BPA Suite to BPEL: A Case Study

Lonneke Dikmans
Vennster
Utrecht, the Netherlands

Keywords:

SOA, BPM, BPA Suite, BPMN, BPEL, Case study

Introduction

Often organizations that build applications with Oracle SOA Suite11g, also have Architecture and BPM efforts in their organization. This session describes the experience of the project while modeling and developing processes, and the relationship with Architecture and BPM efforts in organization. Finally some licensing issues we ran into and our final conclusions are discussed.

Context

The province we are talking about has 1.130.644 inhabitants, living in 26 municipalities. They employ approximately 1000 people. The organization consists of 10 units and they offer services in the field of grants, permits and development. The province is executing a program to realize common capabilities for the entire organization. Examples of these common capabilities are case management, document management, CRM services, a portal for employees and citizens and identity and access management. On the Business Process Management side, they adopted 'zaakgericht werken', or case management according to the GEMMA. This process consists of several building blocks: A citizen or company can ask a question or submit a request (Inform & intake); if the request needs to be processed, a case is created and the case is handed over to the correct back office department. Otherwise, the answer is given and the process stops; once the case has been created, the specialist processes the request; if a formal decision has to made, the draft decision is handed over to the authority that takes the formal decision; finally the result is delivered to the applicant. During this, the process is governed, monitored and managed. The process is visualized in the figure below:

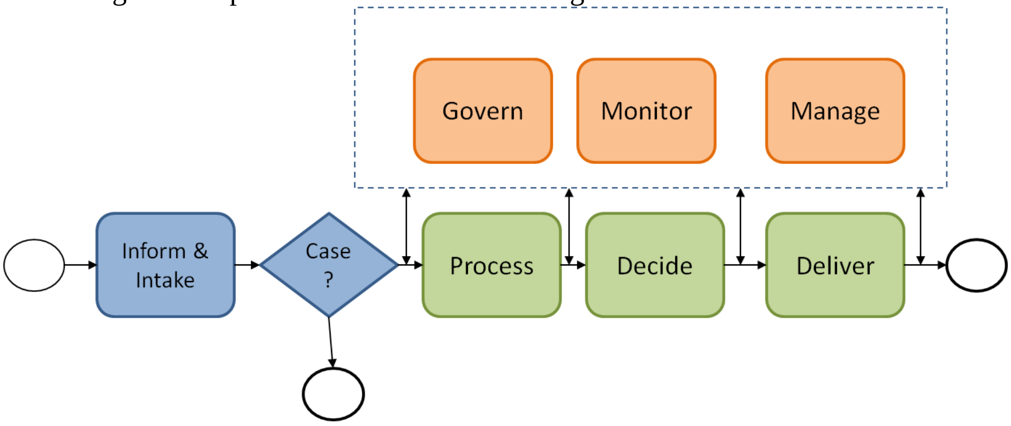

Illustration. 1: Generic case management process defined for the public sector in the Netherlands

Modeling the processes

The program started by modeling two major business processes that span multiple departments: grants and permits. The grant process was the first to be realized, with the permit project in mind. The grant process consists of multiple sub processes that all need to be handled in compliance with ‘zaakgericht werken’: Initial application, change requests (optional), reporting (optional), final application, appeal (optional).

The end-to-end grant process spans multiple departments: the front office where they receive the applications (by mail and through the internet channel), the care division where they evaluate the applications, the finance department where they keep track of the budget and make the payments and finally the mail room where they handle the mail that needs to be sent. The final application is sent in by the organization after the work has been done. This can lead to money being returned, or extra money being paid. The following requirements were stated:

- Processes needs to be monitored end-to-end;
- As little changes to the current process as possible (silent requirement, not stated explicitly);
- It should fit the generic process because of reuse;
- Process should be flexible: changes need to be made by the business;
- Complete tracking of every attribute in every step.

The initial modeling was done in BizAgi. The process models were created as part of the requirements in the bidding process. A process modeling tool for the final models was part of the RFP.

Design perspective

The province selected Oracle SOA Suite11g and Oracle BPA Suite11g for the process support. The case management metadata was based on a COTS case management system for the Dutch public sector that would be customized for the province.

The process models would be executed using BPEL, using the human task service for manual steps.

During the design phase, the following decisions had to be made:

- Model and support the main grant process, or just the sub processes (intake, process, decide, deliver). The main process from initial application to final application can span years;
- How to handle user interaction versus process flow;
- Transformation of BPMN to BPEL;
- How to handle ad-hoc tasks;
- Error handling. Both technical errors and business exception need to be modeled.

