Winter 12-5-2005

Setting up Extended Enterprises: A Data Aspects Framework

Frank Goethal

Jacques Vandenbulcke

Wilfried Lemahieu

Monique Snoeck

Setting up Extended Enterprises: A Data Aspects Framework

Frank Goethals, Jacques Vandenbulcke, Wilfried Lemahieu, Monique Snoeck
F.E.T.E.W. – K.U.Leuven – Naamsestraat 69, B-3000 Leuven, Belgium
{Frank.Goethals, Jacques.Vandenbulcke, Wilfried.Lemahieu, Monique.Snoeck}@econ.kuleuven.ac.be
SAP-leerstoel Extended Enterprise Infrastructures

Abstract: Nowadays, companies want to share information. When doing so, many issues have to be taken care of, and many options are available for most of these issues. Realizing B2Bi is a very complex task. It is the aim of this paper to make it possible to oversee the complexity of information sharing in a B2B context by structuring the issues that have to be taken care of in a new framework: the DA (Data Aspects) – framework; and by relating this framework to the existing FADEE framework.

Keywords: Data Warehousing and Data Mining

I. Intro: Structuring the Problem Space

Rather than *owning* information, *sharing* information is nowadays often said to be an important source of competitive advantages. Sharing data is, however, not an easy task. Many different issues have to be dealt with, and the many decisions that are taken should fit together, and should fit the requirements of the business.

This paper reports on our research-in-progress (updates will be posted at www.frankgoethals.tk – publications). The research is meant to overcome one of the frustrations one may get by reading through literature on B2B (Business-to-Business) data exchanges: such papers are typically very *fragmentary* without positioning the niche they cover within the entire field of data-exchange related issues. Moreover, while the discussions are often fragmentary, they are not '*normalized*'. With this we mean that concepts are used which seem to possess a number of properties in the mind of the authors, but which are not defined explicitly. These concepts can be looked at from different points of view, and by not presenting those viewpoints, or by highlighting only one, the concept remains obscure for the reader. For example, terms like an 'extended enterprise wide datawarehouse' are very obscure. Does this mean all the data is stored geographically in a location that is central and which can be accessed by all parties? Or does every party have a replication of all data? Or is the warehouse actually only a database with meta-data through which the scattered databases are connected? Can one request data from the datawarehouse, or should one subscribe at the warehouse to get the information the datawarehouse publishes? Is the 'central database' (whatever that means) located on the premises of one of the partners, or on a third party's webfarm? Is one of the parties the 'owner' of the database, or are all parties owner of a part of the database, or… Many questions and answers are thus hidden under such a title.

When looking at the data side of the B2Bi story alone, many issues can be identified that should be taken care of. In this paper we try to bring all these issues together in a simple framework.

In order to identify what are the data-related issues that should be dealt with in B2Bi in general, investigating a single in-depth case study was considered inappropriate. After all, choices and options become better visible if one case is contrasted with another one. Therefore, we decided to draw up the framework on the basis of a study of many cases that were published (also by others), as well as on papers that do not present specific cases but relevant constructs (e-business standards and the like). After some time of working bottom-up (i.e., starting from practices and trying to put them in a structure), we noticed the topics fitted in the often discussed classification of Zachman [17]. Zachman states that different (normalized) views are possible on some entity. Following Zachman, a discussion of an entity has to include the following six questions in order to be complete: what, how, where, who, when and why[1]. Consequently, if one wants to find out how to deal with data, it is actually logical to stumble across these six questions. In what follows we handle the six questions as six boxes in which we can put data-related issues. As such, the six boxes function as a simple placeholder, a framework. We call this the DA (Data Aspects) – framework.

Figure 1: The DA (Data Aspects) framework

Knowing the six questions, we were able not only to work bottom-up, but also top-down. That is, while topics identified in literature nicely fitted with the six questions, the six questions helped us identifying issues that were not visible in literature. We could formulate questions such as 'who wants the data?', 'where is the data?', 'how will the data be moved from point A to point B?', etc.[2]

By working in both directions we believe we have covered a good part of the ground. Above that, the

Proceedings of the Fifth International Conference on Electronic Business, Hong Kong, December 5-9, 2005, pp. 109 - 116.

[1] While in his ISA framework the entity under consideration is typically an entire enterprise, we consider 'data' to be the entity.

