Automatic derivation of BPEL4WS from IDEF0 process models

1 Introduction

Integration definition for function modelling (IDEF0) is one of the most popular notations for modelling business processes. It employs a rather simple and intuitive modelling construct, consisting of boxes representing functions and arrows connecting them signifying flow of information and materials. Web services on the other hand are an emerging technology for implementing distributed systems. Web service orchestration languages, such as Business Process Execution Language for Web Services (BPEL4WS), are the emerging approach for describing processes as networks of coordinated web services. Business processes as captured in IDEF0 models, however, may contain both web services as well as other types of activities which need to be coordinated. By automatically analysing the Extensible Markup Language (XML) definition of an IDEF0 model, we can identify how web services interact with other activities and at runtime generate code to support the orchestration of web services with the overall business process. The approach proposed is independent of the orchestration language and ensures an implementation independent model for specifying web service orchestrations. This approach also enables the top-down analysis of a business process to its constituent web services and avoids any misalignment problems during design time between the two.

Keywords IDEF0 · UML · Web services · Orchestration · WSDL · BPEL4WS
2 Importance of modelling semantics of business processes and identification of corresponding web services

The semantics of business processes—what they do and what information elements they manipulate—is a key issue. Business value results from Business-to-Business (B2B) collaborations that do the right thing. If they lack consistency, the damage may be dramatic. Therefore, the issue of how to trust that a business process does the right thing, before it is used, is a crucial one.

The term semantics, as used here, refers to the meaning, in human and business terms, of the business process, its inputs, side effects, and outputs/results. Even in the abstract, the semantics of a given category of business processes is often much more complex than could be completely described by natural language text. Work in systems theory offers support to humans in order to deal with that level of complexity; however software technology that is available today is difficult to capture and represent properly these semantics.

One solution is to hard code the business semantics in software enabled solutions, such as web services and let humans statically determine the collaborations at design time, using business process execution languages such as BPEL4WS. However, hard coded solutions tend to take away the flexibility required by business processes.

A second solution is to try to create process models describing both the procedural aspects of business processes, including feedback loops and the less observable aspects of the business service dynamics (e.g. nonlinear relationships, intuitive theories about the business performance and collections of beliefs, preferences and networks of knowledge) that determine the real business semantics. Appropriate process models must provide the means to assert the required web service specifications as well as to ensure correct continuous functioning needed to support the business process at any level of abstraction appropriate for B2B collaboration.

We primarily explore the process model option here, because the business value of B2B processes depends on business dynamics (feedback loops etc.) and the flexible framework of business collaboration. In order to efficiently encode the business dynamics in a process model and form the basis of a flexible business collaboration framework, we need a well accepted and well established modelling technique which will provide us the means to reach the goals we have set. We also need a neutral modelling approach which will enable us to apply the Java paradigm to the area of business process modelling and orchestration, i.e. to be able to model a business process only once and execute it using any orchestration language. We propose that this can be achieved by employing the IDEF0 modelling technique.

We advocate a meta-model for business process orchestration based on IDEF0. This meta-model enables us to capture the business semantics of a business process, as described above, as well as to define at any level of abstraction the corresponding web services that we would like to employ, thus ensuring the semantic integrity of business processes and web services that support them. Finally, it captures all the semantic required in order to be able to derive web service orchestration code for the orchestration language of our choice, as we will describe in Sect. 5.

3 IDEF0 modelling technique

IDEF0 [1] is a modelling technique that has been extensively used for the modelling of business processes. IDEF0 is based on the Structured Analysis and Design Technique (SADT), a graphical approach to system description, introduced by Douglas T. Ross in the early 1970s. An IDEF0 activity diagram contains one or more levels of decomposition of a process. Boxes within a diagram show the sub-processes (activities) of the parent process named by the diagram (Fig. 1). Arrows between the boxes show the flow of products between processes.

The main difference with other modelling techniques such as UML, BPML, etc. is that what in IDEF0 is considered as an activity is a very generic concept which can include:

- Non software activities carried out for example by humans.
- Automated activities carried out by a software application, such a web service.
- A combination of the above.

These 3 unique characteristics of IDEF0 make it ideal for capturing the semantics of business processes in an abstract level. Then by decomposing this level to sub-levels we reach...