

Association for Information Systems

AIS Electronic Library (AISeL)

ICEB 2007 Proceedings

International Conference on Electronic Business
(ICEB)

Winter 12-2-2007

Inter-Organizational Process Integration Problems

Frank G. Goethals

Follow this and additional works at: <https://aisel.aisnet.org/iceb2007>

This material is brought to you by the International Conference on Electronic Business (ICEB) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICEB 2007 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

INTER-ORGANIZATIONAL PROCESS INTEGRATION PROBLEMS

Frank G. Goethals, IESEG School of Management - France, f.goethals@ieseg.fr

ABSTRACT

In this paper, we discuss eight high-level problems that show up when partnering companies decide to set up cross-organizational processes. It is important in practice to be aware of the existence of these problems and to deal with each of these problems. The problems concern the facts that it is difficult to agree on pre and postconditions of tasks, that parties need to agree on how to trigger task executions, that investments have to be distributed among the parties, that service levels should be agreed upon, that partners should preserve the value of the functions that are executed in the process, that process ownership must be considered at an inter-organizational level and that partners may change over time

Keywords: Inter-organizational process integration problems, preconditions and postconditions, process ownership, service levels.

INTRODUCTION

For many years companies have been optimizing their internal functioning. With the advent of the Internet, however, information sharing among companies has become much more feasible so that the optimization effort is nowadays mostly no longer confined to that of individual enterprises: optimization can happen at the level of Extended Enterprises (i.e., collections of partnering companies [5]). Clearly, Extended Enterprises that are successful in their Extended Enterprise wide optimization effort will undercut other value chains that fail to do so and only optimize the individual components of the value chain rather than the value chain in its entirety. In this paper, I present an extract of the literature review of my PhD-dissertation [3], so as to discuss a number of inter-organizational process integration problems.

To create competitive end-to-end processes, companies within an Extended Enterprise need to deal with the following challenge: Partners should appropriately align the tasks that need to be executed in an end-to-end process and control the execution of those tasks. The term 'control' has many meanings [10,13]. With execution control in the previous sentence, we mean that some party (1) can initiate the task and that he (2) can monitor the task execution and steer the flow of the process. When looking for a way to stand up to the challenge, Extended Enterprises will be confronted with a number of problems. In this paper we present eight problems. To a large extent the problems we discuss here can be seen as modifications of the inter-organizational data integration problems we have discussed in [4]. Roughly stated, the problems we consider (i.e., the problems of which we think they may be important for companies in their decision w.r.t. the process solution) are the following:

- If we look at the building blocks of the process, the tasks, we notice that these blocks have to be arranged: the tasks should be executed in some order. The different partners may, however, have different requirements with respect to the order.
- While trying to find an appropriate arrangement of the building blocks, the companies may note that they have a different understanding of what the blocks at hand actually are. That is, the outcome they expect from some task execution may be different. Moreover, the outcome they expect from the entire process execution may be different too.
- Once it is known what tasks need to be executed and in what order they will be executed, it has to be decided how the task executions can be triggered.
- To realize the processes investments are needed. Partners have to agree on who will bear the costs and who will do the effort.
- The partners have to agree on what are the required service levels for the process, and for the individual tasks being part of it. At runtime, the service levels that are needed from one task depend on the service levels that are actually provided by other tasks.
- On itself, an individual task is often worthless. It is in combination with a number of other tasks that the function provided by one task becomes useful. The value of executing a single task depends on the value of the functionality of the entire Extended Enterprise process. In an Extended Enterprise partners should preserve the value of the functional proposition of the process in order to preserve trust.
- A party has to be designated who is responsible for making decisions with respect to the process: What are competitors doing, and should a new design be considered? Are new service levels required? Should some party get a rap on the knuckles? etc. This party should be enabled to do his job: he should get access to information about the current process, and get control over the execution of the process.
- The partners may change over time so that the parties that are designated to provide some service differ, and/or the parties that may want the execution of some service differ.

In what follows, we subsequently discuss each of these problems in more detail. Each time we (1) define the problem, (2) show how it is related to the challenge defined above, (3) discuss the relevance of the problem, especially in the context of the

Extended Enterprise, and (4) illustrate the problem.

