

Configuration Approach for Analytical Service Models – Development and Evaluation

Christian Hrach
Information Systems Institute
Leipzig University
Leipzig, Germany
hrach@wifa.uni-leipzig.de

Rainer Alt
Information Systems Institute
Leipzig University
Leipzig, Germany
rainer.alt@uni-leipzig.de

Abstract—In data-driven operational processes, analytical applications are increasingly being used to support business users in processing their individual tasks. During development of analytical applications, the extensive content and design coordination between heterogeneous user groups on the one hand and the technical software developers on the other hand is often accompanied by ambiguous and fragmented documentation of user requirements. To support clear and comprehensive requirements documentation, this work describes the development and evaluation of a modeling approach for the user-sided conceptual configuration of planned analytical applications utilizing integrated analytical services and their coupling with process models. This approach addresses the specification of individual analytical use cases as well as of the overall analytical application and considers the desired scope of analytical self-service functions.

Keywords—analytical service, service configuration, requirements specification, analytical process

I. INTRODUCTION

In the course of the spread of data-driven processes, the use of analytical software applications within these processes has become very important in many domains. In research, existing approaches such as "Process-Oriented Business Intelligence (BI)" and "Operational BI" (e.g. [1]) already postulate the close integration of process design and the use of analytical software for direct operational process support including analytical triggered automatic actions (e.g. alerts) [2]. In this connection, the assignment of the supporting analytical applications and the provided analytical information is of special importance in context of (re-)designing processes [3] in order to determine and plan the mutual causal / temporal relationships and dependencies of individual analytical information / applications, and to show the contribution of the analysis chain within the process flow. For example, in order to design an urban park monitoring process, it is necessary to specify the information required by the operational process workers for specific activities within the process (e.g. providing an overview of plant growth in the "review of the vegetation" activity step) in a corresponding process model. Unfortunately, standard process modeling notation (e.g. Business Process Model and Notation (BPMN)) and supplementary approaches (e.g. [4]) insufficiently afford to visualize and specify conceptual and business user-sided characteristics of activity supporting analytical information objects. In the example process scenario of public park monitoring, process actors (like botanist and public administration staff) who review the associated process model don't just need the identifier of a specific information object (e.g. a dashboard named "Green Plants Cover and Rainfall"), but they need more insight (e.g. regarding the form of dashboard presentation, calculation of performance

indicators, data sources, information update cycles, ...) to specify their information requirements and to assess the quality and adequacy of information objects for their operational tasks (e.g. to review vegetation status).

However, ready-made analytical applications and the provided analytical information objects often do not meet the expectations of the analytical business users. Besides the above mentioned missing consideration of analytics in operating process models [5], other related reasons are an insufficient involvement of future users in the requirements analysis [5] and an insufficient requirements documentation [6]. The deeper root cause for this include amongst others an insufficient communication basis between software developers on the one hand and the often analytically not deeply trained casual users [7] on the other hand, because both sides speak different "languages" based on their respective level of knowledge [8]. In addition, the use of analytical applications is increasingly located in cross-domain contexts [9] (e.g. due to the spread of Internet of Things (IoT) technologies and a spread of sensors and corresponding sensor data), where actors from different industries as well as with different technical backgrounds and mindsets working together both in the user community (e.g. botanists and public administration staff) and on the developer side (e.g. experts of data visualization and satellite data analysis) to elaborate new analytical use cases (e.g. monitoring of plant growth in urban parks). At the same time, analysis cases increasingly base on IoT-data [10] with heterogeneous content structure (e.g. stationary rainfall data time series vs. optical spectral satellite data) and from different organizational sources, whereat the existence, origin and structure of these data are often initially unknown to people from other domains or industries. Both heterogeneous users as well as cross-domain data impede the development of a common univocal and fine-granular conceptual understanding of a planned analytical solution meeting the requirements of all participating analytical users.

Analytical information requirements and related models are different from operational requirements, because they focus on information needs of business users, and not on the efficient support of transactional operations [11]. Besides the limited ability of standard process modeling notations, there are other approaches to realize the conceptual specification of analytical applications. In general, text-based modeling approaches are less appropriate than approaches providing graphical models, because the use of plain texts holds an increased risk of an ambiguous and incomplete requirement specification [12]. Existing scientific approaches for the documentation of functional and non-functional analytical requirements from conceptual perspective (cf. chapter II) are not sufficient for a modularized and comprehensive configuration of analytical applications: They are either too generic and don't fit to the special content of analytical

The authors received funding by the Development Bank of Saxony (SAB) and the European Social Fund (ESF) within the project S2DES.

applications (e.g. [13]) or refer just to single conceptual requirement aspects (e.g. [14]). In addition, these approaches don't consider the possibly presence of personal data and their significant relevance for analytical application design [15], because a late recognized ban on use of personal data can lead to major changes in the subsequent analytical application design and implementation. For example, geo-located temperature data recorded with the mobile phone of a botanist may not be used without adjustments in a urban park monitoring process due to data privacy restrictions, whereat components for anonymization should be considered right from the start during system implementation.

In recent years, designing service-oriented analytical applications (e.g. [16]) has gained some importance in research and practice to create flexible and composite applications [17] allowing a less costly preparation and subsequent redesigns (due to moving requirements) of these applications over time. Therefore, software vendor build analytical applications increasingly based on a service-oriented technical architecture (e.g. MicroStrategy, Oracle Business Intelligence Suite) and offer their products as services on service marketplaces (e.g. Amazon Web Services). Although there is a need to create modular service-oriented analytical solutions in practice [18], there are only first coarse-grained description models for analytical services from a user's perspective in research (e.g. [19, 20]) and in practice (cf. [21]).

The configuration space for analytical services and service features presented in this paper is based on design requirements derived from usage contexts and functional characteristics of analytical applications, from data privacy regulations and their implications for analytical application design, from the enablers for cross-domain analytics ([9], e.g. semantic description of heterogeneous data) and from self-service analytics as a trend to empower business users to change design and content of analytical applications on their own [7] (cf. chapter III and IV.A). The configuration of individual analytical service models with direct participation of business users generates user-oriented requirement artifacts, which are in general very well suited to coordinate and clarify requirements between different heterogeneous actors [22]. Conceptual analytical services provide a detailed draft for the subsequent technical system implementation and bridge to the area of technical software design modeling (cf. [23]) (e.g. concerning update rates of indicators, graphical

diagram designs or escalation scenarios in the course of threshold value exceedance (alerts)).

In summary, the following main research question (RQ) arise:

RQ: How must analytical services and their relationships be designed in order to represent the conceptual requirements for analytical process support?

For a detailed definition, this elaboration distinguishes three sub-questions (SQ):

SQ1: Which design requirements for the modeling of analytical services can be derived from analytical process support examined in practice scenarios, functional and non-functional characteristics of analytical applications, data privacy regulations, self-service analytics and from the enablers of cross-domain analytics?

SQ2: Which services and service features are included in the modeling approach and how are they related to each other?

SQ3: How can analytical services enrich process models?

