5

Modular architecture framework for cross-organizational electronic interaction

Christoph Schroth and Beat F. Schmid
University of St. Gallen, MCM Institute
Switzerland

1. Introduction

Nowadays governmental authorities as well as companies have to manage a rising number of dynamic inter-organizational business relationships, calling for means that allow for their efficient and effective management (Hagel & Singer, 2008). Gartner Research emphasizes the increased relevance of information technology (IT) in this context: "We expect that by 2011, midsize- and large companies will have at least doubled the number of multienterprise integration and interoperability projects they're managing and will be spending at least 50% more on B2B projects, compared with 2006. We also believe that, from 2008 to 2013, multienterprise traffic will at least triple."(Lheureux & Malinverno, 2008)

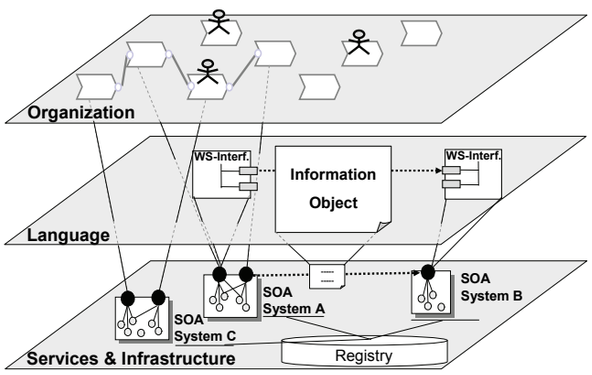


Fig. 1. Service-oriented architecture, according to St. Gallen Media Reference Model (MRM)

During the past years, **Service-Oriented Architectures** (SOAs) have become an acknowledged general architectural style underlying the implementation of cross-organizational electronic interaction (see Figure 1). The widely accepted normative OASIS Reference Model for SOA 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"(Mackenzie et al., 2007). Figure 1 illustrates the fundamental concept of SOA. On the lowest level,

organizations expose dedicated services (black circles) and make them publicly retrievable via certain registry mechanisms. Such services can be composed of a number of other, only internally visible services (symbolized as small gray circles), complying with the principle of information hiding. On the basis of this infrastructural level, a common understanding of the semantics of exchanged messages needs to be ensured. The second level in Figure 1 illustrates the requirement of mutually comprehensible information objects. On the third level, services are orchestrated according to the previously specified structural as well as process-oriented organization.

Despite of the advent of SOA, **the emergence of a huge number of mutually unconnected island solutions could not be prevented**. As illustrated in Figure 2, most business communities rely on a medium (in our context, the term medium refers to any kind of IT infrastructure that enables the interaction of agents) that is designed in a proprietary fashion, both on an organizational, a semantic, and an infrastructural level. In case governmental authorities or companies (referred to as agents in Figure 2) which are connected to different media intend to exchange information, all involved stakeholders face huge challenges. Organizational models, the "languages", as well as infrastructural standards differ significantly (as symbolized by the red flash in Figure 2), and thereby prevent from seamless interoperability.

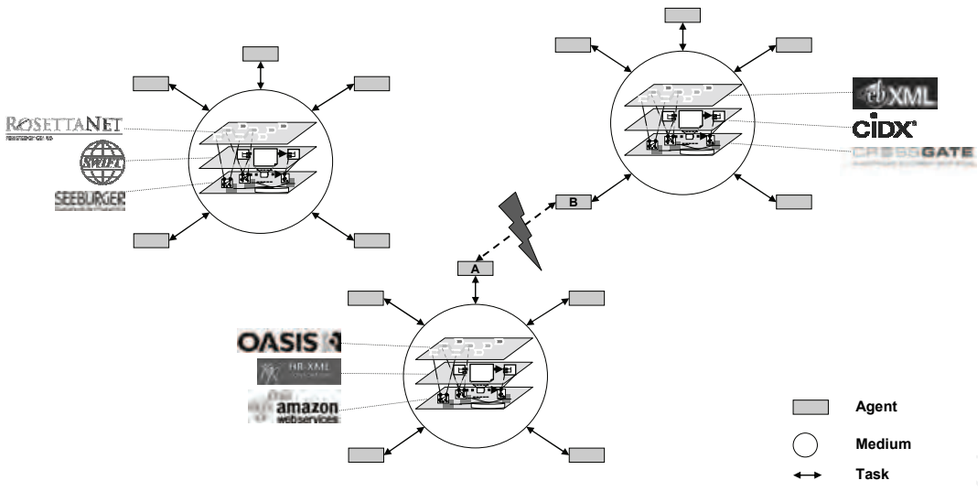


Fig. 2. Challenge: huge number of unconnected island solutions (Schroth & Schmid, 2009)

This article will address these challenges inherent in cross-organizational electronic interaction. Through the identification and establishment of a **commonly acceptable architecture framework**, different organizations and business communities will be enabled to build electronic media which both consider their individual requirements and still allow for seamless interoperability with the media of other business communities. The design rules constituting this architecture framework rely on the basic paradigm of service-orientation and cover all three above mentioned levels: Organization, Language, and Services & Infrastructure. First relevant approaches on all three levels have been published in (Müller et al., 2009) as a comprehensive position paper. The following paragraphs briefly outline some major aspects to be covered by the design rules:

- **Organization**: The types of the interacting agents have to be defined explicitly through the introduction of role descriptions. Also, the procedures of interaction must be defined, be it in a declarative way as rules, or procedurally as traditional processes. Existing methods for modelling cross-organizational interoperation predominantly follow a unitary, process-oriented approach. In this article, we argue for a novel modular method that considers both structural as well as process-oriented organization.
- **Language**: The second "language" level defines the types of the objects of interaction, i.e. the objects, on which the agents act, and which they exchange. Today, experts are forced to understand every syntactic and semantic detail of proprietary application interfaces (based on, e.g., CIDX, HL7, PIDX, SWIFT, etc.). in order to interconnect them. A novel approach is required which provides common design rules for this layer and also proposes a library of modular semantic building blocks that act as common basis for modelling different information objects.
- **Services & Infrastructure**: On the technical level, agents and the services they provide have to be connected physically in order to allow for their interaction. Existing platforms and standards for the implementation of cross-organizational business relationships can mostly be considered as proprietary island solutions. For this reason, we propose an augmented version of the Event-Bus Switzerland standard which allows for the set-up of a federated event bus infrastructure that shall act similar to a cross-organizational operating system (Schroth, Schmid & Müller, 2009).

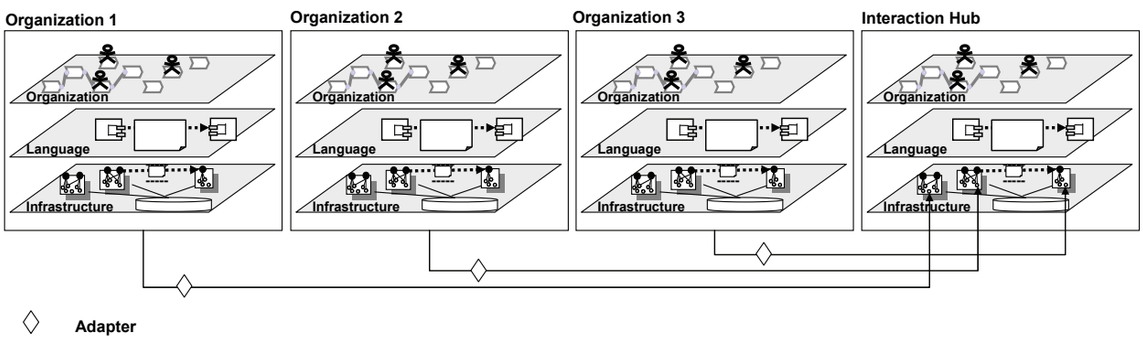


Fig. 3. Goal: seamless cross-medium interoperability

Figure 3. illustrates the focus of this article: Different organizations or business communities may build their respective electronic media (see organizations 1, 2, and 3) to support the interaction of their internal agents (such as individuals, or whole departments). As long as they adhere to a minimal set of architectural design rules, they may also act as agents in another business community (referred to as interaction hub in Figure 3) and mutually interact. The **design principles of modularity and recursivity are key to success**: Each organization must be allowed to encapsulate internal design information from the outside and act as a single agent in arbitrary, other business communities. A Swiss canton, for example, may want to hide internal operations and data from other governmental offices, but still desires to interoperate with other cantons, municipalities, or federal offices. **Adapters are required** for the interconnection between media which follow different design

rules. Such adapters need to intermediate between different technical, but also semantic, and organizational models

The remainder of this article is organized as follows: In section two, existing approaches regarding the design and implementation of cross-company electronic interaction are presented. The systematic analysis of merely organisational approaches, approaches aiming at a common language, technical solutions, service-oriented development methodologies, as well as existing architecture frameworks and reference models builds an adequate foundation for the specification of our research contribution. In section three, a novel modular architecture framework is presented which extends the basic design paradigm of SOA and covers all three above mentioned layers (organization, language, and services & infrastructure). The framework shall provide general design rules which allow for the organization and implementation of both centralized, hierarchical, and decentralized, heterogeneous cross-company business relationships. Through the incorporation of modularity, it shall enable the efficient set-up as well as the redesign of multiple autonomous, yet interconnected groups of companies. On the basis of a case study in the field of public administration in Switzerland (section four), we show the framework's real-world applicability and its improvement potential. Section five closes the work with a brief summary and an outlook on future work.

2. State-of-the-Art

While SOA represents a general architectural style, it does not provide concrete methodologies for the design and implementation of electronic business relationships that span across corporate boundaries. In fact, as depicted in Figure 4, five major clusters of approaches have been identified that help enterprise architects to actually build a cross-organisational SOA:

First, a substantial variety of approaches exists which focus on merely **organizational modelling**. The UN/CEFACT Modelling Methodology (Hofreiter & Huemer, 2003), as well as the Business Process Execution Language (BPEL), or the ebXML Business Processes Specification Schema (BPSS) are examples of practical, widely used standards which are readily available. Significant scientific contributions to the modelling of cross-organisational interaction include, but are not limited to the following examples: Chebbi, Tata, & Dustdar (2006) proposed a meta-model for defining cooperation policies, including partner roles as well as their coordination, data flows, and inter-visibility levels. Jiang, Shao, Qiu, & Li (2008) suggest a timed coloured Petri-net and process-view combined approach to design cross-organizational workflows for Collaborative Product Development (CPD). Gugliotta et al. (2005) proposes a semantic model for the central, business process-based service orchestration. These and related methods are invaluable for capturing and reorganizing the process-oriented (Hammer, 1990), organizational aspects of a given interaction scenario. However, they often do not provide integrated methods or modelling notations for the specification of the structural organization or the information objects exchanged in the course of interaction of companies. The lacking consideration of structural organization entails deficient organizational abstraction and thus flexibility (Schmid et al., 2009). As also argued by other scholars, the principle of information hiding is required to allow for the encapsulation of company-internal design information and also for clearly specified

interfaces between private and public (visible to other organizations) views. Non-modular, comprehensive workflow models, which are based on predefined process logic, offer little support for today's complex and dynamic business environments. Particularly business networks that comprise knowledge-intensive tasks (which may also be subject to strong variations) require novel ways for reducing the complexity and increasing the agility are required. Also, all surveyed approaches act on the assumption that the business processes governing the interaction of one single specific business community need to be specified. They do not foresee possibilities to organize several communities in parallel of which each follows an individual organization but is still interoperable with the other communities. In other words, organizational artefacts available today require the design of each community from scratch and do not support their mutual interoperability.

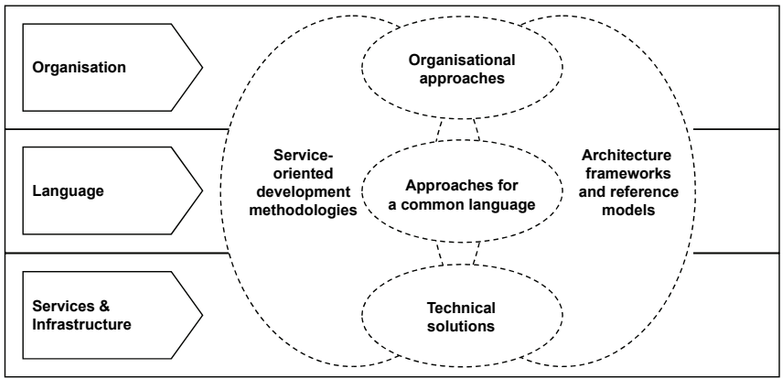


Fig. 4. State-of-the-art overview

The second "**language**" level of analysis concerns the objects of interaction, i.e. the objects, on which the agents act, and which they exchange. In fact, a plethora of different, mostly industry-specific standards exist which enable and at the same time often prevent from cross-domain interoperability. In order to seamlessly interconnect applications which follow different standards across corporate boundaries, huge efforts are usually undertaken. "Mapping experts" and "data consultants" are forced to understand every syntactic and semantic detail of the proprietary application interfaces in order to allow for their connection. BASDA, CIDX, cXML, FpML, HL7, HR-XML, IFX, Open Travel Alliance, papiNet, PIDX, RosettaNet, RSS, SIF, TWIST, UBL, and XBRL represent only a few of the existing data standards in the field of cross-company electronic interaction. These mostly domain-specific, often monolithic standards prevent true interoperability and require huge mapping efforts between the different data fields and their respective representation. A novel approach is thus required which defines commonly acceptable design rules for the flexible structuring of information objects, while leaving a sufficient degree of design freedom for the incorporation of individual requirements. Such a novel approach shall also provide modular semantic building blocks that act as common basis for modelling different information objects. This set of common design rules as well as libraries encompassing information object building blocks are expected to increase cross-organizational interoperability without establishing yet another proprietary standard (Schroth, Pemptroad, & Janner, 2007).

The third level of investigation concerns existing solutions for the **technical implementation** of cross-organisational electronic interaction. Since the wide acknowledgment of the above introduced design paradigm of SOA, the concept of Web services has experienced great interest: The so called Web services stack today represents one of the most accepted means for building a SOA. It provides a system-independent way for interlinking potentially dispersed software applications in a flexible way. Web services can be considered as "plumbing […] for information systems to interact without human involvement" (McAfee, 2005); they provide an invaluable fundament for the uniform description, retrieval, and consumption of heterogeneous capabilities and incorporate the principle of information hiding. However, as also found by the analyst firm Gartner as well as scholars such as Andrew McAfee from Harvard Business School, Web services alone are not sufficient to prevent from the establishment of insular IT service infrastructures for cross-organizational electronic interaction. Service interconnections that span across company borders require electronic business media that implement the structural as well as the process-oriented organization of interaction, define the formats and types of information objects exchanged between the companies and also offer a set of operational services (e.g. for encryption, routing, format adaptation). Since the Web services stack does not provide a comprehensive specification for such IT service infrastructures, groups of companies today frequently build or buy rather proprietary integration platforms which can be considered as island applications.