To illustrate the problems, we will generally start out from the same exemplary process. The example is a fictitious process that could show up in the Personal Computer (PC) industry. Say there is a customer who wants a computer. A Reseller, for example the IT department of his company, can deliver computers. This Reseller then requires the computer Manufacturer to build a computer. This Manufacturer books the orders and produces the desired computer, but he relies upon a Shipper to pick up the computer to ship it to the Reseller, and upon a Factoring company to make sure the payment is settled.

Figure 1 shows a simplified BPMN (Business Process Modeling Notation) model of how the process between the different partners could look like. In BPMN, the solid arrows show sequence flows, the dashed arrows show message flows.

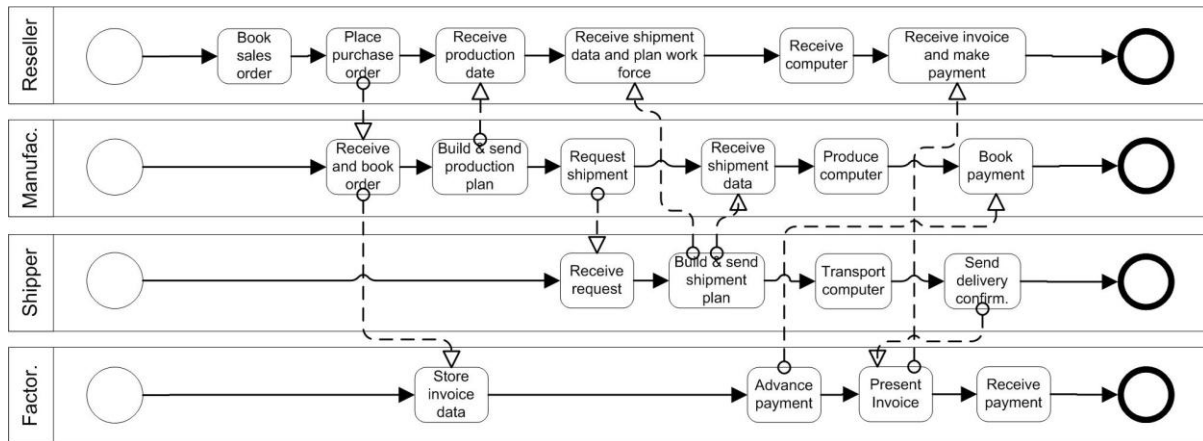


Figure 1: Example of a process

OVERVIEW OF INTER-ORGANIZATIONAL PROCESS INTEGRATION PROBLEMS

Problem 1: Companies Foresee The Execution Of Tasks In Different Situations

The different partners may have different expectations with respect to the situations in which a task will be executed; this in terms of previously finished and unfinished tasks, available capacity, timing, etc.

Problem Description

If different parties foresee different conditions for task execution the process may never be executed entirely. Partners may fundamentally disagree on the preconditions for each task, or may just not be aware of the fact that they disagree.

In [13, p35] a precondition is defined as “A logical expression which may be evaluated by a workflow engine to decide whether a process instance or activity within a process instance may be started.” Companies are only willing, able and planning to execute some task if the world they perceive is in an appropriate state. The other way around, other companies may expect them to execute some task if the world is in some specific state. If there is a difference between what one company expects the other to do on the one hand, and what the other company can, is willing and is planning to do on the other hand, the process cannot be executed successfully.

This problem has both, a task and a process component. A task can only be executed if the necessary preconditions are fulfilled. The preconditions can be fulfilled at different moments in the entire process, but (in general) the task will only be executed once during the process; it has to be agreed when that will be.

Relationship to the challenge

If there is no agreement about the order of execution, the execution of the tasks is clearly not decently controlled. In this case one cannot talk about the end-to-end process that is to be implemented, as different parties perceive the end-to-end process differently.

Relevance

An agreement on the preconditions is important because the system may otherwise enter some state of deadlock. This is especially true in Market B2Bi (as defined in [5]). In Market B2Bi companies can only deploy the ‘lowest common denominator’ process, possibly leading to a process that is not rendering the envisioned outcome [14]. In an Extended Enterprise (as defined in [5]) companies will be less reluctant to execute the tasks they are requested to execute even though their own preconditions are violated. This, however, does not imply that preconditions should not be negotiated. In the Extended Enterprise companies are likely to be willing to negotiate and to adapt preconditions upon tasks, while this is not true in Market B2Bi.