First, chapter II shows existing modeling approaches with their deficits and chapter III presents the research design. Chapter IV elaborates design requirements regarding the configuration approach as well as the deduced analytical services assigned to different configuration areas, the inter-service relationships, the detailed analytical service models and their integration into the graphic representation of process models. After presenting an application example and the results of the evaluation in chapter V, chapter VI concludes this work.

II. RELATED WORK

Stroh et al. 2011 [11] published a literature review about methods supporting requirements analysis for analytical information systems, and six of the identified methods in this survey [14, 24–28] at least partially covered requirements documentation. To add recent approaches, this research rerun the literature review in an updated version of Stroh et al. 2011 [11] with a keyword-based search in all journals between 2010–2020 which were rated with “A” or “A+” by the scientific commission for Business and Information Systems Engineering (VHB-JOURQUAL 3), and additionally in the Journal of Requirements Engineering. As keywords were used “information requirement” and an additional term related to

TABLE I. EVALUATION OF THE EXAMINED APPROACHES

	Models utilized by business users	Provision of configuration alternatives	Graphical modeling notation	Service-oriented design	Process-relation	Modeling content for analytical requirements							
						Data / information	Periodicity	Presentation	Users	Analysis methods	Automatic actions	Self-services	Data privacy
Shanks / Darke 1999 [27]	-	-	x	-	-	x	-	-	x	-	-	-	-
Bonifati et al. 2001 [24]	x	-	(x)	-	-	x	-	-	x	-	-	-	-
Strauch 2002 [28]	(x)	-	x	-	-	x	(x)	(x)	x	(x)	-	-	(x)
Goeken 2004 [14]	-	-	-	-	-	x	(x)	-	x	-	-	-	-
Calvanese et al. 2006 [25]	-	-	x	-	-	x	-	-	-	-	-	-	-
Giorgini et al. 2008 [26]	-	-	x	-	-	x	-	-	x	-	-	-	-
Maté / Trujillo 2012 [34]	x	-	x	-	-	x	-	-	-	-	-	-	-
Mayer et al. 2012 [35]	x	x	-	-	-	-	-	(x)	x	(x)	-	-	-
Horkoff et al. 2014 [33]	x	-	x	-	-	(x)	-	-	-	-	-	-	-
Jovanovic et al. 2014 [32]	(x)	-	(x)	-	-	x	-	-	-	-	-	-	-
Rosenkranz et al. 2016 [31]	(x)	-	(x)	-	-	x	-	-	-	-	-	-	-
Ferrández et al. 2016 [30]	-	-	x	-	-	x	-	-	-	-	-	-	-
Teruel et al. 2019 [29]	x	-	x	(x)	(x)	(x)	-	-	x	-	-	-	-

analytical applications (“management IS”, “decision support system”, “executive information system”, “data warehouse”, “data warehousing”, “business intelligence”, “OLAP”, “analytical service”, “analytical application”, “analytical software”, “analytical IS”, “big data”) with alternative ditions in title and abstract. This new literature search generated 29 papers, and after a closer content check regarding the focus of this research (requirements specifications / modeling of analytical applications), this number was reduced to seven papers [29–35].

The overall 13 approaches were examined according to whether they exhaustive (“x”), partially (“(x)”) or not (“-“) consider content-related requirement aspects addressed in this research (Table 1): main characteristic content areas of analytical applications (including *periodicity* and *presentation of analytical information* generated with *analysis methods* for special users [36]), automatic actions as significant features in Operational BI [2], analytical self-service functions [7] and data privacy constraints [15]. Furthermore, additional design characteristics (model usage by analytical users [22], graphical modeling [12] and provision of configuration alternatives [35] to foster self-service; process-relation to link to usage context [1]; service-orientation for flexible application design [17]) were examined.

These predominantly model-based approaches (Tab. 1) often just address single requirement aspects with a focus on analytical data / information requirements (especially as a prerequisite for data modeling (e.g. [25, 32]) and the specification of different user groups with their business objectives (e.g. [24, 26]). The lack of requirements specification approaches with reference to automatic actions and analytical self-services, as well as the very poorly considered aspects of process-relation, data privacy and service-oriented design show a distinct need for research in these fields regarding requirements specification. Only the approach of situational management support systems [35] contains configuration alternatives, and it addresses some important content-related design elements for analytical applications (e.g. access options). However, this approach just provide an incomplete list of functional analytical elements that are predominantly not described in sufficient detail (e.g. “monitoring of predefined content” is rather generic), and it is not suitable for content configuration of individual analytical use cases (e.g. there are no information about performance indicators and their calculation).

Beyond methods for requirement documentation of analytical applications, existing scientific approaches for analytical service design just provide coarse-grained textual descriptions of the potential functional range of analytical services (e.g. [20, 37–39]). The approach of Besemer 2007 [19] already insert a set of analytical services in a two-tier hierarchical model distinguishing frontend and backend services. These analytical service approaches don’t contain detailed guidelines for modeling service specifications, nor sufficient clues for the systematic coupling of different services for the configuration of analytical use cases. Likewise, the service structures of service-oriented analytical software products (e.g. MicroStrategy and Oracle Business Intelligence Suite) are usually designed from a technical perspective (e.g. “Session Service”), and therefore they are not suitable for a conceptual and user-oriented specification of analytical applications.

In previous research regarding data privacy in the context of analytical application design, there is often a focus on the development and description of technical functions and software modules to support and ensure data protection requirements. That comprises available technical features and methods (e.g. [40]) and derived technical architectures (e.g. [41]), as well as research with special focus on the provision of personal data in an inter-organizational setting [15]. On the other hand, numerous research works address organizational (e.g. [42]) and content-related guidelines (e.g. [43, 44]) for the use of personal data within analytical applications. However, there are no methodological approaches to distinguish personal data with regulatory restrictions from other freely usable personal-related data (e.g. effectively anonymized data, group-related data) during configuration of analytical applications to support an early assessment of likely upcoming data privacy complications.

Standard process modeling notations (e.g. BPMN, Event-driven Process Chains (EPC)) are suitable for the presentation of activity sequences, but only provide black box data objects linked to activities for the documentation of informational process requirements. Furthermore, approaches based on these standard process notations and expanding the specification of data objects in many cases concentrate on the specification of underlying technical data models (e.g. [4]).

III. RESEARCH METHOD

The Design Science Research (DSR) Methodology Process by Peffers et al. 2007 [45] was used as a research method, because the current research is aligned with the objectives of DSR [45] and creates IT-related artifacts (analytical service models as a level 2 DSR contribution type (cf. [46]) to solve an organizational problem (insufficient requirements documentation of analytical applications). Within the first research step (**problem definition**), a self-conducted survey about the usage situation of analytical applications in the intensively analytical supported call center domain [47] provided initial practice-related information regarding analytical applications that did not meet business user requirements. Practice-oriented literature about problem areas in BI projects (e.g. [6]) as well as self-conducted interviews with BI project managers confirmed the inadequate specification of user requirements to be a main cause for project delays and for a non-requirement-based design of analytical applications. To **determine the design requirements for the analytical service models**, the authors initially conducted a survey and additional case studies regarding the use of analytical applications in the call center industry [47, 48] to gather information about the different areas of analytical-supported operation in practice (cf. chapter IV.A.1) and about the relevant design aspects and functional areas of real life analytical systems from different perspectives supplemented with a literature research (e.g. [49, 50]) (cf. chapter IV.A.2). To address data privacy risks regarding basic data and performance indicators during the requirements phase, the authors investigated the legal limits of data privacy when using analytical applications [43] (cf. IV.A.3). Because self-service analytics has become a trend in recent years [7], the service models should address analytical self-service functionality as well (cf. IV.A.4). Furthermore, an increasing spread of cross-domain analysis scenarios presupposes the consideration of enablers for cross-domain scenarios [9] within the analytical requirements specification (cf. IV.A.5).

