Safe Evolution Templates for Software Product Lines - Catalogue

Software Productivity Group

January 31, 2017

1 Templates for Compositional Product Lines
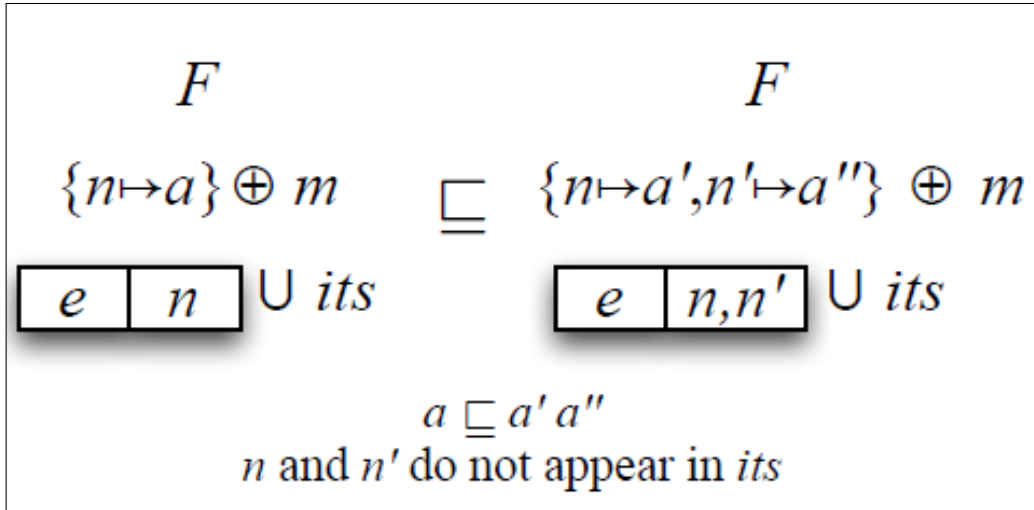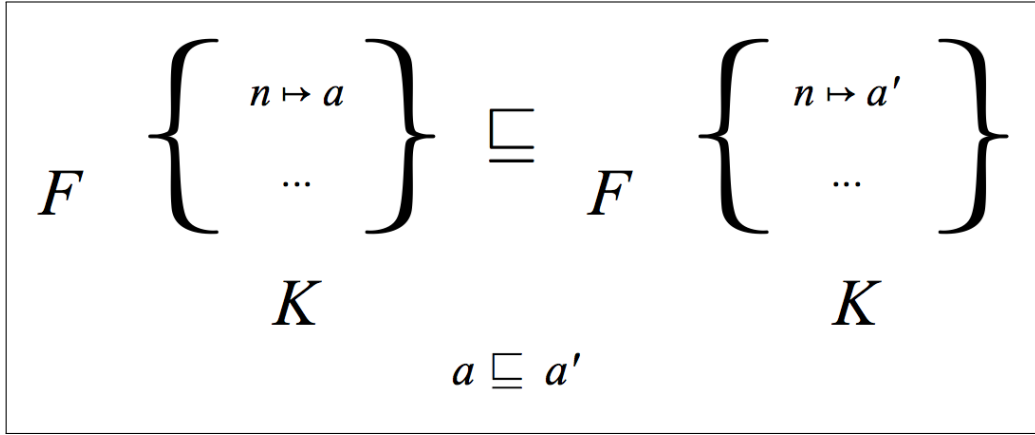

Figure 1: Split Asset (Taken from [1])

$$F \left\{ \begin{array}{c} n \mapsto a \\ \cdots \end{array} \right\} \quad \sqsubseteq \quad F \left\{ \begin{array}{c} n \mapsto a' \\ \cdots \end{array} \right\}$$

$$K \qquad\qquad\qquad K$$

$$a \sqsubseteq a'$$

Figure 2: Refine Asset (Taken from [1])

$$e' \Rightarrow O$$

O and *n'* are new

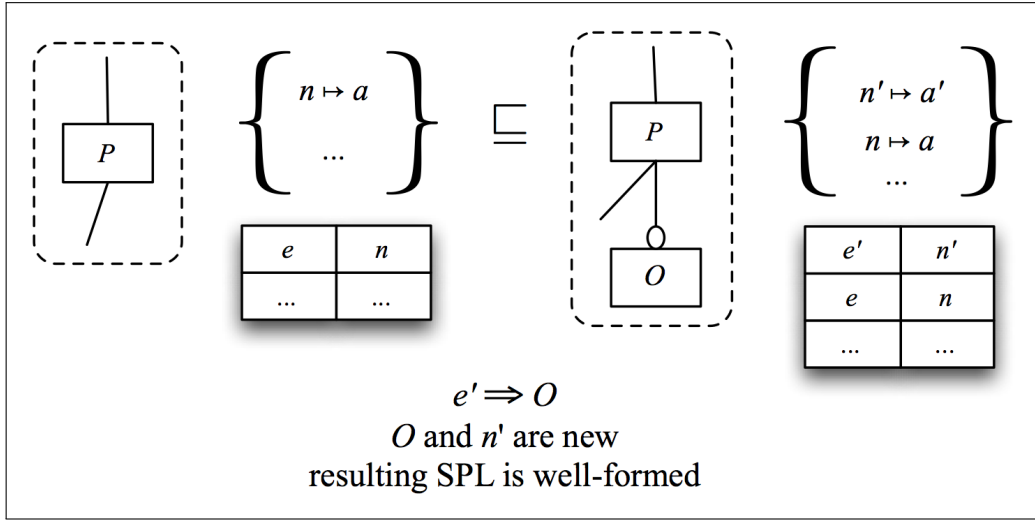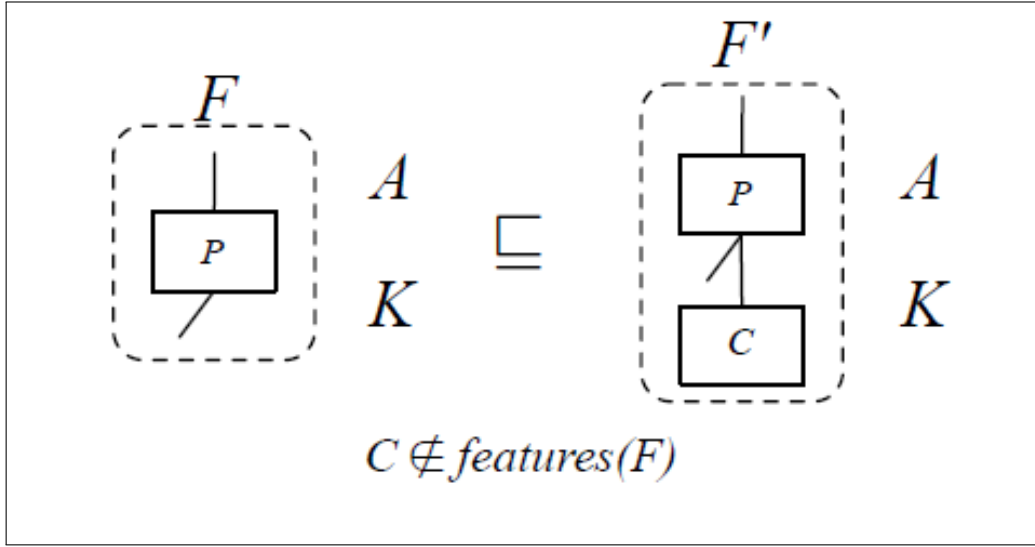resulting SPL is well-formed

Figure 3: Add New Optional Feature (Taken from [2])

Figure 4: Add any feature without changing the AM and CK (Taken from [1])

