Interoperability in Business Information Systems

Business Document Taxonomy#
Comparison of the State-of-the-art and#
Recommendations for Future Applications#

Christian Janiesch#
European Research Center for Information Systems (ERCIS)#
University of Münster#
Münster, Germany#
christian.janiesch@ercis.de

Susan Marie Thomas#
SAP Research CEC Karlsruhe#
SAP AG#
Karlsruhe, Germany#
susan.marie.thomas@sap.com

Abstract: Interoperability is key to successful collaboration of one or more enterprises. However, due to the diversity of document exchange standards available, collaboration is hindered by extensive mapping of interfaces and transformation of messages. Intermediate standards such as CCTS provide a means to standardize business semantics, but only lately their implementation reached a stage of maturation that is adequate for practical implementation. The aim of this paper is to extract recommendations from a comprehensive juxtaposition of the current state-of-the-art exchange standards and business document libraries. Six standards (including multiple versions) are examined based on a framework of five key criteria covering actors and roles, processes, protocols, message (assembly) as well as document taxonomy and assembly. The insights are combined to form recommendations for the future design of business semantics and business syntactics for business documents. While a comprehensive repository of documents can only originate from practice, a framework to properly manage the standards’ contents should be provided by research.

Introduction#

The current state-of-the-art of document exchange involves a number of issues apart from the actual documents, which are exchanged. Previous endeavors to compare document exchange frameworks have been undertaken. While Penistera and Tarabanis [PT06] compared interoperability typologies on a broader scale, Zhao and Sandahl [ZS01; Zh03] focused on XML-based frameworks with regard to vocabulary. Schmitz et al. [SLD03a; SLD03b] concentrated on catalog standards, also only covering XML. Kotinummi et al. and Nummilaakso, Kotinummi [KNL03; NK04] covered mainly supply chain management and, again, only XML-specific issues. Androutsellis-Theotokis et al. [An+05] tackled a broader scope, however, their comparison only examined the applicability of standards for their PRAXIS framework. The representation of documents and, thus, degree of exploitation of XML is one thing, but in order to acknowledge the importance of service-based architectures (SOA) [Er05], not only the appearance and structure of the actual...
business document is of importance, but also the meta model, associated actors, messages, and choreography.

Since the area of application for most document exchange standards is different, their definition varies and makes it difficult to establish a holistic communication framework. However, a common framework to manage the essential contents of different standards does seem like a feasible way to avoid message standard proliferation and an increasing divergence of standards. The first step in that direction is a uniform picture of the possibilities and challenges of interoperability in current business information systems.

In regard to interoperability, several aspects of exchange standard specifications are relevant to the design of collaborative processes, as well as the design of the messages exchanged. For the subsequent analysis, a number of current standards for document exchange will be examined for their coverage of the following relevant aspects (cf. Figure 1): actor and/or role model, processes, protocols and protocol types, messages and message choreography, document taxonomy and document assembly.

![Diagram]

**Figure 1:** Decomposition of Standards for Document Exchange

- **Actor and/or role model:** Different actors take different roles in business processes. The standards surveyed had different interpretations of actor and role. For example, in UBL, a customer is an actor, and a buyer is a more specific role that a customer can play in a process. ebXML, on the other hand, specifies only roles.

- **Processes:** Business processes are goal-oriented, value creating sequences of activities. They provide the choreography of a collaboration scenario. Processes
can be detailed into interface processes that then can be best regarded as protocols. So, the transition of processes and protocols is gradual.

- Protocols and protocol types: Protocols describe how messages are exchanged in a business process and therefore describe the interaction on a more detailed level. They provide rules for the interaction.

- Messages and message choreography: The analysis of messages is twofold. Different message types as well as the message choreography within a business process can be specified. While the first aspect provides a dictionary of possible message types, the second aspect provides patterns that can be reused and provide input for variant handling of messages and processes. The former can also be detailed to analyze the actual interface that has to be implemented since most messages involve an XML Schema.

