Americas Conference on Information Systems (AMCIS)

AMCIS 2009 Proceedings

Association for Information Systems *Year* 2009

Generic Business Model Types for Enterprise Mashup Intermediaries

Volker Hoyer* Katarina Stanoevska-Slabeva†

*University of St. Gallen, volker.hoyer@unisg.ch

†University of St. Gallen, katarina.stanoevska@unisg.ch

This paper is posted at AIS Electronic Library (AISeL).
http://aisel.aisnet.org/amcis2009/157

Generic Business Model Types for Enterprise Mashup Intermediaries

Volker Hoyer
University of St. Gallen, Switzerland
SAP Research St. Gallen, Switzerland
volker.hoyer@unisg.ch

Katarina Stanoevska-Slabeva
University of St. Gallen, Switzerland
katarina.stanoevska@unisg.ch

ABSTRACT

The huge demand for situational and ad-hoc applications desired by the mass of business end users led to a new kind of Web applications, well-known as Enterprise Mashups. Users with no or limited programming skills are empowered to leverage in a collaborative manner existing Mashup components by combining and reusing company internal and external resources within minutes to new value added applications. Thereby, Enterprise Mashup environments interact as intermediaries to match the supply of providers and demand of consumers. By following the design science approach, we propose an interaction phase model artefact based on market transaction phases to structure required intermediary features. By means of five case studies, we demonstrate the application of the designed model and identify three generic business model types for Enterprise Mashups intermediaries (directory, broker, and marketplace). So far, intermediaries following a real marketplace business model don't exist in context of Enterprise Mashups and require further research for this emerging paradigm.

Keywords (Required)

Enterprise Mashups, Business Models, Intermediaries, Interaction Phase Model, Design Science.

INTRODUCTION

Motivation and Problem Scope

Since the beginning of the 1990s, companies have optimized their corporate IT by introducing transaction systems such as enterprise resource planning (ERP), customer relationship management (CRM), or supply chain management (SCM). By following a process-oriented approach (Hammer and Champy 1993) and evolving towards modular Service-Oriented Architectures (Alonso et al. 2004), IT departments were enabled to adapt their automated IT systems according to their business needs. The next wave in corporate technology adoption, the Web 2.0 and peer production philosophy, addresses ad-hoc and situational application (Chui et al. 2009). In this context, a new trend for software development paradigm, known as Enterprise Mashups, has gained momentum. Enterprise Mashups bridge the gap between the automation transaction and the peer production world as indicated in Figure 1. The market research institute Gartner identifies the paradigm in the top 10 strategic technologies for 2009. Forrester also predicts that Enterprise Mashups will be coming to a $700 million market by 2013 (Young 2008).

At the core of the Mashup paradigm are two aspects: First, empowerment of the end user to cover ad-hoc and long tail needs by reuse and combination of existing software artefacts. Second, broad involvement of users based on the peer production concept. According to Yochai Benkler, who coined the term peer production, *"it refers to production systems that depend on individual action that is self-selected and decentralized rather than hierarchically assigned"* (Benkler 2006). Thereby, the creative energy of large number of people is used to react flexible on continuous dynamic changes of the business environment. Instead of long-winded software development processes, existing and new applications are enhanced with interfaces (so-called Application Programming Interfaces, APIs) and are provided as user friendly building blocks.

Companies considered this trend and opened their IT systems for their ecosystem (customer, supplier, government, etc.) by encapsulating them via well defined APIs. In addition, the Internet evolves towards a programmable platform. Web providers offer value added services to the Internet community. Besides simple services such as news feed, weather information, maps, or stock information, business relevant services such as storage, message queuing, or payment came up in the last years.

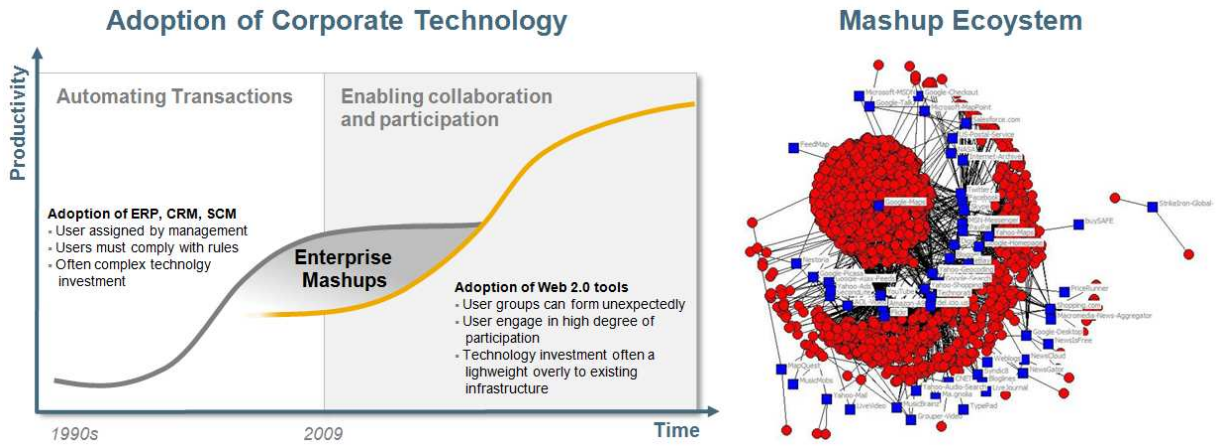

Figure 1. Adoption of Corporate Technology (adapted from Chui et al. 2009) and Mashup Ecosystem (Yu, 2008)

The explosive growth of these mashable components[1] and the emergence of the Enterprise Mashup paradigm (Hoyer and Fischer 2008) will have an enormous effect on intermediation. As indicated in Figure 1, existing services (rectangles) are composed to new value added applications (cycles) in an ad-hoc fashion. Existing research efforts focus mostly on technical aspects as well as relevant platform and tools for the composition of these components – i.e., IBM Mashup Center, Intel Mash Maker (Ennal et al. 2007), Microsoft Popfly, and SAP Research RoofTop Marketplace (Hoyer et al., 2009). The underlying technical concepts and principles are presented by Maximilien et al. (2008), Yu et al. (2008), or Hoyer et al. (2008). However, the discussion of the intermediary role from a business perspective of these Enterprise Mashup environments is still missing in the scientific community. Important questions in this context are: Which features have to be provided by Enterprise Mashup intermediaries to match the supply and demand? What generic business model types exist?