Illustration

The following examples show cases where there is actually no (agreed-upon) process that the companies try to execute.

The Reseller may not be willing to pay for the computer until the shipment has arrived, whereas the seller may not be willing to hand over the computer if no payment was made.

The Reseller may be willing to pay for the computer before he received the computer, but not be planning to pay before receipt.

The seller may be postponing the delivery of the computer until he received the payment. The lack of explicit agreement about

which of the possible process sequences (both are acceptable) is to be used actually suspends the process.

Say that the shipping costs are calculated ad hoc for each package, depending on traffic conditions, fuel prices of the day, etc. In this case the reseller may be willing to pay before delivery, but he cannot pay until he knows how much he has to pay, something that is only determined once the computer is delivered.

Problem 2: Different Parties Have A Different view On The Post-Conditions Of Tasks And Processes

The concept of delegation typically supposes that only a task executor knows how some task is executed; others are generally only supposed to know what the task does. Still, others may have a wrong perception of what the task does. A task is executed to achieve some results, some of which are not relevant or not visible for all parties. By extension, a process is executed to achieve some results, some of which are not relevant or visible for all parties.

Problem Description

Partners may have a different perception of what a task and of what an entire process actually do. Some results of the execution may be relevant for a party, while others may not be relevant. Partners have to agree on (relevant) results of some execution.

Partners have to agree on the so-called post-conditions of a task and of a process. In [13, p36] a post-condition is defined as ‘A logical expression which may be evaluated by a workflow engine to decide whether a process instance or activity within a process instance is completed.’ Stated differently, the post-conditions show what the task/process is supposed to realise.

Relationship To The Challenge

The requirement to align tasks within a process implies that someone who determines what one task should do has to know (to some extent) what the other tasks do. A process realizes post-conditions by executing a number of tasks. One can only correctly align the tasks if it is clear how the tasks are supposed to fit together for the realization of the intended outcome of some end-to-end process.

Relevance

There are several reasons why it is important to understand what a task does:

- As we stated earlier, tasks executed by different companies are typically linked through so-called ‘outcome interdependencies’, but in a collaborative setting they are linked through ‘behaviour-interdependencies’: their execution (i.e., behaviour) gets intertwined. To mix the tasks decently it is imperative to know what the tasks actually do, and what the entire process is expected to do.
- Secondly, if some task cannot be executed successfully by one party, another party may be brought in who can execute it. In that case, however, one has to map the task as it was to be executed by the original party to the task as the new party is planning to execute it. If both tasks are not realizing the same post-conditions they can probably not be exchanged.
- It may be the case that some ‘new’ task actually realizes what was to be done by several ‘old’ tasks (or the other way around: one old task is replaced by several new tasks) resulting not only in changes in who executes some task but also in the control flow.
- Some of the task execution results may be visible, while others may not. In case undesired effects ensue from the task execution it is important to know that the task execution actually entails more than just realizing what is agreed upon.

The fact that tasks are perceived differently creates obstacles to change tasks and/or processes. It is difficult to creatively use some task in some process design if the task is not really understood. Fortunately, for the execution of the process parties only need to understand their own part in the process.

Illustration

In the computer-sales process the Manufacturer’s request for a shipment results in an update of the Shipper’s planning and a transmission of the shipment date to the Manufacturer and the Reseller. This information may be important for the Reseller as an input for tasks such as planning the work force for installing computers at the customer’s site. In case a new Shipper would be needed, the shipment request might not result in a transmission of the shipment date to the Reseller. The Manufacturer would then get the additional task to forward the shipment date to the Reseller.

Hidden, undesired effects may ensue from some task execution. For example, the Shipper may forward data to competitors about which Reseller buys which computers from the Manufacturer. Competitors could then strive to do direct marketing.

Problem 3: It Has To Be Revealed How Task Executions Are To Be Triggered

Once it is known what tasks need to be executed in what sequence it has to be stipulated how one can make sure that some task will be executed.

Problem Description

Task requestors may not be aware of how they can make sure the task execution is initiated, or they may not be enabled by the executor to initiate the execution.

A task can be executed if (1) the system is in an appropriate state to execute the task, and (2) the inputs necessary for task execution are available.