[2] We note that one may also apply the six questions to the meta-level. For example, in stead of discussing 'who needs what data' and 'how he will get the data', one may investigate 'who should define who needs what data' and 'how he should define that'. For now, however, we do not look at this meta-level.

framework is well-structured so that at any moment new topics (if any would have been overlooked) could be added without repercussions for the existing framework (i.e., the six boxes will stand the test of time).

In the following section we present the data-related issues in the framework. While space limitations make it impossible to deal with the issues in detail, we discuss some options that are available in each box. After that, we bring a short case study from the health care industry, covering a number of different aspects.

The DA-framework enables people to easily pinpoint the data-related topics that have to be (or are being) dealt with in B2Bi. While the framework is a nice contribution because of this enablement, it does not show how to actually *deal* with the issues. Classifying the issues might create the impression that setting up B2B data exchanges is simple. Indeed, the classification gives people a clear view of what B2Bi entails. Still, dealing with the issues (and their interdependencies) is a complex task in which many people are involved. This is shown in the final section of the paper where we relate the issues in the boxes to the FADEE-framework (the Framework for the Architectural Development of the Extended Enterprise, see e.g. [8]).

II. The Design Variables to Play With

In what follows, we discuss the different issues that should be dealt with. The issues are grouped in six subsections showing the HOW, WHERE, WHO, WHAT, WHEN and WHY aspects respectively of the data sharing.

II. 1 How

One has a large choice with respect to the medium that is to be used to exchange the data. Knowledge may for example be transmitted visually (e.g., with pictures and models), orally (e.g., telephone conversation), or written (e.g., a facsimile, a letter, e-mail), or by a combination of these (e.g., through video-conferencing). A further discussion of all the options is not the goal of this research. In practice, however, choosing the medium is very important because a decision with respect to the medium has stringent consequences for options in the other boxes in the framework. Indeed, for choosing a medium, one will have to keep the other questions (mentioned in the following sections) in consideration: where the data is stored, who is involved, what kind of data it is, etc.

II. 2 Where

With respect to the where dimension two questions have to be assessed: where is the data located?, and what is the structure of the network?.

Network structure

Two nodes will always exist when information has to be shared: the location(s) of the data before transmission, and the location(s) of the data after transmission. Beside these nodes, other (intermediate) nodes may also be involved; for example because they will store the information that is transmitted (temporarily or permanently).

One has to investigate which relevant nodes do exist, and whether new nodes should be added, and where they should be added. Each node may be located on the premises of the parties, or at some other location (e.g., a webfarm, external to the business relationship). The location of the nodes, of course, has consequences for the communication medium (see HOW) that can be used. We note that, in general, the communication medium also has some topology. This topology is, however, transparent. For example, when using the telephone or the Internet many switches are included in the network. For realizing data transmissions the presence of these switches is transparent. They can be seen as infrastructure.

Data location

Companies that want to share data may agree to create a replication of the data. This replication may be located outside their company or not. The replication-dimension should be seen as a continuum, from ‘no replication’, over ‘thin replication’ to ‘fat replication’. Besides the data itself, data about the data can be stored too: meta-data about where the data can be found.

Data replication should be considered for all the nodes in the network. If an intermediate node is used, the node may store copies of (some of) the data. The node may then be a thin or a fat centralized datawarehouse. In case the node does not possess replications of the data itself, it can use meta-data to function as an information broker. The information broker can then act like a buffer for changes in relationships between companies. In case the counterparty in some process changes, the sending/requesting system does not need to be aware of this change: the information broker keeps track of such changes, and makes sure valid data (i.e., data from the right party) is transmitted to the right party. This is done on the basis of the meta-data the broker can investigate.

II. 3 Who

The ‘who question’ pertains to the parties that are involved in the information exchange, and the roles they may play. Again, several issues come to the front: Who is involved? Who initiates the information exchange? and Who is authorized to ask/send/get data?

Needy/Sender/Intermediary

In every data exchange, at least two parties are involved: a party that wants to get the data (who we call ‘the needy’), and a party that has to transmit the data (called ‘the sender’). Besides these two parties, intermediaries may be involved in the information exchange. An intermediary may be owned by the organization of the needy, by the organization of the sender, by a third party, and by any combination of these. With ‘ownership’ we refer to the party who can make the

decision[3] as to how the intermediary should take part in the information exchange. By including an intermediary in the exchange, a many-to-many exchange infrastructure can be set up, rather than a point-to-point infrastructure.