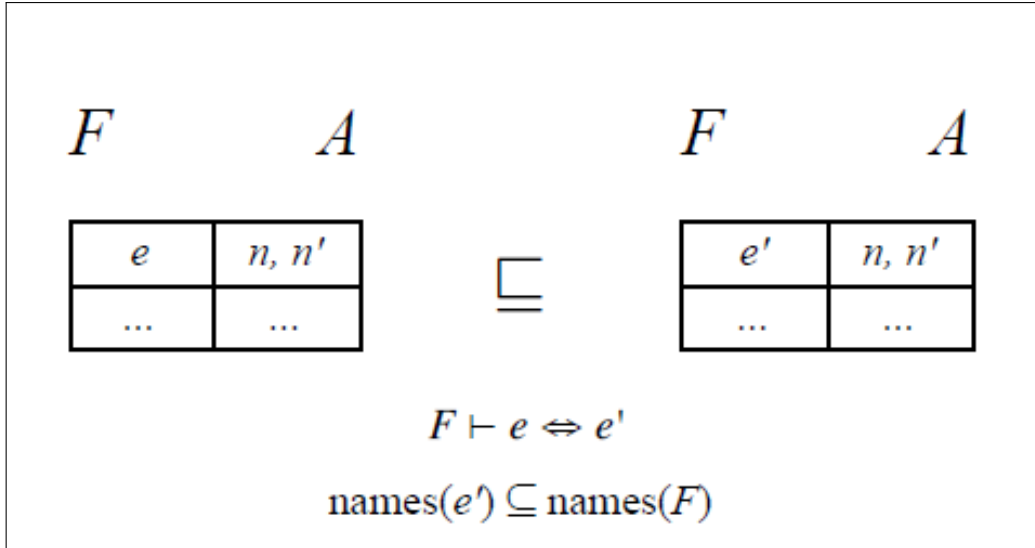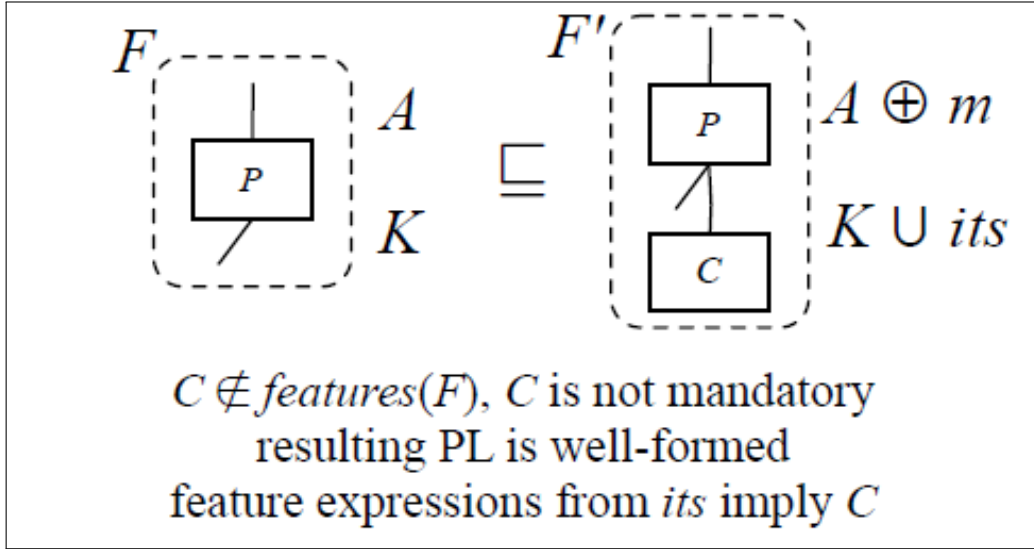Figure 5: Replace Feature Expression (Taken from [1])

F A F' $A \oplus m$

$\sqsubseteq$

K $K \cup its$

$C \notin features(F)$, C is not mandatory
resulting PL is well-formed
feature expressions from *its* imply C

Figure 6: Add variable feature with implementation (Taken from [1])

$$F \qquad\qquad F$$
$$A' \oplus m \quad \sqsubseteq \quad A'$$
$$K \qquad\qquad K$$

$\forall\, n \in dom(m) \bullet n$ does not appear in K
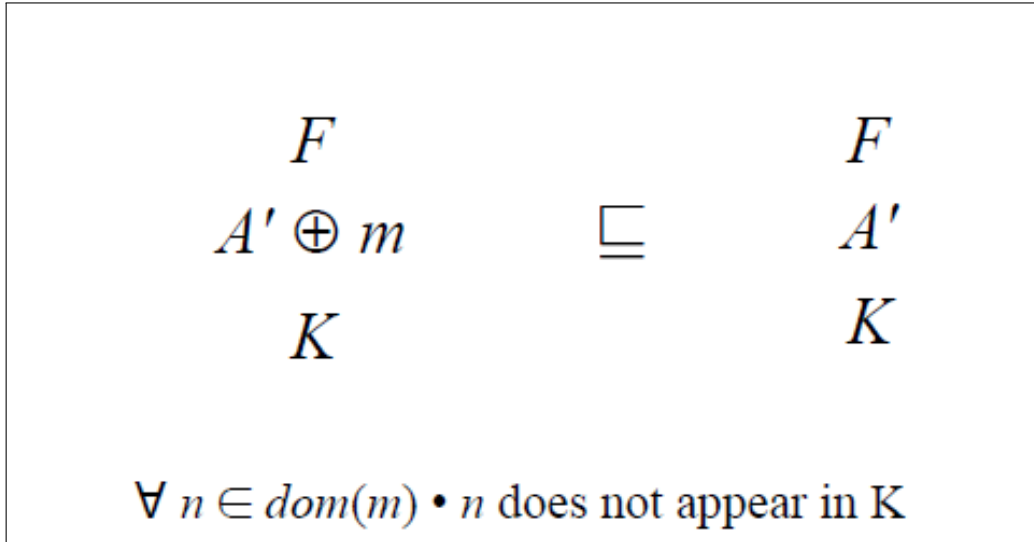
Figure 7: Remove Unused Assets (Taken from [1])

$$\begin{array}{ccc} F & & F \\ A & \sqsubseteq & A \oplus m \\ K & & K \end{array}$$

$$\forall\, n \in dom(m) \bullet n \text{ does not appear in K}$$
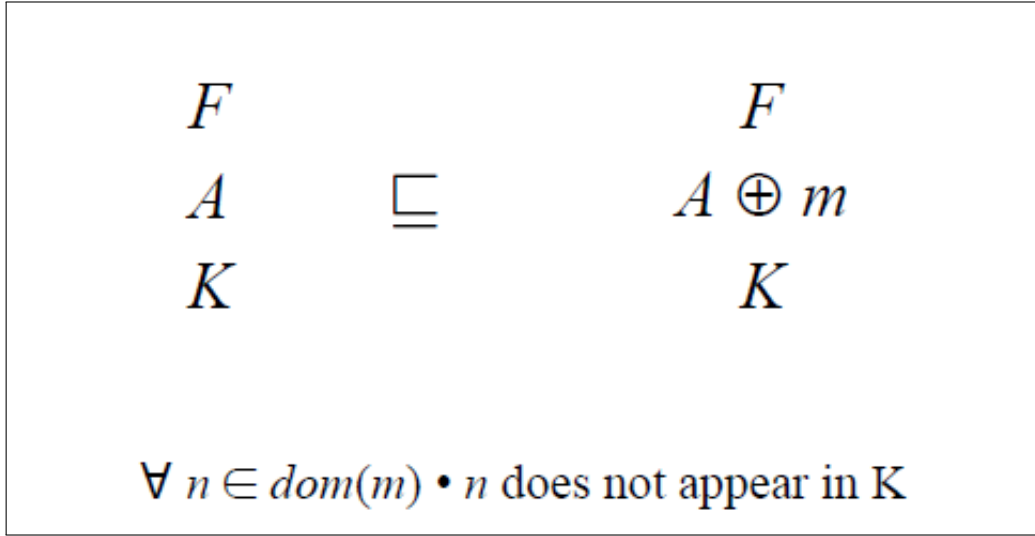
Figure 8: Add Unused Assets (Taken from [1])