In order to **create the analytical service models**, the top-down executed identification, naming and mutual connection of individual services (chapter IV.B) was based on existing analytical service architectures [19, 37, 39] and on the main functional and content areas of analytical applications (e.g. [35]). The bottom-up executed elaboration of detailed service specifications and features (chapter IV.C) considered the design requirements presented in chapter IV.A. To combine service specification with process modeling, chapter IV.D shows the integration of analytical services with the graphic process modeling notation BPMN. The **demonstration of the analytical service models** (chapter V) includes the retrospective specification of existing analytical applications and the configuration of a prospective cross-domain analysis use case. The **evaluation of the analytical service models** (chapter V) and their integration with process models was conducted by discussing and revising the model content with analytical experts.

IV. DEVELOPMENT OF ANALYTICAL SERVICE MODELS

A. Determination of Service Design Requirements

1) Analytical Supported Task Categories and Derived Configuration Areas

Case study research in call centers [48] led to four categories for operational task supported by analytical applications (1. "Cross-project Tasks"; 2. "Project Preparation"; 3. "Project Implementation"; 4. "Project Monitoring and Control"). Call centers are suitable as an object of investigation within this research, since 1.) there is a strong penetration of operational processes with analytical applications, and 2.) there are heterogeneous user groups (e.g. agents, team managers, external clients) and various data sources (e.g. telephone systems, voice analysis systems) reflecting a strong relationship to a cross-domain setting. The task categories mentioned above provide a clue for a delimitation of the configuration areas for analytical services:

- **Use Case-Overlapping Configuration Content** refers to the provision of the overall analytical application (e.g. navigation) independent from specific analytical use cases (i.e. individual dashboards).
- **Configuration Content for Analysis Preparation** addresses preliminary customizing work prior to the operational use of analytical applications by the users and describes the scope of analytical self-services.
- **Use Case-Specific Configuration Content** is based on the levels of Operational BI [2] and addresses the design of single reports or dashboards with diagrams, analytical information and functions for analytical information distribution, user access and automation.

2) Consideration of Comprehensive Content / Feature / Functional Areas of Analytical Applications

With focus on the "Use Case-Specific Configuration Content" and according to Schulze / Dittmar 2006 [36], an analytical application supports 1.) the right information 2.) created with the right analytical function 3.) to the right user 4.) in the right form and 5.) in the right time [36]. The presented modeling approach has to cover and enrich these five areas, since their individual design is a prerequisite for user-adapted analytical solutions [36]. Specific design requirements were derived from survey results regarding the practical use and functionality of analytical applications [47], the analysis of analytical products (e.g. MicroStrategy and

Oracle Business Intelligence Suite) and from a literature analysis (e.g. [7, 35, 51, 52]) regarding the functional characteristics, non-functional features and usage alternatives of analytical applications (e.g. forms of information visualization and distribution, user collaboration).

- From a **content perspective**, it is necessary to specify data sources (e.g. internal or external persistent or stream data originate from heterogeneous databases or file-oriented sources), the semantic meaning (e.g. performance indicators and dimensions) as well as the structure (e.g. numerical value) of analytical information and required data transformations. Here, increasing integration of cross-domain and sensor-based data sources in the context of IoT scenarios are of particular importance.
- **Periodicity** addresses update cycles when determining (e.g. calculating indicators) and providing (e.g. updating dashboard diagrams) analytical information.
- Regarding the **form of presentation**, there is a distinction between interactive dashboards and static reports [2], which can either be accessed online or offline with different (mobile) devices (running with specific systems software) within the analytical application. Reports can also be forwarded (e.g. via email) to different target systems in the case of different events (e.g. data update). Furthermore, specifications are necessary regarding diagram design and filter functions.
- Particularly in cross-domain settings, there are often different analytical business **user** groups or roles.
- **Analysis methods** specify the procedures for obtaining analytical information (e.g. calculation of indicators based on data mining algorithms).

Further design requirements relate to automatically executed actions (including real-time alerts [2])) for specific constellations of analytical information (e.g. threshold value violations of performance indicators). Furthermore, features and functions of an entire analytical application beyond individual analysis cases ("Use Case-Overlapping Configuration Content") has to be considered (e.g. navigation structure, collaborative elements (e.g. forums), retention of user settings, system languages, authentication).

3) Design Constraints Derived from Data Privacy Regulations

An appraisal about data privacy risks of basic analysis data or indicators within the requirements phase facilitates an early search for alternative data and a proactive consideration of necessary organizational (e.g. obtaining a permission for data use, enlistment of data privacy experts) and technical (e.g. consideration of de-personalization functions) aspects to impede subsequent and extensive system changes. The following aspects elaborated from previous own studies on the implementation of data privacy regulations in analytical applications [43] has to be considered:

- A distinction must be made between the use of personal customer data and personal employee data.
- For personal employee data, their use for abuse control and for cost analysis must be considered separately. For employee performance monitoring, it is necessary

to assess the degree of violation of personal rights and to consider existing company agreements and works council consents regarding personal employee data.

- Consideration of different personal data aggregation levels (formation of groups) and of requirements for data de-personalization and pseudonymization.
- Consideration of the customer's consent to data use and the permission of personal data usage in terms of self-published data and in (pre-)contractual relationships.

4) Design Requirements Derived from Self-Service Analytics

The consideration of upcoming analytical self-services [7] has two effects for the design of the modeling approach:

- In the sense of mass customization [53], it means the direct participation of business users in the model-based and flexible service-oriented configuration of the entire analysis content and the functional scope of an analytical solution, as well as in the assessment of different design alternatives prior to the initial technical implementation. The goal is that users should conduct the specification of the analytical service models autonomously.
- A desired adaptation of the future analytical solution by the business users during runtime (e.g. redesigning dashboards, changing the underlying data model [7]) necessitates corresponding functions (e.g. user wizards for data model manipulation) in the later analytical application. According to that, the analytical service approach must address the selection and specification of such analytical self-services.

5) Design Requirements Derived from the Enablers for Cross-Domain Use Cases [9]

- **Semantic data specification:** Description of format, data availability and origin from a user's perspective.
- **Collaboration of actors:** To represent the compilation of an application with elements provided by different actors, the model elements have to be designed in a modular and flexibly combinable way as services (e.g. exchange of a data service).
- **Big data:** Consideration of different data formats as well as the specification of maximum response times and data quality requirements.
- **Internet of Things:** Consideration of both human and machine "users" of the analysis results.

