Abstract: Process modeling and workflow applications have become more and more important during last decade. The main reason for this increased interest is the need to provide computer aided system integration of the enterprise based on its business processes. This need requires a technology that enables to integrate modeling, simulation and enactment of processes into one single package. Predecessors of this paper introduced method BPM (Business Process Modeling) and its implementation in a software system BP Studio (Business Process Studio). This paper describes how this system (primarily focused on process analysis, simulation, and control) can be extended into real workflow management system. It means, how the automated aspects of business process can be controlled.

Key Words: Business Process, Process Modeling and Simulation, Workflow Management System, Java, Design Patterns, Adapter

1 Introduction

Basic definitions as were defined by Workflow Management Coalition are introduced at the beginning of this paper to clarify what is the difference between Business Process, Process Model, Workflow, and Workflow Management System:

- **Business Process**: A set of one or more linked procedures or activities which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles and relationships.

- **Business Process Model**: The representation of a business process in a form which supports automated manipulation, such as modeling, or enactment by a workflow management system. The process definition consists of a network of activities and their relationships, criteria to indicate the start and termination of the process, and information about the individual activities, such as participants, associated IT applications and data, etc.

- **Workflow**: The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules.

- **Workflow Management System**: A system that defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of IT tools and applications.

Understanding of these notions is very important for understanding of how the original BP Studio was designed and how it was extended to obtain complete Workflow Management System.

* Dept. of Computer Science, VSB-Technical University of Ostrava, Czech Republic
2 Business Process Studio

Business Process Studio is the software system that implements method for modeling called BPM (Business Process Modeling) [VSK99, VSK99b].

2.1 BPM Method

BPM builds three different kinds of models for each business process that is being captured:

- The main aim of the **Functional Model** is an identification of the business process architecture, as well as the identification of process customers and products.
- **Object Model** identifies static structure of all entities (objects) that are essential for the enactment of the process. In other words, the answer to the question by *whom and what* the process is realized is searched. This model tries to capture all active objects responsible for an execution of activities and passive objects that can be understood as material, products or documents that are manipulated by the process. All these objects have a set of attributes associated.
- **Coordination Model** is based on previous two models and its goal is to show *how* the process will be enacted. The coordination model specifies interactions among objects (active and/or passive) and defines the way *how* all these activities are synchronized based on principles used in Petri Net [Pet77]. The coordination view is the most important because it enables to define the execution order of all activities, including conditions for their potential concurrency. It means that the correct order is defined, as well as sharing of used resources. Each activity can have more than one scenario with the duration time and costs associated to provide necessary information for the analysis.

2.2 BP Studio Architecture

The system called BP (Business Process) Studio consists of four main applications:

- **BP Model** is intended for modeling and analysis of business process. User friendly graphics editor is used for visual modeling. In addition, BP Modeling can be used for simulation purposes. One menu command is responsible for switching between modeling and simulation stage.
- **BP Viewer** is an Intra/Internet application used as Java applet through Web browser. The purpose is to make all process models available to people interested in process specification. The model can be browsed and analyzed in the same way as BP Model provides it. The only restriction is that the model cannot be changed.
- **BP Control** instantiates process model and enables process control and monitoring. Activities are scheduled automatically based on presence of all input objects and distributed through network together with all associated objects to so called actors – instances of active objects responsible for execution of the activity. The mechanism of distribution is based on remote object access enabled by ORB (Object Request Broker). Process is monitored by process manager through Gantt diagram generated from the coordination diagram of process model.
- **BP Actor** is the stand-alone application or applet that is used by instances of active objects responsible for execution of activities. BP Actor publishes task list generated by BP Control to the actor who starts the activity, finishes it and selects the used scenario. Since the activity is also associated with passive objects, the actor defines values of their attributes. Once the attributes are changed, they become available to other actors and
process manager. The Intra/Internet nature of the described solution enables to use mobile equipment (notebook and mobile phones) to participate in the process enactment even when the actor is on the road.

All applications operate with a repository of process models and repository of process instances. The first one is used for modeling and analysis, while the second one is used only for process control. Models are also used for instantiation of process instances. Once the model is instantiated, it becomes a part of process instance definition and cannot be saved back to the model repository. The architecture of the BP Studio is following (Fig.1):

If the presented software system is compared with the definitions mentioned at the beginning of this paper following features required by Workflow Management System are missing to consider BP Control as full featured workflow application:
- Activities are enacted only by human actors.
- BP Studio has no open interface to integrate it with other IT tools, data and documents.

### 3 Process Enactment

The enactment of the process requires to instantiate the process model. The Coordination Model serves as the engine for the enactment where the passive and active objects are instantiated (Fig. 2). This instantiation means that instances with appropriate attributes and relationships are created based on the specification found in Object Model (this is a direct analogy with CASE tools). The only difference is that active objects are mapped by process manager (user of BP Control) to actual human resources at the same time.
The enactment of the process is based on the enactment of activities. The scheduled activity is moved to actor’s task list. This actor is responsible for its execution. The beginning and the end of the activity is marked by the actor to inform the engine about the actual state of the activity.

A possibility to extend this human-based enactment with the automatic execution of activities by a computer was designed with the following goals in mind:

- the mechanism for an execution of the activity should not limit the original human based enactment,
- the code for the activity execution should be external to BP Control, i.e. it can be developed outside BP Studio,
- the external code could be downloaded through the Internet to actor’s task list,
- the external code should have an access to instances related to the executed activity as well as to the activity itself.

3.1 **External API**

For the above mentioned purposes external API (Application Programming Interface) was designed:
A class *ExtApp* represents an external code responsible for the execution of the code associated with the activity. This class implements interface *BPExtern* with a method *execute()* that is available and called by BP Control when the activity is enabled. External code keeps references to active and passive instances as well as reference to the activity that launched this external code.

### 3.2 Adapters

The access from the external code to the information kept by BP Control is based on the usage of *Adapter Design Pattern* [GHJV94]. In our case *ExtApp* keeps reference to array of active and passive instances that implement interfaces *BPActive* and *BPPassive*. The actual activity can be accessed through the protocol declared by interface *BPActivity*. These references are mediated by appropriate adapters (Fig. 4):

![Diagram](image-url)
The light grayed classes represent classes from the external API while dark grayed classes represent internal classes from BP Control. All adapter classes serve as mediators to the protocol of internal objects and they are instantiated only when the external code is executed. It means when the BP Control or BP Actor execute external code associated with the activity the adapters are instantiated and passed to external code. The only difference between BP Control and BP Actor is that BP Actor downloads external code through Internet while for BP Control all classes are local.

4 Conclusions

The above mentioned approach is already implemented in the BP Studio including several external applications that solve issues typical for workflow systems.

First of all it is a possibility to connect instances from process enactment with external XML documents. Any instance can be associated with XML document that resides on document server. It means when the appropriate external code is executed the attribute values of such instance are synchronized with the content of the related XML document.

The second application enables to connect instances with the content of tables from relational databases. Based on that BP Control becomes a real Workflow Management System that satisfies all requirements specified at the beginning of this paper.
References


[GHJV94] Gamma, E., Helm,R., Johnson,R., Vlissides,J. *Design Patterns, Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1994
