Trading semantically enhanced digital products in electronic markets

Wolfgang Maass[a], Wernher Behrendt[b]

[a]Professor of Computer Science in Media
Hochschule Furtwangen University
Robert-Gerwig-Platz 1
78120 Furtwangen, Germany
wolfgang.maass@hs-furtwangen.de

[b]Senior Researcher
Salzburg Research
Jakob-Haringer Strasse 5/II
5020 Salzburg, Austria
wernher.behrendt@salzburgresearch.at

Abstract: Digital products constitute a growing class of economic goods that are increasingly traded via digital networks. In distributed digital networks, it is required that digital products adapt to heterogeneous requirements of electronic markets and subsequent usage environments. Adaptation depends on various kinds of information types that characterise a digital product. In this article we will present a self-describing container model for digital products, called *KCO*, that integrates characteristic semantic information types (facets) derived from an analysis of electronic market structures. Semantic information supports usage of digital products throughout the whole product life-cycle. During purchase decision phases, facets offer semantically annotated data on the quality and applicability of a digital product while in usage contexts, facets provide data that can be used by content management systems for content processing tasks, such as access control, routing, syndication and aggregration. KCOs allow exchanges between existing heterogeneous application environments on the basis of an open Knowledge Content Transfer Protocol (KCTP) that is part of a general architecture (Knowledge Content Carrier Architecture, KCCA). All architectural elements have been used in three application environments.[1] [2]

1 Introduction

One of the assumptions of the Semantic Web is that structured meta-data about information resources provides a better means for human actors and software agents to access, manipulate, delete and create new information resources via digital net-

[1] This work is part-funded by the European Union (6th Framework Programme under the strategic objective IST-2002-2.3.1.7. - Semantic-based Knowledge and Content Systems) and the Swiss Federal Government.

[2] To appear in the proceedings of the 1st. Pragmatic Web Conference (PRAGWEB 2006), 2006 (http://www.pragmaticweb.info)

works [BLHL01]. Both Semantic Web and World Wide Web, are mainly based on an underlying network metaphor that is driven by intrinsic features of free digital contents where information is perceived as a huge reservoir that can be mixed and used independently of economic interests in a the sense that "allows anyone to make statements about any resource." [KC04].
Instead, the concept of an electronic market for commercial digital content is based on an object metaphor, which places a product at the centre that can be appropriated on the seller's side while buyers want to assess the product's quality based on product information during purchase decision making [SV99, CSW97]. In the world of digital products, heterogeneous languages, semantics and interpretations are prevalent due to the fact that the products often stem from different sources (i.e. applications) which have widely differing underlying assumptions about describing the content, its usage and what they regard as meta data.
As a consequence, electronic markets require that digital products (1) interact with services in electronic markets by defined interfaces and (2) carry directly accessible product information, i.e. information that describes the content according to various attributes. These requirements are at odds with the network metaphor of the World Wide Web and even the Semantic Web is not sufficiently helpful yet for modelling economically viable applications for commercial digital products [CSW97].
First we will discuss the requirements from an economic viewpoint before we present a technical container framework that copes with these requirements (section 2). In section 3 we will describe a generic container model for digital product realisations that can be deployed in open, loosely coupled digital information infrastructures, such as the World Wide Web. We will demonstrate how semantic annotations based on the foundational ontology DOLCE facilitate quality assessments of digital products. In section 4, the overall architecture - which has been used within three application cases - is described.

2 Trading digital products on electronic markets

The WWW can be perceived as a huge information market where supply and demand meet. If a digital product has sufficiently high value for actors representing the demand side, it will generate market prices. This kind of content is generally termed "paid content". It describes a special form of information goods and is a subclass of a digital product. ([BK01, CSW97]).
Benefits associated with simple digital products, such as Baseball scores, books, magazines, movies, music, stock quotes, and Web pages (cf. [CSW97]), are clear-cut and the products are easy to use. However, digital products can become more complex such as multi-media contents and knowledge-intensive contents such as descriptions of innovations, best practices, calculation procedures, specific solutions to complex problems, e.g., procedure for welding pipes in nuclear power plants. Generally, digital products are distinguished by three non-disjoint categories [HC02]: (1)

tools and utilities, (2) content-based digital products and (3) online services. [HC02] argue that "information or knowledge [. . .] needs no physical presence and the same idea or information can be conveyed in many ways".

