Business Process Lines to develop Service-Oriented Architectures through the Software Product Lines paradigm

Nicola Boffoli, Danilo Caivano, Daniela Castelluccia, Fabrizio Maria Maggi, Giuseppe Visaggio
SERLAB - Department of Informatics
University of Bari - Italy
{boffoli, caivano, castelluccia, maggi, visaggio}@di.uniba.it
Outline

SPL + SOA
- Why?
- What?
- How?

Our proposal
- Business Process Line
- Decision Models
- Case Study
SPL + SOA: Why?

Two common perspectives

- **Software reuse**
  - implementing new software systems reusing existing software resources rather than developing the same software capabilities again

- **Software flexibility**
  - allowing to adapt the systems to the different customers of a whole market segment
    - **SPL** focuses on the commonality and variability to build a set of software products
    - **SOA** allows to compose, orchestrate and maintain solutions based on services, implementing business processes
SPL + SOA: What?

Our Proposal
- transferring peculiarities/advantages from SPL to SOA
- build a SOA systems line suitable to customers or market segments needs in a specific application domain
SPL + SOA: How?

We start from a deep analysis of the business processes identifying in them commonality and variability typical of the SPL paradigm.

Business Process Line + Decision Models
Business Process Line (BPL)

A BPL realizes processes able to adapt themselves to different customer needs

- Each process of a BPL can be then transformed into the corresponding SOA system
  - If the business processes are adaptable to the customer needs
  - then the generated SOA system, it will result in its turn suitable to the specific customer requirements
From SPL to BPL: Analogies and Tailoring ...

SPL

- Collection, organization and systematic refinement of the assets (invariant or variant)
- Automatic building of the products
  - **Product Configuration**: through asset integration procedures
  - **Product Specialization**: through the specification of the assets parametric part
... From SPL to BPL: Analogies and Tailoring

BPL

- **Asset** concept is referred to **activities** and **work definitions**
- **Product Configuration** ≠ **Process Configuration**
  - *the assets (activities and work definitions) can be added to a basic business process in order to configure the target business process*
- **Product Specialization** ≠ **Process Specialization**
  - *each asset of the target business process can be specialized through attributes indicating specific architectural characteristics to implement them*
BPL Decision Models

**Hypothesis:** two kind of relations

1. between the *business capabilities* (characterizing the customer needs) and the suitable *processes elements* (that have to be integrated in the target business process)

2. between the *customer requirements* and the specific peculiarities of the *processes elements* previously integrated in the target process.
Decision Table Formalism

A decision table (DT) is divided in four quadrants: conditions (Cond), conditional states (S), actions (Act) and rules (x)

The table is defined so that each combination of conditions and conditional states corresponds to a set of actions to carry out.

- Compact overview
- Modular knowledge organization
- Evaluation of consistency, completeness and redundancy
Configuring DT ...

For each BPL a configuring DT is built in order to select the variant assets characteristic of the requested business capabilities.

They have to be composed with the invariant assets (integrated into a basic process) in order to generate the target business process.
... Configuring DT

- the CONDITION quadrant contains a set of **business capabilities**, \( BC_i \) \( i=1,...,n \)
- the CONDITIONAL STATE quadrant contains the possible **values** of each business capability \( [BC_i]=\{bc_{i1}, bc_{i2}, ..., bc_{iq}\} \)
- the ACTION quadrant contains all the possible **variant assets** \( \{va_1, va_2, ..., va_r\} \) that can be added to the process commonality
- the RULE quadrant relates each capabilities profile to the corresponding variant assets.