$$\begin{array}{ccc} F & & F \\ \{n{\mapsto}a',n'{\mapsto}a''\} \oplus m & \sqsubseteq & \{n{\mapsto}a\} \oplus m \\ \boxed{e \mid n,n'} \cup its & & \boxed{e \mid n} \cup its \end{array}$$

$$a'\,a'' \sqsubseteq a$$
$$n \text{ and } n' \text{ do not appear in } its$$
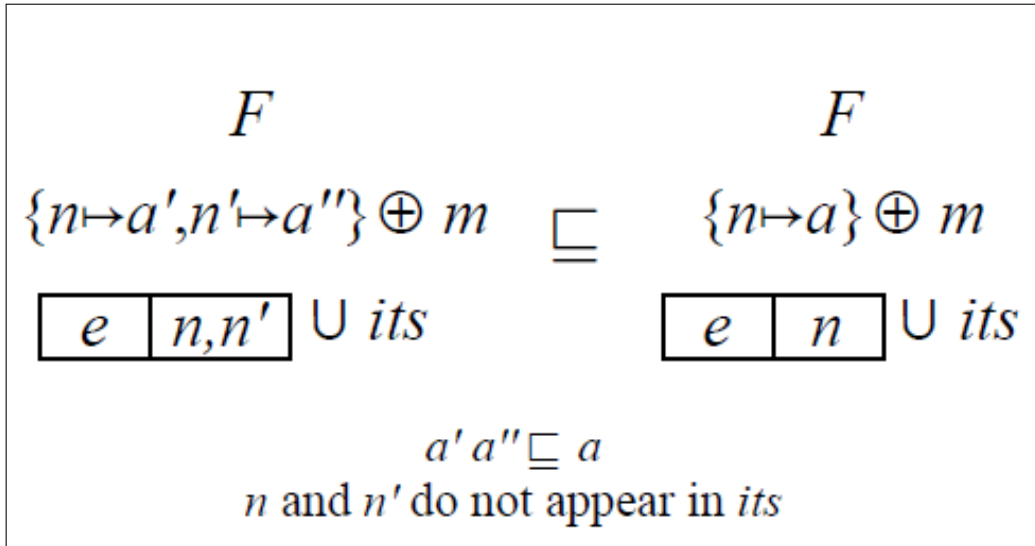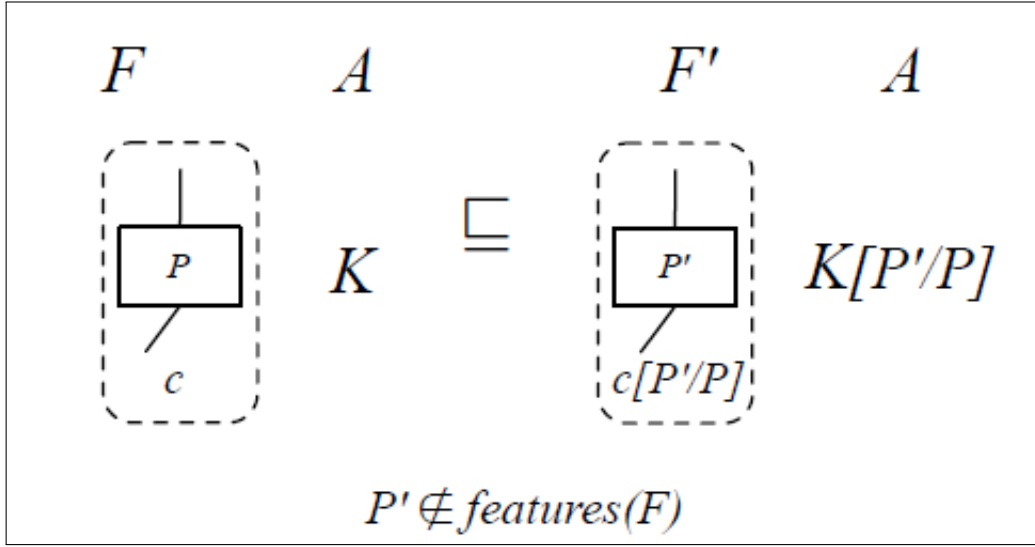
Figure 9: Merge Assets (Taken from [1])
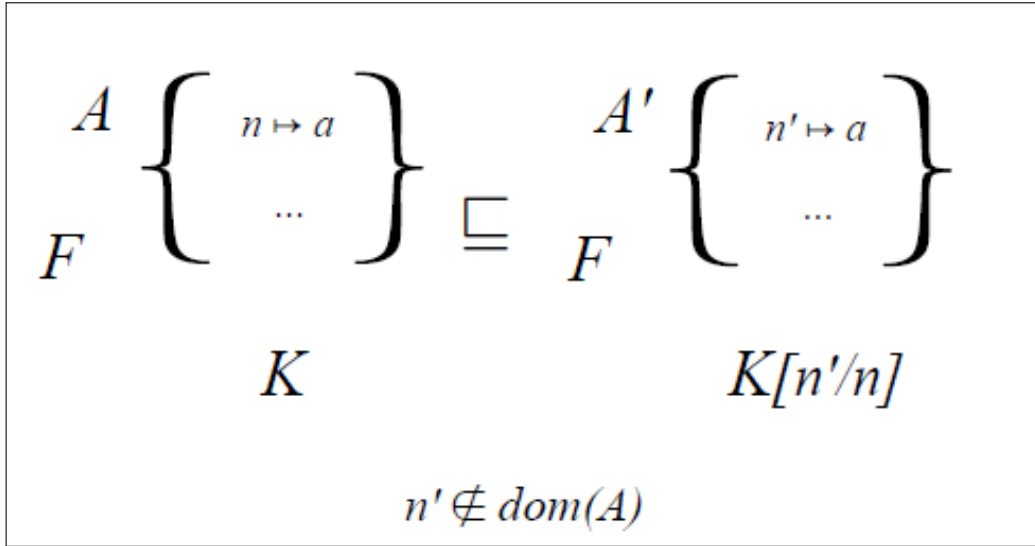
Figure 10: Feature Renaming (Taken from [1])

