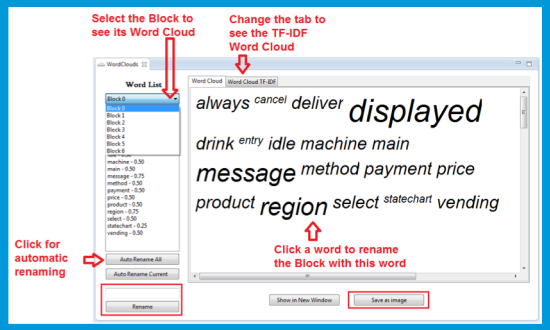
@BUT4Reuse

Tewfik.ziadi@lip6.fr

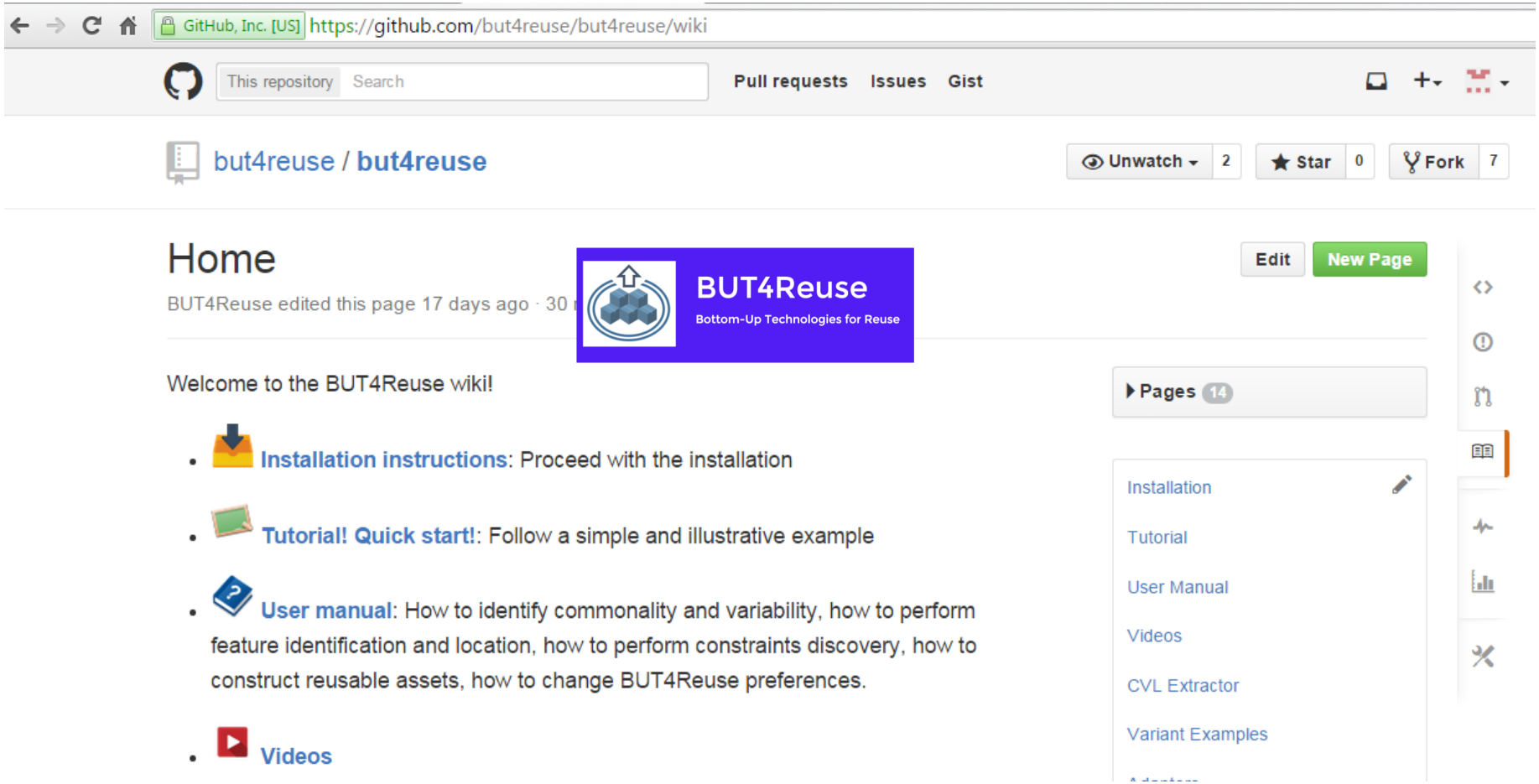






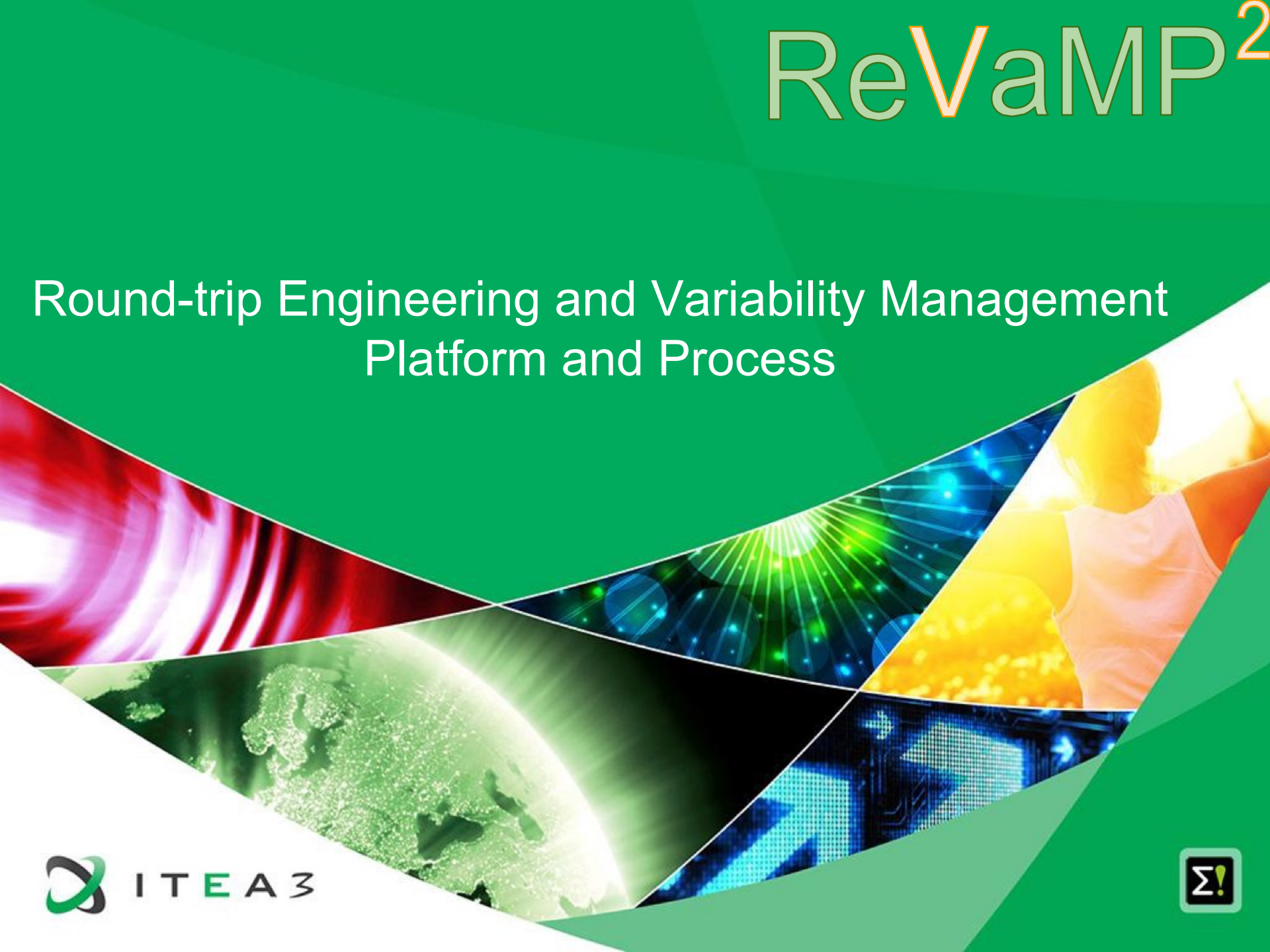


BUT4Reuse


GitHub, Inc. [US] https://github.com/but4reuse/but4reuse/wiki
This repository Search
Pull requests Issues Gist
but4reuse / but4reuse
Unwatch 2 Star 0 Fork 7
Home
BUT4Reuse edited this page 17 days ago
BUT4Reuse
Bottom-Up Technologies for Reuse
Edit New Page
Welcome to the BUT4Reuse wiki!
Installation instructions: Proceed with the installation
Tutorial! Quick start!: Follow a simple and illustrative example
