ASSESSING THE POTENTIAL IMPACT OF WEB SERVICES ON BUSINESS PROCESSES

Jan-Hendrik Sewing[†], Michael Rosemann[‡], Marlon Dumas[‡] and Marcello La Rosa[‡]

[†] Siemens Management Consulting

St.-Martin-Straße 76

81541 Munich, Germany

jan-hendrik.sewing@siemens.com

[‡] Business Process Management Group

Queensland University of Technology

GPO Box 2434

Brisbane QLD 4001, Australia

{m.rosemann, m.dumas, m.larosa}@qut.edu.au

ABSTRACT

Though web services offer unique opportunities for the design of new business processes, the assessment of the potential impact of Web services on existing business information systems is often reduced to technical aspects. This paper proposes a four-phase methodology which facilitates the evaluation of the potential use of Web services on business information systems both from a technical and from a strategic viewpoint. It is based on business process models, which are used to frame the adoption and deployment of Web services and to assess their impact on existing business processes. The application of this methodology is described using a procurement scenario.

Keywords: web services, service-oriented architecture, business process management, business processes modelling, information technology adoption.

INTRODUCTION

Web services (WS) is an emerging set of technologies that aims at facilitating the flexible and standardised implementation of interoperable software systems. Considerably hyped in recent years, Web services are expected to ease many current IT problems, such as the large-scale integration of heterogeneous software applications or the cost-effective establishment of e-business interactions. From a more technical viewpoint, investment in Web services is seen as a prerequisite to adopt a Service-oriented Architecture, an IT systems architecture paradigm that uses the concept of service as a basis for managing inter-connected software applications.

Although the intensity of development efforts and standardisation activities is very high, systematic approaches to assess the actual impact of Web services on existing business information systems are still missing. Thus, many organisations are struggling to assess the real impact of Web services and the accompanying opportunities and threats. Without appropriate business alignment, Web services might be perceived as a technical solution without a clear value proposition, in the sense that their potential benefits might not justify associated software reengineering efforts. This constitutes a potential risk factor in light of current IT spending practice and could eventually hamper a wider adoption.

Addressing the alignment of Web services to business priorities is therefore a critical step towards the success of this emerging technology: it will determine whether Web services can fit into (and more importantly improve) existing business practices and thus increase the competitiveness of the organisations that adopt them.

Business process modelling encapsulates all forms of graphical visualization and structured documentation of business processes and related elements such as events, data, material flows and external interactions. Business process modelling may be conducted for a wide variety of purposes including among others process documentation, process improvement, compliance, software implementation, or quality certification (Curtis et al., 1992; Becker et al., 2000). It is an established approach for analysing and improving existing business processes. Business process models, extended with relevant information, have the potential to serve as a decision support instrument for assessing the potential of Web services. They are able to show the process context and ways of how Web services can enable business process innovation.

This paper proposes a methodology for identifying and assessing opportunities for introducing Web services into existing business information systems by means of business process modelling. After briefly outlining and justifying the research approach, a framework is presented for selecting the most appropriate processes for potential incorporation of Web services. Following this, information domains and types are identified that need to be contained in a business process model to support systematic Web services assessments and to facilitate Web services deployment. This information is then mapped into a specific representation in the context of the ARIS Toolset (Scheer, 1998a), a widely used solution for business process modelling. This mapping as well as the conceptual possibilities of the methodology are then illustrated through an example from the area of e-procurement. Finally, related work, conclusions and directions for future work are discussed.

RESEARCH APPROACH

The proposed assessment methodology is grounded in related literature and complemented by focus group discussions with early and prospective Web services adopters. The purpose of the focus groups was to explore the current practice of web service implementations, and industry’s perception and approaches on how to address the challenge of business alignment. Specifically, two focus groups were organised: one for discussing the uptake and adoption of Web services technologies and a second one for discussing the use of business process models for assessing Web services adoption opportunities.

The participants of the focus groups were selected on the basis of their experience with Web services or their affiliation to organisations that were considering the deployment of Web services. The choice of participants was also guided by the objective of covering different organisations and industry sectors, and striking a balance between participants with a technical and a management background. Overall, the focus groups included 15 participants from 8 organisations and covering 4 different groups (IT users, vendors, consulting firms, and research).

The reason for choosing focus groups as the empirical basis for this study lies in their effectiveness for gathering the general opinion of a target audience by providing an environment that allows probing for clarification and justification of opinion ((Morgan, 1988) and (Saulnier, 2000)). Focus groups are especially suitable for generating hypotheses when little is known in a specific research area (New Mexico State University - College of Agriculture and Home Economics 1999). Sofaer et al. suggest that if the previous work in a field is limited (which is the case here) then the research needs to be, at least initially, exploratory in approach (Sofaer, Kreling, Kenney, Swift, & Dewart, 2001).

IDENTIFYING SUITABLE PROCESSES FOR WEB SERVICE DEPLOYMENT

Two main assessment scenarios can be differentiated: (i) an organisation has a specific need and wants to evaluate the applicability of Web services within a selected business process, or (ii) an organisation wishes to identify those business processes, which would benefit most from the deployment of Web services. In both cases it is assumed that the organisation has conducted business process modelling and business process redesign activities beforehand, in order to start from an informed perspective.

Decision Methodology - Overview

In the following, a general decision methodology for the deployment of Web services is proposed for scenario (ii). It is shown what information would be required at which phase and to what extended business process models can be utilised.

The methodology is intended to serve as a guideline for systematically assessing the potential of business processes regarding the deployment of Web services and selecting the most appropriate processes and Web services. It helps answer “outside-in” questions (Bibby &

Brea, 2003) such as “to what business processes could Web services be best applied?”, and “what economic impact could that have?”

The methodology consists of a framework that follows a top-down structure with four decision phases. It includes several checklists to make it a practical instrument. A process’ Web services potential, i.e. the technical and economic feasibility and suitability of Web services integration within a selected business process is the main outcome of this methodology. The assessment is based on a scoring model, in which criteria and their weighting can be adapted and modified, making the tool highly flexible.

The assumed starting input are business process models, which could be the result of a comprehensive process modelling or business process improvement project. In the first phase, this existing set of business processes is evaluated against a shortlist of criteria, which allows to immediately disqualify some business processes for the deployment of Web services. These criteria could be that the processes are definitely unable to be Web services enabled or are already working very well so that running the risks associated with the change process would be unreasonable.

Within the second phase, the remaining subset of processes is evaluated with respect to its “web service – process suitability”. Processes can be classified into four categories based on organisation-independent characteristics, i.e. a) strong web service suitability, b) web service “learning chance”, c) future web service potential and d) limited web service potential.

The processes which fall into the categories (a), (b), and (c) are subject to further investigation within a third phase. Here, organisation-dependent criteria come into play, further reducing the set of potentially suitable business processes. This includes among others an assessment of the strategic importance of Web services for the organisation.

During the fourth and final phase, the organisation finally prioritises the remaining potential Web services projects, largely based on methods and measures known from conventional evaluation of alternative IT investments such as Return on Investment and Net Present Value (Remenyi, Money, Sherwood-Smith & Irani, 2000).

The following sections describe the different phases and the required input in more detail.

First Phase – Process Rejection Based on Disqualifying Criteria

At this phase business processes that match at least one of a list of disqualifying criteria are rejected. These organisation-independent criteria should be easy to assess without requiring a detailed investigation of the process models. Care must be taken to ensure that the criteria are chosen in such a way that they do not reject potential processes over-hastily. On the other hand they should be selective enough to reject as many unsuitable processes as possible and reduce the effort of detailed evaluations in the following phase. Thus, there is a trade-off between the amount of accidental disqualification of business processes and the workload at the following decision point.

Given that Web services are driven towards automated program-to-program interactions, and given the cost of reengineering existing processes and software to introduce Web services, a conservative set of criteria could be:

- The process involves only physical performance that cannot be digitised
- Human intelligence or sophisticated interpretation is required
- Isolated process which is working well, stable, efficient, and cannot be leveraged (i.e. does not represent “hidden value”).

Second Phase – Assessing General Web Services Suitability

At this phase the remaining subset of business processes from the first phase is evaluated using a “web service – process suitability” scoring table. The criteria are still independent from the specifics of an individual organisation. The goal of the scoring table is to assess the suitability of the business process for the application of web services based on two dimensions.

- The first dimension measures whether the *business* needs match with potential business drivers for Web services.
- The second dimension evaluates whether the *technical* requirements could currently be met by available Web services technology.

Each dimension is represented by a number of criteria which can be weighted and contain weighted sub-criteria. For both dimensions possible criteria are summarised in Table 1. The criteria and framework are based on criteria proposed by (Patricia Seybold Group), (The Stencil Group 2002), (Robins, Sleeper, & McTiernan, 2003), (John Hagel III & Brown, 2002), (Linthicum, 2002), (John Hagel III, Brown, & Layton-Rodin, 2002), (Wright, 2002), (Wilkov 2002), (Burdett, 2003), and findings from our two focus groups.

Second Phase – Qualifying Analysis Criteria

Business Need	Business Process Characteristics
Reduction of asset investment:	Unique expensive resources are currently used to support the process and could be replaced
Reuse and easier maintenance:	Redundant functionality in several application systems exists and shall be reduced/existing functionality shall be leveraged
Support for heterogeneous endpoints:	Business process requires support for multiple, heterogeneous interfaces
Automation of manual interventions and intensive data entry (human intervention for exceptions only):	Multiple manual, error-prone interventions dealing with digitized data are currently required for the business process
Automation of transaction chains:	Multi-step process, involving different business parties, shall be automated
Introduction of Self-service mechanism (enabling direct access to core system rather than cached or replicated data):	A batch process shall be replaced by a self-service, real-time mechanism
Higher transparency/visibility:	Frequent access to dynamically produced data has to be supported
Ad-hoc business:	Ad-hoc business with previously unknown parties shall be supported
Higher flexibility and business agility, dynamic process support:	"On demand" reconfiguration of business process required
Low impact of failure:	Financial risk of system failure is low for the business process
Technical Need	**Business Process Characteristic**
Processing speed:	No extremely short responses are required
Processing time guarantees:	No precisely predictable response time is required
Distribution of transaction volume:	Low transaction burst probability
Response to failure:	No failure compensation, roll-back, "state capture" are required
Security Requirements:	No non-repudiation, "chains of trust" and tamper-proofness are required
Semantic heterogeneity:	Shared meaning can be defined
Process repetition:	High repetition frequency
Process stability:	Process and involved application systems are likely to change over time
Transaction mode:	Real-time mode is required
Support for heterogeneity:	Multiple, diverse hardware and software systems are involved
Implementation effort:	Significant custom development would be required for conventional approach

Table 1: Process evaluation for Web services

Scores are then calculated independently for both dimensions. Every criterion which has been answered with a "yes" gets a score of one, every "no" results in a score of zero. The scores are then weighted and added as shown in Table 2. It is not in the scope of this work to determine how the scores should or could be derived, but we can note that traditional multi-criteria decision-making methodologies (Keeney & Raiffa 1992) could be employed for this purpose.

Characteristic	[No = 0, Yes = 1]	Weight; Σ(rows) = 1 (each ranging [0..1])	Score per Characteristic
xyz	{0;1}	[0..1]	= ({0;1} x [Weight])
…	…	…	…

		Σ(rows) = 1	Σ(rows)

Table 2: Simple scoring table for each of the two dimensions

The resulting score for each dimension of the business process under evaluation can then be visualized as a dot in a two dimensional matrix which represents its potential for Web services deployment. A possible matrix is given in Figure 1.

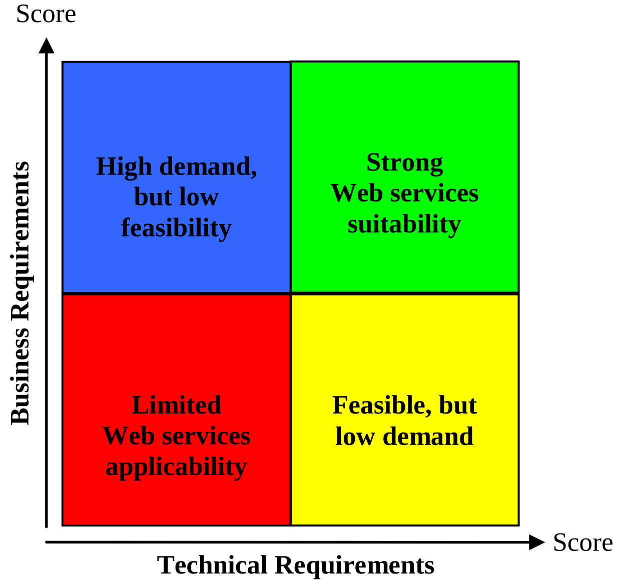


Figure 1: Web services suitability matrix

Third Phase – Organisational Characteristics

The third phase evaluated the general process suitability for Web services. After this phase, at least the processes marked with “limited applicability” can be eliminated. The following evaluation is not based on general process characteristics anymore but on *organisational-dependent* criteria. Here, it has to be established whether the qualified processes from phase 2 are suitable for web services with regard to the specific characteristics of the organisation. For example, in spite of high costs and risks, an organisation could decide to experiment with a “future web service” application etc. Potential questions leading to assessment criteria were identified with the help of the focus group sessions:

- Who are the involved business partners for the business process under consideration? What is the level of trust and the level of knowledge about their internal processes and systems? Do the partners already use standardised data formats or are they in the process of adopting Web services?
- Would the required technical resources be available?

- Would the required skill set be available?
- Are there example implementation and/or best practice available?
- What are potential risks? Consider risk affinity.

Some of these questions relate to internal organisational factors while others deal with external market characteristics. Clearly, the organisational assessment of Web services should consider criteria along both of these categories. The set of criteria presented in Table 3 derive from the above questions as well as relevant literature ((Christiansen, 2002), (Chen, 2003), (The Stencil Group 2002)).

Organisational Assessment Criteria	
Internal Factors	
Funding & Backing:	Business units specify and fund most major IT projects
Role of IT for organisation:	Use of IT is a competitive advantage
Role of innovation for organisation:	Innovation is a competitive advantage, organisation is risk taking
Current application architectures:	IT maintenance and integration costs are high
Importance of optimisation:	Increasing productivity is a strategic need
Current IT resources:	Current development & deployment platforms support Service-Oriented Architectures
Current available IT skills:	Adequately skilled personnel is available
External Factors	
Industry characteristics regarding specified data formats:	Industry uses standardised data formats (esp. XML)
Industry characteristics regarding data regulations:	Use or sharing of data is regulated by law
Industry e-commerce capabilities:	Industry has experience using B2Bi
Support from current IT vendors:	IT vendors have strategic support for Web services
Current business partners IT capabilities:	Partners have heterogeneous B2B capabilities
Market structure:	Oligopoly, more than one dominating player are present
Current business relationship characteristics:	Several trust-based relationship with deep mutual understanding of internal structures exist

Table 3: Organisational assessment criteria

Scores can then be calculated and combined in the same way as for the process analysis in phase 2. The resulting degree of current importance of Web services for an organisation could be again visualized as a dot in a two-dimensional matrix similar to the process evaluation phase. This is shown in Figure 2 and Table 4.

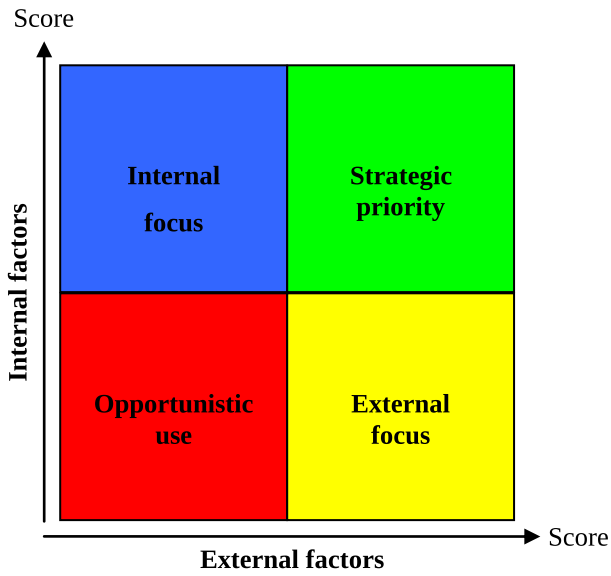


Figure 2: Matrix for WS strategic importance with respect to organisational criteria

Field in Matrix	Description	Likely Approach	Potential Risks
NE = Strategic priority	Web services should represent a significant element of the overall IT strategy. Strategic business processes will be affected. All major IT efforts should be considered in the context of fulfilling the Service Oriented Architecture vision.	Going for vision of service-oriented enterprise	Over architecting
NW = Internal focus	The organisation is positioned to make use of Web services. However, many of the partners and customers may not be. Therefore it makes most sense to look at how Web services-based integration can optimise internal processes and help better utilise existing assets.	Focusing on fixing while ensuring performing applications	Ignoring interesting market opportunities
SE = External focus	Web services represent an important way to connect to customers and business partners because of market dynamics. Web services-based offerings could represent a potential competitive advantage for first movers.	Using innovation for competitive advantage	Opening holes regarding security and scalability
SW = Opportunistic use	Web services may be an appropriate solution for specific projects. However, they do not represent a critical element of the overall IT strategy. Nevertheless, developers should be encouraged to experiment with the Web Service standards and related software tools.	Small steps for incremental benefits	Missing the strategic vision

Table 4: Explanation of the cells in the matrix of Figure 2.

Fourth Phase – Assigning Web services Implementation Priority

The remaining business processes that were generally suitable regarding their process characteristics (2nd phase) and met the organisations' specific situation (3rd phase) are prioritised in a last step. The goal here is to define for which processes should web services be deployed in the first instance. This can be determined by considering organisation-dependent factors. A list of proposed factors is presented in Table 5. Different weights could be assigned to the factors depending on the importance the organisation attaches to them. The set of criteria is based on (Christiansen, 2002), (Estrem, 2003), (Hammer & Champy, 1993) (John Hagel III & Brown, 2002), (Patricia Seybold Group), (Samtani & Sadhwani, 2002), and the focus groups.

4th phase – Priority criteria
Choose main "pain areas" where business partners or customers would like to be able to do things they cannot do at the moment Consider importance of involved business partner/customer for organisation. Choose projects where a new business need has to be satisfied and aggregated applications from remote systems can be leveraged. Choose projects for identified stable (proven) core business functionalities. However, the pilot area should not endanger established, mission-critical processes. Choose highly repeatable scenarios. Evaluate project's feasibility. Analyse of value proposition with (risk-adapted) return on investment analysis, economic value added (EVA) etc. for Compare estimated costs. • Compare estimated project duration. • Compare potential benefits. • Compare potential risks. Financial decision for evaluating investment alternatives, likely to be based on strategic cost management methods like total cost of ownership.

Table 5: Priority criteria

Based on these outlined logical steps an organisation should be able to systematically integrate web service technology as a facilitator of its business processes. The questions proposed for an assessment and their sequence guarantee that the most suitable and feasible activities and business processes for Web services support are identified. The framework can moreover be tailored to the individual characteristics of an organisation, as the criteria and their weighting are adaptable. Figure 3 summarises the outlined phases.

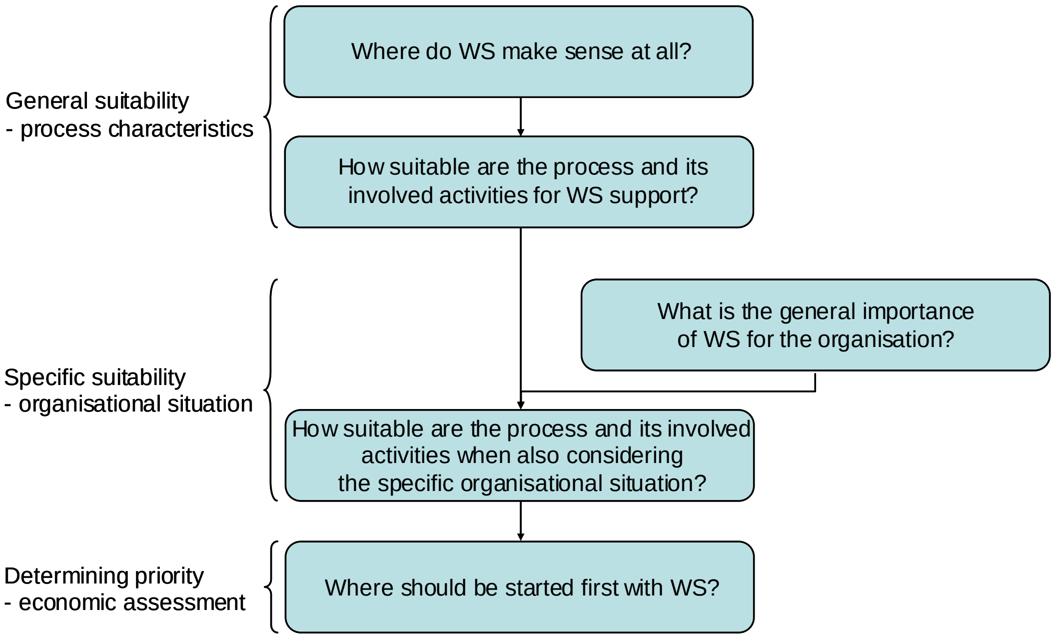


Figure 3: Identifying and evaluating Web Service opportunities

IDENTIFYING WEB SERVICE REQUIREMENTS FROM PROCESS MODELS

Information for the Assessment of Web Service Potential

Business process modelling can support the decision making process described above in all phases. Apart from presenting an overview of an organisation's processes, assigned actors and resources and their interrelationships, which is of great value in order to gain a fundamental understanding of how the enterprise works (e.g. compare with Schmelzer & Bloomberg, 2002), a wide spectrum of information can be captured in the process model and help answer the questions above. A list of identified critical information for the evaluation of web services and their deployment is presented in Table 6 and Table 7. This list was developed based on the information which has been identified as important for the different evaluation phases as well as with the help of additional literature (e.g. ebXML Business Process Team, 2002, Papazoglou, 2003). Furthermore, a classification into separate domains has been carried out.

Related Phase	Information Domain and Type	Detailed Description
	Process characteristics	
2	**Process stability**	Adaptability requirements (process' ability/likelihood to change)
2	**Process repetition frequency**	Frequency process is carried out with
2, 4	**Process's level of mission-criticality**	Degree to which the organisation relies on the

		process under consideration
2, 4	**Process's estimated monetary value if quantifiable**	Quantification of the value of the business process if possible
	Transactions characteristics	
2	**Business transaction type** E.g. Request/confirm, Request/ response, Notification, Distribution	Transaction type information supports an estimation of the degree of complexity of a potential service
2	**Composition requirements**	Required interrelations with other services, supports an estimation of the degree of complexity of a potential service
2	**Interaction mode** Synchronous, asynchronous or "as-agreed-by-parties"	"As-agreed-by-parties" indicates that the flow of control would be specified in trading partner agreements
2	**Message exchange requirements** Reliable delivery Ordered delivery Atomic delivery Message expiry	 Delivery of message until acknowledged Messages are to be delivered and processes in the order they are sent "All or nothing" delivery of messages to multiple partners Definition of validity of involved message
2	**Processing speed requirements**	Time constraints that have to be met
2	**Processing speed guarantees**	Accepted level of speed/time deviation
2	**Throughput requirements**	Rate (and peak rate) at which potential service is required to be able to process requests
2	**Scalability requirements**	Based on estimation of service's future use
2	**Security requirements** Authentication Authorization Confidentiality Data integrity Non-repudiation	 Identification and validation of message sender Assignment of rights to message sender Transport security/encryption requirements To ensure that data has not been altered between communication entities To ensure that transaction route is traceable and no aspect of the transaction can be denied
2	**Failure response requirements**	Compensation requirements for sub-transactions, roll-back, sub-transactions might also produce valuable results that should not be completely lost in case of failure (state capture)

Table 6: Information for Web Services evaluation: processes and transactions characteristics

Related Phase	Information Domain and Type	Detailed Description
	Involved systems' characteristics	
1, 2	**Description of modules and functionality**	Description of functionality to track redundancies, encourage re-use etc.
2	**Capacity utilisation level**	Description of system's current degree of

		utilisation and relationships to processes
2	**Costs (initial & maintenance)**	Description of current costs for, supporting potential reduction of asset investments etc.
2	**Existing interfaces**	Description of interfaces to asses degree of required support for heterogeneity
2	**Used communication protocols**	Description of communication protocols
2	**Systems ability to change**	System's adaptability support and requirements
	Involved data's characteristics	
3	**Data format & standards compliance**	Description of data structure
2	**Dynamics, frequency of change**	Description of current level of dynamics of involved data
2	**Importance of timeliness**	Requirements for timeliness of involved data
2	**Required level of data security**	Description of required degree of end-to-end security for data
	Involved business partners' characteristics	
2, 3	**Total number of involved parties**	Higher number usually means higher complexity that has to be supported
4	**Assumed frequency of cooperation**	Information could be used for assigning priority to potential supporting IT project
4	**Importance of business partner to organisation**	Information could be used for assigning priority to potential supporting IT project
3	**Autonomy, degree of individuality**	Partner's IT compliance to existing "global" standards and agreements
3	**Existing level of business trust**	A high level of business trust is especially considered to be important for near-term external Web Service projects
3	**Existing process insight, manageability, shared meaning**	External visibility, understanding, and manageability of partners' applications
3	**Existing technological base**	Description of partners' current IT systems
3	**Existing IT skill base**	Description of partners' current IT skills

Table 7: Information for Web Services evaluation: systems, data and partners characteristics

Information for facilitating Web services deployment

Apart from supporting the *identification and evaluation* of opportunities for Web services deployment to improve business processes, business process models are also a valuable tool for facilitating Web services deployment. Business process models could capture a) patterns, b) Web services taxonomies and c) Web services semantics.

Patterns may be identified in Web services practices and added as additional, classifying information to a model. Once patterns are identified and captured they provide opportunities for simplifying structures and processes. Besides, these identified patterns promote the re-use of knowledge and functionality which reduces the development effort. Encouraging and reinforcing consistency and standardisation (e.g. compare with Glushko & McGrath, 2002) can also lead to reduced maintenance. IBM, for example, offers a set of e-Business patterns to facilitate the process of developing web-based applications. As a general rule, they expect that the emerging Web services affect the implementation of all their presented patterns, i.e. business, integration, and application patterns, whenever there is a boundary between businesses, applications, or logical components of a solution across which information must be exchanged (Adams, Gisolfi, Snell, & Varadan 2002).

Because Web services are presumed to be re-used, a prerequisite for efficient service development is also the creation of a comprehensive reusability strategy. One of the bases for this strategy should be a taxonomy of services (Scholler, 2003). *Web service taxonomies* help categorise Web services, e.g. based on their role or function they provide within an overall enterprise. Scholler (2003), for instance, proposes a 2 x 2 matrix taxonomy consisting of the dimensions provider scope (e.g. the provider may be a particular organisational unit, and its associated applications or the provider may also be enterprise wide in scope) and consumer scope (e.g. consumers may be local to a particular organisation, or the consumers may be global and outside the boundaries of the enterprise). These dimensions result in at least four classes of services with accompanying different strategies that should be followed (Scholler, 2003). Other examples for possible Web services taxonomies include the business purposes that trigger Web services implementations. Web services taxonomy information could be attached to the Web services implementations that are captured in business process models.

Apart from capturing patterns and web service taxonomies, *web service semantics* (i.e. service capabilities, additional functional and non-functional properties) can also be recorded in business process models to facilitate Web services deployment. Documentation of Web services capabilities and additional functional properties will support the re-use of services and facilitate communication with internal and external parties involved in the Web services implementations. Furthermore, capturing non-functional service properties will be necessary for Web services (provider) evaluation.

Table 8 lists information that is important for Web services deployment and can be captured in a business process model. The information domains and types presented are based on (Scholler, 2003), (National Health Supply Chain Taskforce Interoperability Working Group, 2002), (ebXML Business Process Team, 2002).

Information Domain and Type	**Detailed Description**
Business purpose for web service	
Capturing the business purpose of Web services implementations provides a basis for identifying knowledge and know-how for future implementation projects.	
Interaction pattern	
Web services transactions could potentially also automate more complex interaction patterns to great advantage in the future. Capturing the interaction patterns supported by existing Web services implementations would offer the chance to identify reusable knowledge if the same interaction pattern was to be supported in a new project.	
Simple Transaction (1:1)	Any Web services where the objective is for the provider to execute an operation on behalf of the consumer. (E.g. order taking, billing, buying, reporting, finding, reserving)
Agent (1:1:n)	A Web services that acts as an agent providing intelligence in the selection of other services. (E.g. search engine, travel agent that maintains up-to-date arrangements, automatic trading agent)
Dealer/Intermediary (n:1:n)	A third party that locates, aggregates, potentially inserts value-adding services
Auction (1:n)	An auction service allows an individual or enterprise to offer various forms of auction service on a private or public basis. (E.g. personal auction service, bid processes)
Virtual hub (n:n)	Core business services are exposed and executed directly by other parties in a collaborative process. (E.g. supply chain process)
Relationship type	
Buyer to major supplier, Buyer to small supplier, Buyer/supplier via e-Marketplace, Buyer/supplier via a third party (“exchange hub”), Ad-hoc, previously unknown	Identified patterns would encourage re-use of functionalities, thus facilitate deployment etc.
Service semantics	
Service ontology & capabilities	Description of what the service is about and how it can be discovered (e.g. synonyms for name etc.)
Functional service properties e.g. identification, location, etc.	Potentially supporting re-use of services and communication with business partners
Non-functional service properties e.g. availability, costs, ownership, quality, etc.	Potentially supporting provider evaluation etc.

Table 8: Identified critical information for Web Service deployment

INTEGRATION IN ARIS AND EXAMPLE

This section discusses how the proposed methodology for Web services assessment and deployment can be supported by a mainstream business process modelling solution, namely ARIS. The implementation of the methodology is then illustrated through an e-Procurement scenario.

Introduction to ARIS

ARIS is a mature business modelling tool, which is regularly ranked in market studies as the most advanced solution for process modelling and analysis. Its sophisticated capabilities and its wide distribution is practice motivated us to select ARIS for the purpose of this research. ARIS (Architecture of Integrated Information Systems) is a process-oriented business process documentation, analysis, and improvement framework (supported by a toolset) that attempts to span the gap between business theory and information/communication technology (Scheer, 1998a). In ARIS, business processes are represented in diagrammatic form as chains of events and functions (EPCs). Apart from processes, ARIS can be used to model systems, resources, data, software, information flow, organisation, knowledge, skills, business objectives, risks, and costs (Davis, 2001). The result is a highly intricate model which is divided into views in order to reduce its complexity. With such division, the contents of the individual views can be described by special methods. The description may either be performed from a purely functional point of view, or the applications may be considered from the point of view of the data. A third perspective is the organisational one, where organisational units and responsibilities are presented. In order to maintain the relational structure between functions, data, and organisation, the control view shows, for instance, what data is processed by which functions (Scheer, Abolhassan, Jost, & Kirchmer, 2002, p. 17). A fifth view, the output view, represents resulting products and services. Output is the result of processes and describing output is seen as one of the key processes in describing business processes (Scheer, 1998b, p. 93).

The ARIS Toolset supports a range of modelling techniques. Several model types were evaluated regarding their suitability for supporting the integration of the identified critical information for web service assessment and deployment. Among them are the extended Event-Driven Process chain (eEPC), Column eEPC, Process Chain Diagram (PCD), and the recently introduced E-Business Scenario Diagram. In our example, the latter was used for the top-level modelling. The extended Event-Driven Process Chain is the chosen model type for modelling greater levels of detail. Both techniques, E-Business Scenario Diagram and the Event-Driven Process Chain, are also used within the enterprise System SAP. Thus, we believe that our examples can easily be understood by the wide community involved in SAP-related modelling activities.

ARIS model types employed

EPCs are activity-oriented diagrams which are depicted in the process view. The structure of an EPC is that of a directed graph with active nodes ("Functions") and passive nodes ("Events"). A process is described via an EPC as a chain of business functions, where each function describes an activity and is preceded by and succeeded by events. The latter represent the prior and subsequent situation regarding the function (Soderstrom, 2002). In ARIS, events are graphically represented by a hexagon shape; functions are displayed as soft rectangles. In addition to that, rule operators, represented by circles, illustrate AND, OR, and XOR decisions and are used to model the internal structure of a process (e.g. branching, re-branching, parallel sub-processes etc.). Dotted arrows connect the elements depicting the control flow.

eEPCs are event-driven process chains which are "extended" by the inclusion of elements that are specified in greater detail in other views. That way eEPCs can represent how the available resources implement a process and how the process interacts with its environment. Based on such a model the following types of questions could be answered: a) who does it? (organisational unit), b) what do they do? (function, information carrier), c) how do they do it? (knowledge, application system), d) why do they do it? (objective), and e) when do they do it? (event) (Davis, 2001, pp. 162-163).

Table 9 shows common object types that were also used for the implementation of the exemplary business process model presented later. Their description has been adapted from the ARIS Methods Manual (IDS Scheer AG, 2002).

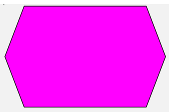
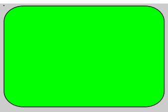
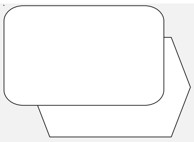
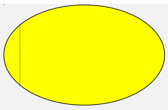
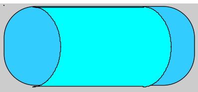
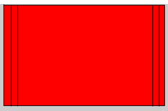
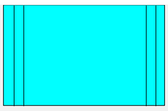
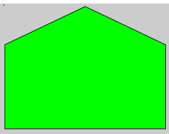
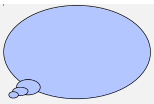
Symbol	Object Type Name	Description
	Event	Events trigger functions and are the results of functions.
	Function	A function is a technical task or activity performed on an object.
	Process Interface	A process interface indicates from which process the related event has been created, or which process the event triggers.
	Rules X-OR AND OR	The rules describe how the events and functions are related. The X-OR means that one and only one input/output is possible, the AND that all the inputs or outputs must be true, and the OR when any combination may be possible.
Resource objects		
	Organisational Unit Type	An organisational unit type represents a typification of individual organisational units, i.e. performers of the tasks required to attain the business objectives.
	Information Carrier	An information carrier is a means to store information.
	Cluster	A cluster represents the logical view on a collection of entity types and relationship types of a data model.
	Application System Type	The Application System Type is representative of a related group of IT systems.
	Objective	An objective is the definition of future company goals
	Knowledge Category	A knowledge category is used to classify knowledge by topic

Table 9: Common objects within the ARIS Toolset

To facilitate the modelling of e-Business processes, the ARIS framework incorporates a dedicated type of diagram, namely “E-business Scenario diagram”. Using this type of diagram it is possible to view a value-added chain holistically, i.e. from the end customer through all the companies involved in the process. By adopting the column representation style, the E-Business Scenario diagram provides an abstraction of the interfaces between different process partners.

Apart from involved business participants that are placed in the “header row” and the central elements, business processes, different information carrier objects (e.g. Internet) are also available to present the underlying media by which business documents are passed across boundaries (Davis, 2001, p. 345). Business component objects, which represent the application system type used in normal eEPCs, can also be included. Furthermore, security protocol objects can be attached to the business documents to specify security requirements. As with eEPCs, the organisational, data and systems description can be specified in greater detail within additional assignable models. The symbols (representing different objects) offered by the E-business Scenario Diagram type are shown in Figure 4.

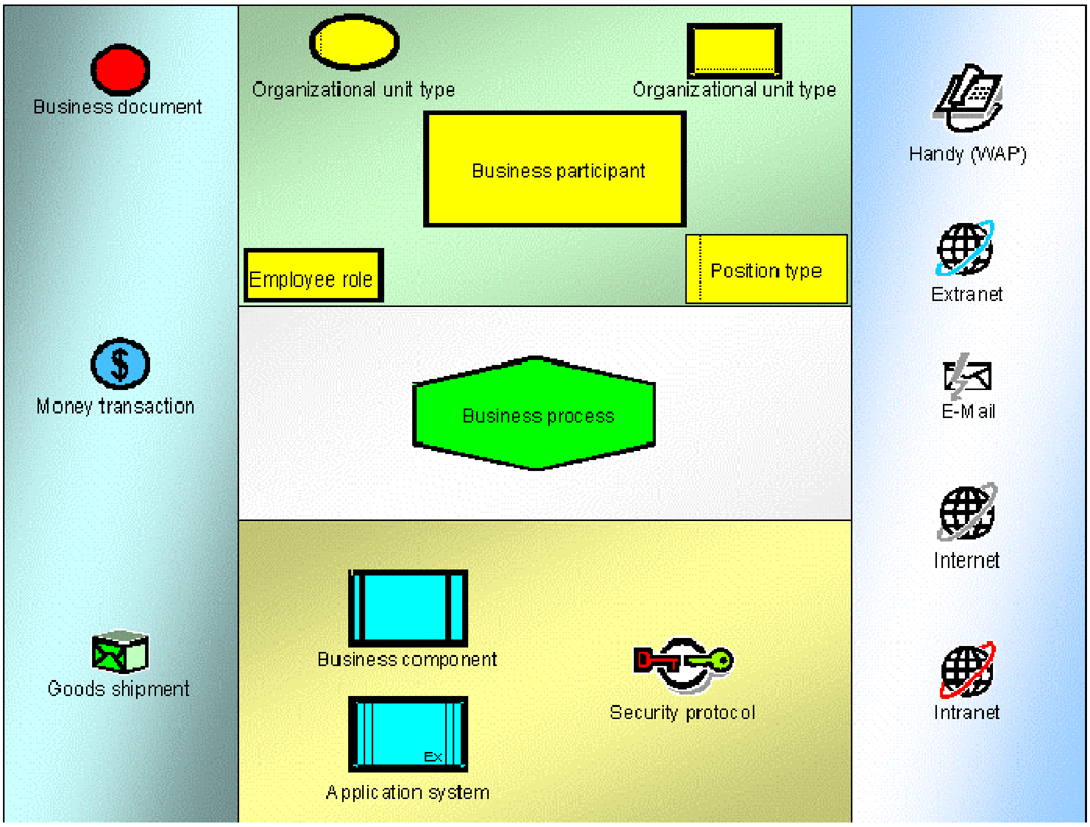


Figure 4: Modelling Symbols for E-Business Scenario Diagrams

Description of Relevant Modelling Constructs in ARIS

The following sections present the modelling constructs and techniques that could be used – in addition to the standard elements “function”, “state”, “operator” and “connection” – to capture the information relevant to Web services assessment and deployment (hereafter referred to as “Web Service modelling”) in an ARIS business process model.

Hierarchical decomposition

It is a natural design technique to start by creating a high-level concept and then to drill down into successive levels of detail (Davis, 2001, p. 242). Process decomposition is achieved by assigning hierarchies of eEPCs to functions (Davis, 2001, p. 243). Apart from that, most model assignments in ARIS are made to models that provide additional details about the particular object. The most relevant for this work are presented in Table 10.

Object	**Assigned Model**	**Hierarchical Representation**
Function	eEPC	Decomposition of the Function into a more detailed sub process
Application System Type	Application System Type Diagram	Decomposition of the systems into sub-systems, modules and IT functions
Organisational Unit Type	Organisation Chart	Description of the organisational set-up of the involved businesses
Information Carrier	None	However, the Technical Terms Model or the eERM Model can be assigned to show structure of carried data
Cluster	eERM Model	Formal description of the data structure
Technical Term	Technical Terms Model	Decomposition of the Technical Term into its information structure
Knowledge Category	Knowledge Structure Diagram	Description of the structure of business knowledge
Business Objective	Objective Diagram	Composition / decomposition of the business objectives and description of related critical factors

Table 10: Hierarchical models in ARIS that can be assigned to objects, adapted from (Davis, 2001)

Attributes

Attributes are populated with values either through the process of drawing the models or by inserting them manually. Apart from storing modelling related information that is necessary for the administration of the databases, models and its objects, additional information about the real world items that the model represents can be added. Special attributes further allow for linking business documents and web sites or other applications to objects, models and databases. Thus, although ARIS’ attributes are not intended for storing vast amounts of detailed information about the items themselves, a business process model can act as a central repository (Davis, 2001, pp. 25, 97). For convenience, the attributes can also be displayed directly on the model graphic (Davis, 2001, p. 91).

Organisational objects

Organisational objects represent information on business participants that are involved in the process tasks. Many practitioners model every organisational object, be it a single person, department or a whole organisation, as an organisational unit object which is feasible and keeps the models simple (Davis, 2001, pp. 145, 147). However, a hierarchical approach is more appropriate if complex projects shall be modelled and shared to ensure a common, standard-based approach. Therefore, the detailed relationships between organisational objects can be modelled within the Organisational Chart model. In both model types, eEPCs and E-Business Scenario Diagram, it is possible to assign organisational chart models to organisational objects (Davis, 2001, p. 145).

Application system objects

Application system objects represent the IT assets in ARIS that are used to support the business. Many objects exist to define detailed hierarchies of systems, sub systems, software modules, and even specific IT functions. Although in practice only the application system type element is used within the majority of eEPCs, an application system's internal relationships can be displayed in the assignable application system model type (Davis, 2001, p. 148).

If processes are entirely carried out by application systems the corresponding function symbol can be replaced by a designated object called system function (Davis, 2001, p. 150).

Data objects

Involved data in IT systems and communication can be modelled formally, i.e. using recognised modelling standards such as ER-diagrams, or less formally using "business language". Whereas the Technical Term object is used for modelling data informally from a business perspective, the cluster, entity type and attribute objects represent formal data modelling in ARIS (Davis, 2001, p. 150). Their internal relationship can be shown via the eERM model type. The Technical Terms model can be used to model how Technical Terms map to Clusters, Entities and Attributes of the formal data model (Davis, 2001, p. 151).

Information Carrier objects

Information carriers can be thought of defining how the data is stored and delivered or "carried" to and from the Functions (Davis, 2001, p. 155). Symbols are available for EDI, Intranet, Internet, Email, Fax, etc. Explicit relationships between the involved data and its carrier can also be modelled (resulting in so called "secondary relationships)" (Davis, 2001, p. 157). However this relationship cannot be modelled (visually) in the E-Business scenario diagram type.

Objective objects

A hierarchy of Business Objectives and related Critical Success Factors can be modelled within the Objectives Diagram model. The specific objectives could then be added to an eEPC and assigned on a Function/EPC level to show which process steps support their realisation (Davis, 2001, p. 161).

Knowledge objects

Knowledge is considered everything that is known to be of relevance to a process (Davis, 2001, p.158). In process modelling one would not want to try to model all of the knowledge related to the process but only where it was key to a process step (Davis, 2001, p.158). The ARIS object chosen for knowledge here is the Knowledge Category object. For more detailed levels, e.g. to represent the structure of knowledge or interrelationships, specific designated model types exist as with the other resources as well. The Knowledge Structure diagrams are useful models for representing and structuring aspects of business knowledge, thus also facilitating communication and re-use of the latter. For Web services modelling they can be redefined in order to depict available case studies, benefits realised through Web services application etc.

Matching Web Service Information with ARIS Constructs

In the following, it is outlined how necessary information supporting Web services evaluation and deployment could be captured as elements of a collaborative business process model created with the ARIS Toolset. Here, a matching between the identified critical information supporting Web services assessment and deployment (Table 6, Table 7 and Table 8) on the one hand and the outlined appropriate ARIS constructs for representation in a collaborative business process model on the other hand is performed. The outcome is shown in Table 11, Table 12 and Table 13.

Phase	Information domain and type	How to capture in a process model?
	Process characteristics	
2	**Process stability**	To be specified on Function/EPC level as discrete attribute (predefined list)
2	**Process repetition frequency**	To be specified on Event level as attribute
2, 4	**Process' level of mission-criticality**	To be specified on Function/EPC level as discrete attribute (predefined list) or through colour coding (i.e. representing value of attribute by displaying corresponding Function objects in different, designated colours)
2, 4	**Process's estimated monetary value if quantifiable**	To be possibly specified on Function/EPC level as attribute or through colour coding
	Involved electronic transaction characteristics	
2	**Business transaction type**	Information implicitly available through process structure
2	**Composition requirements**	Relationships of constituting parts of transaction are implicitly available through process structure
2	**Transaction mode**	To be modelled as a discrete attribute (predefined list)
2	**Message delivery requirements**	Information may be attached as to Information Carrier or Function
2	**Failure response requirements**	Information may be attached to Information Carrier or Function
2	**Required processing speed**	To be specified on functional level as attribute
2	**Processing speed guarantees**	To be specified on functional level as attribute
2	**Throughput requirements**	To be specified on functional level as attribute
2	**Scalability requirements**	To be specified on functional level as attribute
2	**Security requirements**	Can be modelled in E-Business Scenario Diagram as designated symbol with own attributes. However, no mapping of the symbol to other diagrams (e.g. eEPCs) is possible. Could alternatively be modelled as attributes of other practical object that was assignable to information carriers or as direct attributes of Information carrier or Function.

Table 11: Capturing WS-relevant process and transaction characteristics in process models

Phase	Information domain and type	How to capture in a process model?
	Involved systems' characteristics	
1, 2	**Description of internal structure and functionality**	To be described as attribute of Application System Type Module or IT Function
2	**Capacity utilisation level**	To be specified as attribute of Application System or Module
2	**Costs (initial & maintenance)**	Same as above
2	**Existing interfaces**	Supported input and output formats to be specified as Data objects and/or as attributes of Application System
2	**Used communication protocols**	Depicted through information carrier
2	**Systems ability/likelihood to change**	To be specified as discrete attribute (predefined list) of Application System or Module
	Involved data's characteristics	
3	**Data format, standards compliance**	To be described as attribute of Cluster object and specified in eERM model if complex
2	**Dynamics, frequency of change**	To be specified as attribute of Cluster object
2	**Importance of timeliness**	To be specified as discrete attribute (predefined list) of Cluster object
2	**Required level of data security**	To be described as attribute of Cluster object
	Involved business partners' characteristics	
2, 3	**Total number of involved parties**	Derived from relationships with Organisational Units
4	**Frequency of cooperation**	Discrete attribute (predefined list) of Organisational Unit
4	**Importance of business partner**	As above
3	**Autonomy, degree of individuality**	As above
3	**Existing level of business trust**	As above
3	**Existing process insight and manageability, shared meaning**	As above (possibly multiple attributes)
3.	**Existing technological base**	Implicitly contained in model through Application Systems if process & resource insight is granted
3.	**Existing IT skill base**	Same as **Frequency of cooperation**
	Types of characteristics of the involved business parties could also be modelled as redefined Knowledge Category objects and be assigned to the organisational objects representing the business participants via the Knowledge Map model type.	
	Examples, first implementations and maturity, risks, and pitfalls	
3, 4	Can be referenced as attribute on Function/EPC level. Colour coding could be used to assign the implementation's level of maturity on the Function/EPC level (designated colours for discrete levels of maturity). Could alternatively be modelled as redefined Knowledge Category objects that could be assigned on Function/EPC level and colour coded according to the maturity. The advantage of using Knowledge Category objects is that they could carry further details (e.g. experienced issues, benefits) in assigned Knowledge Structure Diagrams.	

Table 12: Capturing WS-relevant system, data and partners characteristics in process models

Information domain and type - Web Service deployment -	How to capture in a process model?
Business purpose for Web Service	
Business drivers could be captured in several plausible ways, as: a) a discrete attribute (predefined list + free text if value not yet in list) on Function/EPC level. b) redefined specific Knowledge Category objects. The advantage would be that these objects could then be directly assigned to Functions as well as be included in Knowledge Structure diagrams detailing known Web Service implementations. Objective objects can only be assigned to Functions. c) a discrete attribute (predefined list + free text if value not yet in list) of the Knowledge Category objects which refer to first implementations and are assigned to Functions. d) Objective objects in a hierarchical Objective Diagram + assigned on Function/EPC level after first Web services projects.	
Interaction patterns	
Interaction type classifications could be captured in several plausible ways, as: a) a discrete attribute (predefined list + free text if value not yet in list) on Function/EPC level. b) redefined specific Knowledge Category objects (that could be part of a Knowledge Structure Diagram attached to a Knowledge Category object for first implementations) c) Comments, after first own or reported projects. d) a discrete attribute (predefined list + free text if value not yet in list) of the Knowledge Category objects which refer to first implementations and are assigned to Functions. This is recommended, because it constitutes a simple, discrete type of information.	
Relationship type	
See **Interaction patterns** above	
Service semantics	
Functional service properties	To be described on Function/EPC level as attributes or as attributes of the Knowledge Category objects representing Web services examples
Non-functional service properties	To be described on Function/EPC level as attributes or as attributes of the Knowledge Category objects representing Web services examples
Service ontology & capabilities	To be described on Function/EPC level as attributes or as attributes of the Knowledge Category objects representing Web services examples
Misc.	
Information can be included that is critical or has proven to be important in the past	To be captured in model as attributes, Comments or Knowledge Objects

Table 13: Capturing information for Web Service deployment in a Business Process Model

E-Procurement Scenario

Figure 5 depicts the high-level business processes of the e-Procurement scenario modelled with the ARIS Toolset. These processes are presented in an E-Business Scenario Diagram. The header row contains the involved business parties; the following row holds the related business processes and resources. This model is of the “swim lane” type. The column presentation therefore visualises the interface between the two business partners. Electronic communication takes place in form of business document exchanges. Most of the business processes carry an “assignment” symbol in their bottom right corner which depicts the fact that an associated eEPC is available that further details the business process.

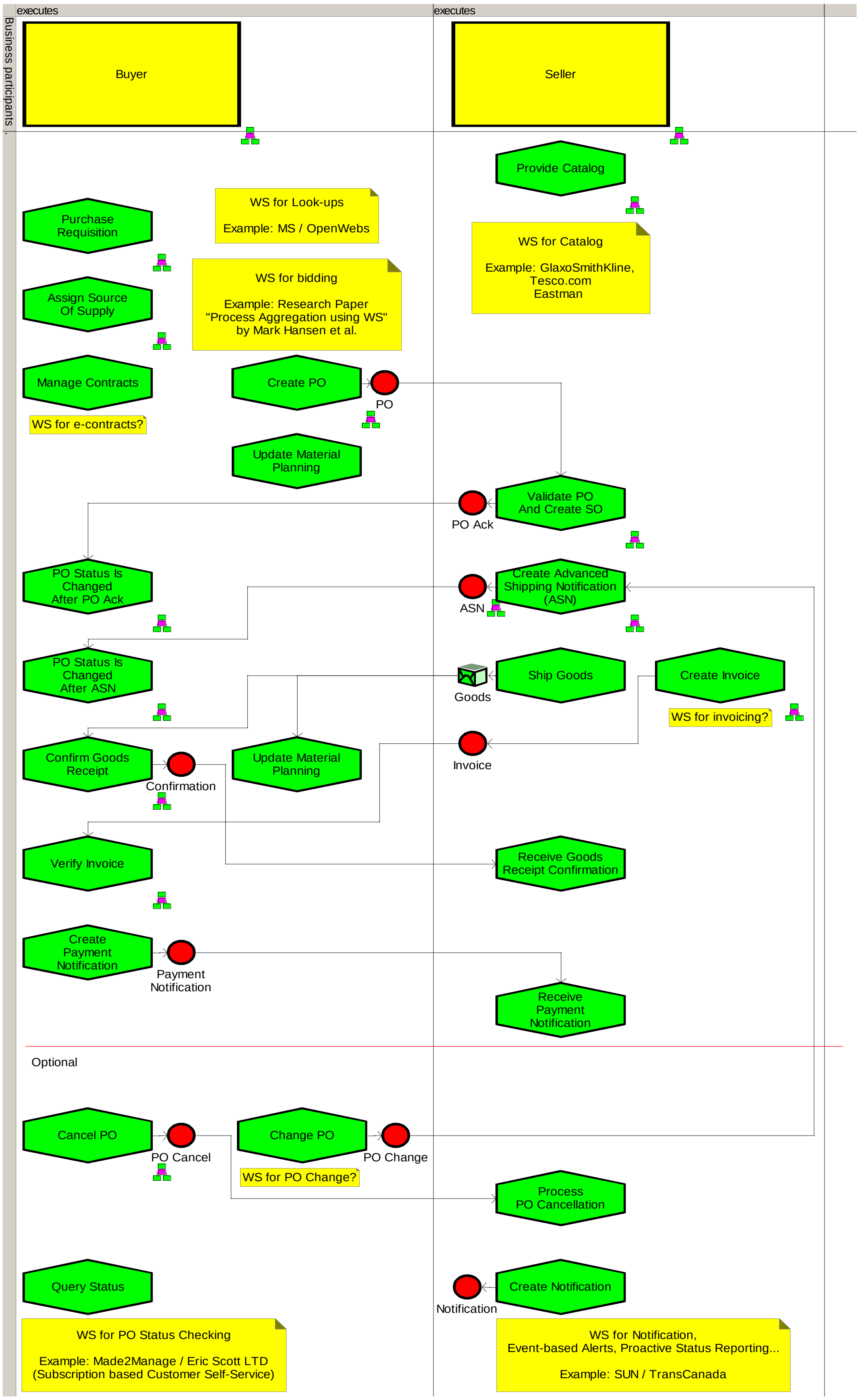


Figure 5: High-level processes for e-procurement in E-Business Diagram

The user can quickly browse through the models with the help of these visual links. The modelling symbols have been introduced in Figure 4.

The scenario starts with the product catalogue provision by the Seller. The catalogue can be provided as a Web service which would be a service to the Buyer. An advantage would be the support for heterogeneous systems, i.e. the catalogue Web services could be integrated on a web site, as a small desktop application at the Buyers site etc. Upon identification of a specific product need, a purchase requisition is triggered on the Buyer's side, who assigns a source of supply which may lead to an update of contract information. Thereafter a purchase order (PO) is created and the material planning system is updated. Upon reception of the purchase order the Seller validates it and creates a sales order (SO). A PO acknowledgement (PO Ack) is sent to the Buyer who changes the status of the purchase order. The Seller provides the required goods and sends an advanced shipping notification (ASN) to the Buyer which leads to another update of the PO at the Buyer's side. Finally the goods are shipped and an invoice is created and transmitted to the Buyer. Although no examples could be found, the invoicing process might prove suitable for Web services integration. The Buyer confirms the reception of the goods, updates his material planning and verifies the invoice upon arrival. A payment notification is sent to the Seller when the invoice has been verified. In addition, message exchange is required for PO amendment, status querying and further notifications. Existing Web services solutions are attached to the model in the form of comments. Comments are also attached to processes where Web services applications should be clearly considered, e.g. automation of (parts of) contracts, support for the invoice process and purchase order changes.

Figure 6 depicts the process "Create ASN" in greater detail. The representation chosen is an eEPC model. The column type gives a corresponding "swim lane" view on the involved business parties. In ARIS the model shows up after a double click on the assignment symbol of the "Create ASN" process in the E-Business Diagram. The model shows the required activities and resulting states to carry out the business process. It also includes involved application systems, data and information carriers and their interrelationships. Applications systems and data can be further detailed in related diagrams such as the eEPC in Figure 6. The activities (Function objects) supported through Web services carry a Knowledge object that represents the corresponding type of Web services example, differentiated according to their maturity, i.e. research prototype, vendor proposal, success story. The different degree of maturity is depicted through colour coding of the Knowledge objects (i.e. predefined colours for low maturity, moderate maturity, high maturity: red-yellow-green). The Web services examples can be detailed in Knowledge Structure Diagrams, which can include information about benefits, issues and hyperlinks to relevant documents.

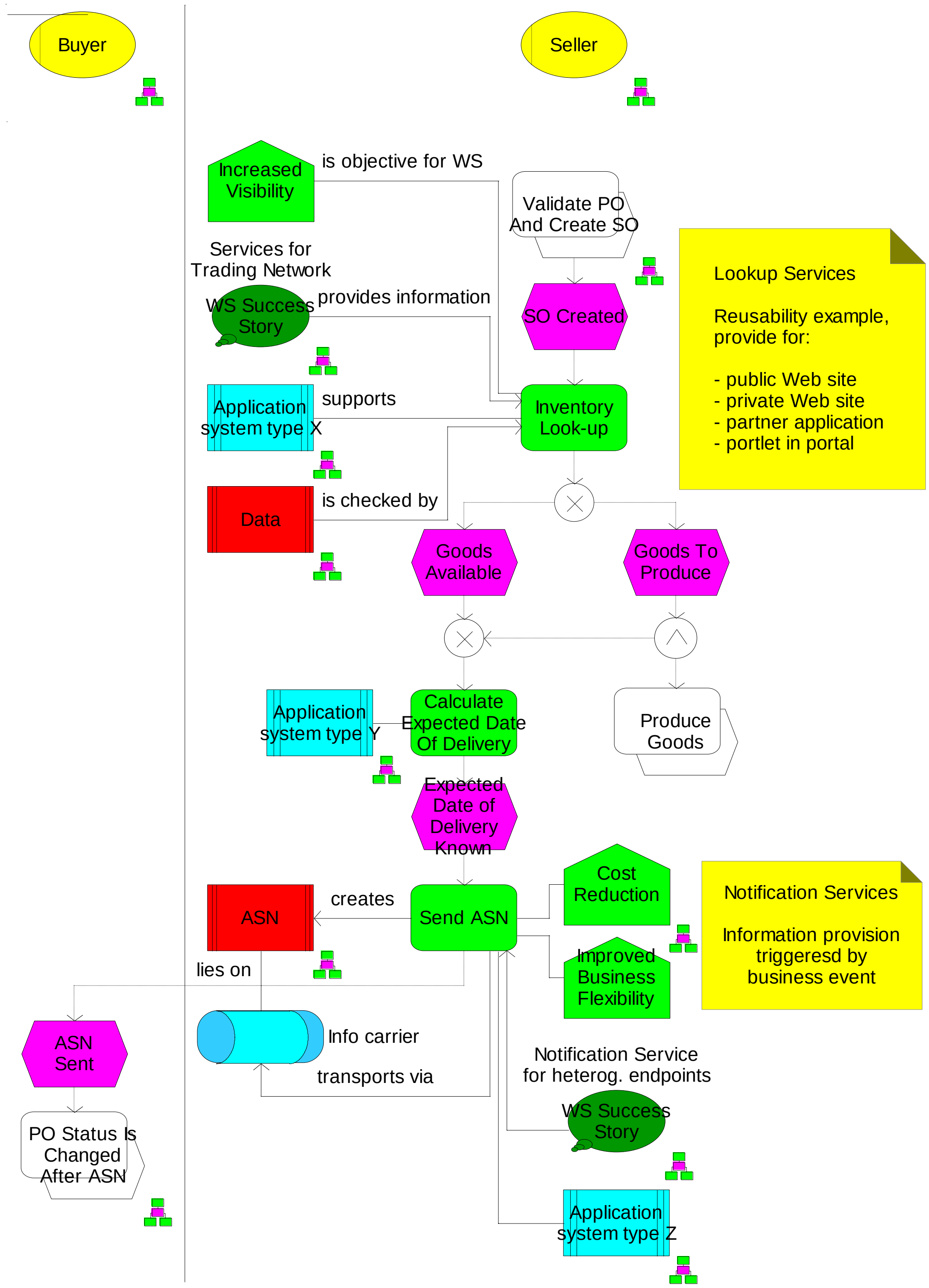


Figure 6: Seller's process "Create ASN" in greater detail via an eEPC

RELATED WORK

The methodology proposed in this paper is closely related and complementary to other methodologies proposed in the area of Service Analysis and Design (SAD). SAD refers to the stages in the lifecycle of a service-oriented architecture where the purpose, scope and interfaces of individual services are identified, and a priori relationships between services are documented. It is also in these stages that the relationships between service models and other enterprise modelling artifacts, such as organisational charts, document flow charts and business process models are specified. Various methodologies for SAD have been proposed. In this section, we discuss three SAD methodologies and relate them to the methodology proposed in this paper.

Jones (2005) proposes a top-down approach for the definition of a *Business Services Architecture*. The methodology revolves around four questions (What, Who, Why and How) which are considered at different levels of granularity. The methodology relies on a recursive identification of the business services of an organization (the so-called *What*), through the analysis of the consumers of functionality provided by the services (*Who*), the interactions involved (*Why*), and the details of the implementation (*How*). A systematic consideration of these questions leads to the identification of services, actors and interactions, which can be represented in "interaction diagrams", out of which interaction scenarios can be subsequently discerned. A service, according to this methodology, is a discrete domain of control containing a collection of tasks to achieve related goals, and can be identified with a business function. Thus, the methodology primarily emphasises the identification of coarse-grained business services. These coarse-grained services are then be decomposed into finer-grained services, down to the level of "technical Web services" which are identified at the lowest level of the decomposition, and are seen as an implementation of the business services.

Erl (2005) proposes a methodology for identifying technical Web services and their interaction points (called "operations"). This three-step methodology starts with the definition of the business requirements relevant to a service-oriented solution. The author works out two possible approaches for this first step, depending on the source from which business requirements are drawn: (i) the *Process-based approach*, which takes business processes as the starting point for the requirement analysis; and (ii) the *Entity-based approach*, which takes business documents and transactions as starting points. Having discerned the business requirements, the second step in Erl's methodology is to identify existing application logic which already automates any of these requirements. This step helps to scope the potential systems affected by the introduction of a service-oriented architecture, and it is particularly useful for large-scaled solutions. The last step aims at identifying suitable *service operations* that can fulfil the business requirements (e.g. a service operation that provides an implementation for a process step or that supports a business transaction) and that encapsulate or replace the legacy application logic. These service operations are then grouped into *services* according to their logical context.

The IBM Service Oriented Analysis and Design methodology (Zimmermann et al., 2004) is essentially an adaptation of traditional Object Oriented Analysis and Design (OOAD) methods to the realm of Web services. The purpose is to identify a suitable set of technical Web services to be deployed, from the analysis of a given business scenario. The starting point is the definition of a class diagram of the software objects involved, which aids the construction of a *Services Model*. Here services are identified by their operations on the basis of related behaviour. Next, business processes are defined as state transition models, in order to capture the behaviour involving each software object that has been previously identified. Finally, the interaction behaviour of business processes and services is sketched in a sequence diagram leading to a so-called *service choreography*.

A classification of Service Analysis approaches is depicted in Figure 7. Each quadrant in this figure identifies a distinct combination of an *abstraction level* and an *analysis driver*. The abstraction level can be *Business* or *Technical*, and depends on the purpose of the methodology and its deliverables. Meanwhile, the main analysis drivers can be *business processes* or *business entities*, depending on which of these is taken as a starting point.

According to this classification, our methodology is process-driven, as we start from the analysis of candidate business process models. Besides, the scope of applicability of the methodology can be placed in-between the business and technical levels, as we do not drill down into the details of Web services implementation, although we identify technical requirements for Web services based on process models.

Jones's methodology (Jones, 2005) falls in the business level and entity-driven quadrant: it aims at defining a business services architecture and it starts with the identification of business services corresponding to organizational functions. The methodologies of (Erl, 2005) and (Zimmermann et al., 2004) are at the technical level, as they aim at defining technical interfaces of Web services. They take business process models as analysis driver (in the case of Erl's process-based approach) or entity models (in the case of Erl's entity-based approach and Zimmermann's object-oriented approach).

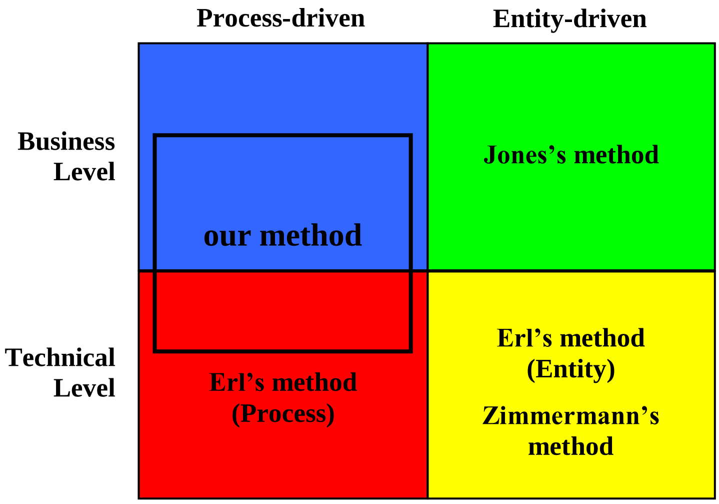


Figure 7: Classification of the SAD approaches

Technical and process-driven approaches, as the one proposed by Erl, are suitable for organizations with detailed business process models or as part of an SOA transition plan. They focus on meeting immediate requirements (i.e. the SOA automation of an existing business process) and require little service analysis effort. These approaches can also enable reuse and prepare business processes for integration with partner processes. However, in order to provide immediate benefits, they usually focus only on a business process model at a time. As a result, these approaches can limit the long-term reusability potential of the resulting Web services, as an initial study of multiple process models is often required to identify commonalities. Therefore, they should be complemented with other methodologies.

On the other hand, entity-driven approaches like Jones's, Erl's entity-based approach and Zimmermann's methodology, identify highly reusable services due to their inherent generic nature, and can significantly increase the agility through which service-oriented processes can be redesigned. However, they require more up-front analysis and thus they usually increase the cost and time for service production. Besides, if applied in an overly abstract manner, they may yield conceptual service models of little practical use in practice.

Process-driven approaches naturally lead to situations where processes are the dominant feature. If applied in a simplistic manner, they may lead to extreme scenarios where each task is equated to a service, there is little reuse of services across processes, and hence fine grained services proliferate and are too hard to manage (Jones, 2005). Meanwhile, entity-driven approaches are more suitable for SOA deployment in “green field” scenarios. However in the real world such situations are quite rare due to the legacy applications that need to be taken into account (Zimmermann, 2004). In addition, entity-driven approaches may lead to the design of services that are not aligned with business operations.

As a perspective for future work, we plan to design a hybrid methodology aiming at reconciling top-down entity-driven approaches with bottom-up business-driven approaches, so as to combine the benefits of both.

CONCLUSION

This paper addressed one of the currently perceived issues surrounding Web services, namely the lack of a sound methodology to demonstrate the actual business impact of Web services adoption in specific settings. The main contributions are: a) a process-oriented framework for systematic assessment of web service adoption opportunities including checklists and scoring tables; b) a structured set of identified critical information for Web services evaluation and deployment through business process models; and c) a mapping of this information types into ARIS constructs, thus enabling the representation of this information in a business process model.

The study has drawn on an extensive review of the literature as well as reported case studies and best practices. From these resources, a list of assessment criteria for potential application areas of Web services could be derived. These criteria were then tested through an e-Procurement scenario and refined through feedback obtained from focus groups.

Further research leading to the refinement, extension, and testing of the proposed assessment methodology is needed. In particular, the methodology could be extended by depicting trade-offs between benefits and risks of Web services deployment. Also, additional requirements for the methodology should be identified through further case studies, and the implementation of the proposed methodology in other tools than ARIS should be considered. Finally, it would be highly desirable to validate the proposed methodology through more case studies in order to identify and understand the risks and pitfalls of its application.

Another relevant direction for future work is the exploration of requirements that collaborative e-Business interactions impose on business process modelling and Web services as well as their implications. Issues that still need to be addressed and overcome include dealing with business trust, semantic heterogeneity and exceptions, all of which were identified as crucial during the focus groups.

Acknowledgments. An earlier version of this book chapter was published with the title "Process-oriented Assessment of Web Services" in International Journal of e-Business Research, vol. 2, n. 1, pp. 19-44, January 2006, IDEA Group Publishing. The authors are grateful to the participants of the focus groups who remain anonymous due to a confidentiality agreement. The work of the third author has been partly funded by a Queensland Government "Smart State Fellowship" co-sponsored by SAP Research.

REFERENCES

Adams, H., Gisolfi, D., Snell, J., & Varadan, R. (2002). *Best practices for Web services: Back to the basics, Part 2*: IBM Developer Works.

Becker J, Rosemann M and Von Uthmann C (2000) Guidelines of business process modeling, in van derAalst W, Sedel J and Oberweis A, Eds. *Business Process Management: Models Techniques and Empirical Studies*, pp. 30-49, Springer-Verlag, Berlin et al.,

Bibby, A., & Brea, C. (2003). Business Process Reengineering Through Web Services

A user-experience perspective. *Web Services Journal*.

Bielski, L. (2003). Web services goes live: One deployment at a time. *American Bankers Association. ABA Banking Journal, 95*(7), 49-58.

Burdett, D. (2003). *The Business Web Services Scorecard: Using Web Services for Business - the present and the future*. Commerce One Operations Inc.

Chen, M. (2003). Factors affecting the adoption and diffusion of XML and Web services standards for E-business systems. *International Journal of Human - Computer Studies [H.W. Wilson - AST], 58*(3), 259.

Christiansen, S. (2002, December 8-13 2002). *The Business Case for XML Web Services.* Paper presented at the XML Conference 2002, Baltimore.

Curtis B, Keller MI, Over J (1992). Process Modeling, *Communications of ACM* 35 (9), September.

Davis, R. (2001). *Business Process Modelling with ARIS: A practical guide*. London: Springer.

Erl T. (2005). *Service-Oriented Architecture (SOA): Concepts, Technology, and Design*, Prentice Hall PTR.

ebXML Business Process Team. (2002, April 2002). *Business Process and Business Document Modeling Worksheets and Guidelines*. Retrieved 01/06/2003, from http://www.collaborativedomain.com/standards/documents/bpWS-1.05-2%20DRAFT%20(May%2016).doc

Estrem, W. A. (2003). An evaluation framework for deploying Web Services in the next generation manufacturing enterprise. *Robotics and Computer-Integrated Manufacturing, In Press, Corrected Proof*.

Glushko, R. J., & McGrath, T. (2002). *Document Engineering for e-Business.* Paper presented at the Document Engineering 2002, McLean, Virginia, USA.

Hagel III, J., & Brown, J. S. (2002). *Orchestrating Business Processes - Harnessing the Value of Web Services Technology*. Retrieved 01/06/2003, from http://www.johnhagel.com/paper_orchestratingwebservices.pdf

Hagel III, J., Brown, J. S., & Layton-Rodin, D. (2002). *The Secret to Creating Value from Web Services today: Start simply*. Retrieved 01/06/2003, from http://www.johnhagel.com/paper_startsimply.pdf

Hammer, M., & Champy, J. (1993). *Reengieneering the Corporation. A Manifesto for Reengineering*. New York: Harper Business.

IBM. (2003). *IBM Patterns for e-business*. Retrieved 01/09/2003, from http://www-106.ibm.com/developerworks/patterns/

IDS Scheer AG. (2002). ARIS Methods (for Version 6).

Jones, S. (2005). *Enterprise SOA Adoption Strategies*, InfoQ Mini-Book Series, 2006. Retrieved 21/01/2007, from http://www.infoq.com/minibooks/enterprise-soa

Jost, W., & Wagner, K. (2002). The ARIS Toolset. In A.-W. Scheer, F. Abolhassan, W. Jost & M. Kirchmer (Eds.), *Business Process Excellence: ARIS in Practice* (pp. 16-31). Berlin Heidelberg New York: Springer.

Keeney, R. & Raiffa, H. (1993). *Decisions with Multiple Objectives : Preferences and Value Trade-offs*. Cambridge University Press.

Linthicum, D. S. (2002, 17/01/2002). *An Application Integration Value-Add? expoQ's David Linthicum on Web Services in EAI*. Retrieved 18/08/2003, from http://searchwebservices.techtarget.com/originalContent/0,289142,sid26_gci788646,00.html

Morgan, D. L. (1988). *Focus Groups as Qualitative Research* (Vol. 16). Newbury Park, California: SAGE Publications.

National Health Supply Chain Taskforce Interoperability Working Group. (2002). *Electronically Connecting Trading Partners: A Proposed Interoperability Standard*: National Health Supply Chain Taskforce Interoperability Working Group.

New Mexico State University - College of Agriculture and Home Economics.*1999-2003 Evaluation Handbook*. Retrieved 01/06/2003, from http://www.cahe.nmsu.edu/grants/evaluation/eval_section2.html#focus

Papazoglou, M. (2003). *Web Services and Business Transactions*.

Patricia Seybold Group. (2002). *An Executive's Guide to Web Services*: Patricia Seybold Group.

Remenyi, D., Money, A., Sherwood-Smith, M. & Irani, Z. (2000). *Effective Measurement and Management of IT Costs and Benefits*. Butterworth-Heinemann, 2nd edition.

Robins, B., Sleeper, B., & McTiernan, C. (2003). *Web Services Rules: Real-World Lessons from Early Adopters*: The Stencil Group.

Samtani, G., & Sadhwani, D. (2002). ROI and Web Services. In *Web Service Business Strategies and Architectures*: Expert Press.

Saulnier, C. F. (2000). Groups as data collection method and data analysis technique: Multiple perspectives on urban social work education. *Small Group Research, 31*(5), 607-627.

Scheer, A.-W. (1998a). *ARIS - Business Process Frameworks*. Berlin et al: Springer.

Scheer, A.-W. (1998b). *ARIS - Business Process Modelling*. Berlin et al: Springer.

Scheer, A.-W., Abolhassan, F., Jost, W., & Kirchmer, M. (2002). *Business Process Excellence: ARIS in practice*. Berlin Heidelberg New York: Springer.

Schmelzer, R., & Bloomberg, J. (2002, 18/06/2002). *Want to service-enable your enterprise? Model first!* Retrieved 01/06/2003, from http://searchwebservices.techtarget.com/tip/1,289483,sid26_gci833705,00.html

Scholler, D. (2003, 07/04/2003). *Web Services Architecture: Part 1 - Service Development Excellence*. Retrieved 01/06/2003, from http://www.ge-bac.com/article.php?story=20030407225246949

Singh, M. P., Huhns. M.N. Service-oriented Computing: Semantics, Processes, Agents. John Wiley and Sons. 2005.

Soderstrom, E., et al. (2002). Towards a framework for comparing process modeling languages. In *A. Banks Pidduck et al. (Ed.s), Proceedings of CAiSE 2002* (pp. 600-611). Heidelberg Berlin: Springer.

Sofaer, S., Kreling, B., Kenney, E., Swift, E. K., & Dewart, T. (2001). Family Members and Friends Who Help Beneficiaries Make Health Decisions. *Health Care Financing Review; Washington, 23*(1), 16.

The Stencil Group. (2002). *The Laws of Evolution: A Pragmatic Analysis of the Emerging Web Services Market*: The Stencil Group.

Wilkov, R. (2002). *How to Get Real Value fro Web Services... Today*. Retrieved 01/06/2003, from http://www.ebizq.net/topics/dev_tools/features/1567.html

Wright, D. (2002). *Beyond the Hype: How To Make Web Services Work for Your Organization, Part 2 - How to Begin to Use Web Services in Your Business*. Retrieved 01/06/2003, from http://www.lightshipinc.com/lightship/download/download.aspx?Title=BeyondHypePart2

Zimmermann, O., et al. (2004). *Elements of Service-Oriented Analysis and Design: An interdisciplinary modeling approach for SOA projects*, IBM. Retrieved 21/01/2007, from http://www-128.ibm.com/developerworks/webservices/library/ws-soad1/