Dozens of firms have emerged by today that offer integration platforms: Integration-as-a-Service (*IaaS*) providers, for example, offer reliable communication, partner management, technical integration services and application services. Firms such as Amazon have very recently started to provide Web services with business functionality that can be used by developers to implement Web-based business applications which span across company boundaries. In addition, *Amazon* has started offering computing ("Amazon EC2") and storage ("Amazon S3") resources as well as a message-bus platform ("Amazon SQS") that allows for the reliable, secure, and transparent exchange of XML-based messages between distributed business applications. Finally, *Mashup platforms* are considered as tools that empower users to loosely couple readily available pieces of content or functionality, to enrich and compose them into novel services, which can eventually again be made publicly available. The wealth of existing technical e-Business standards (particularly the Web services stack), in combination with novel promising architectural styles such as Event-Driven Architectures (EDAs) represent a valuable technological basis for implementing cross-organizational electronic business relationships. However, particularly in case of large-scale, heterogeneous interaction scenarios which involve different business communities, existing technical solutions are often insufficient. Existing platforms tend to interconnect application interfaces in a proprietary fashion. Registries for information objects, implementations of structural and process-oriented organization, as well as additional operational services (e.g., for encryption, data mapping, identity and access management, or data management) are built according to individual requirements. In other words, today's technical approaches allow for tight application integration rather than loosely coupled interoperation. Many "B2B communities" merely focus on shared, isolated business functionality and are implemented as stand-alone island solutions for specific purposes. Investigations conducted by the international analyst firm Gartner confirm these findings (White, Wilson, & Lheureux, 2007). Potential changes such as the on-boarding of a

new organization into an existing community or the partial interconnection of two or more existing communities requires considerable effort, thereby delimiting operational agility of the involved enterprises.

Apart from the above discussed, rather focused artefacts for organizing and implementing cross-organizational electronic interaction, a number of more holistic, **service-oriented development methodologies** have emerged which explicitly build on the aforementioned SOA paradigm and cover different parts of a software lifecycle such as analysis, design, and implementation. IBM´s Service-Oriented Application Development (SOAD) proposes elements that should be part of a service-oriented analysis and design methodology. SOAD builds upon existing, proven technologies, such as Object-oriented analysis and design (OOAD), Component-based development (CBD), and Business Process Management (BPM). It also introduces SOA-specific techniques, such as service conceptualization, service categorization and aggregation, policies and aspects, and more. This and other methodologies represent first relevant attempts to systematically develop SOAs. The architecture framework proposed in this article will consider the most valuable aspects of existing methodologies and extend these to particularly address the inter-rather rather than the intra-organizational realm where one single homogeneous service architecture can be established.

Finally, existing **architecture frameworks and reference models** need to be examined with regard to their strengths and weaknesses in the context of cross-organizational electronic interaction. The discipline of enterprise architecture treats organizational and technical aspects as two distinct, but complimentary viewpoints on an overall company: According to Lankhorst et al. (2005, p. 3), an enterprise architecture comprises "a coherent whole of principles, methods and models that are used in the design and realization of an enterprise's organizational structure, business processes, information systems, and infrastructure". Enterprise architecture frameworks and reference models are required to align business and information technology in a seamless fashion, to reduce costs for technology through discovery of redundancies, and to have a comprehensive "building plan" available which allows for the systematic design and implementation of information technology. Exemplary frameworks include the Business Engineering Framework (Österle, 1995), the Federal Enterprise Architecture Framework (Lee et al., 1999), and the Department of Defense Architecture Framework (DoD, 2007). These and other, related frameworks represent essential means for managing intra-enterprise architectures as they structure architectures into domain-specific views to reduce inherent complexity. However, many of them can be considered system-centric since they mainly focus on aspects within the boundaries of an enterprise and thus do not necessarily optimise the design or governance of federated environments which need to accommodate heterogeneous requirements. Others already acknowledge the need for federated architectures but do not provide comprehensive methodological means for the decomposition of interaction scenarios and their subsequent assembly.

To sum up: artefacts focusing on merely organizational aspects have been found to frequently lack support for the combined modelling of process-oriented and structural organization. They also exhibit deficient support for organizational abstraction, modularisation, and consequent agility. Finally, they do not provide methodological and notational means for coping with multiple (potentially heterogeneously organized), interconnected business communities. Efforts undertaken for the establishment of a

common "language" have led to a plethora of standards which are highly domain-specific. The existence of such monolithic and proprietary standards, however, prevents from efficient cross-organizational interoperability. From a technical perspective, existing e-Business standards have been analysed. Also, providers of readily available products or services for cross-organizational interoperation - both in the private and the public sector - have been investigated. These have been found to focus on integration rather than loosely coupled interoperation: B2B platforms tend to offer shared business functionality and are implemented as stand-alone island solutions for specific purposes, including proprietary message standards as well as realizations of the process-oriented and structural organization governing their interaction. In case of changes, huge efforts are required to implement the novel organization. Architecture frameworks and reference models encompass both an organizational and a technical viewpoint. However, existing approaches to cross-organizational enterprise architectures are either system-centric as they follow an integrated approach or lack methodological advice for the decomposition, modelling, and subsequent implementation of IT service infrastructures that span across corporate boundaries.

3. A Modular Architecture Framework

In general: MRM, basic structure of the whole thing.

3.1 Physical Component

The physical component of the architecture frame work discussed in this article builds on the two complementary architectural styles of Service-Oriented Architectures (SOAs) and Event-Driven Architectures (EDAs). In specific, it relies on and augments the Event-Bus Switzerland (EBS) specification (Müller, 2007).

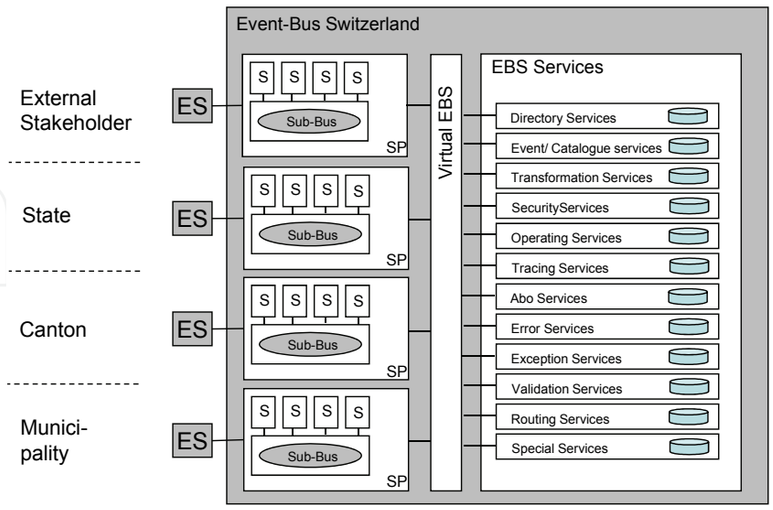


Fig. 5. The basic Event Bus Switzerland architecture (Müller, 2007)

The EBS standard comprises a set of design guidelines for the creation and continuous evolution of a federated system of numerous event buses (referred to as EBS sub-buses;

"Teilbus" in German) which allow for fulfilling heterogeneous, individual requirements and still enable cross-bus interoperability: Figure 5 shows an overview of the basic EBS architecture. The EBS realm comprises several service providers (SPs) who operate the so called sub-buses (TBs). These sub-buses act as electronic media which allow for the seamless interconnection between agents. In Figure 5, these agents are represented by "End-systems", i.e. local and potentially very heterogeneous IT applications. As one of the central design paradigms inherent in the EBS concept, all these sub-buses need to adhere to certain design rules in order to be interoperable. In that case (all sub-buses comply with these rules), agents which are connected to different sub-buses can still collaborate across sub-bus boundaries. The Event-Bus Switzerland can thus be considered as virtual concept comprising design rules but does not represent a physical medium itself. In fact, the EBS delineates infrastructural services such as routing services, error services and directory services (see Figure 5). As long as all sub-buses adhere to these service specifications, interoperability is ensured.