- Document taxonomy and document assembly: The content as well as the structure of documents are both of major concern to this analysis. Since the different approaches cover different specific areas, the scope and number of documents is not uniform and a common ground must be identified. Furthermore, only a consistent assembly of the documents ensures that reuse and variant handling can be executed in a feasible manner. Thus, number and level of detail at which the documents are specified, as well as the inherent structure are of importance.

All these areas will be subject to the analysis and they will form the basis of the resulting framework. The first step provides a comprehensive table comparing the existing standards. In a second step the conclusions of this analysis are refined into recommendations for future standards.

Comparison of Exchange Standards

Overview

As a starting point, Figure 2 gives an overview of the different layers that currently exist in the focal domain. The approaches covered in this research have been highlighted in bold letters. 
On a very basic level information is available only as stream of data that does not carry any additional information about its semantics. One of the goals of modern exchange standards is to avoid the confusion on how to treat data correctly in terms of structure, format, and meaning. The former is currently achieved by utilizing XML as a markup language to describe and structure data. Legacy systems often do not provide the option of exchanging XML data but only provide proprietary interfaces. That is why the Data Format Description Language (DFDL) is proposed as an open industry standard. Rather than prescribing a data format, DFDL is descriptive of existing formats and utilizes XML to do so. The ebXML Core Components Technical Specification (CCTS) provides the definition of data types for document exchange as well as a way to organize, i.e., aggregate and associate data on a higher level to create business documents. Being a meta standard CCTS does not provide any concrete implementation but a way of standardizing business semantics. The Universal Business Language (UBL) and the Open Applications Group Integration Specification (OAGIS) are two initiatives to provide a uniform language for document exchange based on CCTS.

While DDFS is an approach to exchange data available mainly from proprietary formats, CCTS is the dominant approach underlying the design of most of the currently available open standards for business semantics such as UBL. OAGIS as well as application specific data models such as SAP’s Global Data Types. Apart from the CCTS-based open standards mentioned above, a number of other mostly industry-specific exchange standards exist, which focus on static message definitions that have not enabled a comparable degree of interoperability or flexibility with other standards. RosettaNet, and xCBL are two examples of other important standards. The former is industry-specific, and does not employ a component-based approach, but simply urges the user to aggregate data belonging together. xCBL is somewhat aligned with UBL and features complex and simple types, however no direct connection to CCTS is mentioned. All of these approaches form their own data types which are mostly based on XML data types, and their own approach to document assembly.

Figure 2: Classification of Standards for Document Exchange
EDIFACT on the other hand is a mature standard for document interchange. It is widely used in Europe; its Northern American counterpart is ANSI ASC X12. Their structure is similar. As EDIFACT is not based on XML, it provides its own messaging structure as well as semantics. However, a structured composition of messages is not provided.

**ebXML Core Components Technical Specification (CCTS)**

CCTS does not define any actors, processes, protocols, messages or documents. It is merely a framework on how to assemble documents and form semantically enriched data types as well as a methodology to create context based instantiations of a business document master. However, certain external standards (cf. below) are named as a basis for the design of the actual document exchange. The current version of the specification is 2.01 [Cr03].

- **Actor and/or role model:** The roles of the Catalogue of Common Business Processes [SW01] are proposed as a guideline. The description of roles in the catalogue, however, “is intended to be illustrative only, not exhaustive”. These roles can apply to a sender, a receiver or a supporting role that specifies requirements but does not send or receive.

<table>
<thead>
<tr>
<th>Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent, Beneficiary, Buyer, Cashier, Customer, Customer Service, Employee, Factory Worker, Logistics Worker, QC Inspector, Salesperson, Seller, Student, Supervisor, Trucker, Vendor</td>
</tr>
</tbody>
</table>

*Table 1: Roles in the Catalog of Common Business Processes [SW01]*

- **Processes:** CCTS suggests the use of the Catalogue of Common Business Processes [SW01]. The processes therein are intended to be a library of common industry-independent processes. The specification does also include industry-specific processes but their selection seems to be merely illustrative. The processes are clustered into normative categories and sub-categories covering the complete procurement process.