- If a task execution completes (successfully or unsuccessfully) within some company this changes the state of the Extended Enterprise into a state where some tasks can, have to, or cannot be executed by some of the partners. Consequently it may be important that those partners are informed about this state change. Through message transmissions partners can communicate state changes so that they perceive the (relevant parts of the) state of the Extended Enterprise the same. A partner which notices or is notified of a state change may then start the execution of some activities.

- Inputs that are necessary for some task may already be available or not when the system enters a state where execution is desired. Inputs are – in the context of the processes we consider – data.

In an Extended Enterprise, initiating a task will thus typically require the transmission of information (see e.g. [17] about BPMN). This fact creates a direct link with [4]: it is acknowledged that there is some need to share information and several problems can arise when trying to share that information. Understanding and enabling business processes is critical to providing relevant, timely and accurate information to individuals [18].

The inputs that are necessary for a task and the state change that triggers the task are not necessarily communicated in a single message. Often it will be possible to transmit the inputs well before the state changes, so that the inputs are already available when the task executor gets another message that signals the state change. In other cases making an input available is supposed to signal a state change itself. As an example, let us consider the product engineering process. If the engineers have updated the engineering BOM (Bill of Materials) they may publish this BOM to a shared storage space. If the task of creating a manufacturing BOM should actually be triggered by the fact that a new engineering BOM was created, the manufacturing company should be subscribed with the shared storage space to be informed whenever a new engineering BOM has arrived, or the engineering company would have to send directly a message to the manufacturer that a new BOM was uploaded. That is, a message should be sent to signal the state change, while the inputs are available in the shared storage space.

The problem is thus that a party that is involved in initiating a task that is to be executed by another party has to know what inputs he has to make available and/or what state changes he has to mention to the task executor. The other way around, a party that is sending messages may not be aware of the fact that these messages actually are involved in the initiation of a task execution.

Relationship To The Challenge

The execution of tasks can only be controlled if it is known what state information and what inputs are needed to start the task execution.

Relevance

If a transmission from some party triggers the execution of a task by another party, but the transmitting party is not aware of this, he may not execute the transmission suitably. That is, his transmission may not be in line with the process requirements.

If the task executor does not enable the other party to initiate the execution, or this other party is not aware of how he could do so, this other party cannot control the execution of the task.

The importance of triggering task executions by informing a partner about some event that happened is clear if one considers the growing importance of SCEM (Supply Chain Event Management). Nowadays companies want to be informed fast about changes in the state of the Extended Enterprise in order to be able to quickly seize opportunities and to swiftly react to threats. Two basic aspects of SCEM are just monitoring and notification [1]. If one partner monitors the tasks he executes and immediately notifies his partners about relevant events, the partner can react to those events. It has, of course, to be stipulated what are ‘relevant events’. What is relevant to one company is not necessarily relevant to another company. That is, not every company can give a valuable response to each event. In an Extended Enterprise partners can monitor events that are of particular interest to each other (and not to others).

Of course, informing a partner about an event that has occurred can only trigger the execution of a task by that partner if that partner is already in an appropriate state for the rest. Maybe the partner still has some preconditions to fulfil himself before the task can be executed, and the partner will then be self-triggering the task execution.

Illustration

A new Shipper may not be aware of the fact that he triggers the ‘present invoice’ activity of the Factoring company when he sends a delivery confirmation to the Factoring company. Therefore, the way the Shipper deals with this message transmission may be inappropriate. He might for example only send confirmations once a month instead of several times a day.

In our example the shipment may be initiated on the basis of the manufacturing planning (i.e., the internal clock of the Shipper triggers the shipment task). It could be, however, that a new Shipper only uses that information to make a provisional shipment planning and that he waits to send out trucks until he gets the notification that the system is in the state ‘computer manufactured successfully and waiting for transportation’ (i.e., the Manufacturer triggers the execution). In this case the Manufacturer would have to inform the Shipper explicitly about this task completion event in order to initiate the shipment task. This requires the Manufacturer to enable his systems (be it computers or humans) to monitor the production and to transmit information on the completion event, and it requires the Shipper to enable his ‘systems’ to receive and process such information and to let such receipt trigger the shipment-activity (e.g., by sending a message to the GPS-system in the truck of the driver who is closest to the Manufacturer).

