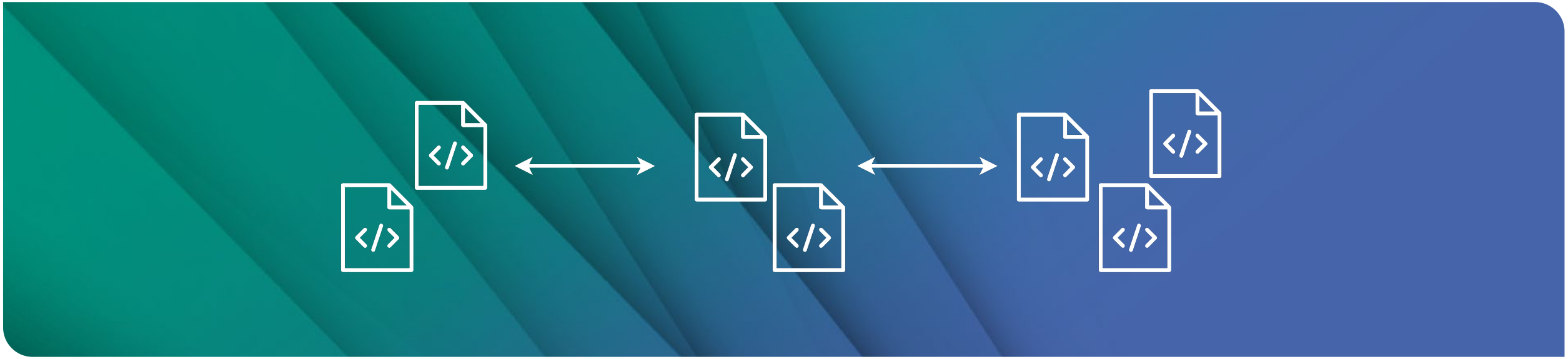
Karlsruhe Institute of Technology

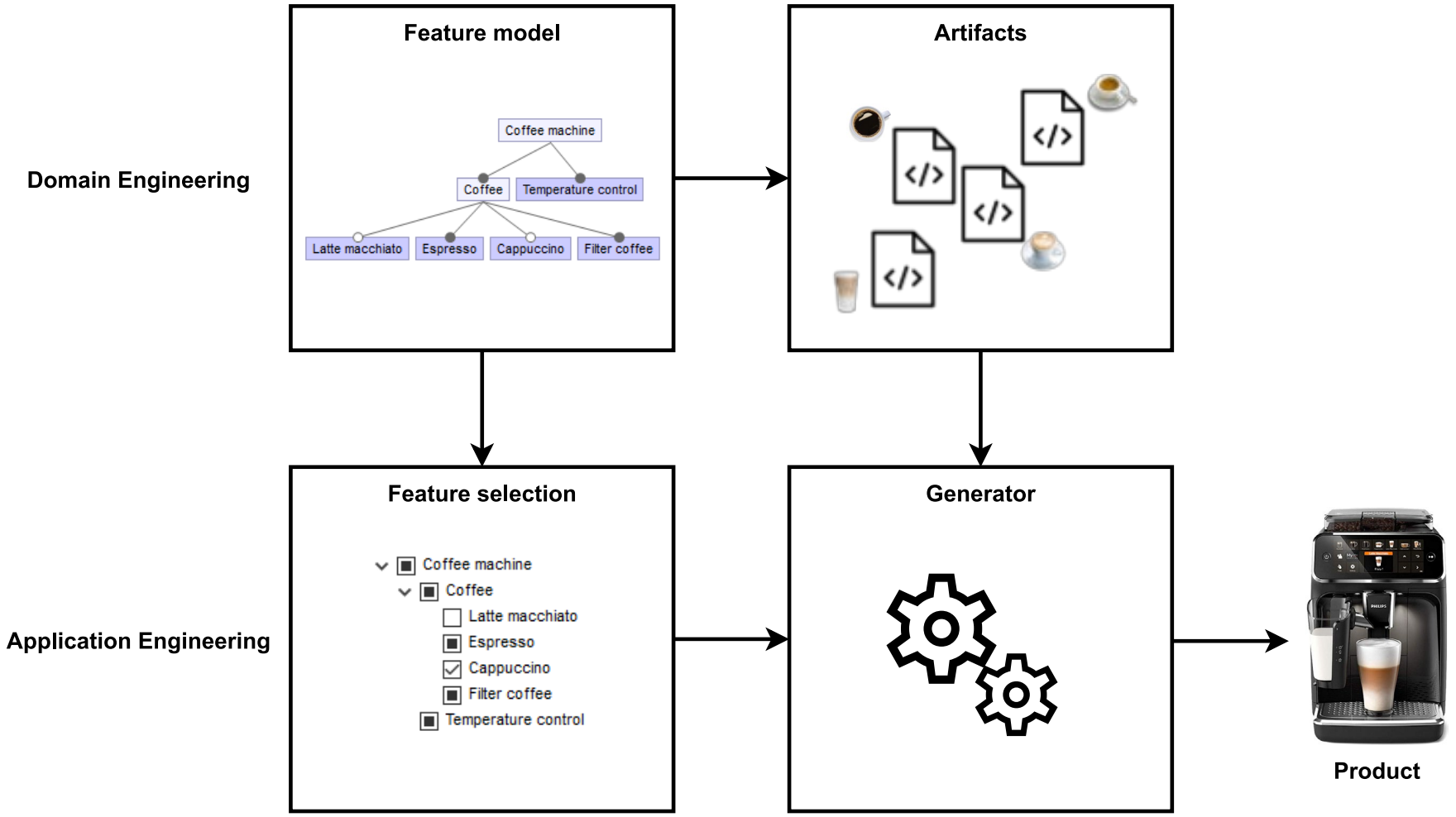
Technische Universität Braunschweig

Synchronizing software variants: A two-dimensional approach

Christoph König, Kamil Rosiak, Lukas Linsbauer, Ina Schaefer



Software Product Lines



Horizontal cloning

Basic

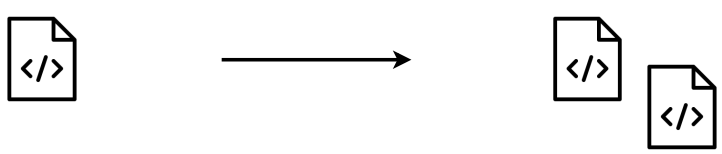
+ Filter coffee
+ Espresso
+ Temperature control

