9805595-7

IPO

TITLE: WORKFLOW MECHANISM FOR A STATELESS ENVIRONMENT

FIELD OF THE INVENTION

The present invention relates to a workflow solver, and more particularly to a workflow mechanism for a stateless or connectionless environment such as the Internet or an Intranet.

BACKGROUND OF THE INVENTION

Business processes are typically analyzed in terms of workflow items. These workflow items can be business processes or documents. In the art, there are workflow management systems to automate routing of the workflow items based on specific business rules and roles employees play during the process.

Known workflow management systems include Electronic Procurement Services or EPS. An EPS provides government, corporate and consumer markets with a simplified and streamlined approach to traditional procurement processing. An EPS typically utilizes client/server technology to buyers and suppliers. Buyers can distribute the EPS client application throughout their organization, thereby bringing online purchasing directly to the desktops of purchasing agents.

The purchasing agents, or other employees who have been authorized to use the EPS, construct purchase orders by

selecting from electronic catalogues. The EPS manages procurement processing from this point forward, and electronically routes the purchase orders through the approval process to the appropriate suppliers for fulfilment. The purchasing agents receive updates and confirmations from suppliers up to the point of the shipment.

Electronic procurement systems have gained acceptance, particularly in large governmental and corporate organizations, because they automate and enhance the procurement process without the need for extensive retraining of the purchasing agents and support staff. By automating labour-intensive tasks, such as purchase order requisition and authorization, an electronic procurement system raises service levels while at the same time reducing costs and clerical errors.

Known electronic procurement systems are implemented as a client-server application on a local area network or LAN. A well-known EPS is World Purchasing Professional™ or WPP from the IBM Corporation. The WPP system comprises a client-server application that gives the purchasing department in corporations or government the ability to link electronically to both internal departments and external agencies and suppliers.

The architecture of a conventional workflow management system 1 according to the prior art is shown in Fig. 1. The workflow management system 1 comprises a workflow server 2 and a plurality of workflow clients 3, shown individually as 3a, 3b,..., 3n. The workflow server 2 and clients 3 are connected via a local area network (LAN) 4, or a wide area network (WAN). Each workflow item 3 comprises two

principal components: a data element denoted by W *data*, and a business rules or semantics element denoted by W *semantics*. The server 2 includes a workflow engine 5 which processes the workflow data (W *data*) according to the semantics (W *semantics*) for the workflow item. In a traditional workflow management system 1 as shown in Fig. 1, the workflow server 2 functions as a central server which contains all the business rules (i.e. W *semantics*) for the system 1, and the workflow clients 3 are continuously connected to the server 2 through the LAN 4 as indicated by the solid lines. Because the clients 3 are continuously connected to the server 2, the workflow engine 5 needs to maintain the context of each client 3. Such demands typically result in a complex workflow engine 5.

In recent years the Internet has emerged as the dominant communications network, and provides enormous potential for electronic commerce. The Internet comprises a system of linked computer networks, e.g. LAN's and WAN's, which is worldwide in scope, and provides data communications over the seamless collection of networks. As such, the Internet is not a computer network, as is commonly thought, but rather a stateless environment which provides a mechanism for connecting existing computer networks.

In a stateless or connectionless environment such as the Internet, connections among the computers are not long term. These connections are made only for a short time in order to transfer data. Once data is transferred, connection between the computers is broken and computers are free to work on the data as they wish. Once the work is done, the computers establish connection again. Because a continuous

connection during the whole business process is not maintained, it is very difficult to maintain the context for a process. In a conventional networked environment as shown in Fig. 1, the permanence of the connections between the server 2 and the client machines 3 allow the server 2 to continuously maintain the context of the business process.

The principle problem that arises for a workflow management system in a connectionless network, such as the Internet, is the support for automatic routing and n-level of approvals in the connectionless network. This problem arises because the context of the business process is not maintained in a connectionless network. There is no single source, i.e. a server, where all the semantics of the workflow items can be kept. Connections between clients and servers are continuously made and broken and as a result context information is lost. By definition, a workflow server must have access to all workflow items in order to manage the workflow.

In a traditional workflow system, rules to control the business logic are kept in the workflow server 2 (Fig. 1) and context information about the workflow item is always maintained by the server. In a connectionless environment this information needs to be maintained by the workflow items.