Findings

After the process models were delivered, we evaluated them in order to move them to the BPA Suite. The following issues were found:

- The design decisions listed above were not known with the process modelers, they focused mainly on the business requirements of the process models;
- The models contained BPMN errors, both syntactical errors and semantic error. An example of a syntax error was the lack of start events in some sub processes. Semantic errors were made, for example using a signal instead of a message to communicate with a halted process;
- There was no layering in the process models; they had started with the lowest model, with sub processes on different levels;
- The business view was intermixed with an implementation view;

- Processes were not translatable to BPEL. For example, BPEL cannot handle ‘go to’, some processes had two starting events;
- There were technical decisions in the model;
- The user interaction was modeled in the process model. This made for very fine grained process steps;
- No distinction between rule based and data-based gateways was made;
- No error handling or exceptions were defined.

Development perspective

Once the BPMN processes were (re)modeled in the BPA Suite, blueprints were generated. These blueprints were then picked up from Oracle JDeveloper, using the BPA suite plug-in.
Sometimes the connection between de BPEL process and the BPA Suite model got lost. It is not possible to reconnect the process to an existing model once it has been associated with a specific database instance in the BPA Suite. This meant starting from scratch once this happened.

BPMN and SCA

When you create a BPEL process from a blueprint, you have to decide on the relationship between the sub processes and SCA composites. In one extreme, you put all the BPELs in one SCA composite. In the other extreme you create a SCA composite for every BPEL. The following guidelines were formulated:

- For each phase/building block (Intake, process, decide, deliver), a separate SCA composite was created. These were started using a queue;
- If a sub process was reusable (callable from multiple processes), it was defined as a separate composite and called using a reference (because Oracle SOA Suite11g does not support nesting of composites);
- Embedded sub processes are defined within the same SCA composite.

Note that it is not visible in the wizard what type of sub process is defined; the developer needs to know what design decision to take for each sub process.

Patterns

The generated BPEL code made adherence to BPEL coding and design guidelines difficult:

- Claim check pattern; the canonical data model was used in the BPA suite, causing too much data to be present in the BPEL process. No mediators were generated, all the processes were calling services directly;
- Retry on ‘onMessage’. A good practice is to use a pick activity for ‘onMessage’: one for success and one for failure. It can also be a good idea, to add an ‘onAlarm’ activity to the scope, to be able to throw a replay fault. This is not the way the onMessage gets generated from BPMN, obviously.
- Handle business errors in the code as regular flow. Ending a process because of a business error sometimes had to be forced by throwing a fault in the BPEL code. This ends up in the console as a fault and is confusing to administrators.

Licensing perspective

At the time of the bidding process, Oracle was very focused on selling Oracle SOA Suite, not Oracle BPM Suite. They sold BPEL as a process language that could be changed by the business analyst, using BPA suite. It was very unclear for the customer that this involves technical people and releases. The consequence of designing and defining something in a process model and then generate code from it versus the flexibility of case management was not clearly explained at the time.

After some time in the project, it appeared that Oracle BPM would have been a better fit. The procurement laws don't allow changes to the offer after the European bidding process has ended. So this makes it very difficult to adjust the licenses, even when there is a clear business case.

Because of the deal between Oracle and SoftwareAG (previously IDSScheer), the people from SoftwareAg started their up-selling and cross selling: training and extra tools like SOA Architect and IT architect write after the deal was closed. This caused confusing at the side of the customer: the difference between BPM and enterprise architecture and business architecture was not explained in advance. They did not expect to buy more tooling then they already bought. The province wanted to use Archimate in the top layers of their architecture models. This plug-in is not available from Oracle; you have to buy a SoftwareAG product to be allowed to run it. This again was confusing to the province; they had to deal with multiple vendors for the same tool.

Conclusion

There are a few lessons learned from this project:

1. Keep the BPMN models on a business level;
    a. This creates better quality in the BPELs;
    b. This creates better user interfaces;
    c. This creates more flexibility.
2. Knowledge of BPMN and BPEL is necessary both during analysis and during development. If the business analysts are no longer available and the programmers don't know how to change the models correctly or update them, the models become obsolete. Once the connection is lost, you have to start over. There is no way to restore the connection;
3. There was very little support from Oracle. This has several reasons:
    a. attention has shifted to BPM studio;
    b. because of the deal with SoftwareAG Oracle doesn't pay a lot of attention to this side of the Oracle BPA suite;
    c. There is little knowledge about the BPMN to BPEL connection in Oracle.

The next time we have requirements like this, it is better to use Oracle BPM studio instead of the Oracle BPA Suite to Oracle SOA Suite connection:

- The requirements for changes by the business and case management are easier to meet with BPM Studio;
- There is no impedance mismatch from BPMN to BPEL, the BPMN is executed;
- There is only one vendor (Oracle), not two (Oracle and SoftwareAG) that are competing;
- Oracle is committed to BPM Studio, it is their product, not an OEM product.

At the time of the bidding we had fewer options. Now we would prefer Oracle BPM Suite, running BPMN 2.0 models.

Contact address:

Lonneke Dikmans
Vennster
Postbus 31457
6503 CL, Nijmegen, the Netherlands

Phone: +31(0)6-15083349
Email lonneke.dikmans@vennster.nl
Internet: www.vennster.nl