Pro 100

+ Filter coffee
+ Espresso
+ Cappuccino
+ Temperature control

Pro 250

+ Filter coffee
+ Espresso
+ Latte Macchiato
+ Cappuccino
+ Temperature control

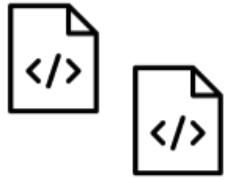


Vertical cloning

Cappuccino

1. Heat water
2. Heat milk
3. Grind coffee
4. Brew espresso
5. Foam milk
6. ...

Pro 250

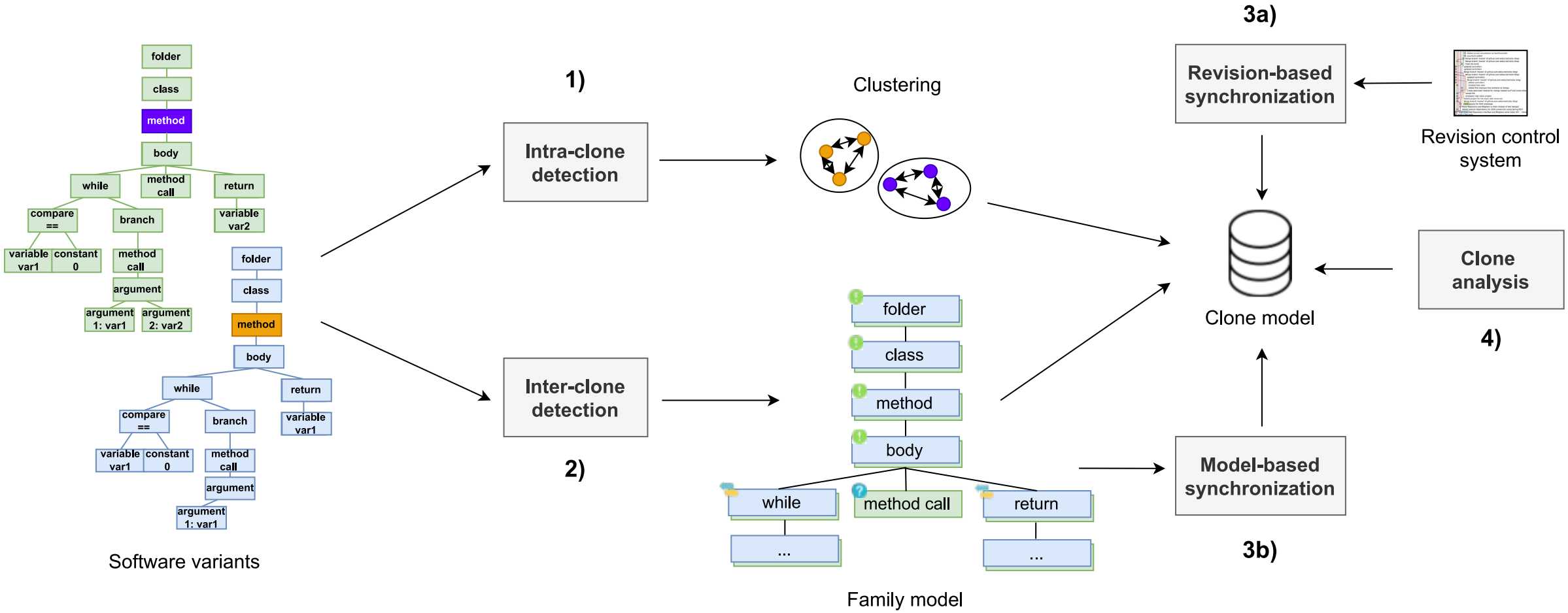


Latte Macchiato

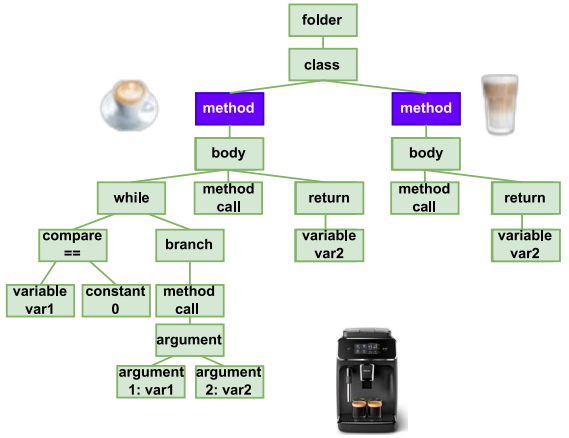
1. Heat water
2. Heat milk
3. Grind coffee
4. Foam milk
5. Brew espresso
6. ...

Synchronizing software variants

Overview

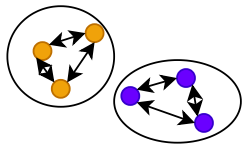

folder
class
method
body
while
method call
return
compare ==
branch
variable var2
variable var1
constant 0
method call
argument
argument 1: var1
argument 2: var2
Software variants
1)
Intra-clone detection
Clustering
Inter-clone detection
2)
folder
class
method
body
while
method call
return
...
...
Family model
3a)
Revision-based synchronization
Revision control system
Clone model
Clone analysis
4)
Model-based synchronization
3b)