Push/Pull

A party can get data via a pull model or a push model.

- The pull model implies that the data is requested by one party, and the other party replies (Request/Reply).
- In the push model, the data is transmitted by the data sender automatically because the sender knows the other party is interested in the data (Publish/Subscribe). This model fits an 'event-driven' way of working (as is advocated in business practices such as Supply Chain Event Management, but also in systems development methodologies such as MERODE).

Irrespective of which model is used, we refer to the party that wants to get the data as 'the needy', and to the party that has to transmit the data as 'the sender'.

We note that if we assume that subscription in the push model cannot only be done by the information needy (but also by the information sender), the push model also allows for sending orders for example (as if the supplier of the product is subscribed for getting the order).

As stated, besides the primary needy and sender also intermediaries may be involved. The interaction between needy and intermediary, and between intermediary and sender may happen through different models. The needy may use the push or the pull model in his interaction with the intermediary, and irrespective of this choice, the intermediary may use the push or the pull model in order to get the required data in his system.

If we now consider the Where-question again, one can see that in case of the Request/Reply model, the needy can choose to make the data he received persistent or not. Irrespective of whether he did so, meta-data can be useful to find out where to request the (updates of, and additional, detailed) information. In case of the Publish/Subscribe model, meta-data is only useful for subscription purposes. Once subscribed, the meta-data is no longer useful for the subscriber, as any information that comes available will be transmitted to him. Not storing the information means the data is lost for the receiver (if the parties stick to the push model).

Choosing to replicate the data at some other location should not be done without thinking about the intermediary's ownership that is created this way. A party may actually loose the ownership over the data if a replication is stored at a location outside the company walls. This may particularly be the case if the intermediary where the replication is stored is owned by another party.

Authorization and authentication

A needy may not accept data coming from just any sender. Authorization and authentication[4] can be needed here. Authorization and authentication are, however, also going to play a big part when turning the picture upside-down: By looking at the needy from the point of view of the sender, it is clear the sender does not want to give his data to just any party. A needy needs to be authorized data access.
Every message exchange between parties can be accepted or rejected. In the Publish-Subscribe model, if one party subscribes (be it the needy or the sender), this subscription has to be accepted by the other party. Given the fact that a subscription is considered valid by both parties, publishing data is authorized as well (authentication may still be needed). In the Request/Reply model, the sender has to assess (after authentication) whether the needy is authorized to get the information. Given the fact that the needy-system has asked the sender-system for the information, the sender is authorized by the needy to send a reply message (authentication may still be needed).

It should be noticed that giving deliberate access to one party may result in giving implicit access to many parties. This may for example be the case if the systems of the counterparty are not well secured. The idea arises that a network is only as secure as its weakest link. From the case study research done by Dynes et al. [3] at five partnering companies, it seems that managers believe that their firm's internal networks "are not at additional risk as a result of using the information infrastructure to integrate their supply chains" [3, p2]. It seemed that the firms in the study did not put big security requirements on their suppliers, although one of the firms posed it would start having requirements in the near future. This company actually believed that having a high level of information security was not merely a 'qualifier' (to be allowed to play the game), but a competitive advantage.

Still, in [4] (where Dynes presents the findings of a roundtable with a number of business leaders and academics) it is stated that in general companies are auditing the information security status of potential partners. One difficulty with such assessments is that they slow down the partnering. Besides that, it should be noticed that security is not just something that the IT department has to take care of. Employees in all parts of the organizations need to be educated (in relationship to their roles) about their responsibilities with respect to security.

Besides the unwillingly sharing of data by the new information possessor, there may also be a problem of the needy willingly sharing the data with other parties. 'Privacy' is a topic that is often mentioned in the context of partnering companies. In general, information privacy concerns the fact that individuals require that information about themselves should generally not be available to others, and that, where

[3] Please note that the term 'makes the decision' is narrower than 'control'. One party may influence another party in making decisions, and another party may influence the execution of the decisions. Control is therefore often shared by many organizations. This, however, depends on the decision making process that is used, and thus varies from case to case.

[4] Authentication has to do with determining who a user or a system is, whereas authorization is about stipulating who is allowed to access which resources [DP15].

data is possessed by another party, the individual should be able to control the data and its use to a considerable extent [2]. If a customer gives personal information to a specific company, this does not mean the customer would agree to give this information to the partners of this company. Moreover, the partners should handle the data with care, fully respecting the agreement the source organization has with its customer (although this partner may not have a direct relationship with the customer). Interestingly, privacy regulations seem to differ widely across countries [4].