User manual: How to identify commonality and variability, how to perform feature identification and location, how to perform constraints discovery, how to construct reusable assets, how to change BUT4Reuse preferences.
Videos
Pages 14
Installation
Tutorial
User Manual
Videos
CVL Extractor
Variant Examples

ReVaMP^2

Round-trip Engineering and Variability Management Platform and Process



ITEA 3

REVaMP²

http://www.revamp2-project.eu/

Project information

Project: Nº 15010 REVaMP² - Round-trip Engineering and Variability Management Platform and Process

Area: Smart Engineering

Number of partners: 30

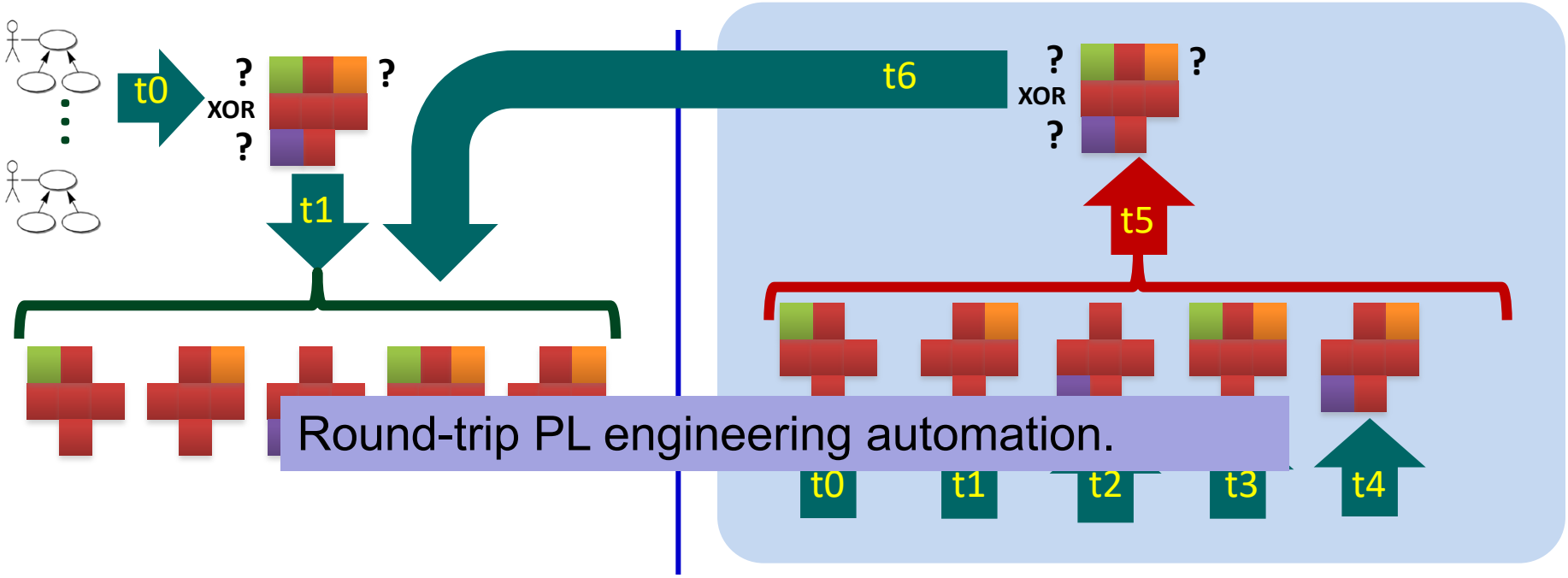
Number of countries: 5

Budget: 22,049 k€

Duration: Nov 2016 - Dec 2019

Coordinator: Softeam

Context: PL adoption



Proactive approach

➔ On established markets, long life-cycle

Extractive approach

➔ from variants created using clon-own

Objectives for REVAMP PLE methods and tools

1. Create REVaMP² **tool-chain prototype** for PL round-trip engineering:
   1. **PL extraction & Visulization**
   2. **Verification**
   3. **Co evolution**
2. Create **methodology**
3. Validate with **case study** demonstrators

