# Privacy and Security in Digital Networks from the Perspective of Modeling and Software Development

Erich Ortner

Darmstadt University of Technology
Department for Development of Application Systems
Hochschulstrasse 1, 64289 Darmstadt, Germany
ortner@winf.tu-darmstadt.de

**Abstract.** This is a report about cooperation projects with the Common IT-department of the Hessian judicial authority (Gemeinsame IT-Stelle der hessischen Justiz, called GIT in the following) and other Hessian public institutions. In the context of modeling and software development using state-of-the-art schema management technologies, the area of protection and security in digital networks undergoes changes towards more responsibility in dealing with technology, organization and human users. From the perspective of language theory, the available commercial systems such as NetWeaver (SAP), WebSphere (IBM) or SharePoint Server (MS) can be seen as schema-management systems. From the point of view of programming and transaction technology it will be interpreted as servicebase-management systems. Concerning the current protection and security of GIT's application systems two solutions will be presented here, parts of which have already been implemented. Thus, this paper focuses on the standardization of data elements and the standardization of functional elements, which ensure the domain-specific semantic integrity of data and functions in enterprises and administrations. At the end, we will discuss the change we are witnessing among IT experts as well as IT users globally due to language-based technologies such as Semantic Web, Wikis, Google, Facebook, etc., here shown using the example of the Hessian judicial authority. This change is reflected also in education, especially with respect to providing language-based modeling know-how for everybody.

**Keywords:** Network Security, Schemata Management, Language-Based technology.

## 1 Introduction

Language-based computer sciences [Ort05] today are intertwined with nearly every field in economy, science, politics and society. This close technological linkage, "ubiquitous computing", more and more frequently raises the question of how much responsibility computing must assume for crises and undesirable

N. Meghanathan et al. (Eds.): CCSIT 2012, Part I, LNICST 84, pp. 125–137, 2012.

developments, as for example in the recent financial crisis. With the description of the current joint project "Protection and security for the digital networks, stationary and mobile IT devices of the Hessian judicial authorities" we aim at clarifying the following:

1. What are today's requirements from the perspective of protection and security in digital networks as well as for end devices used for the development of application systems (with regard to modeling and software development), e.g. [Bur10], and which of these requirements can actually be fulfilled?
2. Which (language-based) schema management systems are available (e.g. [HR01] and [GR02]), and what is their common theoretical basis (e.g. [WOI]) – particularly in the context of a semantic integrity, i.e. concerning the content, of data and functions?
3. Which consequences derive from 1. and 2. for the education of (young) people by IT experts and IT users [Hei10] and additionally for anyone who is confronted with the "ubiquitous" language-based technologies?

We believe that it is possible to reach agreement among the experts in science and practice today that the IT-development of the last 30 years has partly undergone the following transitions:

**Science:** From hardware-oriented programming to user-oriented and content-oriented modeling.
**IT industry:** From computer-based (algorithmic) technologies to language-based technologies (which integrate the human intelligence).
**User-enterprises:** Shift of focus from data to the human being and finally to the processes.

From the perspective of development and operation of "adaptable application systems" based on language-technology the situation grows more acute: In the era of the Internet, protection and security is also a question of using (normal) languages in a disciplined way. A fact which is still not estimated sufficiently but has great potential for the solution in this context is the following principle: "First modeling with linguistic competence (IT), then the execution based on terms (users)". In the following this will be the basis of our reasoning.

## 2 Protection and Security in Digital Networks – State of the Art

In today's permanent debate about protection and security in digital (and social) networks the following subjects are – sometimes unfortunately in an incompetent and anti-progressive way – being fueled by the global press:

- Cloud Security instead of cybercrime using malware;
- on the Internet, a criminal service society has been established, whose criminal energy could fundamentally endanger states and societies;

– alternatively, some strategy papers (e.g. [Bur10]) rightly and constructively stress that nowadays it is vital to protect the "content" (e.g. externalized knowledge, application software) and no longer to concentrate exclusively on securing access to the systems or their "formal consistency" [Ort05] in enterprises and administrations.

After more than 30 years of research in the field (e.g. since [WO80]), it is a known fact in professional circles that computer science today is more a language-based than computer-based engineering science. Hence, here we look at the perpetual subject "protection and security in IT" – 30 years ago it was decreed to EDV (private economy) and ADV (public sector) – primarily against the background of programming and modeling languages as well as "normal languages" (e.g. users' expert languages) which are used in the development and operation of entire application systems (as opposed to only the software parts). One of the last interviews with Edsger W. Dijkstra (1930-2002), excerpts of which have now been published in Comm. of the ACM, No 8, Vol 53, August 2010, forms an excellent integrated background from the point of view of programming languages and their development as well as software engineering.

Due to the development of various languages, our modern options shift to the content-side of IT-solutions. Naturally, the protection measures of securing the "outer form" or the "containers" for these contents as was central 30 years ago continue to be available.

In Germany, the Federal Data Protection Act and its amendment §6.1, the so-called "ten commandments", regulates the wide range of judicial measures available. It became effective on the first of January 1979. With respect to basic software and software engineering, already prior to the commencement of the Act, since the beginning of the 1970s, a multitude of security concepts and techniques were developed, e.g. identification/authentification, cryptography, access matrix, "system enforced integrity", i.e. data integrity ensured by database management systems. Measured against the "ten commandments", Wedekind described them systematically, and from a methodological and ethical point of view, in Germany also pioneeringly. See [Wed78] and [Wed80].

Looking at this subject again today from the perspectives of adaptive application systems and (data) modeling (e.g. [Ort05]) and schema-based software-development [Pei10], we can base a simple model of states and functions of the information transformation in application systems which also includes the language-based approach, i.e. the users' language usage (see figure 1). Hereby, we use the term "term" [Kue10] mainly as a synonym of the term "type" [Pei10]. Schemas, which can be modeled, represent the universal aspect of things [WOI] and form the intension of "terms" or "types", respectively.

Compared with solutions in the physical world, the central terms of this subject can be defined as follows:

– Reliability is a function-related term. If a car starts without problems every morning, we call this reliable.
– Securing is the act of providing security. Thus, security can be considered to be state-related (see fig. 1): A car may be safe due to its crumple zone.

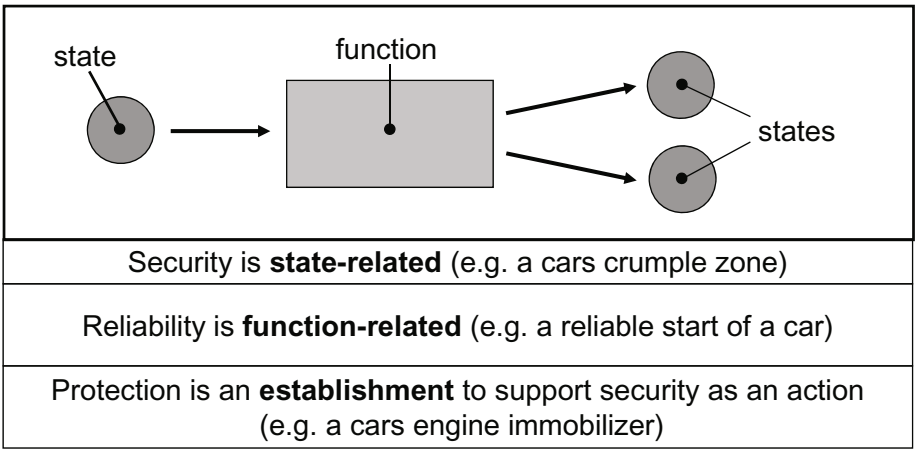


**Fig. 1.** (Data) states and functions in IT-based information processing

– Protection (or privacy for personal data and rights) is an "establishment" for preventing the unauthorized and inappropriate usage of something: A car is protected by its engine immobilizer.

For the protection of content in application systems there are two solutions (establishments): on the one side, data (IT) and externalized knowledge (human beings) and on the other side processing functions (of the IT) and responsible actions (man). Before presenting these in detail, section 3 resumes in short the neutral schema management system architecture, which any application development is based on. As it is this theory which enables managing the above approaches in the first place, namely neutral to changes of content or implementation, as well as appropriate to human beings and IT.

## 3 Servicebase Management Systems and Neutral Schema Management Technology

In line with IBM's slogan "Let us make the planet a little smarter", this section demonstrates how all application fields in IT, as they were practiced, for example, at GIT, can be merged using a neutral schema management technology (basic systems). The foundations of language theory have been regularly published since the 1980s, last in [WOI] as a "constructive schema-tenet" of language-based computer sciences [Ort05].

The joining of application fields through a common, generic basic technology leads us to the implementation of a neutral architecture of schema management systems even prior to the development of the actual application systems (i.e. the object-language schemas). They serve as language-based basic technologies (i.e. basing on a generic theory of "schema and instances", or basic system models) as they may underlie any development of application systems for (digital) networks

as well as IT end-devices, across the operating systems. Figure 2 shows such an architecture of basic systems including in brackets their corresponding primary modeling languages, or their models respectively. In accordance with the inventor of the relational model, Ted Codd, we used the same modeling language for object data (data model) and also for the meta data level (meta model) for database management systems and for the administration of meta data in repositories (meta database management system) [Dat07]. Since their reconstruction these languages are also called rational model-languages in general (fig 2) or in particular rational data models against the background of Lorenzen's ortho-languages and tenet of elementary syntax [Wed81], [Ort82].

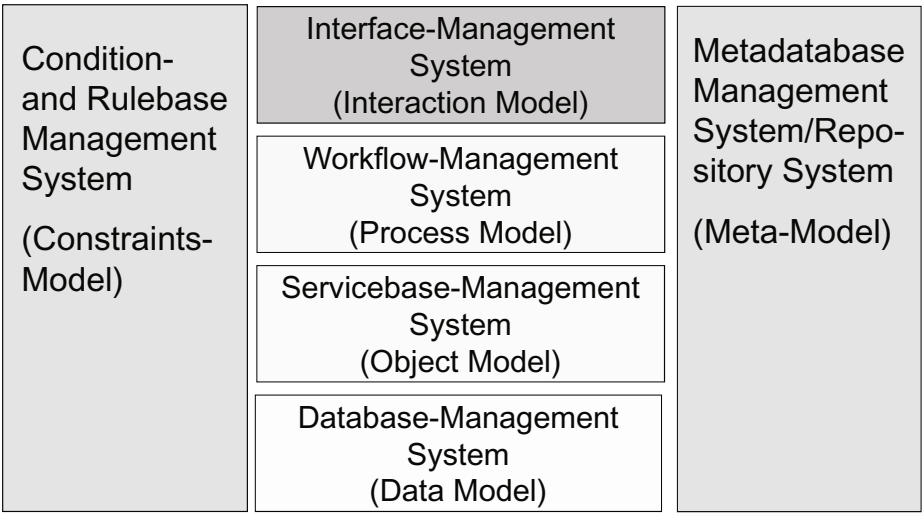


**Fig. 2.** Language-based basic technologies and rational modeling languages

Servicebase-management systems are based on the object model of programming languages such as Java or Enterprise Java Beans (e.g. [BMH06]) for the development of programs or "service-means". We need to consider that with the development of service-oriented architectures (SOA) the term "services" is often used instead of "service means" (software) when algorithms (software) are addressed, which are to be implemented by the IT (really IT-service-means). Here, Paul G. Huppertz has contributed very useful and precise definition work in recent years [Hup06]. His work and results should be integrated in the system development of the various sciences (e.g. computer sciences and business sciences) with regard to a future global "service science" [MKS10].

For workflow management systems (fig. 2) we take the process model as a basis, as to some extent developed in [Fis10]. For the Interface Management System an interaction model, if applicable based on a markup theory (e.g. [Car72]) and the dialogic logic [Inh03] may be sufficient. And for the "Condition and Rule Base-Management System" (fig. 2) we can use a restraints model such as the

SBVR (Semantics of Business Vocabulary and Business Rules) presented by the OMG (Object Management Group). For joining the levels and the components of language-based basic technology (fig. 2) by programming, several technologies may serve. Here, a comprehensive technology from the field of SOA is the so-called "enterprise service bus" (e.g. [EHH+08]). While for example for connecting the service-means-base management system and the database management system, recently JBoss Community's "Hibernate" (e.g. [BK07]) which uses "object relational mapping" (ORM) has become popular. The architecture shown in figure 2 is characterized by the fact that its components are largely neutral towards each other, but also beyond the application system to be developed. This means in particular that changes made to the objects and applications which are maintained in a schema management system are easy to control and have only a limited effect on the changes in another schema management system. This is about the implementation of "neutrality statements" such as "data neutrality", "process neutrality", "controlling neutrality", "rule neutrality", "methods neutrality" or "user neutrality". The architecture is called "Darmstädter Modell" because it has been developed from Darmstadt [HL05] and because of its step-by-step organization and its long-lived orientation for the research in basic systems (since the 1970ies).

## 4 Solution 1: Data Element Standardization

From the perspective of the language-based basic technologies this is a field of application for repository systems, as described for example in [Ort05].

From the perspective of the contents to be managed, basing on this, the data integrity described in this section (e.g. [SW85], [RK88], [ORS90]) and the compliance of application functions – through man as well as IT – described in the next section (e.g. [KH93], [Pei10]) are the subject of solution development.

When standardizing data elements, we need to follow the concept [ORS90] that a sufficient (but limited) number of standardized data elements of an enterprise or administration can be assembled in an assessable and controlled way together with every data storage (files, databases) and uses of data (data descriptions in programs and at user interfaces). The content of these data elements must also be standardized (as related to each particular expert field). In principle, by defining standards for data elements we manage the precepts for the possible propositions about the objects of an organization in the context of all reconstructed (modeled), constructive (basic) constraints, the so-called semantic, structural, pragmatic and technical integrity rules [SW85].

Hereby, data elements are defined as follows [ORS90]:

- A data element is the smallest unit for structuring data or information related to users and/or IT.
- A data element must be defined neutrally with regard to its use and generally as to content, i.e. it is not designed for single, current application situations and from an overall perspective of an enterprise, on schema level (universal form and content).

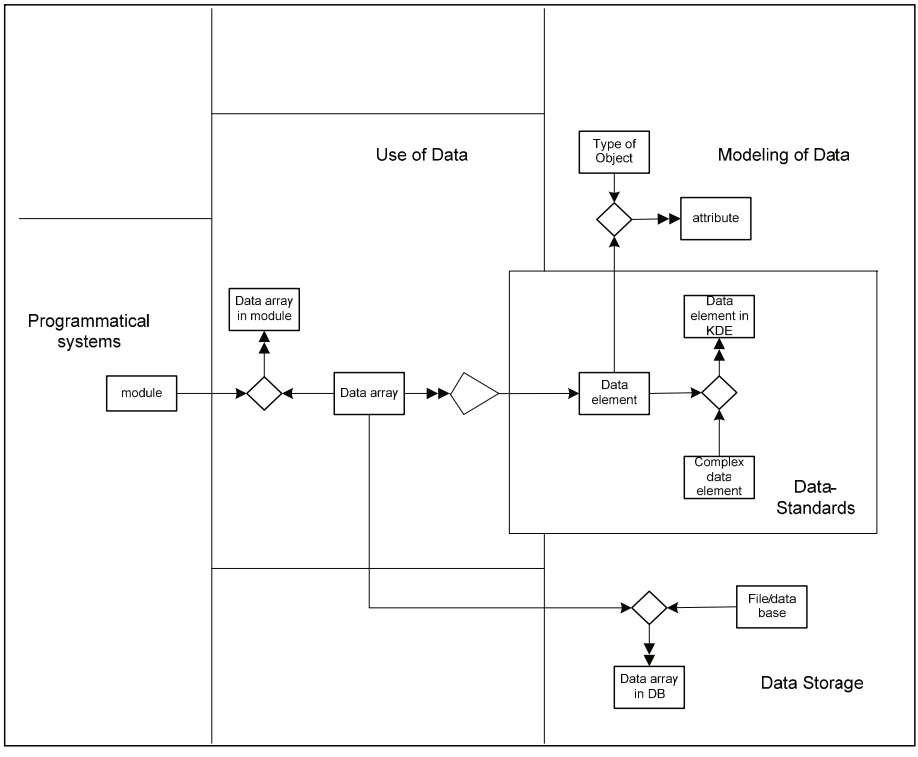


**Fig. 3.** Repository meta schemas for data element standardization and the management of data element standards

– Its "semantics" (meaning, use, subject-specific content) shape the data element, not only its name nor exclusively its form or syntax nor the structure (grammar) of the (subject-specific) linguistic expression.