Problem 4: Different Parties May Have To Make Investments

To realize processes investments are needed at two levels: both tasks and process control need to be implemented.

Problem Description

Partners have to agree on the tasks they should be able to perform and on the service levels they may expect from each other (see later). These requirements all come at some price and it has to be determined who will pick up the check for these investments, and how the benefits will be distributed.

Also, to control the entire end-to-end process investments have to be made so that the necessary task executions can be triggered and followed up.

Relationship To The Challenge

Some tasks may not yet ‘exist’, or may not yet exist in an appropriate quality or form. Investments are thus needed to align the different tasks in the process so as to realize the foreseen end-to-end process with the desired service levels. Some partners may

have to make investments to make it possible for another ‘controlling’ party to trigger a task execution, or to make it possible for themselves to control another’s task execution.

Relevance

If the process only requires standard tasks to be executed the parties do not have to make process-specific investments. This implies that a party performing a task can easily be replaced by another party that performs the same standard task and that few investments are lost (or need to be made) if some executor is replaced by another one.

In the Extended Enterprise partner-specific investments may be made. If a party makes an investment in tailoring some task this investment may become obsolete or even inconvenient if the company has to or wants to leave the Extended Enterprise. Extended Enterprise specific investments create a lock-in.

Subramani [15] states that the benefits from B2Bi efforts (studied in a buyer-supplier context) often only go to one (strong) party, but that in case of relationship-specific investments that create a means of differentiating from other networks all parties are likely to retain some of the benefits.

Illustration

A small Shipper may not offer functionality to customers to trace their packages. The partners in the Extended Enterprise may request him to build such a tracing functionality by investing in RFID technology in its different warehouses. The Manufacturer may also have to invest in the RFID technology so as to put tags on the different products. All parties, the end-customer, the Reseller, the Seller, the Shipper himself, and the Factoring company may be interested to know where the computer is at some moment and to act upon this information.

The event ‘computer is delivered’ enables the Factoring company’s task to collect the money. In fact, this event has to trigger the execution of the task. If the Manufacturer is the process owner it may be up to him to trigger that task execution. He would then have to follow up the task of the Shipper (passively by waiting for a notification or actively by polling) and ask the Factoring company to collect the money. Alternatively it is possible that the Shipper himself has to take care of triggering the task execution of the Factoring company. That is, the Shipper directly has to inform the Factoring company that the delivery has happened. In that case, the Manufacturer does not have to make investments to trigger the Factoring company’s task execution. Yet another solution is that the Manufacturer sends the Factoring company a message with the planned delivery data when he receives an order and that it is agreed that the Factoring company can collect the money on the date stipulated in that message unless he gets an ‘inhibiting message’ from the Manufacturer or the Shipper. The execution of the Factoring company’s task is then not tied to the execution of the Shipper’s task but to the planning of that task, which results in other investments to be made.

Problem 5: The Parties Are Dependent Upon The Service Levels Provided By The Different Systems

The value of executing a specific task depends on the value that is realised by the entire process. In general, business processes can only be valuable if they are executed with appropriate service levels. The partners thus have to agree on what are the required service levels for the process, and for the individual tasks being part of it.

Problem Description

The service levels that should be and that can be provided by some task are increasingly dependent upon the service levels provided by other tasks. Problems with service levels of one task may, therefore, escalate.

Furthermore, the task triggering can only happen if the triggering and the executing systems are both operational.

Relationship To The Challenge

It is acknowledged that a process can only be brought to a favourable conclusion if all the tasks in the process are executed appropriately (and thus if all tasks can be initiated in time).

Relevance

As tasks become more interwoven there are more relationships between tasks. The value of a task that is executed before another task is dependent upon the service levels provided by the succeeding task. Even the most excellent task executions may become useless if other tasks are not executed decently (e.g., in time), so that compensating tasks may be needed.

Also, the other way around, a task that is executed after another task is dependent upon the service levels provided by the preceding task. End-to-end process service levels can only be achieved if individual components provide sufficient service levels. If a preceding task is executed fast a succeeding task has additional time to execute; if the preceding task is executed too slowly a succeeding task has to be executed faster, or may not even need to be executed at all. Clearly, in the real-time enterprise real-time coarse-grained services can only be delivered if the fine-grained services can be delivered in real-time.