- **Protocols and protocol types:** CCTS does not include or reference protocols.

- **Messages and message choreography:** CCTS does not provide any information on messages or message choreography. Neither does it provide any suggestions on message assembly and leaves that to the actual implementations that make use of CCTS.

- **Document taxonomy and document assembly:** A document taxonomy is not in the scope of CCTS since it does not provide any information about concrete documents. Instead CtTS provides a means to identify, capture, and maximize the re-use of business documents. It aims at providing a syntax-neutral framework, e.g., for XML and/or UN/EDIFACT-based messages. Documents are assembled from so-called core components. Although CCTS does not define documents, it does contain approved core component types that are to be used as a basis for any basic core component (BCC), aggregate (ACC) and associate core component (ASCC). Currently there are 10 approved data types that are based on primitive data types such as string and decimal.
Data Types

| Amount, Binary Object, Code, Date Time, Identifier, Indicator, Measure, Numeric, Quantity, Text |

Table 2: Approved Core Component Types of CCTS [Cr03]

To provide a means to instantiate Core Components in a real world scenario, Business Information Entities (BIE) are created from the core components. BIEs are a piece of business data or a group of pieces of business data with a unique business semantic definition. It is intended to provide a common ground for business document discovery and definition. The UN/CEFACT International Trade and Business Processes Group (TBG) provides a dictionary (Business Process and Core Components Harmonisation) including currently 324 BCC, 49 ACC, and 102 ASCC. The activities related to the TBG are within the mission and objectives of UN/CEFACT and its empowered groups [Pr05].

Universal Business Language (UBL)

UBL provides a repository of universal procurement business processes documents and their components. The current version of UBL 2.0 is a greatly enlarged public working draft [BM06].

- **Actor and/or role model**: UBL distinguishes actors and roles. All interaction takes place between a customer and a supplier (e.g., buyer and seller). But processes may also involve supplementary roles that do not influence the actual process but place requirements on data exchange. Supplementary roles may be filled by different parties, e.g., consignor. The roles put the actors into context. They are uniquely assigned to only one actor as the following table shows.

<table>
<thead>
<tr>
<th>Actor</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td>Originator, Buyer, Delivery, Debtor</td>
</tr>
<tr>
<td>Supplier</td>
<td>Seller, Despatch, Creditor, Payee</td>
</tr>
<tr>
<td>Party</td>
<td>Catalogue Managing Party, Information Content Owner, Receiver, Sender, Consignor, Consignee, Freight Forwarder, Carrier, Exporter, Endorser, Importer</td>
</tr>
</tbody>
</table>

Table 3: Actors and Roles in UBL 2.0 [BM06]

- **Processes**: While UBL 1.0 [MS04] only covered the Order-to-Invoice process the current public draft of UBL 2.0 covers the whole procurement process from sourcing to payment. The processes in UBL are of generic nature and describe business rules and choreography to put the UBL document types into context. They can be seen as a general outline of the areas that have to be covered by a comprehensive framework. So the processes of UBL are more abstract than, e.g., those of RosettaNet and have to be instantiated first. Cf. Table 4 for an overview of the process clusters.
# Business Process Collaboration#

Sourcing, Ordering, Fulfillment, Billing, Payment, Initiate Transport Services,
Certification of Origin, Supplementary

Table #4: Process Clusters in UBL 2.0 [BM06]

- **Protocols and protocol types:** UBL does not strictly distinguish different protocol types. There is however a notify interaction that does not expect an answer and a request interaction that receives a generic application response. Some interactions are not typed in any way and therefore may or may not expect answers.

- **Messages and message choreography:** UBL only provides a starting point for message assembly and choreography. The business documents of UBL can be regarded as the payload of messages. Message assembly, however, is not the scope of UBL. Thus, a proper choreography can only be derived from the use cases provided in the specification, but is not explicitly mentioned.

- **Document taxonomy and document assembly:** UBL provides a library of core components that conforms to CCTS. Cf. Table #4 for an overview of the document clusters. 29 documents are specified in XML Schema and make use of currently 107 aggregate components and 196 basic components. The terminology of UBL suggests that it provides already BIEs and not core components. However, for implementation it is more than likely that projections of UBL documents have to be created, which only contain a subset of the information.

# Business Document Cluster#

Sourcing, Order-to-invoice, Fulfillment, Billing, Payment, Supplementary

Table #5: Business Documents in UBL 2.0 [BM06]  

Recommendations for the development of derivative implementations such as national and industry profiles of UBL are to be provided as part of the UBL 2.0 Support Package, which is not yet available. It is expected that UBL will conform completely to the CCTS specification.

# Open Applications Group Integration Specification (OAGIS) #

OAGIS aims at delivering a horizontal message structure that provides a common business understanding for all participants. Business Scenarios are offered as models for starting points of proprietary situational implementations but not as blueprints [OAG06]. The business documents of OAGIS 8.0 were based on their own component based approach, but OAGI adopted the CCTS approach in its latest release OAGIS 9.0.

- **Actor and/or role model:** OAGIS does not specify any actors or roles. This has been a conscious decision since document routing, which is one of the main reasons to define roles, is not regarded as part of a business document standard [OAG06].

© IBIS-# Issue 2 (2), 2006 #
• Processes: OAGIS does not provide any processes but only components of verb and nouns to assemble messages for basically every business case. Business scenarios are said to be available but are not part of the current specification offered as a download. OAGIS 8.0 [OAG02] provides 61 unsorted scenarios covering in detail order management, warehouse management, purchasing and other financials as well as some singular processes.

• Protocols and protocol types: Since no processes are given, no collaboration protocols can be found either. However, OAGIS provides a component based approach to message assembly by providing business documents as well as common actions that can be either to the object. These actions hint at the need for either unidirectional or bidirectional protocols since some of them expect a response or confirmation.

• Messages and message choreography: OAGIS explicitly defines messages. Messages (Business Object Documents) consist of an “Application Area” and a “Data Area”. The former carries information that an application may need to know in order to communicate in an integration scenario involving two or more business applications. The latter is composed of a “Verb” and a “Noun”. The verb includes the action to be applied to the object while the noun is the actual payload, object, or document that is being acted upon. The naming of messages is Verb+Noun, e.g. AcknowledgeRFQ. Cf. Figure 3 for a structured summary of OAGIS Business Object Documents, i.e. OAGIS messages.

![Business Object Document (BOD)](image)

Figure 3: Message Structure of OAGIS 9.0 [OAG06]

Verbs are either action verbs (8 different actions) or response verbs (4), i.e. they are used either to initiate communication or to respond to a request. In an effort to simplify the usage of the verbs certain verbs are marked as deprecated (7) and will be removed from future versions of the standard.

<table>
<thead>
<tr>
<th>Current Action Verbs</th>
<th>Current Response Verbs</th>
<th>Deprecated Verbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel, Change, Get,</td>
<td>Acknowledge, #</td>
<td>Add, Allocate,</td>
</tr>
<tr>
<td>Load, Post, Process,</td>
<td>Confirm, Respond, #</td>
<td>Create, #</td>
</tr>
<tr>
<td>Sync, Update</td>
<td>Show</td>
<td>GetList, Issue,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>List, #</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receive #</td>
</tr>
</tbody>
</table>

Table 6: Verbs of OAGIS 9.0 [OAG06]
OAGIS distinguishes 434 messages assembled from business documents and verbs. Their choreography within the scenarios in Version 8.0 is not explicitly defined in most cases but can be extracted from the scenario models. Some of these integration scenarios have details concerning their workflow, assumptions, and exception requirements while others do not. The structure of the nouns is described in the following.

- **Document taxonomy and document assembly**: OAGIS does not provide a taxonomy of documents but a flat list of 77 nouns that are aggregated according to CCTS from 61 reusable BIEs (which are based on 57 reusable Core Components defined by the Business Process and Core Components Harmonisation Group, TBG17 [SW01]). In addition, OAGIS defines common as well as domain-specific OAGIS components which can be used to cluster documents. For an overview of the OAGIS components cf. Table 7.

