Supporting optimization of business-to-business e-commerce relationships

William Kuechler Jr. a,*, Vijay K. Vaishnavi b, David Kuechler b

a Department of Accounting and Computer Information Systems, University of Nevada at Reno, Reno, NV 89557, USA
b Department of Computer Information Systems, Georgia State University, Atlanta, GA, USA

Abstract

Much current e-commerce subscribes to very simple interaction models. Many of the potentialities of e-commerce are identical to those that have been under study for some time in the field of automated workflow management systems. In this paper, we describe a new workflow interoperability model, the monitored–nested model (MNM), and show that it can support optimized, extended e-commerce transactions that are not supported by current models.

Like other interoperability models, MNM is dependent on process activities, and thus is brittle under real-world conditions of process evolution. This is overcome by augmenting the model with goal-based meta-data and the use of a coordination inferencing algorithm. ©2001 Elsevier Science B.V. All rights reserved.

Keywords: Workflow; E-commerce; E-business; Inter-organizational systems

1. Introduction

Recent studies by technology consulting groups predict more than one fourth of all business to business (B2B) purchases will be transacted on the Internet by 2004—a dollar volume 10 times that of Internet consumer purchases. The explosion in Internet-based B2B is driven by economics—the Internet offers the potential for reduced prices for goods and reduced transaction costs, but this is not simply derived from the Internet as a communications infrastructure. The capability for (relatively) inexpensive electronic B2B communications has existed for some time in highly evolved form, as witnessed by EDI. Newer, Internet based e-marketplaces as they are currently conceived overcome some of the problems encountered with traditional EDI, and constitute essentially a better–cheaper-EDI.

As business-to-business e-commerce moves closer to its full potential, it will progress beyond a better–cheaper-EDI to the support of business relationships that match or exceed the dimensionality of non-electronic relationships. However, support and optimization of such relationships require more sophisticated models of interaction than those currently in use.

1.1. Beyond EDI to online trading communities

Traditional EDI is a one-to-one technology: buyer and seller must locate each other and then perform substantial work to link their systems. The new
Internet-based online trading communities, such as Ariba [1], i2 [7], and CommerceOne [3], in addition to enabling B2B transactions so facilely that they have been termed e-commerce vending machines [12], are true markets. Within each community, an ontology—common business processes and electronic documents including product definitions and pricing—is defined in a standard language (XML) and published for universal access. Product offerings and prices from multiple suppliers can be electronically scanned and processed at very low error rates leading to predictions of billions of dollars per year in procurement cost savings.

1.2. Improving online trading communities

The efficiency of many operations in e-commerce transactions is less than optimal because they operate from very simple relationship models. Essentially, current online trading communities automate the most basic order–receive–pay scenario:

- The purchaser electronically scans the marketplace and chooses a product
- The price, quantity and a delivery date are agreed upon using standardized XML-described order documents.
- An electronic invoice is sent from supplier to purchaser at or about the time of delivery.
- The purchaser schedules automated payment on confirmation of delivery.

Notice that there is no provision in this scenario for multi-party transactions and no provision for exceptions. As e-commerce matures, we believe online trading partner relationships will rapidly push toward the sophistication and complexity found in non-automated business relationships, which are desirable precisely for their ability to integrate the efforts of multiple legally independent entities while accommodating exceptions and changes.

In the basic order–receive–pay scenario, transactions between companies are considered to be atomic, that is, no detail on the execution of the transaction is available. The lack of detail makes it impossible for companies to interact with the richness that current manual relationships possess. For example, delays in production are hidden until the delivery date is exceeded. Or, changes in specification that appear minor to one party but are significant to another remain unknown until the delivered product is closely inspected, possibly on the assembly line floor!

1.3. Workflow management system concepts and e-commerce

Workflow management systems (WFMS) are software systems that facilitate, augment and sometimes control the flow of work within and (for our purposes especially) between organizations. As interorganizational WFMS become increasingly common and as the interaction between WFMS becomes increasingly web-based, much automated workflow enactment becomes by definition electronic commerce [22].

