Domain Based Identification and Modelling of Business Component Applications

Antonia Albani, Alexander Keiblinger, Klaus Turowski, and Christian Winnewisser

Chair of Business Information Systems (Wirtschaftsinformatik II)
University of Augsburg
Universitätsstraße 16, 86135 Augsburg, Germany
{antonia.albani, alexander.keiblinger, klaus.turowski, christian.winnewisser}@wiwi.uni-augsburg.de

Abstract. This paper presents a process for the design of domain specific business component applications to enforce the use of component based software technologies. Starting from the notion that the direct mapping of business processes to business components often leads to inadequate results, a methodology for identifying and refining business components based on the functional decomposition of an application domain will be presented. The usability of the resulting process will be shown with the example of a concept for strategic supply chain development, which extends the traditional frame of reference in strategic sourcing from a supplier-centric to a supply-chain-scope including the dynamic modelling of strategic supply-chains.

1 Introduction

The idea of software systems made up from pre-fabricated software components that could be exchanged via software component markets has been on the agenda of software engineering at least since McIlroy has outlined his vision in 1968 [15]. The underlying idea is to combine components from different vendors to an application which is individual to each customer and where the compositional plug-and-play-like reuse of black box components enables software component markets. Ideally, the advantages of both standard and individual software production are combined. Compositional reuse is a special kind of reuse technique as *generative* techniques [7] or *code and design scavenging* [19, pp. 25–28]. The principle of modular design that is underlying component based software systems is equally important for the discussion of the technological as well as the economic advantages of component based software systems. A rich literature on the general advantage of such systems exists, c.f. [4], [3], [20], [21], [23]. Modular systems have been described as the result of a functional decomposition [32] and the conception of modular systems has been thoroughly analysed by system theory.

According to [8, pp. 3–4], we define the term *component* as follows: A component consists of different (software-) artefacts. It is reusable, self-contained and marketable, provides services through well-defined interfaces, hides its implementation and can be deployed in configurations unknown at the time of development. A *business*

L. Kalinichenko et al. (Eds.): ADBIS 2003, LNCS 2798, pp. 30–45, 2003.

component is a component that implements a certain set of services out of a given business domain. Refer to [27 pp. 164–168] and [8] for an in depth discussion of various other component approaches given in literature.

For the concept of business components as described above, a standard approach for specification has been defined in the memorandum "Standardized Specification of Business Components" [31]. In section 2 the specification will be briefly explained. Whereas an established approach for the specification of business components already exists, no methodology for the identification of reusable, marketable and self-contained business components has been defined so far. Therefore, in section 3 a business component modelling process is proposed for the identification of suitable business components. Component based domain analysis and component based domain design with the resulting models are introduced and described in detail.

To illustrate and evaluate the proposed methodology, the business domain of strategic supply chain development is introduced. Based on this domain scope a business component model has been identified by applying the business component modelling process. The phases of the respective component based domain analysis will be described in section 4. In section 5 the component based domain design will be illustrated.

2 Business Components

For the use of business components it is necessary to *specify* them, i.e., standardise them in a domain dependent context and describe their interface and behaviour in a consistent and unequivocal way. Such a specification is crucial for the creation of component markets, since component buyers are unable to specify their demands or to cross check offered components against their demands without it [10].

The memorandum "Standardized Specification of Business Components" [31] is a recommendation of the working group *business components* at the German Informatics Society (GI). It aims to set a *methodical standard* for the specification of business components enabling a common understanding of component specifications. This is achieved by identifying the objects to be specified and by defining a notation mix that is standardised, accepted and agreed by all participating parties. This ensures the reusability of components and their exchange between companies and software developers can be simplified. The s*pecification* of a business component is defined as a complete, unequivocal and precise description of its external view. It describes which services a business component provides under which conditions.

Based on the ideas of [5], [30] and [29], the memorandum states that the specification of business components is to be carried out on different contract levels (see Fig. 1). Besides arranging the specifications' contents according to contract levels, a specific notation language is proposed for each level of abstraction.

In the context of systematic specification of business components it is helpful to use a well-known and well-accepted formal notation, which can be used on more than one contract level. A notation is called *formal* in this context if syntax and semantics of the notation are unequivocal and consistent. For this reason, formal notations seem to be particularly suitable for the specification of software contracts. A detailed characterisation of the different contract levels is given in [31].

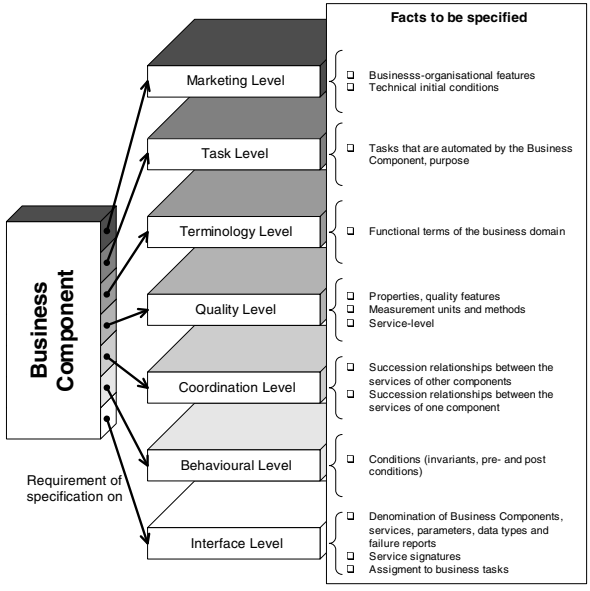


Fig. 1. Software contract levels and facts to be specified

3 Business Component Modelling Process

A precondition to component based development of application systems by using business components is a stable component model. In order to obtain stable business component models, a well defined derivation process for such components is necessary. Based on the fact that business components do not only satisfy the requirements for a single application system but rather for a family of systems – and therefore for a certain domain – the derivation process requires throughout all development phases the consideration of the specific business domain.

Domain Engineering [7 , p. 19–59, 19 , p. 159–169] aims at the development of reusable software and a large number of Domain Engineering processes exist, among the most important ones being [11, 26]. The methods mentioned contribute to different aspects of the Domain Engineering area as for example in identifying prominent or distinctive features of a class of systems or in defining maintainability and understandability characteristics of family of systems.

For business components additional prerequisites are required, the most important ones being reusability, marketability and self-containment for different domains. Therefore a new methodology for business components – the Business Component Modelling (BCM) process – is introduced next. In section 4 and 5 the BCM is used to define suitable business components for the domain of Strategic Supply Chain development (SSCD).

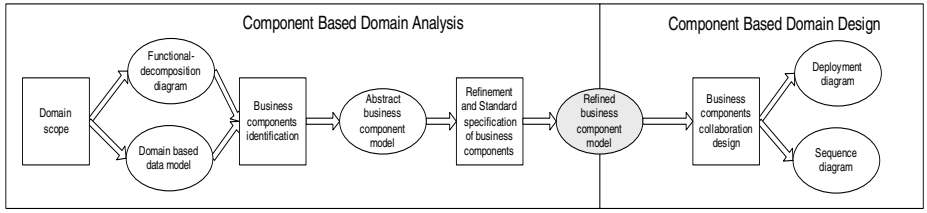


Fig. 2. Business Component Modelling process

An overview of the BCM process focusing on the *Component Based Domain Analysis* and *Component Based Domain Design* phases is given in Fig. 2. Rectangles denote sub phases of the process and ellipses contain the resulting diagrams and models of each sub phase.

In the following the single phases and their performed tasks are explained. During *domain scope* the domain of interest is identified, characterised and business processes with their functional tasks are defined. In addition data is collected to analyse the information objects and their relationships. Possible sources of domain information include existing systems in the domain, domain experts, handbooks, requirements on future systems, market studies, and so on. As a result of the first sub phase a *functional-decomposition diagram* and a *domain based data model* are generated. This information is fundamental for the next sub phase in the BCM process, namely the *Business Components Identification.*

Having defined the functional-decomposition diagram the most intuitive way of identifying business components would be the mapping of single business processes to business components and their functional tasks to business component services. Having experienced that, the resulting components are not satisfactory. They are often not reusable, self-contained or marketable as postulated in [31], and the quality of services does not satisfy criteria of performance engineering – e.g. adaptable, performant, maintainable – as described in [25, 24]. In order to optimise the process of identifying high quality, reusable and marketable business components the Business Components Identification (BCI) method is introduced.

BCI is based upon the Business System Planning (BSP) [1] method and has been modified for the field of business components identification. BCI takes as input the business tasks of a specific domain, as defined in the functional-decomposition diagram, and the domain based data model, both obtained from the domain scope phase. In a first step a matrix is built defining the relationships between the single business tasks and the informational data. The relationships are visualised inserting “C” and “U” in the matrix. “C” denotes that the data is *created* by the specific business task, and “U” denotes the *usage* of informational data by a given task. In changing the order of data and of business tasks and placing the “C” as far left and as up as possible in the matrix, groups of relationships can be recognised, see Table 2 and Table 3. These groups identify potential business components. If some “U”’s are outside of the groups, arrows are used to identify the data flow from one group to the other. The result of the BCI is an abstract business component model with some already defined dependencies between components. In section 4 the BCI method is explained by means of the domain of strategic supply chain development.

In the next sub phase each component of the abstract model needs to be refined and specified as stated in the standardised specification of business components docu-

ment. An overview of the specification levels has been given in section 2. The result is a *refined business component model* with all seven business component levels – marketing, task, terminology, quality, coordination, behavioural and interface – specified.

The refined business component model is the main input for the *Component Based Domain Design* and *Component Based Domain Implementation* phases. The component based domain design phase is explained next, whereas the implementation phase is not within the scope of this paper to be defined.

The sub phase *business components collaboration design* of the design phase is used to define the deployment of component instances on different systems and to identify dependencies in form of service calls between business components. As a result different kinds of diagrams are produced, e.g. deployment and sequence diagrams. These diagrams together with the refined business component model are the outcome of the BCM process and are the main input for developing component based information systems with high quality, reusable and marketable business components. The complete BCM process is summarised in Table 1.

Table 1. Summary of the BCM Process

BCM Phase	BCM sub phases	Performed Tasks	Results
Component Based Domain Analysis	Domain scope	Identification and characterisation of the domain	Functional-decomposition diagram
		Definition of the business processes and functional tasks of the domain	
		Data collection and definition of information objects	Domain based data model
		Identification of relationships between information objects	
	Business Component Identification (BCI)	Grouping of functional business tasks and informational object for the identification of business components	Abstract business component model
	Standard specification of business components	Specification of all business component levels (marketing, task, terminology, quality, coordination, behaviour, interface)	Refined business component model
Component Based Domain Design	Business components collaboration design	Definition of component instances	Deployment diagram
		Definition of dependencies between component instances	
		Identification of service calls between component instances	Sequence diagram

To illustrate the process of BCM it is applied in the following sections to the domain of strategic supply chain development detailing both phases, sub phases, tasks and the resulting diagrams and models.

4 Component Based Domain Analysis

The goal of the component based domain analysis is to generate a refined business component model with all business components being specified as defined in the memorandum for standardised specifications of business components.

4.1 Domain Scope – From Strategic Sourcing to Strategic Supply Chain Development

The relevance of the purchasing function in the enterprise has increased steadily over the past two decades. Till the 70ies, purchasing was widely considered an operative task with no apparent influence on long term planning and strategy development [16]. This narrow view was broadened by research that documented the positive influence that a targeted supplier collaboration and qualification could bring to a company's strategic options [2]. In the 80ies, trends such as the growing globalisation, the focus on core competencies in the value chain with connected in-sourcing and out-sourcing decisions, as well as new concepts in manufacturing spurred the recognition of the eminent importance of the development and management of supplier relationships for gaining competitive advantages. As a result, purchasing gradually gained a strategic relevance on top of its operative tasks [12].

Based on these developments, purchasing has become a core function in the 90ies. Current empiric research shows a significant correlation between the establishment of a strategic purchasing function and the financial success of an enterprise, independent from the industry surveyed [6, p. 513]. One of the most important factors in this connection is the buyer-supplier-relationship. At many of the surveyed companies, a close cooperation between buyer and supplier in areas such as long-term planning, product development and coordination of production processes led to process improvements and resulting cost reductions that were shared between buyer and suppliers [6, p. 516].

In practice, supplier development is widely limited to suppliers in tier-1. With respect to the above demonstrated, superior importance of supplier development we postulate the extension of the traditional frame of reference in strategic sourcing from a supplier-centric to a supply-chain-scope, i.e., the further development of the strategic supplier development to a strategic supply chain development. This re-focuses the object of reference in the field of strategic sourcing by analysing supplier networks instead of single suppliers. Embedded in this paradigm shift is the concept of the value network that has been comprehensively described, e.g., [14], [33], [18].

The main reason for the lack of practical implementation of strategic supply chain development can be found in the high degree of complexity that is connected with the identification of supply chain entities and the modelling of the supply chain structure, as well as the high coordination effort, as described by [13].

In a next step the functional tasks of strategic supply chain development have to be defined. Those tasks will be derived from the main tasks of strategic sourcing. The most evident changes are expected for functions with cross-company focus. The functional tasks of strategic supply chain development have been illustrated in a function decomposition diagram (Fig. 3).

Processes and tasks that will be included in the component model have been shaded. Following, selected tasks will be described. The focus will be on changes to current tasks of strategic purchasing.

Task "Model strategic supply chain": The process *supplier selection* from strategic purchasing undergoes the most evident changes in the shift to a supply chain centric perspective. The expansion of the traditional frame of reference in strategic sourcing requires more information than merely data on existing and potential suppliers in tier-1. Instead, the supply chains connected with those suppliers have to be identified and

evaluated, e.g., by comparing alternative supply chains in the production network. As a consequence, the task *supplier selection* is part of the process that leads to the modelling of strategic supply chains.

The perception of the supply chain as dynamic network constitutes the basis for the identification of strategic supply chains. To visualise the network in a structured way, a specific strategic demand is communicated to existing and/or potential suppliers in tier-1. Subsequently, those tier-1 suppliers report their own demand to their respective suppliers. The requested information are split-lot transferred the other way round, aggregated and finally visualised as supply network, in which each participant of the supply chain constitutes a network hub. Without information technology, such an approach could not be realised.

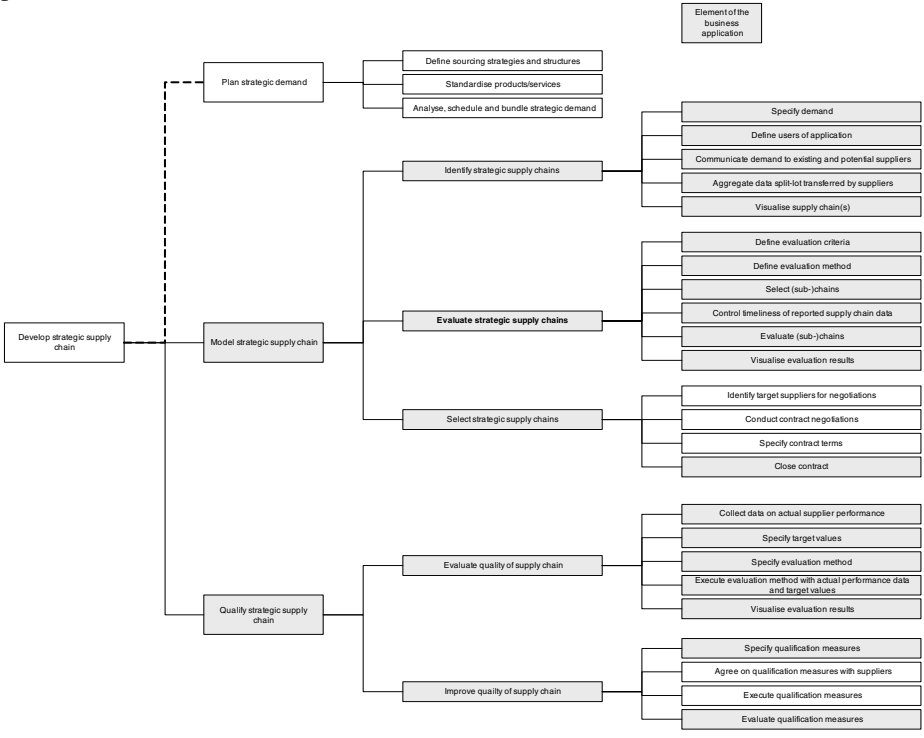


Fig. 3. Functional decomposition diagram for the supply chain development

According to the assumptions described above, the rating of supply chains requires the evaluation of networks instead of single suppliers. There has been preparatory work on evaluation methods for business networks (e.g., [28, 22]) on which we have based initial methods for the described application. However, there is need for further research, especially in the area of aggregation of incomplete information.

Task "Qualify strategic supply chain": In addition to the selection of suitable supply chains and composition of alternative supply chains, the performance improvement of strategically important supply chains is one of the major goals of strategic supply chain development. Main prerequisite is the constant evaluation of the actual performance of selected supply chains by defined benchmarks. The application should

support respective evaluation methods and enables the user to identify imminent problems in the supply chain and to initiate appropriate measures for qualification of supply chain partners.

This is important because of the long-term character of strategic supply chain relationships. As a result of the long-term perspective, qualification measures – e.g., along the dimensions product, processes and management abilities – require deployment of resources on the buyer side as well. Because of this effort, problems in the supply chain should be identified proactively and qualification measures should be tracked.

Task "Plan strategic demand": Strategic planning, i.e., analysing, scheduling and grouping of long-term demand, primarily affects intra-company processes that will not change significantly by switching to a supply chain perspective in strategic purchasing and therefore will not be automated in a first step.

Data Model

Having identified and characterised the domain of strategic supply chain development and defined the business processes and tasks of that domain, it is necessary to determine the relevant information objects involved in the modelling of the strategic supply chain development. The resulting data model is listed as UML class diagram [17 p. 294] in Fig. 4.

Starting from the demand of a *buyer* it is relevant to collect data not only from the suppliers in tier-1, but from all suppliers in tier-n. A *demand* consists of *services*, *material groups* and more which can be generalised to a *demand category*. To each demand category different *characteristics* can be added, as for example *method of production*, *service level*, *time*, *volume* etc. A whole supply chain specified by a demand is a network of suppliers providing information to the buyer which is used for the development of the supply chain. This network of suppliers is represented in the data model as a *complex monitoring object* whereas a single supplier is represented as an *elementary monitoring object*. With the affiliation of elementary monitoring objects to a complex monitoring object and with the identification of predecessors and successors of such elementary monitoring objects the whole supply chain is defined. At a particular time each elementary object provides information about the *product range, bill of material, financial data* or more. This information is known as *supplier generated data*. In addition the buyer generates own data, called *buyer generated data*, specifying the performance of the supplier, respectively of the elementary monitoring object, as termed in the data model. Examples for data generated by the buyer are *target performance data* and *actual performance data*. Target performance data are guidelines for the supplier, and the actual performed data are the work performed measured by the buyer. The buyer holds with the acquisition of supplier generated data and with the definition and the measurements of performance data for all different complex monitoring objects the information needed to evaluate the supply chain. Different evaluation methods are defined by different evaluation criteria.

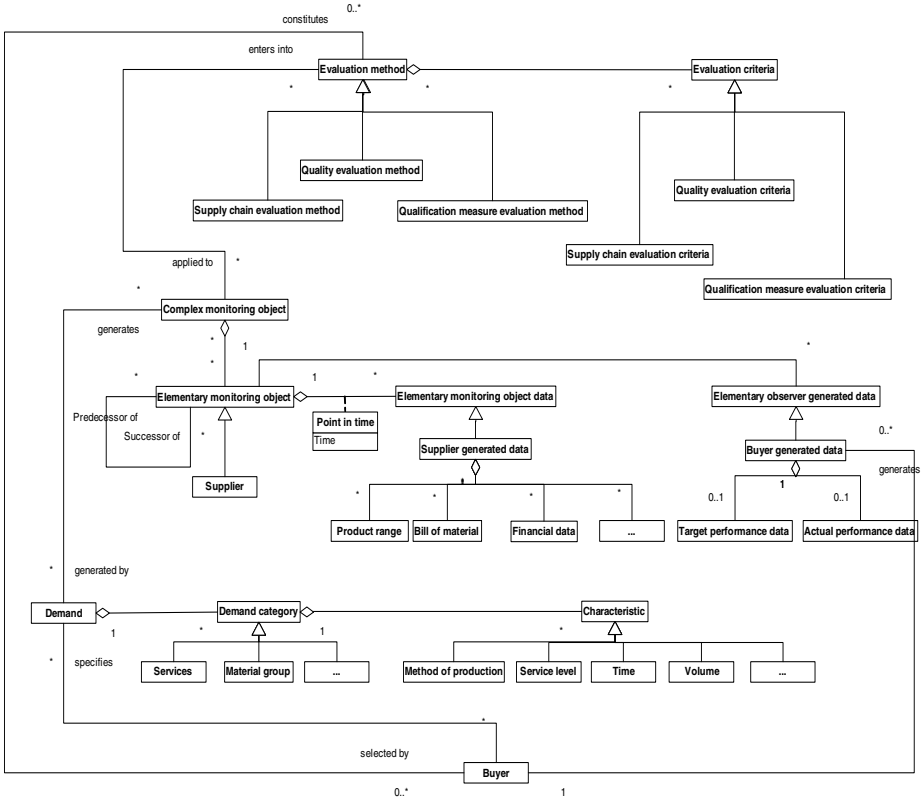


Fig. 4. Data model for the domain of strategic supply chain development

The data model and the functional-decomposition diagram derived from the domain scope are the main information needed for the identification of high quality business components explained in the next subsection.

4.2 Business Components Identification (BCI)

With the BCI method relationships between business tasks and information objects are defined and grouped. In Table 2 the relationships for the domain of strategic supply chain development are shown.

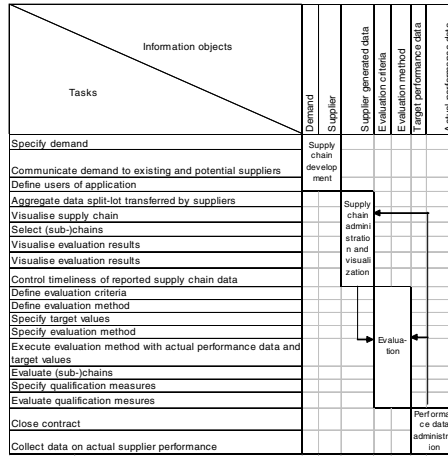
The business tasks are gained from the functional-decomposition diagram (Fig. 3) and are listed left on the table. Relevant groups of data are gained from the data model (Fig. 4) and utilised for the BCI process as information objects, listed on top. Such information objects are demand, supplier, supplier generated data, evaluation criteria, evaluation method, target performance data and actual performance data.

Table 2. Grouping of tasks and information objects

Tasks \ Information objects	Demand	Supplier	Supplier generated data	Evaluation criteria	Evaluation method	Target performance data	Actual performance data
Specify demand	C						
Communicate demand to existing and potential suppliers	U	U					
Define users of application		C					
Aggregate data split-lot transferred by suppliers			U				
Visualise supply chain			U				
Select (sub-)chains			U				
Visualise evaluation results			U			U	U
Visualise evaluation results			U				
Control timeliness of reported supply chain data			U				
Define evaluation criteria				C			
Define evaluation method					C		
Specify target values				C			
Specify evaluation method					C		
Execute evaluation method with actual performance data and target values				U	U	U	U
Evaluate (sub-)chains			U	U	U		
Specify qualification measures				C	C		
Evaluate qualification mesures				U	U	U	U
Close contract						C	
Collect data on actual supplier performance							C

An example relationship would be the usage "U" of supplier generated data for the visualisation of the supply chain. Four areas result for the domain of strategic supply chain development in changing the order of tasks and information objects in the matrix as defined in chapter 3. The four areas are potential business components. The first business component offers services for the specification of the demand and for the definition of application users. The component therefore provides services for the *supply chain development*. The second business component is responsible for the *supply chain administration and visualisation* in aggregating and managing the data received and in providing visualisation services. The *evaluation* component provides services in the area of evaluation methods and criteria, and the component on the right is responsible for the *performance data administration.* The "U"'s outside the grouped areas are substitute by arrows defining the dependencies of the single business components, shown in Table 3.

Table 3. Business components identified using the BCI method



4.3 Refinement and Standard Specification of Business Components

The business component *supply chain administration and visualisation* gained from the BCI method needs to be partitioned in two components, the *supply chain administration* and the *visualisation and selection* component, in order to separate the administration of data from the visualisation of data. The separation from data and presentation is common within software development e.g. by patterns as Model View Control (MVC) [9] and can be used in a more abstract way also for business component development.

Two additional components are added to the business component model, namely the *communication manager* and the *communication*. These ones are attained from infrastructure requirements and not from business requirements and belong to the component system framework. Therefore they are not resulting from the BCI method.

The business component model gained from the BCI method and refined as just mentioned is shown in Fig. 5. For each component listed a specification for all levels of abstraction – marketing, task, terminology, quality, coordination, behavioural and interface – as defined in section 2, needs to be given. The description of the detailed specification of all levels for each single component is not within the scope of this paper.

Interesting for this paper instead is the specification of the services provided by the identified business components, defined in the task level of the specification memorandum. Single business tasks need therefore to be assigned to component services. The mapping for the strategic supply chain development domain is shown in Table 4 and is defined according the memorandum of standardised specification of business components.

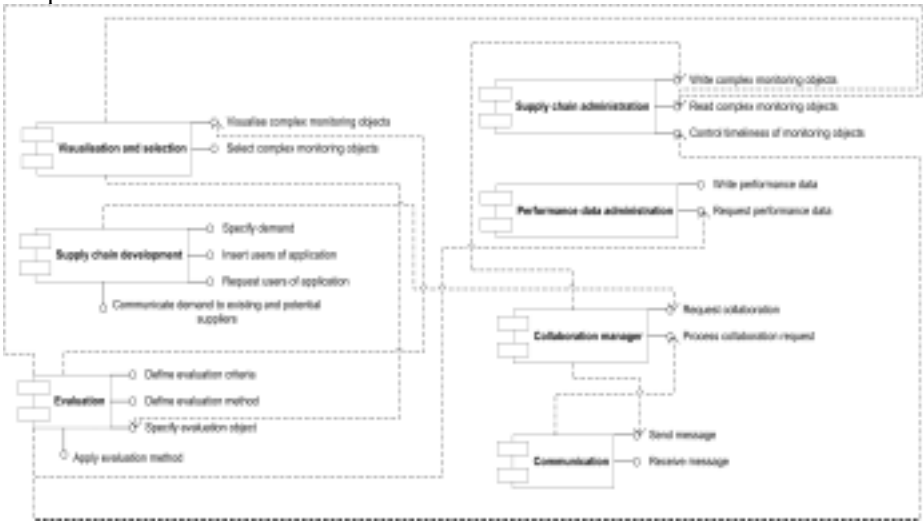


Fig. 5. Refined business component model

Some business tasks are mapped one to one to business component services, e.g. *specify demand* or *communicate demand to existing and potential suppliers*. However most of the business tasks are mapped to a superordinate more abstract defined service, e.g. the business tasks *visualise supply chain* and *visualise evaluation results* are

mapped to the service *visualise complex monitoring objects*. Fig. 5 shows the refined business component model with all services defined for each component and with the dependencies between the individual business components represented by arrows.

With the definition of a very abstract data model for the strategic supply chain development domain, as given in Fig. 4, powerful, high quality, reusable and self-contained business components can be identified which can be used with minimal integration effort and different parameterisations for different application systems.

Table 4. Mapping of business tasks to component services

Task	Service
Define users of application	Insert users of application
Specify demand	Specify demand
Communicate demand to existing and potential suppliers	Communicate demand to existing and potential suppliers
Aggregate data split-lot transferred by suppliers	Process collaboration request
Visualise supply chain(s)	Visualise complex monitoring objects
Define evaluation criteria	Define evaluation criteria
Define evaluation method	Define evaluation method
Select (sub-)chains	Selection of complex monitoring objects
Control timeliness of reported supply chain data	Control timeliness of monitoring objects
Evaluate (sub-)chains	Apply evaluation method
Visualise evaluation results	Visualise complex monitoring objects
Identify target suppliers for negotiations	*(execute manually)*
Conduct contract negotiations	*(execute manually)*
Specify contract termn	*(execute manually)*
Close contract	Write performance data
Collect data on actual supplier performance	Write performance data
Specify target values	Define evaluation criteria
Specify evaluation method	Define evaluation method
Execute evaluation mehtod with actual performance data and target valu	Apply evaluation method
Visualise evaluation results	Visualise complex monitoring objects
Specify qualification measures	Write performance data
Agree on qualification measures with suppliers	*(execute manually)*
Execute qualification measures	*(execute manually)*
Evaluate qualification measures	Apply evaluation method

5 Component Based Domain Design

The goal of the component based domain design is to elaborate different diagrams providing more information about instantiations and collaboration of the different components. Therefore two types of diagrams are appropriate, a deployment and a sequence diagram. Both are presented in the following subsection using the Unified Modelling Language [17 p. 362 and 432] notation.

5.1 Business Components Collaboration Design

For the domain of supply chain management different systems exist where the components are used, namely the producer system and all suppliers systems. Fig. 6 shows the deployment of business component instances on these systems.

Three systems are shown, one producer and two supplier systems. The producer holds all components belonging to the strategic supply chain development system including the supply chain development and the evaluation components. The suppliers utilise those components necessary for recording and sending the own data to the producer and for acquiring the data from their own suppliers. For a more coherent display not all component services and dependencies are shown. The dependencies and services calls are presented by arrows. The dependencies and the dataflow are given by means of an example and are illustrated in Fig. 7.

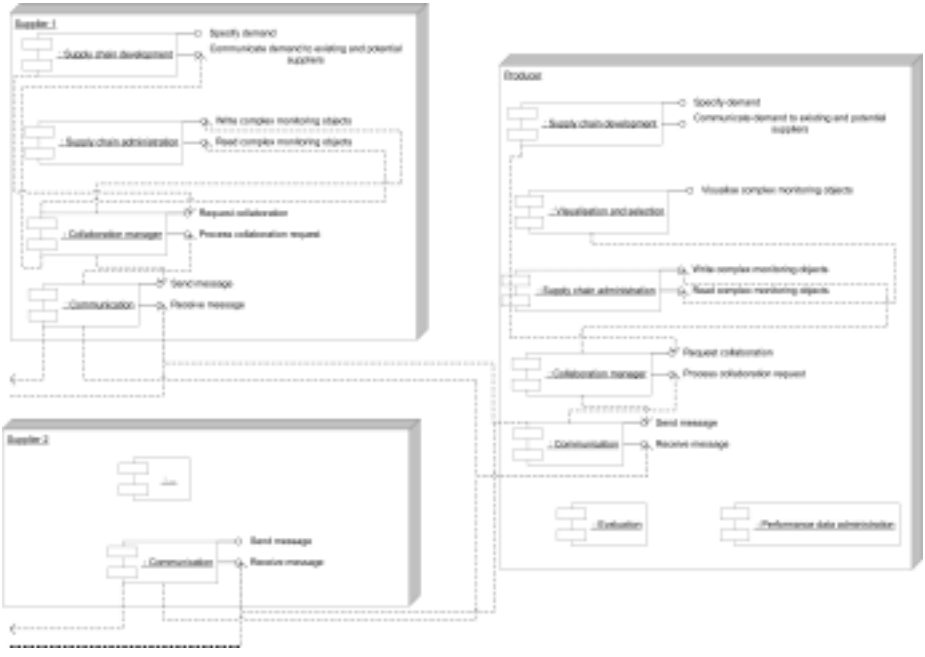


Fig. 6. Instantiation and deployment diagram

For the strategic development of a supply chain, defined by a demand, it is essential to collect complete data not only from the supplier in tier-1 but also from all suppliers in tier-n in order to be able to model and evaluate existing and alternative supply chains. Using the services of the supply chain development component a producer is able to specify the demand and to communicate the demand to all suppliers in tier-n. Triggered by that request, the supply chain development component accesses the service *request collaboration* of the collaboration manager component which uses the *send message* service of the communication component in order to send the demand to the suppliers in tier-1. The service requests of the different components are visualised in Fig. 7 by arrows. The communication components of the suppliers in tier-1 receive the message sent by the producer and forward the request to their collaboration managers accessing the *process collaboration request* service. Each collaboration manager uses the *communicate demand to existing and potential suppliers* service of the supply chain development component to forward the demand to exist-

ing and potential suppliers. The collaboration manager and the communication components are responsible for communication between the systems.

Supplier 1, having received the information data from supplier 3, stores the data using the service *write complex monitoring objects* in its own system. This information together with information about all the suppliers is sent back to the producer. At any time the producer receives the aggregated information from its suppliers in tier-1 and their suppliers. This information is given to the supply chain administration component in form of a complex monitoring object. Each user can then request to *visualise the complex monitoring object*. The complete supply chain is presented containing all information about the single supplier nodes. The producer is therefore able to evaluate and to develop the complete supply chain according to its requirements.

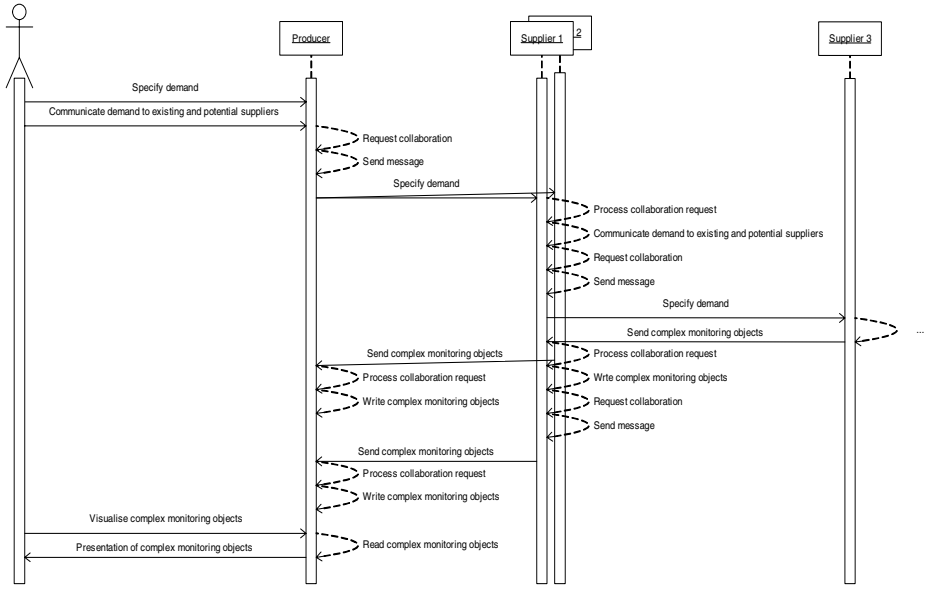


Fig. 7. Sequence diagram

6 Conclusions

Having illustrated the necessity of building advanced information system using business components, a detailed explanation of business components has been given, explaining all seven contract levels, which need to be defined in order to specify a business component in a standard way. The identification of domain based business components which are reusable, self-contained and marketable is not as intuitive. Therefore the Business Components Modelling process has been introduced, explaining the single phases, sub phases and the resulting diagrams used in order to achieve a maximum result. With the BCM method high quality components can be identified and reused in different systems by different parameterisations and with view integration effort.

The BCM method has been verified for the domain of strategic supply chain development, which extends the traditional frame of reference in strategic sourcing from a supplier-centric to a supply-chain-scope including the dynamic modelling of strategic supply chain. All phases and sub phases for that domain have been detailed and the resulting diagrams have been explained.

Further work for the validation and refinement of the proposed methodology is required. In particular additional business domains need to be examined to obtain a broader assessment base. Additionally, the BCI process needs to be extended for the identification of infrastructural business components.

References

[1] *Business Systems Planning-Information Systems Planning Guide*, International Business Machines, Atlanta, 1984.
[2] D. Ammer, *Materials Management*, Homewood, 1968.
[3] C. Y. Baldwin and K. Clark, *Design Rules: The Power of Modularity*, MIT Press, Cambridge (Mass.), London, 2000.
[4] C. Y. Baldwin and K. Clark, *Managing in an age of modularity*, Harvard Business Review, 75 5 (1997), pp. 84–93.
[5] A. Beugnard, J.-M. Jézéquel, N. Plouzeau and D. Watkins, *Making Components Contract Aware*, IEEE Computer, 32 (1999), pp. 38–44.
[6] A. S. Carr, Pearson, J.N., *Strategically managed buyer - supplier relationships and performance outcomes*, Journal of Operations Management, 17 (1999), pp. 497–519.
[7] K. Czarnecki and U. Eisenecker, *Generative programming : methods, tools, and applications*, Addison Wesley, Boston, 2000.
[8] K. Fellner and K. Turowski, *Classification Framework for Business Components*, in R. H. Sprague, ed., *Proceedings of the 33rd Annual Hawaii International Conference On System Sciences*, IEEE, Maui, Hawaii, 2000.
[9] E. Gamma, R. Helm, R. Johnson and J. Vlissides, *Design patterns: elements of reusable object-oriented software*, Addison-Wesley, Reading, Mass., 1995.
[10] H. Hahn and K. Turowski, *General Existence of Component Markets*, in R. Trappl, ed., *Sixteenth European meeting on Cybernetics and Systems Research (EMCSR)*, Vienna, 2002, pp. 105–110.
[11] K. C. Kang, *Feature-Oriented Domain Analysis (FODA) feasibility study*, Carnegie Mellon University Software Engineering Institute, Pittsburgh, Pa., 1990.
[12] L. Kaufmann, *Purchasing and Supply Management - A Conceptual Framework*, *Handbuch Industrielles Beschaffungsmanagement*, Hahn, D, Kaufmann, L. (Hrsg.), Wiesbaden, 2002, pp. 3–33.
[13] D. M. Lambert and M. C. Cooper, *Issues in Supply Chain Management*, Industrial Marketing Management, 29 (2000), pp. 65–83.
[14] T. W. Malone, Lautbacher, R.J, *The Dawn of the E-Lance Economy*, Harvard Business Review (1998), pp. 145–152.
[15] M. D. McIlroy, *Mass Produced Software Components*, in P. Naur and B. Randell, eds., *Software Engineering: Report on a Conference by the NATO Science Committee*, NATO Scientific Affairs Devision, Brussels, 1986, pp. 138–150.
[16] R. McIvor, Humphreys, P., McAleer, E., *The Evolution of the Purchasing Function*, Journal of Strategic Change, Vol. 6 (1997), pp. 165–179.
[17] OMG, *OMG Unified Modelling Language Spezification Version 1.4*, 2001.

[18] C. Rautenstrauch, H. Tangermann and K. Turowski, *Manufacturing Planning and Control Content Management in Virtual Enterprises Pursuing Mass Customization*, in C. Rautenstrauch, K. Turowski and R. Seelmann-Eggebert, eds., *Moving into Mass Customization - Information Systems and Management Principles*, Springer Verlag, Dubai, 2001, pp. 103–118.

[19] J. Sametinger, *Software engineering with reusable components*, Springer, Berlin ; New York, 1997.

[20] R. Sanchez, *Strategic flexibility in product competition*, Strategic Management Journal 16 (1995), pp. 135–159.

[21] R. Sanchez and J. T. Mahoney, *Modularity, flexibility and knowledge management in product and organization design*, Strategic management Journal, 17 (1996), pp. 63–76.

[22] M. Sawhney, Zabin, J., *The Seven Steps to Nirvana: Strategic Insights into eBusiness Transformation*, New York, 2001.

[23] M. A. Schilling, *Toward a general modular systems theory and its applications to interfirm product modularity*, Academy of Management Review, 25 (2000), pp. 312–334.

[24] A. Schmietendorf and A. Scholz, *Spezifikation der Performance-Eigenschaften von Softwarekomponenten*, *Modellierung und Spezifikation von Fachkomponenten - Workshop im Rahmen der MobIS 2000*, Siegen, 2000, pp. 41.

[25] A. Schmietendort and R. Dumke, *Spezifikation von Softwarekomponenten auf Qualitätsebenen*, *2. Workshop Modellierung und Spezifikation von Fachkomponenten (im Rahmen der vertIS 2001)*, Universität Bamberg, 2001, pp. 113–123.

[26] M. Simos, D. Creps, C. Klinger, L. Levine and D. Allemang, *Organization Domain Modeling (ODM) Guidebook*, 1996.

[27] C. Szyperski, *Component software: beyond object-oriented programming*, Addison-Wesley, Harlow, 1998.

[28] D. Tapscott, Ticoll, D., Lowy, A., *Digital Capital: Harnessing the Power of Business Webs*, Boston, 2000.

[29] K. Turowski, *Spezifikation und Standardisierung von Fachkomponenten*, Wirtschaftsinformatik, 43 (2001), pp. 269–281.

[30] K. Turowski, *Standardisierung von Fachkomponenten: Spezifikation und Objekte der Standardisierung*, in A. Heinzl, ed., *3. Meistersingertreffen*, Schloss Thurnau, 1999.

[31] K. Turowski, ed., *Vereinheitlichte Spezifikation von Fachkomponenten*, Arbeitskreis 5.3.10 der Gesellschaft für Informatik, Augsburg, 2002.

[32] K. T. Ulrich, *The role of product architecture in the manufacturing firm*, Research Policy, 24 (1995), pp. 419–440.

[33] H.-J. Warnecke, *Vom Fraktal zum Produktionsnetzwerk. Unternehmenskooperationen erfolgreich gestalten*, Berlin, 1999.