Another shortcoming with existing workflow management systems is the limitations with respect to granularity. In a practical system, this means analyzing the business workflow either in terms of processes or documents. It will however be appreciated that real world business comprise a union of processes and documents.

THIS SHEET HAS BEEN REPLACED

Accordingly, there is a need for a document management system or workflow system for stateless environment such as the Internet or an Intranet. Preferably, such a system provides a compact design which is configurable for different applications and supports nth-granularity (where document state is determined by line-level approval as well as other business criteria).

BRIEF SUMMARY OF THE INVENTION

The present invention provides a workflow mechanism for a connectionless or stateless environment such as the Internet or an Intranet. According to the invention, the mechanism includes means for maintaining the business logic and context information about the workflow in the workflow item itself.

The mechanism comprises a workflow architecture which includes a data component for the workflow item or object, a semantics component and an inference engine component. According to the invention, all three components are packaged in the workflow item or object.

It is a feature of the workflow mechanism that workflow objects are attached to the documents and transported with the document. This allows the document (or process) to be manipulated by a client without requiring the client to remain connected with any server. Accordingly, the workflow mechanism eliminates the need for the workflow server as in conventional workflow management systems.

The present invention also provides an inference engine for solving workflow in a connectionless environment.

This Sheet Has Been Replaced



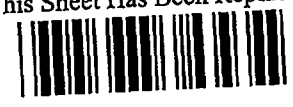
THIS SHEET HAS BEEN REPLACED

The inference engine includes means for maintaining the context of the workflow items in the workflow management system and means for implementing the arbitrary business rules and organizational structure as first order constraints for the engine. The engine includes a Justification-based Truth Maintenance System. The engine according to the present invention results in a compact design which makes it suitable for integration with workflow items.

It is a feature of the engine that Justifications can be introduced as needed, and in most cases the Justifications can be introduced at runtime. Justifications are encoded as propositional logic expressions which serve as constraints for the inference engine. Accordingly, arbitrary business rules which affect workflow can be added. As such a workflow system utilizing a mechanism according to the present invention can be used to manage both processes and documents, and provides nth-degree granularity.

The present invention provides a workflow mechanism for a stateless environment such as the Internet or an Intranet. The workflow mechanism includes the following features: (a) support for complete workflow for a document or set of documents for complex business rules; (b) support for n-level of approvals and user configurable; (c) support for authority delegations and re-routing of the document; (d) support multiple users with multiple roles and multiple privileges on multiple documents; and (e) support for arbitrary business logic. The workflow mechanism also includes the capability to maintain and control line level as well as document level approval states for n-level of approvers.

This Sheet Has Been Replaced



In a first aspect, the present invention provides a mechanism for processing a workflow item in a stateless environment such as the Internet, the mechanism comprising: a data module for data elements associated with the workflow item; a semantics module for rules associated with the workflow item; an engine for applying the rules to the data elements; wherein the data module, the semantics module and the engine are contained in the workflow item and transported with the workflow item when the workflow item is moved in the stateless environment.

In a second aspect, the present invention provides a method for processing a workflow item in a stateless environment comprising client machines and servers, the method comprising the steps of: storing data elements associated with the workflow item in a data module; storing rules associated with the workflow item in a semantics module; loading the data module and the semantics module with the workflow item; loading an engine in the workflow item together with the data module and the semantics module; routing the workflow item together with the loaded data and semantics modules and the engine to one or more of the client machines in the stateless environment; utilizing the engine to apply rules from the semantics module to data elements in the data module to process the workflow item on the associated client machine independently of the server.

BRIEF DESCRIPTION OF THE DRAWINGS

Reference will now be made to the accompanying drawings, which show a preferred embodiment of the present invention, by way of example, and in which:

Fig. 1 shows a conventional workflow management system according to the prior art;

Fig. 2 shows in block diagram a workflow management system according to the present invention;

Fig. 3 shows in a workflow mechanism for the workflow management system of Fig. 2.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention provides a workflow mechanism for workflow management systems. In the following description, the details of the workflow mechanism are described in the context of a workflow management system. However, other components of the workflow management system which do not form part of the invention, such as the mechanisms for workflow item delivery, tracking, monitoring, security, are not part of this disclosure.

