UN/CEFACT Service-Oriented Architecture
Enabling Both Semantic And Application Interoperability

Christoph Schroth[1,2], Till Janner[1,2], and Gunther Stuhec[3]

[1] SAP Research CEC, Blumenbergplatz 9, 9000 St.Gallen, Switzerland
[2] Institute for Media and Communications Management, University of St. Gallen, Blumenbergplatz 9, 9000 St.Gallen, Switzerland
[3] SAP AG, Dietmar-Hopp-Allee 16, 69190 Walldorf, Germany

Abstract. Service-Oriented Architectures are about to emerge as the next-generation standard for seamless and automated e-Business infrastructures and cross-organizational Enterprise Application Integration. With the help of unified interfaces, individual systems' particularities can be hidden, thereby facilitating the exchange of data between heterogeneous applications. However, existing concepts mainly concentrate on technological issues and must be extended toward the semantic level to ensure that different applications have the same understanding of the business meaning of services. We propose a novel composite of several methodologies issued by the UN/CEFACT that bridges the semantic gap existing in current SOA deployments by introducing a loose coupling between the meaning of business data and its representation. This standards composite builds upon an evolutionary and collaborative approach and supports the modeling of both commonly comprehensible business documents and collaboration processes.

1 Introduction

Service-Oriented Architectures (SOAs) have enormous potential to enable seamless and automated e-Business infrastructures and cross-organizational Enterprise Application Integration (EAI). With the help of Web Services, particularities of individual middleware platforms can be encapsulated and made publicly available as uniform interfaces, thereby eliminating the need for huge application integration efforts. One challenge of cross-organizational interoperability that has not yet been solved by the SOA concept, however, is that of semantic integration. Philip Merrick, former chairman and CEO of webMethods, has described this issue in the following way [1]: "...there's a whole other layer to deal with, what I call the semantic integration problem. Web services are great but they standardize pure connectivity between applications. The applications still have highly varied data models, extremely different ideas of what business processes should look like". According to Roger Sippl, co-founder and chairman of Above All Software [2], the establishment of a common understanding of business semantics is one of the last "silo walls" that need to be taken down on the way to true enterprise application interoperability.

Since semantic interoperability is often overlooked or an afterthought in the development of a SOA [3], we propose a novel approach that enables applications to exchange commonly comprehensible information. According to the Open-Edi Reference Model [4], two relevant aspects of electronic business transactions can be considered: The Business Operational View (BOV) and Functional Service View (FSV). The BOV mainly addresses the meaning of business data, whereas the FSV accounts for the related supporting services with the only focus on technological implementation details. Our concept aims at spanning a bridge between the two views and builds upon a novel composite of several specifications issued by the United Nations Center for Trade Facilitation and Electronic Business (UN/CEFACT) as a foundation of a technology-independent semantic integration layer on top of SOAs. This composite allows for modeling both business data and processes in a collaborative and evolutionary way. Our approach is component-oriented and aims at a loose coupling between semantics and technical implementation since different technologies can be used to represent the same semantic content. This approach significantly facilitates cross-organizational interoperability without prescribing users to leverage yet another inflexible standard.

The remainder of the paper is structured as follows: In Chapter 2, we present related research work. Chapter 3 highlights the potential of SOAs to become the basis for seamless e-Business infrastructures and also elaborate on the remaining challenges inherent to SOA concepts. In Chapter 4, we propose a novel, evolutionary and collaborative e-Business stack that mainly focuses on the semantic layer and is based on several UN/CEFACT specifications. Finally, Chapter 5 provides the interconnection of the above mentioned semantic-level e-Business stack and SOAs as underlying technology for application integration. Chapter 6 closes this work with a brief outlook on future research topics.

2 Related Work

Achieving semantic interoperability on top of SOA implementations is a research topic of high current relevance. In [3], for example, reasons for semantic gaps between applications are discussed and different approaches as well as best practices for bridging these gaps are shown: The heterogeneity of legacy IT assets, the quick change of business requirements, the lack of information integration logic reuse and the business terms and definitions chaos are considered as major hurdles on the path to a common cross-application interpretation of business information. Point-to-point and hub-and-spoke semantic integration, master data management and industry information models are cited as existing approaches to bridge cross-organizational semantic gaps. To really achieve semantic interoperability, the authors recommend best practices such as establishing collaborative environments for defining a common understanding of data and embracing in industry standards.

In [5], three measures are assumed to be crucial for achieving semantic interoperability: First, standardization bodies will play a major role in lowering the

barriers to interoperability by reducing the differences among various data and process representations. Second, the meaning of data and process models on the one side and their technical representations (e.g., schemas and process execution languages) on the other hand must not be bound tightly together. In case data structure and meaning are inextricably affiliated with each other, integration problems occur frequently due to a different representation of the same content. According to the authors, only a "loose coupling at the semantic level" is a solution to this problem. Third and last, methodologies such as data transformation, data categorization and encoding of unstructured data with metadata are mentioned as enablers for semantic integration. Apart from that, the ontology paradigm is frequently cited as a means for resolving the business standards dilemma [6]. Ontologies are envisioned to work as carriers of shared semantics and to have the potential to improve (re-)usability and comprehensibility of resources published in the Web. They are a proper means for defining a common basis for understanding information exchanged between services and can also be leveraged for a uniform service description (Semantic Web Services) [7], but usually do not allow users to flexibly introduce new ontological elements and conventions. As described in [8], current research focuses on the application of SOA concepts for the definition of extensible ontologies that can be dynamically adapted to user requirements.

In the sections below, we will elaborate on a novel approach that accounts for numerous of the best practices and recommendations mentioned above and also goes beyond as it unifies both process and data modeling and provides an extensible basis for semantic interoperability on top of SOA implementations.

3 SOA: State-of-the-art and Interoperability Challenges

3.1 SOA Reference Model

Since the late 1990s, numerous different definitions of SOA have been published ([9], [10]). Within this work, we adhere to the OASIS Reference Model for SOA ([11]). It defines SOA as *"... a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations."* [11]. The components of a basic SOA and their possible interactions are: a service provider *publishes* his service interface via a service registry where a service requester can *find* the service and perform the *binding* to service provider. The central concept of the SOA Reference Model is the existence of *services* which provide access to capabilities by well-defined interfaces to be exercised following a service contract with constraints and policies. This enables a loose coupling of services and complies with some of the probably most-known principles in software-engineering, information-hiding and modularization [12]. Services are provided by entities, the service provider, and are to be used by others, the service consumers.

3.2 Web Services

The concept of supporting loosely coupled, business-aligned and networked services can be realized with the help of numerous different technologies. As WSDL/-SOAP-based web services are the most widely spread application of SOA on the Web, we take these as a foundation of our work. The Web Service Description Language (WSDL) thereby defines a XML format for the interface description which is divided into an abstract part (interface details such as messages, operations and port-types on a technology-independent level) and a concrete part (details of the technical binding based on protocol and data formats). On the protocol level, SOAP specifies the data format of the messages to be sent between service provider and requester whereby HTTP is the most used data transfer protocol. The Universal Description, Discovery, and Integration (UDDI) standard is leveraged to define publicly available registries that are needed for service search and identification. The Business Process Execution Language (BPEL) is widely accepted as a standard for orchestrating different services into one process choreography. Also based on XML, it focuses on the description of a business process from the view of one participant.

3.3 Interoperability Challenges

A web service-based SOA realization as described above improves the interoperability of different applications by encapsulating system specifics with the help of interfaces and also allows for flexibility as processes can be quickly orchestrated on the basis of existing services. Web services usually rely on the XML Schema language to determine the structure of documents that are exchanged between the applications [8]. However, it is not possible to capture semantics of message elements with the help of such schemas. Applications in different contexts or domains might name the same data item in different ways, thereby preventing a common understanding of the information transmitted. As outlined above, this lack of a uniform interpretation of business data is due to the uncontrolled and non-harmonized emergence of a huge number of e-Business standards and languages that tightly couple semantics and technical representation of data [6]. Current SOA implementations also usually lack a semantically rich description of the services involved. Automatic discovery and composition of services cannot be realized unless the naming of service properties is unambiguously defined and thus machine-readable. Especially in case of cross-organizational business processes, SOAs only provide the technical means for composing services towards a desired collaboration process, but do not build upon a widely spread and well defined framework for naming or orchestrating services.

4 The Collaborative and Evolutionary UN/CEFACT Stack

As argued in the previous chapters, achieving a common understanding of the data exchanged between services and establishing an interoperable way of or-

chestrating services are the major remaining challenges inherent to SOAs. As presented in [13], several standards exist to model business processes and information. An important cornerstone of e-Business history was the establishment of the UN/EDIFACT standard [14] that tried to consolidate the existing multitude of ASCII-based EDI-standards. However, static definitions and industry-specific formats still dominate nowadays, impeding any horizontal integration. The emergence of various XML-based standards promised to solve the main drawbacks of traditional EDI-based technologies like very specific expert knowledge, non-flexibility and complexity. However, state-of-the-art e-Business frameworks such as RosettaNet [15] still do not sufficiently facilitate semantic interoperability but provide static message definitions that aim at the needs of specific industries and tightly couple data semantics and their technical representation.

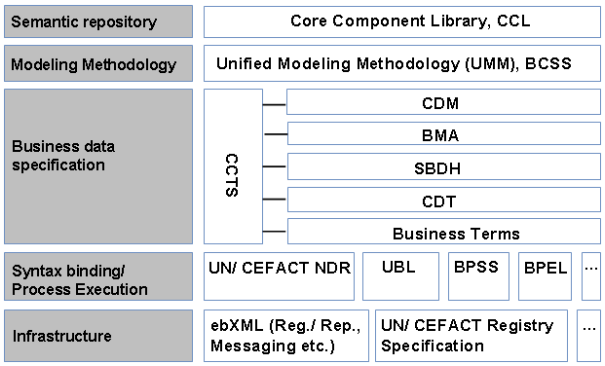


Fig. 1. The modular UN/CEFACT e-Business stack

The UN/CEFACT standardization organization desires to bridge the semantic gap in cross-organizational collaboration which has emerged from a non-controlled definition of business libraries for data and process modeling as well as a huge number of different syntactical representations [6]. For solving this issue, the UN/CEFACT issued eleven major specifications as modular blocks of a next-generation e-Business stack [13]. We provide a detailed description and a maturity analysis for each of them in the following. As can be seen in Fig. 2, most of the standards of the envisioned UN/CEFACT e-Business stack are technology-agnostic and can be considered to be part of the Business Operational View (BOV) [4], whereas only few recommendations are provided that are related to implementation details (Functional Service View, FSV).

CCL and CCTS. First, the Core Component Library (CCL) [16] represents the repository for generic business data components, the so called Core Components. Based on the experiences gained in previous data standardization efforts, the CCL does not provide pre-determined, static or industry-specific data definitions, but comprises a huge set of context-agnostic, generally valid data templates (e.g. postal address, personal information) that are syntax-independent

and represent the general business data entities which are commonly used in today's business processes. Major benefits of leveraging such a Core Component Repository include an increase of reuse of data elements during modeling and improved enterprise interoperability due to a common basis for business information description. In February 2006, UN/CEFACT announced the formal release of the first UN/CEFACT Core Component Library. It has been produced in compliance with existing procedures and is considered satisfactory for implementation. As depicted in Fig. 2, we consider this specification to be rather mature, but the integration of more Core Components is necessary for ensuring that arbitrary business documents can be setup on their basis. UN/CEFACT envisions this library to grow and also change over time as users can either modify existing components or design and submit new Core Components in case the existing ones are not sufficient to fulfill the actual business requirements. The Core Component Technical Specification (CCTS) [17] is the associated methodology comprising meta models and rules for the semantically unambiguous definition of business information on a syntax-independent level. This second building block of the UN/CEFACT stack provides users with guidelines for correctly naming and combining Core Components and introduces ways to apply context-specific restrictions on the generic data templates by adding semantic precision for a given business requirement. The generic data templates can thus be adapted and restricted to the individual requirements of users, thereby ensuring the scalability of the whole system.

An example for the use of syntax-free Core Components is as follows: a postal address is a so-called Aggregate Core Component (ACC), that consists of numerous components for representing data at leaf-level, such as "Street Name", "Zip Code" etc. These basic data items that cannot be further decomposed into other components are called Basic Core Components (BCCs). The generic address template comprises all BCCs that could ever occur in an arbitrary user context and is stored as an unstructured, syntax-free list. In case a Greek user desires to model a business document that comprises a delivery address, he is able to access the CCL, apply his Greek context and is provided an address component that features only those BCCs that are relevant for his specific situation. This contextualized subset of the generic template is then called Business Information Entity (BIE). To semantically enrich the BIE even more, it is also possible to add qualifiers and, for example, transform a "Zip Code" into a "Post Office Box_ Zip Code" to allow for differentiation from other "Zip Code" instances. In this way, all users have a common understanding of how the different parts of an address are named and are yet able to use arbitrary syntactical representations for their business documents.

UMM. The UN/CEFACT Modeling Methodology (UMM) [18], the third major composite of the e-Business framework presented in this paper provides a basis for modeling cross-organizational, collaborative business processes in an efficient fashion that also facilitates reuse. UMM is a syntax-neutral and technology-agnostic methodology for modeling processes by assembling business transactions that use CCTS compliant data models for representing messages

exchanged between trading partners. UMM provides a formalism for systematically capturing business logic that has originally been defined in agreements or contracts between business partners. It is based on UML and encompasses a set of pre-defined patterns, process stereotypes and so-called tagged values (for defining quality of service parameters) and thus fosters reuse of already existing process and data components. On the basis of these stereotypes, whole business processes can be assembled that comply with the standard, thereby ensuring interoperability between different users. The main idea of UMM is the guided and seamless modeling of collaborative business processes from the initial agreement/ contract to the implementable artifacts. Because the methodology intends to integrate users with different degrees of technical sophistication, UMM features three major views for modeling business processes of which each provides a different level of detail. The most detailed view (Business Transaction View, BTV) can be modeled with the help of UML-based activity graphs representing business transactions and business collaboration protocols. Hofreiter et.al. [19] have published a first comprehensive exemplary implementation of an UMM-based modeling tool that works in an intuitive, graphical way and enables business users to model both processes and data in UML. In this way the maturity of this specification has been proved in a real-world implementation scenario.

BCSS. As a fourth part, the Business Collaboration Schema Specification provides a UML based representation of the CCTS rules and artifacts [12]. Modeling environments may leverage this standard which is available as a first draft to make sure users are enforced to only create CCTS-compliant data models.

CDM, BMA, SBDH, CDT and Business Terms. The UN/CEFACT has issued several specifications that are strongly related to the CCTS: the Context Driver Methodology (CDM) will provide a possibility to apply context-specific categorization and restrictions on both business processes and data artifacts. Next, the Business Message Assembly (BMA) specification features guidelines for assembling higher level business information for complete, electronic messages, which can be used in the UMM based business transactions. Both the previously explained CDM and the BMA have not yet been completed by UN/CEFACT so far, but standardization efforts have been initiated (see Fig. 2). The Standard Business Document Header (SBDH) specification can be considered part of BMA and supports the determination of application based logical routing requirements of business information. The respective specification is fully available, only minor changes are expected in the future. The Core Data Type (CDT) catalogue defines the smallest pieces of information in a business data model. In this way, UN/CEFACT has created an unambiguous basis of atomic business information parts that are used to assemble all higher level parts up to a complete business document. A specification defining 21 Core Data Types and detailed rules for restricting them to meet individual requirements has been made available already. Further standards such as the Business Terms (which are devoted to translating names of Core Components into industry-specific vocabulary) are currently under development.

XML NDR for CCTS and XML Schema for CDT. As a ninth major component, the XML Naming and Design Rules (NDR) defines a set of rules for transforming CCTS based data artifacts into XML Schema and XML based instances. The Schema for CDT represents the tenth part of the UN/CEFACT e-Business stack. These FSV-related standards [4] are available ready for implementation, but are only one of several options for the syntax-specific representation of business data. UN/CEFACT aims at being as technology-independent as possible and thus does not force users to utilize any specific language. For this reason, languages such as BPSS or BPEL can be used for process execution and orchestration of web services, but users are not limited to these.

Registry Implementation Specification. Last, UN/CEFACT also offers details about the architecture of a central registry that users need to search for and identify proper building blocks for business data and process modeling. The basic architectures defined by the ebXML standard may alternatively be used.

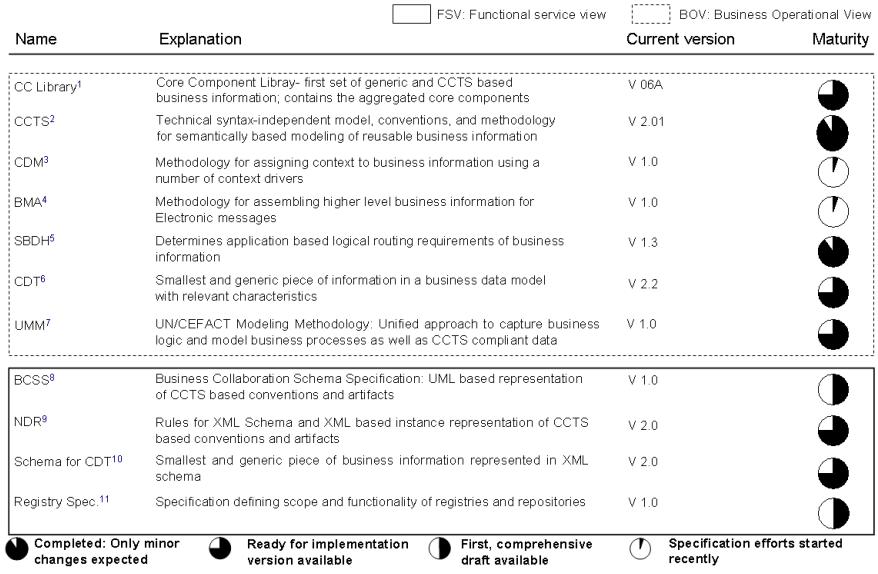


Fig. 2. Maturity of the UN/CEFACT specifications

5 UN/CEFACT Meets SOA: USOA

After elaborating on the potential of SOAs to realize application interoperability and pointing out the remaining challenges such as semantic integration and a common approach for business process modeling, we have presented the emerging UN/CEFACT e-Business stack. As a modular composite of several specifications, it provides a novel approach for modeling both processes and data on

the basis of existing components and templates and thus ensures interoperability. This section provides a step-by-step description of how business logic can be captured on a merely semantic level and how the resulting models can then be interconnected to a SOA as technical foundation. We refer to this integrated approach as *UN/CEFACT Service-Oriented Architecture (USOA)*.

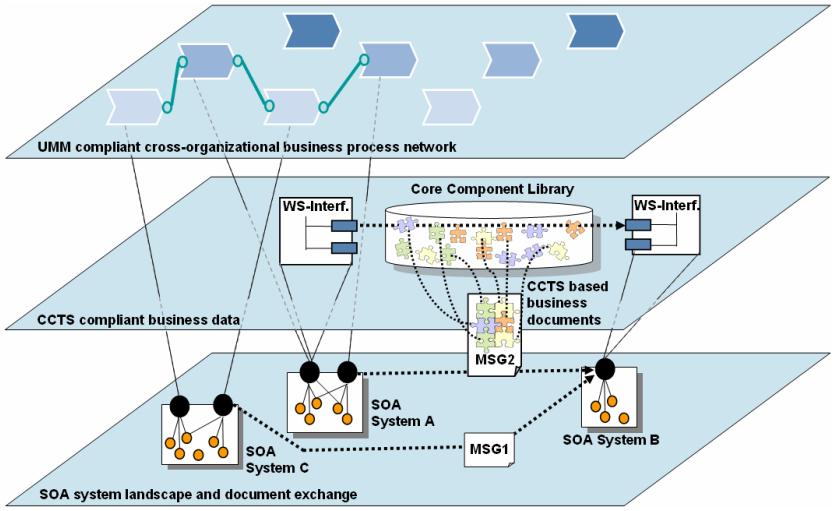


Fig. 3. The Multi-layer USOA Concept

Fig. 3 visualizes an exemplary USOA that consists of three SOA systems (of which each may orchestrate several internal services) publicly offering web services (big bullets in the lower layer) to other applications. To start modeling a cross-organizational collaboration process, our USOA approach envisions a top-down procedure that starts with business experts extracting their business knowledge in a formalized and yet intuitive way by leveraging the UMM described above. Beginning with the Business Domain View (BDV), users fill out pre-determined worksheets that help structuring the respective processes. Second, the Business Requirements View (BRV) is devoted to detailing the models that result from the BDV and provides modelers with existing user roles that can be assembled to certain use cases. The most detailed view (Business Transaction View, BTV) finally builds upon the models from the BRV and provides a precise framework that allows for representing a collaboration process with the help of given *process stereotypes* (which represent minimum process composites) and *tagged values* (for determining performance parameters such as maximum delay times). The resulting Business Transactions and Business Collaboration Protocols are represented in UML based activity charts and can be studied in detail in [19]. Each Business Transaction unambiguously references services offered by the different business partners and thus represents the connection from the merely