Therefore the standardization of data elements is an application, implementation, operational and organizational establishment of "system enforced integrity" (e.g. [GR02]) – with the help of a 3-schema organization for the integrity rules (e.g. [RK88]) – on the data resources side in an enterprise or another organizational unit. Figure 3 shows the meta schema (documentation structure) of a repository system [Ort97] which is able to manage data element standards as well as the documentation of their use in the partial systems or partial results of the development. In addition to the repository system, the solution must comprise a further establishment, namely a centralized organization of all integrity rules of an enterprise and, if applicable, in accordance with the 3-schema approach. This is necessary to truly fulfill the requirement of "system enforced integrity".

The decisive advantages of "system enforced integrity" in the case of data element standardization with a centralized management of the integrity rules in an enterprise can be summarized as follows:

- The interfaces between the developed solutions or components are unified (based on content) and the integration and reusability of data elements is improved.
- The user interfaces are subject-specifically standardized with regard to their semantics and put on a common, subject-specific terminology so that users will be able to intelligently handle IT-solutions.
- Redundancies and the multiple use of data collections in an organization can be controlled, which enables effective and efficient data integrity management.

The third component, no less important for reaching the goal of "system enforced integrity", is the realization of the permanent tasks involved in data element standardization and data element management in enterprises or administrations. In [ORS90] and [Ort97] several organizational rules and job descriptions to this avail are discussed.

## 5 Solution 2: Standardization of Functional Elements

The introduction to [Ort82] still conceded that in this work a constructive and language-critical foundation of the development of application systems based on Frege's language logic could only succeed with regard to the data resources of an enterprise. Due to research results such as [Pie02] and [Fis08], however, in the meantime this has proved successful theoretically also for performing the (subject-specific) functions in an organization (enterprise, administration, department, etc.) – at any rate from the perspective of computer sciences or the programming languages and language-based modeling. What was practically possible with data element standardization in enterprises (firstly at DATEV) in the 1980s under IBM's database developers' slogan "system enforced integrity" seems to become possible today under the slogan "system enforced compliance" [BHT09], also with regard to the standardization of functional elements, concerning the IT-functions and user functions of an organization in the context of protection and security requirements, it can even be found on the World Wide Web [SHBL06].

Here, the standardization of functional elements intervenes far more in the planning of actions (e.g. work) and in the event management (e.g. business processes) in an organization than data element standardization. Data element standardization quasi standardizes what can be said in an organization. The standardization of functional elements, however, additionally regulates and controls what happens in an organization.

We want to stress once more, that functional element standards represent normalized and standardized possibilities of operations (and actions) with objects on schema level (event schemas). Their instances must be understood as singular operations or actions which are bound to particular places and times, i.e. singular, volatile events. Figure 4 shows the "content" (schema and instances) of a service-base management system, modeled as a prototype for a repository application for the standardization of functional elements in a service company,

e.g. a hairdresser, because a practical implementation at the Hessian judicial authority is not yet available. A separate but integrated repository [Ort05] was added. It includes an inventory of all the business rules (regulation) of an enterprise or administration (Condition and Rulebase Management System, figure 2) as well as the management of this inventory using language-based basic technologies [FLOZ10] during the development and operation of the application systems as a further establishment that is organized in accordance with the 3-schemas approach.