As a side note, we notice that data access management requires the storage of data for authorization and authentication purposes, and that – just like for all other data – many questions have to be answered about the storage and the transmission of the access information (e.g., the information can be stored centrally or not, can be replicated or not, can be accessed directly or not, etc., see below). Based on access information, needy and senders can be further divided into categories, based on their specific permissions to create, read, update, or delete the data (see e.g., [5, and 10] for a discussion on this).

II. 4 What

When exchanging data with another party, one has to make choices on a number of properties of the data itself. These properties, and the associated different options, are discussed below. First, however, we shortly have to mention a property that all data exchanges are likely to have: the data exchanged is a so-called 'Boundary Object'; an object that links two fields which are divided by a boundary.

Boundary objects

Levina and Vaast [11] identified two basic requirements for an object to be a boundary object. First, the artefact has to acquire a local usefulness. That is, agents in each field must use and make sense of the artefact in the context of their field. Secondly, the artefact needs a common identity. To make this possible, a joint field (which serves to bridge the separate fields) must be established within which agents jointly recognize and value the artefact.

Levina and Vaast found that organizations rely on boundary spanners to establish the local usefulness and a common identity of boundary objects. The boundary spanners (1) reflect on objects from each field and on their utility within the context of the joint field, and (2) they create new artefacts and try to establish their new identity within the joint field. Then (3) they try to establish the local usefulness of these artefacts.

What (type of) information is being shared

Companies that pursue B2Bi can do this in various practices. On the one extreme, standard practices may be automated. The transmission of purchase orders may for example be automated (with no people being involved in the B2B practice). At the other extreme, the ICT systems may only be used as the start for a close collaboration: to find people in a partnering company who can help on some practice. The information companies want to share in the latter case is typically tacit in nature, and is thus not present in computer systems. Therefore, they share other data to find out who has the necessary tacit knowledge.

In order to align the ICT with the business, one first needs to decide what *type* of data fits the *type* of the business. Only then it can be decided what data should appear in the system. This is also true in a B2B scenario. Companies have a relationship for some business reason. If this business reason is that the supplier sells some half-finished product at the cheapest rate, the data exchange will be limited to placing orders automatically. If the companies collaborate because their production processes are similar, and one can help the other out on improving its production process, the data exchange will concern the experiences, background, education, etc. of the personnel. Aligning the *type* of data with the *type* of business the companies are in, is what we call 'high-level alignment'. Unfortunately there is no space to deal with this issue in detail. One may refer to Hansen, Nohria and Tierney [9], and to Birkinshaw and Sheehan [1] for studies within the frame of isolated enterprises (thus not B2Bi).

Batches or singles

Data that is needed by a user may be requested and transmitted as an exchange on it own. Alternatively, requests and/or replies may be grouped in batches.

Annotation

If the data is meant to be made persistent in another system, the job of storing the data usefully will be greatly facilitated if the data is annotated. That can be done by sending the data in an XML format that follows some (standardized) XML schema. To this purpose, the sender can try to push forward the degree of structuration, so as to transform unstructured data into semi-structured data.

The other way around, if the sender would not like the user to store the information, he can make things more complex to the user by not annotating the data, and sending highly unstructured documents. In this case, data that is highly structured at the sender's site (e.g., prizes of products) may be transmitted in unstructured documents (e.g., highly graphical brochures).

As an example, a purchaser who is interested to buy products from a supplier may want to look into the supplier's catalogue. The supplier does not want the purchaser to load the entire catalogue into his system (e.g., because the catalogue gets out-dated fast; because this would make it easy for the customer to compare prizes with competitors more easily, etc.). Still, it is important for the purchaser that when he selects the products in the catalogue, that the data on the order is made persistent in his ERP system. A solution to this problem was defined by SAP and Ariba. The Open Catalog Interface (OCI) or 'punch-out' solution offers a purchaser the possibility to see the catalogue via the web. The catalogue does not enter the ERP system of the purchaser. The purchaser can select the

products he wants to order in the web interface, and he can have a standard message sent to his ERP system so that the data on the desired products enters his own ERP system.

