# Business Activities in an Industrial Context

Boris Gruschko[1], Friedrich H. Vogt[1], Simon Zambrovski[1]

[1] Hamburg University of Technology
Telematics Institute
Schwarzenbergstrasse 95
D-21073 Hamburg
Germany

**ABSTRACT**

Recent developments in IT industry lead to the notion of a Service, as an abstraction for an encapsulated business activity, hidden behind a well-defined interface. This abstraction fosters the view of complex business activities, as interactions between Services, allowing for an agile business activity modification. Furthermore, a breakdown of the monolithic business activity, into cooperating services, enables a simpler identification of Service's capabilities, needed to accomplish given business goal. This fosters a clearer view of the overall process, in regard to outsourcing opportunities for external allocation of Services. The description of Services and collaborations between them, leads to the definition of Service Oriented Architecture (SOA). In our understanding of SOA, a business activity is a set of messages being exchanged between Services, to enable the achievement of a common business goal.

To allow for a correct business activity execution, well-defined choreography of, and coordination between, participating Services is necessary. To enable the coordination, the participants (Services) of a business activity have to exchange messages, representing the coordination flow. Thus, the services are being controlled by a coordination protocol. It is not advisable to create a coordination protocol from scratch, due to the large number of pitfalls concealed in this undertaking. An amount of well understood and researched protocols exists, ready for usage during business activity coordination. The utilization of standardized protocols raises the safety of the resulting system. Nevertheless, it is a non-trivial task to identify the appropriate protocols and use them as distributed building blocks for the emphasized business tasks.

In this paper we show an approach to business activity modelling, with emphasis on identification of Services, messages being exchanged between them, collaboration scenarios and coordination protocols to be considered for the fulfilment of those scenarios. The modelling technique has been developed during business process analysis at Lufthansa Technik AG. An airframe related components overhaul process has been researched, in order to advise the concerned department, on application of loosely coupled systems. The analysis provides a basis for the inception of a SOA-based system, for the handling of airframe related components. The presented technique is not limited to the aerospace industry, but constitutes a generic approach to Service-Oriented Modelling. The main focus of this technique is the transformation from traditional, activity-based model, towards a model based upon message exchange between the participating Services. Further transition leads to the identification of collaboration patterns involved in the business activity. Given those patterns it is possible, to prove the applicability of standardized coordination protocols to the particular collaboration. We derive a methodology for structural approach to the management of business activities in SOA context, based on our experience gathered during the described analysis.

**Keywords:** Service Oriented Modelling, Business Process Management, Modelling Methodology

## 1 INTRODUCTION

The notion of a virtual enterprise has been discussed for a considerable amount of time. In our opinion a virtual enterprise represents a service, with a clearly defined interface, through which the customer interacts with the enterprise. Behind the scenes the virtual enterprise, as the name implies, consists of many interoperable and interconnected services. All of these services execute their parts of the overall activity, in order to achieve a common business goal.

The achievement of the enterprise's business goal, in most real world scenarios, requires intermediate allocation and release of resources. Respecting this fact, a graceful handling of intermediate application specific exceptions is needed. Thus, the establishment of coordination between the participating services is required, to allow for a successful business goal achievement.

The context of the presented business activity is the overhaul of airframe related components at Lufthansa Technik AG in Hamburg. Lufthansa Technik AG is an MRO[1] provider for aircrafts and related components. The overhaul of airframe related components requires the participation of a substantial number of services with diversified specializations. The services specializations vary from mechanical work being done on the component, to the inception of complex engineering solutions for untypical damages at the components to be repaired. The maintenance of airframe related components itself, is a part of the overall activity of aircraft overhaul, thus this service is itself a part of a larger service.

The development and analysis of distributed business activities require systematic methods, in order to achieve the safety of their execution. A major problem is the determination of the right coordination protocol for the task at hand. In our opinion, it is advisable to use generic coordination protocols between participating services. To reason about the protocol's applicability, the right level of abstraction has to be found. The finding of the right abstraction level is a two sided problem. Firstly, the owners of the business domain need a terminology, they are comfortable with. On the other hand, the existing protocols have to expose their (more abstract) terminology, to the business domain owners, to allow for judgement on protocol's applicability. These two terminologies have to be matched upon one another, to enable the protocol selection. Further improvement of the results of this matching is to be achieved by the application of formal methods. Firstly, the formal methods help to determine whether the described activity and collaborations between the partaking actors do achieve, a desirable activity outcome, under modelled conditions. Secondly, the formals models in the business and technical domains can be matched upon one another, to guarantee, that the selected coordination protocol ensures the outcomes of business activity, or to ease coordination protocol determination.

During our analysis of the airframe related components overhaul, we derived a methodology, to allow for the formulation of business domain safety properties[9]. The safety properties constitute the essence of the business activity, thus the verification, whether these safety properties are being guaranteed by the coordination protocol under consideration, delivers the desired answer about protocol's applicability.

## 2 EXAMINED ACTIVITY

The examined activity, of airframe related components overhaul, has the overall business goal of regaining component's airworthiness. The overall activity has two possible terminations. The component is either airworthy, or it has to be disassembled. In the first case, the component's useful life goes on, in the second, it is ended.

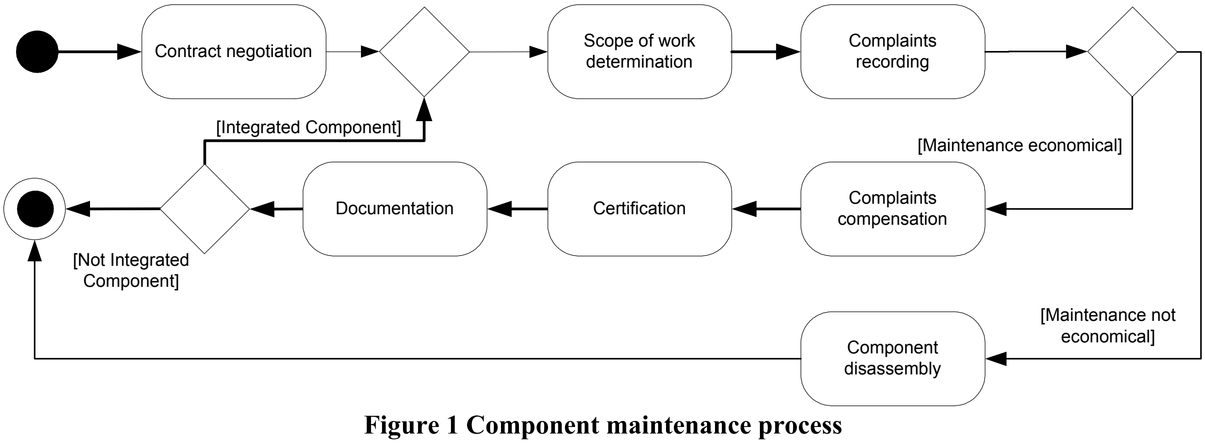


**Figure 1 Component maintenance process**

[1] Maintenance, Repair and Overhaul

Figure 1 shows an overview of the component maintenance process. Firstly, a commercial contract with the customer has to be negotiated. The contract may concern a single maintenance event, or a series of maintenance events for one or many components. If the customer and the maintenance event executor agree upon a continuous contract, the customer becomes an integrated customer. In this case, the contract's terms are not to be negotiated for every maintenance event. Following the contract negotiation, the scope of work is being determined. The scope of work contains the routine tasks to be performed upon the component. These tasks are routine inspection of the component. This preliminary scope of work is prescribed by the CMM[2] and is certified by the authorities like FAA[3] or LBA[4]. After the scope of work has been negotiated, the maintenance event executor proceeds with inspections. Complaints found during these inspections, are being recorded, alongside the compensating action, needed to be performed, to regain component's airworthiness.

Up to this point in the activity execution, the activity proceeds in a linear manner. Thus, there are no decisions to be met, which could terminate the activity. After the scope of work has been identified, the economical feasibility of the maintenance event is being examined. This activity involves the customer, who is responsible for the final decision, upon the cancellation or further execution of the maintenance event. If the maintenance is deemed economical, the event's execution proceeds with complaints compensation, component's certification and documentation of the event. Otherwise the component is being disassembled, or shipped to customer without further actions being taken.

## 3 EXAMINED ACTIVITY ANALYSIS

The examined process contains numerous caveats in regard to correct and economical execution. Firstly, the matter of airframe related components maintenance is highly regulated. The regulations are being constantly revised. Thus, changes in practices applied to airframe related component have to be certified by the concerned authorities. Secondly, the MRO market is highly competitive, thus the economical efficiency of components maintenance is a matter of highest concern to the customer and maintenance event executor alike. To ensure the economical efficiency, the maintenance event executor may be tempted, to outsource parts of the process, lying beyond it's core expertise. Outsourcing of parts of the activity to be executed still requires the maintenance event executor, to fulfill all regulations required by the authorities. Thus a close monitoring and information gathering becomes necessary across enterprise borders. This raises the question of requirements formulation. In this chapter we describe our proposal for such requirements formulation in terms of coordination protocol's safety properties. First we describe some of the outsourcing potentials of the described activity, and then we proceed, depicting a way, to ensure the correct execution of the overall activity, despite parts of it being outsourced.

### 3.1 Outsourcing potential

Eventually every step of the described activity can be outsourced. An enterprise like Lufthansa Technik AG however is interested, in retaining the execution of the core aspects of the overhaul, within enterprise borders. Nevertheless, the outsourcing within the enterprise itself is essential, because it is raising the overall transparency of the process.

The most likely candidates, in regard to outsourcing are second tier activities like material provision, contract acquisition and documentation archiving.

### 3.2 Correctness enforcement

Examination of the maintenance activity reveals conditions, which are to be fulfilled, after or during the maintenance event execution. These observations have to hold, regardless of outsourcing of the parts of the overall business activity. These properties are the core part of the business domain. Their origins are the regulations existing in the business domain[5] or the wishes of the business domain owners, like customer or maintenance event executor.

In case of airframe related components maintenance, the business domain safety properties are:

- Component mounted upon aircraft => Component certified airworthy (3.1)

---

[2] Component Maintenance Manual
[3] Federal Aviation Administration
[4] Luftfahrt Bundesamt (German equal of FAA)
[5] In this case, the regulations are constituted in the CMM

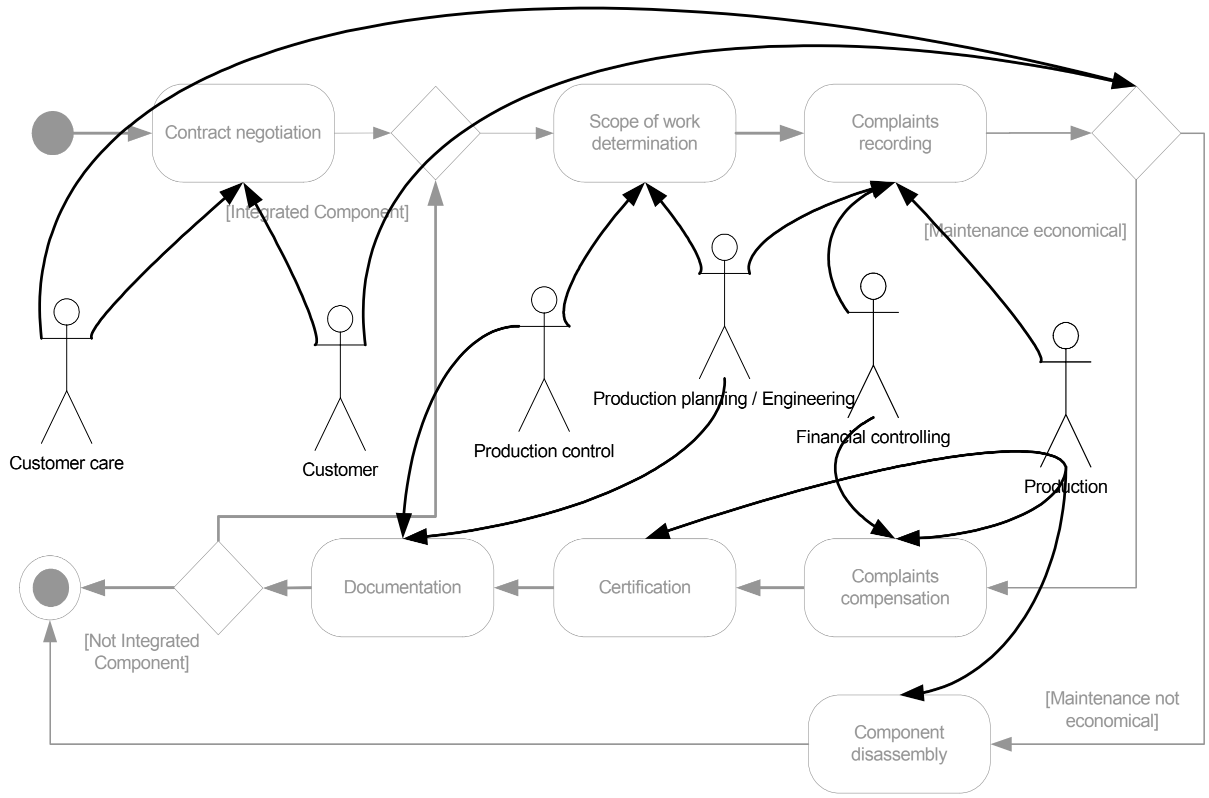


**Figure 2 Actors assignment.**

- Component dismantled => Customer rejected maintenance AND Customer ordered disassembly. (3.2)

The described view of the overall business activity reveals two major problems, of the correct distributed business activity execution enforcement. To allow for the correct execution, the fulfillment of the properties is to be assured. Second problem is the assurance of properties catalogue's completeness.

To assure that the identified properties are not being violated, we propose the usage of a standardized coordination protocol, between participating actors. To enable the usage of a generic coordination protocol, the identified properties have to be matched, upon the safety properties of the coordination protocol. The description of protocol's safety properties result from protocol examination like the work on WS-AtomicTransaction[7] protocol given in [1]. The exact process of properties matching is subject to ongoing research at the Hamburg University of Technology.

The problem of properties catalogue's completeness is closely related to the methods used during process analysis phase. During this phase, a model of the business activity under consideration is being created. From this model, the properties of the overall system are being derived. Thus, the model has to generalize the business activity in such a way that the essence of the considered business activity becomes visible, while preserving all correctness relevant properties of the business domain. The next section presents a modeling methodology, developed during the analysis at Lufthansa Technik AG.

## 4 PROPOSED MODELLING METHODOLOGY

The emphasis of the proposed methodology is the disclosure of services taking part in the modeled activity and interactions between them. From these interactions, the safety properties the coordination protocol has to meet will be derived. In our understanding, a service is an actor in the overall business activity executing system.

### 4.1 Activity description

During the analysis of the airframe related components overhaul, the activity diagrams of the UML proved themselves, as a description, well understood by the owners of the business domain. Figure 1 shows an activity diagram of the described business activity. This diagram however does not reveal the actors, taking part in the business activity.

### 4.2 Actors assignment

To determine the actors taking part in the business activity, a further modelling phase is needed. During this phase, the actors are being assigned to the activities, in which they are taking part. A graphical example of an assignment is shown in Figure 2. The graphical notation is helpful, but not mandatory, since the diagram tends to become cluttered for complex business activities.

In our understanding, actors can be departments of an enterprise, external suppliers, IT systems or any other entity contributing to the achievement of the overall business goal. Because an existing business activity is being described, the actors and their contributions to the achievement of the business goal are known to the owners of the business domain. During this modelling phase, actors executing two orthogonal functions can be identified and spitted. This fosters the exploitation of outsourcing and monitoring opportunities.

### 4.3 Results dependency graph, actor's state transitions and interface determination

The essential part of the business activity analysis, is the determination of actor's result dependency. The assumption during this phase is the dependency of an actor taking part in an activity, upon the results of a previously executed activity. If the actor is not dependent upon the results of a previous activity, there still has to be an initiating condition for actor's actions.

By defining actor's dependencies, a result dependency graph is being induced. Figure 3 shows the result dependency graph from the activity depicted in Figure 1. The controlling actor has been omitted. This graph is directed. The direction of the edges is from the depending actor, to the actor holding the results. The dependencies can be seen as signals, being sent to dependant, upon completion of the activity, the proceeding is contingent on.

From the results dependency graph, an interface of every actor is being derived. The interface exhibits sinks for every message type an actor can receive and sources for every message the actor can emit. Figure 4 depicts the interface of the Production planning / Engineering actor, derived from the results dependency graph, as shown in Figure 3. The message sinks are labelled A through E, for simpler reference. In the actor's interface model, the incoming messages do not posses any order. The order of the incoming system is described in the accompanying internal state transitions graph. Dashed lines, in the state transition graph, show state transitions triggered by actor's internal event. Thus, these transitions were not influenced by any external event. The internal state transitions can not be derived solely from the results dependency graph. For their identification, the knowledge of the business domain owners is being needed. The emitted messages are not represented in the states transition graph.

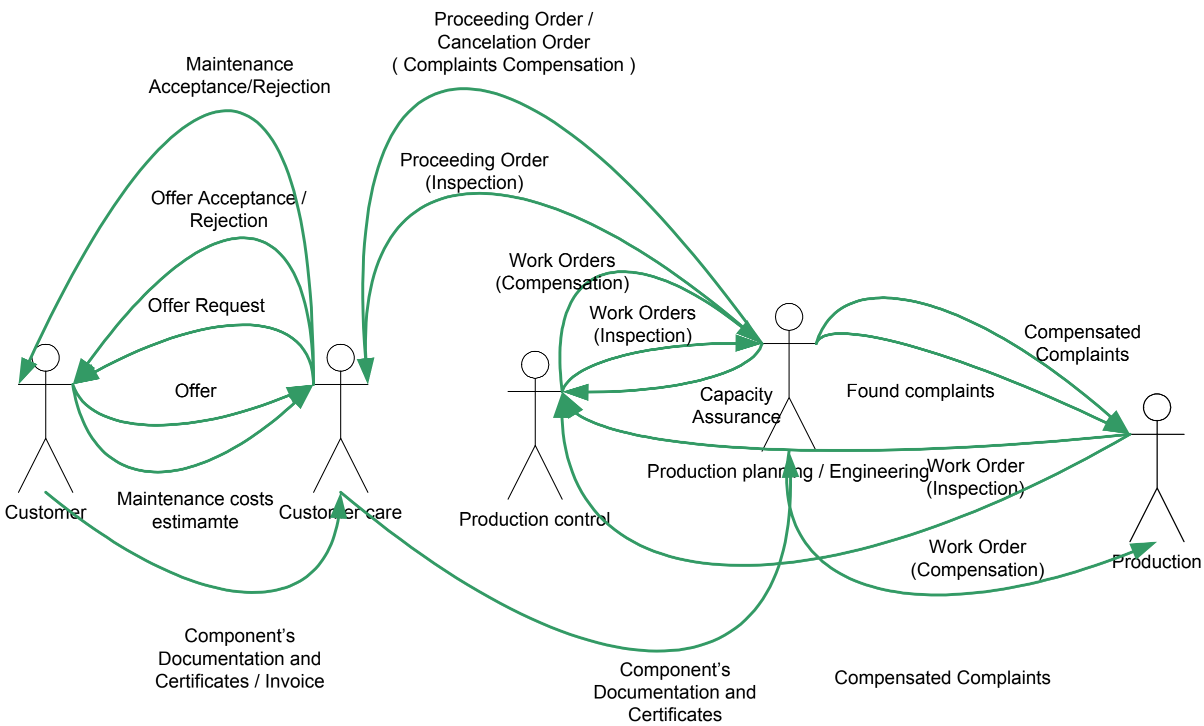


**Figure 3 Results Dependency Graph**

During the creation of the interfaces and internal state transition graphs, those in charge of the concerned actor have the possibility, to review the model of their responsibility area. The benefit of this modelling phase is that the actors can be seen in isolation from the overall system. The model, created by the results dependency graph, is hereby refined. The refinements of the models of single actors can proceed in parallel, contributing the review results[6] to the higher level (result dependency graph) model.

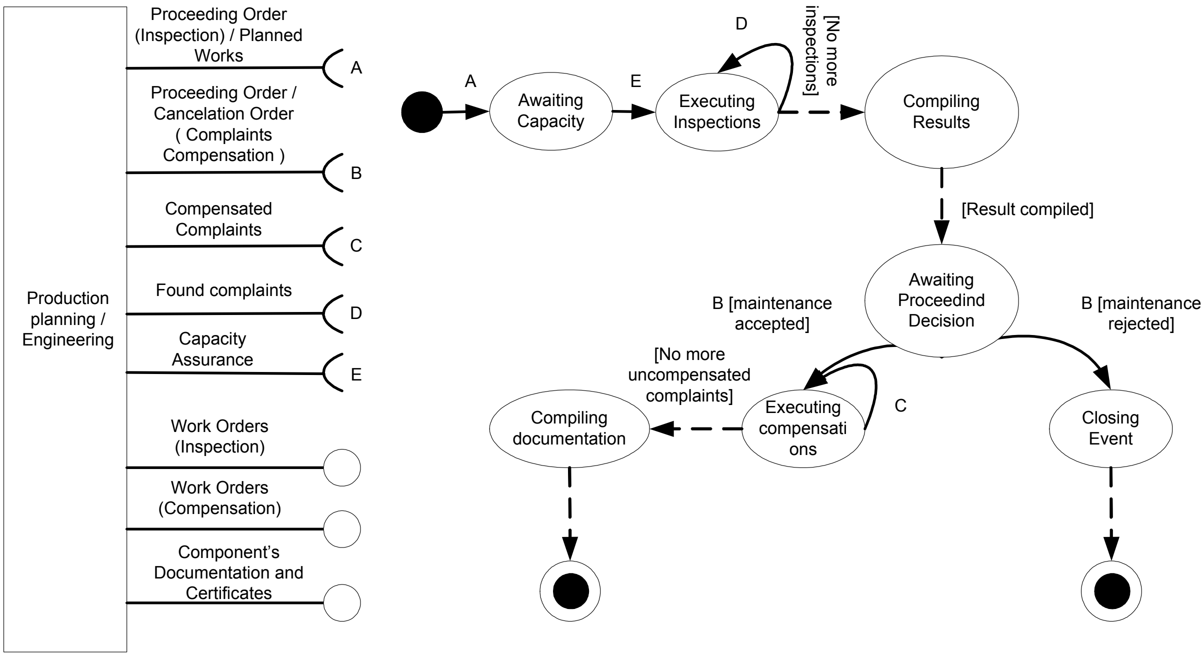

**Figure 4 Production planning/Engineering Actor's Interface and internal state transitions graph**

## 4.4 Safety properties derivation

As described in Section 3, the goal of the presented modelling methodology is the creation of a safety properties catalogue, to enable the achievement of the overall business goal. These properties are to be derived, from the created activity model.

The distinction between business domain safety properties and technical domain safety properties is to be made. The business domain safety properties have the form of properties presented in section 3.2. These are stated in form, understandable to the owners of business domain. However, safety properties stated in this form can not be exploited for formal model validation. Therefore a safety properties translation from the business, towards the technical domain is to be made. Ensuing safety properties are to be stated in terms of actors' internal states and messages exchanged between actors.

An example of the described safety property translation is the translation of the safety property 3.1, into safety property (4.1). A more complete depiction of the safety properties translation process is given in [6].

- Component certified airworthy => State *Compiling Documentation* reached (4.1)

## 4.5 System validation

Assumed, the system would be modelled in a perfect manner, the implementation will be error prone, due to the imperfections of the construction process, or the limitations of the underlying systems. Thus, it is essential to monitor the deployed system, to assure that no violations, of the modelled behaviour have occurred.

The approach shown in [2] proposes a system validation based upon traces left, by the communication between services. The validation is being performed by evaluation of a set of predicates upon the traces. To exercise the validation, the occurred communication has to be recorded and the set of predicates is to be created. For communication recording, filters [8] placed in front of actor's interfaces can be used. The predicates, to be evaluated upon the traces can be directly constructed from the safety properties described during the properties derivations phase. For instance, a predicate can state, that that for every message D from Figure 4 a message C has to be received. This predicate has been constructed from the safety property of the system stating, that for every found complaint, some action has to be taken.

[6] The contributions are for example omissions made in the results dependency graph.

## 5 FUTURE WORK

We identified three directions for the further development of the proposed approach. First challenge is development of methods, for the matching of business domain safety properties onto protocols safety properties. Further work can be done in the analysis of the existing generic protocols, towards the identification of the safety properties they provide. Another development direction is the tool support for the proposed methodology. We are considering the development of a UML 2.0 Profile[10] , to facilitate usage of the methodology, by re-using an existing notation and tools. Further improvement can be achieved by the creation of a set of OCL constraints in the profile, to allow for automatic checking of rudimentary model properties inside the used UML tool.

## 6 CONCLUSION

The presented modelling methodology provides a tiered view of the specified system. It makes an appropriate abstraction level available, to the modellers concerned with the particular view of the system. Through refinement of the successive abstraction levels, a more precise system description is being provided. Since every modelling tier calls in business domain owners concerned with the tier's abstraction level, valuable feedback about the quality of the previous abstraction levels can be provided. This results in model's quality increase during review iterations.

The artefact of the modelling by the means of our methodology is a catalogue of safety properties. These are to be fulfilled, if the system is to achieve its business goal. The safety properties have to be fulfilled regardless of the distribution of the business activity execution. Therefore, the resulting safety properties constitute the essence of the system. We want to be able to determine applicable generic protocols, by mapping business domain safety properties onto safety properties guaranteed by the considered protocols. This will enable the safe usage of generic protocols as system building blocks.

The proposed modelling methodology in conjunction with ongoing work on formal methods application enables the usage of generic protocols as building blocks, for virtual enterprises and their supporting IT systems.

## REFERENCES


[1] Johnson, J.E., Langworthy, D.E., Lamport, L., & Vogt, F.H. (2005). *Formal Specification of a Web Services Protocol*. Submitted for a special issue of Journal of Logic and Algebraic Programming.

[2] Venzke, M. (2004). *Specifications using XQuery Expressions on Traces.* Retrieved March 1, 2005, from Hamburg University of Technology, Telematics Institute

[3] Friedrich H. Vogt, Simon Zambrovski, Boris Gruschko, Peter Furniss, Alastair Green: *Implementing Web Service Protocols in SOA: WS-Coordination and WS-Business Activity*, IEEE International Workshop on Service oriented Solutions for Cooperative Organizations (SoS4CO '05), Munich, Germany

[4] Jurga Kazlauskaite, Arsalan Minhas, Friedrich H. Vogt: *Applying Service Oriented Architecture in the Aerospace Industry.* Proceedings of the IMCM'05: International Mass Customization Meeting 2005, Klagenfurt/Austria, June 2-3, 2005.

[5] Simon Zambrovski: *Protocol Engineering for the Management Process in Product Lifecycle.* Master Thesis. Hamburg University of Technology, June 2005.

[6] Boris Gruschko: *Service Oriented Modeling of a Maintenance Process in the Aviation Industry.* Master Thesis. Hamburg University of Technology, October 2005.

[7] Luis Felipe Cabrera and George Copeland and William Cox et al.: *Web Services Atomic Transactions (WS Atomic Transactions).* January 2004.

[8] Deepak Alur, John Crupi and Dan Malks: *Core J2EE Pattern. Prentice Hall PTR. June 2003.*

[9] Leslie Lamport: *Specifying Systems. Addison-Wesley Professional. August 2002.*

[10] Object Management Group, 2005. *UML 2.0*. http://www.uml.org/