Key Concepts of Data Management – an Empirical Approach

Andreas Grillenberger
Computing Education Research Group
Friedrich-Alexander-Universität Erlangen-Nürnberg
91058 Erlangen, Germany
andreas.grillenberger@fau.de

Ralf Romeike
Computing Education Research Group
Friedrich-Alexander-Universität Erlangen-Nürnberg
91058 Erlangen, Germany
ralf.romeike@fau.de

ABSTRACT

When preparing new topics for teaching, it is important to identify their central aspects. Sets of fundamental ideas, great principles or big ideas have already been described for several parts of computer science. Yet, existing catalogs of ideas, principles and concepts of computer science only consider the field *data management* marginally. However, we assume that several concepts of data management are fundamental to CS and, despite the significant changes in this field in recent years, have long-term relevance. In order to provide a comprehensive overview of the key concepts of data management and to bring relevant parts of this field to school, we describe and use an empirical approach to determine such central aspects systematically. This results in a model of key concepts of data management. On the basis of examples, we show how the model can be interpreted and used in different contexts and settings.

CCS CONCEPTS

• **Social and professional topics** → *K-12 education*;

KEYWORDS

Data Management, CS Education, Key Concepts, Principles, Mechanics, Practices, Core Technologies, Model

ACM Reference Format:
Andreas Grillenberger and Ralf Romeike. 2017. Key Concepts of Data Management – an Empirical Approach. In *Proceedings of Koli Calling 2017*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3141880.3141886

1 MOTIVATION

In computer science education research, there is strong consensus that teaching should focus on aspects that are fundamental to the subject and relevant in the long term instead of short-lived technical developments. For this reason, various catalogs of principles, ideas and concepts, which characterize computer science or one of its areas, have been proposed over the past 30 years. These catalogs can, for example, be used when *preparing new topics for teaching*, as a basis for *developing curricula* and for *gaining insight into the field* and its central aspects. Also, according to Denning [8], such characterizations also *raise understandability* by shifting the focus from a technological perspective to the principles. They also support coming to a "balance between concepts and practice" [8] by *emphasizing the practices* in the field, and help *shape its public image* by giving a broader overview.

Well-known examples in computer science are the *Great Principles of Computing* [7] and the *Fundamental Ideas of Computer Science* [24]. As concrete decisions for lesson design or for developing curricula can be made based on such catalogs, these are highly popular in not only in CS education research but also in practice. Although they emphasize long-lasting aspects of CS, with the continuous development and differentiation of computer science, new topics are added to these catalogs and traditional ones are revised. For example, about ten years after Schwill proposed the Fundamental Ideas of CS, Modrow added *Fundamental Ideas of Theoretical Computer Science* [22]. Some years later, Modrow and Strecker [10] regarded the fundamental ideas from a different perspective and identified *Fundamental Ideas of CS in Schools* [10]. Also, in 2011 Bell et al. investigated the *Big Ideas of K–12 Computer Science* [1].

Even in newer approaches, it has not been examined whether the field *data management* contains aspects that may be considered as fundamental, probably because it is a rather young field of computer science that was subject to significant changes in recent years. As a result, most catalogs only consider aspects from this field marginally. Given the high relevance of data management not only to CS but also to our daily life, it is surprising that this field was not investigated further from an educational perspective yet. *Data management* is the field of CS concerned with *controlling, protecting, delivering and enhancing the value of data* [5]. It is reasonable to assume that many "data management" topics bring ideas that are (at least partly) relevant and suitable for (secondary) CS education. Ideas from this field are becoming increasingly relevant for everyday life and competencies related to data management are important not only for computer scientists. As everyone constantly generates and handles large amounts of data, basic competencies in data management are needed [14]: Such competencies are not only relevant when safely storing data in professional environments, but also for handling synchronization conflicts, creating appropriate backups of data, profiting of today's possibilities for analyzing and visualizing data and when using popular data-related services in a meaningful and responsible way. Thus, a basic understanding of data management becomes a matter of general education: According to Heymann [16], general education should, i. a. prepare for later life, support developing a understanding of the world and help to develop critical thinking—aspects which can directly be related to how students encounter and use data throughout their lives.

Koli Calling 2017, November 16–19, 2017, Koli, Finland

ACM ISBN 978-1-4503-5301-4/17/11…$15.00
https://doi.org/10.1145/3141880.3141886

As basis for preparing this subject area for CS teaching, it is essential to determine its key concepts. In the following, we will first characterize the field *data management* and its central developments from a scientific perspective and outline its current representation in CS teaching at secondary schools. We will then present an empirical approach for systematically identifying key concepts. By applying it to *data management*, we will then develop and present a *model of key concepts of data management* that characterizes this field from a CS education perspective with a focus on its core technologies, practices, and central principles.

2 RELATED WORK

2.1 Data Management in Computer Science and Computer Science Education

Data management is one of the most important and innovative areas of computer science, from both a technical and an everyday perspective. This is not only because a multitude of new topics, methods and concepts was added to this field in recent years. With this increasing relevance and the growing influence of innovations and developments, the traditional area *databases* evolved into *data management*. One of the most well-known and central topics of this development is *big data*, which is concerned not only with enormous amounts of data, but also the high speed of data generation and analysis, and the increasingly varying types of data[1]. With this shift in focus, new demands on storing and processing data arise: For example, partition tolerance is becoming central for distributed data storage, synchronization of data is rather the rule than the exception, and data quality needs to be assessed depending on the use-case. Also, proven requirements on data management systems are reassessed: As an example, storing data consistently had long been the focus in the field *databases*. Nowadays, however, modern non-relational NoSQL databases set their focus on high availability and distributed data storage, often neglecting consistency. This is for example suitable for many modern web applications, which often do not depend on permanent consistency. Behind this focus on the technologies of data management, fundamental decisions can be recognized that have to be weighed when choosing or developing appropriate data management systems for a use-case: The *CAP theorem* (also known as Brewer's theorem) [2], for example, emphasizes the challenge that data cannot be stored consistently, partition tolerant and highly available at the same time; only two of these properties can be achieved simultaneously. Hence, identifying the relevant aspects for a use-case is important when selecting an appropriate system. In addition to the new technologies which have evolved in the last years, less technology-oriented areas of research are also emerging in data management. For example, data quality and its relevance for data analysis processes are central in this field today, so are the concepts of metadata and data security. The whole field data management has already been investigated by the international data management association DAMA, resulting in an extensive body of knowledge, which describes this field from a professional perspective (DAMA-DMBoK, [5]).

[1]Typically, *big data* is characterized by the three terms *velocity*, *volume* and *variety*, called the "three Vs"[19].

All these innovations have direct impact on how we use computer systems, e. g. when storing data in cloud storages, synchronizing personal data across different devices, using bases or coming into contact with metadata. Although competencies required for dealing with such systems are particularly important in computer science and in daily life [14], related topics have hardly been discussed in computer science education research since the introduction of database teaching in the early 1990s. Although the DAMA-DMBoK emphasizes several central aspects of data management, in general a systematic review of this fields and its topics from a CS education perspective has not yet taken place: Due to its purpose, the DAMA-DMBoK does not take into account requirements on general education, such as . Consequently, also only few approaches address aspects of data management other than databases. Hence, the gap between data management from a scientific perspective and its implementation at schools is evident: While (secondary) CS teaching in the context of data concentrates on traditional topics such as databases and data modeling, other aspects are hardly considered at all. As basic data management competencies and skills are becoming increasingly necessary in various contexts of our everyday life (cf. [14]), students should be able to acquire those in their CS education.

2.2 Principles, Concepts and Ideas of Computer Science

Characterizing and structuring a subject area by identifying its underlying ideas, concepts or principles is a widely accepted approach in (computer) science. In particular, such catalogs are often used for preparing a scientific discipline or one of its fields for school teaching. In computer science, the "Fundamental Ideas of Computer Science" [24] in particular are well-known and often referred to: based on Bruner's work [3], Schwill defined four (later five) criteria for fundamental ideas. According to these, a fundamental idea has to be represented in the broad range of computer science ("horizontal criterion") and should have been relevant in the past but also in the expected future ("criterion of time"). At the same time, however, it should also have a relation to everyday life ("criterion of sense"), be understandable on different cognitive levels ("vertical criterion") and it should strive for achieving an idealized but mostly unachievable goal ("criterion of objective"). By analyzing the software development process, which Schwill considers to be central to CS, he has identified a set of 63 fundamental ideas and three superordinate master ideas. Because of the focus on software development, other areas of CS were less represented in this set of ideas, which caused Modrow to apply this approach to theoretical computer science [22]. Thereby, he expanded and revised the original catalog by Schwill. Later, Modrow and Strecker [10] revised this approach again and concentrated it to six fundamental ideas that are characteristic for computer science. Zendler and Spannnagel [26, 27] also used Schwill's criteria for selecting central concepts and processes of computer science: In a questionnaire study, they asked university professors in CS to evaluate these criteria for a given list of terms. By clustering the results, they were able to determine which concepts and processes are accepted as being central to CS. In a different approach, Denning [7] characterized CS by

determining various core technologies that represent different applications and research areas of computer science, as well as five practices, design principles and mechanics. For determining these "Great Principles of Computing", he started with proposing central concepts of CS with a scientific perspective in mind.

Although there are many different approaches, of which not all can be summarized here, surprisingly none of the resulting catalogs considers data management more than marginally: For example, in the Great Principles of Computing, this field is mainly considered in one of the mechanics, "recollection". The short descriptions ("principle stories") of this category cover several important aspects like *hierarchies*, *persistence* or *sharing* [9]. Yet, several aspects of data management seem to be underrepresented. This may be a result of a strong focus on software and software development in CS, which can also be recognized in most computer science curricula and educational standards.

3 DETERMINING CENTRAL PRINCIPLES OF DATA MANAGEMENT

Taking into account the significant technological progress made in recent years, the entire field has to be thoroughly examined in order to determine its central aspects. As these aspects on the one hand are structuring the field and can be used as a key for getting insight into it, in the following we will refer to them as *key concepts*. For determining these, we will outline an approach for answering the questions *"Which are the concepts that are central to the field data management?"* and *"How can the key concepts of data management be structured in a comprehensible model?"*. In contrast to, for example, Schwill and Denning, who base their work on informal characterizations of computer science, we chose an empirical approach for investigating data management. We divided the process into two phases, which we will explain in the following. A complete overview of the analysis process is shown in fig. 1.

3.1 Phase 1: Investigation of the Field from an Educational Perspective

In the first phase, the goal is to obtain a comprehensive overview of the subject area. For investigating the field, we decided to follow the methodology of the qualitative content analysis as described by Mayring [21], which is appropriate for getting an overview of a corpus of literature. However, due to the specific objectives of our analysis and due to the division into two phases, we made some adjustments. In particular, some steps that were described separately by Mayring were combined in our approach. Hence, this first phase contained the following steps:

(1) **Literature Selection:** As basis for the analysis, the literature corpus had to be defined. As we wanted to gain a comprehensive overview of the field, we included six widely accepted textbooks on data management. Some of these cover the field in general and hence are suitable for gaining a broad overview. But we also include books focused on particular topics of data management and on recent issues in order to take into account these and related developments. Although most of the books were in German language [11, 17, 18, 23, 25], the analysis was not distorted: In order to include an international perspective, not only the text book on "Fundamentals of Database Systems" by Elmasri & Navathe [12] has been included, but also the previously mentioned DAMA-BMBoK, which takes into account the perspective of the international data management organization. During the analysis, we could not determine differences between the German and international literature. Hence, our analysis covers a wide range of books with different levels of detail and broadness. Hence, it gives deep insight into the field from various perspectives.

