Investigating the Safe Evolution of Software Product Lines

Laís Neves, Leopoldo Teixeira, Demóstentes Sena, Vander Alves, Uirá Kulesza and Paulo Borba

lmn3@cin.ufpe.br
SPL evolution scenario

New field introduced by ExcelPreferencesAspect

New test suite format

New field introduced by ExcelPreferencesAspect

<table>
<thead>
<tr>
<th>Component Keyword:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Case ID:</td>
</tr>
<tr>
<td>Test Case Initial ID:</td>
</tr>
<tr>
<td>Empty Field:</td>
</tr>
<tr>
<td>Objective Prefix:</td>
</tr>
<tr>
<td>Print Use Case Description:</td>
</tr>
<tr>
<td>Print Flow Description:</td>
</tr>
<tr>
<td>□ Keep Requirements</td>
</tr>
</tbody>
</table>

Cancel  OK
Scenario implementation

Error when updating CK!

Component Keyword field present in all products!
Problems

manually extracting and changing different parts of the code when evolving a SPL is error-prone

usually involves the analysis of many artifacts

bugs introduced during this process can be difficult to track
Our proposal

✓ discover and analyze concrete SPL evolution scenarios

✓ describe a set of safe evolution templates for SPLs

✓ transformations that deal with CK and FM in addition to SPL assets
Agenda

- SPL refinement
- SPL safe evolution templates
- Evaluation
- Concluding remarks
SPL refinement (safe evolution)
Safe Evolution
Templates for SPL
TaRGeT SPL evolution

11 safe evolution scenarios

- **Release 4.0**
  - Jan '09

- **Release 5.0**
  - Jul '09

- **Release 6.0**
  - Jan '10
Templates’ discovery

Analysis of changes in each scenario

Templates abstract, generalize and factorize the scenarios
Recalling...
Add new optional feature template

\[ e' \Rightarrow O \]

\( O \) and \( n' \) are new resulting SPL is well-formed
Using the template...
Another example...

**TaRGeT**

Interruption

<table>
<thead>
<tr>
<th>TaRGeT</th>
<th>TargetProjectRefresher</th>
<th>FeaturesView</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FeaturesTreeViewLabelProvider</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Java Files**

PMInterruptionAspect.aj

**TargetProjectRefresher**

FeaturesView

FeaturesTreeViewLabelProvider

PMInterruptionAspect
Split asset template

\[ F \left\{ \begin{array}{c|c} n \mapsto a \\ \vdots \end{array} \right\} \subseteq \begin{array}{c|c} e & n, n' \\ \vdots & \vdots \end{array} \]

\[ \begin{array}{c|c} e & n \\ \vdots & \vdots \end{array} \]

\[ a \subseteq a' a'' \]

\[ n' \text{ is new} \]

\[ n \text{ does not appear in other CK lines} \]
SPL safe evolution templates

<table>
<thead>
<tr>
<th>Template</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Split Asset</td>
</tr>
<tr>
<td>2</td>
<td>Refine Asset</td>
</tr>
<tr>
<td>3</td>
<td>Add New Optional Feature</td>
</tr>
<tr>
<td>4</td>
<td>Add New Mandatory Feature</td>
</tr>
<tr>
<td>5</td>
<td>Replace Feature Expression</td>
</tr>
<tr>
<td>6</td>
<td>Add New Alternative Feature</td>
</tr>
<tr>
<td>7</td>
<td>Add New Or Feature</td>
</tr>
<tr>
<td>8</td>
<td>Delete Asset</td>
</tr>
</tbody>
</table>
Evaluation
SPLs analyzed

**TaRGeT**

- Automatic test generation
- 42 features
- ~32 KLOC

**MobileMedia**

- Media management on mobile devices
- 12 features
- ~3 KLOC

**Main goal:** to evaluate the templates’ expressiveness
TaRGeT SPL

- 11 safe evolution scenarios
- 8 minor releases
- Templates' discovery
TaRGeT SPL

- Templates' evaluation
- 9 safe evolution scenarios
- 2 minor releases

Release 4.0
Jan '09

Release 5.0
Jul '09

Release 6.0
Jan '10
MobileMedia SPL

- 2 safe evolution scenarios
- 1 safe evolution scenario
- 3 safe evolution scenarios
- 2 safe evolution scenarios

Release 4
Release 5
Release 6
Release 7
Release 8
## Templates frequency

<table>
<thead>
<tr>
<th>Template</th>
<th>TaRGeT</th>
<th>MobileMedia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Split Asset</td>
<td>4 (13.7%)</td>
<td>6 (25%)</td>
</tr>
<tr>
<td>Refine Asset</td>
<td>5 (17.2%)</td>
<td>6 (25%)</td>
</tr>
<tr>
<td>Add New Optional Feature</td>
<td>5 (17.2%)</td>
<td>4 (16.6%)</td>
</tr>
<tr>
<td>Add New Mandatory Feature</td>
<td>3 (10.3%)</td>
<td>1 (4.16%)</td>
</tr>
<tr>
<td>Replace Feature Expression</td>
<td>3 (10.3%)</td>
<td>4 (16.6%)</td>
</tr>
<tr>
<td>Add New Alternative Feature</td>
<td>8 (27.5%)</td>
<td>2 (8.3%)</td>
</tr>
<tr>
<td>Add New Or Feature</td>
<td>1 (3.4%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Delete Asset</td>
<td>0 (0%)</td>
<td>1 (4.16%)</td>
</tr>
</tbody>
</table>

2 scenarios introduced bugs
4 scenarios introduced bugs
Threats to validity

- Some evolution scenarios not considered
- Overlooked errors introduced by manual changes
- MobileMedia is a small benchmark
Concluding remarks

- **28** evolution scenarios identified
- **6** of them introduced bugs
- **8** safe evolution templates proposed

**soundness** proved for all templates

templates set is **expressive** regarding the analyzed SPLs
Concluding remarks

SPL safe evolution templates

- useful to guide developers to perform safe modifications
- preserve existing products behavior and may prevent errors
- templates automation can improve productivity
Investigating the Safe Evolution of Software Product Lines

Laís Neves, Leopoldo Teixeira, Demóstentes Sena, Vander Alves, Uirá Kulesza and Paulo Borba

lmn3@cin.ufpe.br