The notion that automated business processes, executed over the Internet between multiple organizations, are the future of electronic commerce is at the core of the visions of authors from many fields. It also figures prominently in the literature of many commercial Internet marketplace-hosting organizations (see Fig. 1). According to Sheth et al. [17],

![Workflow Diagram](image)

Fig. 1. Commercial e-commerce service providers view of interorganizational workflow. A composite view taken directly from the marketing literature of commercial e-marketplace hosting organizations [1,3,7].
“...we see processes as an organic component of any enterprise application integration or e-commerce solution. In this sense, workflow-process technology will conduct the emerging networked economy from behind the scenes.” Jain et al. [8] see agent enacted workflows and a form of intention-driven e-commerce transaction similar to the one presented in this paper as the enabling technology for virtual corporations. These visions propose workflow-as-e-commerce in the near future since by definition the automation of formal process definitions is workflow management. Charles Petrie, executive director of the Stanford Networking Research Center, writing with Sunhil Sarin, a commercial WfMS developer and researcher, state the case even more strongly: “Internet mediated workflow will be the most important technology of the early 21st century” [14].

We propose that many of the problems and potentialities that exist for e-commerce are exactly those that have been under research for some time in inter-organizational WfMS. Workflow interoperability models provide the foundation for the study of relationships between autonomous workflows and for the technology to model and enact such relationships.

2. Workflow interoperability models and business scenarios

Although they are not complex, workflow interoperability models are best discussed in relation to business scenarios that give depth and concreteness to the concepts. This is the approach taken by the Workflow Management Coalition, an international group of commercial WfMS producers and WfMS researchers. The order fulfillment example given here (see Fig. 2) is a minor variation on a benchmark used in Workflow Management Coalition documents to give a standard for comparisons. The example is simple, but is capable of illustrating most points that arise in far more complex real world interactions. We will later expand this example to illustrate desired optimizations of B2B e-commerce relationships that go beyond the order–receive–pay model implicit in EDI type interaction. We propose two expansions of the base scenario as part of a benchmark suite for extended WfMS interoperability.

In the base scenario illustrated in Fig. 2, all participants are assumed to have a WfMS. Retailer sells an item of furniture that must be custom-made and drop-shipped from Manufacturer. Manufacturer is responsible for contracting for shipping the item to customer. Each column of the figure represents the progression of high-level tasks required for that portion of order fulfillment processing required of each of the cooperating business entities. The high-level tasks, such as Schedule Production may be instantiated in the Manufacturer’s workflow by a series of lower-level activities that accomplish the higher-level task; this detail has been omitted for clarity. Each entity’s tasks are interleaved, taking place in parallel. The events proceed in the conventional WFMC scenario as they have been numbered in Fig. 2, from 1 to 10. The shaded segment of the figure depicts the optimized workflow discussed below.

For even the base scenario to proceed, the WfMS in each of the cooperating organizations require information on:

1. the location of contractors on which it depends (interoperates)

![Fig. 2. Basic and optimized order fulfillment via interoperating processes.](image-url)
2. a format for the communications (requests for service) that will be understood by the other WfMS
3. how the workflow that is requesting service is to behave during the performance of service
4. an expectation of and format for a return communication(s) indicating various states of the requested service(s).

Information items 2, 3, and 4 can be abstracted to a workflow interoperability model; reference to common interoperability models will be useful in discussing improvements to the ordering process. The Workflow Management Coalition defines three interoperability models in its interoperability document [21]: chained interoperating processes, nested interoperating processes, and parallel-synchronized interoperating processes (see Fig. 3).

The Chained Interoperability Model is the least complex in that it requires a WfMS to possess the least information about cooperating WfMS. Chained processes (or subprocesses) are invoked by one WfMS in another WfMS and no further interaction takes place with the subprocess. In Workflow Management Coalition terms, they are ‘trusted’ processes and delegation to them is total. Chained processes are defined in WPDL (the Workflow Management Coalition’s process description language) as activities whose type is subprocess and whose execution mode is asynchronous.