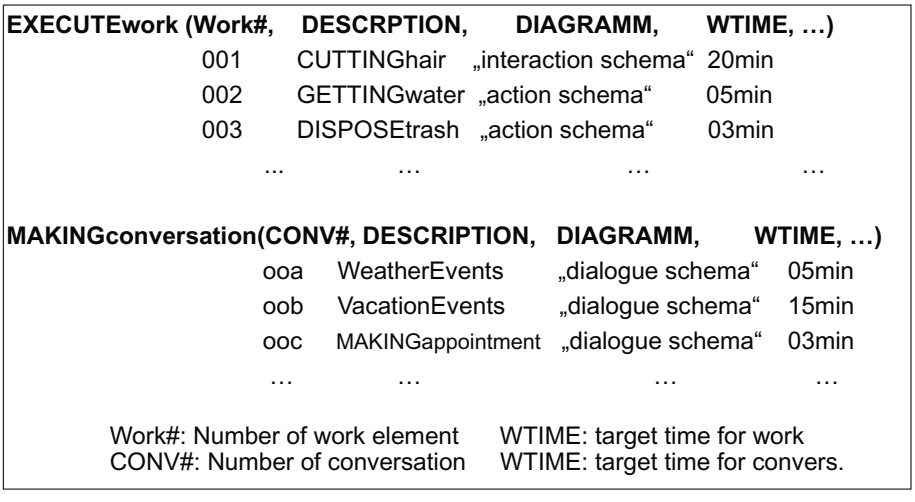


**Fig. 4.** Standardization of functional elements and management using database management systems

The dynamic management of the service delivery, e.g. by preparing the work (specific organizational unit) in an enterprise or administration, could happen as follows due to the relational model on the second linguistic level [Ort05] similar to SQL:

SELECT interaction schema is: CUT_hair
AND dialog schema is:
DISCUSSING_will_Italy_be_world_champion
FROM Service enterprise (or processing-place) is:
Hair salon Dietrich, Constance
WHERE Service provider (hair dresser) is: Fina Origlio
AND Service user (customer) is: Erich Ortner
FOR Date is: Saturday, April 24, 2010, 8:00 am

To entirely establish the task “standardization of functional elements” in an enterprise or an authority, the operational and organizational structure of this

function must be included. It would be useful to establish it together with the function "standardization of data elements" as a department or staff position.

Both solutions, the standardization of data elements as described in the previous section, and the standardization of functional elements described in this section, are therefore "establishments" (consisting of man, organization and technology) for protecting the "content" [Bur10] and not solely the "form" (structure, architecture) of unauthorized access to or by the application system that uses IT. As not to be suspected of censorship of the content, we suggest both solutions for systems for a fair, competitive an humane way to lead our professional dependent life and not as a regulation for creative people or the jointly secured "free" life of the citizens of a country. Today, digital networks are ubiquitous all over the world. We ought to keep in mind the difference between the practice of "need-driven" engineering sciences and their systems and the effect of the "free sciences", (e.g. art or those sciences who deal with the free world citizens who are equipped with universal rights and their way of life) and their respective orientation knowledge [Mit01].

## 6 Outlook

If language-based technologies are to be used efficiently, the future IT users must be assisted at the best from their early youth on [WO04] in their normal language competence. Only IT experts must additionally be trained to deal with programming languages (algorithmics) with regard to developing new applications in an "engineer-like" way (systematically and creatively). Teaching language competence is the responsibility of our general schools with regard to a "world-class education" [Inh08]. IT experts, however, are usually trained at universities and the organization of their studies should be "T-shaped" [Hei10]. Choosing the teaching subjects (e.g. [KL84]) for language education very carefully would be wise in order to prevent producing once again "typographers" (programmers) instead of linguistically competent "authors" (modelers) and enlightened "readers" (users) who are able to develop and use language-based applications in digital and social networks responsibly.

Application development using language-based technologies in digital networks or for end devices today – in times of ubiquitous computing – is to educate users, and actually all of us, from early on to use language in a disciplined, responsible and reflected way. Languages (verbally) include ethics. Here, we ought to work together, so that we can cope, peacefully and with measured prosperity for all, with the huge challenges we are facing, as predicted already about 40 years ago in a report by the Club of Rome (published in 1972), also counting on man's practical reason.

Being "constructive language-critical modelers", often, using constructive logic and rational grammar [Lor87], we can control the language worlds more precisely than the physical world. If we can control the physical world with language artifacts [ZLO10], then we should be able to cope more easily with our real problems on the basis of a constructive logic, ethic and philosophy of science [Lor87] than

on a merely algorithmic and analytic (formal) basis [HB34], [HB39], [BK08]. Or, to say it in Paul Lorenzen's words with the intention of saving the world: "In a constructive re-orientation, all sciences serve a peaceful life and modest prosperity."

## References


[BHT09] Burgemeestre, B., Hulstijn, J., Tan, Y.-H.: Rule-based versus Principle-based Regulatory Compliance. In: Proceeding of the 2009 Conference on Legal Knowledge and Information Systems, pp. 37–46. IOS Press, Amsterdam (2009)

[BK07] Bauer, C., King, G.: Java persistence with Hibernate: Revised edition of Hibernate in action. Manning, Greenwich, Conn. (2007)

[BK08] Baier, C., Katoen, J.-P.: Principles of model checking. MIT Press, Cambridge (2008)

[BMH06] Burke, B., Monson-Haefel, R.: Enterprise JavaBeans 3.0, 5th edn. Auflage. O'Reilly Media, Sebastopol (2006)

[Bur10] Burke, B.E.: Unified Content Security: Securing the Borderless Enterprise, Framigham, MA (2010)

[Car72] Carl, W.: Bertrand Russell: Die Theory of Descriptions. Ihre logische und erkenntnisteoretische Bedeutung. In: Speck, J. (Hrsg.) Grundprobleme der großen Philosophen, pp. 215–163. Vandenhoeck u. Ruprecht, Göttingen (1972)

[Dat07] Date, C.J.: Logic and databases: The roots of relational theory. Trafford Publ., Victoria (2007)

[EHH+08] Engels, G., Hess, A., Humm, B., Juwig, O., Lohmann, M., Richter, J.-P., Voß, M., Willkomm, J.: Quasar enterprise: Anwendungslandschaften serviceorientiert gestalten. dpunkt.Verl., Heidelberg (2008)

[Fis08] Fischer, M.: Über Sinnhaftigkeit und Bedeutsamkeit der Frege'schen Logik im Software-Engineering: Diplomarbeit, Darmstadt (September 8, 2008)

[Fis10] Fischer, M.: Eine Logik zur Abbildung arbeitsteiliger komplexer Prozessschemata. In: Ortner, E. (Hrsg.) Konstruktive Informatik, pp. 23–30. Shaker Verlag, Aachen (2010)

[FLOZ10] Fischer, M., Link, M., Ortner, E., Zeise, N.: Servicebase Management Systems: A Three-Schema-Architecture for Service-Management. In: Fähnrich, K.-P. (Hrsg.) Informatik 2010, Bonn, pp. 730–735. Gesellschaft für Informatik (2010)

[GR02] Gray, J., Reuter, A.: Transaction processing: Concepts and techniques, 9th edn. Auflage. Morgan Kaufmann, San Francisco (2002)

[HB34] Hilbert, D., Bernays, P.: Grundlagen der Mathematik I, Berlin (1934)

[HB39] Hilbert, D., Bernays, P.: Grundlagen der Mathematik II, Berlin (1939)

[Hei10] Heinemann, E.: Jenseits der Programmierung: Mit T-Shaping erfolgreich in die IT-Karriere starten. Hanser, München (2010)

[HL05] Haerder, T., Lehner, W. (Hrsg.) Data Management in a Connected World: Essays Dedicated to Hartmut Wedekind on the Occasion of His 70th Birthday. Springer-Verlag GmbH, Heidelberg (2005)

[HR01] Haerder, T., Rahm, E.: Datenbanksysteme: Konzepte und Techniken der Implementierung, 2nd edn. Auflage. Springer, Berlin (2001)

[Hup06] Huppertz, P.G.: IT-Service - Der Kern des Ganzen. live IT!L, Bad Homburg (2006)

[Inh03] Inhetveen, R.: Logik: Eine dialog-orientierte Einführung. Ed. am Gutenbergplatz, Leipzig (2003)

[Inh08] Inhetveen, R.: Abschied von der Bildung. In: Heinemann, E. (Hrsg.) Anwendungsinformatik, pp. 235–242. Nomos-Verl.-Ges., Baden (2008)

[KH93] Knolmayer, G., Herbst, H.: Business Rules. Wirtschaftsinformatik 35(4), 386–390 (1993)

[KL84] Kamlah, W., Lorenzen, P.: Logical propaedeutic: Pre-school of reasonable discourse. University Press of America, Lanham (1984)

[Kue10] Kuenne, W.: Die philosophische Logik Gottlob Freges: Ein Kommentar. Mit den Texten des Vorworts zu "Grundgesetz der Arithmetik" und der "Logischen Untersuchungen I - IV". Klostermann, Frankfurt am Main (2010)

[Lor87] Lorenzen, P.: Lehrbuch der konstruktiven Wissenschaftstheorie. BI-Wiss.-Verl., Mannheim (1987)

[Mit01] Mittelstraß, J.: Wissen und Grenzen: Philosophische Studien. Suhrkamp, Frankfurt am Main (2001)

[MKS10] Maglio, P.P., Kieliszewski, C.A., Spohrer, J.C.: Handbook of Service Science. Springer Science+Business Media LLC, Boston (2010)

[ORS90] Ortner, E., Rössner, J., Söllner, B.: Entwicklung und Verwaltung standardisierter Datenelemente. Informatik Spektrum 13(1), 17–30 (1990)

[Ort82] Ortner, E.: Aspekte einer Konstruktionssprache für den Datenbank-Entwurf. Dissertation, Technische Universität Darmstadt, Darmstadt (Mai 1982)

[Ort97] Ortner, E.: Methodenneutraler Fachentwurf: Zu den Grundlagen einer anwendungsorientierten Informatik. Teubner, Stuttgart (1997)

[Ort05] Ortner, E.: Sprachbasierte Informatik: Wie man mit Wörtern die Cyber-Welt bewegt. Edition am Gutenbergplatz, Leipzig (2005)

[Pei10] Peirce, B.C.: Type Theory Comes of Age. Communications of the ACM 53(2), 16–17 (2010)

[Pie02] Pierce, B.C.: Types and programming languages. MIT Press, Cambridge (2002)

[RK88] Roehrle, J., Kratzer, K.: Verwaltung von Integritätsbedingungen in einem Datenwörterbuch. Angewandte Informatik 30(1), 18–26 (1988)

[SHBL06] Shadbolt, N., Hall, W., Berners-Lee, T.: The Semantic Web Revisited. IEEE Intelligent Systems 21(3), 96–101 (2006)

[SW85] Steinbauer, D., Wedekind, H.: Integritätsaspekte in Datenbanksystemen. Informatik Spektrum 8(2), 60–68 (1985)

[Wed78] Wedekind, H.: Zur Durchführung des § 6,1 BDSG. Datenschutz und Datensicherheit (4), 181–185 (1978)

[Wed80] Wedekind, H.: Erweiterung einer Entwicklungsmethodologie für Datenbanksysteme um eine Komponente zum Schutz vor Missbrauch von personenbezogenen Daten: Zu einer Methodologie und Teleologie einer Angewandten Informatik. Angewandte Informatik (7), 255–265 (1980)

[Wed81] Wedekind, H.: Datenbanksysteme: Eine konstruktive Einführung in die Datenverarbeitung in Wirtschaft und Verwaltung, 2. Auflage. B.I.-Wiss.-Verl., Mannheim (1981)

[WO80] Wedekind, H., Ortner, E.: Systematisches Konstruieren von Datenbankanwendungen: Zur Methodologie der Angewandten Informatik. Carl Hanser Verlag, München (1980)

[WO04] Wedekind, H., Ortner, E.: Toward universal literacy: From computer science upward. Communications of the ACM 47(6), 101–104 (2004)

[WOI] Wedekind, H., Ortner, E., Inhetveen, R.: Informatik als Grundbildung. 6 Aufsätze: Informatik-Spektrum 27(2) (2004), bis 28(1) (2005)

[ZLO10] Zeise, N., Link, M., Ortner, E.: Controlling of Dynamic Enterprises by Indicators – A Foundational Approach. In: zur Muehlen, M., Su, J. (eds.) BPM 2010 Workshops. LNBIP, vol. 66, pp. 521–530. Springer, Heidelberg (2011)