In Market B2Bi (as defined in [5]) dependencies will be relatively low because one party can easily be replaced by another one. In Extended Enterprise integration (as defined in [14]) companies consciously become more dependent upon each other: only this specific party is to execute some part of the process, and if he fails to do so in time there is no possibility to take recourse to another service provider (e.g., in another instance of the process execution).

Illustration

If the systems of the Shipper go down, no shipment can be requested. Also, if those systems are down, the signal from the RFID tag that tells the company the computer is delivered may not get through, and the Factoring company may not be informed about the fact that the delivery has happened.

A small problem may escalate. As an example, let us consider the situation where the Shipper is only allowed to deliver after payment. If an employee of the Factoring company postpones entering some settlement in the system (i.e., the Factoring company fails to do its job with appropriate service levels), the Shipper cannot fulfill its task either. The fact that the customer doesn’t get his goods (or has to wait a prolonged time for the goods) even though he has prepaid the delivery – something he does not like to do anyway (!) – may cause him to switch suppliers.

Problem 6: Partners Must Preserve The Value Of The Functional Proposition Of The Process In Order To Preserve Trust

A process has some value because of its functional proposition (i.e., the functions it promises to execute). Companies are dependent upon each other for maintaining that functional proposition valuable.

Problem Description

According to Smith and Fingar [14, p4] “processes are the main intellectual property and competitive differentiator manifest in all business activity.” The value of a functional proposition is directly related to the uniqueness of the proposition. Therefore, partners should not bring the process design into the open.

Furthermore, a process may be valuable for it has some unique tasks in it. Partners trust the providers of these unique tasks not to offer this task execution to others (i.e., not to include this task execution in the functional proposition of a competing Extended Enterprise), as this would lower the value of their process’s functional proposition.

Partners that make process specific investments count on it that their task will be executed during the execution of some process. If their task is eventually, however, omitted from the functional proposition or the process is dropped all together this harms their believe that future investments will pay off.

The other way around, partners trust each other to be really able to execute the tasks they promise to execute. If an initiating party has sent the necessary initiating messages, it should be able to trust the other party that the task is truly executed. The party is trusted to realise the promised outcome, and therefore to try to retain specialized people that manually execute a part of the process for example. Furthermore, the party is trusted not to execute any other, hidden tasks that lower the value of the process (e.g., call the customer of the partner to ask for information which is not strictly necessary for the task at hand).

Relationship to the challenge

Partners should only do the effort to align the tasks of some process if they can trust their partners.

Relevance

This is very much a problem that is confined to the realm of Extended Enterprise integration. In Market B2Bi companies offer their services apart from a specific process in which they should be included. There is no need to preserve some process’s functional value as there is no specific process for which the service was developed.

In Extended Enterprise integration the services that are offered are expected to be part of a specified Extended Enterprise process. Harming the functional proposition of this process jeopardizes the value of the task the party implemented. This can be expected to harm trust, to reduce the willingness to make new investments, to reduce creative impulses, etc.

Illustration

If a Manufacturer produced a screen and a computer that were ordered, but the Shipper does not pack the screen decently and it gets damaged during transport, the Resellers’ orders are not fulfilled (although the Manufacturer did his job). Similarly, if the Factoring company uses wrong tax rates in its calculations, problems arise. That is, the value of the promise to deliver some functionality is damaged because one of the parties is actually not capable of executing the task it said it would execute.

Problem 7: Process Ownership May Not Be Well Arranged

If the process is to be managed end-to-end a process owner has to be designated.

Problem Description

There may be several parties or (more probably) no party designated as end-to-end process owner.

The process owner has an end-to-end accountability for the process [7]. He is responsible for reengineering the process [8]. He decides what partners to involve in the business redesign and how far the process integration should go [2]. He must ensure that the process is understood by the people who execute it, and that they possess the required tools for execution and that they follow the specified design [9]. If there is no clarity about who is the process owner no party can be pointed out to take these tasks on him.

Relationship To The Challenge

Who has the responsibility to determine what is an appropriate execution of the process, and to make sure the process is executed appropriately? Who decides who can execute a subprocess (or task) and who can initiate that execution, who has to monitor the process, etcetera? Who decides how different tasks are brought together to form an end-to-end process?