Figure 6 illustrates the most central components of our extended specification (Schroth, Schmid, & Müller, 2009): Rather than implementing the interaction between a set of agents based on one electronic medium, interaction scenarios are decomposed into so called interaction modules (IAMs). For each of those modules, a sub-bus is realized (in this example, a first module IAM0 comprises medium M1 which enables the communicative exchange of agents 1, 2, and 3) and implements a number of services:

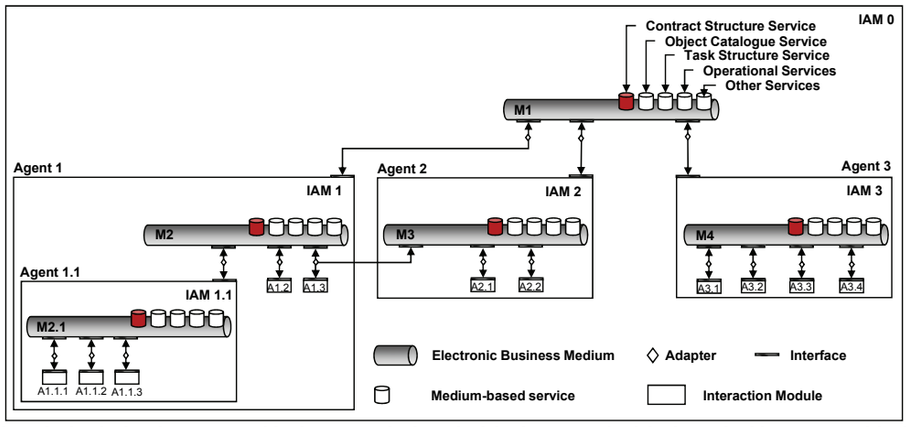


Fig. 6. Modular system of bus media (Schroth, Schmid, & Müller, 2009)

The *contract structure service* implements the structural organization within each bus. It specifies the agents connected to the bus, their roles, and the tasks they are authorized to perform within their respective IAM. The *task structure service* implements the process-oriented organization established within the respective IAM. For each of the tasks that can be performed by defined agents within an interaction module, this service documents precedence relationships to other tasks. It also specifies the mutual relationships between tasks as they constitute one-way or two-way interaction patterns (this organizational specificity will be explained in section 3.3). The *object catalogue service* specifies all the information object schemata which may be exchanged via the bus medium. *Operational services* (e.g. encryption, decryption, routing) assume operating system functionality and are

well described in (Müller, 2007). The recursivity inherent in this design allows for information hiding where required and thus supports decoupling: Agent 1, for example, may assume a defined role within IAM0, while it encapsulates the interaction of a number of sub-ordinate agents (A1.1, A1.2, A1.3) who interact via a hidden medium. As long as the general design rules (based on the above outlined services) are considered when designing a modular system of such sub-buses, cross-bus interoperability and efficient redesign is facilitated. The following example illustrates the flexibility of this design: In case a service provided by a so far hidden agent (e.g., A1.1.1) shall be made available to a number of other agents (connected to different bus media), for example, all design information required for its consumption can be "propagated upwards" through the design hierarchy (by means of updating the above mentioned services). The red data base symbols in Figure 6 represent the contract structure services of the 4 depicted bus media. In order to "publish" a service provided by agent A.1.1.1 to other agents (for example those connected to medium M4), the respective data base entries in the M2.1-based contract structure service can easily be transferred to the M4-based service as they obey to the same design rules.

3.2 Logical Component

The architecture framework propose a modular, core-component-based modelling approach which augments emerging standards such as the OASIS Universal Business Language (UBL), the UN/CEFACT Core Component Technical Specification (CCTS), and, on a technical level, the W3C XML schema (see Figure 7). The approach is new as it spans the bridge between unstructured modelling of data and core-component-based, formal representations and also because it integrates contextual information in order to allow for deriving tailor-made business information documents from generic information object classes. The resulting modelling approach ranges from (tool-supported) graphical data models to the technical representation of the business documents such as XML schema documents designed in compliance with the UN/CEFACT XML schema Naming and Design Rules (NDR).

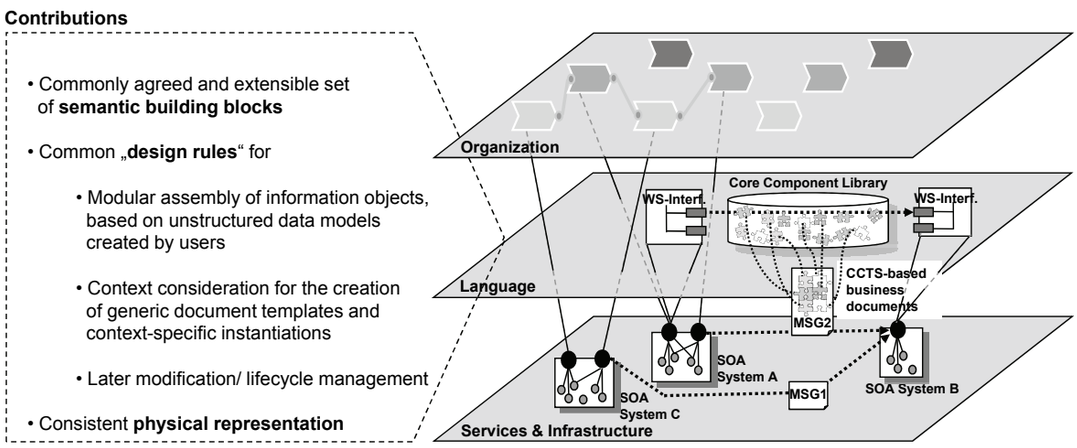


Fig. 7. Modular design of information objects

Four abstract entities constitute the core of the information object modelling approach: First, generic core components are employed as reusable building blocks for the design and

assembly of comprehensive generic business documents. The CCTS methodology proposes the four core component types Core Data Type (CDT), Basic Core Component (BCC), Association Core Component (ASCC), and Aggregate Core Component (ACC). Generic document descriptions (see the rectangle on the left in Figure 8) encapsulate the organization of whole documents such as order or invoice documents. They can be compared with classes in the software programming context as they can be instantiated several times in different contexts. The instantiations of generic business documents are referred to as specific business documents. Such specific business documents are constituted of specific core components, i.e. the context-specific instantiations of their generic counterparts, the generic core components. The mechanism by which specific documents and core components are derived corresponds to the mechanism of "restriction inheritance". Only those information object constituents are selected that are relevant in a given context (see the three context-specific instantiations on the right side of the figure). Our framework augments the above mentioned standards as it provides a guided procedure for the graphical modelling of unstructured data and its subsequent transformation into standard-compliant data components, as it introduces a comprehensive methodology for the incorporation and management of contextual information, and as it proposes an XML schema-based representation of generic business documents, including context parameters (Schroth, Pemptroad, & Janner, 2007).