semantic collaboration process model to the underlying SOA architecture. This structured top-down approach particularly facilitates the specification of cross-organizational business processes: As opposed to company-internal processes, processes that span across the boundaries of enterprises normally do not have a centralized control instance or process owner. Such cooperations thus require agreements and conventions on how to interact, which are provided by the UMM framework. A SOA only provides encapsulated services as a technical foundation but does not offer common patterns for orchestrating services in an interoperable and widely accepted way. To follow the central UN/CEFACT paradigm of an evolutionary standard that focuses on the changing needs of its users, a publicly accessible repository of conventionally used basic process patterns will be setup. This strongly facilitates the procedure of orchestrating services to a cross-organizational business collaboration (see the upper layer of Fig. 3).

Within the BTV, the UMM also foresees a graphical and intuitive way to model the business documents that are exchanged between the different services: All business partners model data based on a common repository of data artifacts (Core Components) [16] and thus foster a common understanding of the data exchanged between different enterprises. The Core Components are, as previously mentioned, technology- independent and syntax-agnostic templates. As opposed to traditional static business message definitions that inextricably merge technical representation, structure and actual content, the loose coupling of semantic meaning of data components and their syntactical representation allows for using different and even changing underlying technologies. In other words, with the help of the CCTS, the payload of exchanged business messages will be based on a common set of basic data building blocks, where the message envelope can vary depending on the respective user's preferences. The middle layer of Fig. 3 visualizes this concept of assembling business documents out of publicly available and common data Core Components.

The business collaboration model, that has been setup on the semantic level (BOV-layer), can then seamlessly be mapped to a SOA based technical representation (referred to as FSV in [4]). As argued in Chapter 4, users represent business documents by means of context-specific BIEs (created on the basis of the generic Core Components) that reflect their individual business needs. These BIEs can be mapped to various technical representations such as XML Schema as defined by the UN/CEFACT XML Naming and Design Rules (NDR). The interfaces (WSDL compliant) of applications that are part of SOAs then reference documents that are CCTS compliant and thereby bridge the semantic gap that exists in current SOA deployments. Researchers at the University of Vienna [20] also proved that UMM models reflecting collaboration processes can automatically be mapped to process execution languages such as BPEL or BPSS. For each of the UMM artifacts such as Business Transactions and their process stereotypes or the tagged values, a proper equivalent in the BPEL syntax is available. As a consequence, the semantic process models designed by business experts can easily be transformed into executable code that strictly follows a common framework while avoiding any media breaks.

Finally, UN/CEFACT proposes ways to realize a common and also living registry for both the process and the data artifacts [21]. All elements are commonly available and can be amended, extended and even new components can be published in case existing templates are not adequate to reflect actual business requirements. The registry proposed will serve as an advanced service registry (as an alternative or add-on to the conventionally used UDDI) that businesses use to identify and connect to proper trading partners. It is also ideal for maintaining and harmonizing the process and data components in an efficient way.

Summing up, the USOA approach facilitates semantic interoperability on both a process and data level and leverages SOAs as a technical foundation. A collaborative and evolutionary library of semantic data building blocks is made available which are flexibly applicable (as they can be customized according to the respective user contexts with the help of the Context-Driver Methodology (CDM)) and extensible if needed. In addition, the library provides UMM-compliant process components and options to include tagged values as a basis for orchestrating services to a commonly comprehensible choreography. The major requirements for achieving true interoperability in SOAs mentioned in [3] and [5] are thus fulfilled: as a center of competence, the UN/CEFACT fosters collaboration among various stakeholders and documents disagreements rather than forcing a consensus with the help of yet another static standard. The data and process component repositories of UN/CEFACT are considered collaboration rooms that allow people to openly express opinions, resolve and document differences if a consensus cannot be reached. The two publications mentioned above also stress the importance of aggregating all semantic integration efforts into widely recognized standardization organizations. UN/CEFACT goes beyond that and offers a horizontally integrated, universal repository of data and process artifacts that evolves dynamically and perfectly meets the requirements of its users.