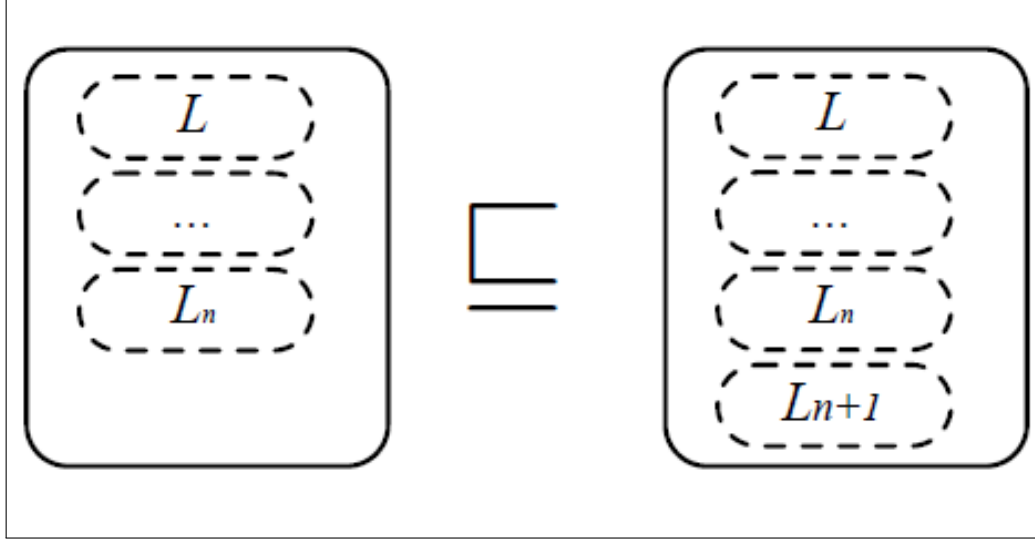Figure 11: Asset Name Renaming (Taken from [1])

Figure 12: Add Product Line (Taken from [1])

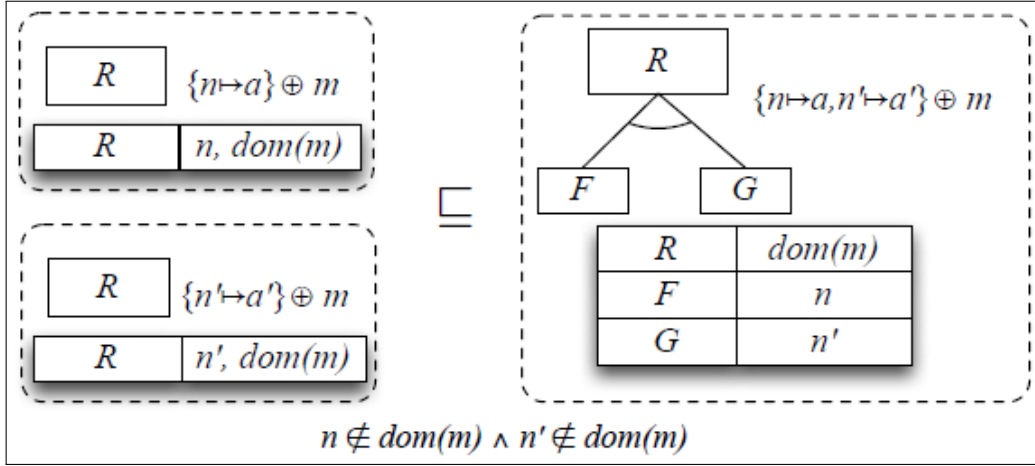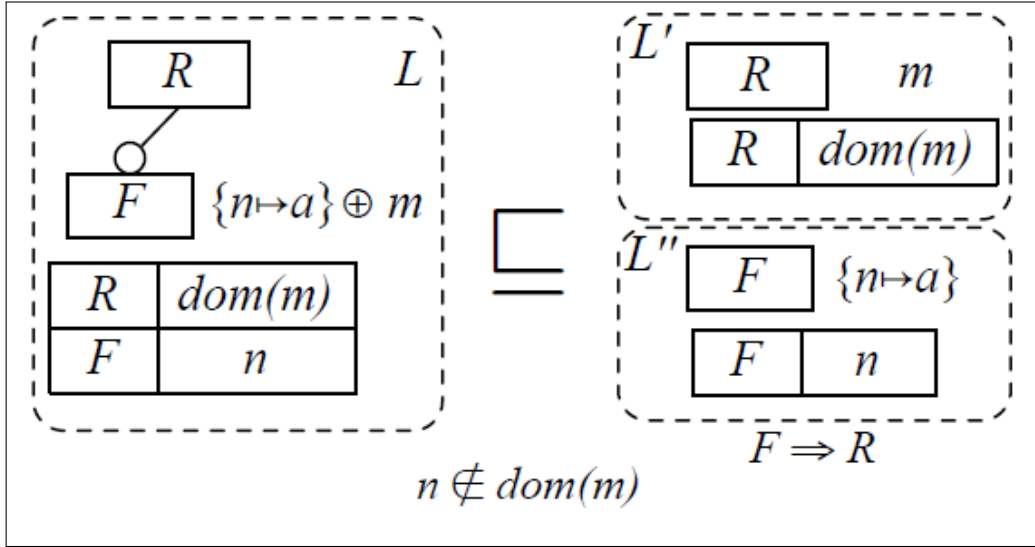Figure 13: Merge Products Into a PL (Taken from [1])

Figure 14: Derive Multi Product Line from a Product Line (Taken from [1])
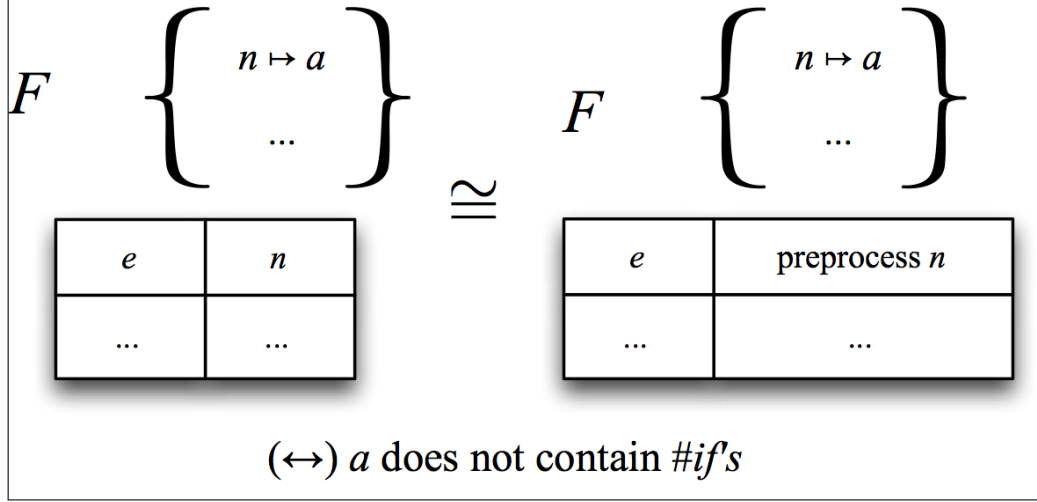
2 Templates for Annotative Product Lines

$$F \left\{ \begin{array}{c} n \mapsto a \\ \ldots \end{array} \right\}$$

e	n
...	...

$\cong$

$$F \left\{ \begin{array}{c} n \mapsto a \\ \ldots \end{array} \right\}$$

e	preprocess n
...	...

$(\leftrightarrow)$ a does not contain #*if's*

Figure 15: Preprocess Asset Without Preprocessor Directive (Taken from [2])

$$F \left\{ \begin{array}{c} n \mapsto a[\text{-}] \\ \ldots \end{array} \right\}$$

e	preprocess n
...	...

$\cong$

$$F \left\{ \begin{array}{c} n \mapsto a[\#\text{if } x\ c] \\ \ldots \end{array} \right\}$$

e	preprocess n
...	...