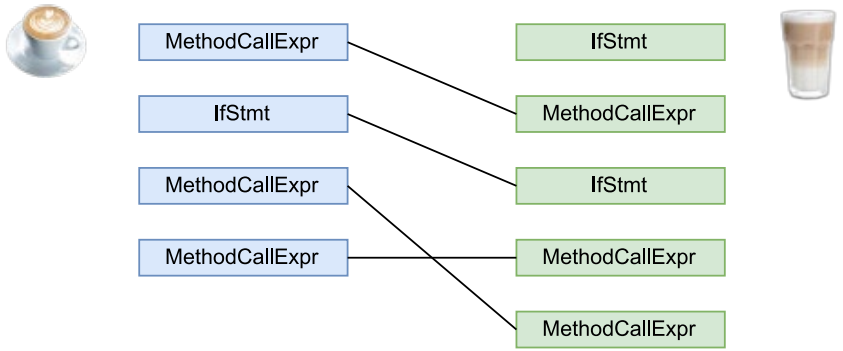
Intra-clone detection

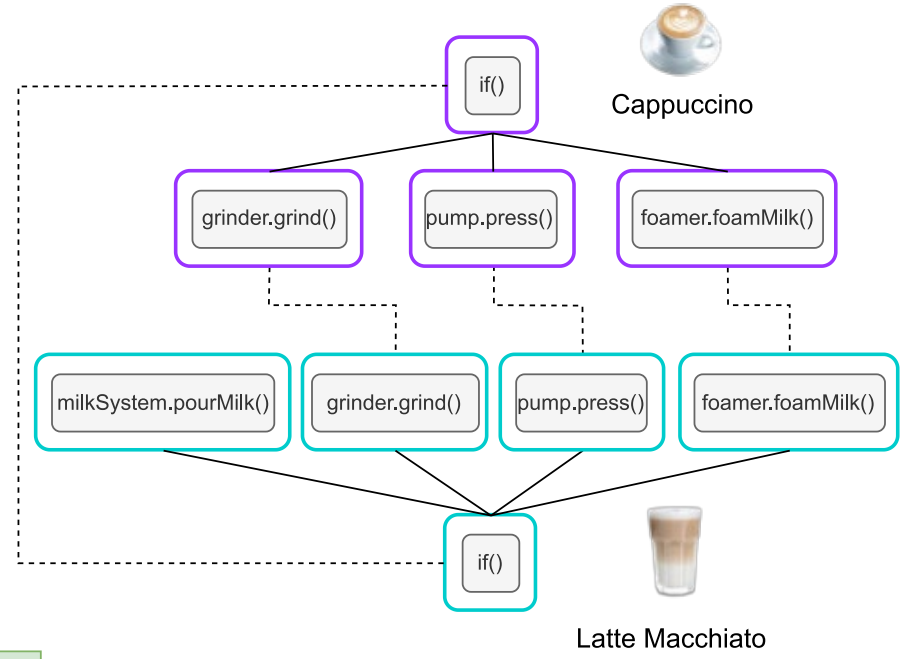


1) AST transformation and extraction

2) Clustering and matching

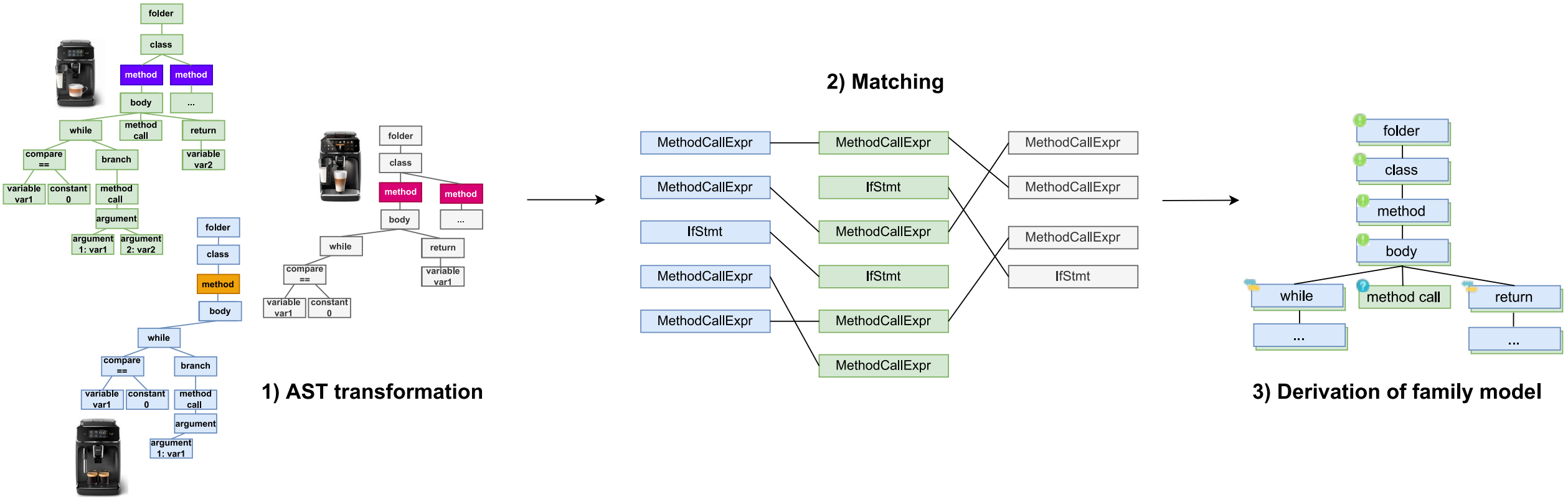




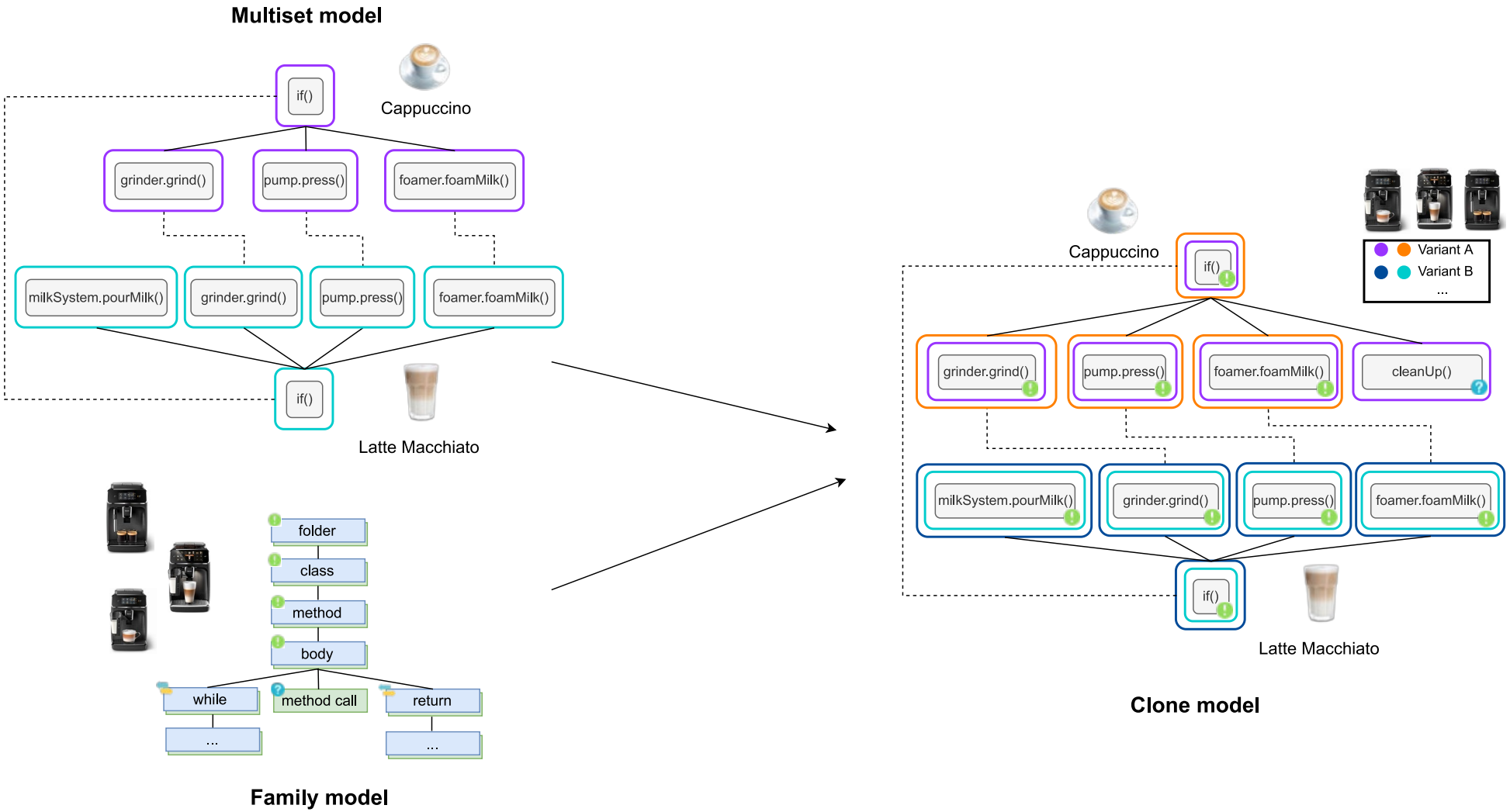


3) Derivation of multiset model

Inter-clone detection



Clone model


Multiset model
if()
Cappuccino
grinder.grind()
pump.press()
foamer.foamMilk()
milkSystem.pourMilk()
grinder.grind()
pump.press()
foamer.foamMilk()
if()
Latte Macchiato
folder
class
method
body
while
method call
return
...
...
Family model
Cappuccino
if()
Variant A
Variant B
...
grinder.grind()
pump.press()
foamer.foamMilk()
cleanUp()
milkSystem.pourMilk()
grinder.grind()
pump.press()
foamer.foamMilk()
if()
Latte Macchiato
Clone model

Evaluation

- **RQ1:** How relevant is two-dimensional clone synchronization?
- **RQ2:** How correctly can we synchronize intra- and inter-clones?
- **RQ3:** How does synchronization scale in time and space with increasing number of variants?