Relevance

In an Extended Enterprise processes are supposed to be more actively managed, redesigned and optimized than in Market B2Bi. This naturally requires clear process ownership. If there is no clear owner, it is not clear who is accountable for the process or where to go with suggestions with respect to process improvements. Also, parties may not be well-informed about their role in the process and they may feel like there is no party that is appointed to rap them on the knuckles if they perform badly.

In [12, p1] it is stated that several research areas (such as Supply Chain Management) assume that the responsibility for process management and process design rests in the hands of one powerful party, but that in many inter-organizational settings there is often no (explicit or implicit) agreement of process ownership. Still, unambiguousness in process ownership seems to be one of the critical success factors of combining IT support and business process redesign [6].

It is not only considered to be important to know who is the process owner, but also to know who the others think the process owner is. The customers may perceive the party that is closest to them as the party that is responsible for the process, while this party actually does not own the process.

Illustration

If the goods arrive too late at the Reseller’s store, how will this be dealt with? Should the Reseller suggest to replace the Shipper or is this the task of the Manufacturer? Or should the Shipper not be replaced, but the process be redesigned? Who is allowed to

make changes to the process? Can the Factoring company require the Reseller to pay before delivery, or can this only be decided through the agency of the Manufacturer?

While the Manufacturer (the big company in the value web) is likely to be the process owner of the process in Figure 1, the end-customers may assume the Reseller is responsible for the process: he is responsible for choosing his partners. The process owner can then ask the assistance of the Reseller in redesigning the process. If the Manufacturer wants to know what requirements the end-customers have with respect to the process, he will probably have to talk with the Resellers about this. For a similar real-life example of a problem with process ownership (at AT&T), see [11].

Problem 8: The Involved Parties May Change Over Time

Problem Description

The partners may change over time so that the parties that are designated to provide some service differ, and/or the parties that may request the execution of some service differ.

Former task initiators should then no longer be able to initiate tasks, nor should they get feedback about task executions. Former task executors should no longer be requested to execute the task, nor should their output be considered. New task executors and/or initiators should be brought into the process (resulting in a re-appearance of the problems mentioned in the other sections of this paper).

Relationship To The Challenge

The parties that control the process may 'disappear' and new parties may enter the network that need to control the process. New parties can be added that execute tasks which need to be controlled too. Their tasks have to be brought in line with the rest of the process (or the rest of the process has to be aligned with their tasks).

Relevance

Flexibility is nowadays often mentioned as an important requirement upon companies. One way to achieve flexibility is to make it easy to remove and add parties in the process so that companies can swiftly deal with problems and take advantage of opportunities that show up. This is something that is not likely to happen often in Extended Enterprises. Still, this does not mean that partners never will be dropped or added.

Illustration

The party that ships the products may change. The Manufacturer's system then has to be informed about who is allowed to do the shipping now. It also needs to know how the new Shipper works: is he continuously re-planning the transportation schedule or only making a planning once a day. On the one hand, he can use this knowledge to determine whether it makes sense to ask for new shipments (and to expect a reply with 'expected delivery time') from the moment the computer production is finished, or whether it only makes sense to do so once a day. On the other hand, the new Shipper may be required to change the way he executes the task (e.g., to set up a continuous re-planning system by having the re-planning task triggered by the Manufacturer's message) so as to align his task with the remainder of the process.

The Reseller may need to agree that this new Shipper will do the shipping. After all, the customer may perceive the Reseller as the 'responsible party' for the entire process, and the Reseller then has to make sure that the Shipper can provide the desired service levels at reasonable prices.

For the part of the new Shipper: he has to find out whether he is expected to transport the goods as fast as possible when the computer is produced, or as fast as possible when an entire pallet of computers is ready, or when this could happen at the lowest cost, or only when he got a message that the payment of the customer was successful, etc.

