TECHNISCHE UNIVERSITÄT DRESDEN

Query-Time Data Integration

Kurzfassung der Dissertation

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

vorgelegt an der
Technischen Universität Dresden
Fakultät Informatik

eingereicht von
Dipl. Medien-Inf. Julian Eberius
geboren am 28. September 1987 in Dresden

Betreuender Hochschullehrer: **Prof. Dr. Wolfgang Lehner**
Technische Universität Dresden
Fakultät Informatik, Institut für Systemarchitektur
Lehrstuhl für Datenbanken
01062 Dresden

Dresden, im Oktober 2015

1 INTRODUCTION

While the term *Big Data* is most often associated with the challenges and opportunities of today's growth in data volume and velocity, the phenomenon is also characterized by the increasing *variety* of data (Laney, 2001). In fact, data is collected in more and more different forms, and from increasingly heterogeneous sources. The spectrum of additional data sources ranges from large-scale sensor networks, over measurements from mobile clients or industrial machinery, to the log- and click-streams of ever more complex software architectures and applications. In addition, there is more publicly available data, such as social network data, as well as Web- and Open Data. More and more organizations strife to efficiently harness all forms and sources of data in their analysis projects to gain new insights, or enable new features in their products. While generating value out of these novel forms of data is not trivial and requires new methods, algorithms, and tools, their potential is universally recognized (Labrinidis and Jagadish, 2012).

New opportunities always come with their own set of novel challenges. In the case of Big Data they comprise the entire data life cycle, from acquisition to interpretation, and introduce a wide range of new problems, including issues of scale, timeliness, privacy and visualization, just to name a few (Jagadish et al., 2014). One specific challenge related to the *variety* aspect of Big Data, or in other words, the heterogeneity and multitude of data sources utilized today, is *data integration*. In this thesis, we focus on this data integration aspect, especially in relation to changing data management practices. These include analytical processes that concern a larger variety of non-traditional data users and contexts, that are based on ad-hoc information needs, and are not easily mapped to traditional analytical architectures.

Data integration is a common problem in data management, which deals with combining data of different origins and making it usable in a unified form. In general, it is a laborious and mostly manual process that has to be performed ahead-of-time, i.e., before queries on the combined data can be issued. Due to its complexity, it is usually performed by experts, for example ETL and integration specialists. In the era of Big Data, with an increasing variety of data sources, the traditional challenges in data integration, that is to say bridging the heterogeneity and ambiguity between data sources at schema- as well as instance level, are only becoming more complicated (Dong and Srivastava, 2013).

At the same time, data-driven approaches are applied in increasing numbers of contexts, involving more and more user groups outside of traditional IT. Novel *agile* approaches to data management, such as MAD (Cohen et al., 2009), are complementing or replacing the static processes of data warehouse infrastructures. It is increasingly recognized that organizations profit from enabling domain experts to perform their own data management and analysis tasks without or with less involvement of IT personnel (McAfee and Brynjolfsson, 2012). This direct access to data and methods is especially critical because these user's information needs are often *ad-hoc* or *situational* (Marchionini, 2006), or require the use of heterogeneous or unstructured data that is not integrated in a data warehouse. The title "data scientist" is becoming a standard term for highly trained data professionals that perform these kinds of agile analytics as a contrast to the more traditional "business analyst", who is mainly concerned with classical BI (Davenport and Patil, 2012). In addition, a new class of users, sometimes called *data enthusiasts* (Morton et al., 2014), is becoming increasingly prominent. These users also want to utilize data to support decisions or illustrate a story, but are lacking in formal education that is

the hallmark of the data scientist. They are another driving factor towards the development of *self-service analytics and integration*.

However, conventional data infrastructures assume controlled ETL processes with well-defined data sources and target schemata, that define the data pipelines in the organization. These schemata also define what is immediately queryable for an analyst. If there is an ad-hoc information need that can not be satisfied with the current schema because external information has to be integrated, an intricate process has to be followed. Because the warehouse is a crucial piece of infrastructure it is highly controlled: ad-hoc integration is not a feature that it is designed for. Furthermore, new datasets are now often collected at a rate that surpasses the organizational capabilities to integrate them with a central schema. In many cases instant integration is not even the desirable, as the future use cases of the data is not known.

