# Workflow Systems for Inter-Organizational Business Processes

Herbert Groiss, Johann Eder
University Klagenfurt
{herb,eder}@ifi.uni-klu.ac.at

## Introduction

A major goal of information technology is the effective support of a company's processes. Workflow systems are the emerging technology to provide flexible support for specification, execution, and monitoring business processes. Processes (or workflows) are decomposed into several interacting components and described using a high level language.

Business processes normally require the interaction of persons and systems of different departments or enterprises. A simple selling process for example, requires the collaboration of four partners (buyer, seller, shipping agency, and bank(s)). Moreover, in large, distributed organizations various levels of automization and heterogeneous IT-infrastructure complicate the design and execution of a general business process. The definition and execution of processes spanning over several products and using different forms of communication with the partners is therefore necessary.

The platform of choice for electronic interaction between enterprises is the Internet with the services e-mail, ftp and WWW. Especially the latter has become very popular for accessing information and most workflow vendors built interfaces of their products to the Web. The services of the Internet can be used to support all steps of a business process:

*Advertising*: The seller of goods or services can provide information on Web pages, or mails potential customers directly. Interested persons find the Web sites by using a search engine or following a link.

*Initialization*: After a customer has found a supplier (or vice versa) a business process can be initialized: the customer initializes the process by making a request (for quote, for information, ...) or placing an order. Using the Internet, this can be done via a HTTP (Berners-Lee et. al., 1995) form submission, the customer interacts with an application on the Web server of the supplier and is guided for providing the necessary information for initializing a process.

*Cooperation*: After initialization the business process normally contains further interaction between the partners: gathering additional information, bargaining, arrange delivery and payment conditions, and payment.

*Monitoring*: Customers should have the possibility to monitor the forthcoming of the process (a popular example of such functionality is the possibility to track the state of Federal Express packets, http://www.fedex.com/track_it.html).

Although email and the Web allows paperless communication, for interaction between workflow systems the standardization of a common set of interfaces is required. Several organizations are currently working on such standards: The Workflow Management Coalition proposes a standard for the interface 4 (server to server interaction) of its reference architecture (WfMC, 1996). The Object Management Group will define a workflow management facility IDL interface (request for proposals deadline was in August 1997 (OMG, 1997)).

These standardization efforts have two common drawbacks according to the above requirements:
(a) The kind or structure of the data exchanged by the processes is not defined.
(b) The only form of interaction considered is between two workflow systems implementing the standard. Normally, one wants to use the same processes with different partners, probably having no Internet access, or having only e-mail.

In this paper, we describe a slightly different approach based on the following design criteria:

- The interaction mechanism should support various levels of integration: (a) no electronic access of the partner, (b) email only, (c) interoperable workflow system installed, (d) tight integration of interoperable workflow systems (processes and data structures known by each other).

- Loose coupling between the systems: minimal number of interactions and interfaces.

- No necessity to change the business process when one of the partners changes his environment.

In the next section we present the Web-based workflow system Panta Rhei (Heraklit: "Everything flows"), which follows this design and allows the flexible collaboration of agents in the Internet.

## Panta Rhei

The heart of a workflow management system is the way how processes are defined and executed. In Panta Rhei (see (Eder et. al., 1997) and (Groiss and Eder, 1995) for a detailed description) workflows are specified in the script language WDL, defining the structure of the steps in the workflow using control structures for loops, branches, and parallelism. Every such step specifies *who* does *what* with *which data*, for example:

```
sek make_approval(application);
```

This step defines that the role `sek` performs the task `make_approval` using the form `application`. The agent of a task can be either a user, a role (a group of users), the agent of a previous task, or the value of a form field. This form of dynamic agent assignment allows the specification of ad-hoc tasks. The following example shows the definition of a complete process:

```
process job_process ( )
forms f jobform_1;
version 1;
subject f.subj;
timeouttask tk inform(f);

begin
  all order(f) a1;
  repeat
     f.rec a_task(f);
   until f.fin=1;
 a1:user inform(f);
end;
```

This is a process for general jobs, where a person wants some work done, makes an order and sends the order to some other person. The recipient can either perform the task or forward it to another person. When the task is performed, the initiator of the process is informed.

After a header defining the name and parameters of a process, in the declaration part the `version`, `description`, the `forms`, and other process properties are specified. The flow of control is written between the keywords `begin` and `end`. The first activity is an `order` and can be performed by `all` (a special role containing all users). After the order is finished, the step `a_task` is repeated as long as the `fin` field of form `f` is not set to 1. After the repeat loop the activity `inform` is performed by the user of the first step - the label `a1` specifies this reference.

The definition of tasks in WDL contains the specification of a postcondition, a maximum execution time, or a program, if the activity is executed automatically.

The data objects used in the activities are represented as HTML forms, this allows us to use a standard HTML-editor to create a form description, which can then be loaded into the workflow system.

Furthermore, WDL allows the specification of the organizational structure and the definition of roles and role members.

The processes can also be defined with a graphical editor, which is - following the Web-based architecture - written as Java Applet. Processes designed graphically can be translated to the WDL representation and vice versa.

