15th ICCRTS

"The Evolution of C2"

Business Values of Automating Unstructured Decision Processes with Dynamic Enterprise Mashups

Hans Peukert, Volker Hoyer, Andrea Fuchsloch, David Lincourt and Marek Kowaliewicz

Point of Contact

Hans Peukert

Chief Strategist

Global Defense & Aerospace Sector

SAP AG

Dietmar-Hopp-Allee 16

69190 Walldorf

Germany

Telephone +49 6227 7 41296

E-mail Address: hans.peukert@sap.com

Abstract: *The typical information worker in military organizations faces on a daily basis the need to make decisions. These processes are unstructured and most often non-repeatable meaning it is not possible to create relevant IT applications in advance by traditional software development approaches such as Service-Oriented Architectures. The information worker has to react on alerts and events in ad-hoc fashion following Boyd's OODA loop. A new development paradigm, known as enterprise mashups, has gained momentum in recent years. It enables information workers to create situational applications on their own. However, current discussions both in the scientific and industrial community are limited on technical aspects such as the development of relevant mashup platforms which allow composing dynamically applications to make a decision. We extend the previous research in facilitating the building and collaboration through social software. Thus, the question arises what are the actual business values of this new paradigm for military organizations? This present paper closes this gap by analyzing a real-world defense scenario which we have implemented by means of the mashup prototype SAP Research Rooftop Marketplace. A case study reports about the challenges of typical decision processes in military organizations and how dynamic enterprise mashups can improve the decision quality as well as the decision time. For each phase (observe, orient, decide and act) of the OODA loop, we analyze and quantify the business values.*

Keywords: Enterprise Mashup, Decision Processes, Business Values, Situational Applications, Common Operating Picture, OODA Loop

1 Motivation and Introduction

Since the beginning of the 1990s, companies have optimized their corporate Information Technology (IT) by introducing business transaction systems. In the first wave, enterprises introduced the business process idea to overcome the functional-oriented organization structure (i.e. by introducing an Enterprise Resource Planning system). The technological enabler is the Service-Oriented Architecture (SOA). Modular components defined by well-defined and standardized interfaces are loosely coupled and allow flexible adaptation of the business transaction systems by IT experts. The next wave in corporate technology adoption promises further gains although the capabilities differ from the first automation wave. It will exploit new productivity potential by means of a broad collaboration and a high degree of participation. In contrast to transactional systems, where most users process information in the form of reports or execute business transactions (i.e. entering replenishment orders), new technologies and tools from the Web 2.0 philosophy are interactive. They integrate users to generate new information or edit the work of others. Renowned management scholars such as Andrew McAfee (McAfee 2006) and Don Tapscott (Tapscott & Williams 2006) envision and Enterprise 2.0. It leverages these new consumer-driven technologies in corporate environments in order to put people in the centre of the information-centric work.

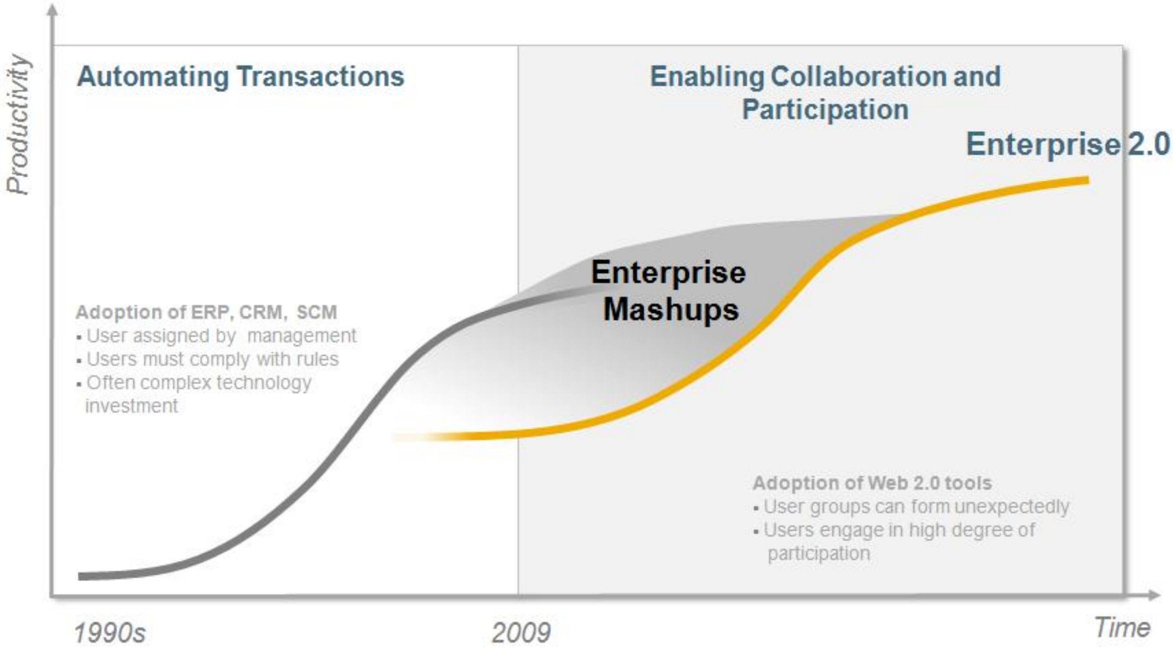

Figure 1. From Automating Transaction to Peer Production

At the interaction between the two corporate technology adoption waves, a new trend for a software development paradigm, known as enterprise mashups, has gained momentum in recent years. It combines the characteristics of both technology adoption waves. At the core are two aspects: First, empowerment of the end users to cover ad-hoc and long tail needs by reusing and combining existing modular software artifacts from an organization's internal IT system as well as external resources. Second, a broad involvement of decision makers based on the peer production concept. Thereby, the creative energy of a large number of people is used to react flexible in continuous and dynamic changes of the business environment. Instead of long-winded software development processes, existing and new enterprise-class application components are enhanced with interfaces (so called Application Programming Interfaces, APIs) and are provided as user friendly building blocks which can be combined individually to solve ad-hoc business problems (Hoyer & Stanoevska-Slabeva 2009b).

Various market research institutes such as McKinsey, Gartner and The Economist Intelligence Unit highlight that the transfer of the mashup paradigm in corporate environments need additional capabilities beyond those typically associated with consumer mashup offerings. In particular, concrete mashup scenarios and demonstration of their benefits and limitations are missing so far (Gootzit & Bradely 2009).

The goal of this present article is to fill this gap by reporting about the implementation of a typical defense scenario which leverages the capabilities of the enterprise mashup paradigm. The scenario deals with the automation of unstructured business processes relevant for the information worker which cannot be modeled in advance. Thereby, the information worker is supported by providing automatically relevant information from company internal and external sources based on the current context.

For answering the research question motivated in the previous paragraph, engaged research is needed in order to provide rigorous solutions. We apply the case study research method as proposed by Yin (2003). In general, there exist four types of designs for case studies: single-case (holistic) designs, single-case (embedded) designs, multiple-case (holistic) and multiple-case (embedded) designs. Thereby, two rationales serve as major reason for conduction a single-case study: First, the scenario represents a critical case in evaluating a well-formulated artifact. And second, the case study represents a typical business situation which appears in other defense processes as well. It can even help to refocus future investigations in the enterprise mashup field. The single-case study may be also used as a pilot case that is the first of a multiple-case study.

The figure below depicts the applied research design. The data collection is based on semi-structured interviews with employees from in the defense industry, literature analysis (Webster & Watson 2002) and the experiences from the scenario implementation. Our research is situated in the field for Information Systems. It is ultimately an applied research discipline to solve practical problems at the interaction of organization, people, and information technology (Lee 2009). According to this definition, the case study is structured by means of two analysis units: First, from an information technology perspective and second, from an organizational (business process) perspective.

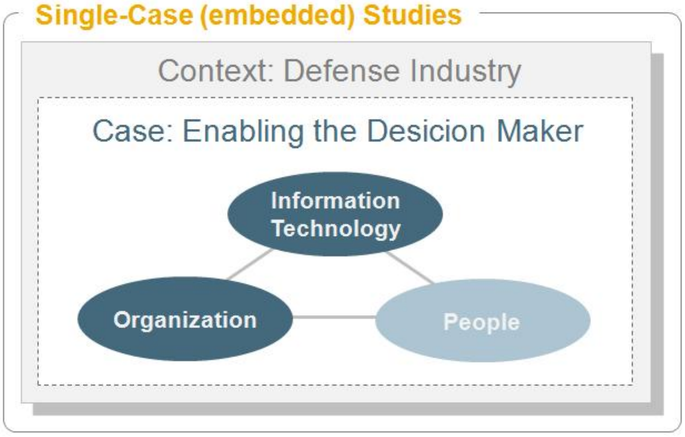

Figure 2. Research Design

The remainder of this article is organized as follows: Second section elaborates on the underlying philosophy of enterprise mashups and presents the SAP Research RoofTop Marketplace prototype which is leveraged for implementing a typical decision process in military organizations. In the third section, we present the actual case study. Apart from the industry background and the decision process scenario, we discuss the current challenges and the opportunities. By means of the new solution leveraging the concept of dynamic enterprise mashups, we identify and quantify the business value of the emerging concept. Finally, the last section closes the article with a brief summary, limitations of the conducted research and an outlook to further research.

2 Related Work and Background

2.1 Enterprise Mashups

The definition of enterprise mashups is open to debate. In this paper, we refer to the following definition: *"An enterprise mashup is a Web-based resource that combines existing resources, be it content, data or application functionality, from more than one resource by empowering decision makers to create individual information centric and situational applications"* (Hoyer et al. 2008). By simplifying concepts of SOA and by enhancing them with the Web 2.0 philosophy of peer production, enterprise mashups generally focus on software integration on the user interface level instead of traditional application or data integration approaches. In contrast to SOA that is characterized by high technical complexity of the relevant standards and requiring specialists' technical knowledge, mashups enable the integration of decision makers with no or limited programming skills in the development process.

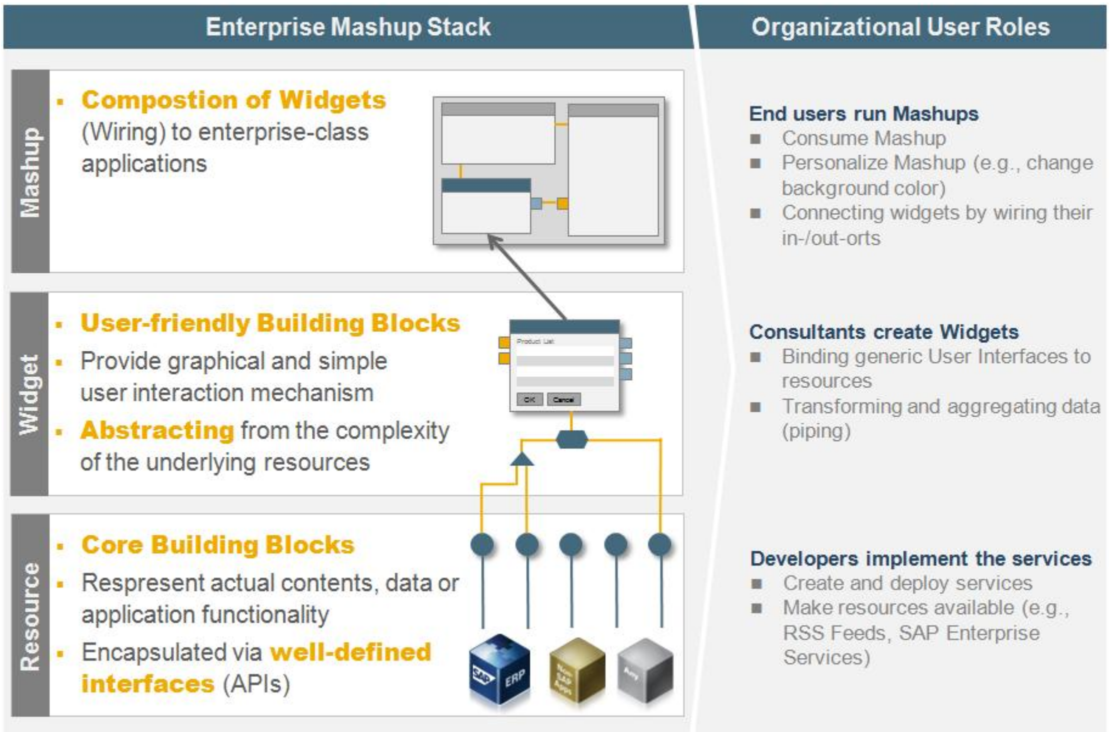

Figure 3. Enterprise Mashup Terminology and User Roles

The relevant architectural components of the enterprise mashup paradigm are resources, widgets, and mashups and can be structured in an enterprise mashup stack comprising three layers (see Figure 3): **Resources** represent actual contents, data or application functionality that are the core building blocks of mashups. They are encapsulated via well-defined public interfaces (Application Programming Interfaces; i.e., WSDL, RSS, Atom, etc.) allowing the loosely coupling of existing resources – a major quality stemming from the SOA paradigm (Alonso et al. 2004). These resources are provided by enterprise systems or by external Web providers (i.e., Amazon, Google, etc.) and are created by traditional developers who are familiar with the technical development concepts. On the second layer, **widgets** provide simple user interaction mechanism abstracting from the complexity of the underlying resources. For example a widget "Customer Data" might provide results for a predefined query requesting the data for all customers of a sales manager. The creation of these widgets can be done by consultants in the business units who understand the business requirements and know basic development concepts. Finally, decision makers with no programming skills are able to combine and configure visual widgets according to their individual needs, which results in a **mashup**. For example, the analyst wires the "Unit Data" with a map to show the location of the unit.

The shift from delivering finished applications created by the IT department towards delivering of user friendly building blocks that can be combined individually implicates a changing development model. This involves both managing the mashup components and managing the relationships between the involved people.

In contrast to SOAs, enterprise mashups usually aren't constructed by a team of traditional software developers from the IT department. Instead, they are created by decision makers themselves from the business units characterized by no or limited programming skills. In that regard, enterprise mashups particularly serve as a means to address newly emerging requirements in the implementation process of service-oriented information systems. The rationale behind this argumentation is that making use of the SOA-potentials, the specific business needs of a company has to be taken into account precisely. Here, the concept of enterprise mashups comes in, providing a promising means to involve users from the business engaging into the system design.

These decision makers, however, desire specific functionality that mainstream SOA-based enterprise applications don't provide. In this sense, the enterprise mashup paradigm aims at creating ad-hoc or "good enough" solutions which address daily and tactical needs. According to the continuous changed business environment, they are often adapted and don't follow the traditional development phases. Instead, they evolve organically in a decentralized manner. Non-functional requirements like scalability, maintainability or availability mostly play a minor role. If created applications don't fulfill the decision maker requirements from a non-functional perspective, they are replaced immediately by the community. In other words, the users define the threshold for the applied quality of services and non-functional requirements. In a further step, governance policies can be introduced to guarantee a minimum of quality. In this kind of grassroots computing (Cherbakov et al. 2007) individuals share their customization efforts with a like-minded community. The focus on delivering a set of user friendly building blocks rather than finished enterprise applications enables to automate also tactical and opportunistic applications.

Instead of following the traditional software development phases (requirements, specification, development, testing and deployment), community-driven enterprise mashup environments can be organized in a similar way to electronic markets. Besides the support for easy integration of software artifacts, they also require support for efficient management and matching of supply and demand for mashable components (Hoyer and Stanoevska-Slabeva 2009a).

2.2 SAP Research Rooftop Marketplace.

In the recent years, several software tools for creating mashups have been developed (Hoyer & Fischer 2008). However, the discussion from a collaborative and peer production perspective is still missing in the scientific community and in the relevant mashup tools. In the frame of the SAP Research RoofTop Marketplace, we have developed a prototype which follows typical market transactions to handle the organization challenges (Hoyer et al. 2009).

Starting with the **knowledge phase**, the information workers (registration is done by Single Sign On) are able to find information about the offered mashable components (widgets and mashups) and about the person in form of profiles. Specified individual preferences and user contexts (i.e., industry, department, country, etc.) allow the navigation within the enterprise mashup medium. By means of examples in form of short videos, the benefits of the enterprise mashup environment are demonstrated to potential users. Only if a huge amount of decision makers are convinced of using the environment, it will exploit its actual potential. During the **intentions phase**, decision makers articulate their intention and needs such as wish lists or favorites. Concepts from the Web 2.0 philosophy, like rating, tagging, or recommending are integrated for browsing through the growing number of offers (published and available mashable components). In case the consumer accepts the underlying business model (costs, payment model, consumption license, etc.) of a widget that is defined by the provider, he can compose the component with others by connecting the input and output parameters of the widgets in the actual **contract (design) phase**. By certifying widgets or providers, indicating compatibility, trust or reputation aspects, the intermediary offers additional selection criteria to the consumers. In contrast to the classical software development, the design of ad hoc applications uses real resources and no demo systems. In this sense the consumption in the **settlement phase** differs only from the hidden configuration capability in contrast to the design phase. In case a new business situation comes up, the consumer shifts quickly to the design or intention perspective to adapt the individual operational environment. The traditional separated design and run time is converging and characterized by continuous return loops.

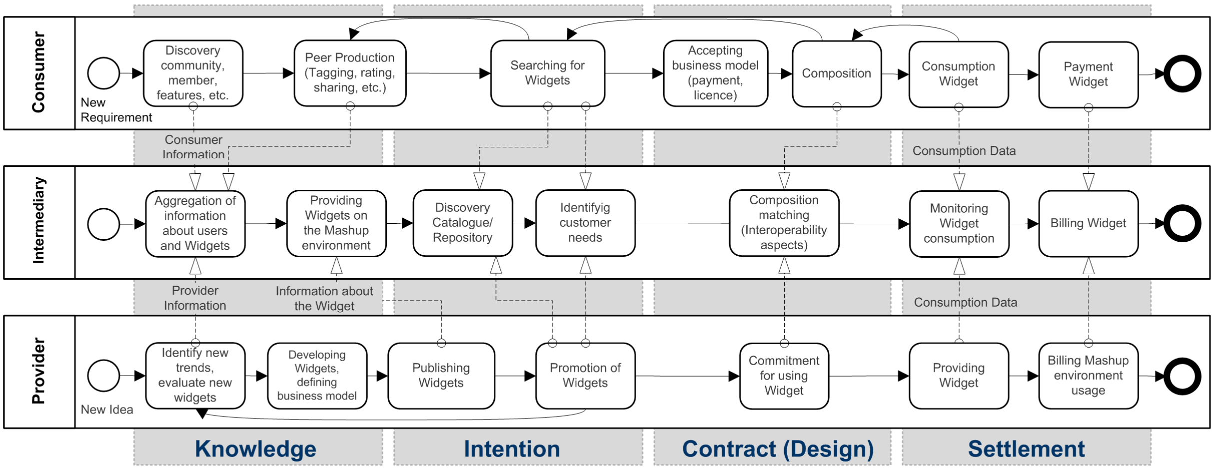

Figure 4. Organizational Phases of Enterprise Mashups

User experiences with a first version of the SAP Research RoofTop Marketplace prototype (Lincourt, Peukert, & Kowalkiewicz, 2008) have demonstrated that the composition of visual widgets into mashups is often too complex. The typical information worker is not familiar with any kind of composition technique. A successful exploitation of the mashup paradigm requires a critical mass of users who are willing to play an active role in the community. They are the innovative drivers and create the content of grassroots enterprise mashup environments. In general, the creation of new content by composition leads to two main challenges:

- *Discovery.* In the first step, the consumer is browsing the catalogue for finding the right component which fulfils his current requirement or task. Based on the enterprise mashup ontology presented in the previous section, the consumer can search by keywords, can browse the offers in the categories, can open a tag cloud or is looking in recently published widgets. However, there are billions of mashable components and it's a challenge to find a suitable component.
- *Composition (Interoperability).* The second step deals with the composition of two or more selected components by connecting their output and input ports. The user must know which output and input fits together. Otherwise, the components cannot interact with each other.

This composition challenge is addressed by a concept called **dynamic and context-aware enterprise mashups** which was implemented in the SAP Research RoofTop Marketplace prototype (Gilles et al. 2009). The concept of dynamic and context-aware is the key to providing the decision maker with capability to adapt to emerging situations. Adaptation in this context is the ability to alter force organization and work processes necessary as the situation and/or environment changes. It also involves the alteration of the way information is distributed and consumed (Albert and Hayse, 2003, p153). As a positive consequence, organizations can more easily innovate how they provide solutions to the challenges they face.

Our enterprise mashup environment automatically recommends relevant information-sensitive services based on existing compositions created by users in the community who are acting in a similar social context. After the user selects a component, it is automatically added to the individual workspace and connected. Neither have the users to switch between different applications nor have they to search for a business transaction. No installation and no programming skills are required.

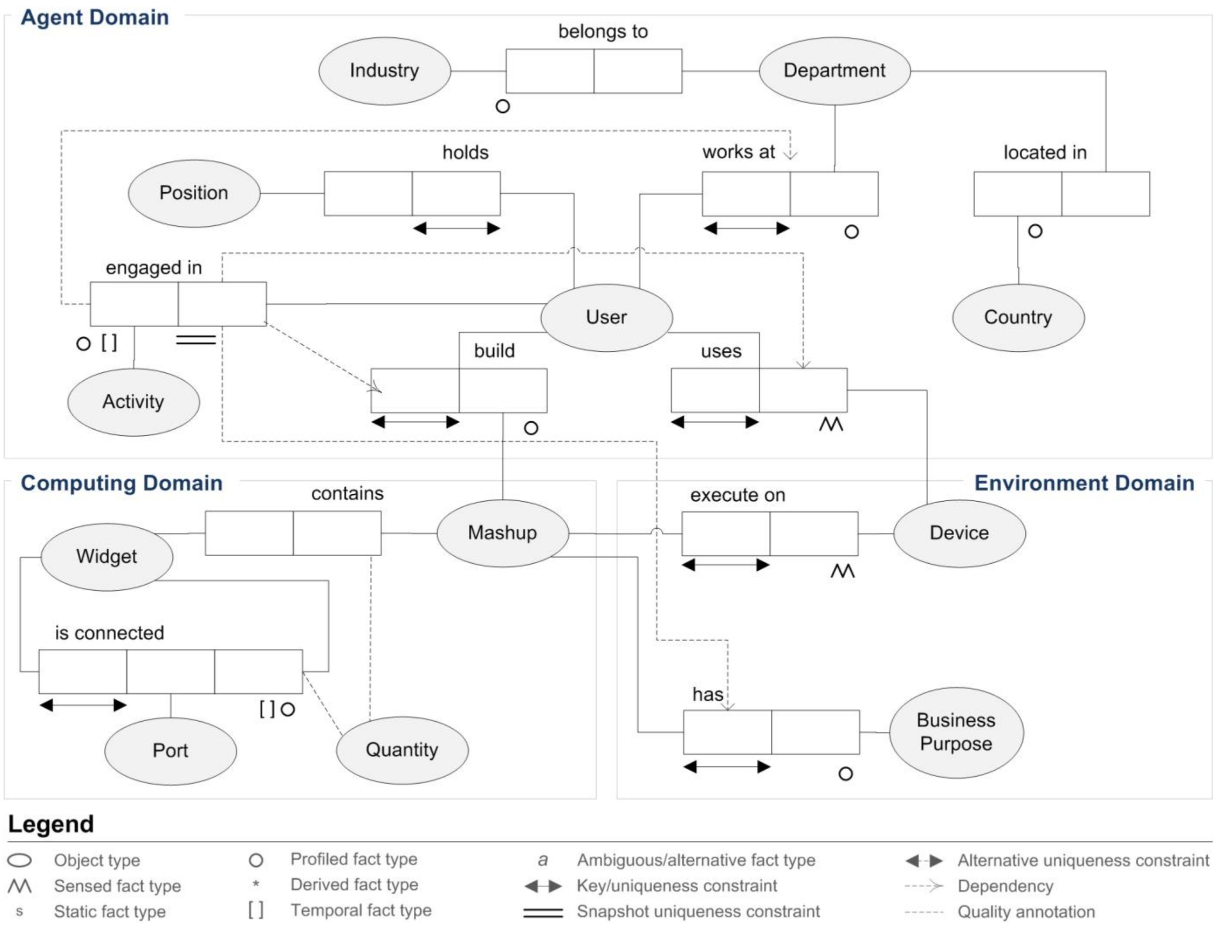

Figure 5. Dynamic Enterprise Mashups

The underlying context model specifying the space of dynamic enterprise mashups is depicted in the figure above:

- **Agent Domain (Who?).** The first domain deals with the interacting agents (users) within the community and their relationships. The agent domain represents all human users who make use of an enterprise mashup environment in their daily work. These agents act in a particular context which is determinate on their position (i.e. sales manager) in the company. It includes information such as the department (specialized division of the business organization where the user is working for, i.e. sales department), country (where the department is located at, i.e. Switzerland), or the industry (i.e. manufacturing). Besides this working related context information of the agent domain, the social structure and agent preference are an addition context sources. In a similar way to social networks such as Facebook and Linkedin, users working in the same department consume similar mashups to solve their daily problems.
- **Computing Domain (What?).** The computing domain handles the actual content of the mashup platform which is based on existing composition of widgets into mashups.

It mainly includes the widgets and the wiring connections of their input and output ports. This domain builds the primary foundation for context-aware enterprise mashups. The popularity, the availability or the user rating of a mashable component directly influence the context-aware recommendation.

- **Environment Domain (Where?).** The third and last domain addresses environmental aspects. A mashable component can be optimized for a particular environment spanning the device or the business purpose. A mobile device implicates other requirements than a portal application. First, the limitation on the resolution or the reduced bandwidth must be taken into consideration. Second, an enterprise mashup aims at a dedicated business purpose meaning an enterprise mashup is created for a special enterprise resource planning feature.

3 Case Study

After elaborating on the technical issues of how to integrate the social software idea into enterprise mashups, this section is devoted to present a case study about a typical defense scenario.

3.1 Background and Scenario

In the defense industry, military organizations are facing with a high complexity. In a multinational and dynamic environment, operational requirements on military forces continuously change. Since it is impractical or even impossible to prepare for all these contingencies, military organizations must develop the ability to deal rapidly with all kind of situations. Their key contributor is timely, accurate and relevant information, which enables high responsiveness and flexible decision making. Military organizations have created numerous stand-alone information systems that by definition hinder the sharing of information. Thereby, information transparency plays a major role in the 21st century (Kasper, 2001, p. 33 f.).

Five to ten percentages of all defense employees are in the position of decision making, called information workers. On the contrary, transaction workers do work based on guidelines and rules with little room for decision making. Information workers are challenged in a broad range of complex process areas, such as logistics management, planning and support of deployed forces, financial management, acquisition and material management, deployment as well as personnel management. Up to 80% of these processes are organized in a structured way and can be implemented by traditional SOA-based solutions. However, decision making is hindered in 20%, when processes occur in unexpected and tactical situations.

To succeed, information workers require the ability to create ad-hoc applications. Enterprise mashups help to assemble a Common Operating Picture (COP) or the more recent moniker User Defined Operating Picture (UDOP) enabling profound decisions (Lincourt, Peukert, & Kowalkiewicz, 2008). Thereby, relevant data, mainly generated in the back office, need to be operationalized and provided to the information worker in the front office (“Power to the edge”) in real-time. The back office contains administrative functions that support business operations at the front but are not directly involved in business operations.

The scenario of the case study focuses on the tactical level of unstructured decision processes. Unstructured refers to decision processes that don’t run in quite the same form and which cannot be modeled in advance. Decision is a specific commitment to a particular action which usually involves the commitment of resources. The scenario represents a typical decision process in the management area and focuses on the resource management of personnel and material. In general, military decision-making processes are structured along four phases of the so-called OODA Loop (Observe, Orient, Decide, Act) which was defined by US Colonel John Boyd. It provides a model of the myriad of interactions and feedback mechanisms which occurs in decision-making processes (Richards, 2004).

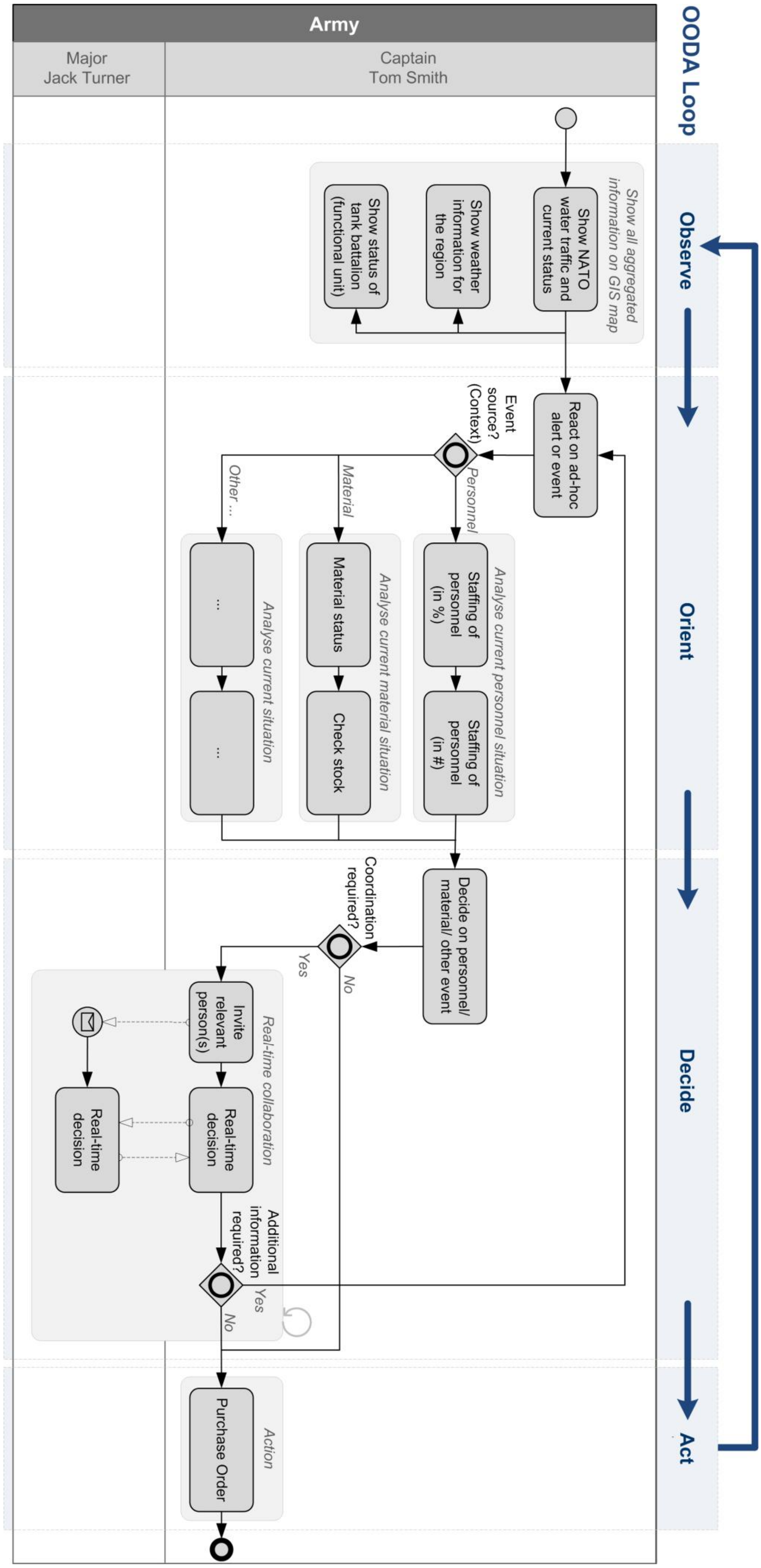

Figure 6. Business Process of the Scenario along the OODA Loop

The structuring along the OODA loop allows transferring the experiences and the results of the case studies to other defense processes, as all of them follow this decision making cycle. Although the OODA model was clearly created for military purposes, elements of the same theory can also be applied to business strategy in other industries. *“A time-compressed company does the same thing as a pilot in an OODA loop […] It’s the competitor who acts on information faster who is in the best position to win.”* (Stalk & Hout, 2003, p. 180).

The underlying process of the scenario is depicted in Figure 6. By applying the notation of Business Process Modeling Notation (BPMN), the two swimlanes show the two user roles involved, the activities they perform and the information flow between the single process steps.

- **Observe Phase.** The first user role, Captain Tom Smith, is a staff officer in the logistics operations centre of an armored division. It is his job to ensure that the functional units of the divisions are always ready to use. In the first phase, Tom observes the current status of the environment. Typical for the defense industry, a map serves as a central representation of the information which is sourced in real-time from different internal and external sources. Besides data from an ERP backend system handling the status of the functional units, his job needs the aggregation of the current water traffic and weather information for a dedicated region.
- **Orient Phase.** In the orient phase, Tom makes sense of the observation by analyzing the information received. In case an unexpected situation appears, he as to react to this ad-hoc event. Therefore, the he has to identify the source of the alert in order to prepare a decision. In the scenario, this could be an understaffing of a functional unit or an insufficient material status. A deeper analysis helps to understand and locate the problem, e.g. by analyzing the current personnel situation in more detail or by checking the material stock.
- **Decide Phase.** In the third phase, a decision is made based on the previous analysis. The phase deals with the development and evaluation of alternatives. If coordination is required, Tom starts a real-time collaboration and invites other persons who are involved in this decision. In the case of missing material, Tom’s boss, Major Jack Tuner has to be involved in the decision. Without time delay and initializing a whole workflow, a solution can be discussed for the material issue.
- **Act Phase.** In the last phase, the decided action is conducted. This could be relocation of personnel or, in case of the missing material, the creation of a purchase order for receiving the missing material.

By today, information workers in military organizations use various IT systems for performing the presented scenario such as maps, customer data base or external information systems. Relevant information is not aggregated available by one central entry point, rather the information worker has to switch between the systems and copy-paste data. This working process is highly inefficient and the error rate is high, as information is transferred manually to different systems.

3.2 Challenges and Opportunities

The typical decision scenario described in the previous section already indicates several challenges and opportunities. This section generalizes and enhances them by means of additional documents and conducted interviews:

- **Speeding up the OODA loop.** By speeding up the OODA loop, a military organization can get inside the enemy’s decision-making process which represents a competitive advantage. The actual end user (for example staff planner, intelligence analysts, battle captains, etc.) needs to be empowered to create their own workspace instead of waiting in the traditional software development queue for further additions or new system integration. Success favors the side that can react sooner than the enemy (Heier, 2008, p. 10).

- **Understanding the current situation.** Besides speeding up the whole decision process, the understanding of the current situation is even more important. In particular, the observe and orient phases become the critical element of the OODA loop (Hammes, 2006, p. 216). By obtaining information and data from sources that are by today unavailable or difficult to gather, the information worker is able to better assess the actualities of a situation. Thereby, the seamless integration of non-military related sources improve the understanding.
- **Reducing uncertainty in decision processes.** All decisions are made under uncertainty (Daft & Lengel, 1986). However, uncertainty can be reduced by constructing a decision on more and real-time information. The obtaining of non-defense sources in the decision processes improves the decision quality.

In summary, there exist a huge potential of enterprise mashups to support the information worker. Ideally, enterprise mashups enable the automation of unstructured decision processes in order to operate inside a competitor's OODA loop as mentioned by Boyd. The goal is to change the situation ("acting on the information") before the competitor can understand what's going on (Richards, 2004).

3.3 New Solution with dynamic Enterprise Mashups

After elaborating on the current challenges and opportunities of unstructured decision processes in military organizations, this section presents the new solution by leveraging the SAP Research RoofTop Marketplace prototype. A screenshot of the resulting enterprise mashup is shown in Figure 7. A produced video demonstrates the dynamic and ad-hoc creation of the application[1].

The SAP RoofTop Marketplace prototype serves as a central entry point for all decision phases. No switching between various IT systems is required. In the first phase, the aggregated information is visible on a Geographic Information Systems (GIS) map. The JavaScript based API[2] allows easily embedding a map. As depicted in **Error! Reference source not found.**, it shows the current water traffic in this region he is responsible for (in this case the port of Rotterdam, Netherlands). The public available data is sourced from a NATO system called Networked Interoperable Real-Time Information Services (NIRIS)[3]. The widget located in the top right (*NATO Track Stores by NIRIS*) lists the current position of the ships. The technical foundation of the resource is a SOAP/ WSDL-based Web service[4]. The wiring composition connects the two widgets with each other. Now, Tom can select a specific ship he is interested in and its position and further information is displayed on the map. Additionally, Tom continuously observes the current status of the functional units of his army from a SAP ERP backend system. The widget calls a SOAP/ WSDL-based SAP enterprise service which is called "Manage Functional Unit"[5].

---

[1] http://www.youtube.com/watch?v=Rtc_ulwimYA, last checked 2010-01-18

[2] http://resources.esri.com/arcgisserver/apis/javascript/arcgis, last checked 2010-01-19

[3] http://www.npc.nato.int/htm/niris.htm, last checked 2010-01-19

[4] http://niris.nc3a.nato.int:8080/tixo/tide1/TixoTide1?wsdl, last checked 2010-01-19

[5] https://wiki.sdn.sap.com/wiki/display/ESpackages/Read+Functional+Unit, last checked 2010-01-19

Figure 7. Enterprise Mashup – Automating Unstructured Decision Processes

In the recent days, Tom has recognized that he requires weather information for his observations as well. Thanks to the catalogue of the SAP Research RoofTop Marketplace prototype, Tom is able to search for a relevant weather widget. Because one of the weather widgets is rated as very good by Tom's colleague who also use the enterprise mashup environment, Tom is evaluating the annotations of the widget (such as the quality and user rating). The widget based on a REST-based[6] resource from the Norwegian Meteorological Institute and the Norwegian Broadcasting Corporation[7] fulfils his requirement and he is adding it. Then, located in the bottom left corner of the mashup, the widget shows a six-day weather forecast for the region the ship is located.

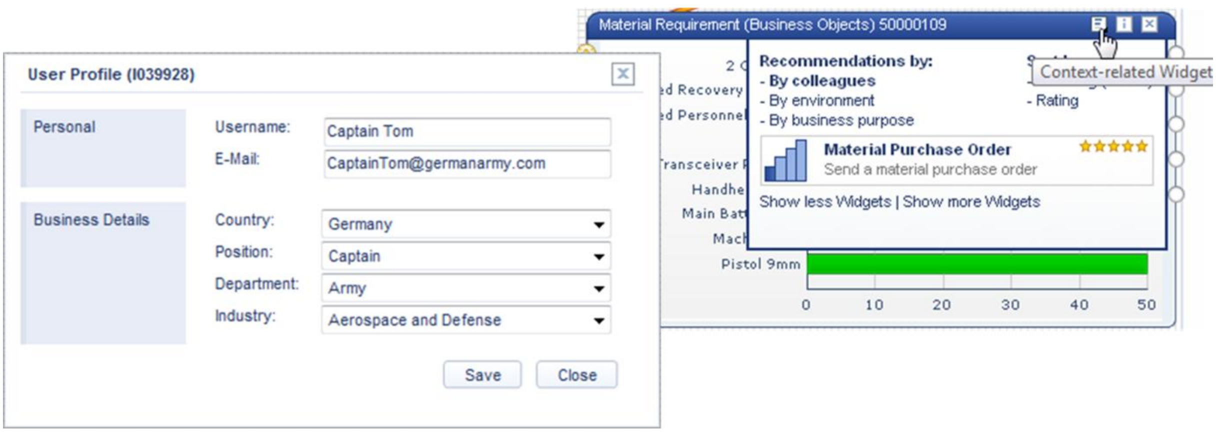

Figure 8. Context-related Orientation

---

6 http://api.yr.no/weatherapi/documentation, last checked 2010-01-19

7 http://www.yr.no/english, last checked 2010-01-19

In the orient phase, the material status of a functional unit (111th tank battalion) is switching to red. This indicates a certain anomaly to which Tom has to respond. Unfortunately, Tom doesn't know how to react. Now, the context approach comes into place. Based on Tom's individual context, it automatically suggests adding a new widget called material requirements as depicted in **Error! Reference source not found.**. The widget contains a small Adobe Flash application created with SAP Xcelcius. It obtains the material stock for a given functional unit through the SAP enterprise service "Functional Unit ERP Material Requirement". The material analysis shows that the material transceiver portable is out of stock. That was the reason for the material status being red.

In the next phase, Tom is starting a real-time collaborative decision with his boss, Major Jack Tuner. For that, the new SAP real-time collaborative decision platform 12Sprints[8] is incorporated. We also replicated this scenario using Google Wave[9] instead of 12Sprints and obtained similar results. The REST-based API[10] allows creating new decision activities directly in the SAP Research RoofTop Marketplace prototype. Figure 9 depicts the view for the two involved users. After adding the relevant 12Sprints widgets, Tom is able to create a new decision activity about the ordering of the missing material. Now, Jack has to decide about the request. He is responsible for the budget and he has to approve the request. In the 12Sprints inbox, Jack finds a new activity (*Decision@RoofTop*) and decision item (see left side in Figure 9). Before making the decision, he asks Tom if he already contacted the 112th tank battalion in order to help out. Without changing the application, Tom answers that the transceiver portable material is also out of stock. Then, Jack approves Tom's request. Tom can see the result of Jack's decision in his enterprise mashup.

In the final act phase, Tom is adding a purchase order widget which is suggested by the context approach. After checking the automated generated details such as quantity and item, Tom creates the purchase order. The widget accesses the SAP backend system through the SAP enterprise service "Manage Purchase Order".

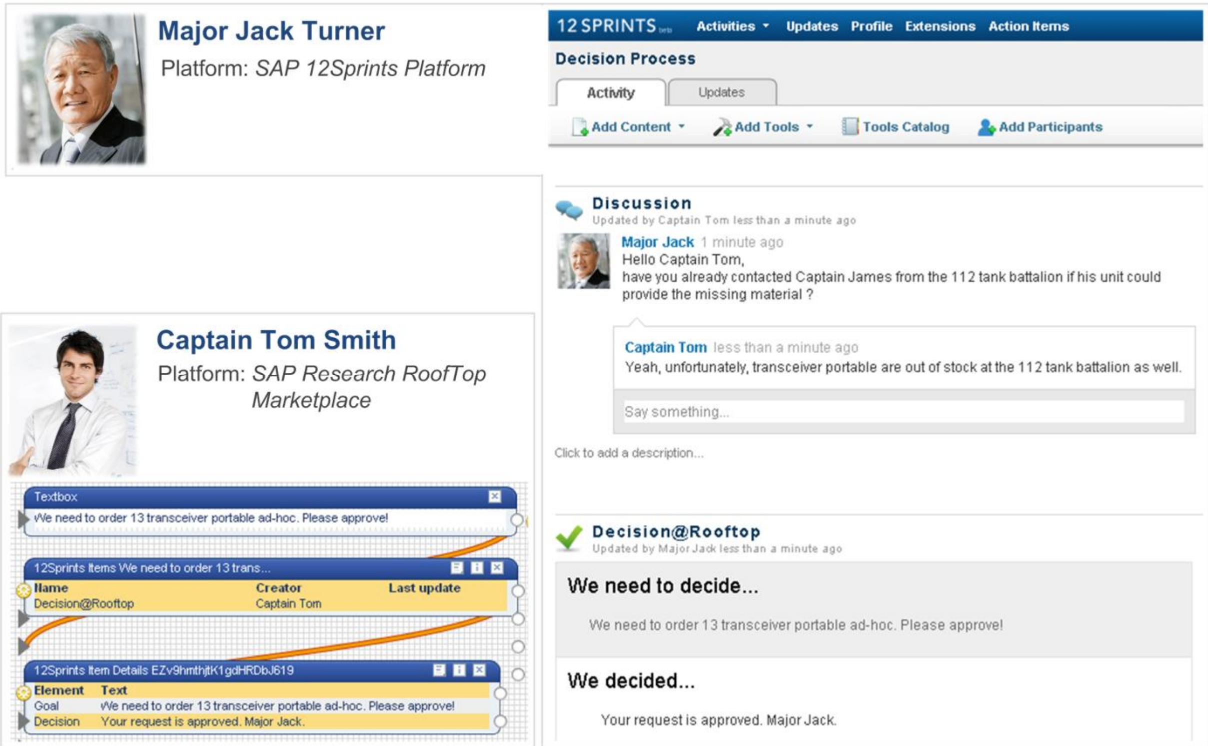

Figure 9. Collaborative and Real-Time Decision

---

[8] http://www.12Sprints.com, last checked 2010-01-19

[9] http://wave.google.com, last checked 2010-01-19

[10] https://beta.12sprints.com/documentation, last checked 2010-01-19

3.4 Business Values

In the presented scenario, Tom manages to observe his environment, to recognize an anomaly and to react immediately to the ad-hoc event. But how do enterprise mashups improve organizational excellence and performance? This important question is relevant for corporations in order to invest in enterprise mashups. By the end of the day they will not introduce the paradigm without pointing out clear business values.

The main benefit in the scenario is the automation of unstructured decision processes. Thus, the following business value investigation focuses on the process level. In order to develop an understanding of adequate value metrics, the process-oriented framework for assessing the business value of IT developed by (Mooney, Gurbaxani, & Kraemer, 1996) is applied. According to the framework, IT and therewith enterprise mashups can have three separate but complementary effects on business processes: automational, informational and transformational effects. The implemented scenario serves as foundation to quantify the business values. The four phases of the applied OODA loop structures the investigation. Table 1 summarizes the findings for the effect types.

Effect Type	Observe	Orient	Decide	Act
Automational				
Processing time	6 → 1 min (- 83%)	12 → 3 min (- 75%)	6 → 4 min (- 33%)	2 → 1:20 min (- 38%)
Collaboration and sharing	++	+++	+	+
Informational				
Process breaks	5 → 0	3 → 0	3 → 0	1 → 0
Decision quality	++	+++	++	o
Transformational				
Competiveness	++	+++	+	+

Table 1. Business Values

First, **automation effects** refer to the operational efficiency. Within this dimension, value derives primarily from the reduced *processing time* and the improved *collaboration and sharing effects*. The processing time measurement of the traditional and the new enterprise mashup solution proves the time savings in all decision phases. The passive support based on the integrated context approach also enables the collaboration and sharing. By suggesting relevant information building blocks (widgets), the user is able to identify and to solve an ad-hoc event. No time-consuming documentation is required from the information worker. Both effects have a very positive impact on the observation and orientation phase as indicated in Table 1.

Second, **informational** effects emerge from the enterprise mashup to combine, to process and to disseminate information to the relevant users. The business values associated with these effects are *reduced process breaks* and an *improved decision quality*. Process breaks within and between the phases are eliminated. Thus, more efficient working processes are established resulting in a seamless information processing. No switching between different applications is required. Even if the uncertainty in decisions cannot be eliminated at all, the obtaining of heterogeneous resources improves the decision preparation and therewith the quality of the decision process. In particular, in times of crisis where the information worker has to react on up to 100 alerts per day, the information and automational effects leads to a new stage of user productivity.

Third, **transformational effects** refer to the value facilitating and supporting process innovation and transformation. Faster and better decision making with enterprise mashups increases the *competitiveness* of military organizations. In the context of the Command and Control Maturity Model (Alberts & Hayes, 2003, p. 109), this approach brings us much closer to the upper right hand quadrant of shared awareness and self-synchronization.

4 Conclusion

The aim of this article is to report about how to facilitate the building and collaboration through social software in enterprise mashup environments. In order to achieve this, we apply the case study research methodology. After defining the main terms related to enterprise mashups and after presenting the technical concept of context-aware enterprise mashups, we report about a typical defense scenario. The integration of social software allows automating unstructured decision processes. The business value analysis proves the impacts on the operational excellence. The structuring and analysis along the four phases of the OODA loop allow generalizing the results to other defense processes.

In particular, the implemented scenario demonstrated how the decision maker in military organizations can easily construct an enterprise mashup for responding to an emerging situation. Based on information from the social context, the SAP Research RoofTop prototype assists the decision maker in a passive from. This adaptive framework also facilitates coordination and cooperation in a collaborative workspace to ensure that the best decision is made. Besides a reduced decision time and decision quality, it leads to self-synchronization and shared awareness.

Open issues are related to the integration efforts of legacy systems. As demonstrated by means of the case study, the sourcing of existing systems implicates additional costs. At this stage, several backend systems aren't ready for the integration into mashups. That's why future work will deal with an effort and cost estimation of its actual integration and its integration paths.

References

Alonso, G., Casati, F., Kuno, H., & Machiraju, V. (2004). Web Services: Concepts, Architectures and Applications. Berlin et al.: Springer.

Albert, D.S., Hayes, R.E. (2003). Power to the Edge, Command … Control … in the Information Age. Command & Control Research Program.

Bitzer, S., & Schumann, M. (2009). Mashups: An Approach to Overcoming the Business/ IT Gap in Service-Oriented Architectures. In M. L. Nelson, M. J. Shaw & T. J. Strader (Eds.), Value Creation in E-Business Management (Vol. LNBIP 36, pp. 284-295). Berlin et al.: Springer Verlag.

Cherbakov, L., Bravery, A., Goodman, B. D., Pandya, A., & Baggett, J. (2007). Changing the Corporate IT Development Model: Tapping the Power of Grassroots Computing. IBM Systems Journal, 46(4), 743.

Chui, M., Miller, A., & Robert, R. R. (2009). Six Ways to make Web 2.0 work. The McKinsey Quarterly, February 2009, 7.

Gootzit, D., & Bradley, A. (2009). Key Issues for Enterprise Mashups 2009. Gartner Research, G00166227.

Hammes, T. X. (2006). The Sling and the Stone: On War in the 21st Century. St. Paul, MN, USA: Zenith Press.

Heier, J. E. (2008). Mashup the OODA Loop. Paper presented at the 13th International Comannd and Control Research and Technology Symposia (ICCRTS), Seattle, WA.

Hoyer, V., & Fischer, M. (2008). Market Overview of Enterprise Mashup Tools. Lecture Notes in Computer Science (LNCS), 5364, 708-721.

Hoyer, V., Gilles, F., Fleischmann, K., Dreiling, A., & Stanoesvka-Slabeva, K. (2009). SAP Research RoofTop Marketplace. Paper presented at the 2nd Workshop on Mashup, Enterprise Mashups and Lightweight Composition on the Web (MEM) in conjunction with the 18th International World Wide Web Conference (WWW2009).

Hoyer, V., & Stanoesvka-Slabeva, K. (2009a). Generic Business Model Types for Enterprise Mashup Intermediaries. In M. L. Nelson, M. J. Shaw & T. J. Strader (Eds.), Value Creation in E-Business Management (Vol. LNBIP 36, pp. 1-17). Berlin et al.: Springer Verlag.

Hoyer, V., & Stanoesvka-Slabeva, K. (2009b). Towards a Reference Model for grassroots Enterprise Mashup Environments. Paper presented at the 17th European Conference on Information Systems (ECIS2009).

Kasper, B. M. (2001). The End of Secrecy? Military Competitiveness in the Age of Transparency. Alabama: Center for Strategy and Technology, Air War College.

Lee, A. (1999). Inaugural Editor's Comments. MIS Quarterly, 23(1), V-XI.

Lincourt, D., Peukert, H., & Kowalkiewicz, M. (2008). Situational Common Operating Picture for Collaborative Sensemaking. Paper presented at the 13th International Command and Control Research and Technology Symposium (ICCRTS): C2 for Complex Endeavors.

McAfee, A. P. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. MIT Sloan Management Review, 47(3), 21-28.

Mintzberg, H., Raisinghani, D., & Theoret, A. (1976). The Structure of "Unstructured" Decision Processes. Administrative Science Quarterly, 21(2), 246-275.

Mooney, J. G., Gurbaxani, V., & Kraemer, K. L. (1996). A Process-Oriented Framework for Assessing the Business Value of Information Technology. ACM SIGMIS Database, 27(2), 68-81.

Richards, C. (2004). Certain to Win: The Strategy of John Boyd, Applied to Business: Xlibris Corp.

Stalk, G., & Hout, T. M. (2003). Competing Against Time: How Time-based Competition is Reshaping Global Markets: Free Press.

Tapscott, D., & Williams, A. D. (2006). Wikinomics: How Mass Collaboration Changes Everything. New York: Portfolio.

Webster, J., & Watson, R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Quarterly, 26(2), xiii-xxiii.

Yin, R. K. (2009). Case Study Research. Design and Methods (4th ed.). Thousand Oaks, CA: Sage.