B. Identified Analytical Services and their Relationships

In addition to the configuration areas for analytical applications (chapter IV.A.1), additional downstream structuring levels have been added (Fig. 1) designed according to the analytical architecture model of Burmester / Goeken 2006 [54]. The supplement of these levels follows the observation that an analytical application designed from a user's perspective has a modular structure: Single dashboards and reports (specified in the **Dashboard / Report Level**) each consist of one or more diagrams (**Diagram Level**), whereby a diagram again contains performance indicators / results from data and text mining analyzes (**Analysis Level**), and these performance indicators / results are based on one or more basic data sources (**Basic Data Level**).

Fig. 1 shows the assignment of the analytical services to the individual structuring levels in the configuration area "Use Case-Specific Configuration Content" and their relationships in form of a multi-level service network. A directed edge shows that a service is assigned to another service and enriches it in terms of configuration content. The dashboards described by "Dashboard Services" differ from reports on the one hand by the possibility of direct user interaction. On the other hand, only reports can be exported to different (cross-domain) users and target systems in various forms (e.g. as a jpg or ppt file) with the help of a "Distribution Service".

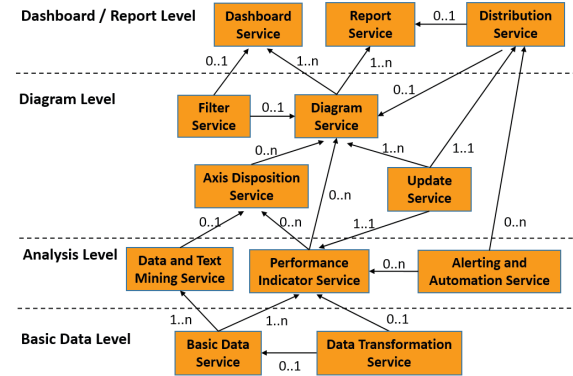


Fig. 1. Analytical services and their relationships in the configuration area "Use Case-Specific Configuration Content"

There are numerous design variants for diagrams in the "Diagram Service" from the user's perspective (e.g. table, bar diagram). In the case of diagrams with different axes, the assignment of the performance indicator values (e.g. "Rainfall", specified in a "Performance Indicator Service") and the associated indicator dimensions (e.g. "Time") to the individual diagram axes has to be recorded in the "Axis Disposition Service". In an "Alerting and Automation Service", user-independent and automated reactions to threshold violations of performance indicators must be specified, which can also come along with information distribution (e.g. sending an alert via email, specified in a "Distribution Service"). "Update Services" determine how and when to recalculate performance indicators, to update diagram visualizations and to redistribute reports. A "Basic Data Service" describes the data to be used for performance indicator calculations or mining analyzes from the user's perspective, including source system and data format, as well as with information relevant for an initial assessment of the criticality of any involved personal data from a data privacy's perspective. Finally, a "Data Transformation Service" contains demands in terms of data enrichment, data quality checks and data corrections regarding performance indicators that have already been calculated or other basic data.

Fig. 2 shows the analytical self-services identified on the basis of the functional sub-areas of self-service BI [7, 55] for the configuration area "Configuration Content for Analysis Preparation". These self-services specify the functional requirements of analytical users to adapt analytical application in order to customize reports ("Report Self-Service") and dashboards ("Dashboard Self-Service"), to change the data model of the underlying analysis data source ("Data Model Self-Service"), to integrate external data into the analysis data source ("Data Integration Self-Service") and to perform standalone data quality analyzes and data cleansings ("Data Quality

Self-Service"). Because self-services can be used independently from each other in specific analytical use cases, there are no direct mutual dependencies.

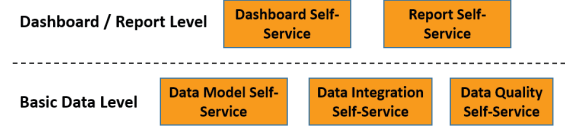


Fig. 2. Analytical self-services in the configuration area "Configuration Content for Analysis Preparation"

For the configuration area "Use Case-Overlapping Configuration Content" (Fig. 3), the "Superordinate Allocation Service" addresses requirements regarding the access and provision of the entire analytical application independent of specific analytical use cases (e.g. language support, temporal system availability, authentication). If desired, the "Collaboration Service" specifies requirements for user collaboration in analytical applications (e.g. provision of commenting functions and wikis).

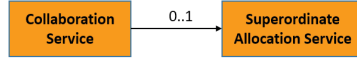


Fig. 3. Analytical services in the configuration area "Use Case-Overlapping Configuration Content"

C. Detailed Specifications of Analytical Services

The self-adapted modeling notation used to design and visualize the detailed specifications of analytical services (Fig. 4) is based on the Configuration Tree modeling approach [56]. A Configuration Tree is a suitable basic notation for this modeling context, because this approach allows building configuration models of applications and use cases enclosing structural and appliance elements with their attributes. An UML class diagram (as a widespread notation in software development) was not used, because it can't display the necessary XOR-relationships. In the current model version of the Configuration Tree, "Services" are the structural elements and "Service Features" are the appliance elements within the analytical service models. In addition, there are "Contain Relationships" including cardinalities (to visualize the links between different services, between services and subordinate

features as well as between superordinate and subordinate features) and "XOR Relationships" (to visualize alternative links between services and subordinate features as well as between superordinate and subordinate features). The attributes are both used to label service identifiers and to provide necessary information to specify the individual content of a service features if necessary (e.g. the feature "Max. amount of parallel users" needs a specification in form of an attribute entry (e.g. "500"), whereas other features only have to be selected and are therefore sufficiently specified (e.g. "Just online" as a needed work mode (Fig. 5)). The attributes marked with italic text contain identifiers as a reference to modeling objects (e.g. a dashboard or a performance indicator) that have already been specified in other service models.



Fig. 4. Modeling objects for the specification of analytical services

To specify a "Dashboard Service" (Fig. 5), the user has to assign a unique "Dashboard Identifier", link the required "Diagram Services" that contain the analytical content including the presentation of performance indicators, and link a "Filter Service" applicable for all diagrams within the dashboard if necessary. Afterwards, the user has to select or deselect (in case of "0..1"- or "0..n"- relationships) service features (e.g. the ability to hide dashboard elements during runtime) and has to specify the attributes of selected or required ("1..1"- or "1..n"-relationships) dashboard features (e.g. work mode (online / offline) for dashboard access).

For each basic data required for analysis (Fig. 6), its origin / source system, the type of data provision (streaming or persistent data) and the data type must be specified in a "Basic Data Service". In the case of personal data, additional information characterizing the data context are required to permit a first data privacy assessment.

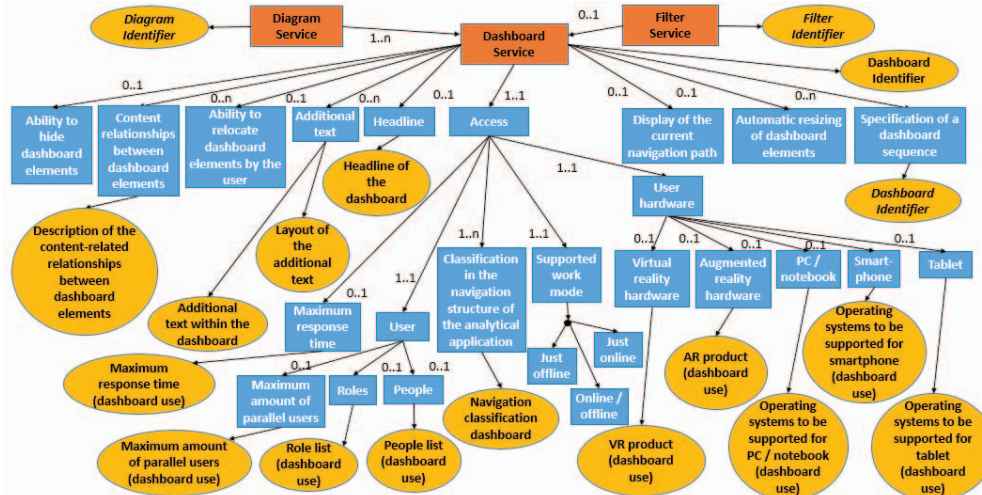


Fig. 5. Analytical service model „Dashboard Service“

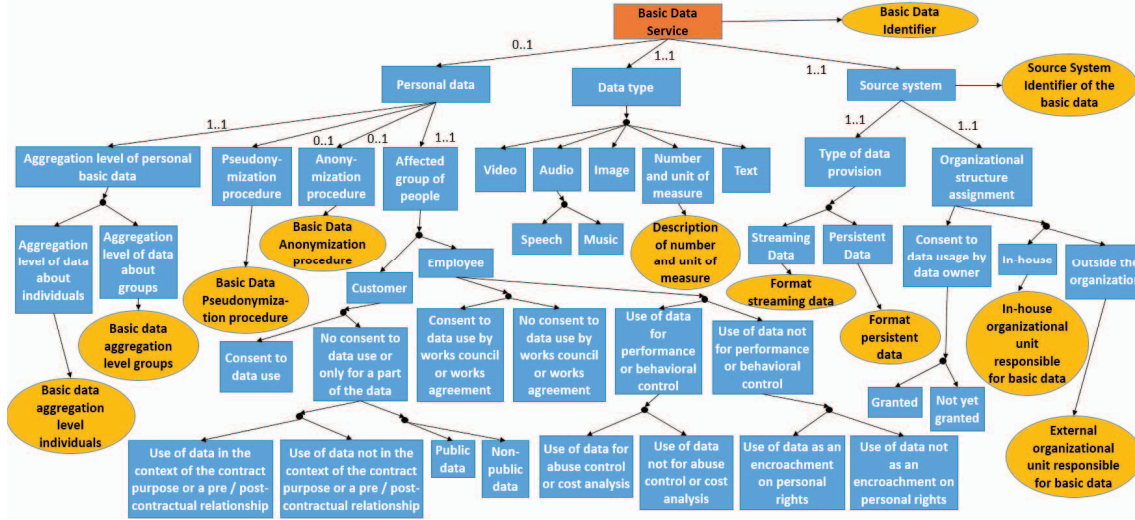


Fig. 6. Analytical service model „Basic Data Service“

D. Integration of Analytical Services in Process Models

Reports and dashboards can provide information essential to execute operational process activities (process input) and / or they can visualize analytical results of activities (process output). Therefore, "Dashboard Services", "Report Services" and their assigned analytical services on the other structuring levels represent information or data objects as they already exist in standard process modeling notations (e.g. EPC, BPMN). These data objects can serve as a starting point for the graphic representation of analytical service (belonging to the configuration area "Use Case-Specific Configuration Content") in process models. The following identifiers are proposed to mark data objects in process models as analytical services (in addition to the guidelines of their standard process notation): "Dashboard Service" - "Dash-S", "Report Service" - "Report-S", "Distribution Service" - "Dist-S", "Filter Service" - "Filter-S", "Diagram Service" - "Diagram-S", "Axis Disposition Service" - "Axis-S", "Update Service" - "Update-S", "Data and Text Mining Service" - "Mining-S", "Performance Indicator Service" - "Indicator-S", "Alerting and Automation Service" - "Alert-S", "Basic Data Service" - "Data-S", "Data Transformation Service" - "Transform-S".

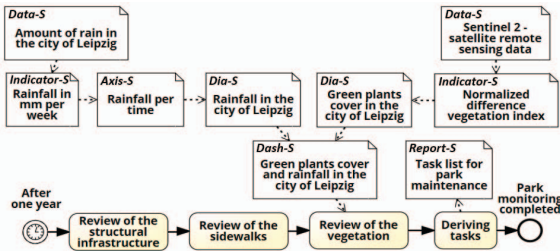


Fig. 7. Representation of analytical services in the form of data objects in a BPMN process model using the example of a "Park Monitoring" process

In a first reduced visualization variant, only "Report Services" and "Dashboard Services" are connected to the respective activities in the process model. In addition, there is another variant to add the identifiers of the other linked services (see the example process for "Park Monitoring" in Fig. 7) in order to gain a deeper insight into the content of the reports / dashboards used in the process context. However, a

complete integration of the detailed service specifications (chapter IV.C) in process models doesn't seem appropriate due to the high complexity of single service models and the resulting increasing complexity in process models.

V. EVALUATION AND APPLICATION EXAMPLE

The validation of an artifact is a central requirement of the DSR approach [46]. The evaluation ("Proof of Concept", cf. [46]) of an initial version of the analytical services was carried out through four 2-4 hours presentations and discussions of the analytical service approach with analytical experts (project managers for analytical applications, software engineers, scientists with expertise in conceptual and technical software development) from March to June 2019. The results of these discussions led to iterative adjustments to the model structures (e.g. shifting service features to other services, reducing the total number of services, adding missing features and deleting less relevant features). At the last of these presentations with a software engineer from a large provider of analytical applications, a questionnaire-based feedback was collected on the practical application potential of this approach:

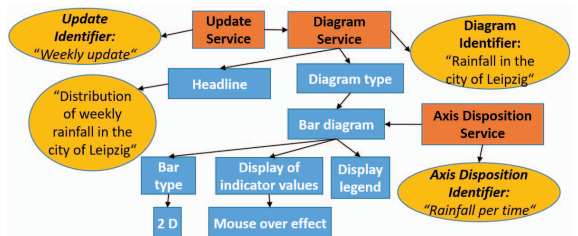
- High benefits to achieve clear / complete requirements.
- Increased benefits in saving effort in practical projects.
- High utility as a template for deriving a technical concept.
- Increased benefits when matching requirements between heterogeneous business users.
- Increased benefits for the transparent representation of service relationships.
- High utility for gaining an overview of the analytical process support and for displaying the sequence of analysis content in processes.
- Experienced modelers / consultants have to support analytically unexperienced casual users during the configuration of the analytical service models.

The presentation of the applicability of the modeling approach ("Proof of Use", cf. [46]) was initially carried out by the retrospective modeling of existing analytical applications

- To develop a new analytical use case, a business user should start with the top-down identification of needed services on the different structuring levels (Fig. 1) beginning with the “Dashboard / Report Level”.
- Afterwards, the user should proceed with the detailed specification of the selected services either top-down or bottom-up (starting with the “Basic Data Level”).

The diagram illustrates the conceptual model of a dashboard, showing the relationships between various components. The central component is the **Dashboard Service**, which is connected to three **Diagram Service** components and a **Diagram Identifier** component. The **Diagram Identifier** components are labeled with specific identifiers: "Rainfall in the city of Leipzig", "Green plants cover in the city of Leipzig", and "Green plants cover and rainfall in the city of Leipzig". The **Diagram Service** components are connected to the **Dashboard Service** and the **Diagram Identifier** components. The **Dashboard Service** is also connected to a **User** component, an **Access** component, and a **Roles** component. The **User** component is connected to a **Maximum amount of parallel users** component. The **Access** component is connected to a **Classification in the navigation structure of the analytical application** component. The **Roles** component is connected to a **Role list (dashboard use):** component, which lists roles such as "Employees department Field Management", "Employees department public park management", and "10". The **Classification in the navigation structure of the analytical application** component is connected to a **Supported work mode** component, which is further connected to **Online / offline** and **Navigation classification dashboard: for management / public racks**. The **Supported work mode** component is also connected to a **PC / Notebook** component, which is further connected to **Operating systems to be supported for PC / notebook (dashboard use):** and "Windows 7, 8, 10". The **Content relationships between dashboard elements** component is connected to a **Description:** component, which contains a text description of the dashboard's content relationships.

Fig. 9 shows the specification of the “Diagram Service” “Rainfall in the city of Leipzig” as a 2-dimensional bar diagram, whereat the linked “Axis Disposition Service” “Rainfall per time” (Fig. 10) illustrates the assignment of the performance indicator “Rainfall in mm per week” and the related indicator dimension “Time” to the respective diagram axes including further axes visualization parameters. For the indicator to be calculated “Rainfall in mm per week”,



For the “Performance Indicator Service” “Rainfall in mm per week”, Fig. 11 specifies the indicator unit (“mm of rain / m²”), the scenario-relevant indicator dimension (“Time”), the link to the update service (to specify the weekly indicator recalculation interval as a basis for the weekly diagram update (cf. Fig. 9)) and the analysis method needed here. In this

```

graph TD
    PIS[Performance Indicator Service]
    US[Update Service]
    UI([Update Identifier:  
"Weekly update"'])
    ND[Number of decimal places]
    IU[Indicator unit]
    ID[Indicator dimension]
    IHC[Indicator has to be calculated]
    BDS[Basic Data Service]
    BDI([Basic Data Identifier:  
"Amount of rain in the city of Leipzig"'])
    AM[Analysis method]
    BA[Basic arithmetic operations]
    T([Time])
    MR([mm of rain / m^2])
    Z([0])

    PIS <--> US
    PIS --- UI
    PIS --- ND
    PIS --- IU
    PIS --- ID
    PIS --- IHC
    IHC --- BDI
    IHC --- AM
    IHC --- BA
    BDI --- AM
    BDI --- BA
    AM --- BA
    IU --- T
    IU --- MR
    ND --- Z
  
```

The diagram illustrates the structure of a data service. It features a central 'Performance Indicator Service' box. This box is connected to an 'Update Service' box and an 'Update Identifier' oval. The 'Performance Indicator Service' box is also connected to four blue boxes: 'Number of decimal places', 'Indicator unit', 'Indicator dimension', and 'Indicator has to be calculated'. The 'Update Identifier' oval is connected to an 'Update Service' box. The 'Indicator has to be calculated' box is connected to a 'Basic Data Service' box. The 'Basic Data Service' box is connected to a 'Basic Data Identifier' oval and an 'Analysis method' box. The 'Analysis method' box is connected to a 'Basic arithmetic operations' box. The 'Indicator unit' box is connected to a 'Time' oval and a 'mm of rain / m²' oval. The 'Number of decimal places' box is connected to a '0' oval. The 'Indicator dimension' box is connected to a 'Time' oval. The 'Basic Data Identifier' oval is connected to a 'Basic Data Identifier' oval. The 'Basic Data Identifier' oval is connected to a 'Basic Data Identifier' oval.

Finally, the “Basic Data Service” in Fig. 12 contain first information about the structure and data format of the expected rain gauge sensor data and about the already known organizational affiliation of the data source including status information regarding the already obtained data access permission.

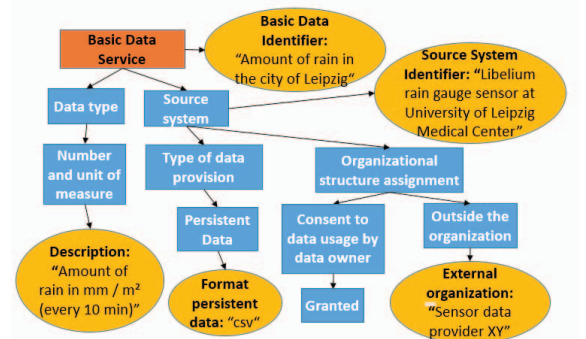


Fig. 7 shows the integration of the analytical services of this use case into the operational process.

This research work presents a modeling approach for the configuration of analytical services and their integration into process models, which can be used in particular to document user requirements regarding the design of analytical applications and analytical use cases. The design requirements, initially developed from different perspectives, form the framework for the development of the analytical

service models. This approach represents a significant enhancement both compared to approaches of model-based requirements documentation focusing analytical applications (e.g. [30]), and compared to conceptual-oriented modeling approaches for analytical services (cf. [19, 37, 39]). This affects primarily the introduction of three configuration areas and four structuring levels, the representation of analytical service relationships across the multi-level service network, the elaboration of detailed analytical service specifications and service features in all five essential design areas of analytical applications [36] enabling individual analytical use case configuration, as well as the consideration of analytical self-service functions and data privacy constraints.

The detailed specification of individual analytical services and the coupling of individual services via identifiers enables the single use case-related conceptual configuration of complex dashboards and reports in form of conceptual service networks. This configuration approach, with its flexible options for connecting different data with different analysis functions and the assignment of performance indicators to diagrams and higher-level reports / dashboards, meets the practical need to build analytical applications and functions in a modular and flexible manner and to reconfigure these applications based on changing user requirements [18]. As a result for software engineering, the service-oriented approach currently focused on the technical implementation of analytical applications is extended to the previous conceptual design to support seamless service-oriented specification from the beginning of system development. In order to represent the use of individual analytical applications (respectively of the dashboards and reports as information-transmitting machine-human interfaces) within operational process activities, data objects are used as existing modeling elements of standard process modeling languages. The presented guidelines to add analytical service labels to process model data objects represent a lightweight enhancement of existing and established process modeling notations and facilitate the applicability of this modeling approach.

The evaluation has shown that the analytical service models support the coordination between heterogeneous analytical user groups in the cross-domain context, and that they enable extensive documentation of functional and non-functional requirements as a conceptual blueprint for a subsequent technical implementation. Due to the complex model structures and to support an efficient elaboration and coordination of analytical design variants, this modeling approach should be used in a direct interaction between analytical users and software developers in terms of a customer co-design [53].

The analytical service models provide a configuration space containing the essential content-related and functional / non-functional aspects of the analytical application design. They are therefore not complete and must be expanded and adapted in the future to take account of new and changing requirements (e.g., emergence of new analytical functions and application scenarios). Furthermore, this approach has been evaluated in interviews and in a prototypical use case scenario, but has not yet been tested in real business projects. Further research should examine the design of modeling platforms for analytical service specification and the reduction of complexity regarding analytical service modeling in order to enable an autonomous service configuration for casual users.

The development of a procedure method for this modeling approach is another possible issue for future research.

REFERENCES

- [1] A.D.N. Sarma, "A Generic Functional Architecture for Operational BI System," *International Journal of Business Intelligence Research*, vol. 9, no. 1, pp. 64–77, 2018, doi: 10.4018/IJBIR.2018010105.
- [2] W. W. Eckerson, "Best Practices in Operational BI – Converging Analytical and Operational Processes," Renton (WA), 2007.
- [3] M. Hammer, "The process audit," *Harvard Business Review*, vol. 85, no. 4, pp. 111–123, 2007.
- [4] A. Meyer, L. Pufahl, D. Fahland, and M. Weske, "Modeling and Enacting Complex Data Dependencies in Business Processes," in *11th International Conference Business Process Management BPM 2013*, Beijing, China, 2013, pp. 171–186.
- [5] P. Uria-Recio, Top 25 Mistakes Corporates Make in their Advanced Analytics Programs. [Online]. Available: <https://towardsdatascience.com/top-25-mistakes-corporates-make-in-their-advanced-analytics-programs-c51e76218e20>
- [6] A. Cordoba, Three reasons reporting and analytics projects fail and how to avoid the pitfalls. [Online]. Available: https://www.sas.com/en_us/insights/articles/analytics/three-reasons-reporting-and-analytics-projects-fail-and-how-to-avoid-the-pitfalls.html
- [7] P. Alpar and M. Schulz, "Self-Service Business Intelligence," *Bus Inf Syst Eng*, vol. 58, no. 2, pp. 151–155, 2016, doi: 10.1007/s12599-016-0424-6.
- [8] B. Goodwin, Poor communication to blame for business intelligence failure, says Gartner. [Online]. Available: <http://www.computerweekly.com/news/1280094776/Poor-communication-to-blame-for-business-intelligence-failure-says-Gartner> (accessed: Apr. 27 2016).
- [9] S. Bär, O. Reinhold, and R. Alt, "The Role of Cross-Domain Use Cases in IoT - A Case Analysis," in *Proceedings 52nd Hawaii International Conference on System Sciences (HICSS) 2019*, Maui, 2019.
- [10] M. H. ur Rehman, I. Yaqoob, K. Salah, M. Imran, P. P. Jayaraman, and C. Perera, "The role of big data analytics in industrial Internet of Things," *Future Generation Computer Systems*, vol. 99, pp. 247–259, 2019, doi: 10.1016/j.future.2019.04.020.
- [11] F. Stroh, R. Winter, and F. Wortmann, "Method Support of Information Requirements Analysis for Analytical Information Systems," *Bus Inf Syst Eng*, vol. 3, no. 1, pp. 33–43, 2011, doi: 10.1007/s12599-010-0138-0.
- [12] J. Misra, S. Sengupta, and S. Podder, "Topic cohesion preserving requirements clustering," in *Proceedings of the 5th International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering - RAISE '16*, Austin, Texas, 2016, pp. 22–28.
- [13] F. G. C. Ribeiro, C. E. Pereira, A. Rettberg, and M. S. Soares, "Model-based requirements specification of real-time systems with UML, SysML and MARTE," *Softw Syst Model*, vol. 17, no. 1, pp. 343–361, 2018, doi: 10.1007/s10270-016-0525-1.
- [14] M. Goeken, "Anforderungsmanagement bei der Entwicklung von Data Warehouse-Systemen," in *Auf dem Weg zur Integration Factory - Proceedings der DW2004*, 2004, pp. 167–186.
- [15] S. Sharma, K. Chen, and A. Shet, "Towards Practical Privacy-Preserving Analytics for IoT and Cloud-Based Healthcare Systems," *IEEE Internet Computing*, March-April, 2018.
- [16] J. Schiefer and A. Seufert, "Towards a Service-Oriented Architecture for Operational BI," in *Multikonferenz Wirtschaftsinformatik 2010*, 2010, pp. 1137–1149.
- [17] Z. Panian, "How to Make Business Intelligence Actionable through Service-oriented Architectures," in *2nd WSEAS International Conference on Computer Engineering and Applications*, 2008, pp. 210–221.
- [18] E. Colangelo and T. Bauernhansl, "Usage of Analytical Services in Industry Today and Tomorrow," *Procedia CIRP*, vol. 57, pp. 276–280, 2016, doi: 10.1016/j.procir.2016.11.048.
- [19] D. Besemer, "Getting started now on SOA for BI," *DM Review*, vol. 17, no. 5, 26–37, 2007.
- [20] Z. Sun, L. Sun, and K. Strang, "Big Data Analytics Services for Enhancing Business Intelligence," *Journal of Computer Information Systems*, vol. 58, no. 2, pp. 162–169, 2018, doi: 10.1080/08874417.2016.1220239.

- [21] C. Hrach and R. Alt, "Functional Service Description on Service Marketplaces," in *Proceedings Multi-Konferenz Wirtschaftsinformatik (MKWI 2018)*, Lüneburg, 2018, pp. 569–575.
- [22] O. Liskin, "How Artifacts Support and Impede Requirements Communication," in *Lecture notes in computer science, Requirements Engineering: Foundation for Software Quality*, S. A. Fricker and K. Schneider, Eds., Cham: Springer International Publishing, 2015, pp. 132–147.
- [23] U. Frank, "Conceptual Modelling as the Core of the Information Systems Discipline - Perspectives and Epistemological Challenges," in *AMCIS 1999 Proceedings*, 1999, pp. 695–697.
- [24] A. Bonifati, F. Cattaneo, S. Ceri, A. Fuggetta, S. Paraboschi, and P. Di Milano, "Designing Data Marts for Data Warehouses," *ACM Transactions on Software Engineering and Methodology*, vol. 10, pp. 452–483, 2001.
- [25] D. Calvanese, L. Dragone, D. Nardi, R. Rosati, and S. M. Trisolini, "Enterprise modeling and Data Warehousing in Telecom Italia," *Information Systems*, vol. 31, no. 1, pp. 1–32, 2006, doi: 10.1016/j.is.2004.07.002.
- [26] P. Giorgini, S. Rizzi, and M. Garzetti, "GRAnD: A goal-oriented approach to requirement analysis in data warehouses," *Decision Support Systems*, vol. 45, no. 1, pp. 4–21, 2008, doi: 10.1016/j.dss.2006.12.001.
- [27] G. Shanks and P. Darke, "Understanding corporate data models," *Information & Management*, vol. 35, no. 1, pp. 19–30, 1999, doi: 10.1016/S0378-7206(98)00078-0.
- [28] B. Strauch, "Entwicklung einer Methode für die Informationsbedarfsanalyse im Data Warehousing," Universität St. Gallen, St. Gallen, 2002.
- [29] M. A. Teruel, A. Maté, E. Navarro, P. González, and J. C. Trujillo, "The New Era of Business Intelligence Applications: Building from a Collaborative Point of View," *Bus Inf Syst Eng*, vol. 61, no. 5, pp. 615–634, 2019, doi: 10.1007/s12599-019-00578-3.
- [30] A. Ferrández, A. Maté, J. Peral, J. Trujillo, E. de Gregorio, and M.-A. Aufaure, "A framework for enriching Data Warehouse analysis with Question Answering systems," *J Intell Inf Syst*, vol. 46, no. 1, pp. 61–82, 2016, doi: 10.1007/s10844-014-0351-2.
- [31] C. Rosenkranz, R. Holten, M. Räkens, and W. Behrmann, "Supporting the design of data integration requirements during the development of data warehouses: a communication theory-based approach," *European Journal of Information Systems*, vol. 26, no. 1, pp. 84–115, 2017, doi: 10.1057/ejis.2015.22.
- [32] P. Jovanovic, O. Romero, A. Simitsis, A. Abelló, and D. Mayorova, "A requirement-driven approach to the design and evolution of data warehouses," *Information Systems*, vol. 44, pp. 94–119, 2014, doi: 10.1016/j.is.2014.01.004.
- [33] J. Horkoff, D. Barone, L. Jiang, E. Yu, D. Amyot, A. Borgida, J. Mylopoulos, "Strategic business modeling: representation and reasoning," *Softw Syst Model*, vol. 13, no. 3, pp. 1015–1041, 2014, doi: 10.1007/s10270-012-0290-8.
- [34] A. Maté and J. Trujillo, "A trace metamodel proposal based on the model driven architecture framework for the traceability of user requirements in data warehouses," *Information Systems*, vol. 37, no. 8, pp. 753–766, 2012, doi: 10.1016/j.is.2012.05.003.
- [35] J. H. Mayer, R. Winter, and T. Mohr, "Situational Management Support Systems," *Bus Inf Syst Eng*, vol. 4, no. 6, pp. 331–345, 2012, doi: 10.1007/s12599-012-0233-5.
- [36] K. D. Schulze and C. Dittmar, "Business Intelligence Reifegradmodelle," in *Analytische Informationssysteme: Business Intelligence-Technologien und -Anwendungen*, P. Chameni and P. Gluchowski, Eds., 3rd ed., Berlin: Springer Verlag, 2006, pp. 72–87.
- [37] B. Dinter and F. Stroh, "Design Factors for Service-oriented Architecture Applied to Analytical Information Systems: an Explorative Analysis," in *Proceedings of the 17th European Conference On Information Systems*, 2009.
- [38] B. Dinter, "Einsatzmöglichkeiten serviceorientierter Architekturen in der Informationslogistik," in *Active Enterprise Intelligence: Unternehmensweite Informationslogistik als Basis einer wertorientierten Unternehmenssteuerung*, J. Töpfer and R. Winter, Eds., Berlin: Springer Verlag, 2008, 221–242.
- [39] W. Martin, "Analytics meets Enterprise SOA," S.A.R.L. Martin, 2006.
- [40] G. d' Acquisto, J. Domingo-Ferrer, P. Kikiras, V. Torra, Y.-A. d. Montjoye, and A. Bourka, "Privacy by design in big data: An overview of privacy enhancing technologies in the era of big data analytics. Heraklion: ENISA, 2015], 2015.
- [41] Y.-T. Lee, W.-H. Hsiao, Y.-S. Lin, and S.-C. T. Chou, "Privacy-preserving data analytics in cloud-based smart home with community hierarchy," *IEEE Trans. Consumer Electron.*, vol. 63, no. 2, pp. 200–207, 2017, doi: 10.1109/TCE.2017.014777.
- [42] H. Drachler and W. Greller, "Privacy and analytics," in *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge - LAK '16*, Edinburgh, United Kingdom, 2016, pp. 89–98.
- [43] C. Hrach, R. Alt, and L. Nöbel, "Datenschutz im Call-Center," *HMD - Praxis der Wirtschaftsinformatik*, vol. 48, no. 281, pp. 71–79, 2011.
- [44] M. J. Rodríguez-Triana, A. Martínez-Monés, and S. Villagrà-Sobrino, "Learning analytics in small-scale teacher-led innovations: Ethical and data privacy issues," *JLA*, vol. 3, no. 1, pp. 43–65, 2016, doi: 10.18608/jla.2016.31.4.
- [45] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, 2007, doi: 10.2753/MIS0742-1222240302.
- [46] S. Gregor and A. R. Hevner, "Positioning and Presenting Design Science Research for Maximum Impact," *MISQ*, vol. 37, no. 2, pp. 337–355, 2013, doi: 10.25300/MISQ/2013/37.2.01.
- [47] C. Hrach and R. Alt, "Anwendungspotenziale für Business Intelligence-Technologien im Call Center-Bereich," in *Proceedings 9. Internationale Tagung Wirtschaftsinformatik (WI2009)*, Wien, 2009, pp. 369–378.
- [48] C. Hrach and R. Alt, "Operational Business Intelligence bei Call Centern – Erkenntnisse einer Fallstudienuntersuchung," in *Tagungsband der Multikonferenz Wirtschaftsinformatik (MKWI) 2012*, Braunschweig, 2012, pp. 1131–1143.
- [49] W. W. Eckerson, *Performance dashboards: Measuring, monitoring, and managing your business*. Hoboken, NJ: Wiley, 2006. [Online]. Available: <http://www.loc.gov/catdir/enhancements/fy0622/2005011800-d.html>
- [50] R. Kimball and M. Ross, *The data warehouse toolkit: The complete guide to dimensional modeling*, 2nd ed. New York, NY [u.a.]: Wiley, 2009.
- [51] R. Kimball, M. Ross, W. Thornthwaite, J. Mundy, and B. Becker, *The Data Warehouse Lifecycle Toolkit*. Indianapolis: Wiley, 2008.
- [52] P. Chameni and P. Gluchowski, "Analytische Informationssysteme – Einordnung und Überblick," in *Analytische Informationssysteme: Business Intelligence-Technologien und -Anwendungen*, P. Gluchowski and P. Chameni, Eds., 5th ed.: Springer Verlag, 2016, pp. 3–12.
- [53] F. T. Pillar, "Mass Customization: Reflections on the State of the Concept," *Int J Flex Manuf Syst*, vol. 16, no. 4, pp. 313–334, 2004, doi: 10.1007/s10696-005-5170-x.
- [54] L. Burmester and M. Goeken, "Method for User Oriented Modelling of Data Warehouse Systems," in *Proceedings of the 8th International Conference on Enterprise Information Systems, ICEIS 2006*, Paphos/ Zypern, 2006.
- [55] F. Geist, T. Klun, and H. Ritz, "Self-Service Business Intelligence (SSBI) – Nutzenpotenziale für einen verbesserten Austausch von Informationen im Unternehmen," in *Tagungsband zur 26. AKWI-Jahrestagung*, 2013, pp. 47–58.
- [56] C. W. Krueger, "Multistage configuration trees for managing product family trees," in *Proceedings of the 17th International Software Product Line Conference on - SPLC '13*, Tokyo, Japan, 2013, p. 188.