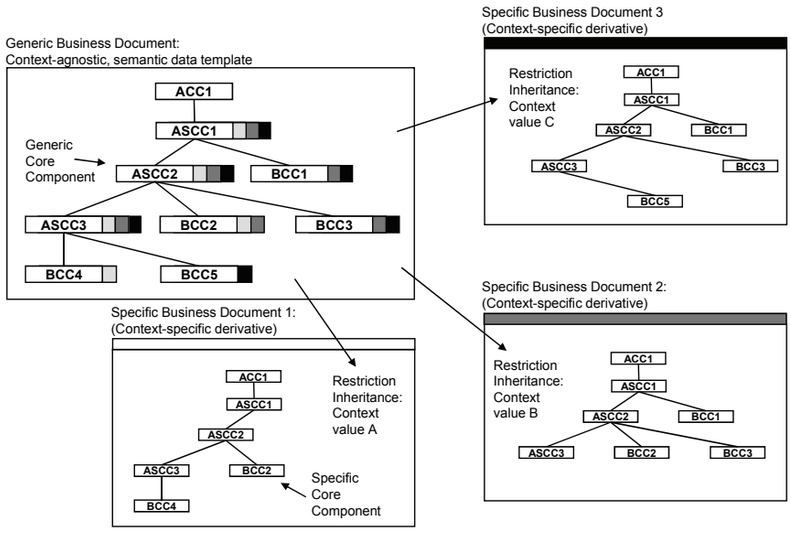


Fig. 8. Context-specific instantiations of a generic business document (Schroth, Schmid, & Müller, 2009)

This information object design approach allows for increased reuse and thus productivity through the establishment of a common library of standardized data building blocks. It also improves modifiability of information object representations (and thus also service interfaces) since components of an object model can be easily augmented, excluded, split, or substituted. This differs from conventional, rather monolithic approaches since information objects do not require a holistic redesign in case changes are required (Janner et al., 2008).

3.3 Organizational Component

To achieve a modular organization of electronic interaction, an interaction scenario first of all has to be decomposed into its constituent, fine-granular tasks (business activities, performed by agents, defined as operations related to specific information objects).

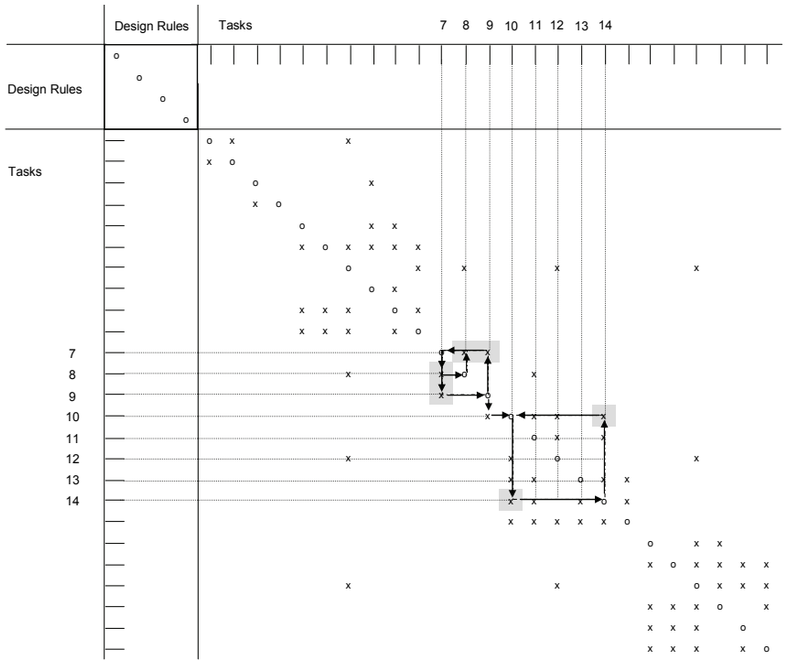


Fig. 9. Capturing interaction patterns from the structural and process-oriented organization

The identified tasks (lowest level of organizational abstraction) are then assigned to both the x- and the y-axis of a task structure matrix (Baldwin & Clark, 1999). In case task i precedes task j, a mark (x) is put in column i and row j of the matrix to document precedence relationships between the various tasks (see Figure 9). From the marks, we can then identify one-way and two-way interaction patterns (IAPs) which represent the second level of organizational abstraction. IAPs encompass two tasks and feature parameters for detailed behavioural modelling (Schroth & Schmid, 2009). An IAP comprehensively addresses a dedicated unit of interaction among agents (who assume roles): It can be instantiated as either one-way or two-way pattern: In the former case, the modelled pattern comprises the transmission of an information object from a sending to a receiving role ("one-way"). In the latter case, an information object is sent from one role to another one who is required to respond in a clearly specified manner ("two-way").

Figure 9 illustrates the identification of three two-way and a single one-way IAP: First, task 7 needs to precede task 8. In other words, the information object associated with task 7 represents a necessary input for task 8. Task 8, in turn, is a direct response to task 7. This request/ response relationship can be modelled as two-way IAP: One mark is entered into row 8 and column 7, while another one is entered into row 7 and column 8. In order to make the interrelationship of the two marks visible, the respective cells in the matrix are coloured gray. In this exemplary case, task 8 is not the only possible successor of task 7. Instead, task 9

may be executed alternatively. As a consequence, a second two-way IAP can be identified (marks in row 9, column seven, as well as in row 7, column 9). Tasks 9 and 10, on the other hand, are governed by a one-way IAP: A single mark is entered into row 10, column 9. As a final example, tasks 10 and 14 can again be described with the help of a two-way IAP (see Figure 9).

The resulting fields within the matrix which feature an increased amount of marks mean highly interdependent groups of business activities and thus suggest the specification of the afore mentioned interaction modules (IAMs) that reside at the third level of abstraction (see Figure 10). As few as possible interdependencies shall exist between the tasks comprised by different IAMs (indicated by off-diagonal xs which are not included in one of the IAMs). These interdependencies either need to be made explicit and become the basis for the development of interfaces (relying on context-specific, descriptive design rules (CSDR)) or can be removed through the definition of prescriptive design rules. The organizational design is completed by defining a design hierarchy diagram that clearly specifies the nested hierarchy and the inheritance relationships between the modules (Figure 10 corresponds to Figure 6).

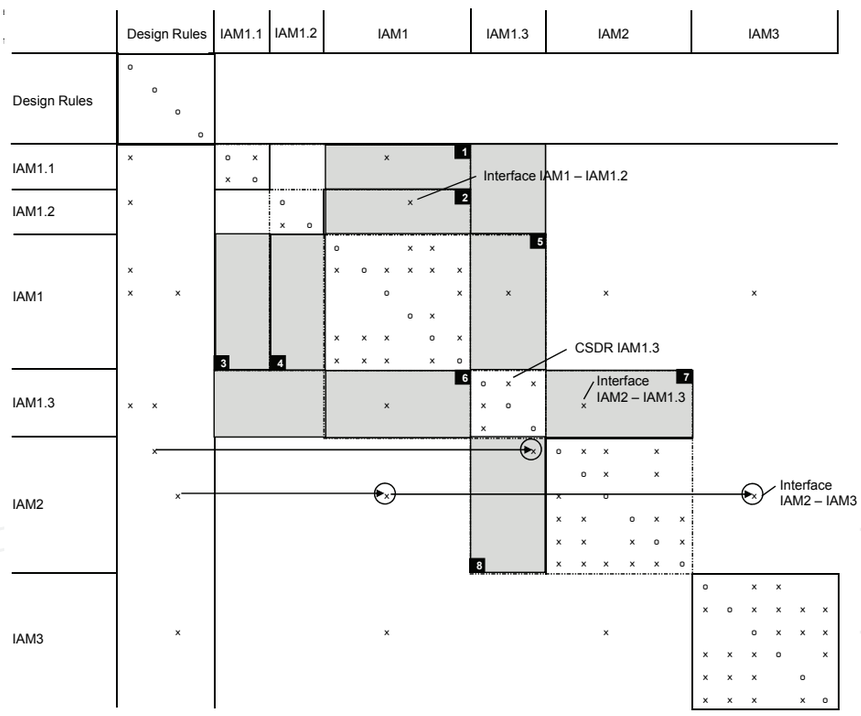


Fig. 10. Organisational modularisation into IAMs (Schroth, Schmid, & Müller, 2009)

4. Case Study: Applying the Framework in the Swiss Public Sector

4.1 As-is-Situation

In this section, we elaborate on a case study that was conducted in the course of the government-funded project HERA (HERA, 2009). The project aimed at the improvement of the tax declaration procedure in the federal system in Switzerland. It serves as example for

the interaction of defined stakeholders who electronically interact to achieve a common goal: As depicted in Figure 11, there are mainly four stakeholders involved in the cross-organizational process of creating a tax declaration. First, a company (also referred to as JP) itself submits a tax declaration that complies with laws, is consistent with the forms issued by the various cantons (Swiss states) and is optimised with respect to the resulting tax load in an as efficient way as possible. Accountants can either be represented as company-internal departments or external service providers. They create comprehensive financial statements and also provide consulting services with respect to profit appropriation strategies. Auditors have to be organizationally separated from accountants (by law) to ensure their independency. They examine and verify compliance of financial statements and profit appropriations. Finally, the cantons (states) receive the completed tax declaration and initiate the assessment/ enactment process.

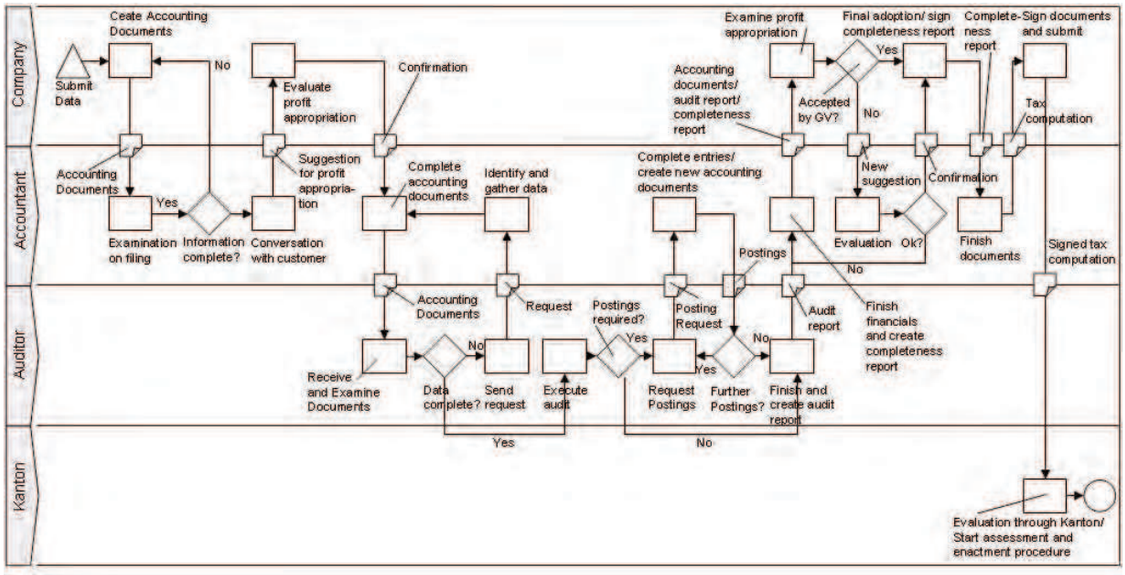


Fig. 11. Simplified tax declaration business process (Schroth & Schmid, 2009, p.86)

Municipalities play a certain role within the tax declaration process in some of the Swiss cantons, but are left out in this work due to space constraints. Also, the visualized, cross-organizational business process represents a cut-out (which is valid in the canton of St. Gallen) of the full process with all its canton-specific deviations. During this procedure of creating a tax computation, the division of labour among the players induces the need for coordination and information exchange between them which follows certain process choreographies. As a consequence, numerous documents (as visualized in the graphic) are passed from one stakeholder to the other and are thereby processed in different ways until they reach the end of their respective "lifecycles".

Today, all stakeholders depicted in Figure 11 interact with each other via different communication channels. Some information is exchanged in paper format; other documents are transferred via e-Mail or proprietary electronic interfaces. Resulting media breaks, the lack of standardized interfaces and the strong involvement of humans into information processing induces high transaction costs and increases the risk of errors, thereby limiting

service quality. Also, services are only rarely subject to quantifiable performance metrics. The study has shown that especially non-functional properties of services such as delivered quality or exact time required for completion are usually not provided in a clear, formal and quantifiable way. The heterogeneity of used media prevents from standardization with respect to terminology, processes, pieces of information, and therefore deteriorates the productivity of seamless collaboration across the stakeholders' boundaries. Frequently, decisions have been found to be made on the basis of best practices instead of formalized rule sets. Also, the cross-organizational process-oriented organization strongly varies from case to case, depending on a number of parameters. The concerned canton's legislation, the individual stakeholders and their particular preferences, the exact partner constellation (is a separate accountant involved or is the company in charge of accounting activities), context-dependent factors and the quality of exchanged information (this may cause iterative, additional claims for documents) represent only some of the factors influencing the exact process organization.

4.2 Modular Reorganization and Evaluation

Following a traditional, non-modular approach (see Figure 11), designers would try to comprehend the situation as a whole, model "one", fixed business process governing the interaction and implement an electronic business medium to avoid media breaks and partially automate the interaction. However, as also argued by Gartner (Lheureux & Malinverno, 2006), the design of another highly proprietary, inflexible and thus inefficient island solution would not add sustained value. In fact, the above described approach resembles the "Spaghetti-code era" which abetted the software crisis in the 1970s. The resulting business medium would require huge effort for design and creation (as it was not modular, thus complicating concurrent work or redesign), could only hardly be changed and could not be seamlessly connected to other business media as it would not follow any global design rules. For this reason, an electronic business medium (**the HERA platform**) has been designed and implemented, based on the modular architecture framework presented above.

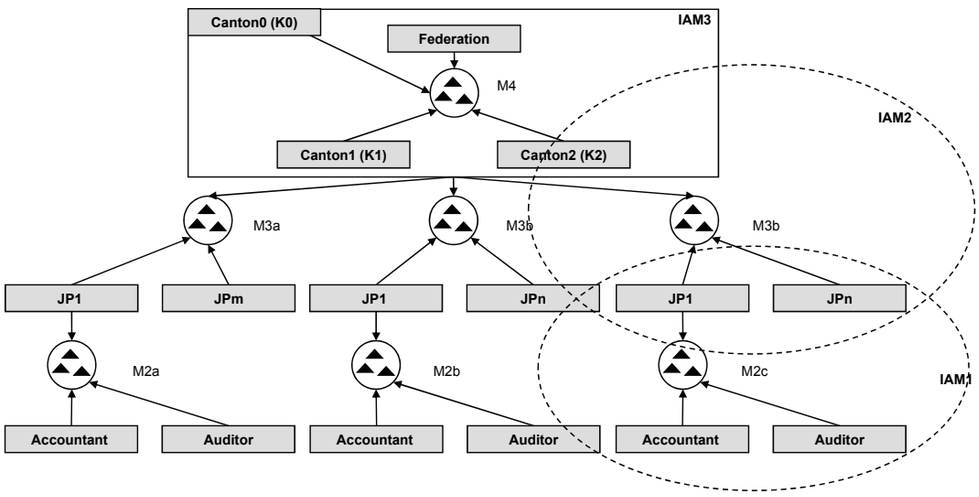


Fig. 12. System of media and agents, organized in interaction modules (IAMs)

As a first step towards our modular design, the overall interaction scenario to be structured (decomposed) into fine-granular, atomic activities. Secondly, we assign each of the tasks specific roles to define the agents who are allowed to perform these. As a third and final step, the diverse sub-tasks shall be decoupled by defining mutually independent, organizational interaction modules (IAMs) as argued above. In our case, three generic modules could be identified: the first one concerns the specific interactions between companies, accountants and auditors; the second one exclusively comprises the interactions between companies and the cantonal tax offices; the third and final one focuses on the interaction between the governmental tax offices. Benefits resulting from this task modularisation include: first of all, responsibilities for tasks and related information (data access rights) can be clearly separated and limited to those roles which are explicitly involved in a certain module. Secondly, operational agility and manageability can be improved: In case of modifications (e.g., required by legal changes), the modules can be re-organized without affecting other modules. The IAMs were defined based on the task structure matrix-based methodology discussed above. Due to length restrictions, the detailed derivation of the three IAMs is not provided here.