<table>
<thead>
<tr>
<th>BC</th>
<th>bC11</th>
<th>bC12</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC1</td>
<td>bC11</td>
<td>bC12</td>
</tr>
<tr>
<td>BC2</td>
<td>bC21</td>
<td>bC22</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCn</td>
<td>bCn1</td>
<td>bCn2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VA</th>
<th>bC11</th>
<th>bC12</th>
</tr>
</thead>
<tbody>
<tr>
<td>VA1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VA2</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>VA3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VA4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VA5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VA6</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

- The table illustrates the mapping between business capabilities and variant assets, highlighting the selected variant assets for each profile.
### Specializing DT

For each asset, variant or invariant, a specializing DT is built as follows:

- the CONDITION quadrant contains a set of customer requirements, $\text{CR}_j$, $j=1,\ldots,m$, to specialize the parametric part of the asset;
- the CONDITIONAL STATE quadrant contains the possible values of each requirement: $[\text{CR}_j]=\{\text{cr}_{j1}, \text{cr}_{j2}, \ldots, \text{cr}_{jt}\}$;
- the ACTION quadrant contains the parameters $\{p_1, p_2, \ldots, p_s\}$ and their values allowing to specialize the parametric part of the asset;
- the RULE quadrant relates each customer requirements values set to the corresponding specializing parameters.

<table>
<thead>
<tr>
<th>CR_1</th>
<th>fr_{11}</th>
<th>fr_{12}</th>
<th>fr_{13}</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR_2</td>
<td>fr_{21}</td>
<td>fr_{22}</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CR_n</td>
<td>fr_{n1}</td>
<td>fr_{n2}</td>
<td>fr_{n1}</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- $p_i = \text{“Y”}$
  - $-$ $-$ $-$ $x$ $x$ $-$ $-$ $-$ $-$ $x$ $x$
- $p_i = \text{“N”}$
  - $x$ $x$ $x$ $-$ $-$ $x$ $x$ $-$ $-$ $x$ $x$
- $p_i = \text{“Z”}$
  - $-$ $-$ $-$ $-$ $x$ $-$ $x$ $-$ $-$ $x$ $x$
Case Study ...

Our proposal has been investigated in an industrial case during the research project “DAMA” (Data Archiving Management and Acquisition)

- A specific part, Document Recognizing, is here summarized

Invariant Part

- the process contains an OCR (Optical Character Recognition) activity requiring a scanned Document Image as input and produces a recognized Text Document as output
Case Study...

Configuring DT

- the table provides the following business capabilities: Signature Extraction, Layout Analysis and Image Extraction

<table>
<thead>
<tr>
<th>Signed Documents</th>
<th>Digital</th>
<th>Autographic</th>
<th>Without Sign</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document Type</td>
<td>Structured</td>
<td>Unstructured</td>
<td>Structured</td>
</tr>
<tr>
<td>Documents with Images</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>Layout Analysis</td>
<td>X</td>
<td>X</td>
<td>-</td>
</tr>
<tr>
<td>Image Extraction</td>
<td>X</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Autographic Sign Extraction</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Digital Sign Extraction</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Scenario

“The enterprise needs besides to elaborate and archive typewriting and structured documents, containing images and without signature”

<table>
<thead>
<tr>
<th>Signed Documents</th>
<th>Digital</th>
<th>Autographic</th>
<th>Without Sign</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Document Type</strong></td>
<td>Structured</td>
<td>Unstructured</td>
<td>Structured</td>
</tr>
<tr>
<td>Documents with Images</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>Layout Analysis</td>
<td>X</td>
<td>X</td>
<td>-</td>
</tr>
<tr>
<td>Image Extraction</td>
<td>X</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>Autographic Sign Extraction</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Digital Sign Extraction</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
... Case Study

Diagram:

1. Document Image → OCR → Text Document

2. Document Image → Layout Analysis → Doc_Layer_Structured
   → OCR Typewriting → Text Document

   → Image Extraction → Extracted Image
Conclusion ...

This work proposes to apply the good practices of SPL to SOA, the authors introduce:

- the concept of **BPL** in order to identify commonality and variability of SOA systems at the process level
- two kind decision models supporting BPL activities
  - Configuring Decision Model
  - Specializing Decision Model

The case study **DAMA** is ongoing and encourages further investigations in other applicative domains in order to confirm and generalize the preliminary results.
... Conclusion

In order to support the application of the proposal here presented, the authors are developing two tools:

- the former aims to **automate** the decision tables management (design and consulting)
- the latter is able to **transform business process models** in executable workflows for SOA systems