The goal of this research paper is to fill this gap by designing an interaction phase model for Enterprise Mashup intermediaries. The general research questions guiding this research are to model the required features regarding from a consumer and provider perspective as well as to identify generic business model types for Enterprise Mashup intermediaries.

Research Design: Design Science applied

All activities within a research project as well as its scope are defined by the research design. For answering the research questions motivated in the previous section and characterized by a practical nature, engaged research is needed in order to provide rigorous solutions. Design science research aims at solving practical and theoretical problems by creating and evaluating IT artefacts indented to solve identified organizational problems. Hence, it is considered as a problem-oriented approach (Hevner et al. 2004). Artefacts represent the final result of a design process. They can be characterized as constructs, model, methods, or instantiations (March and Smith, 1995).

To come to rigorous and relevant research results, we draw upon on Peffers et al. (2008) to specify the following phase of the design science research process applied:

1. **Problem Identification and Motivation.** In section one, we specify the specific research problem, show the practical relevance and justify the value of a solution. Based on the problem scope, we derive the research questions guiding this paper.
2. **Define the Objectives for a Solution.** In the second section, we infer the objectives of a solution from the problem definition and knowledge of the state of problems. A literature review in section two presents the state-of-the-art of Enterprise Mashups, describes the interacting agents and their roles (consumer, provider, and intermediary) and presents a business model hierarchy to structure relevant terms and concepts of business models.
3. **Design and Development.** In section three, we propose an interaction phase model artefact based on a literature review in order to structure the features of Enterprise Mashup intermediaries. Thereby, we built on the research results of Legner (2008), Hoyer and Stanoesvka-Slabeva, (2008), and Carrier et al. (2008) who observed many similarities of Enterprise Mashup environments and marketplaces. Enterprise Mashup intermediaries should enable

[1] 1171 Mashup APIs (http://programmableweb.com), 27.813 online Web Services (http://seekda.com)

the matching of supply and demand in a way similar to conventional market phases (knowledge, intension, contract/ design, and settlement).

4. **Demonstration.** By means of five case studies of relevant Mashup intermediaries (StrikeIron, Seekda, ProgrammableWeb.com, iGoogle, and IBM Mashup Center), we demonstrate the application of the designed artefact in section four. In addition, we identify three generic business model types for Enterprise Mashup intermediaries: Directories, brokers and marketplaces.

The results of each of the above activities are presented in the remaining part of the paper. Finally, the last section closes the paper with a brief summary, limitations of the conducted research and an outlook to further research.

OBJECTIVES OF THE SOLUTION: BACKGROUND AND RELATED WORK

Enterprise Mashups – Definition of Terms and Characteristics

In literature, the exact definition of Enterprise Mashups is open to debate. In this work, we refer to the definition of Hoyer et al. (2008). *"An Enterprise Mashup is a Web-based resource that combines existing resources, be it content, data or application functionality, from more than one resource in enterprise environments by empowering the end users to create and adapt individual information centric and situational applications"*. Thereby, Enterprise Mashups focus on the User Interface integration (Daniel et al. 2008) by extending concepts of Service-Oriented Architecture (SOA) with the Web 2.0/ Peer Production philosophy (Janner et al. 2007).

With the assistance of a layer concept, the relevant components and terms can be structured in an Enterprise Mashup Stack consisting of the elements resources, widgets, and Mashups (Hoyer et al. 2008). **Resources** represent actual contents, data or application functionality. They are encapsulated via well-defined public interfaces (Application Programming Interfaces; i.e., WSDL, RSS, Atom, CSV, etc.) allowing the loosely coupling of existing Web-based resources – a major quality of SOA (Alonso et al. 2004). These resources are provided by existing enterprise systems or Web providers (i.e., Amazon, Google, etc.) and are created by traditional developers who are familiar with technical development concepts.

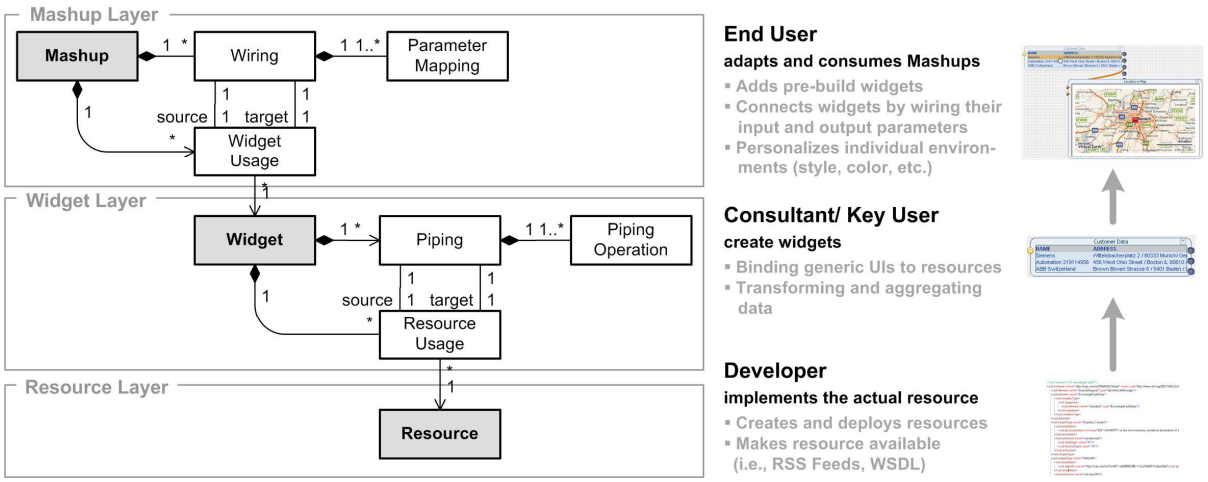

Figure 2. Enterprise Mashup Stack – Meta Model and User Roles (Hoyer and Stanoevska-Slabeva, 2009).