In a Nested Interoperability Model, once the call is made by a process to a subprocess, the calling process suspends operations until the subprocess completes. Nested subprocesses are defined in WPDL as activities whose type is subprocess and whose execution mode is synchronous. Note that since WPDL supports multiple threads in a workflow, the overall workflow that issues the nested subprocess call need not suspend; only the calling thread suspends. Nesting may recurse.

In the parallel-synchronized Interoperability Model, workflows run in parallel on different WfMS and are required to achieve periodic synchronization points. Parallel-synchronized interoperability has been designated by the Workflow Management Coalition as outside the scope of their current interoperability specifications. This is due to the complexity involved in generalizing this type of interaction between systems, and its lack of robustness under activity changes, a problem detailed in Section 4.

The workflow interaction in the order fulfillment scenario is modeled by the nested interoperability model. Nested subprocesses correspond exactly in
the example to the calls for service from the Retailer to the Manufacturer and from the Manufacturer to Shipper.

2.1. Optimizing B2B relationships: proposal for a benchmark scenario

As presented in WFMC literature the order fulfillment scenario requires confirmation from both Shipper and Manufacturer before the Retailer invoices the client. Suppose, however, that Shipper’s performance became predictable. A desirable optimization under such conditions (one first widely demonstrated by the Japanese) would be to use that predictability to decrease the overall business cycle time by invoicing the client at an earlier point—say when the shipping of the item was scheduled—to improve cash flow. However, the invoice must come from the Retailer, and using simple nested interaction, the Retailer is two levels removed from the information required. That is, the Retailer invoked the Manufacturer’s process and has no information on that process state until it completes. The Manufacturer’s process invoked the Shipper’s process, and has no information on that process state till it completes. There are thus two levels of isolation using the nested workflow interoperability model from the information required to optimize the overall process.

The suggested improvement in overall workflow is shown in the shaded section of Fig. 2. The dashed arrow from activity 5 Manufacturer’s Schedule Shipping directly to Retailer’s activity 10 Invoice Customer graphically illustrates this modification. Invoicing under the optimized scenario takes place prior to activities 6 through 9 in the original scenario. This scenario constitutes the first of our proposed suite of scenarios for extended WfMS interoperability; the second scenario is described in Section 5.

The suggested optimization, however, requires a more sophisticated interaction model than any existing interoperability model. A key functionality of the required model is that the processes of cooperating business partners be at least partially visible to each other. This allows each cooperating business partner to modify its processes, including the initiation timing of all of its activities and all other external workflows for which it is responsible, in an optimal manner.

3. Monitored–nested interoperability model

A slight increase in conceptual complexity over simple nested processes confers considerable potential for flexibility and optimizability of WfMS (Fig. 4). We call this new interoperability model the monitored–nested model (MNM); in it the tasks of the calling activity have been expanded to include monitoring the state of the subprocess it has invoked, and the initiation of new process activity in its own workflow environment when a monitored subprocess state has been reached. Implementation of the activity initiation portion of the new model can be a simple Java-like event registration:

(a) the monitored state in workflow B is registered with B and as a precondition for a workflow activity in A

(b) on achievement of state, a message is sent from B to A

The simple nested model is a limited special case of the monitored–nested model in which the only con-
A task monitored by workflow engine A is the end point of the task invoked by A in workflow engine B. The monitored–nested model is similar to the parallel-synchronized model in that the synchronization points of the parallel-synchronized model may be considered equivalent to monitored state communications. The models are different in that in the parallel-synchronized model workflow engine A is not responsible for initiating the workflow in workflow engine B. Moreover, synchronization communications do not explicitly initiate other workflow threads as state communications between workflows do in the monitored–nested model. Finally, in both of the nested models, activity in the workflow thread in engine A that initiated the external workflow suspends for the duration of the invoked activity whereas in the parallel-synchronized model activities proceed continuously in both workflows.