$(\leftrightarrow)$ n does not appear in other CK lines
$(\rightarrow)$ x is a new tag
$(\leftarrow)$ x does not appear in other CK lines
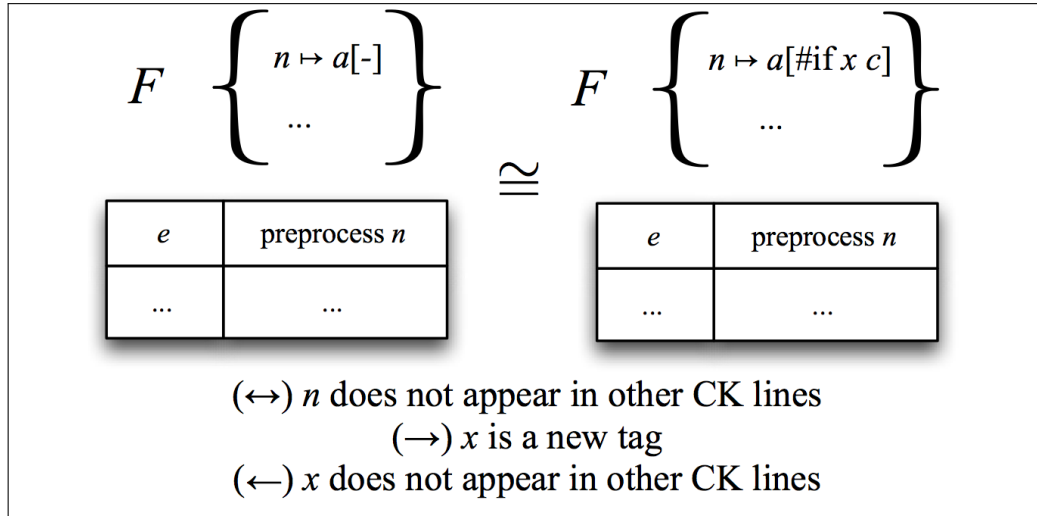
Figure 16: Add Dead Preprocessed Code (Taken from [2])

$$F \left\{ \begin{array}{c} n \mapsto a[c] \\ ... \end{array} \right\} \cong F \left\{ \begin{array}{c} n \mapsto a[\#\text{if } x\ c] \\ ... \end{array} \right\}$$

e	preprocess n
...	...

e	preprocess n, tag x
...	...

($\leftrightarrow$) n does not appear in other CK lines
($\rightarrow$) x is a new tag
($\leftarrow$) x does not appear in other CK lines
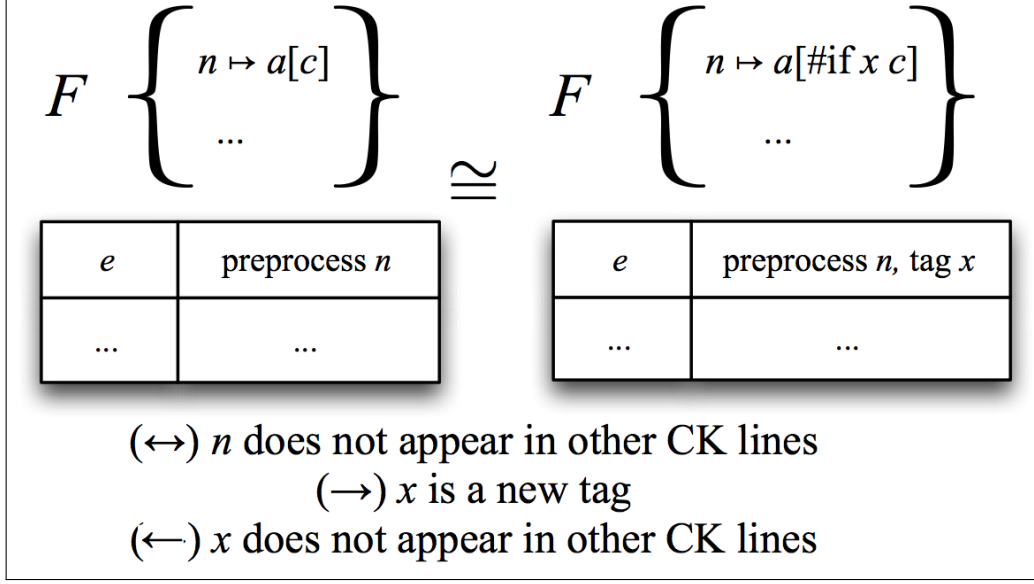
Figure 17: Add Harmless Preprocessing Directive (Taken from [2])

$$\left\{ \begin{array}{c} n \mapsto a[\text{-}] \\ ... \end{array} \right\} \sqsubseteq \left\{ \begin{array}{c} n \mapsto a[\#\text{if } x\ c] \\ ... \end{array} \right\}$$

e	preprocess n
...	...

e	preprocess n
e'	preprocess n tag x
...	...

$e' \Rightarrow F$
F is not mandatory
F and x are new
n does not appear in other CK lines
resulting SPL is well-formed
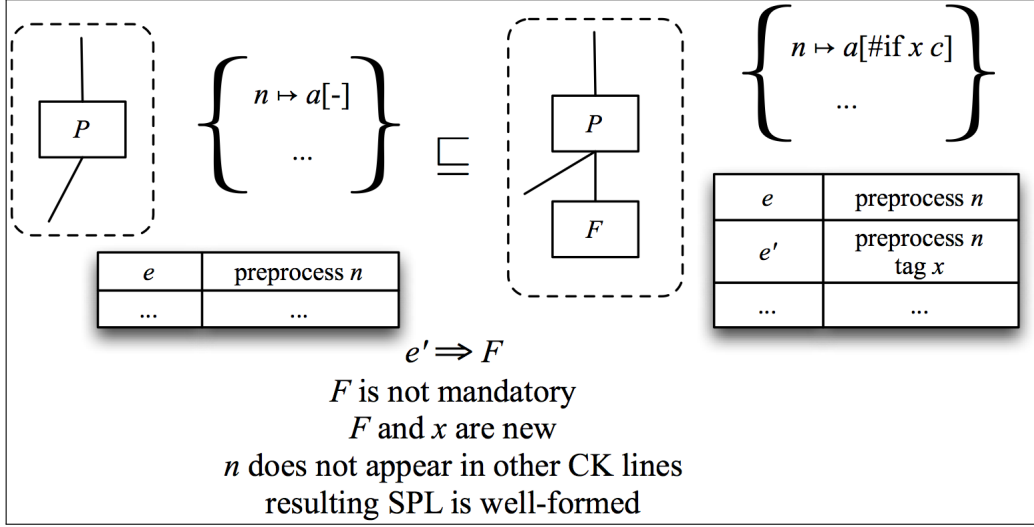
Figure 18: Add New Preprocessed Feature (Taken from [2])

$$F \left\{ \begin{array}{c} n \mapsto a[\#\text{if } x \, c] \\ ... \end{array} \right\}$$

e	preprocess n
e'	tag x
...	...

$\cong$

$$F \left\{ \begin{array}{c} n \mapsto a[\text{-}] \\ n' \mapsto a'[\#\text{if } x \, c] \\ ... \end{array} \right\}$$

e	preprocess n
e'	preprocess n' tag x
...	...

$(\leftrightarrow) \, e' \Rightarrow e$
$(\rightarrow) \, a[\#\text{if } x \, c] \sqsubseteq a[\text{-}] \, a'[\#\text{if } x \, c]$
$(\leftarrow) \, a[\text{-}] \, a'[\#\text{if } x \, c] \sqsubseteq a[\#\text{if } x \, c]$
$(\rightarrow) \, n'$ is new
$(\leftrightarrow) \, x, n$ do not appear in other CK lines
$(\leftrightarrow)$ resulting PL is well-formed
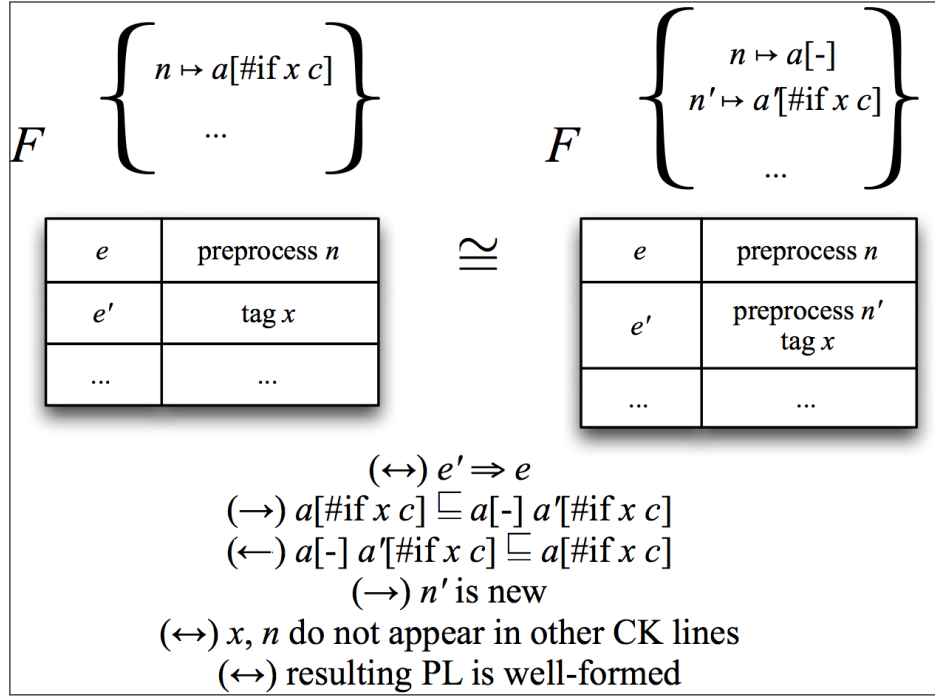
Figure 19: Extract Preprocessed Code (Taken from [2])

$$F \left\{ \begin{array}{c} n \mapsto a \\ n' \mapsto a' \\ ... \end{array} \right\}$$

e	n, n'
...	...

$\sqsubseteq$

$$F \left\{ \begin{array}{c} n \mapsto a \\ ... \end{array} \right\}$$

e	n
...	...

a' is not referred to in any other asset in the Asset Mapping

n' do not show up in other CK lines
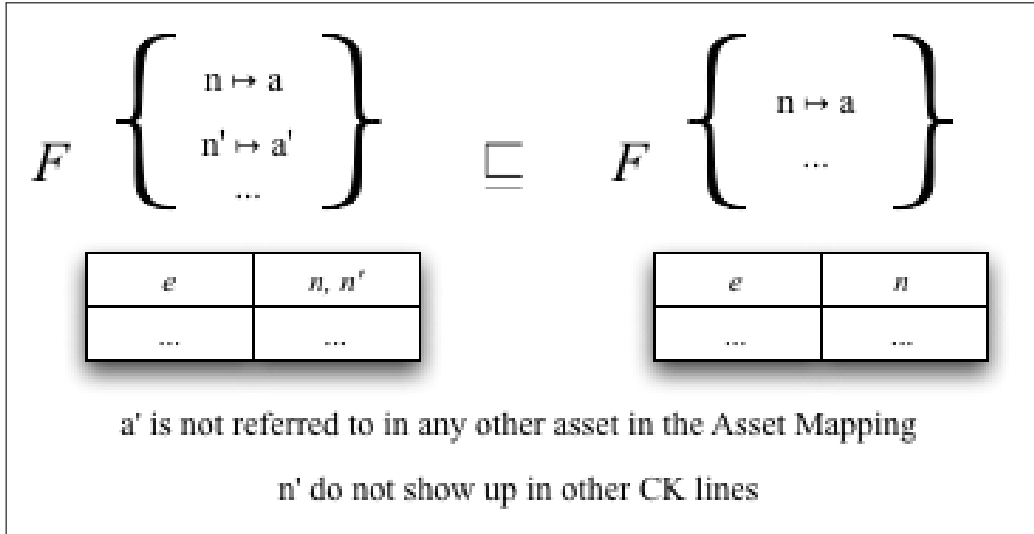
Figure 20: Delete Asset (Taken from [2])
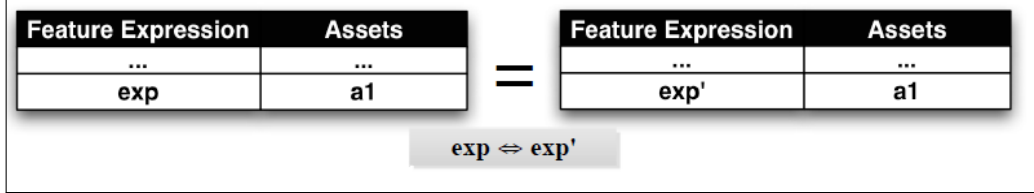
3   Templates for CK Refactoring

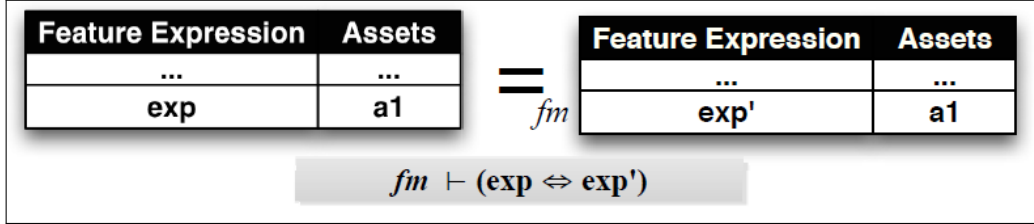Figure 21: Simplify Feature Expression using Propositional Reasoning (Taken from [3])

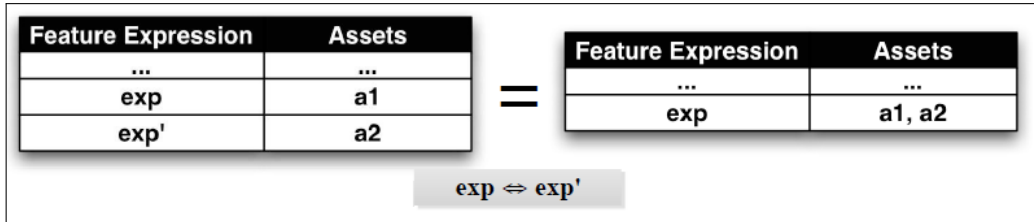Figure 22: Simplify Feature Expression using the FM (Taken from [3])

Figure 23: Merge items with propositionally equivalent feature expressions (Taken from [3])
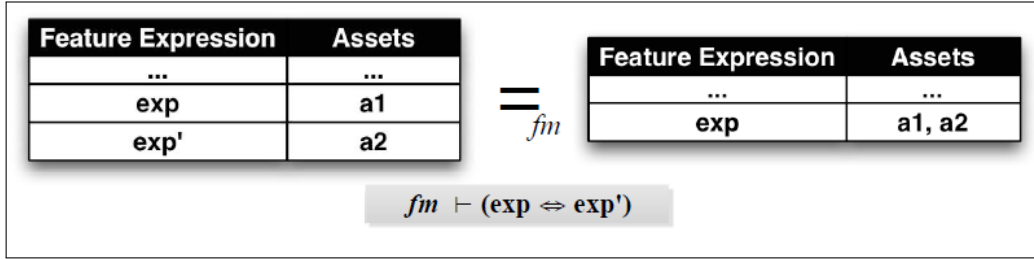
Feature Expression	Assets
...	...
exp	a1
exp'	a2

$=_{fm}$

Feature Expression	Assets
...	...
exp	a1, a2

$$fm \vdash (exp \Leftrightarrow exp')$$

Figure 24: Merge items with equivalent feature expressions by the FM (Taken from [3])

Feature Expression	Assets
...	...
exp	a1
exp'	a1

$=$

Feature Expression	Assets
...	...
exp ∨ exp'	a1

Figure 25: Duplicated Assets (Taken from [3])

Feature Expression	Assets
...	...
exp'	a1
exp	a2

$=_{fm}$

F	Assets
...	...
exp	a1

$$fm \vdash \neg exp$$
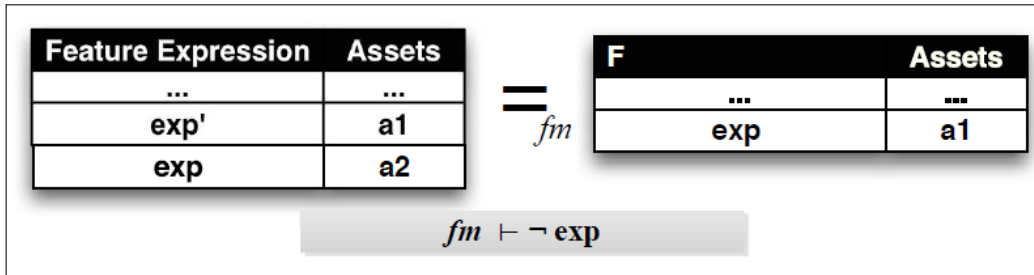
Figure 26: Dead Feature Expression (Taken from [3])

Feature Expression	Assets
...	...
exp	a1
exp'	a2

$=$

F	Assets
...	...
exp'	a2
exp	a1
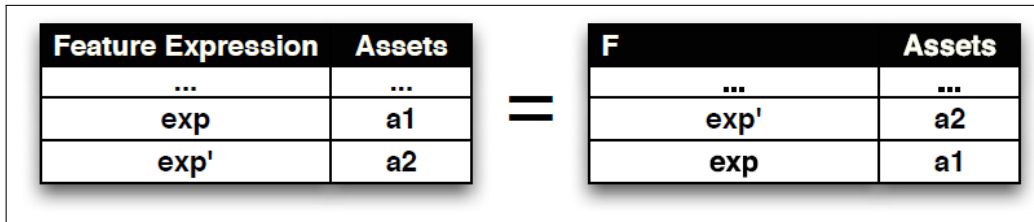
Figure 27: Change Order (Taken from [3])

4   Templates for FM Refactoring
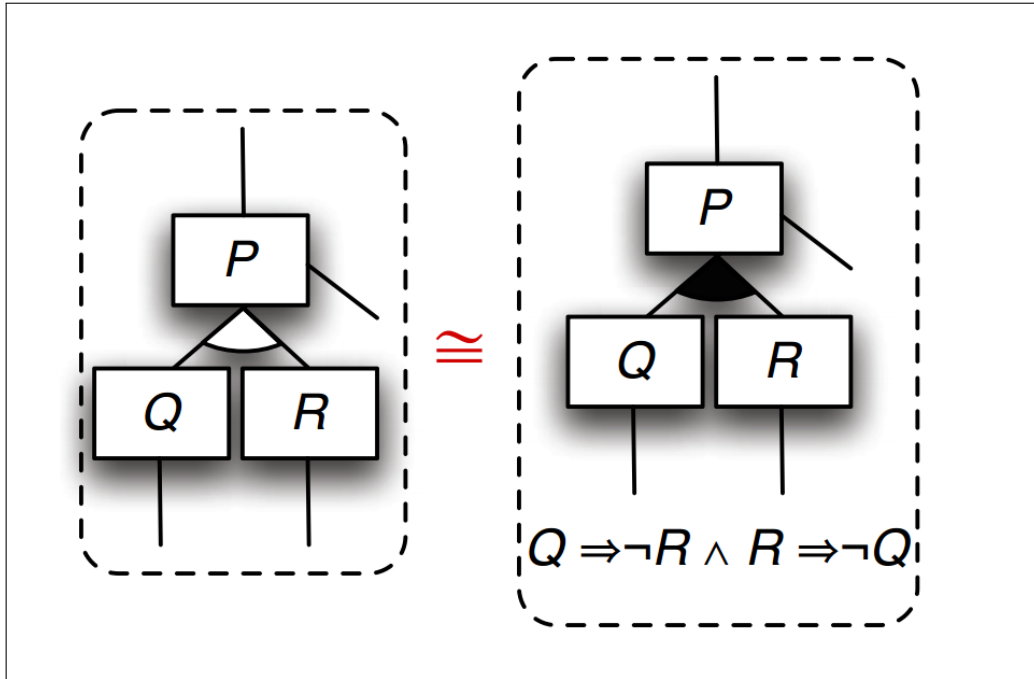
Figure 28: Replace Alternative Equivalence Template (Taken from [4])
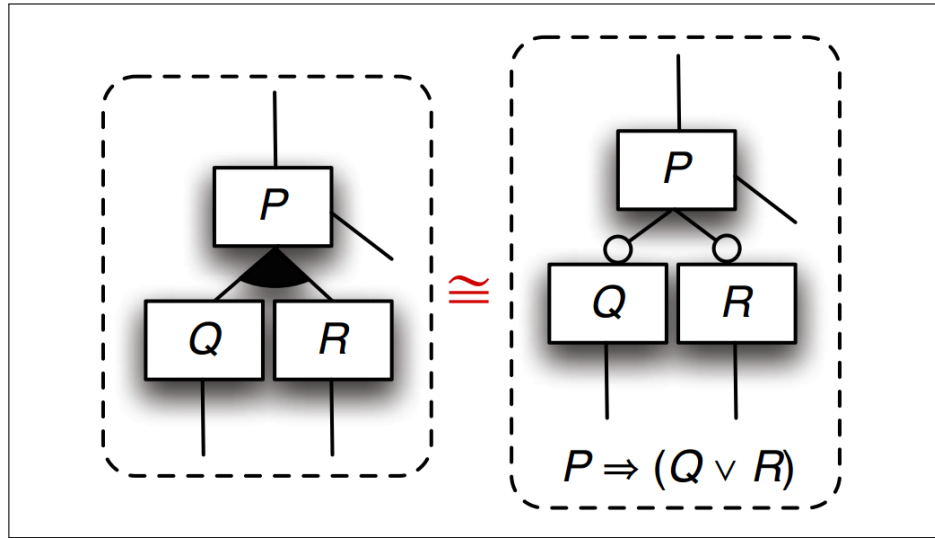
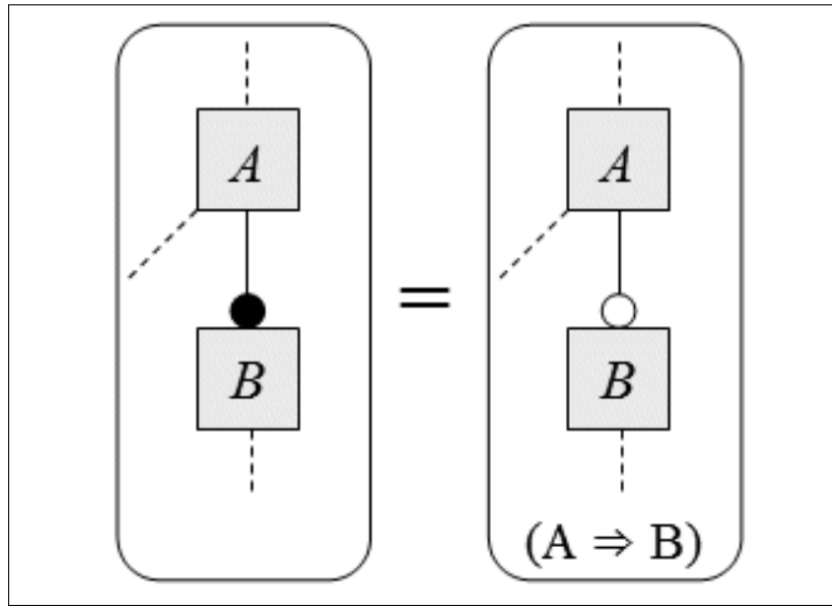Figure 29: Replace Or Equivalence Template (Taken from [4])
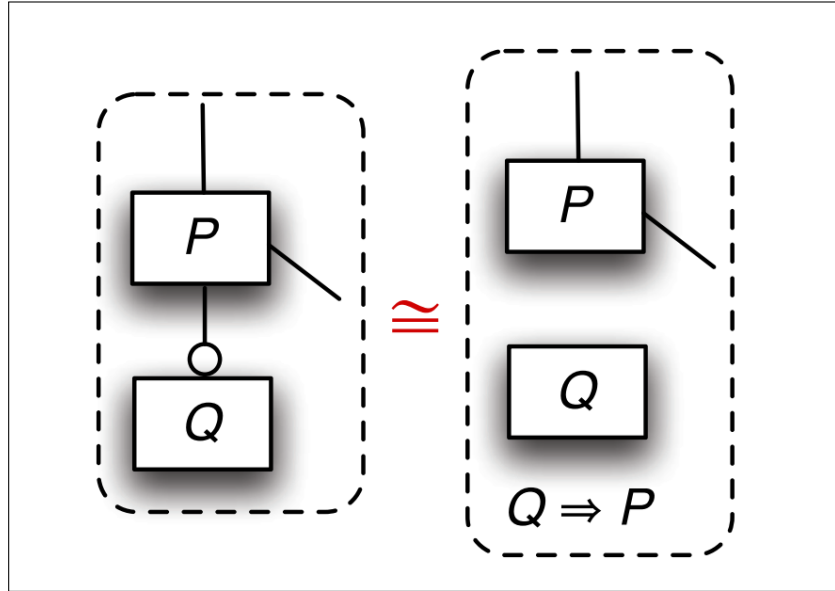