Figure 12 illustrates the resulting organization of decentrally operated interaction modules: the module "governmental interaction" is instantiated once (IAM3): it allows cantonal authorities to exclusively exchange data in order to define the share of the tax load as described above. All the internal interaction between different cantonal tax offices can thus be hidden from the outside in order to reduce operational complexity. The assessment/ enactment module (IAM2) is supposed to be instantiated once per canton to account for their individual needs with respect to data formats, business processes and other organizational artefacts. In other words, each canton may establish an individual assessment/ enactment interaction module which encompasses all the tasks dealing with submitting a tax declaration and the subsequent assessment as well as enactment procedures. Independent from these modules, the accounting/ auditing interaction modules (IAM1) can be instantiated. On the basis of the HERA business medium, each company shall be enabled to establish an individual structural and process-oriented organization governing the interaction between itself and external accountants and auditors. The independence of this module can be emphasized with the following example: Companies may submit their tax declaration via the HERA business medium without having used HERA for accounting/ auditing purposes before. The two modules can be considered fully independent and may thus be reorganized autonomously. However, to ensure interoperability and fast exchangeability, all interaction modules follow a set of common design rules and provide clear interfaces to the outside.

In order to **implement** these mutually independent yet interoperable interaction modules, the HERA platform (HERA bus) has been developed which augments the above mentioned Swiss governmental initiative "Event Bus Switzerland (EBS)" (Figure 13): first, in order to physically realize the interaction of agents, a bus medium has been proposed which features a set of operational services: Abonnement services (supporting Publish/subscribe message dissemination), directory services (allowing for publishing and retrieving business partners and their respective profiles), event catalogue services (documenting all messages which may be disseminated via the bus including the agent roles which may send/receive them),

transformation services (accounting for mediation of electronic artefacts which adhere to different format standards), security services (encryption and decryption), operating services (for media administration purposes), error services (automatic failure detection and removal), routing services, and validation services (e.g., for evaluation of correctness and integrity of exchanged information. Agents (individuals or software applications) are connected to the HERA bus via defined interfaces describing the events they are authorized to send and to receive. If agents do not obey to the design rules established as part of the HERA bus specification, adapter modules are required. Within the HERA bus, additional coordination services (e.g., completeness control, process visibility and due date monitoring) have been deployed which do not only enable reliable message transport but also interpret and react upon message content. In addition, a Process Server service, a Document Management System (DMS) service as well as an Identity & Access Management (IAM) service have been deployed. The process server service stores the structural and process-oriented organisation for each of the interaction modules. In other words, it ensures that interaction patterns (IAPs) are only used by authorized agents, and that precedence relationships between the tasks are considered. The HERA platform does not only foresee agile interoperability within the sphere of one "business community" and its business medium, but also allows for loosely coupling of several buses which again may connect diverse agents. For cross-medium interoperability, each bus can incorporate an individual service design as long as it adheres to minimal "global design rules" which require the implementation of a standardized directory service, an event-catalogue service and the conformance to a specific message envelope standard (Müller, 2007).

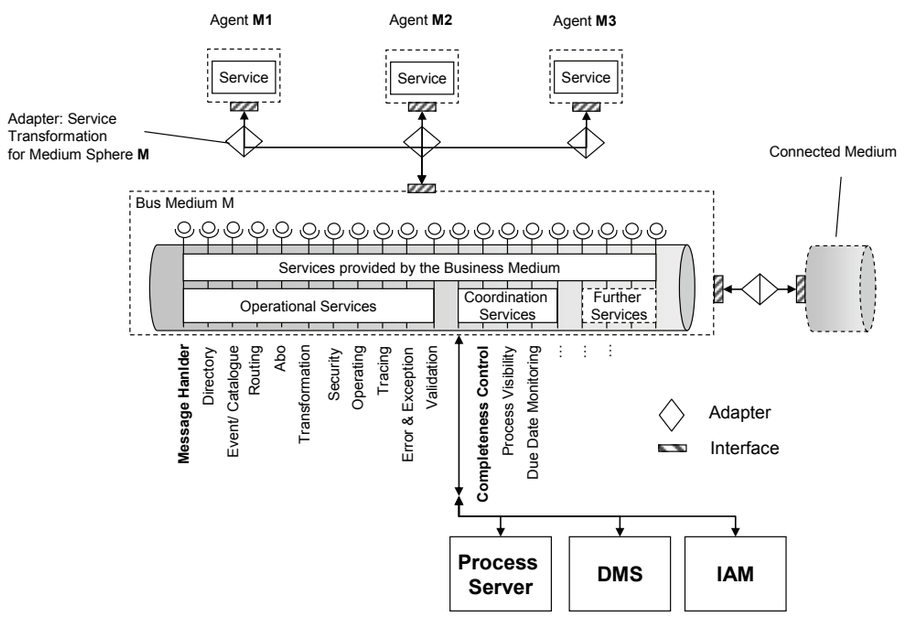


Fig. 13. Technical view: the HERA-bus and its core constituents (Streit et al., 2009)

5. Conclusion and Outlook

In this work, we analysed organisational and technical weaknesses inherent in existing approaches to support the electronic interaction across corporate boundaries. To cope with these challenges, we presented a modular architecture framework and employed it to the scenario of collaborative tax declaration in Switzerland to illustrate its real-word applicability. In this way, we proved that both the physical medium's design as well as the organization of agent interaction could be truly modularised.

As one of the next steps, the economic potential of the HERA platform has to be investigated thoroughly in order to exactly quantify the advantages of the architecture framework compared to existing, more monolithic designs. In (Schmid, Schroth, Miche, & Janner, 2009), we proposed an initial architecture valuation method which tries to capture the value of modular designs. With the help of expert workshops conducted in the course of the HERA project, we identified interoperability, agility, and data security as the major business drivers underlying the analysis of architectural benefits. Based on fine-granular quality attributes and associated scenarios, we were able to estimate the economic value of organizing and implementing electronic interaction based on our architecture framework. One of the key insights gained was: The systematic splitting apart of cross-organizational interaction scenarios as well as their underlying information technology into modules with clearly defined interfaces allows for an unprecedented degree of agility. Modularity accommodates uncertainty and multiplies design options, thus creating a "portfolio of options" rather than an "option on a portfolio" (Baldwin & Clark, 1999). In modular designs, options can be leveraged through applying one or more of the modular operators discussed above. In the case of systems supporting cross-organizational electronic interaction between agents, for example, agent or media modules can be added, excluded, split, substituted, inverted and ported. These operators are applied entailing costs and benefits different from those in case of monolithic systems.