3.1. Use and benefits of MNM

According to information-processing-based organization theory [5], certain forms of organizational structure (or in this case, transaction structure) emerge whenever multiple tasks involving uncertainty must be coordinated efficiently. A common name for one frequently observed such form is the contractor–subcontractor relationship in which a complex process is accomplished by multiple entities (the subcontractors) under the monitoring and supervision of an entity responsible for the overall result (the contractor). This situation is ubiquitous in business; entire industries are currently based on the contractor–subcontractor relation, which is formally modeled by the MNM model. The construction industry is prototypical; different aspects of construction are handled by specialized subcontractors working under the supervision (monitoring) of a general contractor. Airplanes, missiles, ships, large software systems and many other complex, assembled products are constructed using this coordination model.

The primary benefit of MNM type transactions over simple nested task interoperation is efficiency in the face of high natural and unavoidable variability in the completion times for the individual activities that comprise the workflow. By monitoring the status of activities, as the uncertainty of those activities becomes progressively lower, it is possible to continuously re-plan the overall process and optimize the result relative to simply waiting for each activity to complete.

An example of MNM type transactions in current business practice (though established ‘manually’) is the much cited Wal-Mart (WM)/Proctor & Gamble (P & G) alliance. In this example, status information from WM’s normal inventory and procurement workflow is opened to its trading partner, P & G. P & G actively monitors sales information in WM’s system and this information triggers in P & G’s production and supply workflow the ordering, and actual restocking on WM’s shelves, of P & G products. The interaction is illustrated in Fig. 5.

3.2. Implementation problems and solutions

The problems involved in the actual implementation of workflow-based B2B e-commerce transactions and specifically MNM type transactions derive...
from the difficulty in coordinating WfMS from different manufacturers. Like many coordination problems, these are communications problems derived from differences in WfMS architectures, process models, communication protocols, and so on. In the workflow community all such problems fall under the heading of interoperability: the ability of heterogeneous WfMS to work together to monitor, coordinate and to some degree, control each other in working toward a common goal.

Much recent research on workflow management systems from both universities and WfMS manufacturers has been focused on interoperability. This research has been quite successful and practical solutions have been found for many early interoperability problems. The problems that have been solved, those currently under research and those raised by the MNM interoperability model presented in this paper are best understood with reference to a protocol stack model (see Fig. 6). The model partitions interoperability issues into levels of abstraction. At each level of the stack above the physical, incompatibilities at one level are overcome by meta-level information supplied by the next higher level.

Physical and communications level interoperability issues have been effectively solved by standardization on the WWW as the communications infrastructure and low level communications protocol set (TCP/IP and HTTP). Early efforts at linking WfMS engines, the next higher level of abstraction in coordinating workflows, were directed at a standardized API’s for the engines. Many of these efforts were based initially on remote procedure call (RPC) syntax and later on CORBA and IIOP standards. However, these direct and closely coupled enactment engine links have been largely superseded by higher level XML based protocols—effectively, high level languages by which one WfMS may request services of another. Specifically, the interoperability problems introduced by different WfMS software architectures are overcome by Wf-XML, a high level protocol proposed for standardization by the WfMC.

However, Wf-XML assumes a single process definition is shared by coordinating agents. When this assumption is violated, that is, when processes change dynamically and autonomously, then interoperability problems arise at the level of process definition (see Fig. 6). Several solutions to interoperability prob-

![Fig. 6. WfMS interoperability layers [19].](image-url)
lems at this level have been proposed [10,15,20], each aimed at solving a different portion of the broader issue. What all solutions implicitly share is the notion that the process definition language must be augmented with additional information to support dynamic modification. In our research, information on process intention is used to support robust MNM interoperability; MNM thus remains a research model, currently implemented only in our prototype, pending widespread adoption of a solution to the dynamic process change problem. However, the ubiquity of the model in manual commercial transactions argues strongly for its adoption into workflow enabled e-commerce once remaining implementation problems are solved. In the next section, the problem of dynamically changing processes is presented in detail to facilitate a discussion of our solution to the problem.

4. The dynamic process evolution problem

The monitored–nested interoperability model can easily support the desired transaction optimizations we have suggested for the order fulfillment example discussed above. Since all activities in subordinate workflows are visible to Retailer, sufficient information is available to optimize the overall process. There is, however, a significant practical problem with the deployment of the model or any model dependent on process states: it is brittle under real world conditions of dynamic process evolution. The logic to illustrate this is straightforward:

- The monitored–nested model depends on knowing the states of processes
- The state of a process in a WfMS is inevitably linked to specific activities
- But the processes being monitored take place outside the control of the monitoring agent and may change at any time
- And when an activity is changed, even to a simple equivalent activity, WfMS do not have sufficient information to link the monitored state (what is important) to the new activity.

Relative to our example, the Manufacturer and the Shipper are autonomous and may change suppliers and internal operations (activities) at any time for a variety of beneficial and necessary reasons. In Fig. 2, the optimization of early invoicing is dependent on Retailer receiving notice that shipment of the product has been scheduled. The completion of Schedule Shipping is effectively a high-level process state, which can be satisfied in any number of ways. In the specific workflow being used, however, the state is linked to the specific activity Schedule UPS pickup. Now suppose that Regional Shippers (RSI) offers the Manufacturer a better price on shipping its goods. The specific activity in the workflow that indicates the state [completion of] Schedule Shipping now becomes Schedule RSI pickup. Due to the lack of knowledge about work processes inherent in most WfMS, this simple replacement of one activity by an equivalent is sufficient to confuse the monitoring on which the optimized process coordination depends—the state communication between workflow engines will never be sent. Of course, in the general case, the changes to a work process can be much more complex than the simple substitution of an activity.

The interrupted coordination will undoubtedly be brought to human notice eventually. Likewise, coordination can be reestablished through human intervention—specifically, by reprogramming the system to link the state achievement notification (called a trigger in WfMS terms) to the new activity. However, the driving intention behind e-commerce is to automate transactions to the highest degree possible. A more ideal situation would be to incorporate into the WfMS the ability to:

1. link triggers to high-level states of the process
2. have the system recognize which activities correspond to achievement of the high-level states, even when activities are changed.

Naturally such automatic adjustment would be subject to human approval, but given a sufficiently high reliability for such a system, human intervention could be minimal and coordination-cost savings of the overall system significant. We have developed a technique for automating the re-coordination of activity-based process state notification under process change that makes use of goal meta-data about the process.
4.1. HOPI and goal based meta-data for processes

Conventional workflow management systems contain no information about their processes other than [11]:

- The activity
- The role (actor type) required to perform the activity
- Resources needed to perform the activity
- Precedence information, frequently in the form of pre and post conditions for the activity

The monitored–nested model requires more information about the process than that shown in the above list since the activities may change at any time. The flexibility we seek for the monitored–nested model is enabled by meta-data, essentially data about data, which can be used by systems to reason about changes in the activities that constitute a work process. The more meta-data is available in standardized form such as XML descriptions for incorporation into inter-organizational information systems, the more sophisticated and efficient are the automated relationships that are practically realizable. From an information-theoretic perspective, meta-data and the sophisticated B2B relationships, it enables lower uncertainty in those relationships and thus permit a variety of cost-saving efficiencies.

Our research has shown that one type of information about processes, the goals of the process and its activities, when properly structured, allow inferences to be made about changed processes. Goal information permits these inferences because goals are far more stable than activities [9]. Our technique essentially attaches coordinating triggers to goals, which at some level remain constant even as activities to achieve the goal are dynamically changing. That is, the set of subgoals, functions, and activities originating from a goal are always recognizable as “the same as” in some important sense, any other set of subgoals, functions and activities that also satisfy that goal [2]. It is precisely the ability of an intentional structure to change its instantiation while retaining the explanatory power of the higher-level nodes that gives our technique its utility.

A Hierarchical Overlay of Process Intention model (HOPI) is a tree structure of goals and subgoals (intentions) terminated by the activities that actualize the goals which are attached to generalized functions, as shown in Fig. 7. The general applicability of HOPI derives from its basis in the widely accepted decomposition model of problem solving activity [18]. With reference to Fig. 7, all processes begin as a high level intention to accomplish a complex (multi-activity) action. This intention is called the root goal in HOPI. The root goal is hierarchically decomposed during the design process into progressively less abstract sub-goals. The decomposition takes place recursively in a problem domain or problem space. At some point in a successful decomposition, the goals are sufficiently con-
crete that techniques for accomplishing the sub-goals can be assigned to them.

The path from the root goal (R) to any activity in a work representation is termed the \textit{intentional context} of the activity since tracing a path from activity to root goal explains the activity at progressively higher levels of abstraction. For example when we ask why the activity is performed, we explain that it is necessary to implement the generalized function to which it is attached. The reason for the enactment of the generalized function is the operationalization (satisfaction) of the sub-goal to which it is attached, and so on.

HOPI does \textit{not} provide a process description or representation. Rather, HOPI captures additional information that can be used in conjunction with a process representation for reasoning about processes, and especially for drawing inferences about process changes. As such, it is general and applicable to any representation. Naturally HOPI and the coordination model based on it depend on the availability of intentional information. Today a strong case can be made from a review of the knowledge management literature that such information is being sought and captured for its considerable value even when it is not structured into a formalism such as HOPI [4]. At least one other workflow architecture has incorporated goal information into its process descriptions [13]. This system uses goals for dynamic re-planning of a workflow within a single organizational entity rather than for coordination of multiple workflows, but the increased functionality of that system supports the high utility of goal information about process.

Several other mechanisms are under research as potential solutions to the interoperability problems at the process definition level. These typically involve graph theoretic methods with formal logic, e.g. Petri nets, to infer the similarity of changed to original process models [15,20]. While each technique has its merits, we believe the hierarchical nature of HOPI is more flexible in the sense that more radically altered process descriptions can be recognized as “similar” using this scheme.

4.2. \textit{How it works}

In a WfMS using HOPI, each workflow, described in virtually any format, is linked to its goal structure. The triggers that indicate the states of process that should be monitored are linked to specific activities \textit{and} portions of the goal structure that indicate the meaning of the state (see Fig. 8). In any process model, the process state must also be linked to a specific activity, since it is only by receiving confirmation that a concrete activity has been started or finished that the state can be observed. Within the HOPI conceptualization the completion of a set of activities linked to a goal is said to satisfy the goal and goal satisfaction takes the process to a new state. If a new altered process is substituted for the original

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig8.png}
\caption{Fig. 8. Intentional specification of activity monitoring (cf. Fig. 2).}
\end{figure}
process, providing the new process has a HOPI also, an intelligent subsystem can compare the two process descriptions and determine the new concrete activity to monitor to indicate achievement of a high level state.

The HOPI inferencing algorithm [10] works by finding the best match in the new process goal tree to the path from the root goal to the original trigger activity in the original process. Fig. 8 shows the algorithm graphically as it relates to the order fulfillment scenario of Fig. 2. In both figures and the scenario, the activity that indicates the state of Completion of Schedule Shipping is the activity descended from that goal. Irrespective of how that activity changes, or even how many activities are required to satisfy the goal under different process implementations, an appropriate concrete activity to monitor to alert cooperating workflows of the achievement of state can always be located.

The Original Process of Fig. 8 corresponds to the Manufacturer’s workflow as both Manufacturer and Retailer understood it at one time (say, \( t_0 \)) in their relationship. However, at some future time (\( t_1 \)) the Manufacturer changes vendors (Shippers) and re-instantiates the actual activities of the workflow as shown in the Changed Process segment of Fig. 8. Without the HOPI interpreter, coordination between the Manufacturer and all business partners who depend on the specific activity Schedule UPS Pickup to trigger activities in their workflows is disrupted until all parties have been formally notified and their WfMS reprogrammed. This is a high cost modification. With HOPI, the change is detected, corrections made automatically, and human supervisory personnel notified of the change. Human (high cost) intervention required for process changes is, in many cases, limited to quickly assessing, and then accepting the modification suggested by the system.