---

### Components

<table>
<thead>
<tr>
<th>Common, CRM, Financial, Logistics, Order Management, Manufacturing</th>
</tr>
</thead>
</table>

Table 7: Common and Domain-specific Components of OAGIS 9.0 [OAG06]

---

### XML Common Business Library (xCBL)

xCBL is a set of XML business documents and their components. It aims at assisting business to business e-commerce. The current version of xCBL, v4.0, adopted XML Schema for specification, and is somewhat aligned with UBL and, thus, indirectly close to CCTS. [CO03]

- **Actor and/or role model**: xCBL does not specify any actors or roles.
- **Processes**: xCBL provides only basic support concerning business processes. In fact, only for order management does an explicit choreography exists. The specification provides a textual overview of four sections of business cases each with three to four different groupings of activity.
- **Protocols and protocol types**: Since no processes are given, no collaboration protocols can be found either.
- **Messages and message choreography**: As mentioned above, there is one single choreography covering order management.
- **Document taxonomy and document assembly**: xCBL distinguishes 8 functional areas of business documents, covering altogether 44 business documents. These documents are assembled from altogether 643 complex types as well as numerous simple types that are based on XML schema data types. Cf. Table 8 for an overview of the functional areas.

---

### Functional Areas

<table>
<thead>
<tr>
<th>Application Integration, Catalog, Core, Financial, Materials Management, Message Management, Order Management, Preorder Management, Statistics and Forecasting</th>
</tr>
</thead>
</table>

Table 8: Functional Areas of xCBL 4.0 [CO03]

---

© IBIS+ Issue 2 (2), 2006
Apart from a Standards Methodology (RSM) RosettaNet offers so-called Partner Interface Processes (PIP) as well as dictionaries and an implementation framework. The latter provides a comprehensive overview of the structure of RosettaNet message exchange. Since RosettaNet PIPs cover a large variety of tasks only cluster 3 Order Management was examined for this analysis.

- **Actor and/or role model**: RosettaNet does not distinguish actors and roles, but just employs roles. Due to the fine-grained design of the PIPs only two partners interact in each process. All roles displayed below can be used for senders as well as receivers.

<table>
<thead>
<tr>
<th>Roles</th>
</tr>
</thead>
</table>

### Table 9: Roles in RosettaNet Partner Interface Processes in Cluster 3 [Ro02b] |

- **Processes**: RosettaNet provides a huge amount of PIPs which describe the interaction between two business partners. RosettaNet divides the entire e-business supply chain domain for which PIPs are specified into broad classifications called clusters. Each cluster is further subdivided into two or more segments. Each segment comprises several PIPs. PIPs contain one or more activities, and activities in turn specify actions. Each PIP in a segment represents a well-defined business process subset, and is named with the cluster, segment, and sequence number of the PIP in the segment [Ro02a].

- **Protocols and protocol types**: Most of RosettaNet's PIPs are named according to the protocol used: request/response and notification. RosettaNet distinguishes two broad categories: business action message and business signal message. Business actions are messages with content that is of a business nature. Business signals are positive and negative acknowledgment messages that are sent in response to business actions. Only business actions are acknowledged. Business signals are never acknowledged. Both categories can be implemented for synchronous or asynchronous communication [Ro02a].

- **Messages and message choreography**: The individual business documents involved in a PIP (i.e. action and signal messages) are exchanged in a container that packs together other related entities such as headers, attachments, and digital signatures. A RosettaNet Business Message always contains a preamble header, a delivery header, a service header, and service content. Service content comprises an action message or a signal message. If service content is an action message, one or more attachments may be included. Cf. Figure 4 for an overview.
The service content part of the payload contains business content that is in XML format. The service content is always either an action message or a signal message. The attachments contain additional data, e.g., PDF or TIFF files.