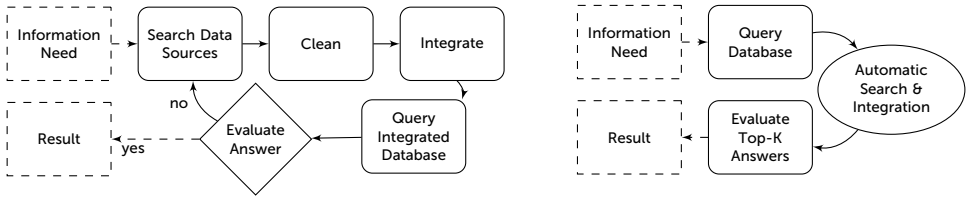
So while a new wealth of data is available, the integration of a large variety of sources is still a complicated, laborious and mostly manual process that has to be performed by experts, that is required before queries on the combined data can be issued. This obviously collides with the aim of enabling domain users to generate value from Big Data directly with little IT support. It is well established in traditional warehousing that the majority of the effort in a complex analysis project is actually the integration of heterogeneous data sources (Dasu and Johnson, 2003), which has to be performed before any actual analytical methods can be applied. Without additional tool support, the effort of data integration will likely prevent those users from taking advantage of the wealth of possible data sources available today.

Put succinctly, we identify two trends: first, we observe an *increasing availability of valuable, but heterogeneous data sources*, and second, there is an *growing demand for self-service BI and ad-hoc integration*, driven by the increasing number of different user groups and usage contexts for data. So while the first trend enables more and more situations in which data can potentially be enhanced and enriched through integration with external data sources, it also introduces additional complexity. The second trend, however, demands that the tools and processes of data integration become simpler, and able to cater to a larger audience.

1.1 Query-time Data Integration

This thesis aims at aligning the mentioned trends, increasing *availability of heterogeneous data sources* and increasing demand for *self-service, ad-hoc integration*. This entails enabling users to tap the power of vast amounts of diverse data sources for their analysis tasks. Specifically, in this thesis we introduce methods and systems aimed at fulfilling the following requirements:

Exploratory and Ad-hoc Data Search and Integration The volume and variety of data sources in the mentioned scenarios makes single, centralized integration effort infeasible. Instead of consolidating all available sources as soon as they are available, sources should be retrieved and integrated based on a specific information need. In such a scenario, data search takes the place of exhaustive schemata covering all available information. Throughout this thesis, we discuss this argument in detail, and show how exploratory analysis over large numbers of heterogeneous sources can be supported with the right methods and systems.



(a) Regular Ahead-of-time integration

(b) Top-k Query-time Data Integration

Figure 1: Alternative methods for ad-hoc queries based on external data

Minimal Up-front User Effort When developing methods for answering ad-hoc information needs, an important factor is the minimization of up-front costs for the user. In other words, our ambition is to minimize the user effort necessary before first results become available. Specifically, we want to reduce the reliance on IT personnel and integration experts, or the need to search data repositories or master integration tools before external sources can be included in a data analysis scenario. In the ideal case, a user should be able to declaratively specify the missing data and be presented with workable answers automatically.

Trustworthy Automated Integration The two goals introduced above amount to reducing user effort and time-to-result in data search and integration. As a consequence, this thesis proposes novel methods for automating these highly involved processes. However, exact data integration has been called an "AI-complete" problem (Halevy et al., 2006), and is generally considered not to be automatically solvable in the general case. In fact, all proposed methods return results with varying confidence values, instead of perfect answers, requiring human validation in many cases. In this thesis, we therefore introduce methods that automatize the data search and integration processes as far as possible, while also facilitating user understanding and verification of the produced results.

System Integration Finally, ad-hoc integration queries should not introduce an isolated, new class of systems into existing data management architectures. As discussed above, the wealth of new data sources collected by organizations or found on the Web promises opportunities. However, most data analysis tasks will still focus on the core databases inside organizations, with ad-hoc integrated sources supplementing, not replacing, them. Therefore, methods developed to support ad-hoc integration can not be deployed in a vacuum, but need to work hand in hand with systems managing core data.

1.2 Query Model

We will now introduce our notion of *Query-time Data Integration*. Consider Figure 1a, which gives a high-level overview of a manual process for an ad-hoc database query that depends on data from yet unknown external sources. Before the user can pose the actual query, data sources that contain relevant data for integration need to be identified manually, for example through a regular search engine. Then, the data has to be extracted and cleaned, i.e., converted

into a form that is usable in a regular database. In a next step, it needs to be integrated with the existing database, which includes mapping of corresponding concepts, but may also include instance-level transformations. Only after this process is finished, the original query can be issued. Still, the result may not be what the user originally desired, or the user may want to see the query result when using a different Web data source. In this case, another iteration of the process is necessary.

In this thesis, we propose *Query-time Data Integration* as an alternative concept to ahead-of-time data integration. We aim at allowing a user to issue ad-hoc queries on a database while referencing data to be integrated as if it were already defined in the database, without providing specific sources or mappings. We call such database queries *Open World Queries*, since they are defined over data not contained in the database. The goal is to enable users to specify information needs requiring external data declaratively, just as if only local data was used, without having to integrate data up-front. The database system then automatizes the process of retrieving and integrating possible Web data sources that could be used to answer these queries.

However, a fully automatic retrieval and integration process is not feasible. The underlying methods, information retrieval and automatic matching, both work with top-k results or provide uncertain answers with confidence values. We argue that database query results based on such automatic methods should therefore also reflect the uncertainty of the basic methods, as well as the ambiguity of the query and the uncertainty of the data sources themselves.

To cope with these various sources of uncertainty, we propose to extend the concept of providing top-k alternative answers, well known from general search engines, to structured database queries. Under this paradigm, the system will not respond to Open World Queries with a single, perfect result, as it would be the case with normal database queries. Instead, it should produce a ranked list of possible answers, each based on different possible data sources and query interpretations. The user can then pick the result most suitable to his or her information need. This alternative process is depicted in Figure 1b.

1.3 Architecture and Thesis Contributions

This section introduces our proposed architecture for a query-time integration architecture, depicted in Figure 2. It also serves as the outline for this thesis, and gives a high-level overview of our contributions. Our proposed architecture tackles the problem of ad-hoc integration of heterogeneous sources from several sides. We assume a large collection of external data sources, i.e., a pool of potentially useful but independent and heterogeneous sources, that could be used to fulfill ad-hoc information needs. For the purpose of this high-level architecture, we do not make many assumptions about the nature of this corpus. For example, it could be a corpus of Web tables, i.e., data-carrying tables extracted from the open Web, a collection of datasets published on a Open Data platform, or a corporate *data lake* (Mohanty et al., 2015).

To capitalize on such a loosely coupled corpus of data sources, we propose three systems that complement each other, and together enable our vision of Query-time Data Integration. The first is a system, called *REA*, enables *top-k entity augmentation*, which forms the basic building block for ad-hoc data integration. The second system, called *DrillBeyond*, is an extended RDBMS which can process Open World SQL queries, i.e., queries referencing additional attributes that are not defined in the database schema. The third component is a data curation

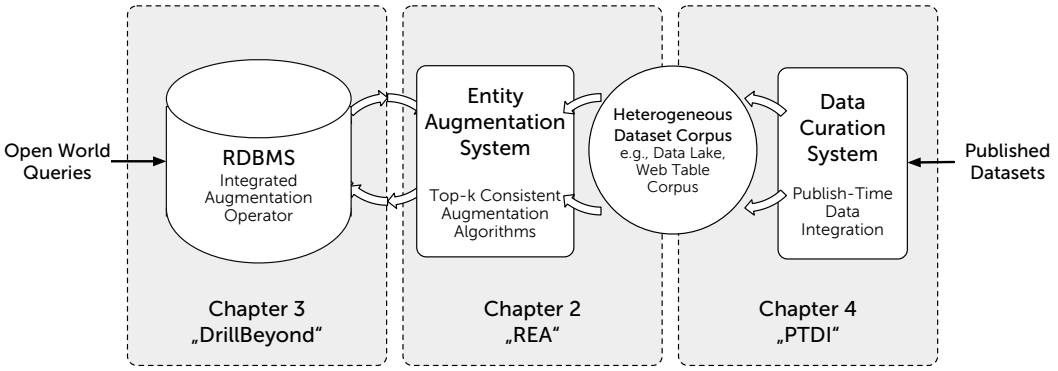


Figure 2: Query-time data integration architecture and thesis structure

system based on the novel *Publish-time Data Integration* paradigm for managing heterogeneous dataset collections.

The novel methods and algorithms introduced in these three systems form the main contributions of this thesis. In the thesis, they are introduced in three chapters, which we will sketch in the following.

2 TOP-K ENTITY AUGMENTATION

In this chapter of the thesis, we introduce novel algorithms for producing consistent integration results from a large corpus of possible data sources. Specifically, we discuss the *entity augmentation* problem, which aims at extending a given set of entities with an additional, user-requested attribute that is not yet defined for them. This attribute is typically materialized by automatically retrieving and integrating relevant data sources. As an example, consider Figure 3, where the query is represented as a table on the top, and the available candidate data sources below it. The query table consists of a set of five companies, and the augmentation attribute "*revenue*". The candidate data sources depicted below the query table vary in coverage of the query domain, the exact attribute they provide, and their context. The task is to fill the missing attribute of the query table with values using the set of heterogeneous and overlapping candidate sources.

In recent related work, several systems that process such queries on the basis of large Web table corpora have been proposed, for example *InfoGather* (Yakout et al., 2012; Zhang and Chakrabarti, 2013), the *WWT* system (Pimplikar and Sarawagi, 2012; Sarawagi and Chakrabarti, 2014), and the *Mannheim Search Join Engine* (Lehmberg et al., 2015). However, these existing methods create single results by choosing augmentation values on a per-entity basis, with limited respect to the consistency and number of data sources used. In the thesis, we show how this leads to weaknesses with respect to problems such as *Attribute Variations*, *Unclear User Intent*, *Trust & Lineage*, *Exploratory Search* and *Error Tolerance*.

To alleviate these weaknesses, our approach aims at creating *consistent* and *relevant* solutions from a *minimal* number of sources, and answering augmentation queries with *diversified top-k* results instead of providing a single answer. To this end, we present an extended ver-

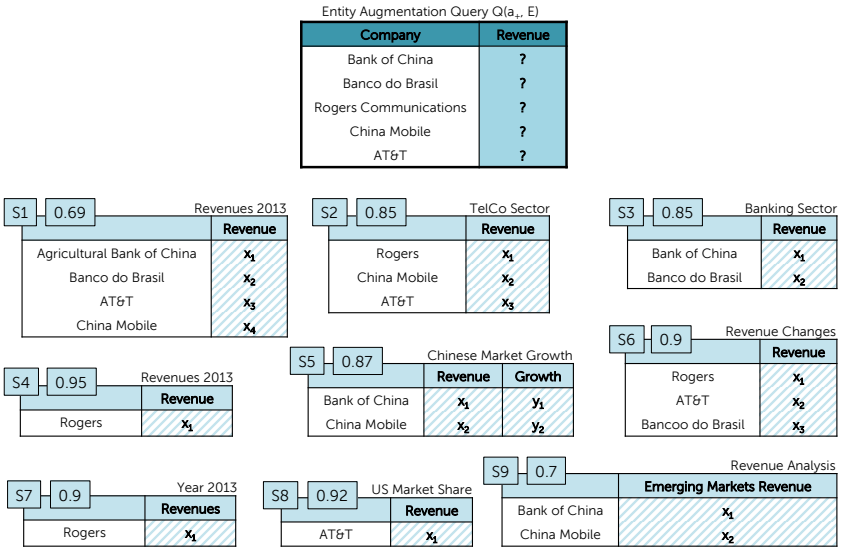


Figure 3: Example augmentation scenario: query table and candidate data sources

sion of the Set Cover problem, called *Top-k Consistent Set Covering*, onto which we map our requirements. We introduce a core framework for solving top-k consistent set covering, which allows us to jointly optimize for those four problem dimensions. As a baseline, we propose a basic greedy algorithm, modeled after the corresponding classic Set Cover algorithm. It picks consistent data sources to construct individual augmentations, while maximizing the diversity between the results generated in several runs of the algorithm, to provide meaningful alternative top-k solutions. Then we extend this algorithm with an approach that explores a larger part of the search space, and adds a selection phase to filter the larger number of solutions that are created. Finally, we map the problem into the domain of evolutionary algorithms, to reach a more sophisticated genetic algorithm, that naturally models the creation of a diverse set of individually strong solutions, i.e., solutions that are composed of a minimal number of relevant and consistent data sources.

To illustrate the advantages of our approach, let us consider the problem of *attribute variations* mentioned above. Even for the seemingly simple "*revenue*" query shown in Figure 3, the real-world concept is complex, with many variants such as "US revenue" or "emerging markets revenue" and derived attributes such as "revenue growth". Furthermore, many types of values come with temporal or spatial constraints, such as different years of validity, or may have been measured in different ways, for example using different currencies. While most related work assumes a single-truth answer is sufficient, our approach returns multiple, diversified solutions. Through the use of consistent top-k set covering, our approach ensures that individual solutions are created from consistent sources, i.e., sources expressing the same attribute variation. Our minimality requirement creates solutions that are easier to understand and verify, while explicitly modeling source similarity gives us a way to not only create more consistent

```
select
    n_name,
    avg(o_totalprice)
from nation, customer, orders
where
    n_nationkey=c_nationkey
    and c_custkey=o_custkey
group by n_name
```

(a) Initial query on local data

```
select
    nation.creditRating,
    avg(o_totalprice)
from nation, customer, orders
where
    n_nationkey=c_nationkey
    and c_custkey=o_custkey
    and nation.gdp > 10.0
group by nation.creditRating
```

(b) Extended query with ad-hoc attributes

Figure 4: Exemplary analysis queries illustrating an ad-hoc need for external data

results, but also to create useful alternative solutions. Finally, the top-k paradigm alleviates many of the weaknesses stemming from the uncertainty of Web tables as data sources, ambiguous queries, and errors from fully automatic data search- and matching techniques, by allowing users to choose the most fitting solution to their query.

We implemented our novel algorithms in a new Web table retrieval and matching system called REA, and evaluate the system using the DWTC, a corpus of 125 million Web tables which we extracted from a public Web crawl and made available to the research community. In this evaluation, we measure the effects of our proposed algorithms on the base measures precision, coverage and runtime, but also on new dimensions consistency, minimality and diversity of the top-k query results. Our experiments show that our genetic set covering-based approach improves both consistency and minimality of the results significantly, without loss of precision or coverage, and while producing a diverse set of results for the user to choose from.

3 OPEN-WORLD SQL QUERIES

So far we only studied entity augmentation in an isolated context. The scenarios we studied dealt with only one single query table that was the sole focus of the user's information need. We also implemented the augmentation process in a standalone system. However, it is natural to assume that ad-hoc data integration will be most useful in analytical scenarios, in which the user works with complex databases, and the augmentation query is only one step in a chain of analytical operations. In such a scenario, the data already resides in a DBMS, most likely a traditional relational system. The user would not be interested in the entity augmentation per-se, but would require the integration of additional attributes in the context of a larger analytical task. Consider a user in a traditional warehousing environment, analyzing sales totals by country, as illustrated through the SQL query shown in Figure 4a. As a next step, the user may want to add external context to gain insight into the aggregated sales numbers. For example, it may be beneficial to introduce some global economic context into the analysis, such as the countries' gross domestic product (GDP) or credit rating. However, those attributes are not defined in the local warehouse schema, which necessitates the integration of external sources. To fulfill these kinds of ad-hoc information needs in classic warehouse settings, the analyst has to perform a multitude of complex steps and adhere to strict policies before new data can be

integrated with the warehouse schema (Cohen et al., 2009). Ideally however, the user would be able to tackle this ad-hoc integration scenario in the same way he would tackle the analysis of the purely-local data: by using a single, declarative query. This is illustrated in the extended SQL query shown in Figure 4b.

In this thesis, we propose a RDBMS/IR system hybrid system called *DrillBeyond*, that allows querying and analyzing a regular relational database while using additional attributes not defined in said database. Using the methods developed previously in the thesis, the original database is then augmented at query time with Web data sources providing those attributes. To this end, our system tightly integrates regular relational processing with a novel data retrieval and integration operator that encapsulates our augmentation techniques. Specifically, in this chapter of the thesis, we discuss issues regarding placement of this operator in a query plan with respect to runtime and answer quality, and propose a cost model as well as plan- and runtime optimization rules. Further, we detail how to efficiently process a query multiple times based on different entity augmentations, to transfer the advantages of top-k augmentation discussed in Section 2 to the world of relational query processing. Our core idea is to optimize query plans with respect to invariant subtrees, i.e., parts of the query plan that produce the same result when using different augmentations. This enables us to create plans that may be less efficient in a single execution, but are more efficient over several execution because of their higher caching potential.

We also explore how the entity augmentation process itself can be improved through information available in the context of an SQL query. For example, query predicates on the ad-hoc attributes can be used to guide the search for candidate Web tables comparing the predicate data types and values to the data found in the data sources. Finally, we introduce a practical implementation of the operator and our optimizations in PostgreSQL, which allows us to meaningfully evaluate the operator on standard test databases and fully-featured SQL queries. Our evaluation demonstrates the effectiveness of our optimizations in minimizing the runtime overhead of producing multiple SQL query results based on alternative augmentations. Finally, it also shows that pushing SQL query context into the augmentation system can improve quality and performance of the augmentation processing, when compared to standalone processing.

To summarize, our DrillBeyond system, with its tight integration of augmentation and relational query processing and its various optimizations, enables the use of ad-hoc data search and integration in new contexts, and greatly increase the practicality of the augmentation methods we introduced in Section 2.

4 PUBLISH-TIME DATA INTEGRATION

As introduced in Section 1, we assume a large collection of heterogeneous, individual datasets as the data source for our augmentation techniques. In the last chapter of this thesis, we discuss challenges of managing such collections, and introduce *data curation systems* (Stonebraker et al., 2013) as a complementary system type to database and data warehouses. We first discuss examples for such systems, including Open Data platforms, scientific data repositories, and enterprise data lakes. These platforms are characterized by the fact that data is stored there mostly in its raw form, as provided by source systems or human publishers, and is not orga-

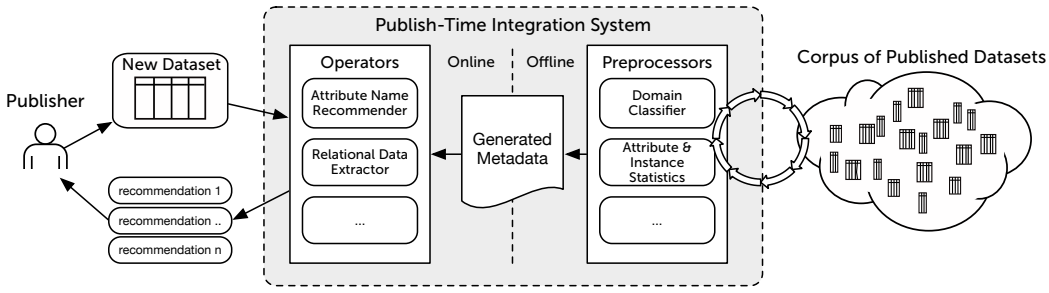


Figure 5: Architecture of a PTDI Platform

nized with a central schema. The motivation is to store all incoming data and make it available, even if the future use case is not always known.

We surveyed a large set of public Open Data platforms to gain real-world insights into usage patterns and problems of this new form of data management. The survey results showed that those platforms, while containing vast amounts of potentially reusable data, do not support reuse as well as they could. Especially automated ad-hoc discovery and integration of Open Data, as we aim for in this thesis, is hindered by the lack of unified metadata or even standardized data and file formats. Since traditional full integration is not applicable on such platforms, the full effort of searching through, cleaning and integrating the data falls to the future reuser of published datasets. This view of data curation systems is similar to the concept of dataspaces (Franklin et al., 2005). The current philosophy in working with such dataspaces is the so-called pay-as-you-go approach (Madhavan et al., 2007), in which integration is postponed until it is clear how the data should be reused. Still, we argue that some integration tasks can and should be tackled a priori, to improve the usefulness of data curation systems as a whole. Therefore, we propose the *Publish-time Data Integration* (PTDI) paradigm. Under this paradigm, the publisher of a dataset is encouraged to optimize the reusability of newly published dataset through automatically generated recommendations, which just have to be verified by the publisher.

We introduce the concept of *PTDI operators*, components which are triggered when a new dataset is published, and which generate various types of recommendations to improve the dataset's reusability. The runtime-generation of recommendations is supported by metadata about the existing datasets in the system, that is collected by *PTDI preprocessors*. Figure 5 gives an overview of this architecture. We then introduced two specific PTDI operators, one for *generating attribute name recommendations* and one for *extracting relational data from partially structured documents* such as spreadsheets or HTML.

The idea of the first operator is to use statistics on attribute names usage with the datasets already existing in the data curation system, to recommend attribute names that fit well with the vocabulary in use in the system. The motivation for this operator is to constrain heterogeneity, without using a global schema. The generated recommendations are based on instance set overlap as well as a preceding domain classification, in which datasets in the data curation system are automatically clustered according to their schema to identify related datasets. We evaluate the method with respect to run-time, number of recommendations and their precision on real-world Open Data, and use crowdsourcing to evaluate the quality of our recommendations.

The second operator, called *DeExcelerator*, is aimed at transforming partially structured documents, i.e., document in which structured data is freely intermingled with textual or layout elements, into pure relational data with accompanying metadata. We analyze a collection of real-world Open Data spreadsheets to identify a set of typical spreadsheet *denormalizations*, which are usage patterns that hinder automatic reuse of the data contained in the document. Based on these, we propose a pipeline of abstract operators that successively remove these denormalizations and transform the spreadsheet. We evaluate the relevance of the denormalizations we identified as well as the correctness of the generated transformations on real world spreadsheet by means of a user study.

From our experiments, we conclude that the methods introduced in this chapter allow to greatly increase reusability of data published in data curation systems, without changing their free-for-all nature, and while requiring minimal user effort.

5 CONCLUSION

In the era of Big Data, the number and variety of data sources is increasing every day. However, not all of this new data is available in well-structured databases or warehouses. Rather, data is collected at a rate that often precludes traditional integration with ETL processes and global schemata. Instead, heterogeneous collections of individual datasets are becoming more prevalent, both inside enterprises in the form of *data lakes*, and in public spaces such as Open Data repositories. This new wealth of data, though not integrated, has enormous potential for generating value in situational or ad-hoc analysis processes, which are becoming more common with increasingly agile data management practices. However, in today's database management systems there is a lack of support for ad-hoc data integration of such heterogeneous data sources. