Evaluating the Benefits of Using Domain-Specific Modeling Languages

An Experience Report

Timo Wegeler*, Friederike Gutzeit*, Aurèle Destailleur*, Bernhard Dock²

*Fraunhofer Institute for Open Communications Systems FOKUS, Embedded Systems Quality Management
²Klopotek & Partner GmbH
PROJECT SETTING

Fraunhofer FOKUS
TU Berlin, DIMA
adesso AG
akquinet tech@spree GmbH
Cedavis Technology GmbH
ClinPath GmbH
eTASK Service-Management GmbH
Klopotek & Partner GmbH
Model Labs GmbH
PI Informatik GmbH

09/2010 – 08/2013
PROJECT GOALS

- Industrial Partners:
  - Develop DSLs for their domains
  - Identify demands for tools and guidance
  - Improve the software development process
  - Achieve a better management of complexity
  - Enable quicker reaction to customer’s demands
  - Business growth

- Academic Partners
  - Help identifying demands and provide guidance
  - Analyze and generalize
  - Identify research issues and provide solutions
SHARED DSL DEVELOPMENT EXPERIENCES

• “Brown field development”: No one starts from scratch
  • Usage of tools at hand
  • Some kind of notation already exists
• Need to integrate into existing toolkits
• Need to involve Customers directly and indirectly
• Concrete Syntax matters, but is adaptable

• Most challenging aspects:
  • Finding the right abstractions
  • Finding an appropriate syntax
  • Collaboration with customers or other people
  • Lack of evolution support
EVALUATION OF DSL DEVELOPMENTS

• How to evaluate
• Which aspects to take into account
• Use of metrics
• Use of feedback
• Quantitative vs. Qualitative

• Assessment of motivation
• Qualitative interviews
• Validation of DSL design issues
• Quantify benefits
• Comparison of impacted workflows
EVALUATION DIFFICULTIES

- Moving targets
  - Experimental DSLs
  - Shifting Goals
- Late start
- How to compare alternative developments (w/wo DSL)
- Method not clear in advance

=> quite experimental
=> not covering all aspects at all DSLs

- Two examples:
  - Graphical DSL
  - Textual DSL
EVALUATING THE TEXTUAL DSL
EVALUATING THE TEXTUAL DSL

Figure 3: LOC for each generator in relation to the original DSL LOC
EVALUATING THE TEXTUAL DSL

- Interactive Visualization of LOC relations of all DSL scripts in productive use
EVALUATING THE GRAPHICAL DSL

Figure 4: Graphical DSL implemented with Visual Studio DSL Tools

<table>
<thead>
<tr>
<th></th>
<th>script 1</th>
<th>script 2</th>
<th>script 3</th>
<th>script 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>graphical elements used (max: 14)</td>
<td>14</td>
<td>11</td>
<td>6</td>
<td>14</td>
</tr>
<tr>
<td>total elements used in diagram</td>
<td>75</td>
<td>112</td>
<td>17</td>
<td>111</td>
</tr>
<tr>
<td>total properties defined</td>
<td>394</td>
<td>542</td>
<td>68</td>
<td>517</td>
</tr>
<tr>
<td>LOC generated</td>
<td>5075</td>
<td>7108</td>
<td>614</td>
<td>7158</td>
</tr>
</tbody>
</table>

Table 1: Model Metrics
EVALUATING THE GRAPHICAL DSL

- Concrete Syntax Evaluation
  - 14 items used
  - 8 groups of items with similar appearance
  - Semantic net providing information about items’ meaning
  - Detect an anomaly based on similarity groups and distances in the semantic net

- Work in progress
CONCLUSIONS

• Different qualitative and quantitative criteria can be taken into account when evaluating DSM solutions
• Amount of criteria depends on type of DSL
  • Includes motivation and goals
  • Some aspects are hard to measure, but important

• DSL evolution aspects were neglected

• More research needed

• More concrete DSL examples wanted for evaluation purposes