Reference is made to Fig. 2, which shows a workflow management system indicated generally by 100. The workflow management system 100 is intended for operation in a connectionless or stateless environment such as the Internet or an Intranet. The workflow management system 100 comprises one or more servers 101 and a series of clients 102 and workflow items or objects 103. In Fig. 2, the servers are shown individually as 101a, 101b,..., 102m, and the clients are shown individually as 102a, 102b,..., 102n. The clients 102 comprise computers or workstations which are situated, in a corporate application for example, in head office, branch office, or remote computers with dial-in access. The dotted connections depicted in Fig. 2 represent transient

connections, and therefore, in a connectionless network, the role of client and server may not be so clearly defined.

The workflow items 103 are shown individually as 103a, 103b, 103c,..., 103n in Fig. 2. According to the invention, the workflow item 103 comprises three components: a data component W *data*, a semantics or business rules component W *semantics*, and an inference engine component W *engine*. In Fig. 2, the data component is denoted by 104, the semantics component by 105 and the inference engine by 106. According to this aspect of the invention, when a workflow item 103 is downloaded to a client, the data component W *data* (104 in Fig. 2) and the rules component W *semantics* (105 in Fig. 2) are downloaded together with the workflow engine W *engine* (106 in Fig. 2). The workflow engine 106 is the component which works on the data 104 (W *data*) in accordance with the business rules contained in the semantics component 105 (W *semantics*). In order to provide the capability for downloading, the workflow engine 106 features a lightweight and compact design, while at the same time having the processing power to work on the business rule logic defined by the rules component W *semantics* or 105.

The workflow engine 106 according to this aspect of the invention is shown in more detail in Fig. 3. The workflow engine 106 comprises an inference engine 111 and a truth maintenance system (TMS) 112. The truth maintenance system 112 comprises a constraint network which utilizes justifications. In the context of the present invention, justifications comprise simple first order propositional clauses. Advantageously, a constraint network composed of

justifications comprising simple propositional clauses is extremely fast to evaluate. It has been found that the current state of the document (or a business process) can be inferred by the justifications currently believed. As depicted in Fig. 3, the truth maintenance system 112 records justifications (link 113), and the inference engine 111 draws inferences from the justifications recorded in the TMS 112 (link 114).

Referring to Fig. 3, the inference engine 111 communicates the constraints on a document to the truth maintenance system 112 in the form of justifications, i.e. propositional clauses (link 113). A justification comprises three parts:

(1) Consequent in the fact in the inference engine which is inferred.

(2) Antecedents are the facts which are used for the inference rule.

(3) Informant is supplied by the inference engine to explain the inference in more detail.

In the context of the present invention, each justification is interpreted as a simple propositional definite clause or Horn clause comprising antecedents An and consequences Cm. For example, a clause *A1,A2,A3...An => Cm* means that if the antecedents, *A1,A2,A3...An*, are believed in the current context, then that implies consequent *Cm*. It has been found that most of the constraints or justifications needed to control the workflow in an electronic procurement system (EPS)

can be formulated within the propositional calculas for the truth management system 112. Antecedents (An) and consequences (Cm) are represented within the constraint network in the truth maintenance system 112 in the form of "nodes". The most common query posed by the inference engine 111 to the truth maintenance system 112 is whether a node (i.e. consequence Cm) logically follows from the current database state (i.e. the antecedents An stored in the W *data* component 104) for the document or process.

As such, the specification of the truth maintenance system 112 is defined by two sets of data which are stored in the W *data* component 105 of the workflow item 103. The two sets of data comprise: (1) a set of antecedents (i.e. A1, A2,...An), and (2) a set of consequences (i.e. C1, C2,...Cm). These two sets of data will evolve as the various workflow processes evolve. As a result, the truth maintenance system 112 can correctly answer queries on the contents of these two sets at the moment the query is made. A fundamental task of the truth maintenance system 112 is to answer queries about what the nodes are holding (i.e. what is believed in) given the current set of enabled justifications. The general implementation aspects of a truth maintenance system in accordance with the present invention are within the understanding of one skilled in the art and are further described in Building Problem Solvers by Kenneth D. Forbus and Johan de Kleer available from The MIT Press, Cambridge, Massachusetts.

According to another aspect, automatic routing of workflow items 103 is provided by encoding routing information as justifications, i.e. sets of propositional constraints, which are also stored in the W *data* component 104 of the

workflow item 103. The propositional constraints provide routing information and the engine 106 uses the routing information to route the workflow item 103 without any further communication with a workflow server 101 (Fig. 2).

As the workflow mechanism is designed to work in a connectionless environment, such as the Internet, changes in the organization structure and business rules are not reflected immediately until the workflow item is downloaded again. However this can be solved by implementing a notification scheme when business rules change or the organization structure changes. For example, the clients 102 may access a server 101b (e.g. Website) to download the update, or the clients 102 may be automatically notified by the server 101b, e.g. using e-mail.

The operation of the workflow mechanism, i.e. the workflow engine 106 is now described in the context of an electronic procurement system, such as the World Purchasing Professional™ (WPP) system from the IBM Corporation. Briefly, the WPP system first records "justifications" in the truth maintenance system 112. As the document progresses, the WPP enables or retracts antecedents (i.e. An) in the truth maintenance system 112. The WPP (i.e. work management system) queries the truth maintenance system 112 about the current state of the document. The truth maintenance system 112 utilizes the recorded justifications (i.e. propositional constraints) in the constraint network to provide the current state of the document.

For the workflow engine 106, the state of a document is a consequence of various antecedents (An) which are currently believed in. For example, the antecedents (An) for

a document may be represented and partitioned into the following sets:

(1) Role of the current user;

(2) Privileges of the user;

(3) Action of the user;

(4) Type of the document being worked on;

(5) Original state of the document;

(6) Resultant states of components (i.e. Line level states, etc.);

(7) Result of Business Rule 1;

(8) Result of Business Rule 2, etc...

As the processing of the document continues, the WPP will choose or enable one antecedent from each of the above sets of antecedents. The truth maintenance system 112 utilizes the justifications already recorded to answer which consequences are now believed in and the inference engine 111 determines the new state that the document can have. This feature is further described by considering the processing of a "Purchase Requisition (PR)" document. The Purchase Requisition document has the following sets of antecedents:

(1) Role of current user (i.e. Initiating Officer, Supporting Officer, Approving Officer);

(2) Privileges of the user (i.e. Can browse, Can approve, Can modify);

(3) Action of the User (i.e. Modifying, Approving, Browsing);

(4) Type of the document (i.e. PR (Purchase Requisition), PO (Purchase Order), ITT (Invitation To Tender), ITQ (Invitation To Quote))

(5) Original state of the document (i.e. Work in Progress, Approved, Rejected, Ready for Approval, etc...);

(6) Resultant states of components (i.e. All Line Items Approved, Some Are Rejected, Some Are Being Investigated, All Are Rejected, No Action Taken, etc...);

(7) Business Rule 1 (i.e. PR Value < 30,000, PR Value More Than 30,000).

For this example, the following justifications (i.e. propositional clauses) are encoded in the truth maintenance system 112:

(1) Set1(InitiatingOfficer) & Set2(canApprove) & Set3(Approving) & Set4(PR) & set 5(WorkInProgress) & set 6(NotApplicable) & set 5(NotApplicable) = = = = > Work in Progress by Initiating Officer.

(2) Set1(Approving Officer) & Set2(canApprove) & Set 3(Approving) & Set4(PR) & Set5(ReadyForApproval) &

Set6(AllLineItemsApproved) & Set7(Value < 30,000) = = = = = > PR ready for PO process.

(3) Set1(ApprovingOfficer) & Set2(canApprove) & Set3(Approving) & Set4(PR) & Set5(ReadyForApprovel) & Set6(AllLineItemsApproved) & Set7(Value > 30,000) = = = = = > PR ready for Tender process.

During the processing of the document (i.e. workflow item 103), the WPP enables antecedents (for example, "Initiating Officer can Approve") from each set, and the truth maintenance system 112 provides the WPP with the state of the document depending upon which justification is true and therefore the associated consequence is to be believed in.