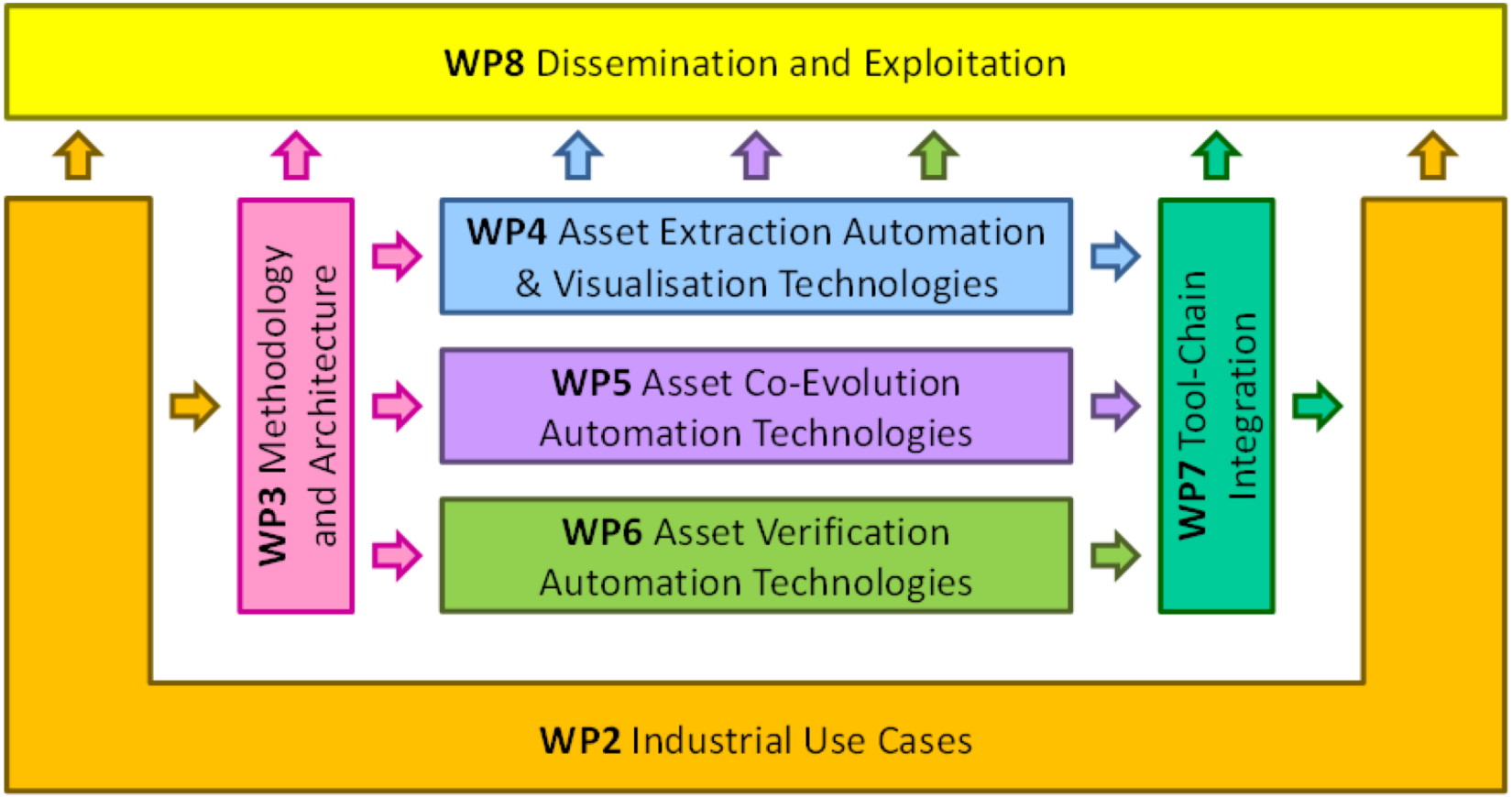
European consortium

Academic expertise / PoC providers

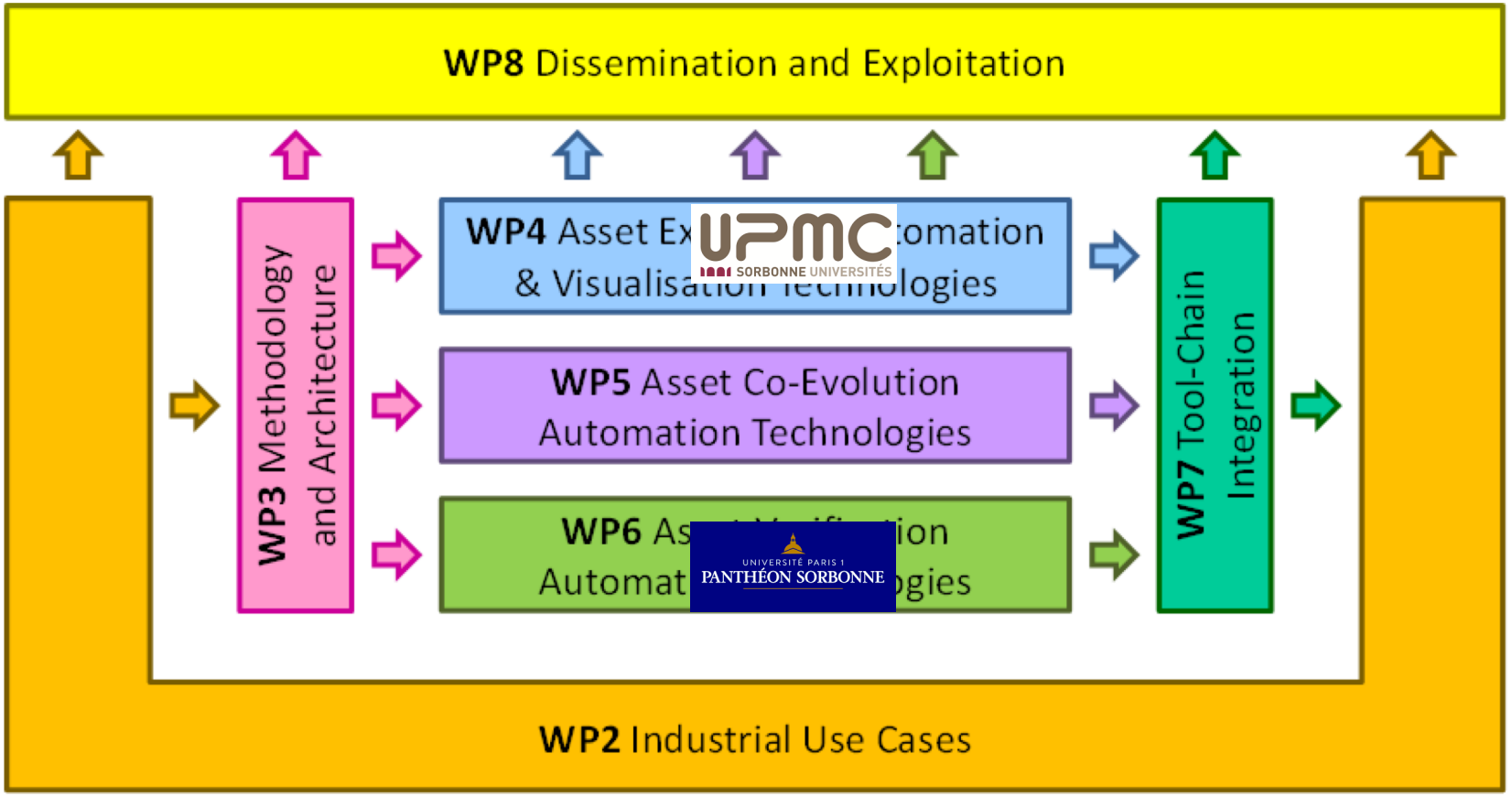
Industrial tool providers / integrators

Industrial use case providers

Work packages



Work packages



Extractive SPL adoption

- In the rest:
  - **Scenario 1 (LIP6)**: BUT4Reuse. a generic and extension framework for SPL extraction.
    - The principals of the framework
    - Concrete examples to extract SPL from model variants.
  - **Scenario 2 (Mobioos - RedFabriQ)**:
    - The Variability Engine in Mobioos
    - Demo by Karim Ghalab