Evaluation

- **Experimental subject:** Five revisions of MobileMedia SPL[1]

- **Metrics**
  - Co-occurrence: Relationship between two inter-clones c_1 and c_2, where c_1 has at least one intra-clone c_{1a}
  - Subsequent revisions R_i and R_j ($i < j$)
    - True positives (TP): Clones manually synchronized in R_j and two-dimensionally in R_i
    - False negatives (FN): Clones manually synchronized in R_j but not two-dimensionally in R_i
    - False positives (FP): Clones not manually synchronized in R_j but two-dimensionally in R_i
    - Precision (PR): $\frac{TP}{TP + FP}$
    - Recall (RC): $\frac{TP}{TP + FN}$

[1] Eduardo Figueiredo, Nelio Cacho, Claudio Sant'Anna, Mario Monteiro, Uira Kulesza, Alessandro Garcia, Sergio Soares, Fabiano Ferrari, Safoora Khan, Fernando Castor Filho, and Francisco Dantas. Evolving software product lines with aspects

RQ1: Relevance

		#Intra-clones			#Inter-clones					
Revision	**#Variants**	T_I	T_{II}	T_{III}	T_I	T_{II}	T_{III}	**∅ Intra-group size**	**∅ Inter-group size**	**#Co-occurrence**
R_1	1	1	1	1	0	0	0	4.40	0	0
R_2	2	2	4	2	86	0	5	4.00	2.00	24
R_3	4	4	8	4	88	2	6	4.00	3.92	48
R_4	8	8	8	52	104	2	7	3.19	7.61	172
R_5	16	16	44	100	134	2	10	3.30	13.62	424

The number of intra- and inter-clones per clone type, the average group size and the number of co-occurrences in each revision of MobileMedia

RQ2: Correctness

	Intra-clones					Inter-clones				
	TP	FP	FN	PR	RC	TP	FP	FN	PR	RC
R_1	92	3	6	0.97	0.93	94	0	0	1.00	1.00
R_2	146	5	26	0.97	0.85	211	87	0	0.71	1.00
R_3	419	31	153	0.93	0.73	1308	352	4	0.78	0.99
R_4	404	111	28	0.79	0.94	2040	804	0	0.72	1.00

Precision and recall of variant synchronization between subsequent revisions of MobileMedia

Threats to validity

- Bias of matching algorithm
- MobileMedia as synthetic clone-and-own experimental subject
- Java source code only

Summary

- Approach: Synchronizing intra- and inter-clones simultaneously
- High precision and recall in change propagation
- Relevance of two-dimensional software variant synchronization