Standardization

The data may be represented in a standard or a proprietary format. For two companies to exchange data they need some agreement. This agreement may be established by a third party in neutral standards, or may be set up by the partners themselves. Of course, by choosing for a proprietary format, chances for including other parties in the data exchange later on are reduced.

We note that during the last years many standards have arisen on messages that may be exchanged between companies. While some of these standards can be used in any industry (i.e., it are 'horizontal standards'), others or tailored to a specific industry (so called 'vertical standards'). To a big extent, these standards try to bring a standard structure in the messages that are exchanged. This is pretty straightforward when dealing with common business messages such as orders and invoices (see cXML, UBL, and CBL). For files, such as patient records in the social sector, contracts in the context of legal arrangements, or curriculum vitae in the human resources, it also seems possible to find a structure. These unstructured documents can then be translated into semi-structured documents (see for example the HL7 standard, the Legal XML standard, and the HR-XML standard respectively). We note that this fact can give rise to new intermediaries in the data exchange process. For example, Oracle's iRecruitment module can be connected (through HR-XML) with providers of resume-parsing functionality or background checking functionality. If an applicant submits his resume (be it in Word, PDF, HTML, or some other format), the employer can forward this document to a 'resume parsing company' as an attachment to a HR-XML message. The resume parsing partner then unleashes his algorithms to get elements (such as address, professional experience, skills, et.) out of the unstructured document, and puts these elements in a structured HR-XML document. Information formatted in the HR-XML format may then be transmitted to a background checking partner for checking the correctness of the information in the resume (e.g., about the education). After that, the data can finally be sent to a company who is interested to hire someone with specific characteristics.

Existing or new data

The data that is being exchanged may be data that is already available at the site of the sender. However, sometimes the needy may be interested in new data. That is, he can demand/request the counterparty to create new data (e.g., new indicators for monitoring some process). Moreover, cooperating companies may start to manage data that only exists at the level of the collection of the companies, and not at the level of the individual companies. For example, if an airline company, a car rental company and a hotel chain together offer trips, each of them generally only has information on its own sales. By putting the information together, data is available on how many customers booked an airplane seat as well as a hotel and a car. This data may then be linked to data on (individual/grouped) marketing campaigns for example to do data mining.

A side note on Web services

We note one can distinguish between 'parameter-based' Web services and 'message-centric' Web services. The first type of web services is *usually* linked with synchronous communication, realizing RPC (Remote Procedure Call)-style web services. The second type of web services is *often* related to 'asynchronous' communication [12]. The RPC-style web services may work well for some cases, but for other types of applications (e.g., when human interaction is involved at the partner side) they are less appropriate. Patil [13] states that using document-style web services has some benefits, as there are: (1) they facilitate the exchange of self-describing documents that have a business context instead of data structures that reflect application interfaces (i.e., they show the way businesses really interact with one another), and (2) they provide better insulation from changes to the underlying service because changing a few fields in a document does not break the contract between the two parties, as changing the application interface would. Consequently, the message-centric model can be used to realise a more loosely coupled integration [12].

II. 5 When

Synchronous/Asynchronous - Real-Time/Postponed

In case of the Request/Reply model, the needy may have to defer his activities until he gets a response from the information sender. This is the *synchronous* model of communication. In the *asynchronous* model, the information needy does not halt its execution. That is, in the synchronous model the sender is assumed to reply immediately, while in the asynchronous model he is not.

If the synchronous mode is desired, requests have to be dealt with in *real-time* (that is, they have to be transmitted in real-time, and be processed at the sender side in real-time). In case the asynchronous mode is chosen, requests can be dealt with in real-time or can be postponed. An intermediary may for example postpone the transmission because he wants to group all requests in a batch (see What?) before sending them to the sender. Therefore, replies may concern a single request, or a batch of requests. The real-time/postpone distinction can also be made in the Publish/Subscribe model: the data can be transmitted immediately as it comes available (an event is fired), or can be transmitted at regular time intervals irrespective of when the data came available.

II. 6 Why

In practice, the options discussed so far should be put together. Many different combinations are possible. However, not all combinations are appropriate for a given

situation. While the answers on the five questions presented so far should be chosen with attention for their interdependences, the sixth question is directional. That is, to a big extent the why-question gives direction on which options to choose in the other boxes of the framework[5].
The why-question pertains to (1) why data is exchanged (which will not be dealt with here), and (2) why data is exchanged the way it is done.

As per the question why data exchanges happen in a specific way, a number of issues should be considered. Data is needed as the input for some task. Following Thompson [16], one would call this 'serial interdependency'. Each party who executes a task may thus have requirements on the way the exchanges should happen. Thompson, however, also mentions another type of dependency: 'reciprocal interdependency'. This refers to the case where A is dependent upon B for giving him some data, but B is also dependent upon A for giving him some data.
This results in the following viewpoints:

- The needy needs the data to execute some task. Therefore, he may have requirements pertaining to the availability of the data (see e.g. [3] for a case study on this), response time of the system, consistency and being up-to-date, ease of access vs. privacy and security, reliability, etc.
- The sender not only has to transmit the data, but also may have to create, read, update, and delete (CRUD) the data. Therefore, the sender may also have a number of requirements related to availability, reachability, ownership, etc.
- We note that an intermediary can function as a needy and as a sender. Therefore, he may have similar requirements. The requirements depend on the tasks the intermediary has to do. For example, in the health care industry one may want to run algorithms to find out about the state of the entire population in a country. This algorithm needs data that is actually dispersed over many health care institutions. If running such an algorithm would happen to be the key reason for getting into B2Bi, replication of the data to a central place may be desired.
- The reciprocal relationship is important because it may tie messages together. For example, if a customer asks A for the execution of some service, A may need the help of B. Imagine that A first sends information to B, B then requests additional information from A, A transmits this extra information, and B replies with a final answer. If the customer wants a response in real-time from A, this has a consequence for the reciprocal relationship between A and B. The messages are tied together in a single process. A will require a real-time answer from B, and *therefore* B will require a real-time answer from A on his request for additional information.

Besides tying messages together, the fact that A is dependent upon B, and B is dependent upon A may have a consequence on the way companies deal with the interaction. No party has individual power over the other.

III. A Short Illustration

We will conclude this discussion with a short example from the health care industry (see [14, 15] amongst others). In the health care industry the idea has arisen to share information on patients among authorized institutions if a patient enters one of these organizations for help. Knowing the medical background of the patient can be very important. Because so many different institutions may have information on the patient, an institution needing information would need to contact all other institutions. One can hardly call this an efficient search. Therefore, in the Netherlands, a central point has been entered in the network where a needy can request for information. The central point itself does not have the data on the patients history. However, it has information on where information on some patient can be found. It got this information because it is subscribed with the different institutions who publish this information to the intermediary. In a similar set-up, the English central point *does* contain some of the information on the patients. That is, there is a thin replication in the English system, while there is no replication in the Dutch system.

As the Dutch central point does not possess the patient information itself, one might think the central point would answer the request by replying with the data about which institutions do have data on the patient. This is, however, not the case: the central point contacts these institutions (i.e., the single request from the requestor results in many requests from the intermediary) and groups the replies of the different institutions before transmitting the data to the needy.

Message exchanges between the different parties happen in the format described in the HL7-standard. Using a standard is important because it should be possible to flexibly change the parties that are involved in the whole set-up.

IV. Dealing with the Six Questions

The topics identified so far have been placed into six boxes. While these boxes give a nice categorization of the issues that have to be dealt with, actually doing so is not simple. This is because

(1) many of the questions relate to both, the business and ICT side;

(2) the answers to the questions should not only be

[5] While decisions in the other boxes may have influence on each other, decisions in other boxes do not have consequences for the why-question. The strategy drives the choices that are made. The strategy can be driven a-priori by the many options that are available, but is not driven a-posteriori by the options that were chosen. Of course, strategies are dependent upon decisions made in the past, but a strategy that is adapted that way only takes effect in the next cycle (at time t+1) of making decisions on the other questions.

appropriate for a specific project, but should fit with the longer term strategy for these questions; and

(3) one company cannot answer the questions in isolation: answers that are formulated within an individual enterprise should fit with the way the Extended Enterprise (i.e., the collection of partnering companies) is conceived.

These three dimensions can be found in the FADEE framework we presented in [6, 7, 8] (see Figure 2).

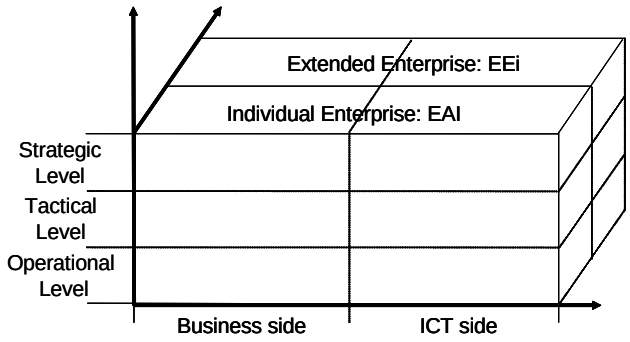

Figure 2: The three dimensions of the FADEE framework

At this point we do not want to deal in detail with the question how to bring the data-related questions into the FADEE. Rather, we give an example of the relevance of each of the dimensions.

(1) The concept of boundary objects is present at the business and at the ICT side. UBL, the Universal Business Language, for example is a standard for realizing B2Bi. As such, UBL defines seven business documents (such as 'order' and 'invoice'), and gives accompanying XML-schema definitions. While the UBL schemas are meant to be exchanged between computer systems of different companies, the documents they represent should function as a boundary object between business people as well. Therefore, specifications have been developed for automatically rendering a classic visual of the content of the XML documents, for example as a .pdf document, meant for human usage.

(2) The fact that companies should have a strategy with respect to these questions can be shown with important issues such as replication. If the strategy of the Extended Enterprise is to leave data ownership in the hands of the original owners, data replication should be limited. In the case of the health care industry in the Netherlands for example it was stated that duplicating some of the information in the central hub could be good for performance reasons. However, duplicating the information would not have fitted the strategy, which states that the individual organizations who create the information should remain the owners of this information (also making it possible to hide information at some future point in time for example).

(3) The Extended Enterprise and the companies that are part of it should be aligned. If the Extended Enterprise believes that it should be possible for other companies to easily enter the network, an individual enterprise should not try to forge its proprietary data formats upon the other parties. Standard data formats are then the way to go.

In fact, while the concepts presented above may seem very simple at the start they become more complex when looking at them in detail. For example, what is a needy and what is a sender? Is this an entire organization? A person? A department? An ICT system? In fact, the concepts needy and sender are not fully defined until one reveals about which cell of the FADEE framework he is talking. At the strategic level, one deals with entire organizations. At the ICT side one talks about systems, not people. Etcetera. Another example would be the concept of reciprocal dependencies. While there may be a reciprocal dependency between two companies, there may be no obvious reciprocal dependency between individuals of these companies (e.g., because the needy and senders from each company belong to different departments).

V. Conclusions

In this paper, we looked at the Extended Enterprise from a data-point-of-view. First, we organized data-related issues in the simple DA (Data Aspects)-framework. While space limitations made it impossible to deal with the six cells in detail, the paper gives a nice overview of the issues that should be dealt with if companies want to share information. A full account of the complexity of dealing with the issues is made when the issues are placed in the FADEE framework.

We believe that – for practitioners – the B2Bi exercise becomes more intelligible by structuring the problem domain in categories that fit with common sense. For researchers, the framework is interesting because it offers a structure to compare different cases and to investigate why some option was chosen in a specific case.

As we stated, this paper presents research-in-progress. Our future research will result in a more complete account of the DA framework. Also, we will broaden the discussion so as to draw up a TA-framework (similar to the DA framework) covering the Task Aspects. These TA issues will then be positioned together with the DA issues in the FADEE framework.

Acknowledgments

This paper has been written as part of the 'SAP-leerstoel'-project on 'Extended Enterprise Infrastructures' at the K.U.Leuven, sponsored by SAP Belgium. See http://www.econ.kuleuven.be/leerstoel/sap for related papers.

Selected References

[1] Birkinshaw J., Sheehan T. (2002). Managing the Knowledge Life Cycle, *MIT Sloan Management Review*, Fall 2002, p. 75-83.

[2] Clarke, R. Internet Privacy Concern Confirm the Case for Intervention. *Communications of the ACM*, 42, 2, (February 19999), pp. 60-67.

[3] Dynes S., Brechbüchl H. Johnson M.E. (2005). Information security in the extended enterprise.

[4] Dynes S. Security and Privacy: at odds with speed and collaboration?, p. 13.
[5] Fürst K., Schmidt T., Wippel G. (2002), Managing Access in Extended Enterprise Networks, IEEE Internet Computing, September-Oktober 2002, p. 67-74.
[6] Goethals F., Vandenbulcke J., Lemahieu W., (3/2004), Developing the extended enterprise with the FADEE, Proceedings of the ACM Symposium on Applied Computing (SAC2004), Nicosia (Cyprus), March 14-17, pp. 1372-1379.
[7] Goethals F., Vandenbulcke J., Lemahieu W., Snoeck M. (6/2004), A framework for managing concurrent business and ICT development, VI International Workshop on Learning Software Organizations LSO 2004 - Banff June 21, 2004, Springer LNCS-series, Volume 3096, pp. 131-136.
[8] Goethals F., Vandenbulcke J., Lemahieu W., Snoeck M. (11/2004), Structuring the development of inter-organizational systems: WISE (Web Information Systems Engineering) 2004 conference proceedings - Brisbane November 22-24, 2004. Springer LNCS-series, Volume 3306, pp. 454-465.
[9] Hansen M.T., Nohria N., Tierney T. (1999). What's Your strategy for managing knowledge?, Harvard Business Review, March-April 1999, p.106-116.
[10] Koch M., Möslein K.M. (2005). Identities management for e-commerce and collaboration applications, International Journal of Electronic Commerce, Spring 2005, Vol. 9, No. 3, p. 11-30.
[11] Levina N., Vaast E. (2005). The emergence of boundary spanning competence in practice: implications for implementation and use of information systems. MIS Quarterly, Vol. 29, No. 2, June 2005, pp. 335-363.
[12] Melgar, D., & Venkatapathy, C. (2003). *Message-Centric Web Services vs. RPC-Style Invocations*, Web Services Journal, April 2003, p 48-51.
[13] Patil, N. (2003). *Document-Based Web Services Using JAXM*, Web Services Journal, April 2003, p 30-33.
[14] Spronk R. (2005). Act Reference Registries: an infostructural core concept, Draft version 1.5 (2005-04-07), Retrieved from http://www.ringholm.de/en/whitepapers.htm on July 26, 2005.
[15] Spronk R. (2005). The Spine, An English national programme, Final version 1.1 (2005-03-29), Retrieved from http://www.ringholm.de/en/whitepapers.htm on July 26, 2005.
[16] Thompson, J. D. (1967). Organizations in Action: Social Science Bases of Administrative Theory. New York: McGraw-Hill.
[17] Zachman J. (1987), A framework for information systems architecture, *IBM Systems Journal*, Vol. 26, No.3, pp. 276-292.

Biographies

Frank Goethals completed his Master studies in economics (option informatics), at the K.U.Leuven, Belgium, in 2001. He is presently researching for a PhD under the theme of 'Extended Enterprise Infrastructures'. This research is conducted at the K.U.Leuven under the guidance of professor J. Vandenbulcke, and is financed by SAP Belgium. Frank has a strong interest in Web services technology and Enterprise Architecture, and is investigating how supply chain partners could integrate their systems efficiently and effectively

Jacques A. Vandenbulcke is professor at the Department of Applied Economic Sciences of the Catholic University of Leuven, Belgium. His main research interests are in Database management, Data modelling, and Business Information Systems. He is co-ordinator of the Leuven Institute for Research on Information Systems (LIRIS) and holder of the SAP-chair on 'Extended enterprise ICT-infrastructures'. He is president of 'Studiecentrum voor Automatische Informatieverwerking (SAI)', the largest society for computer professionals in Belgium, and co-founder and vice-president of the 'Production and Inventory Control Society (PICS)' in Belgium.

Wilfried Lemahieu holds a Ph. D. from the Department of Applied Economic Sciences of the Katholieke Universiteit Leuven, Belgium (1999). At present, he is associate professor at the Management Informatics research group of this department. His teaching includes Database Management, Data Storage Architectures and Management Informatics. His research interests comprise distributed object architectures and web services, object-relational and object-oriented database systems and hypermedia systems.

Monique Snoeck obtained her Ph.D. in May 1995 from The Department of Computer Science of the Katholieke Universiteit Leuven with a thesis that lays the formal foundations of the object-oriented business modelling method MERODE. Since then she has done further research in the area of formal methods for object-oriented conceptual modelling. She now is Associate Professor with the Management Information Systems Group of the Department of Applied Economic Sciences at the Katholieke Universiteit Leuven in Belgium. She has been involved in several industrial conceptual modelling projects. Her research interests are object oriented conceptual modelling, software architecture and software quality.