Future publications will deal with leveraging the fundamental insights gained in the financial sector regarding option pricing for the development of quantitative theoretical valuation framework for modular IT service infrastructures. As opposed to the simple, Net Present Value (NPV)-based valuation techniques often used today, a model considering the availability of design options is now needed to exactly capture the benefit of modular architectures (Banerjee & deWeck, 2009; Feurstein, & Natter, 1998; Yeo, Qiu, 2002).

6. References

Baldwin, C. Y., Clark, K. B. (1999). *Design Rules: The Power of Modularity Volume 1*, MIT Press, Cambridge, Massachusetts.

Banerjee, P., deWeck, O. L. (2009). Flexibility Strategy – Valuing Flexible Product Options, available online at: http://esd.mit.edu/HeadLine/banerjee_downen_presenting/banerjee_paper.pdf

Chebbi, I., Dustdar, S., & Tata, S. (2006). The view-based approach to dynamic inter-organizational workflow cooperation. *Data & Knowledge Engineering*, 56(2), 139 - 173.

DoD. (2007). DoD Architectural Framwork (DoDAF) Version 1.5. *Volume I: Definitions and Guidelines*: United States Department of Defense (U.S. DoD).

Feurstein, M., Natter, M. (1998). Valuing the flexibility of flexible manufacturing systems with fast decision rules. *Springer Lecture Notes in Computer Science*, Vol. 1416, pp. 153-162

Gugliotta, A. C. L., Domingue, J., Roberto, V., Rowlatt, M., Davies, R. (2005). A Semantic Web Service-based Architecture for the Interoperability of E-government Services. Paper presented at the Workshop on Web Information Systems Modeling Workshop (WISM 2005) in conjunction with the *5th International Conference on Web Engineering (ICWE 2005)*, Sydney, Australia.

Hagel, J., Singer, M. (2000). Unbundling the Corporation. *The McKinsey Quarterly*.

Hammer, M. (1990). Reegineering Work: Don't Automate, Obliterate. *Harvard Business Review*, 68 (4), pp. 104-112.

HERA (2009). *Helvetic e-Government Reference Architecture*. http://www.hera-project.ch

Hofreiter, B., & Huemer, C. (2003). Modeling business collaborations in context. *Lecture Notes in Computer Science*, pp. 829-844, Springer.

Janner, T., Lampathaki, F., Hoyer, V., Mouzakitis, S., Charalabidis, Y., Schroth, C. (2008). A Core Component-based Modelling Approach for Achieving e-Business Semantics Interoperability, *Journal of Theoretical and Applied Electronic Commerce Research (JTAER)*, Vol. 3, No. 3, p. 1-16.

Jiang, P., Shao, X., Qiu, H., & Li, P. (2008). Interoperability of Cross-organizational Workflows based on Process-view for Collaborative Product Development. *Concurrent Engineering*, 16(1), 73-87.

Lankhorst, M. et. al. (2005). *Enterprise Architecture at Work: Modelling, Communication, and Analysis*. Heidelberg, Germany: Springer.

Lee, D., Flyzik, J. J., Langston, M., Holcolmb, L., Thomas II, R. T., & Tiemann, M. A. (1999). *Federal Enterprise Architecture Framework Version 1.1*: The Chief Information Officers Council.

Lheureux, B. J., Malinverno, P. (2008). Market Update for Integration Service Providers. *Gartner Research Paper*.

Lheureux, B. J., & Malinverno, P. (2006). Magic Quadrant for Integration Service Providers, 1Q06. *Gartner Research Paper*.

Mackenzie, M. et al. (2007). OASIS - Reference Model for Service Oriented Architecture 1.0: OASIS.

McAfee, A. (2005). Will Web Services Really Transform Collaboration. *MIT Sloan Management Review*, 46(2), 78-84.

Müller, W., Schmid, B. F., Schroth, C., Janner, T. (2009). IT Architecture in the Swiss Public Sector: General Design Rules: A Position Paper, available online at: http://www.alexandria.unisg.ch/

Müller, W. (2007). Event Bus Schweiz. Konzept und Architektur, Version 1.5. Bern, Switzerland, Informatikstrategieorgan Bund (ISB)

Österle, H. (1995). *Business Engineering 1 Prozess- und Systementwicklung*. Berlin, Germany: Springer.

Parnas, D.L. (1972). On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15 (12), 1053 – 1058.

Schmid, B. F., Schroth, C., Miche, M., Janner, T. (2009). Valuating Modular Architectures for Cross-Company Electronic Interaction. *Communications of the IBIMA*, Vol. 9, pp.128-140, ISSN: 1943-7765

Schroth, C., Schmid, B. F., Müller, W. (2009). Designing Modular Architectures for Cross-Organizational Electronic Interaction. *Springer Lecture Notes in Business Information Processing (LNBIP)*, Vol. 20, ISBN:978-3-642-01111-5

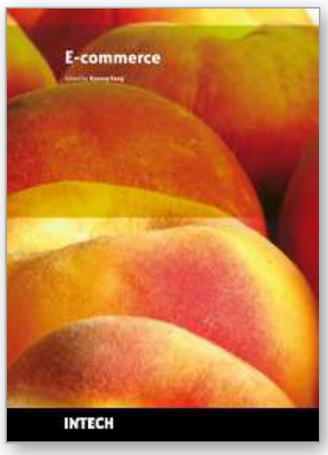
Schroth, C., Schmid, B. F. (2009). Reference Architecture for Cross-Company Electronic Collaboration, *International Journal of e-Collaboration*, 5 (2), pp. 75-91.

Schroth, C., Pemptroad, G., & Janner, T. (2007). CCTS-based Business Information Modelling for Increasing Cross-Organizational Interoperability. In R. J. Gonçalves, J. Müller, K. Mertins & M. Zelm (Eds.), *Enterprise Interoperability II*. New Challenges and Approaches. London: Springer.

Streit, S., Heck, U., Reimer, U., Schroth, C., Janner, T., Collm, A., Hristova, R., Ritsch, R. (2009). Towards cross-organisational e-Government: an integrated approach. In : *Proceedings of the 9. Internationale Tagung Wirtschaftsinformatik Business Services: Konzepte, Technologien, Anwendungen*.Vienna, Austria.

White, A., Wilson, D., & Lheureux, B. J. (2007). The Emergence of the Multienterprise Business Process Platform. USA: *Gartner Research Paper*.

Yeo, K.T., Qiu, F. (2002). The value of management flexibility—a real option approach to investment evaluation. *International Journal of Project Management*, Vol. 21, No.4



E-commerce

Edited by Kyeong Kang

ISBN 978-953-7619-98-5
Hard cover, 284 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

E-commerce provides immense capability for connectivity through buying and selling activities all over the world. During the last two decades new concepts of business have evolved due to popularity of the Internet, providing new business opportunities for commercial organisations and they are being further influenced by user activities of newer applications of the Internet. Business transactions are made possible through a combination of secure data processing, networking technologies and interactivity functions. Business models are also subjected to continuous external forces of technological evolution, innovative solutions derived through competition, creation of legal boundaries through legislation and social change. The main purpose of this book is to provide the reader with a familiarity of the web based e- commerce environment and position them to deal confidently with a competitive global business environment. The book contains a numbers of case studies providing the reader with different perspectives in interface design, technology usage, quality measurement and performance aspects of developing web-based e-commerce.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Christoph Schroth and Beat F. Schmid (2010). Modular Architecture Framework for Crossorganizational Electronic Interaction, E-commerce, Kyeong Kang (Ed.), ISBN: 978-953-7619-98-5, InTech, Available from: http://www.intechopen.com/books/e-commerce/modular-architecture-framework-for-crossorganizational-electronic-interaction

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen