Chapter 2
Research Background

2.1 Compliance Management and Regulatory Requirements

The body of regulation for financial service providers increases steadily and even faster after the world financial crisis of 2008. Corresponding regulation initiatives aim at closing gaps which the financial crisis has exposed (Brunnermeier et al. 2009). Two major streams of compliance requirements for financial institutes exist. One stream focuses on the proper design and execution of business processes (business process compliance), while the other stream focuses on bank supervision and financial reporting requirements. Both types of regulatory requirements need to be addressed by compliance management activities.

The literature provides different definitions of the term compliance management. Abdullah et al. (2009) define compliance management as "mechanisms to keep enterprise's businesses safe from possible violation of regulatory compliance" (Abdullah et al. 2009, p. 2). A broader definition is provided by El Kharbili et al. (2008b), who perceive compliance management as "the term referring to the definition of means to avoid […] illegal actions by controlling an enterprise's activities. By extension, compliance management also refers to standards, frameworks, and software used to ensure the company's observance of legal texts" (El Kharbili et al. 2008b, p. 2). According to Karagiannis (2008), compliance management comprises three elements: the regulatory approach to ensuring the conformance with regulations and corporate governance, the standardization approach, which ensures conformance with standards, such as provided by ISO, and the corporate standards/best practices approach, which ensures the conformance with best practice frameworks. In this book, the term compliance management is defined as the sum of all organizational and technical activities that support the alignment of business processes and information systems with regulatory requirements.

According to an industry study among compliance experts, Abdullah et al. (2010) identifies seven major compliance management challenges, of which italic marked challenges are particularly addressed in this book:

M. Eggert, *Compliance Management in Financial Industries*,
SpringerBriefs in Information Systems, DOI: 10.1007/978-3-319-03913-8_2,

- Lack of Compliance Culture
- *High Cost*
- Lack of Efficient Risk Management
- Difficulties in Creating Evidence of Compliance
- Lack of Perception of Compliance as Value-add
- *Lack of Understanding of its Relevance to Business*
- *Lack of Communication among Staff*

The lack of compliance culture comprises all cultural impacts of compliance on the organization. Many organizations do not support compliance issues properly; for instance, they do not provide management support or assign junior or "non-star" resources to compliance issues only. The cost issue of fulfilling compliance requirements hinders organizations to implement compliance frameworks. Small and medium sized companies are particularly affected by compliance costs because they do not have the capacity to manage compliance issues. The lack of efficient risk management addresses the resistance to allocate sufficient resources to identify and manage enterprise risks. Difficulties in creating evidence of compliance refer to the organization's inability to demonstrate and prove that the organization is compliant. This challenge especially requires the controlling and recording of compliance-relevant incidents. The identified lack of perception of compliance as a value-add summarizes challenges of establishing a common sense about the value of compliance. Many companies do not perceive compliance as a benefit. Rather, they feel that compliance complicates businesses and that it provides no benefit for the business. This provokes the lack of understanding of compliance relevance to business. Organizations must ensure that relevant regulations for their business are identified and that employees understand these requirements. Finally, the lack of communication among staff refers to the challenge of establishing efficient communication channels within the organization. This lack addresses the problem of communicating regulatory changes and their impact on organizations' activities (Abdullah et al. 2010). This book addresses in particular three challenges:

- *High costs* are addressed by the developed business process compliance checking and report modeling approaches, which support compliance checking tasks.
- *The lack of understanding of compliance relevance to business* and the *difficulties in creating evidence of compliance* are supported by the analysis capability of the approaches developed, which enables searching for regulatory affected business processes and reports. In addition, compliance frauds can be detected.
- The developed modeling technique for regulatory reporting requirements aims at building a common ground between system engineers and legal experts, which addresses the *lack of communication among staff*. The communication aspect will also be addressed by investigating the collaboration of IS and legal experts.

2.2 Regulatory Environment of Banks

Financial institutions are, in comparison to other industry sectors, heavily regulated. Barth et al. (2004) provides five main reasons for restricting bank activities. First, conflicts of interests appear when banks act in business areas, such as securities underwriting or real estate investment, and ill-informed investors need to be advised. Second, “moral hazard encourages riskier behavior” (Barth et al. 2004, p. 209). If a more risk taking behavior is allowed, banks will increase their investment risks to engage in more market activities. Third, the more complex a bank is, the more complex it is to monitor such a bank. Fourth, such large banks may become political and economically powerful that they are too big to fail. Fifth, large financial institutes prevent market competition and reduce market efficiency (Barth et al. 2004). In order to address these five reasons, mainly two regulatory types of requirements exist which are described next: regulatory business process requirements and supervisory reporting requirements.

2.2.1 Regulatory Business Process Requirements

Business processes in financial institutes are regulated through many different laws. They mainly come from three different sources: Legislation and regulatory bodies, for instance, Sarbanes–Oxley Act (*SOX*), Markets in Financial Instruments Directive (MiFID) and Basel III, standards and practical guidelines (e.g., SCOR, ISO9000), as well as business partner contracts (Governatori and Rotolo 2010). Relevant process compliance requirements for banks can be found, for example, in the MaRisk (BaFin 2011), the Securities Trading Act (ger. Wertpapierhandelsgesetz (WpHG)) and the Investment Services Conduct of Business and Organization Regulation (ger. Wertpapierdienstleistungs, Verhaltens- und Organisationsverordnung (WpDVerOV)).

One prominent example for a business process compliance requirement in financial industries is the obligation to handout consulting protocols (§14 WpHG, §14, Section 6 WpDVerOV). According to Section 5 of the Investment Services Conduct of Business and Organization Regulation, a customer must receive adequate information about the financial products offered in a financial consultation process. In addition, according to Section 14, the financial service provider must prepare a consultation protocol and is obligated to hand it over to the customer. Figure 2.1 graphically depicts a process excerpt with the corresponding protocol handout requirement. The bold framed elements comprise the process elements that are necessary for fulfilling the handout requirement. Such a set of process elements may be a part of a structural pattern, which can be applied for detecting compliance frauds in several business process models.

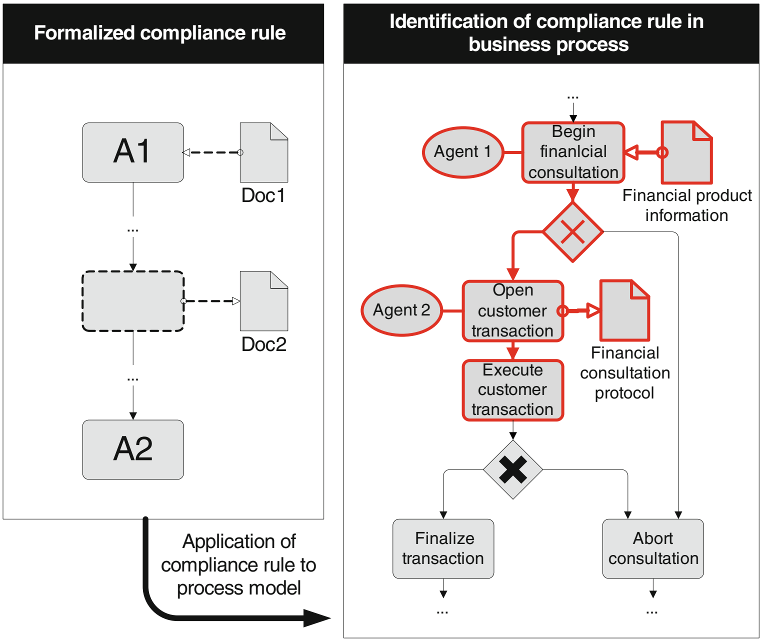

Fig. 2.1 Exemplary business process compliance requirement. Adapted from Becker et al. (2012a)

2.2.2 Supervisory Reporting Requirements

To monitor banks, supervisory agencies require information regarding the full range of activities and risk management procedures (BaFin 2011). In addition, the reporting of adequate capital according to regulations, such as Basel or its transformation into European law through the EU Capital Requirements Directive (CRD) (Bongaerts and Charlier 2009), as well as the reporting of credit information (in particular large exposures) to a central institution is of evident importance (Barron and Staten 2003; Cowan and de Gregorio 2003; Tsai et al. 2011). Besides these banking specific disclosure requirements, several general reporting regulations, such as the International Financial Reporting Standard (IFRS) or the United States Generally Accepted Accounting Principles (US-GAAP), need to be followed. According to Craig and Diga (1996) reporting regulations can be classified in three different dimensions:

Table 2.1 Excerpt of reporting requirements according to the MiFID regulation

Required information to fulfill reporting requirements for the execution of customer orders	Required information to fulfill reporting requirements in financial portfolio management
(a) The name of the company that creates the message (b) The name or other designation of the customer (c) The trading day (d) The trading time (e) Type of order (f) The place of execution (g) Financial instrument (h) Purchase/selling indicator (i) The nature of the order if it is not a purchase or sell order (j) Quantity (k) Unit price (l) Total price (m) Invoiced commission (n) Customer tasks related to the order execution (o) A note, if the contracting party of the investment service company (WpDLU) was the company itself or a person or group to which the WpDLU is a member of, or if it was a customer of the WpDLU	(a) Name of the investment service company (WpDLU) (b) The name and designation of the account (c) The composition and evaluation of the portfolio with details of any financial instrument (d) Total accrued charges and fees, which are splitted by at least total management fees and total costs, and a note that a detailed description is sent on request (e) Comparison of performance with a benchmark (f) Total amount of dividends, interest and other payments during the reporting period (g) Information on other measures of WpDLU, rights regarding owned financial instruments, lend financial instruments and for each executed transaction within the reporting period the Information from (c) to (i) from the left column, in case the customer does not require a separate listing for each transaction

Adapted from Becker et al. (2007a)

- Financial reporting-related legislation, which contains all reporting regulations that can be found in companies' laws, securities laws and tax statutes.
- Official directives and guidelines issued by government agencies, which comprise regulations, such as companies' law administrators, securities market regulators and tax authorities.
- Rules and guidelines, issued by private sector organizations, comprise all regulations that are posted by professional accountancy bodies and stock exchanges.

Regulations such as IFRS and US-GAAP, for example, can be assigned to the category of financial reporting-related legislation. In this book, all three dimensions of regulations are considered, since all regulatory requirements are investigated regarding their relevance for IS design. In the following, the term regulation is used for laws, directives, acts, rules, recommendations, and other principles, which are relevant for financial report design.

The financial industry in Germany has several different supervisory reporting requirements. On its web page the German Federal Bank summarizes all regulatory-driven reports that have to be generated by German banks. Depending on the business environment of a bank, up to 136 reports need to be generated (Bundesbank 2012). In order to exemplarily demonstrate the content of such a report, Becker et al. (2007a) summarizes a part of the report requirements for the MiFID, which was implemented into German law in 2007. Parts of such reporting obligations, which can be seen as data requirements for a data warehouse, are summarized in Table 2.1.

Disclosure regulations and their electronic transformation are strongly related to each other. Nowadays, the electronic exchange of regulatory required reports is prominently supported by using the data exchange standard XBRL (eXtendable Business Reporting Language). XBRL is based on the eXtendable Markup Language (XML) and enables the automated production and consumption of financial reporting and performance information (Bergeron 2004). Corporate information can be incorporated directly into the data warehouse systems of information consumers, such as the Securities and Exchange Commission (SEC) or other stakeholders (Debreceny et al. 2010). Since its initial usage within the Australian Prudential Regulatory Authority in 2002, XBRL has been adopted within several other regulatory directives and guidelines (Kernan 2008; Williams et al. 2006). Authorities like the SEC or European supervisory authorities describe financial reporting obligations by using the data exchange standard XBRL (Bergeron 2004; Debreceny and Gray 2001). Moreover, XBRL becomes the central data exchange standard in the area of credit monitoring and reporting corporate performance (Debreceny 2007). With its standardization potential, XBRL makes an important contribution to a non-ambiguous and well-defined data transfer between firms and supervisory authorities.

2.3 Business Process Modeling

Business processes are the central point of process-oriented corporate design. While an organizational structure divides an organization into task-oriented units, such as divisions and departments, business processes deal with the execution of these tasks (Becker and Kahn 2003). Davenport defines a process as “structured sets of work activity that lead to specified business outcomes for customers” (Davenport and Beers 1995, p. 57). A business process is directed by business objects and is influenced by a company’s environment (Becker and Kahn 2003).

Nowadays, companies try to conceptualize their business processes in order to reduce complexity, increase understanding, and uncover optimization potential. Therefore, conceptual business process modeling techniques are applied (Aguilar-Savén 2004; Bandara et al. 2005; Rosemann 2003). “Conceptual modelling is the activity of formally describing some aspects of the physical and social world around us for purposes of understanding and communication” (Mylopoulos 1992, p. 50). Conceptual models play a significant role in IS development as they allow for an early detection and correction of IS design mistakes (Wand and Weber 2002). Business process models have received even more attention since the business process reengineering trend in the early 1990s came up (Davenport and Stoddard 1994; Hammer and Champy 1993). Several business process modeling techniques have been developed. According to an industry study of Becker et al. (2010a), the most relevant modeling techniques in financial industries are the Business Process Modeling Notation (BPMN) (Object Management Group 2006), Event-driven Process Chains (EPC) (Keller et al. 1992), and the activity diagrams from the Unified Modeling Language (UML) (Object Management Group 2005).

All these modeling techniques have in common that they are not domain-specific and that they can be applied for several purposes. According to Pfeiffer (2008), these properties come along with a couple of deficiencies: semantic inequality conflicts, such as homonym conflicts, semantic equality conflicts, such as synonym conflicts, and order conflicts. These identified problems of classical modeling techniques lead to the development of building block-based modeling techniques, which are semantically enriched by a domain ontology and standardized through the usage of predefined building blocks. Process building blocks are reoccurring activities in a certain business domain (Baacke et al. 2010; Becker et al. 2007b; Lang and Bodendorf 1997; Pfeiffer 2008). One example for a building block-based modeling technique is the PICTURE notation, which was developed for modeling business processes in the public administration (Becker et al. 2007b). Adapting the idea of building blocks for the financial industry, a domain-specific process modeling technique for banks has been developed (Becker et al. 2010b; Weiß 2011).

2.4 Business Process Compliance Analysis

Several compliance-checking approaches have been developed in order to ensure that business processes follow regulatory requirements. The major goal of this research stream is either to ensure that a business process fulfills all regulatory and business requirements or to detect compliance frauds after the execution of a business process. El Kharbili et al. (2008a) separates approaches for the analysis of business process compliance in two time-dependent classes: forward compliance checking and backward compliance checking. Forward compliance checking approaches aim to detect compliance frauds and analyze business processes with respect to the fulfillment of compliance rules before (design-time compliance) or while they are executed (run-time compliance). Hence, the processes are checked during the design-time or execution- time. Backward compliance checking approaches check whether a compliance fraud has appeared in the past. Thus, backward compliance checking approaches allow for a retrospective analysis of concrete process instances (El Kharbili et al. 2008a). This book solely focuses on design-time compliance. In the remainder of this section, the requirements for compliance checking approaches and common forward, run-time, and backward compliance checking approaches are discussed.

2.4.1 Requirements for Compliance Checking Approaches

The business process compliance checking approach, developed in this book, is guided by the fundamental requirements for supporting semantic constraints, such as business rules or policies, in process management systems (Ly et al. 2012):

- Formal language for constraint specification: A modeling language for expressing compliance requirements and process constraints that allows for a formal analysis is required.
- Constraint organization: A way to organize semantic constraints, such as compliance rules, is necessary. The use of repositories (Ly et al. 2006) or directories for such constraints (Sadiq et al. 2007) is suggested (Ly et al. 2012).
- Views on semantic constraints at different abstraction levels: Compliance rules must be represented differently with respect to their viewer, i.e., business and legal experts need a different view on compliance rules than tool and implementation experts.
- Support of lifetime compliance: The compliance of business processes must be analyzable during design-time and run-time. In addition, a process management system must allow for validating process instance changes and process model adaptations.
- Support process-spanning scenarios: The compliance management system must allow for validating multiple processes and process instances at once.
- Providing intelligible feedback: The user of a compliance management system should get comprehensive feedback, i.e., providing an error diagnosis, advices for conflict resolutions, and compensation strategies.
- Support of flexible constraint handling: The approach must allow for classifying compliance requirements according to enforcement levels and enforcement strategies, such as "only a bank branch manager may skip the personal customer identification". Hence, it must be possible to establish a relation between organizational units and compliance rules.
- Support of traceability: The results of each compliance check must be documented properly in order to enable the reconstruction of the past compliance checks and its results. In particular, compliance audits require this feature.

2.4.2 Design-Time Compliance Checking

Design-time compliance checking approaches belong to the group of forward compliance checking approaches and work on process models that are not yet deployed and thus can be changed without consequences for running process instances. The approach of Foerster et al. (2007) allows for checking quality constraints on business processes automatically. The authors specify simple control-flow patterns and formalize them using Linear Temporal Logic (LTL) (for LTL see Clarke et al. 2000). In order to execute a pattern search, the approach makes use of the NuSMV model checker (Cimatti et al. 2002). The result is binary, expressing whether the process model is compliant or not.

Another research stream focuses on the development of modeling techniques for compliance rules as one element for the model checking approach. In many cases these approaches use or extend the Formal Contract Language (FCL) (Governatori and Milosevic 2006; Governatori et al. 2006; Governatori and Rotolo 2010). The approaches detect missing or prohibited activities within a process model. The Process Compliance Language (PCL), which extends FCL, combines defeasible logic and deontic logic and allows for formalizing exceptions, violations and obligations (Governatori and Rotolo 2010). Based on these findings, Lu et al. (2008a; 2008b) developed an algorithm that enables the quantification of the effort that is needed to transform a non-compliant process model into a compliant one. Sadiq et al. (2007) integrate control objectives in business processes using FCL. The approach classifies the control objectives among four types: flow, data, resource, and time tags (Sadiq et al. 2007). Mueller (2010) focuses on the structural analysis of BPMN process models by conceptualizing compliance requirements with the Process Pattern Modeling Language (PPML) and the Process Constraint Modeling Language (PCML). The modeling techniques facilitate the definition of structural model patterns and constraints, which are transformed into LTL in order to enable the formal analysis of BPMN models.

Woerzberger et al. (2008a, b) develop the visual Business Process Compliance Language (BPCL), which enables the definition of a limited set of control flow constraints on simplified syntax Business Process Execution Language (SimBPEL) process models, which are restricted to Web Service Business Process Execution Language (WS-BPEL) models (Alves et al. 2007). Accorsi et al. (2011) present an approach to verify data security requirements in cloud-based workflows, which is based on Petri-nets. They argue for a better comprehensibility of Petri-nets for business experts compared to other formal languages, such as LTL or FCL.

Ghose and Koliadis (2007) develop an approach that can be applied for generating process models and for selecting the most similar one to a given (compliant) process model. The approach aims to support the selection of process models that resolves compliance violations. Schumm et al. (2010) also supports the design of compliant business processes. Their approach uses pre-defined compliance fragments in order to support process modelers in creating compliant process models. For supporting root cause analyses, Elgammal et al. (2010) present a set of predefined composed compliance patterns. Combined with an analysis approach based on LTL, these patterns are used to guide process modelers in resolving compliance violations in business processes.

The Process Entailment from the Elicitation of Obligations and Permissions (PENELOPE) approach (Goedertier and Vanthienen 2006) provides the foundation for an automatic generation of compliant BPMN models using a Prolog algorithm. The approach allows for defining temporal restrictions as well as activity operators. Kuester et al. (2007) developed an approach that generates compliant business process models considering object life cycles, which contain all possible states of a data object, such as generated, granted, settled, and rejected for a credit application.

2.4.3 Run-Time Compliance Checking Approaches

Run-time compliance checking approaches differ from design-time compliance checking in their dependency on the business process execution environment. They require run-time information while executing a processes instance (Rossak et al. 2006). Liu et al. (2007) developed a modeling technique to represent compliance requirements, the so-called Business Property Specification Language (BPSL). The language affords a checking of compliance rules based on a Finite State Machine (FSM). For transforming models developed using the Business Process Execution Language (BPEL) into FSMs and checking them, the LTL and the Pi calculus (Milner 1999) was applied. Milosevic (2005) integrates policy definitions into business processes and enable the monitoring of run-time compliance.

One research stream of run-time compliance checking focuses on the analysis of business contract constraints in business processes (e.g., Alberti et al. 2007; Milosevic et al. 2002; Weigand and van den Heuvel 2002). Alberti et al. (2007) conceptualize contract constraints using three event notions: happened, expected, and not-expected. At run-time the approach records all events and checks whether or not contract conditions have been violated. Milosevic et al. (2002) present control mechanisms for a role-based contract management architecture and provide an assessment approach based on subjective logic. Weigand and van den Heuvel (2002) link the specification of business object-based workflow systems with the formal specification of business contract constraints, using the XML-based business contract specification language.

Ly et al. (2008; 2006) develop an approach for checking the semantic correctness in process instances at run-time. Limitations are discussed in Ly et al. (2012). Based on these limitations, they present a compliance-checking framework that enables visual compliance rule modeling and a subsequent automatic formalization using first-order predicate logic. The authors extend their approach to support the compliance of the whole lifecycle of business processes, which comprises compliance verification during design-time and run-time as well as for process changes and process evolutions (Ly et al. 2012).

2.4.4 Backward Compliance Checking

Approaches for backward compliance check whether a business process has been executed in accordance with all regulatory requirements and business constraints. To check the process instances, process execution logs are analyzed by using mining techniques (El Kharbili et al. 2008b). Van der Aalst et al. (2005) use an LTL checker to verify process logs regarding expected and unexpected behavior. Rozinat and van der Aalst (2008) extend this approach and check whether a process log complies with its process model. Chesani et al. (2007) develop an algorithm to transform process models of the health care industry into a formal language based computational logic and verify its conformance with a given process execution, derived from the event log. Ramezani et al. (2012) apply a Petri-net based approach to formalize 55 control flow oriented compliance constraints and classify them among 15 compliance rule categories.

Summing it up, compliance-checking approaches support the checking of business processes and provide a feedback whether the execution of a business process is in line with regulatory requirements. When the compliance of supervisory reports needs to be ensured, such approaches have a limited applicability. Supervisory reports are data-driven and receive its content mostly from data warehouses, whose development foundations are elaborated in the next section.

2.5 Data Warehouse Development

"A data warehouse is an integrated and timevarying collection of data derived from operational data and primarily used in strategic decision making by means of online analytical processing (OLAP) techniques" (Hüsemann et al. 2000, p. 1). It connects Online Transaction Processing (OLTP) systems with OLAP components (Chaudhuri and Umeshwar 1997).[1] The latter allow for a fast interactive navigation through the so-called multidimensional data space and supports information searches, mainly performed by managers (Colliat 1996). The tasks of extracting (E), transforming (T) and loading (L) data from the transactional systems into the data warehouse is called ETL process (Inmon 1996).

2.5.1 Data Warehouse Concepts

In order to conceptually design data warehouses, Holten (2003) specifies master data of management views. This master data contains concepts that are frequently used in data warehouse projects. Figure 2.2 summarizes common concepts (in the following, concepts are written in italics) and presents their representation in four different modeling techniques for conceptual data warehouse design, namely Multidimensional Entity Relationship Modeling (ME/RM) (Sapia et al. 1998), Application Design for Processing Technologies (ADAPT) (Bulos 1996), the Dimensional Fact Model (DFM) (Golfarelli et al. 1998) and H2 for Reporting (H2fR) (Becker et al. 2007d). Hettler et al. (2003) and Knackstedt et al. (2005) compare these modeling techniques and discuss their similarities and differences.

[1] For an overview of data warehouse and OLAP technologies see Chaudhuri and Umeshwar (1997).

Concept	ME/RM	ADAPT	DFM	H2fR
Dimension	-	Hierarchy	-	Dimension
Dimension Object	-	Dimension Member	-	Instance object
Hierarchy Level	Dimension Level Entity Type	Level	Dimension Attribute	Selection Object
Dimension Scope	-	Scope	-	Dimension Scope
Ratio	Attribut	Dimension Member	Fact Attribute (Measure)	Ratio
Ratio System	-	Measure Dimension	-	Ratio system
Information Object	Fact Relation-ship Type	Hypercube	Fact	Cube

Fig. 2.2 Comparison of data warehouse modeling concepts. Adapted from Knackstedt et al. (2005)

The first data warehouse concept is the *dimension*, which spans a multidimensional space for management views. Each dimension contains *dimension objects*, which represent concrete objects of interest, such as a concrete city or a concrete financial product. Dimension objects are organized in hierarchies, whereas each dimension object is uniquely assigned to one *hierarchy level*. In order to limit the navigation space of a dimension, the concept of *dimension scopes,* which are sub trees of a dimension, has been introduced. *Ratios* are hierarchically organized in *ratio systems* according to their functional or algebraic relationship. To define the (multidimensional) information space, consisting of one or more dimensions and ratios, the concept of *information objects* has been introduced (Holten 2003).

2.5.2 Data Warehouse Development Process

Traditional (transactional) database development processes usually comprise four phases: requirements analysis and specification, conceptual design, logical design, and physical design (Batini et al. 1992; Vossen 2008). These four phases are also identified as a process model for data warehouse design (Hüsemann et al. 2000, p. 1). The phases, their input and outputs as well as the phase results are depicted in Fig. 2.3. The *requirements analysis and specification phase* gets the operational databases and its schemas as input. The main objective of this phase is eliciting the user and business requirements for the data warehouse. A first impression of dimensions, attributes, and ratios is expressed and linked to database attributes of the operational databases.

The resulting semiformal business concept acts as input for the *conceptual design phase*, in which multidimensional data warehouse schemas are developed. The conceptual design phase aims to develop graphical multidimensional

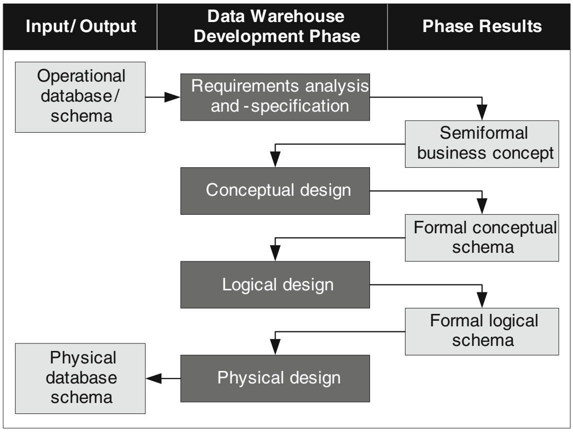

Fig. 2.3 Process model for data warehouse design. Adapted from Hüsemann et al. (2000)

diagrams, containing dimensions, hierarchies, and ratios. Therefore, conceptual modeling techniques help to conceptualize the data warehouse schema. According to Rizzi et al. (2006), conceptual modeling techniques can be classified into three categories: extensions to the entity relationship model (ERM) (e.g., Goeken and Knackstedt 2009a; Sapia et al. 1998; Tryfona et al. 1999), extensions to UML (e.g., Abelló et al. 2006; Harren and Herden 1999; Priebe and Pernul 2001), and ad hoc models (e.g., Bulos 1996; Golfarelli et al. 1998).

In order to improve and accelerate the data warehouse design process, conceptual reference models for data warehousing have been introduced (e.g., Becker and Knackstedt 2003; Becker and Knackstedt 2004; Goeken 2004; Goeken and Knackstedt 2007). Reference models are information models whose content can be reused for several application scenarios (Becker and Knackstedt 2004; Fettke and Loos 2007a; Thomas 2007; vom Brocke 2002). Regarding the conceptual data warehouse development phase, reference models can be used to predefine dimensions, hierarchies, and ratios for a certain group of data warehouse users. The configuration and adaptation of reference models allow for specifying parameters to develop configurable reference models, which are reference models that can be adapted to a company-specific environment (Knackstedt and Klose 2005).

The developed conceptual model of a data warehouse is used as an input for the *logical design phase*. Based on the conceptual models, platform dependent models of the data warehouse are developed that consider the logical structure of the data warehouse (mostly relational). The logical model is constrained, for example, by required response time or disc space, and is tailored for the implementation on the target system (Rizzi et al. 2006). Target database systems are typically relational or multidimensional. For relational database systems, the star or snowflake schemas, which differ in their way of normalization, are applied (Vassiliadis and Sellis 1999). A star schema's dimension tables are denormalized, while a snowflake schema's dimension tables are normalized according to the different hierarchy levels. Multidimensional databases are implemented by using data cube storage procedures, such as condensed cubes, dwarfs, and QC-trees (Rizzi et al. 2006). The result of the logical design phase is a formal logical schema, which simultaneously is the input for the physical design stage.

The data warehouse design process ends with an implementation of a physical database schema, which is the result of the *physical design phase*. The physical database schema contains implementation specific and mostly performance relevant adjustments, such as partition and index considerations. In addition, OLAP-specific justifications, such as pre-aggregation of data and justifications for parallel processing are performed (Bellatreche and Mohania 2009; Rizzi et al. 2006).

The book at hand focuses on the requirements analysis and -specification phase as well as the conceptual design phase. The investigation of the collaboration of IS and legal experts addresses the requirements elicitation and stakeholder collaboration in IS projects. The development and evaluation of a modeling technique for conceptual regulatory reporting requirements addresses the conceptual data warehouse design phase.

2.6 Multisensory Law and Legal Visualization

Multisensory law is a new legal discipline and focuses on the investigation of sensory phenomena of the law, such as visual, audiovisual, or tactile-kinesthetic (Brunschwig 2012). "Multisensory law mainly deals with the law as a uni-and multisensory phenomenon within and outside the legal context" (Brunschwig 2012, p. 714). The major goal of multisensory law is to improve the communication and mediation of legal requirements. One area, which commonly uses legal visualization, is the traffic law. The famous stop sign, for example, signals a car driver that he has to stop at the end of the street in order to look whether the crossing street is free of traffic.

Legal visualization is a sub area of multisensory law and comprises the visual representation of legal requirements. Its history reaches back to the year 1300, where the Saxon Mirror (ger. "Sachsenspiegel") was introduced to illustrate the law in pictures, sequences of images, and texts (Oppitz 1990). Today, graphic designers, psychologists, lawyers, and experts from related disciplines again try to visualize the law by using models or other graphical representations (Sachs-Hombach 2005). Boehme-Neßler (2005), for example, discusses the opportunities and threats of legal visualization and argues for the potential of visualization approaches for the understanding of legal requirements.

Nowadays, legal visualization approaches focus on a more structured representation of legal requirements. The usage of mind mapping for the representation of legal information on E-government websites, for example, is investigated by Brunschwig (2006). As a mediation device between lawyers and client‘s, McCloskey (1998) introduce the concept of legal map making, which comprises two approaches. The first one is called organizing metaphors and comprises images, such as a bridge or a scale to visualizing legal requirements. The second approach is the template approach, which comprises the drawing of cases and the extraction of essential legal elements, which are reused to develop a logical model to explain the case (McCloskey 1998).

A second research stream focuses on the visualization of contracts (Becker et al. 2012f; Berger-Walliser et al. 2011; Kabilan 2005). Becker et al. (2012f) argue for the usage of conceptual process models for the representation of procedural parts of a contract. Berger-Walliser et al. (2011) suggest to use contractual literacy and visualization in order to help establishing a cross-professional communication among business and legal experts. Kabilan (2005) discusses the transformation of contract obligations to BPMN models and argues for the need to semantically integrate contract terms and conditions into business process models.

A third research stream, which is more process model related, focuses on the integration of legal requirements and control objectives in modeling techniques (Alpar and Olbrich 2005; Carnaghan 2006; Olbrich and Simon 2008; Sadiq et al. 2007). Alpar and Olbrich (2005) extend the event-driven process chains by legal aspects to explicating the regulatory requirements for a certain event or activity. They argue for the relevance of including the legal framework in process models and demonstrate the modeling technique with a process example of the German Federal Insurance Institute for Salaried Employees. Olbrich and Simon (2008) discuss the formal modeling of regulations for regulated processes in public administrations. The developed approach allows for deriving process structures, which are implicitly described in the legal paragraphs. Carnaghan (2006) investigated business process modeling approaches for its usefulness for audit risk assessments. The modeling constructs of different business process modeling techniques are compared to those identified as relevant for process level audit risk assessments (Carnaghan 2006). Sadiq et al. (2007) argue for the need to provide a systematic approach that helps understanding business and control objectives of business processes. They visualize four types of control tags: flow (control flow constraints), data (data retention and lineage requirements), resource (access, role, and authorization management), and time tags (deadlines and maximum durations) and annotate them to process model elements (Sadiq et al. 2007).

This book addresses multisensory law and legal visualization in two ways. First, the developed modeling technique for regulatory reporting requirements is a contribution to the field of legal visualization since the law is represented in a structured graphical way. Second, the visual representation of the business process compliance patterns is perceived as a transfer of legal requirements into a structural pattern, which can also be regarded as a legal visualization.

2.7 Regulatory Requirements Engineering

In order to conceptualize regulatory requirements for the sake of developing compliant IS, two major challenges exist. First, compliance experts and IS developers must determine the set of regulatory requirements that are relevant for business processes and IT systems. Second, tasks that result from the regulatory requirements must be determined in order to comply with the regulations (Kerrigan and Law 2003). Thus, an organization must first identify all relevant regulations before it begins to check whether or not an IT system is compliant with regulatory requirements (Otto and Antón 2007). But even if the relevant requirements are identified, it is still challenging to extract requirements for system design because the legal terms and expressions are difficult to understand for IS designers and non-legal experts (Toval et al. 2002).

Systems for supporting legal requirements in requirements engineering are faced with several challenges. Otto and Antón (2007) identify nine elements for systems that support the regulatory-driven analysis and requirement specification. They guide the development of the business process and reporting compliance artifacts in this book. According to Otto and Antón (2007), such systems must allow for:

- identifying relevant regulations
- classification of regulations with meta data
- prioritizing of regulations and exceptions
- managing evolving regulations
- tracing between references and requirements
- ensuring consistency by using data dictionaries and glossaries
- navigating and searching semi-automatically
- annotating regulatory statements
- computing queries for comparing legal concepts and compliance.

According to Massey et al. (2012), research in legal requirements engineering focuses on two streams: techniques that derive legal requirements for software (e.g., Barth et al. 2006; Massacci et al. 2005) and techniques that ensure the compliance of software requirements with regulations (e.g., Breaux and Antón 2008; Breaux et al. 2006; Otto and Antón 2007). This book addresses both streams. On the one hand, a modeling technique to represent legal supervisory requirements is presented, which helps ensuring the compliance of reporting systems (Becker et al. 2011a; Becker et al. 2012a; Becker et al. 2012e). On the other hand, an approach to check the compliance of business processes and conceptual data warehouse models automatically is presented and evaluated (Becker et al. 2012b,d; Eggert et al. 2013a).

This section provides the foundations to develop new and extended artifacts in order to manage the compliance of information systems and business processes. Basic regulatory requirements in the financial service industry and their impact on business processes and the data warehouse are introduced. Related work in the area of business process modeling and process compliance analysis builds the foundation for the development of a new compliance analysis approach for business processes. The data warehouse development process is described in order to explicate the necessity of conceptual data warehouse modeling. Finally, legal visualization and regulatory requirements engineering works are elaborated in order to provide the foundations for the development of a modeling technique for regulatory requirements.

http://www.springer.com/978-3-319-03912-1

Compliance Management in Financial Industries
A Model-based Business Process and Reporting Perspective
Eggert, M.
2014, XII, 96 p. 25 illus., Softcover
ISBN: 978-3-319-03912-1