6 Conclusion

In this work, we have shown a novel approach for affiliating the emerging UN/-CEFACT e-Business stack (which mainly focuses on the semantic level of business transactions) with web services based SOAs as technical foundation. This integrated approach, that we refer to as USOA, is an appropriate framework for establishing highly flexible and collaborative e-Business platforms by providing business users precise frameworks for defining both processes and data on a merely semantic level. The complexity inherent to proprietary enterprise applications can be encapsulated with the help of commonly comprehensible service interfaces. Exchanged business messages are correctly understood by all applications due to the common and also "living" repository of basic data building blocks. Last, the UMM ensures that cross-organizational processes can be defined in an interoperable way. UN/CEFACT is currently working on the finalization of its standards to provide a next-generation modeling methodology that strongly facilitates cross-organizational application interoperability.

References

1. P. Wainewright. Semantic integration. *http://www.looselycoupled.com*, 2003.
2. R. Sippl. Breaking down the Silo walls - Realizing the vision of SOA. *http://realworldsoa.com/resources/BreakingDownSiloWalls.pdf*, 2006.
3. M. Selvage, D. Wolfson, B. Zurek, and E. Kahan. Achieve semantic interoperability in a SOA. *http://www-128.ibm.com*, 2006.
4. International Organization for Standardization. ISO/IEC IS 14662. *http://www.iso.org*, 2004.
5. R. Schmelzer. Semantic Integration: Loosely Coupling the Meaning of Data. *http://www.zapthink.com*, 2003.
6. G. Stuhec. How to Solve the Business Standards Dilemma - CCTS Key Model Concepts. *Sap Developer Network (SDN), http://sdn.sap.com*, 2006.
7. J. Domingue L. Cabral and et. al. Approaches to semantic web services: an overview and comparisons. *ESWS, ser. Lecture Notes in Computer Science, C. Brussler, J. Davies, D. Fensel, and R. Studer, Eds.*, Vol. 3053, 2004.
8. M. Korotkiy and J. L. Top. OntoSOA: From Ontology-enabled SOA to Service-enabled Ontologies. *Proceedings of International Conference on Internet and Web Application and Services (ICIW'06)*, 2006.
9. P. Frost and S. Frost. *Component-based development for enterprise systems : applying the SELECT perspective*. Cambridge : Cambridge Univ. Press, 1998.
10. G. Alonso, F. Casati, H. Kuno, and Vijay Machiraju. *Web Services Concepts, Architectures and Applications*. Springer, 2004.
11. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, and R. Metz. OASIS - Reference Model for Service Oriented Architecture 1.0. *http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm*, 2006.
12. D. L. Parnas. On the criteria to be used in decomposing systems into modules. *Commun. ACM*, 15(12), 1972.
13. T. Janner, A. Schmidt, C. Schroth, and G. Stuhec. From EDI to UN/CEFACT: An Evolutionary Path Towards a Next Generation e-Business Framework. In *The 5th International Conference on e-Business 2006 (NCEB2006)*, 2006.
14. International Organization for Standardization. ISO 9735. *http://www.iso.org*, 1988.
15. RosettaNet. Trading Partner Directory. *http://www.rosettanet.org*, 2006.
16. UN/CEFACT. Core Component Library Version 06A. *http://www.unece.org/*, 2006.
17. UN/CEFACT. Core Component Technical Specification V2.01. *http://www.untmg.org*, 2003.
18. UN/CEFACT. UMM Foundation Module V1.0 (2005). *http://www.untmg.org*, 2005.
19. B. Hofreiter and C. Huemer. UMM Add-In: A UML Extension for UN/CEFACT's Modeling Methodology. In *European Conference on Model Driven Architecture (ECMDA'06)*, July 2006.
20. B. Hofreiter and C. Huemer. Transforming UMM Business Collaboration Models to BPEL. In *Proceedings of the OTM Workshop on Modeling Inter-Organizational Systems*, 2004.
21. I. Bedini and F. Bourge. Un-cefact - registry specifications v1.0 draft. *http://www.untmg.org*, 2006.