(2) **Category System:** The qualitative content analysis generally allows using three methods for creating the category system: deductively deriving it from an existing theory, creating it inductively during the analysis process, or combining both methods. As there was no basis to deductively build upon, and as the in our analysis the category system will

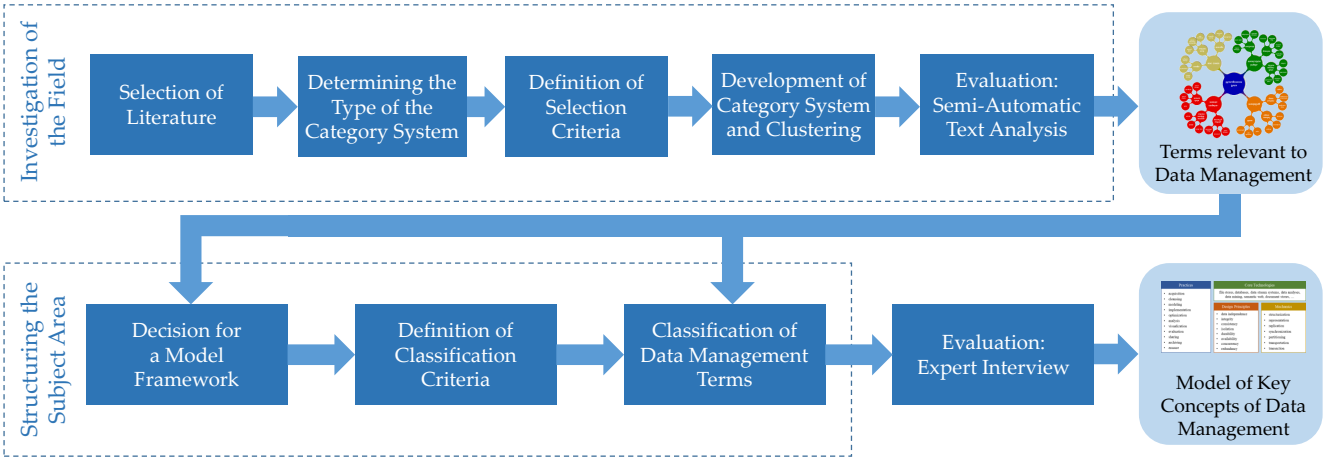

Figure 1: Overview of the analysis process.

represent the characterization which is to be created, using a (completely or partly) deductive system would result in a bias of the analysis. Hence, we decided for building up this system inductively during the analysis process.

(3) **Selection Criteria:** The last step before the coding phase is to set the analysis unit and criteria for deciding which aspects to include in the category system. Our selection criterion was that each analysis unit needs to describe an individual aspect of data management. As this is a relatively soft criterion, when in doubt, we included a term/phrase instead of dropping it. This does not distort the results, as this phase of the analysis aims on gaining an overview of the field, while in the second phase aspects that are too general or too detailed will be removed. The analysis unit was not further restricted in length, so that every term/phrase matching the selection criterion was included. Yet, of course always the shortest form of a term was used. The context of an analysis unit was not explicitly taken into account, except when it describes/characterizes a separate aspect of data management and hence was considered separately.

(4) **Coding Phase:**

   (a) **Development of the Category System:** This step was divided into multiple iterations. In each iteration, one book was analyzed. We decided to start using the DAMA-DMBoK, as it deems itself an extensive characterization of the field. Thus, we assumed that it helps to develop an initial prototype of the categorization system. In the subsequent iterations, additional sources were added. After the fourth step, only few additional topics were added, while additional sources particularly provided more details. This step led to a hierarchically organized category system, which describes central topics of the subject area and several details represented as subcategories. An excerpt of this system is shown in fig. 2.

   (b) **Clustering:** As the category system is relatively extensive, the terms mentioned in it were reorganized by clustering them manually. When clustering the terms, they fall into one of four categories: The practical application of data management, principles that need to be considered when designing but also when using data management systems and terms that describe how they work from a technical perspective. In addition, several activities related to data management were found. In fig. 3, we give an impression of the clustering process.

The results of this phase are not shown in detail here, as they are only intermediate and will function as the basis for the next analysis phase. Hence, they were validated by means of an additional partially automated text analysis. Therefore, we extracted the most common terms from another corpus of documents automatically, filtered out ambiguities and stopwords, and sorted them by their frequency. The resulting list was then searched manually for aspects of data management that fulfill the defined selection criterion. The corpus we used for this consisted primarily of lecture scripts by renowned scientists. Several of the documents cover the entire subject area, while other are focused on special topics. These were complemented with several other textbooks and a large portion of articles from the journal of the German special interest group on databases ("Datenbankspektrum"). Hence, 305 documents with 9447 pages/slides were analyzed, resulting in 229 terms that were mentioned at least 300 times each (after filtering ambiguities, plural forms and stopwords). These results were matched with those of the previous analysis by adding missing terms to the category system. Using this step, we could ensure a high degree of completeness of the intermediate results: Additional terms only provided more details and hence extended the characterization in depth but not in width.