5. Enabling more complex business relationships with monitored–nested workflows

From the ability to correct for minor coordination disruptions arising from dynamic process change (via HOPI) emerges the potential for more sophisticated e-business relationships than are now possible. Consider again the base order fulfillment scenario depicted in Fig. 2. Even greater time efficiencies than those that result from the optimization shown in Fig. 2 are possible if the Retailer serves as a coordinating contractor for both the Manufacturer and the Shipper. This changes the relationship between service providers entirely, as one WfMS is now the coordination nexus for multiple services. This type of coordinating relationship is frequently termed a contractor–subcontractor relationship and appears as the basic interaction model for business areas as old as the construction industry and as new as virtual corporations. The order fulfillment scenario reorganized under a contractor–subcontractor model is shown in Fig. 9.

Three of the many business reasons for adopting this arrangement are quality control, leverage with suppliers and single-source service delivery to the

Fig. 9. Order fulfillment via a contractor–subcontractor relationship.
customer. We propose this as the second scenario in our suite of benchmark WfMS scenarios.

The Retailer (center column) is now the instigating force—the primary contractor—in control of the initiation of the sub-processes in Shipper and Manufacturer. With the ability to monitor the workflows of its subcontractors and preserve their autonomy (that is, monitor while not limiting their ability to modify their internal processes) the Retailer can orchestrate the entire multi-entity process. Scheduling of shipping has been taken over in this scenario by the Retailer who monitors the Manufacturer’s process to observe Schedule Production before activating its Schedule Shipping activity. Retailer then monitors the Shipper’s workflow to determine when the activity Schedule Van has been completed. This initiates the Invoice Customer task in its own workflow. In proposing this scenario as an optimization, we make certain assumptions that may not be true in all business environments. However, the value of HOPI is its ability to allow any desirable inter-organizational workflow to be put in place and then maintained at a relatively low cost under conditions of gradual autonomous process evolution.

5.1. Leveraging today’s online trading communities with HOPI

We discussed at the beginning of this paper how the recent development of online trading communities has been responsible for much of the increase in e-commerce. These same Internet sites can be easily leveraged to more sophisticated trading relationships using the techniques we have described as well as other meta-data enabled methods. The key technology underlying this capability is the nearly universal use of the meta-description language XML (eXtensible Markup Language) in these new trading communities [6].

XML-izing HOPI is a simple matter of specifying a tree (or more generally a directed-acyclic graph—see Fig. 7) in XML. Fig. 10 shows a complete XML document type definition (DTD) for HOPI, as it is implemented in our prototype. We believe the fundamental structure of HOPI and thus of the DTD is unlikely to change. However, as additional richness is added to the model, we can foresee attributes being added to each element specifying XML-LINKs—hypertext references—to lower level XML documents which give structure to each of the currently atomic nodes.

A HOPI tree formally terminates in leaf nodes that represent activities. However, in what is termed a generic HOPI, branches may terminate at function or even subgoal nodes. The level to which the HOPI is defined reflects the level of constraint on the enactment of the root goal. An infinite number of generic HOPIs can be specified for each root goal reflecting the degree of delegation of the process that is acceptable—the greater the specification, the less delegation. A generic HOPI in turn may be expressed by an infinite number of activity sets that implement its goal set the intentional process specification. The implementing activity sets are called instantiations of the HOPI, and in practical use a generic HOPI is linked to an instantiation expressed in any process description language.

XML product, form i.e. invoice, receipt, and process definitions already exist on multiple freely accessible trading sites [16]. In fact, the nature of these electronic marketplaces is to encourage public registration of and access to these descriptions. This is the mechanism by which accurate machine comparison of price and product data proceeds. Although the HOPI XML document type definition that allows it to function as a process description overlay is process definition dependent, only minor changes are required in the HOPI template to link lowest level goals to the activity descriptions of any XML process description. This type of maintenance is very economical given the benefits that result. XML
parsers capable of turning any well-formed HOPI XML specification into the data structure to be utilized by our inferencing algorithms are widely available as subroutines or dll’s or subprograms for virtually any programming environment. The inferencing algorithm itself, though outside the scope of this paper, is also easily implemented [10].

Once generic HOPI overlays have been published for a process, organizations can use their WfMS to enact optimized MNM transactions with suppliers and customers. In another paper we have described how HOPI goal trees are easily translated to XML messages within the Wf-XML standard protocol (http://stunt.cis.gsu.edu/process_coordination/xml_hopi.pdf). Wf-XML itself contains the message primitives necessary for one compliant workflow enactment engine to start and monitor processes in another engine.

Given that the WfMS which will interoperate are all Wf-XML aware, then a simple and efficient HOPI implementation would be as a wrapper layer surrounding the existing commercial workflow enactment engine. The subsystem/layer would operate at the level of Wf-XML messages and at that high level of operation, the impact on efficiency would be acceptable except in the most demanding applications. The HOPI layer would examine all incoming Wf-XML messages for HOPI content, and perform translations if required, much as Message Oriented Middleware (MOM) is used to provide data and message translation between heterogeneous legacy systems. Messages involving non-HOPI aware processes or workflow systems, or messages that did not involve HOPI computation would be passed through to the enactment engine unchanged. HOPI specific messages would be translated to the appropriate (series of) “standard” Wf-XML messages and then passed through to (or from) the enactment engine. Fig. 11 illustrates this process.

With reference to Figs. 8 and 9, our technique resolves the changed activity situation presented in Section 4 and illustrated in Fig. 8 as follows:

- **Retailer**, working from a published XML description of Manufacturer’s workflow, augmented with a HOPI overlay, determines it can optimize its process by monitoring when Manufacturer schedules shipping.
- **Retailer** issues a Wf-XML request to Manufacturer to monitor (advise when it has started) the activity Schedule–UPS–Pickup. Part of this request references the work definition Retailer believes is in operation at Manufacturer.
- **Manufacturer**’s HOPI subsystem intercepts the activity-monitoring request. It compares the work definition transmitted from Retailer with Manufacturer’s current workflow definition in which the shipping activity has changed (see Fig. 8). The subsystem translates Retailer’s request so that the monitored activity is the new, equivalent activity—the first activity descended from the Schedule Shipping subgoal.
- The subsystem then advises administrative personnel at both sites of the change it has detected and the suggested correction.

Fig. 11. HOPI processing implemented as a middleware service layer.
Though simple, this example illustrates both the flexibility that derives from the MNM model, and the robustness that HOPI structured intentional information adds to the technique.

6. Conclusion and implications for practice

Many of the problems and the potentialities of e-commerce are identical to those that have been under study for some time in the field of automated workflow management systems. As e-commerce evolves, the relationships between agents supported by the underlying technology must also evolve. In the workflow community, the study of relationships between autonomous workflows and the technology to model and enact them is termed interoperability modeling. We have shown that common interoperability models are not sufficient to realize more complex commercial relationships, which involve multiple parties and provide for considerable exception handling. The monitored–nested model of workflow interoperability we developed enables considerable flexibility and is sufficient for our proposed benchmark scenarios. Goal data about the processes, when structured in a hierarchy that reflects the design of the process, can be interpreted by a workflow subsystem to overcome some of the practical problems that attend sophisticated interoperability models.

The advent of multiple interoperable XML-based repositories on the Internet greatly extends the possibilities for augmenting basic inter-organizational workflow architectures with meta-data. Once research such as ours has established the type of information required to enable a rich business relationship, an XML description of that data can be posted to a globally accessible repository and downloaded by any trading partner. In many instances, similar to our technique for enabling contractor–subcontractor relationships with an overlay to conventional process descriptions, the additions to current systems can be made incrementally. A HOPI aware subsystem can be implemented as a middleware shell around any commercial WiMS which supports the WF-XML standard. In the same way, new e-commerce potentialities can be explored and evaluated, and the support technology debugged without large investments or disruption to existing processes.

Acknowledgements

This work is partially supported by NSF Research Grants, IIS-9810901 and IIS-9811248, and a research grant to the second author from the Robinson College of Business, Georgia State University.

References