- **Document taxonomy and document assembly:** RosettaNet distinguishes eight clusters of PIPs each containing segments in which documents are exchanged. RosettaNet does not reuse business documents. RosettaNet documents, i.e., service content, are assembled from so-called business data entities, business properties, entity instances, fundamental business data entities, and quantitative fundamental business data entities according to the DTD or XML Schema provided by the PIP specification. Basic elements that constitute these entities and properties are, e.g., a name, a definition, or a representation. They do not follow a building block approach but rather assemble a RosettaNet message line by line.

**UN/EDIFACT**

UN/EDIFACT is currently the international standard for electronic data interchange. It is maintained by UN/CEFACT[UN06]. UN/EDIFACT recommends which pieces of information are mandatory for a particular document, which pieces are optional and give the rules for the structure of the document.

- **Actor and/or role model, processes, protocols and protocol types, messages and message choreography:** UN/EDIFACT does not distinguish between messages and business documents. Actors, roles, processes and protocols are not part of the specification. UN/EDIFACT distinguishes batch and interactive documents which implicitly means that UN/EDIFACT supports notification as well as request-response protocols.

- **Document taxonomy and document assembly:** Documents are not clustered or sorted in any way. They are available as a list. Their structure follows a basic building block approach and utilizes segments and composite elements based on data elements. However their assembly is rather line by line.

Summarizing, there are few things that UN/EDIFACT can provide as input to current development apart from the need to be backwards compatible. Since CCTS...
provides a means to structure UN/EDIFACT messages, too, they can be transformed from and to existing XML standards more easily than before. This means that compatibility has to be achieved on a fine-grained level concerning mandatory attributes and content of messages and documents.

**Comparison Chart**

The following table gives an aggregate overview of the above analysis. It comprises all major standards including their prior versions if major updates have occurred. This is especially applicable for OAGIS, since version 8.0 to 9.0 entailed the utilization of CCTS, and for UBL, which has been greatly enhanced for version 2.0 even though it is still a working draft.

<table>
<thead>
<tr>
<th>#</th>
<th>Actors</th>
<th>#</th>
<th>Roles</th>
<th>#</th>
<th>Processes</th>
<th>#</th>
<th>#</th>
<th>Protocols</th>
<th>#</th>
<th>#</th>
<th>Messages</th>
<th>#</th>
<th>#</th>
<th>Choreography</th>
<th>#</th>
<th>#</th>
<th>#</th>
<th>Documents</th>
<th>#</th>
<th>#</th>
<th>Document Assembly</th>
<th>#</th>
<th>Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCTS 2.01</td>
<td>#</td>
<td>(x)</td>
<td>(x)</td>
<td>#</td>
<td>(x)</td>
<td>#</td>
<td>#</td>
<td>(10/253)</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>-</td>
<td>x</td>
<td>-</td>
<td>X</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UBL 1.0</td>
<td>#</td>
<td>#</td>
<td>(16)</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(x)</td>
<td>#</td>
<td>#</td>
<td>(x)</td>
<td>#</td>
<td>59/143</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UBL 2.0 Draft 13</td>
<td>x</td>
<td>x</td>
<td>X</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>1/8</td>
<td>6/29</td>
<td>107/196</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OAGIS 8.0</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>7/61</td>
<td>190</td>
<td>6/29</td>
<td>191/165</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OAGIS 9.0</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(x)</td>
<td>#</td>
<td>#</td>
<td>(x)</td>
<td>#</td>
<td>#</td>
<td>X</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>61</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RosettaNet (DC 3)</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>124</td>
<td>43</td>
<td>6/77</td>
<td>547/357</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xCBL 4.0</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(x)</td>
<td>#</td>
<td>#</td>
<td>8/44</td>
<td>643</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UN/EDIFACT D.05B</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>209</td>
<td>#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Table 10: Comparison of Standards*

For each standard and aspect of the analysis, the number of occurrences is given. If the aspect is only implicitly mentioned or referred to (as e.g. certain catalogs by CCTS) the result is bracketed. The two numbers in the columns processes, #
IBIS#Interoperability in Business Information Systems#

#
documents, and documents assembly always cover aggregate and simple#
ocurrences, respectively, i.e., #process cluster and processes, document cluster#
and documents, and aggregate/uncomplex components and basic/simple#
components. #Figure#5# gives an overview of the interrelations between the specific#
versions of the analyzed approaches. #Arrows link related versions and the dashed#
line refers to general alignment of the two approaches. #
#

```
xCBL 4.0
```

```
UML 1.0  UML 2.0  RosettaNet
```

```
OAGIS 8.0  OAGIS 9.0  UNEDIFACT
```

```
Non-CCTS Component-Based Approaches
```

```
Approaches based on CCTS Core Components 201
```

```
Other Major Approaches
```

---

**Recommendations for the Future Design of Business#**

**Documents and Exchange Standards#**

The recommendations# offer a holistic viewpoint on interoperability in#
heterogeneous enterprise networks and their applications. Thus, the goal is to#
deck general requirements on such a framework that facilitates collaboration on#
a business level. #
#
- **Actor and/or role model**: The distinction of actors and roles helps to specify in#
more detail who is the:Sender and who is the:Receiver of the message and who#
influences the messages format:Supporting Actor. Such a division in three#
parts provides a feasible means of clustering roles that are relevant for an#
process. While the OAGIS approach, which does not specify receivers, will work#
in an actual implementation, it is not suited for the specification of a document#
repository. This is because without information about receivers, the deployment#
scenario is too vague. The number of roles should be kept at a reasonable level#
to avoid the specification of roles for every new task. UBL provides a good#
overview which has to be matched to the reference processes selected. Cf.#
Figure#6# for the depiction of actors. #

---

© IBIS#Issue#2#(2), 2006#
Processes: The actual number of processes is not nearly as important as a clear structuring of the processes. For an implementation in the real world almost any list has to be extended to provide the correct fit for a concrete scenario. Thus, giving structure to the collaboration processes must be the primary objective. With regards to the other standards an ordering of the actual processes in clusters or categories as well as a sub-structuring in segments or sub-categories provides enough flexibility to store the actual processes. For a simple Taxonomy of Customary Business Processes for procurement covering four distinct sections with a varying number of sub-sections cf. Figure 7. The last section is a reservoir for minor processes that either apply to all other sections and sub-sections, or that do not fit properly into the structure, but do not justify a separate sub-section. The processes actually described in these sub-sections will always have the character of exemplary business scenarios that have to be adapted to the actual business need. They contain sort of a message choreography for the scenario. Each process can be detailed into components which are a container for describing the lowest level of execution.

Protocols and protocol types: Protocols tend to be more complex than suggested by the different analyzed standards. At least three protocols are needed. One unidirectional pattern without an answer (notification) and two bidirectional patterns which either expect an answer (request/confirmation) or which do not necessarily need an answer to terminate (query/response). Furthermore protocols can be categorized by their synchronicity. Communication is either synchronous or asynchronous. i.e. while a message is sent the originating system can wait (logical synchronicity) or continue execution (technical and logical asynchronicity). Cf. Figure 8 for an overview of...
protocals. In addition to the simple one-to-one pattern, one-to-many patterns may be required to execute complex business processes involving multiple buyers or bidders. A good overview describing so-called Service Interaction Patterns is given in [BDH05].

**Figure 8: Taxonomy of Business Protocols**

- **Messages and message choreography:** Business messages provide a wrapper for the actual business document and should be designed as such. Messages need to include a header and payload. While the former contains information about sender and receiver, timestamp, ID, references etc., the latter should contain the action and the document to be acted upon (cf. [OAG06]). The message action and the business document should be separated to decrease and standardize the number of documents. Furthermore a non-mandatory binary object, attachment, e.g., a PDF, ZIP, or TIFF file, should be allowed to ease non-standardized data transfer or include application code. Cf. Figure 9 for an overview of the message structure.

**Figure 9: Message Structure**

It is sensible to standardize the message actions and business documents to avoid uncontrolled growth thereof. Therefore, the following ten message types provide a...
common ground that should enable creation of messages equivalent to those in the standards analyzed. Cf. Figure 10 for an overview.

Document taxonomy and document assembly: While components that are employed to assemble messages should be re-used, the actual business documents should be uniquely placed in a sound Taxonomy of Customary Business Documents. Each section comprises at least one document. To allow for efficient retrieval, the documents should be further analyzed concerning the applicable actors and roles, message types as well as their protocols. This metadata assists the identification of the business document as much as the actual structure of the document. Cf. the following Figure for the structuring taxonomy.

The CCTS approach has been widely adopted so far. Thus, document assembly should be aligned with the building block approach. Cf. the following Figure for an assembly meta model in ER notation [Ch76] and a structural diagram of a simplified CCTS compliant approach to document assembly. Association Components are of an Aggregate Component (AS-AG-As). Aggregate Components are assembled from...
other Aggregate Components or Basic Components. Basic Components are of a Data Type (BC-DT-As), which are of a prescribed Core Component Type (CCT-DT-As). Basic Components and Association Components are used to form Business Documents; both components may or may not be attached with a property. Properties are terms to specialize a component (Address as opposed to MailingAddress). It must be possible to annotate whether a component is a mandatory part of the superordinate component by cardinalities.

Figure 12: Meta Model and Structural Display of Document Components

Conclusions

Summarizing, the development of actual exchange standards is being influenced by other intermediate standards that facilitate communication and standardization of messages and documents. The first notable layer is XML as a general exchange language and more recent CCTS is implemented as a method of business semantics standardization. The acceptance of one general business language has not occurred, but might be the long-term solution, and could be based on the building blocks concept proposed by CCTS.

CCTS provides a comprehensive framework on how to structure documents that is independent of technical representation. Its increasing utilization for new releases of exchange standards provides a promising common ground for further development. Any specification solely describing their documents directly in a technical format such as XML will always have to adapt to revisions, e.g., the transition from DTD to XML Schema. A modular model-based design allows the specification and even more important, the management of documents in an economical and consistent way. Further research in the area of document engineering must therefore focus on data structure independent modeling of business documents. Although document engineering methodologies exist, to our knowledge no comprehensive approach to business document modeling.

© IBIS+Issue2#(2), 2006
exists that exceeds pragmatic reuse of UML‡ [BM06], graphical XML representation‡ [OAG06], text-based approaches‡ [MS04] for more graphical display (cf. SAP R/3‡ business object documentation). Only then, the next big step in document engineering, the contextualization and configuration of business documents‡ [Cr03‡; JDGL06], can be properly dealt with.

Acknowledgements‡

The work published in this paper is (partly) funded by the E.C. through the ATHENA‡ IP. It does not represent the view of E.C. or the ATHENA consortium, and authors are solely responsible for the paper’s content.

References‡


# Interoperability in Business Information Systems


© IBIS+•Issue2+(2), 2006
Biographies

Christian Janiesch is research assistant and PhD student at the European Research Center for Information System (ERCIS) of Prof. Dr. Jörg Becker at the Westfälische Wilhelms-Universität in Münster, Germany. He has been researching at Università della Svizzera italiana (USi) in Lugano, Switzerland and at SAP Research CEC Karlsruhe, Germany. Christian completed his bachelor’s and master’s degree in Information System (BScIS, MScIS) at the University of Münster and Queensland University of Technology (QUT) in Brisbane, Australia. His current areas of research are Conceptual (Reference) Modeling, Business Intelligence, and Enterprise Systems Interoperability.

Susan Thomas is currently researching at SAP Research CEC Karlsruhe, Germany. She received a Master of Science in Engineering in Applied Mathematics (1979) from The Johns Hopkins University, Baltimore, Maryland. She has over 25 years of IT experience, the first 20 years with Digital Equipment Corporation, where she worked as a teacher, a software developer, and a project manager. Her current research interest is semantics, especially the relationship between ontologies and terminology management. Another interest is environmental and social sustainability.