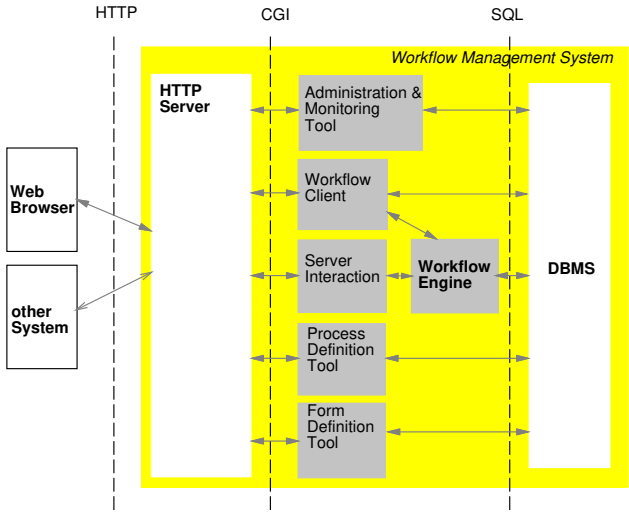


**Fig. 1 Architecture of Panta Rhei**

Fig. 1 shows the system architecture. Every interaction with the workflow system is done via the HTTP server. The database contains the data about the process definitions and instances. The administration, process definition, and form definition tools interact directly with the database.

The engine itself is responsible for the execution of the workflows. When a user starts a process or finishes a task, the system selects the successor task and assigns a user or a group of users with this/these task(s).

### Interaction with other systems

Processes which interact with other systems or agents outside the organization do not differ from locally executed processes. If the engine detects that the agent of a task is not on the local server, some information about the task (process id, process number, task id, due-date, ...) together with the forms are sent to the recipient's URL.

If the recipient has email, then the form data is sent in HTML format (the Postscript version of this form can be faxed or mailed if no electronic access is available). The recipient can perform some actions on the data using the JavaScript buttons of the form. These actions eventually result in an HTTP connection back to the workflow server.

Fig. 2 shows such an email message, it contains information about the process and activity and allows the manipulation of the associated forms. In the job-process of the previous example the field `f.rec` holds the agent of an activity. If an email address is stored in this field, the engine would send the information to this address.

If the URL of the agent uses the HTTP protocol, a form upload to the other server is performed, using the same data as above but without the HTML code. Only the names and values of the fields are transmitted. The receiving-function of the workflow server performs the following steps:

- if the process is not known on this server, start a new one, otherwise add the activity to the process.
- if the HTML representation of the forms is not known, connect to the sender and get it.
- if a recipient is given, put the data in his worklist, otherwise send it to some default persons.

In both forms of interaction, the sender gets back the modified/enhanced data via HTTP. Except from parsing the URL for choosing the right method to send, no assumptions or knowledge about the processing of the data on the remote site is necessary.

A tight integration with automatic processing of incoming forms is possible when business partners agree to use the same forms. Fortunately, there exists a standard for the electronic exchange of structured documents. The EDIFACT standard of the United Nations (UN/EDIFACT Rapporteurs, 1997) defines the structure of several dozens of messages for electronic commerce. Examples are request for quote (RECQUOTE), request for reservation (RESREQ), or invoice message (INVOIC).

Using the structure of these forms (the field names and value restrictions), HTML can be used for visualizing and HTTP for transmitting them.

Specialized workflows can then be written reacting on the receipt of certain forms, for example a process can be forwarded to the right person, when an ORDER of a specific product is received.

**Fig. 2 Mail message containing a workflow activity**

### Conclusions

The ability to interoperate will be the key success factor for tomorrow's workflow products. In this paper, we have proposed a simple and flexible mechanism for interaction of workflow systems in the Internet. Using the standard protocols email and HTTP process information is sent as HTML pages. Using the EDIFACT standard for the structure of the forms, a tight integration of workflows is possible with making the fewest possible

assumptions about the IT-infrastructure of the business partners.

## References


T. Berners-Lee, R. Fielding, and H. Frystyk (1995): *Hypertext Transfer Protocol - HTTP/1.0*. RFC 1945, http://www.w3.org/Protocols

J. Eder, H. Groiss, and W. Liebhart (1997): The Workflow Management System Panta Rhei, in *Advances in Workflow Management Systems and Interoperability*, Springer-Verlag, to appear

H. Groiss, J. Eder (1995): Interoperability with World Wide Workflows In: A. Ertas, C. V. Ramamoorthy, M. M. Tanik, I. I. Esat, F. Veniali, Taleb-Bendiab (eds.):1st *World Conference on Integrated Design & Process Technology*, Austin, TX

Object Management Group (1997): *Workflow Management Facility - Request for Proposals*, OMG Document cf/97-05-06

UN/EDIFACT Rapporteurs (1997): *United Nations Directories for Electronic Data Interchange for Administration, Commerce and Transport*, http://www.unece.org/trade/untdid/welcome.htm

WfMC (1996): *Interoperability - Abstract Specification* (WFMC-TC-1012, 20-Oct-96, 1.0) http://www.aiai.ed.ac.uk/