CONCLUSIONS

In this paper, we have presented eight problems that show up when companies pursue inter-organizational process integration. (1) If we look at the building blocks of the process, the tasks, we notice that these blocks have to be arranged: the tasks should be executed in some order. The different partners may, however, have different requirements with respect to the order. (2) While trying to find an appropriate arrangement of the building blocks, the companies may note that they have a different understanding of what the blocks at hand actually are. That is, the outcome they expect from some task execution may be different. Moreover, the outcome they expect from the entire process execution may be different too. (3) Once it is known what tasks need to be executed and in what order they will be executed, it has to be decided how the task executions can be triggered. (4) To realize the processes investments are needed. Partners have to agree on who will bear the costs and who will do the effort. (5) The partners have to agree on what are the required service levels for the process, and for the individual tasks being part of it. At runtime, the service levels that are needed from one task depend on the service levels that are actually provided by other tasks. (6) On itself, an individual task is often worthless. It is in combination with a number of other tasks that the function provided by one task becomes useful. The value of executing a single task depends on the value of the functionality of the entire Extended Enterprise process. In an Extended Enterprise partners should preserve the value of the functional proposition of the process in order to preserve trust. (7) Inter-organizational process ownership has to be arranged. (8) The partners may change over time so that the parties that are designated to provide some service differ, and/or the parties that may want the execution of some service differ. Importantly, these eight problems show up in a different way in different process configurations. Knowledge about the way process configurations suffer from these problems can be used as a basis for assessing which inter-organizational integration solution is best in some specific situation. Indeed, many different inter-organizational integration solutions exist. In [3], we have presented the Business-to-Business integration solution space, which comprises several (de)centralization dimensions. When considering process integration, it is for example important to assess whether there is one central party that is controlling the process execution or whether the inter-organizational process execution control is more or less decentralized.

REFERENCES

- [1] Bittner M. (2001) "Intelligent Response: The Next Wave of Supply Chain Event Management (SCEM) Capability", <http://www.amrresearch.com>.
- [2] Champy J. (2002) *X-engineering the corporation – reinventing your business in the digital age*, Warner Business Books.
- [3] Goethals F. (2006) "Classifying and Assessing Extended Enterprise Integration Approaches" PhD. Dissertation at the Faculty of Economics and Business, K.U.Leuven, Belgium, pp. 235.
- [4] Goethals F.G., Snoeck M., Lemahieu W. and Vandenbulcke J. (2007) "Inter-organizational data integration problems" Proceedings of the International Conference on Business and Information, Tokyo Japan, July 11-13.
- [5] Goethals F., Vandenbulcke J., Lemahieu W., Snoeck M. and Cumps B. (2005) "Two Basic Types of Business-to-Business integration", *International Journal of E-Business Research*, Vol. 1, No. 1, pp. 1-15, <http://www.idea-group.com/downloads/samples/IJEER.pdf>.
- [6] Hammer M. (1996) *Beyond Reengineering*, Harper Collins Publishers, N.Y.
- [7] Hammer M. (2001) *The Agenda – What every business must do to dominate the decade*, Random House Business Books.
- [8] Hammer M., Stanton S.A. (1995) *The Reengineering Revolution*, Harper Business.
- [9] Hammer M. (2002) "Process Management and the Future of Six Sigma", *MIT Sloan Management Review*, Winter, pp. 26-32.
- [10] Keen M. et al, (2006) Patterns: *Extended Enterprise SOA and Web services*, IBM, pp. 542.
- [11] "Wharton. Special section: The rush to send back-office business overseas – Part 2", <http://knowledge.wharton.upenn.edu>, April 25, 2006.
- [12] Larsen M.H. and Klischewski R. (2004) "Process Ownership Challenges in IT-Enabled Transformation of Interorganizational Business Processes".
- [13] Paterson T.T. (1959) *Management Theory*, London Business published, pp. 212.
- [14] Workflow Management Coalition (1999) Workflow Management Coalition Terminology & Glossary, pp 65, <http://www.wfmc.org/> on April 26, 2006.
- [15] Smith H., Fingar P. (2003) *Business Process Management- the third wave*, Meghan-Kiffer Press, pp 292
- [16] Subramani M. (2004) "How do suppliers benefit from information technology use in supply chain relationships?", *MIS Quarterly*, Vol. 28, No. 1, pp. 45-73.
- [17] Thompson J.D. (1967) *Organizations in Action: Social Science Bases of Administrative Theory*, New York: McGraw-Hill.
- [18] White S.A. (2004) "Introduction to BPMN BP Trends", <http://www.bptrends.com> on.
- [19] Wu J. (2004) "Empowering the information enterprise: business processes are the foundation of an information enterprise", *DMReview*, June 2004.