The layer above contains **widgets** which provide graphical and simple user interaction mechanism abstracting from the underlying technical resources. In reference to the UNIX shell pipeline concept, a so-called *piping* composition allows the integration of heterogeneous resources defining composed processing data chains/ graphs concatenating successive resources. Aggregation, transformation, filter, or sort operations adapt, mix, and manipulate the content of the underlying resources. The creation of the widgets and the piping composition can be done by consultants or key users from the business units who understand the business requirements and know basic development concepts..

Finally, the end users from the business units are empowered to combine and configure such visual widgets according to their individual needs, which results in a **Mashup**. Thereby, the visual composition of input and output parameters of the widgets on the Mashup layer is called *wiring*. For example, the sales person Tim uses daily a "Customer Data" widget, which requests resources from the backend Enterprise Resource Planning system. By wiring this widget with a "Google Maps", Tim

can display the customers on an interactive map as depicted in the figure below. He doesn't need to contact his IT department.

In addition to the lightweight composition styles (wiring and piping) by reusing existing building blocks in new ways, the mass collaboration principle from the Web 2.0/ Peer Production wave is also an important characteristic. The willingness of users to offer feedback to the Mashup creator, who may be unaware of problems or alternative uses, directly contributes to the adoption of the Mashup and can foster its ongoing improvement. Rating, recommending, tagging or sharing features for the different Enterprise Mashup layers support the collaborative reuse of existing knowledge to solve ad-hoc business problems.

Interaction Agent Model

From a conceptual perspective, Enterprise Mashups put a face on Service Oriented Architectures by abstracting from the underlying technical protocols by means of small modular components which can be composed according to individual needs. To describe the relationship between the mashable components (Mashup, widget, and resource) and the interacting agents as well as their tasks and roles, we refer to the following interaction model well known in Service-Oriented Architectures (Papazoglou 2003) but also in electronic markets (Sarkar, Butler, and Steinfield 1995, Legner 2007, Hoyer and Stanoevska-Slabeva, 2009): A *provider* develops and publishes a mashable component via an *intermediary*, where a *consumer* can find it and subsequently may compose and consume it.

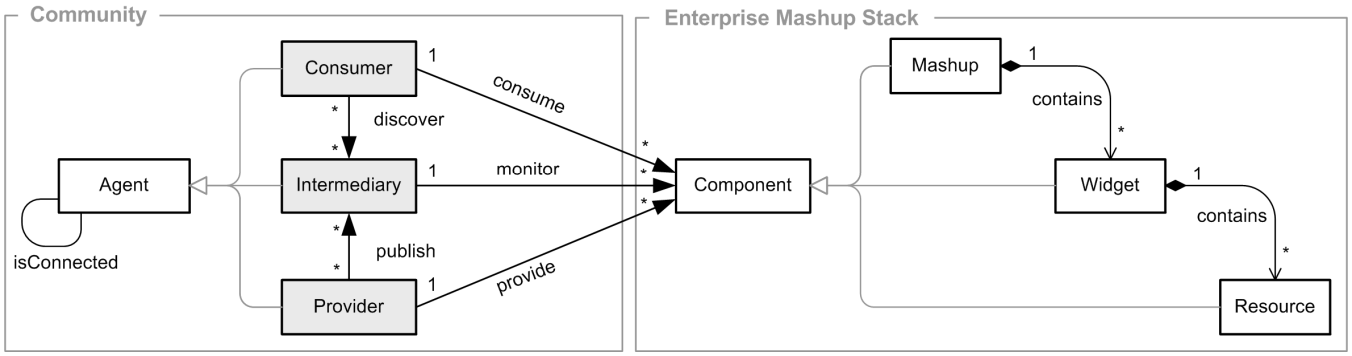

Figure 3. Interaction Agent Model for Enterprise Mashups

As depicted in the figure above, the interaction between consumers and providers is always managed by an intermediary. The tasks of the three agent roles are described in the following:

1. **Provider.** A provider implements and hosts a Mashup component which encapsulates the actual content or knowledge. To promote their provided functionalities, the provider annotates the component with relevant information and publishes it to an intermediary through which the component description is published and made discoverable.
2. **Intermediary.** An intermediary mediates and coordinates between providers and consumers in order to match the supply and demand in a way similar to electronic markets (Legner 2007). Available components are classified and offered by providers and potential customers search for the most suitable ones and if required pay for the usage. In contrast to traditional SOA-based specifications like UDDI or ebXML (Dustdar and Treiber 2005) that provides only directory services to find a component, novel forms of intermediaries are currently about to emerge which improve navigation, transparency, and governance. They monitor continuously the parameters (such as availability or response latency) and provide performance metrics and other evaluation results which may be used by potential consumers to select a right Mashup component (Schroth and Christ 2007). Thus intermediaries play an important role in structuring and classifying the available Mashup components, in providing a platform that can host a Mashup community, in facilitating the process of Mashup integration and in facilitating the process of Mashup payment and delivery.
3. **Consumer.** Based on the information provided by the intermediary, a consumer is able to retrieve a Mashup component according to his/ her individual preferences. Consumers take also over the role of annotating Mashup components by tagging, recommending, or rating them. Therewith, consumers create indirectly a folksonomy, essential a bottom-up, organic taxonomy that can be used to organize the growing number of Mashup components.

According to the peer production characteristic of Enterprise Mashups, users often act as consumer and provider. For example, Tim working in the sales department creates a Mashup by combining a "Customer Data" widget with the "Google Maps" widget. During lunch time, he mentions the Mashup during a discussion with his manager who is also interested in it. So Tim publishes the Mashup (provider role) and recommends it to his manager who is now able to use the Mashup as well. In this sense, he contributes to the community base by providing a created and adapted Mashup back in the community pool.

Business Models

The term business model has been predominantly coined in practice culminating in a buzzword status during the dot.com period. Only gradually it has been adopted and researched by the scientific world (Morris et al. 2005). The concept of the business model is not new, but for a long time the focus in scientific analysis of firms has been on industry (Porter 1980) and resources (Wernerfeld 1984). The business model shall be deemed to be the replacement or complement of the traditional unit of analysis as a result of the changed surrounding conditions. The business model concept itself has been subject of a series of publications (Afuah and Tucci 2001, Osterwalder et al. 2005, Timmers 1998). However, a universal definition has not formed until today, what hinders the realization and comparability of empirical investigation (Morris et al. 2005).

In order to structure relevant terms and concepts, we refer to a business model concept hierarchy proposed by Osterwalder et al. (2005). It classifies business models in three different layers that are hierarchically linked to each other.

1. A **business model concept** is an abstract overarching concept that can describe all real world businesses. This level consists of definitions of what a business model is and what belongs to them. In this work, we refer to the definition of Timmers, who defines a business model as *"[…] an architecture for product, service and information flows, including a description of the various business actors and their roles; and a description of the potential benefits of the various business actors; and a description of the sources of revenues."* (Timmers 1998). Stanoevska-Slabeva and Hoegg (2005) leverage this definition and its business model components as a foundation and enrich it with additional relevant aspects. The resulting business model concept framework consists on seven major components: First the *Features of the Specific Product* comprises the actual design of a product or service, the way it is perceived and consumed by the customer and the value proposition for the customer. The component *Features of the Specific Medium* defines possibilities for transaction and interaction via certain media between the stakeholders of a business model from a technical point of view. The *Customers* component refers to the target groups of an offered product or service and explains their respective business needs. Fourth, the *Value Chain* component is devoted to reflecting all players that are involved in the production and delivery of a product and their respective interrelationships. The component *Financial Flow* identifies in which way the products and services are monetized and explain the roles different stakeholders play. *Flow of Goods and Services* describes the stakeholders' activities that are essential for the creation of the product or services. Last, the *Societal Environment* reflects relevant outside influences on a business model (e.g., legal aspects and competitive situation).
2. **Types of business models** describe and cluster a set of businesses with common characteristics. This distinction reflects different degrees of conceptualization. Furthermore, the type can be a subclass of an overarching business model concept. The classification of business models in types is discussed intensively in literature. Timmers (1998) identified eleven Internet business models: e-shop, e-procurement, e-auction, e-mall, third-party marketplace, virtual communities, value chain service provider, value chain integrator, collaboration platforms, and information brokers. Rapa (2007) proposes a classification of nine Web business model types: brokerage, advertising, infomediary, merchant, manufacturer, affiliate, community, subscription, and utility.
3. A **reald world business model** presents aspects of or a conceptualization of a particular company. This level consists of representations, and descriptions of real world business models.

DESIGN: INTERACTION PHASE MODEL FOR MASHUP INTERMEDIARIES

The design activity of our research is structured according to the business model concept hierarchy. We design an interaction phase model representing a conceptual model to analyze required services (business model component *Features of the Specific Product*) for Enterprise Mashup intermediaries. The model is based on existing concepts and theories from the scientific knowledge base as proposed by Hevner et al. (2004) for design science research.

Legner (2008), Hoyer and Stanoesvka-Slabeva (2008), and Carrier et al. (2008) observed many similarities between the Enterprise Mashup paradigm and electronic markets; Enterprise Mashup intermediaries match the supply and demand between providers and consumers. In order to structure and design an interaction phase model for Enterprise Mashup intermediaries, we leverage the St. Gallen Media Reference Model (Schmid and Lindemann 1998) due to its roots on

electronic markets and due to its successful application for structuring Enterprise Mashup environments (Hoyer and Stanoevska-Slabeva, 2009).

The interaction phase model between the three agent roles (consumer, provider, and intermediary) is structured according to the four market transaction phases. Starting with the *knowledge phase*, the agents of the Enterprise Mashup environment are able to find information about the offered mashable components (resources, widgets, or Mashups) and about the agents. During the *intention phase*, the agents signal their intention and needs in terms of offers and demands regarding the mashable components. In the *contract (design) phase*, consumers combine different mashable components, configure it according to their preferences to new value added applications in order to solve ad-hoc business requirements. Finally, in the *settlement phase* the Enterprise Mashup is executed according to the contract/ design using the Enterprise Mashup environment's settlement services offered for this purpose.

In addition to these market phases, we use the findings of Sarkar et al. (1995) and Legner (2008), who identified relevant features of intermediaries in electronic markets for mediating between consumers and providers. Figure 4 depicts the resulting interaction phase model by using the Business Process Modeling Notation (BPMN). The interaction process is characterized by permanent loops between the four phases (converging design and runtime). The need to adapt the operational environment in an ad-hoc manner leads to adding, removing, or replacing existing mashable components.

Knowledge Phase. After registering to the Enterprise Mashup environment, both agent roles consumer and provider are able to discover the Mashup community, the members, and the provided features of the Enterprise Mashup environment. By means of interactive demonstrations in form of short videos and tutorials, the benefits of the Enterprise Mashup environment are demonstrated to the potential customers. Only if a huge amount of agents are convinced of using the environment, it will exploit its actual potential. In addition, the usage conditions and fees are communicated. By aggregating the continuously monitored consumption data, in particular, providers of a mashable component are able to identify new trends and to evaluate the success of new developed mashable components. The aggregated information – for example the reputation of a provider or the quality of a mashable component (i.e, availablity, reliability, popularity, etc.) – reduces the risk for consumers to select and to use a mashable component that does not fulfill required performance aspects. By certifying mashable components or providers indicating compatibility, trust or reputation aspects, the Mashup intermediary takes care of an improved transparency. On the other side consumers can review, recommend, rate, or share mashable components. All this information is provided to the consumers in order to find and select relevant mashable components. Due to the growing number of components, expert assistant (i.e., wizard) supports the consumer determine their needs according to their context (i.e., industry, department, country) and preferences. Also, providers require services for publishing a Mashup component in order to informate the consumers about the existence and characteristics (underlying business model such as fee, usage license, permission, etc.) of their offer. Ultimately providers are not interested only in providing information for consumers; they are interested in selling their offers by influencing the consumers with service placements

Intention Phase. While in the knowledge phase available components are classified, rated and explained in different ways in the intention phase, the concrete offers are provided in a more structured manner. For example a Mashup component might be purchased based on a subscription or based on pay-per-use. The offer includes the component, the payment mode and price as well as delivery conditions. In context of Enterprise Mashups this might be a description of the quality of service to which the provider is obliged.

Contract (Design Phase). In case the consumer retrieves a mashable component and accepts the underlying business model that is defined by the provider, he/she can compose it with others by connecting the input and output parameters (wiring/ piping). To reduce traditional interoperability challenges, the Mashup intermediary has to provide assistance and to hide the complexity from the consumer who is characterized by limited programming skills. Especially, the composition of information from different and heterogeneous IT systems provided internal and external agents has to be handled in the design phase. In contrast to the classical software development, the design of ad-hoc applications uses real resources and no demo systems.

Settlement Phase. 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 phase to adapt the individual operational environment. As already mentioned before, the Mashup intermediary monitors and protocols all consumption activities. Based on this collected data, the actual billing and accounting process is handled as well as the data aggregation features in the knowledge phase.

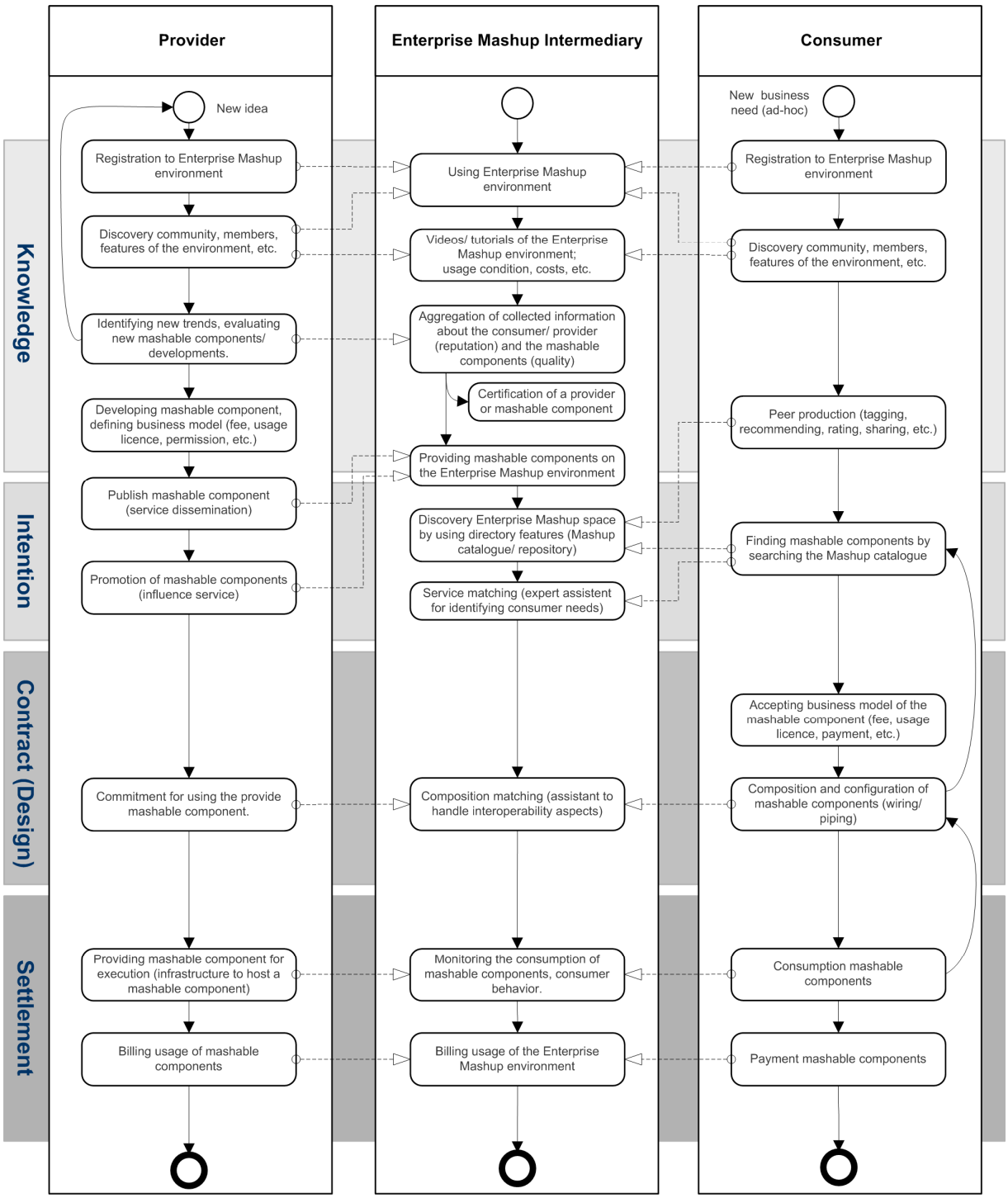

Figure 4. Interaction Phase Model and Features of Enterprise Mashup Intermediaries

Besides these functionalities in the four market phases, we note that often provider and consumer interests are in conflict. So an important intermediary function is to balance and integrate the needs of provider and consumer. For example, a provider of a mashable component may to inform potential consumers about the existence of a mashable component while consumers would rather search and evaluate Mashup components.

DEMONSTRATION: CASE STUDIES

By means of five case studies we demonstrate the application of the designed interaction phase model. According to the business model concept components proposed by Stanoevska-Slabeva and Hoegg (2005) and the designed interaction phase model, we analyze five relevant Mashup intermediaries. They represent business model instances according to the business model concept hierarchy. Summering up, we derive and cluster three generic business model types (directory, broker, and marketplace) for Enterprise Mashup intermediaries and describe their characteristics and provided features.

Instances of Mashup Intermediaries

To mediate between providers and consumers, various Mashup intermediaries arose during the last years. We selected five relevant Mashup intermediaries which focus on different layers according to the Enterprise Mashup Stack. StrikeIron represents a traditional intermediary focusing on heavyweight Web Services resources. An analysis of other traditional intermediaries can be found by Legner (2007). Similar to StrikeIron, the Seekda project, built on results of the EU funded research project DIP[2], focuses on Web Services resources. In addition, Seekda is continuously crawling the Web for services and monitors the performance (in particular the availability) of the services to improve transparency issues as discussed before. ProgrammableWeb.com represents one of the upcoming intermediaries addressing explicit the requirements of the Mashup paradigm. The consumer-oriented iGoogle Gadget Repository provides a simple and initiative navigation concept how to retrieve Mashup components (in this case widgets) by users without any IT skills. Finally, we analyze the IBM Mashup Center hosted by the Greenhouse project of IBM. Due to its business orientation, it gives first impressions about governance aspects that have to be addressed. The results of the case study analysis are depicted in the tables below. In the first table the business models of the five intermediaries are described in general terms based on the structure provided by the generic business model concept of Stanoevska and Hoegg (2005). In the second table the core features of the four market phases applied by the five intermediaries are analyzed and compared.

Intermediary	StrikeIron	Seekda	ProgrammableWeb.com	iGoogle Gadgets	IBM Mashup Center
Potential Customers					
Focus	Inter-Organization	Inter-Organization	Inter-Organization	Inter- and Intra Organization	Intra-Organization
Target Market	Enterprise	Enterprise	Consumer and Enterprise	Consumer and Enterprise (in combination with Google Apps)	Enterprise
Value Chain					
Role	StrikeIron takes over both roles, intermediary and provider. It provides resources from different sources.	Intermediary, research project	Intermediary	Intermediary and provider. The potential benefit of iGoogle is the seamless integration of other Google services (Gmail, Docs, etc.)	Intermediary
Financial Flow					
Revenue Model	Consumer have to pay for using the StrikeIron resources.	Research project, integration of Google Ads	Private Web site, advertisement	-	IBM Mashup Center is a software product. So far, the revenue model is based on software licenses.
Flow of Goods and Services					

[2] http://dip.semanticweb.org/

Intermediary	StrikeIron	Seekda	ProgrammableWeb.com	iGoogle Gadgets	IBM Mashup Center
	All transactions are handled by StrikeIron	Seekda crawls the Web for existing services and monitors their availability.	ProgrammableWeb provides only discovery services	Google provides the presentation layer of the distribute running gadgets	
Features of the Specific Medium (Technology)					
APIs	StrikeIron Marketplace API	-	Full access to the catalogue capabilities based on OpenSearch, Atom Publishing Protocol (Google GData); http://api.programmableweb.com	Google Gadget API	Open Ajax
Web GUI (Portal)	All features are accessible via the Website	All features are accessible via the Website	All features are accessible via the Website	All features are accessible via the Website	All features are accessible via the Website/ Portal

Table 1. Case Studies: Business Model Instances of Mashup Intermediaries

All intermediaries address enterprises as potential customers, while two of them also consider individuals as additional potential customers. Two of the intermediaries are at the same time providers and intermediary. Three are only intermediaries. The five intermediaries differ in the way how they generate revenues. Intermediaries as ProgrammableWeb.com that are basically collecting Mashup components are financed by advertising. The two intermediaries providing also own components apply a certain payment model: StrikeIron a pay-per-use model and the IBM Mashup Center a software licence model. Only StrikeIron seems to cover main parts of all market phases. However, StrikeIron doesn't support the actual design features to compose Enterprise Mashups.

Table 2 summarizes the functionalities offered by the observed cases according to the four market phases.

Intermediary	StrikeIron	Seekda	ProgrammableWeb.com	iGoogle Gadgets	IBM Mashup Center
General Information					
Owner	StrikeIron	University of Innsbruck, STI	John Musser	Google	IBM, Greenhouse Project
Supported mashable components	Resources	Resources (Web Services)	Resources (Web Services, REST, etc.)	Widgets	Resources, Widgets, Mashups
#Components (Mashups, Widgets, Resources)	40 Resources	27813 Resources	1171 Resources, 3731 Mashups	> 50.000 Widgets	475 Resources, 47 Widgets, 107 Mashups
Active	2002 - today	2006 - today	2005 – today	2006 - today	2008 – today
Supported Features of Enterprise Mashup Intermediaries					
Knowledge Phase					
- Registration	Online registration (free)	Online registration (free)	Online registration (free)	Online registration (free)	Online registration (free)
- Support (videos, tutorials, samples, etc.)	-	Introduction how to use Web Services	-	-	Videos on YouTube, sample pages (Mashups)
- Provider description	StrikeIron acts as provider of all resources.	Name, country, home page	Name	Name, company, email address, website	Name, company

Intermediary	StrikeIron	Seekda	ProgrammableWeb.com	iGoogle Gadgets	IBM Mashup Center
- Component description	Name, description, features, benefits, service endpoint, price conditions	Name, provider, country, description, user, rating, tags, availability, service endpoint	Name, provider, description, tags, user rating, date added, technical protocol, security, support and signup/ licensing	Name, provider, description, preview of widgets, popularity, average rating	Name, provider, description, popularity, version, average rating
- Aggregation of collected information about the consumer/ provider (reputation)	-	List of all provided Mashup components	List of all provided Mashup components	List of all provided Mashup components	List of all provided Mashup components
- Aggregration of collected information about the quality of mashable components	Monitoring of performance and uptime	Extensive analysis of the availability (response time,	-	-	Popularity
- Browsing/ discovery of the catalogue	Free text search, sort by categories	Free text search most used, recently found services, providers by country, tag cloud	Free text search, sort by categories, newest, most popular, API scorecard, Mashup matrix tag cloud	Free text search sort by categories, hottest, most users, newest	Free text search sort by categories top ratings, most popular, tag cloud
- Certification of mashable components or providers	-	-	-	-	-
Intention Phase					
- Annotation components (tagging, recommending, rating, sharing, etc.)	-	Tagging, rating, reviewing	Tagging, rating, reviewing	Tagging, rating, reviewing, sharing (integration with Google contacts)	Tagging, rating, reviewing
- Publication of Mashup components	-	Online form to add resources (URL) for the crawling engine	Online form and API to add a resource or Mashup	Adding of new widgets/ feeds (URL)	The provider is also able to specify the permission of the components (view or edit mode)
- Promotion of Mashup components	-	-	-	-	-
- Service matching by wizards (expert assistant)	-	-	-	-	-
Contract (Design) Phase					
- Design and creation of an individual working environment/ application	Sample application allows testing a component.	A Web Services Invoker allows to test a Web Service; a real design environment does not exist	-	Individual environment with several themes and so-called Google gadgets (widgets) which can be added to the environment	It allows to create and individual page by adding widget from the catalogue.
- Composition of mashable components	-	-	-	-	Widgets can be wired and resources can be

Intermediary	StrikeIron	Seekda	ProgrammableWeb.com	iGoogle Gadgets	IBM Mashup Center
					piped.
- Composition matching (assistant to handling interoperability aspects)	-	-	-	-	-
Settlement Phase					
Monitoring the consumption of mashable components and consumer behavior	-	Long term database monitors the availability of the Web Services	The popularity of APIs is documented indirectly by analyzing	Popularity based on the consumer consumption is documented	Popularity based on the consumer consumption is documented
Billing usage of mashable component	Commercial agreements and sales conditions	-	-	-	-
Management of the payment	Online subscription with credit card payment	-	-	-	-

Table 2. Case Studies: The four market phases of Mashup Intermediaries

Generic Business Model Types: Directory, Broker and Marketplaces

Based on the findings of the five case studies presented in the previous section, we observe and identify three generic business model types for Enterprise Mashup intermediaries: Directories, brokers, and marketplaces.

- **Directory.** Similar to traditional repositories well-known in SOA environments such as UDDI or ebXML, directories focus only on the organization, i.e. collection and classification of mashable components. Providers are able to publish a mashable component to the intermediary, where the consumer is able to find it. ProgrammableWeb.com and iGoogle follows a directory business model. Thereby iGoogle covers also the contract (design) phase to create an individual environment. The Mashup directories take a low risk and concentrate on offering added value by just closing the information asymmetry regarding availability of components among providers and consumers. Given this, they can only leverage the available information as a basis for advertising based business models or subscription based business models where providers listed in the directory pay a fee for being listed.
- **Broker.** In contrast to directories, brokers go one step further in diminishing the information asymmetry among providers and consumers. To select and use a component, consumers need additional information concerning the availability, reliability, reputation, or quality. This type of information is provided by brokers (see for example Seekda).
- **Marketplaces.** The third generic business model type is a marketplace. Besides the provided features of brokers, it covers all market phases including the settlement phase with the billing and the accounting features. Only one of the observed cases – StrikeIron - can be considered to be a marketplace.

The analysis of the cases reveals that most of the emerging intermediaries are directories or brokers. Thus, the prevailing intermediaries cover only part of the functionality proposed in the interaction phase model (see figure 4). Marketplaces that completely cover all four market phases are not present yet. This might on the one hand be due to the fact, that trading of mashups and of components for mashups is a very new business area and not mature yet. At the same time, the risk of the intermediary increases the more he covers all four market phases. At the same time the demand for Mashup components might not be mature yet as well. Current low volumes of Mashup trading cannot cover the costs of operating a complete market place. However, intermediaries that offer already broker functionality can evolve to marketplaces when transaction volumes increase.

CONCLUSION AND FUTURE WORK

The aim of this paper is the design of an interaction phase model for Enterprise Mashup intermediaries. In order to achieve this, we follow the design science methodology. After defining the main terms related to Enterprise Mashups and business models, we presented a designed interaction phase model for Enterprise Mashup environments by leveraging the transaction market phases proposed by Schmid and Lindemann (1998) and intermediary features according to Sarkar et al. (1995). By means of five case studies, we demonstrate the application of the model artefact. We observed three generic business model types for Enterprise Mashup intermediaries (directory, broker, and marketplace) and described their characteristics. Figure 5 depicts the results of this research according to the business model concept hierarchy (Osterwalder et al. 2005).

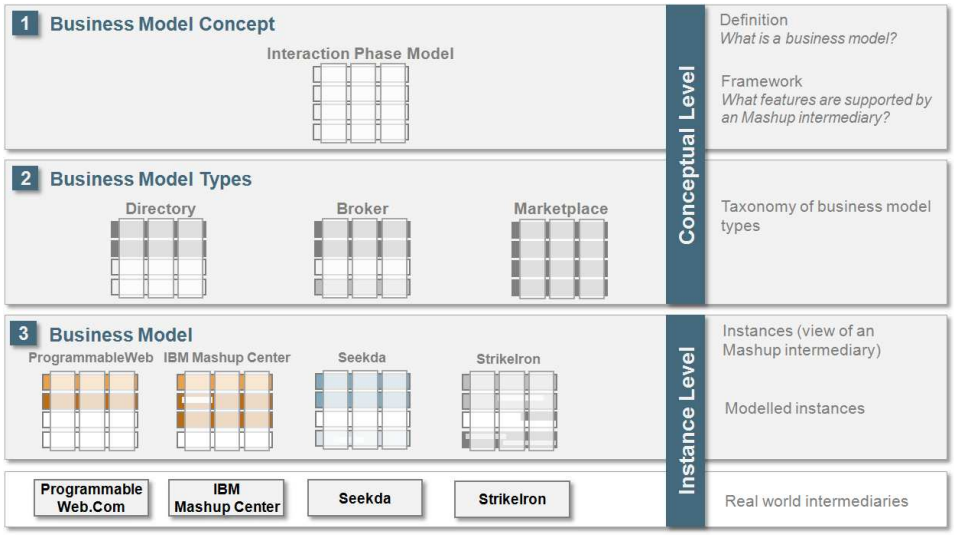

Figure 5. Generic Business Models Types for Enterprise Mashup Intermediaries

What is still missing is a broader application of interaction phase model and the generic business model types. Further research will deal with the design and development of an Enterprise Mashup marketplace which covers all features as identified in this paper. The technical infrastructure will be based on the SAP Research RoofTop Marketplace prototype (Hoyer et al. 2009) and of the EU funded projects FAST/ EzWeb[3] that are currently under development.

ACKNOWLEDGMENTS

This paper has been created closely to research activities during the EU funded project FAST (INFSO-ICT-216048); http://fast.morfeo-project.eu.

REFERENCES

1. Abbot, R. (2006). Open at the Top; Open at the Bottom; and Continually (but Slowly) Evolving, *Proceedings of the IEEE Conference on Systems of Systems Engineering*.
2. Afuah, A. and Tucci, C. (2001). Internet Business Models and Strategies, McGraw-Hill International Editions, New York.
3. Alonso, G., Casati, F., Kuno, H., and Machiraju, V (2004). Web Services Concepts, Architectures and Applications, Springer, Berlin et al.
4. Benkler, Y. (2006) The Wealth of Networks. How Social Production Transforms Markets and Freedom. Yale University Press, New Haven and London.
5. Carrier, N., Deutsch, T., Gruber, C., Heid, M., and Jarret, L.L. (2008). The Business Case for Enterprise Mashups, *IBM White Paper*, August 2008.

[3] http://fast.morfeo-project.eu

6. Cherbakov, L., Bravery, A., Goodman, B., Pandya, A., and Bagget, J. (2007) Changing the corporate IT development model: Tapping the power of grassrots computing. *IBM System Journals*, 46, 4, 743-762.
7. Chui, M., Miller, A., and Roberts, R.P. (2009). Six ways to make Web 2.0 work. *The McKinsey Quarterly*, February 2009
8. Daniel, F., Matera, M., Yu, J. Benatalla, B. Saint-Paul, R., and Casati, F. (2007) Understanding UI Integration. A Survey of Problems, Technologies, and Opportunities. *IEEE Internet Computing*, 11, 3, 59-66.
9. Dustdar, S. and Treiber, M. (2005) A View based Analysis of Web Service Registries. *Distributed and Parallel Databases*, 18, 2, 147-171.
10. Hammer, M. and Champy, J. (1993). *Reegineering the Corporation.* London, Brealey
11. Hevner, A.R., March, S.T., Park, J., and Ram, S. (2004) Design Science in Information Systems Research. *MIS Quarterly*, 28, 1, 75-105.
12. Hoyer, V. and Fischer, M. (2008) Market Overview of Enterprise Mashup Tools. *Proceedings of the 6th International Conference on Service Computing (ICSOC08)*, Springer, 708 – 721.
13. Hoyer, V., Stanoevska-Slabeva, K., Janner, T., and Schroth, C. (2008) Enterprise Mashups: Design Principles towards the Long Tail of User Needs. *IEEE International Conference on Service Computing (SCC'08). Volume 2*, 601-602.
14. Hoyer, V., Stanoevska-Slabeva, K. (2009). Towards a Reference Model for Enterprise Mashup Environments. *Submitted to the 17th European Conference on Information Systems (ECIS'09)*.
15. Hoyer, V., Gilles, F., Fleischmann, K., Dreiling, A., Stanoevska-Slabeva, K. (2009). SAP Research RoofTop Marketplace, *Submitted to 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)*, Madrid, Spain.
16. Janner, T., Canas Vaz, M.A., Hierro, J., Licano, D., Reyers, M., Schroth, C., Soriano, J., Hoyer, V. (2007). Enterprise Mashup: Putting a face on next Generation SOA. *Tutorial presented at the 8th International Conference on Web Information Systems Engineering (WISE 2007)*, Nancy, France.
17. Legner, C. (2007) Is there a Market for Web Services? – An Analysis of Web Service Directories. *Proceedings of the 1st International Workshop on Web APIs and Services Mashups*.
18. March S. and Smith G. (1995) Design and Natural Science Research on Information Technology. *Decision Support Systems*, 15, 4, 251-266.
19. Maximilien, E.M., Ranabhu, A., and Godmadam, K. (2008). An Online Platform for Web APIs and Service Mashups. *IEEE Internet Computing*, 12, 5, 32-43.
20. Morris, M., Schindehutte, M., and Allen, J. (2005). The Entrepeneur's Business Model. Toward a unified Perspective, *Journal of Business Research*, 58, 1, 726-735.
21. Osterwalder, A., Pigneur, Y., Tucci, C.L. (2005). Clarifying Business Models: Origins, Present, and Future of the Concept. *Comunications of the AIS*, 15.
22. Porter, M. (1980). Competitive Strategy Techniques for Analyzing Industries and Competitors, Free Press, New York.
23. Papazoglou, M.P. (2003) Service-Oriented Computing: Concepts, Characteristics and Directions. *Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE).*
24. Plummer, D.C., Bittman, T.J., Austin, T., Cearly, D.W., Smith, D.M. (2008). Cloud computing: Defining and Describing the emerging phenomenon, *Technical Report*, Gartner.
25. Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. (2008) A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24, 3, 45-77.
26. Rappa, M. (2007). Business Models on the Web. *Managing Digital Enterprises*, http://digitalenterprise.org/.
27. Sarkar, M. B., Butler, B., and Steinfield, C. (1995). Intermediaries and Cybermediaries: A Continuing Role for Mediating Players in the Electronic Marketplace. *Journal of Computer Mediated Communication (JCMC)*, 1, 3,
28. Schmid, B. and Lindemann, M. (1998). Elements of a Reference Model for Electronic Markets. *Proceedings of the 31st Hawaii International Conference on Service Sciences (HICCS)*.
29. Schroth, C. and Christ, O. (2007) Brave New Web: Emerging Design Principles and Technologies as Enablers of Global SOA. *Proceedings of the IEEE International Conference on Service Computing*.

30. Stanoevska-Slabeva, K. and Hoegg, R. (2005). Towards Guidelines for the Design of Mobile Services. *Proceedings of the ACIS*.
31. Timmers, P. (1998). Business Models for Electronic Markets, *International Journal on Electronic Markets and Business Media*, 8, 2, 3-8.
32. Webster, J. and Watson, R.T. (2002) Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly*, 26, 2, xiii-xx.
33. Wernerfelt, B. (1984). A Resource-based View of the Firm, *Strategic Management Journal*, 5, 2, 171-180.
34. Young, O.G. (2008). The Mashup Opportunity. How to Make Money in the evolving Mashup Ecosystem. *Forrester Research*, May 6 2008.
35. Yu, J., Benatallah, B., Casati, F., and Daniel, F. (2008) Understanding Mashup Development. *IEEE Internet Computing*, 12, 5, 44-52.
36. Yu, S. (2008). Innovation in the Programmable Web: Characterizing the Mashup Ecosystem. *Proceedings of the 2nd International Workshop on Web APIs and Service Mashups (Mashup08)*, Springer, Sydney, Australia.