When focussing on knowledge trading environments as part of the life-cycle of a digital product, complex information between digital products and electronic market applications is exchanged during (1) all transaction phases on (2) different logical levels. This information is either provided manually when integrating a digital product, added on at the application level or is part of the digital product itself. Instead of perceiving digital products as being an intrinsic and subordinated part of an electronic market application or database system, digital products can be modelled quite similar to physical products [UE04] as self-describing, semantically rich information objects. This design goal provides an increase of knowledge reusability in heterogeneous knowledge sharing and trading environments if the semantics of this additional information can be at least partially evaluated by an application.

2.1 Performative communication acts in electronic markets

The concept of an electronic market is a metaphor for an economic transaction by an institutionalised sequence of four basic performative communication acts [Au62, Se69]: informing, signalling, contracting and executing [SL98, LBD95]. These performative communication acts are operationalised by electronic services offered by the electronic market application environment [SL98]. Informing acts require information about products that describe product characteristics and which match the buyer's preference set [PBJ93], his/her level of expertise [BW99], on the net value of the benefits and costs of both the product/service and the processes of obtaining it [Ke99]. For signalling the willingness on the seller's and the buyer's side to negotiate, both sides use domain specific signalling acts which are sometimes the first step towards a contracting act, e.g., placing a good into a shopping cart. Signalling can be governed by specific business protocols, e.g., art auctions. Contracting acts specify binding and enforceable procedures [Le88, SJL03] for subsequent exchanges of goods that are part of execution acts, that control financial and product logistic procedures [ACK95].

2.2 Requirements of electronic markets for self-describing digital products

Through an analysis of several hundred paid content providers [SSM04], we have derived five logical levels that cluster semantic information on digital products (for details see [BGM05]): (1) content level, (2) community level, (3) business and legal level, (4) presentation level and (5) trust and security level.

At the content level information about the content (propositions), some characterisation of multimedia features and elements of traditional content classification are represented (e.g. the IPTC classification of News items [IPT05]). At the community level, possible roles, user tasks and historical information are represented. The

	Information	Signalling	Contracting	Execution
Content description	•			•
Community description	•	•	•	•
Business description	•	•	•	•
Presentation description	•			•
Trust and security desc.	•	•	•	•

Table 1: Usage of logic level descriptions during transaction phases in an electronic market

business level represents business and legal constraints that govern possible trading scenarios and contractual requirements. The presentation level gives information on spatio-temporal and interaction-based renditions. Security and trust-related constraints are represented on security level. First we will discuss requirements given by transaction phases and relate them to logical levels before we present an integrated model.

Full support of market transactions requires appropriate semantic information on logical levels that can be interpreted and used by application environments for trading digital products. Table 1 summarises the relationship between different logical level descriptions and transaction phases. During the information and execution phase, information from all logical levels is required whereas signalling and contracting phases can remain ignorant of the content and presentation descriptions. In the following, we discuss a generic container model for digital products that matches the informtion requirements above.

3 Semantically enhanced digital products

KCOs are less dependent on (proprietary) application environments thanks to embedded semantic information. Derived from the logical levels, the structure of a KCO contains five facets plus a facet for semantic self-description of the internal structure of a KCO. Semantic facet information is grounded in an axiomatisation provided by the foundational ontology DOLCE (for details see [BGM05]).

1. Content Description: multimedia characterization, content classification and propositional content

2. *Presentation Description*: spatio-temporal rendition and interaction-based rendition

3. *Community Description*: user tasks, user community and usage history

4. *Business Description*: negotiation protocol, pricing scheme and contract information

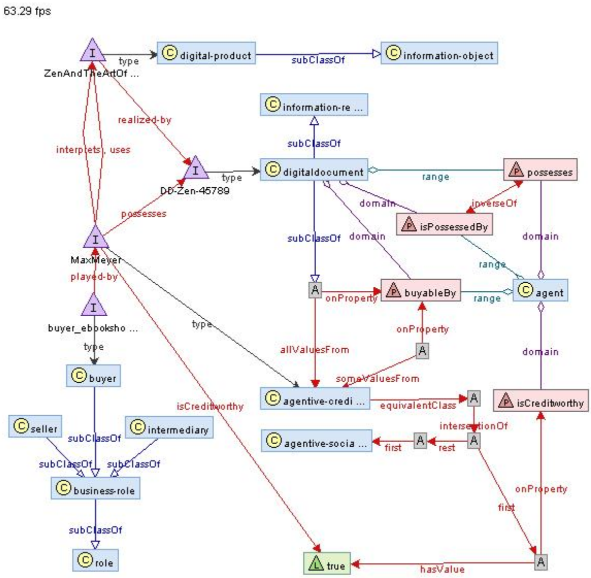

Figure 1: Instantiated design pattern for a digital content product and information object realisation

5. *Trust and Security*: quality assurance information and access control information

6. *Self-description*: specification of the KCO in machine-understandable form.

All facets are initially modelled with concepts of the DOLCE ontology [MBG03, GBC04] which provides an ontological framework for describing digital products [WAM05]. For practical applications domain specific ontologies need to be developed that act intermediaries between foundational and product specific concepts and relationships.

3.1 Example: business facet

Representative for all facets we will discuss the business facet via an example of an ebook of Robert Persig's novel "Zen and the Art of Motorcycle Maintenance". Figure 1 presents a subset of the business facet ontology while integrating only key concepts of DOLCE as interface concepts between a digital product description and a foundational ontology. The novel "Zen and the Art of Motocycle Maintenance" is described as a digital product that in turn is an information object, i.e., an abstract mental entity in the sense of DOLCE [GBC04]. It is realised by a particular digital document "DB-Zen-45789". A digital document can be bought and possessed by an creditworthy agent. For this particular document, Max Meyer qualifies as the agent that bought it. Therefore he can interpret and use the information object provided by this document. This representation describes that a copy of this novel but not the rights to the novel itself can be traded.

Application environments that are able to ingest this representation (see section 4) can infer various conclusions, such that Max Meyer and not Maria Muller is the owner of this particular digital document, that request for "fair use" copies can be sent to him and that he might apply for a refund. In the same way, business descriptions can be used for mapping onto application internal representations, e.g. DRM descriptions (e.g., ORDL [IG04] or XRML [Co02]).

Embedding the semantic description into a foundational ontology (here: DOLCE) supports at least machine interpretation of concepts that are used in facets, for information mapping and other processing, e.g., *information-realization*, *agentive-social-object*, *information-object* and *role*. Some would argue that the use of the foundational ontology is also an important methodological improvement in developing community ontologies, which goes beyond pure agreement on terminology and leads also to agreement in interpretation, for humans as well as machines.

3.2 Implementation of KCOs

The complete KCO model has been implemented on various platforms and for different domains [VLM06]. Digital products were annotated according to the KCO model and transported via a web-based protocol to other applications that leveraged the semantic information. In a publishing context of a media company, KCO-enabled digital products were used to reduce costs during the design of new educational digital products [MSV05]. In a second domain, news from various content sources were annotated which enabled improved targeting of contents according to consumer needs [VLM06]. Another goal of this case was to manage access and usage rights within the business facet by simple rights checking requests. A third application dealt with quality improvements in health care through semantically defined protocols for clinical trials. In all three cases, proprietary information is wrapped into the KCO container and annotated by appropriate semantic information. However, it must be noted that information integration of heterogeneous data sources (wrapping of legacy systems) was not the focus of the project and the proposed wrapping mechanisms do not go beyond the known state of the art in database schema integration, except for the use of a foundational ontology as the common reference.

4 Service-oriented knowledge object exchange environment

The ultimate aim of the KCO model is to become either a standard way of encoding semantically annotated content or to become at least a reference model which is adopted by existing standards committees. It should be noted that we have elected not to take part in any standards activities although there are clear overlaps with new developments in MPEG [SKP02, BH06], notably the MPEG-A [Ch04] initiative which is clearly responding to the problems that organisations have in implementing the MPEG-stack as it has been known in recent years. In developing the KCO model, we wanted to remain independent of any standards politics but rather, give those who work on standards an independent view how content and semantics could be organised for use in future information exchange environments.

The fundamental architecture on which the KCO is based, comes from an insight ascribed to Dijkstra who supposedly characterised all computing endeavours as follows: "computing is data structures plus algorithms"[3]. What Dijkstra implied was that there is always a symbiosis between the data structures one chooses and the algorithms needed to effectively manipulate the structured data. Or, vice versa, given an algorithm there are always data structures which are well suited to the algorithm and others which are not. This principle can - in our view - be applied at all levels of IT systems architecture. Our view of the current IT world is that people working on web services (and *semantic* web services) represent the "algorithms" camp and people working on standards and content models represent the "data structures" camp.

Our intention is to present a model which shows prototypically, the separation of concerns which at the same time, needs close collaboration because in the worst possible world, we will end up with web services - the algorithms - that ignore content standards - the structures - and we will have content standards that ignore web service models. Such a situation will put "computing" at risk, in a fundamental way! And neither of the camps will be able to "go it alone" despite some efforts to that effect.

We developed the Knowledge Content Carrier Architecture (KCCA) as a prototypical and simple distributed system which is intended to provide the services (and thus, the "algorithms" part) of the overall architecture. In a national follow-on project, we are currently bringing together Grid computing, semantic web services (WSMO, [Ro05]) and the KCO model to explore how this could actually work in practice using the most advanced semantic web services model (WSMO) in conjunction with KCOs. In the work reported here, we defined kctp (knowledge content transfer protocol) as a light-weight protocol for the exchange of requests and answers between nodes which are capable of manipulating KCOs.

Each information system which participates in a KCCA universe is represented by a single KCCA node. Inside the KCCA universe, the nodes communicate by using performatives encoded in kctp. The performatives belong to a taxonomic structure which at the top-level, distinguishes only between requests and answers. These are

---

[3] It seems plausible that Dijkstra might have said this, but neither have we found any proven quote on the WWW nor can we claim that he would have drawn the same analogies.

Operation:		*Description:*
`get-kco`		Returns a list of all KCOs available in the current view.
←	KCO [0..n]	
RDF Operation(s)		
1	`SELECT ?kco` `WHERE (?kco, rdf:type, kco:kco)`	

Table 2: KCO operator `get-kco`

then further distinguished into queries and KCO manipulation requests. Each KCO is viewed as a local data- and knowledge base with a predefined fundamental schema and some specialisation depending on the application domain. This means that we can define a set of generic KCO primitives that can be applied to any kind of KCO independently of its refined specification.

4.1 kctp performatives

The kctp performatives define a simple higher-level manipulation language for KCOs which is - on purpose and by design - similar to database languages: We have defined the operators: `query`, `add`, `update` `remove`, `merge`, `render` (for presentation) and they can be applied at the level of KCO facets (`add business model b to KCO k`). These operators are then implemented as KCO RDF operations which work on the underlying KCO RDF model (currently implemented in Jena[4]). Our aim is to define the semantics of these operators more and more in the future. At present, the definition of these semantics is relatively coarse-grain.
To illustrate the problem: if we merge two KCOs whose content is partly free of licenses and partly covered by restrictive licenses then one (conservative) policy could be to make the merged content as restrictive as the most stringent license in any of the two KCOs. To develop a more intelligent license merging policy for KCOs would be an interesting topic for further research.

4.2 KCO RDF operations

The storage for KCOs is defined on an RDF Graph which holds the full description of the KCO. Since these graphs have distinguishable subgraphs thanks to the specification, we can define operations perteining to the KCO, at the level of the RDF Model. In analogy to microprocessors we can say that the KCO RDF operations are the "assembly language" of the KCO world, and the kctp primitives are the higher level programming language for KCOs.
Making external content systems aware of a KCCA enabled system. Since the KCCA world is internally based on the RDF Model of KCOs there is a need to

[4] http://jena.sourceforge.net/

Operation:		*Description:*
create-kco		Creates and Initialises a new KCO-instance in the view. In addition the Agent creating the KCO has to be passed as an Argument (Agent may be a human, a social role e.g. a doctor or even a software agent)
→	KCO [0..n]	
→	Agent	
RDF Operation(s)		
1	ADD (<KCO>, rdf:type, kco:kco)	
2	Call create-content-profile(<KCO>,<Agent>)	
3	ADD (kcoi, rdf:type , ksdf:kco-information)	
	ADD (kcoi, DOLCE-Lite:interpreted-by, <Agent>)	
	ADD (kcoi, DOLCE-Lite:about, <KCO>)	

Table 3: KCO operator create-kco

create "wrappers" for external systems. Again taking ideas from database schema integration of the late 80s and 90s, we use a two-layer integration model: The first level of integration is to map the external schema with as little change as possible, to an equivalent RDF model. This is called the "context profile". Then, we create a mapping from the RDF-context profile to the view which is needed to interact with the KCO-RDF model . This is called the "view profile".

4.3 The architecture of a KCCA node

Any federation of KCCA-connected systems consists of a set of KCCA nodes which are connected to arbitrary applications (see figure 2). For any external application to communicate with its own KCCA node it has to be able to interpret kctp messages and it needs to be able to issue kctp requests. The same is true for backend systems (typically for content management). The kctp messages are transported inside http requests and responses and the query handler de-serialises and serialises the requests (this is like marshalling and un-marshalling in remote procedure calls). Depending on the nature of the request the message is either a query to the registry, the repository or to manipulate a specific KCO. This is handled by the services container.
In the wrapper, the request is translated into the proprietary sequence of operations needed to address the external system and the variables mapping between the kctp request and the external operations is maintained until the response is returned. Then the system serialises the answer into a valid kctp message referencing the KCO RDF Model. The same process applies to front end applications e.g. graphical user interfaces. For any rendering of the information, the KCO's presentation facet actually provides a model for specifying rendering and interaction schemes. This declarative specification can be utilised by external systems, because each KCO brings its own presentation description along to the host system. In this way, the architecture can be seen as a semantically enriched middleware which connects federated, heteroge-

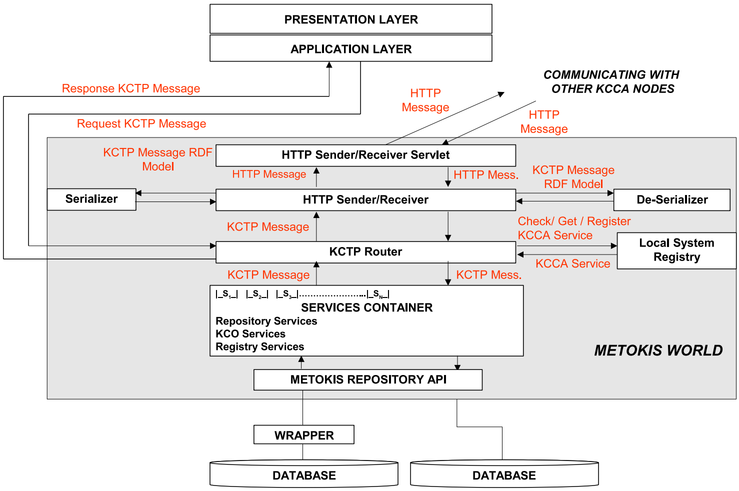

Figure 2: Metokis Architecture

neous content management systems as well as knowledge based systems.

5 Summary and open issues

Currently markets for digital products are built around simple commodity products, such as news, pictures and music [SSM04]. Analogue to the development of electronic markets for physical goods, markets for digital product encompass the potential to grow bottom up towards more complex digital products. But intrinsic characteristics of digital products require dedicated product designs that are generally different than those for tangible goods. The value of digital products is information that cannot be evaluated by buyers without immediately giving away this value [CSW97, SV99]. Semantic information helps buyers to assess the quality of single digital products but also to compare digital products. Automatic processes leverage semantic information so that digital products can be integrated into information systems. Future semantic web services depend on semantic information for advanced automatic processing [MSZ01, Ro05].
Within this context, we have presented (1) with KCO a semantically rich container model for digital products and (2) with KCCA an open integration architecture. In application cases, it has been proven, that the KCO-KCCA combination is able to support the integration of existing commercial information sources, to exchange dig-

ital products and to leverage semantic descriptions carried by KCOs. In next steps, we will set up further KCCA nodes that will allow us to learn more about the requirements of electronic markets for digital products in distributed environments.
In a recent new project, we are investigating the combination of semantic web services with knowledge content objects in a Grid computing environment [5]. The semantic web services should essentially replace the current KCCA and we are exploring what additional semantics are needed for the web services to be a sufficient replacement for a KCCA. Similarly, we are investigating whether the self-description of the KCO provides sufficient information for workflow systems to determine whether a given KCO which is brought into the workflow, can be manipulated somewhere in that workflow. This could lead to self-organising workflow systems. Another issue is that while we now have an infrastructure and an object model, the generic task models and transaction models are still poorly defined. In particular, the connection with business process modelling could be strengthened.
In Metokis, we have only addressed few issues on semantic modelling of digital products. The ontological stack between foundational, domain and product ontologies requires clustering and step-wise construction, stabilisation and enhancement, i.e. ontology management. A natural question is who will create and maintain ontologies for digital products. Can they be created by social networks, such as regular content? From an economic point of view, in future research we will investigate whether semantically rich digital product descriptions have positive impact on decision making of single buyers and in general on diffusion and adoption processes of digital products.

References

[ACK95] R. Alt, I. Cathomen, S. Klein. CIL Computerintegrierte Logistik. Technical report, Instituts fr Wirtschaftsinformatik der Universität St. Gallen, St. Gallen, 1995.

[Au62] J. L. Austin. *How to do Things with Words*. Oxford University Press, New York, 1962.

[BGM05] W. Behrendt, A. Gangemi, W. Maass, R. Westenthaler. Towards an Ontology-based Distributed Architecture for Paid Content. In A. Gómez-Pérez, J. Euzenat (eds.), *Proc. of the 2nd European Semantic Web Conference (ESWC)*, p. pp. 257, Berlin, 2005. Springer.

[BH06] J. Bormans, K. Hill. MPEG-21 Overview v.5. http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm, 2006.

[BK01] S. Boll, W. Klas. ZYX - A Multimedia Document Model for Reuse and Adaptation. *IEEE Transactions on Knowledge and Data Engineering*, 13 (3): 361–382, 2001.

[BLHL01] T. Berners-Lee, J. Hendler, O. Lassila. The Semantic Web. *Scientific American*, May 2001.

[5] http://grisino.salzburgresearch.at

[BW99] L.-T. Bei, R. Widdows. Product Knowledge and Product Involvement as Moderators of the Effect of Information on Purchase Decisions: A Case Study Using the Perfect Information Frontier. *Journal of Consumer Affairs*, 33 (1): 165–186, 1999.

[Ch04] W. Chang. MPEG-A multimedia application format overview and requirements v.2. ISO/IEC JTC 1/SC 29/WG 11. http://www.chiariglione.org/mpeg/standards/mpeg-a/mpeg-a.htm, 2004.

[Co02] ContentGuard. XrML 2.0 Technical Overview. Technical Report version 1.0, 2002.

[CSW97] S. Y. Choi, D. O. Stahl, A. B. Whinston. *The Economics of Electronic Commerce*. Macmillan Technical Publishing, Indianapolis, 1997.

[GBC04] A. Gangemi, S. Borgo, C. Catenacci, J. Lehmann. Task Taxonomies for Knowledge Content. Technical Report D07, METOKIS Deliverable, 2004.

[HC02] K. L. Hui, P. Y. K. Chau. Classifying digital products. *Communications of the ACM*, 45 (6): 73–79, 2002.

[IG04] R. Iannella, S. Guth (eds.). *First International Workshop on the Open Digital Rights Language (ODRL)*, Vienna, Austria, 2004.

[IPT05] IPTC News Classification. http://nets.ii.uam.es/neptuno/iptc/iptc-srs.rdfs, 2005.

[KC04] G. Klyne, J. J. Carroll. Resource Description Framework (RDF): Concepts and Abstract Syntax, Feb. 2004.

[Ke99] R. L. Keeney. The value of Internet commerce to the customer. *Management Science*, 45 (4): 533–542, 1999.

[LBD95] R. M. Lee, R. W. H. Bons, S. K. Dewitz. Performatives, Performatives Everywhere But Not a Drop of Ink. In *16th Annual International Conference on Information Systems (ICIS)*, Amsterdam, the Netherlands, 1995.

[Le88] R.M. Lee. International Contracting – A Formal Language Approach. In *Hawaii International Conference on System Sciences (HICSS)*, Kona, Hawaii, 1988.

[MBG03] C. Masolo, S. Borgo, N. Guarino, A. Oltramari. The WonderWeb Library of Foundational Ontologies. Technical Report D17, WonderWeb, 2003.

[MSV05] W. Maass, M.-F. Schaefer, W. Volz, M. Benari, N. Arora, J. Franz, R. Traphoener, W. Behrendt, R. Westenthaler, H. Ladanyi, A. Gangemi, S. Borgo, C. Catenacci, J. Lehmann. Handbook on the Integrated Knowledge Service Methodology. Technical Report D20, METOKIS, 2005.

[MSZ01] S. MacIlraith, T. Cao Son, H. Zeng. Semantic Web services. *IEEE Intelligent Systems*, 16 (2): 46–53, 2001.

[PBJ93] J. Payne, J. Bettman, E. Johnson. *The Adaptive Decision Maker*. Cambridge University Press, 1993.

[Ro05] D. Roman. Web service modeling ontology. *Applied Ontology Journal*, 1 (1): 77–106, 2005.

[Se69] J. R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge, London, 1969.

[SJL03] M. Schoop, A. Jertila, T. List. Negoisst: A Negotiation Support System for Electronic Business-to-Business Negotiations in E-Commerce. *Data and Knowledge Engineering*, 47 (3): 371–401, 2003.

[SKP02] J. M. Martínez Sanchez, R. Koenen, F. Pereira. MPEG-7: The Generic Multimedia Content Description Standard, Part 1. *IEEE MultiMedia*, 9 (2): 78–87, 2002.

[SL98] B. F. Schmid, A. Lindemann. Elements of a Reference Model for Electronic Markets. In *HICSS*, volume 4, pp. 193–201, Kohala Coast, Hawaii, 1998. IEEE Computer Society.

[SSM04] F. Stahl, F. Siegel, W. Maass. *Paid Content - Paid Services: Analysis of the German Market and Success Factors of 280 Business Models*. =mcminstitute, University of St. Gallen, St. Gallen, 2004.

[SV99] C. Shapiro, H. R. Varian. (eds.). *Information rules - A Strategic Guide to the Network Economy*. Harvard Business School Press, 1999.

[UE04] K. T. Ulrich, S. D. Eppinger. *Product Design and Development*. McGraw-Hill, 3rd edition, 2004.

[VLM06] W. Volz, H. Ladanyi, R. Minio. Final Case Study Prototypes. Technical Report D26, METOKIS, 2006.

[WAM05] R. Westenthaler, N. Arora, W. Maass, W. Behrendt. Final Specification of KCO, KCCA, and KCTP. Technical Report D21, Metokis, 2005.