The workflow engine 106 according to the present invention takes advantage of the fact that most of the justifications can be created beforehand, i.e. during the business process analysis stage. The truth maintenance system 112 appropriately stores these justifications (i.e. antecedents and consequences) in a relational database or in the form of persistent objects which are being used for the application. As work continues, the WPP merely enables or disables justifications, and the truth maintenance system 112 provides the resultant state of the document(s) (i.e. workflow item) by solving the enabled justifications.

The workflow engine 106 according to the present invention provides the capability to dynamically add new justifications. The new justifications are recorded in the truth maintenance system 112 by the WPP. Accordingly, the workflow management process can be refined by introducing or retracting justifications at any time, without the need for

further coding. New justifications can be entered by providing a user interface in the WPP where the user drags and drops various antecedents from the sets of antecedents which define a process. The particular implementation of such a user interface is within the understanding of one skilled in the art.

In summary, the workflow mechanism according to the present invention includes a truth maintenance system 112 in which rule(s) can be attached to each node in the constraint network, so that if a node is believed in, it can trigger those rules which perform a business task. It will be appreciated that the resulting code for the inference engine 111 and the truth maintenance system 112 can be very compact. This means the inference engine 111 and truth maintenance system 112 can be downloaded as part of the applets in order to track the states of the document which is downloaded to be worked on at a remote client 102 (Fig. 2). As the workflow item moves from machine to machine, it carries its context with it. In other words, the workflow item does not need a connection to a workflow server to determine its state. Because the truth maintenance system records all the justifications, the truth maintenance system 112 can also provide an explanation of the consequences for the workflow item. This means that the state of a workflow item at any time can be determined. Such a feature is particularly useful in business re-engineering applications.

The present invention may be embodied in other specific forms without departing from the spirit or essential characteristics thereof. Certain adaptations and modifications of the invention will be obvious to those skilled in the art. Therefore, the presently discussed

embodiments are considered to be illustrative and not restrictive, the scope of the invention being indicated by the appended claims rather than the foregoing description, and all changes which come within the meaning and range of equivalency of the claims are therefore intended to be embraced therein.

WHAT IS CLAIMED IS:

1. In a workflow management system for a stateless environment, a mechanism for managing a workflow item in the system comprising:

(a) a data module for data elements associated with the workflow item;

(b) a semantics module for rules associated with the workflow item;

(c) an engine for applying the rules to the data elements;

(d) wherein said data module, said semantics module and said engine are contained in said workflow item and transported with said workflow item when said workflow item is moved in the stateless environment.

2. The workflow mechanism as claimed in claim 1, wherein said stateless environment comprises the Internet.

3. The workflow mechanism as claimed in claim 1, wherein said engine comprises an inference engine and a truth maintenance system and means for recording justifications associated with the workflow item in said truth maintenance system, and said truth maintenance system including means for providing an inference about the state of the workflow item in response to a query from said inference engine.

4. The workflow mechanism as claimed in claim 3, wherein said means for providing an inference comprises a

* C L M S *

constraint network, said constraint network including propositional clauses for inferring the state of the workflow item.

5. The workflow mechanism as claimed in claim 4, wherein said propositional clauses are formed from one or more antecedents and a consequent, wherein said consequent results when the antecedents are true in the current context for the workflow item.

6. The workflow mechanism as claimed in claim 1, further including means for encoding routing justifications in said truth maintenance system, and said truth maintenance system including means for providing routing information associated with the workflow item in response to a query.

7. A method for processing a workflow item in a stateless environment comprising client machines and servers, said method comprising the steps of:

(a) storing data elements associated with the workflow item in a data module;

(b) storing rules associated with the workflow item in a semantics module;

(c) loading said data module and said semantics module with the workflow item;

(d) loading an engine in the workflow item together with said data module and said semantics module;

(e) routing the workflow item together with said loaded data and semantics modules and said engine to one or more of the client machines in said stateless environment;

(f) utilizing said engine to apply rules from said semantics module to data elements in said data module to process the workflow item on the associated client machine independently of the server.

8. The method as claimed in claim 1, further including the step of loading routing information associated with the workflow item.

9. Data storage media recorded with a workflow management computer program, in combination with a computer in the stateless environment and loaded with an operating system and runtime library means, equipped to read into memory and execute program data from the data storage media, comprising a method in accordance with claims 7 or 8.