Figure 30: Replace Mandatory (Taken from [5])

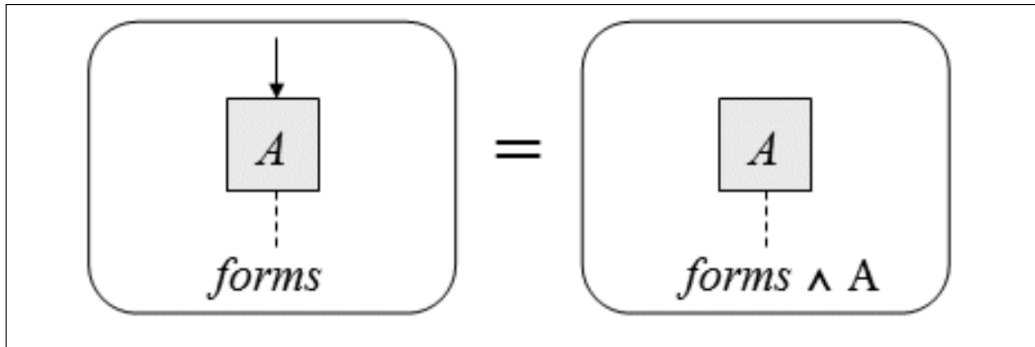Figure 31: Remove Optional Equivalence Template (Taken from [4])

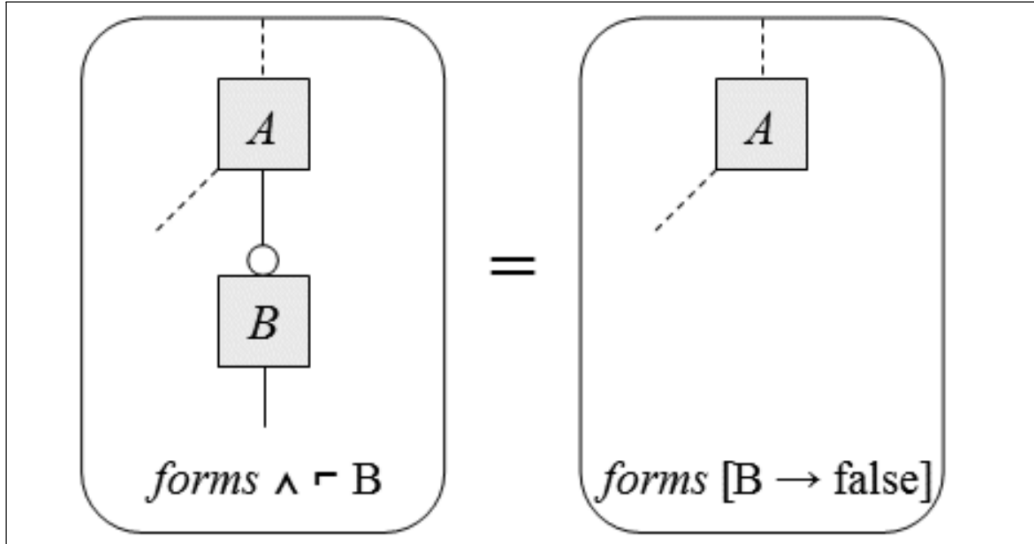Figure 32: Remove Root (Taken from [5])
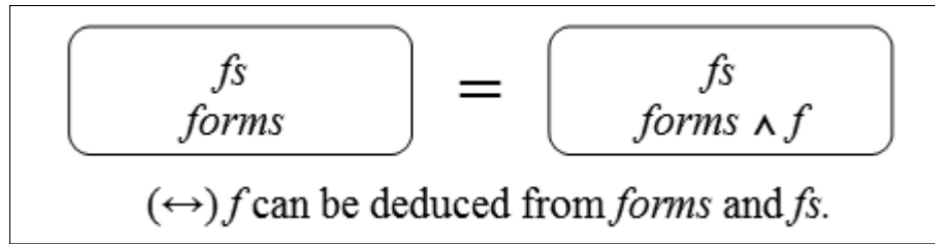
Figure 33: Remove Node (Taken from [5])

Figure 34: Add Formula (Taken from [5])

References

[1] Phd leopoldo. `http://twiki.cin.ufpe.br/twiki/pub/SPG/GenteAreaThesis/lmt-phd-thesis.pdf`. [Online; accessed 10-September-2015].

[2] Twiki refactoring templates. `http://twiki.cin.ufpe.br/twiki/bin/view/SPG/SPLRefactoringTemplates`. [Online; accessed 10-September-2015].

[3] Msc leopoldo. `http://twiki.cin.ufpe.br/twiki/pub/SPG/GenteAreaThesis/lmt-msc-thesis.pdf`. [Online; accessed 10-September-2015].

[4] An introduction to software product line refactoring - gttse 2009, lncs 6491, pp. 1–26, 2011. `http://link.springer.com/chapter/10.1007%2F978-3-642-18023-1_1`. [Online; accessed 15-September-2015].

[5] Algebraic laws for feature models - jucs vol. 14, no. 21 (2008). `http://www.jucs.org/jucs_14_21/algebraic_laws_for_feature/jucs_14_21_3573_3591_gheyi.pdf`. [Online; accessed 14-September-2015].