3.2 Phase 2: Structuring the Subject Area

The previous phase led to an extensive list of terms on different levels of abstraction. In order to provide a more comprehensive overview and characterization of data management, the goal was to structure these terms and to emphasize the central concepts of this field. Looking at existing catalogs of ideas, principles or concepts, it is notable that the framework described by Denning for the "Great Principles of Computing" [7] considers the same four perspectives as we found them when clustering the terms in the previous phase. As it is also the common goal to characterize a scientific area, this framework also fits for our work. Thus, we adapted the great principles framework for our model model of key concepts of data management and thus divided it into four categories:

- **Core technologies** represent specific applications and/or technologies that are clearly related to data management. They also represent central research areas of the field.
- **Practices** in particular are activities/methods that are related to data management. However, they also represent competencies necessary for the use and/or development of data management systems and for handling data in general.
- **Design principles** need to be considered when designing data management systems. They can also be used for describing existing systems, as they represent their central characteristics.
- **Mechanics** are basic laws, assumptions, procedures or arrangements fundamental to the subject. They describe the basic operation of data management systems and the interaction between their components.

After classifying the terms found in the first phase into these categories, the model was still very extensive. For making it comprehensive and easy-to-read, several details had to be removed. This was accomplished in particular by combining multiple related aspects under superordinate terms. For example, the original practices "classification", "association" and "clustering" were all named in the context of data analysis; hence, they were replaced with the term "data analysis". Also, "normalization", "functional dependencies" or "compression" are aspects of "optimization", while "metadata", "data models" and "keys" are part of data "structuring".

In most cases, the assignment of terms to categories was unambiguous, as in particular the practices and core technologies could be clearly distinguished from the principles and mechanics. Yet, in some cases it was difficult to decide if a term rather represents a design principle or mechanism of data management. For such terms, the categorization was left open until the merging of terms

Figure 2: Visualization of an excerpt from the category system.

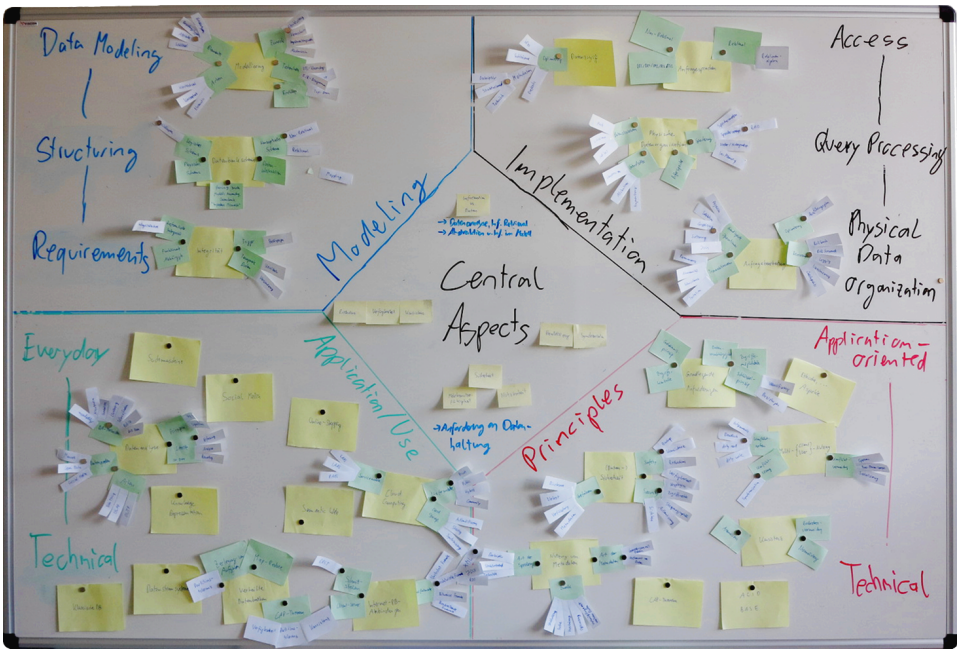

Figure 3: Clustering of terms during the coding phase.

was finished. This applied for example to "synchronization": While it is hard to decide whether it represents rather a mechanism (i. e. how does synchronization take place, which problems may occur, how are they solved) or a design principle (i. e. is synchronization possible in a system, how is data structured and prepared for synchronization, which effects does synchronization have on other design principles), it turned out that "integrity", "consistency" and "concurrency" already cover its design aspects. Hence, "synchronization" was included as mechanism. In other cases and especially when there was a greater scope for interpretation, such aspects were discussed among other researchers and categorized by mutual agreement, in order to achieve high objectivity.

4 MODEL OF KEY CONCEPTS OF DATA MANAGEMENT

Applying the approach to data management resulted in a model of key concepts of data management, which are divided into its core technologies, practices, design principles and mechanics (cf. fig. 4).

Of these aspects, the core technologies are the only category that is not stable over time, because of their special function of representing the current developments and research areas. Nevertheless, they are important to consider in a model characterizing the field as these technologies are usually the first point of contact for people. Some exemplary core technologies of data management are:

- file storages (such as regular file systems or cloud storages)
- data storages such as
  - relational and non-relational databases
  - in-memory databases
  - data stream systems
- document storages (such as web content management systems)
- data analysis / data mining
- methods for assessing data quality
- semantic web
- metadata
- web data interfaces

Similar to Denning's work, also the core technologies presented here are just a selection that is influenced by current developments in the field and may vary over time. Thus, it is to be expected that core technologies will be changed or added in the future or lose their relevance over time.

The typical activities of data management engineers, data scientists and other professionals in the field, but also of everyone else who handles data in everyday life, are represented in the model as "practices". They are clearly more extensive than the practices of the traditional field "databases" and hence give a clear insight into current prospects of data management. The practices of data management determined in our analysis are:

- **Acquisition**, i. e. capturing new data or getting access to existing data sets.
- **Cleansing**, i. e. stripping unnecessary or low-quality data (e. g. measurement errors).
- **Modeling**, especially for structuring data in a way that it can be stored and accessed efficiently.
- **Implementation** of the data model and storage of the data.
- **Optimization**, especially for accessing or storing data as efficient as possible.
- **Analysis**, i. e. gaining new information from these data using analysis methods.
- **Visualization**, which is concerned with visually editing information to make them easier to understand.
- **Evaluation**, especially of the analysis results and the quality of data they are based on.
- **Sharing** of original data, aggregated data and/or analysis results.
- **Archiving** data for a long-time, often for further but not yet predictable use-cases.
- **Erasing** of data marks the end of data usage.

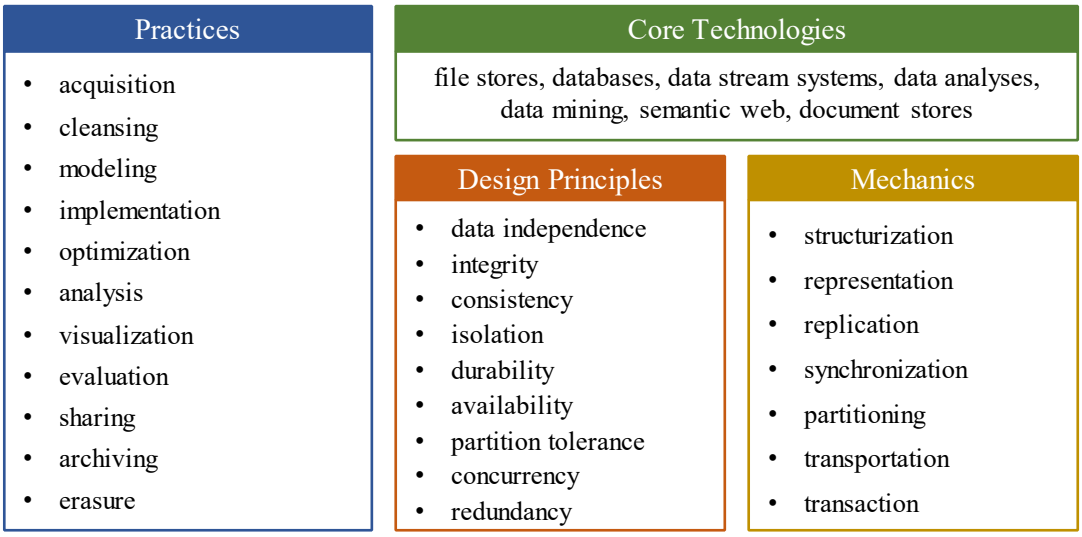

Figure 4: Model of key concepts of data management.

This set of practices is quite extensive. Yet, when trying to reduce the number of aspects mentioned here, we realized that none of them can be removed without losing important aspects.

Besides the practices, the mechanics describe how data management systems work, especially when storing, accessing and analyzing information:

- **Structurization** always takes place (implicitly or explicitly) when data is stored. For example, for facilitating access, index structures or meta data are added.
- **Representation** especially describes methods/techniques for storing data, e. g. in defined data structures.
- **Replication** takes place when data is stored redundantly in multiple data stores, e. g. for increasing fault tolerance of the overall system or its accessibility.
- **Synchronization** is used when concurrent access to data is made possible or when data stored on different data storages is kept identical.
- **Partitioning** of data is necessary when data cannot be stored in one storage but must be distributed to multiple storages. In contrast to replication, partitioning is not focused on storing duplicates of data on several systems, but instead divides data and stores parts of it in different storages. This is done in particular when data needs to be processed in parallel or when storing it on one system is not possible due to its size.
- **Transportation** of data is always involved when storing or retrieving data, but also when multiple data storages are connected in a distributed data storage system.
- **Transactions** or related techniques are involved when making data management systems fault-tolerant.

It is clearly visible that these mechanics underlie all typical data management systems. For instance, for storing data in databases, information needs to be *represented* with respect to its type and is often explicitly *structured* in data models. When data is stored and retrieved for analysis, it needs to be *transported*. Yet, not all mechanics are similarly relevant in each use-case: For most relational databases, *replication* and *partitioning* play a subordinate role, as they are optimized for single-server usage. In contrast, these principles are central to non-relational NoSQL databases, which are typically focused on fault-tolerant distributed data storage.

In addition, the following design principles can be recognized in the field of data management:

- **Data independence** denotes the abstraction between the internal representation of data and the interface to the outside.
- **Integrity** of data, i. e. ensuring that the data is valid and not distorted.
- **Consistency** of data, i. e. that internal constraints of the data store are not violated.
- **Isolation** ensures that concurrent queries do not interfere with each other.
- **Durability** strives to minimize the risk of data loss.
- **Availability** of data means providing suitable and fast ways to access it.
- **Partition tolerance** makes a system tolerant against failure of parts in a distributed data store.
- **Concurrency** of queries in data management systems and data analysis processes is central for multi-user usage and for distributed systems.
- **Redundancy** of data sets is often needed for ensuring fast access or failure tolerance, but is in contradiction to consistency.

As, for example, *redundancy* and *consistency* show, not all these principles are achievable in one system at the same time: The CAP theorem (cf. [2]) describes that the three design principles "availability", "partition tolerance" and "consistency" cannot be achieved at once. This is because ensuring consistency in distributed data stores involves mechanisms such as transactions or constraints that slow down access to the data store and hence decrease availability. When building data management systems, such design principles must be weighed against each other. But also when characterizing and comparing existing systems, these principles can be used, as systems are especially differing in their realization.

Due to the division of the model into the four categories "core technologies", "practices", "design principles" and "mechanisms", it gives a comprehensive but easy-to-understand overview of the field. This model allows for different interpretations and uses, especially for lesson planning: Not only does it support setting a focus on key concepts of the field, it also includes various points of contacts with other aspects of computer science and helps linking the concepts of data management with other topics of CS. Thus, especially teachers can benefit from using this model.

5 VALIDATION FROM A SCIENTIFIC PERSPECTIVE

As the model characterizes the scientific discipline data management, it was important for us to make sure that it is technically correct in order to ensure a high grade of validity. Thus, we checked its plausibility from a scientific perspective using a semi-structured face-to-face interview with an internationally renowned professor on data management. The professor selected has a strong research background in this field for years, thus we assumed that he has sufficient knowledge for evaluating the validity of the model. In the interview, we focused in particular on the following aspects:

(1) Before presenting the model:
   - How can data management be described from a scientific perspective?
   - Which are the central concepts/principles of this discipline?
(2) After presenting the model and describing the methodological approach:
   - Are central aspects missing in the model?
   - Are there any contradictions or aspects that are wrong?
   - How well is data management characterized by this model?
   - Is the model suitable for describing data management from a scientific perspective?

When we presented our model to the expert, he highly agreed not only with the results but also with the methodological approach. The literature used as basis for the analysis was considered suitable and sufficiently comprehensive for characterizing the field. There were no suggestions on how to improve the methodological approach, e. g. by considering additional material.

Concerning the practices, there was clearly positive feedback, as they were seen as sufficiently concrete to emphasize the typical activities concerning data management. It was discussed whether adding additional practices, such as "assigning access rights", might improve the model. Yet, as these aspects are not specific for data management, such an expansion was not seen as necessary for the purpose of the model.

However, in other parts of the model, some adaptations were suggested and afterwards implemented: Especially, the core technologies "file stores" and "document stores" were added to the core technologies, as they are on the same level as databases and also very relevant to this field. Yet, this adaptation did not change the meaning of the model, as the list of core technologies is only a sample and may be changed over time, as described before. Also the relevance of "transactions" was discussed, as they are on the one hand also considered by other aspects in the model, but on the other hand are central to the field. In this discussion, it was decided to explicitly emphasize them by adding them to the mechanisms of data management.

Additionally, some terms were replaced in order to avoid ambiguity, but without changing the meaning. Hence, although the model was created keeping an educational perspective in mind, the interview has shown agreement from a scientific point-of-view. Overall, our model of key concepts of data management was considered as being suitable for characterizing this field and for being used even in the research area itself.

6 USE AND INTERPRETATION OF THE MODEL

Similar to the Fundamental Ideas of Computer Science or the Great Principles of Computing, our model can be considered from various perspectives. In the following, we describe three examples for using it in different contexts. First, we will regard "data management" trough the lenses of central goals of CS. Then, we will use the model for giving an insight into data management practice by investigating the life-cycle of data. In the third example, we will describe how the model supports lesson design by examining which aspects of data management are emphasized in different topics.

6.1 Representation of Central Goals of Computer Science in the Model of Data Management

In the ACM encyclopedia of computer science, Denning [6] describes that computer science is especially concerned with "considerations such as transparency, usability, dependability, hardware reliability, software reliability, and software safety". While it can be expected that these goals should be represented in all areas of CS, these terms are not shown directly in our model of key concepts. Although they were of course found during the literature analysis, when merging the different terms, they were dropped, because they had a strong overlap with other terms. Thus, in the model, they are represented implicitly. Hence, it was an interesting question to investigate how data management contributes to these aspects and thus to general educative goals of CS. When regarding data management through the lenses of these goals, interesting insight into the field can be gained, which we will illustrate for the examples of safety and usability:

- **Safety**, in general, is the protection of computer systems from failures. In data management, safety is ensured by *integrity* and *persistency* of data and by the *isolation* of queries. Also, safety is concerned with various practices of data management, such as data *modeling*, as in order to ensure integrity, for example, restrictions on data types must be defined. But also during *implementation*, *storage*, *optimization*, *analysis* and *visualization*, safety is highly important and can be achieved through measures such as creating backups of data or using checksums (probably internally in the system). Also, when *sharing* data, it is important to ensure transport safety, and when *erasing* data, safe erasure methods are needed.
- **Usability** is especially concerned with enriching the user experience of an application or system in order to enable users to use it in a simple and efficient way. In data management, usability is achieved by facilitating access to and storage of data, by allowing to use the systems in a flexible way, by abstracting from technical implementation details as well as by high speed and the reliability of these systems. Thus, the design principles *availability* and *consistency* are particularly important. Also, *concurrency* and *isolation* contribute to usability, as the first enables multiple users to use a system at the same time, while the latter ensures that there are no unexpected side-effects. In addition to the principles, usability also plays a central role throughout the whole process of data management and hence during *all the practices* mentioned in the model.

As shown in these examples, when looking at data management through the lens of overall goals of CS, it can be recognized that data management does clearly contribute to these and hence is a central aspect of computer science. As it is addressing goals like "usability" and "safety" from a different perspective, data management can contribute to foster a deeper understanding of these terms and of computer science in general. But also the other way round, understanding typical data management topics can be improved when considering them in relation to the central goals of computer science: For example, safety becomes increasingly relevant when storing and processing large amounts of data in one place.

6.2 Data Management Practices and the Data Life-Cycle

Another interpretation of the model sets its focus on the practices of data management. This focus is particularly helpful for getting insight into how people get in contact with data regularly in private and professional contexts. As these practices describe various activities that are all related to handling data, they can give a clear impression of the life-cycle of data and of the activities of data management professionals, such as data scientists.

Having a deeper look on our list of practices reveals that certain aspects, such as optimization and sharing, are not equally relevant to every use-case of data. Yet, the practices generally are based upon each other and were sorted in the model according to their position

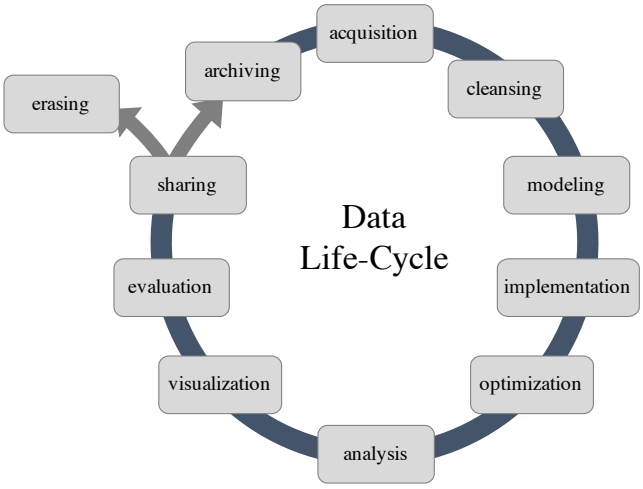

Figure 5: Interpretation of the data management practices as data life-cycle.

in the data life-cycle. Thus, this order can hardly be changed. In fig. 5, we visualized these practices as a life-cycle model.

Although the model has been developed from a CS education perspective and hence did not consider technical aspects in detail, these results are in high accordance with the scientific perspective: In professional contexts, various data life-cycles have already been proposed. For example, Chisholm [4] divides the life-cycle of data into the seven phases *capture, maintenance, synthesis, usage, publication, archival* and *purging*. Comparing these with our practices of data management, shows clear similarities: While the first and last phases of the life-cycle are referring to the same activities, the *maintenance* phase as Chisholm describes it, corresponds to the practices *cleansing, modeling, implementation, storage* and *optimization*. The *synthesis* is mainly focused on drawing logical conclusions from the data and hence represents the *analysis* aspect, while *usage* considers all further use-cases of data and of analysis results, in our model depicted by *visualization* and *evaluation*.

The resulting life-cycle of data can be used for various purposes: It gives insight into handling and managing data, can be used for contextualizing teaching lessons and for emphasizing the importance of concrete practices or even for raising students' awareness that data is not only stored, but typically also analyzed, evaluated or archived for longer times. In addition, as the comparison with the data life-cycle from a professional perspective has shown, considering the practices as data life-cycle also confirms the validity of this part of the model.

6.3 Comparing Different Topics of CS Education

A third interpretation is using the model as tool for developing teaching lessons by examining which aspects of the subject area are related to a concrete topic. For example, the topic *data stream systems* (cf. [15]) particularly emphasizes the real-time processing of ever-growing data sets. Such analyses are becoming increasingly relevant not only to CS but also in our daily lives. In particular, real-time data analyses are suitable for emphasizing how the limits of data management have shifted in the last years in comparison to traditional analyses that typically took much more time. When regarding this topic trough the lens of the key concepts of data management, the following aspects are emphasized:

Core technology:	data flow systems
Practices:	acquisition, cleansing, analysis
Design principles:	availability, concurrency
Mechanics:	structurization, representation, transportation

This consideration obviously shows that the topic "data stream systems" only covers a relatively small section of the entire subject area. While this topic emphasizes aspects that gained in importance in recent years, traditional topics like data modeling or integrity are only considered marginally, if at all.

These aspects are clearly different from those central to current computer science teaching in secondary schools. In the context of data, current teaching in particular sets its focus on databases and data modeling [13]. Therefore, according to an analysis of several curricula for secondary CS education, the following aspects of data management are considered:

Core technology:	database systems
Practices:	modeling, implementation, optimization, analysis
Design principles:	data independancy, integrity, consistency,redundancy
Mechanics:	structurization, representation, transaction

This emphasizes the focus of the topic on structured storage and related practices and principles.

Both topics, databases and data stream systems, are exemplary for further data management issues and can clearly show the diversity of the area. Just as it is essential for the topic *databases* to address the described principles and practices, the further discussion in CS education research has to determine whether CS education needs to cope with aspects currently not part of (secondary school) teaching. From our point of view, understanding these concepts clearly fosters important competencies for being part in the digital society and for living in it in a self-determined way.

7 CONCLUSION

When comparing the issues that were regarded as central to databases by Lockemann [20] in 1986, consistency, persistence and concurrency, to our model of key concepts of data management, the enormous developments in recent years become obvious: While the three aspects mentioned by him are also covered by our model, several additional concepts have been added. Compared with the Great Principles of Computing, in some ways our model is a specialization of Denning's model: For example, the design principle "security" in the Great Principles of Computing is split up into integrity, availability, and isolation in our model. Another principle, coordination, shows clear overlaps with the replication of data in our model. Additional similarities can be found in the stories that Denning uses for describing the principle "recollection": For example, Denning mentions storage systems, fast and persistent

storage, access to stored data and auxiliary data structures such as caches [9]. All these aspects are also considered in the model of key concepts of data management. When regarding the criteria that Schwill suggests for fundamental ideas of computer science, the principles listed in our model may also be candidates for fundamental ideas: On the one hand, they are relevant in various areas of data management, but often also beyond it in CS in general. Thus, they fulfill the horizontal criterion. Also the vertical criterion seems to be fulfilled, as we have shown, for example, by a teaching example on data stream systems [15], in which this topic is prepared in a way that secondary school students can understand the basic concepts and ideas of this topic that is typically part of university education. Because of the strong relevance of all aspects of data management mentioned in the model, the criterion of sense is addressed, and, due to the long-lasting relevance of these aspects (cf. e. g. [20]) and their clearly accepted relevance in the future, the criterion of time is also satisfied. Nevertheless, the character of our key concepts as presented before differs from the ideas presented by Schwill: While the latter represent concrete prototypes from which CS phenomena arise, the principles are located on a more concrete level and describe the laws and functions in data management.

While nowadays, CS teaching in secondary schools is especially focused on databases [13], the developed model clearly shows the diversity of the subject area. When using this model for curriculum and lesson development, as well as for further research in this area, the gap between the requirements of everyday life and the representation of data management topics in school can be addressed. As shown by the three exemplary ways of interpretation, the model of key concepts of data management can function as a versatile tool in this area. The examples on databases and data stream systems show that the model is suitable for characterizing topics of data management and the diversity of this field in a structured way, and it can support getting an overview of their central principles.

When looking back at the reasons that Denning describes for why his great principles framework is interesting [8], it can also be recognized that the same reasons apply for the model of key concepts of data management:

- *Understandability* is raised by emphasizing the key concepts of data management, which strongly supports getting an overview of the innovative and rapidly growing field data management.
- The model strongly *emphasizes the practices* in data management as well as the data life-cycle and thus gives a clear impression of professional data usage.
- The model helps *shaping the public image* of computer science in general and data management in particular, as it shows the broadness of CS as a field by setting its focus on other aspects than typically considered as being most-central to it (such as algorithms and programming).

Thus, the presented empirical approach for determining the key concepts of a field was successfully applied to data management and also seems promising for other areas.

REFERENCES

[1] Tim Bell, Paul Tymann, and Amiram Yehudai. 2011. The Big Ideas of K-12 Computer Science Education . http://cosc.canterbury.ac.nz/research/RG/CSE/big-ideas/BigIdeas-webdocument-7-May-2011.pdf. (2011).

[2] Eric A. Brewer. 2012. CAP twelve years later: How the "rules" have changed. *Computer* 45, 2 (2012), 23–29.

[3] Jerome S. Bruner. 1960. *The Process of Education.* Harvard University Press.

[4] Malcolm Chisholm. 2015. 7 Phases of A Data Life Cycle. *Information Management* (2015). https://www.information-management.com/news/7-phases-of-a-data-life-cycle

[5] DAMA International. 2009. *The DAMA Guide to the Data Management Body of Knowledge - DAMA-DMBOK.* Technics Publications, LLC, USA.

[6] Peter J. Denning. Computer Science. In *Encyclopedia of Computer Science.* John Wiley and Sons Ltd., Chichester, UK, 405–419.

[7] Peter J. Denning. 2003. Great Principles of Computing. *Commun. ACM* 46, 11 (2003), 15–20.

[8] Peter J. Denning. 2004. Great Principles in Computing Curricula. *SIGCSE Bull.* 36, 1 (March 2004), 336–341.

[9] Peter J. Denning. 2010. The Great Principles of Computing. *American Scientist* 98, 5 (2010).

[10] Kerstin Strecker Eckart Modrow. 2016. *Didaktik der Informatik [Didactics of Computer Science].* Oldenbourg.

[11] Stefan Edlich, Achim Friedland, Jens Hampe, Benjamin Brauer, and Markus Brückner. 2011. *NoSQL.* Hanser Fachbuchverlag.

[12] Ramez Elmasri and Shamkant B. Navathe. 2011. *Fundamentals of Database Systems.* ADDISON WESLEY Publishing Company Incorporated. http://books.google.de/books?id=ZdhAQgAACAAJ

[13] Andreas Grillenberger and Ralf Romeike. 2014. A Comparison of the Field Data Management and its Representation in Secondary CS Curricula. In *Proceedings of WiPSCE 2014.* ACM, Berlin.

[14] Andreas Grillenberger and Ralf Romeike. 2014. Teaching Data Management: Key Competencies and Opportunities. In *Proceedings of KEYCIT 2014*, Torsten Brinda, Nicholas Reynolds, and Ralf Romeike (Eds.). Universitätsverlag Potsdam.

[15] Andreas Grillenberger and Ralf Romeike. 2015. Analyzing the Twitter Data Stream Using the Snap! Learning Environment. In *Proceedings of ISSEP 2015.* Springer International Publishing. DOI:https://doi.org/10.1007/978-3-319-09958-3_4

[16] Hans Werner Heymann. 2014. *Why Teach Mathematics?* Springer.

[17] Alfons Kemper and André Eickler. 2015. *Datenbanksysteme [Database Systems].* Gruyter, Walter de GmbH.

[18] Thomas Kudraß. 2015. *Taschenbuch Datenbanken [Paperback Databases].* Hanser Fachbuchverlag.

[19] Douglas Laney. 2001. *3D Data Management: Controlling Data Volume, Velocity, and Variety.* Technical Report. META Group. http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf

[20] Peter C. Lockemann. 1986. Konsistenz, Konkurrenz, Persistenz - Grundbegriffe der Informatik? - Zur Diskussion gestellt [Consistency, Concurrency, Persistency – Basic Terms of Computer Science? – For discussion]. *Informatik Spektrum* 9, 5 (1986).

[21] Philipp Mayring. 2004. Qualitative Content Analysis. (2004), 266–269.

[22] Eckart Modrow. 2003. Fundamentale Ideen der theoretischen Informatik. [Fundamental Ideas of Theoretical Computer Science]. In *Informatische Fachkonzepte im Unterricht, INFOS 2003.* 189–200.

[23] Lothar Piepmeyer. 2011. *Grundkurs Datenbanksysteme [Basic Course Database Systems].* Hanser Fachbuchverlag.

[24] Andreas Schwill. 1994. Fundamental ideas of computer science. *Bull. European Assoc. for Theoretical Computer Science* 53 (1994).

[25] Rainer Unland and Günther Pernul. 2014. *Datenbanken im Einsatz [Databases in Practice].* de Gruyter, Oldenbourg.

[26] Andreas Zendler and Christian Spannagel. 2008. Empirical Foundation of Central Concepts for Computer Science Education. *ACM Journal of Educational Resources in Computing* 8, 2 (2008), 6:1–6:15.

[27] Andreas Zendler, Christian Spannagel, and Dieter Klaudt. 2008. Process as content in computer science education: empirical determination of central processes. *Computer Science Education* 18, 4 (2008), 231–245.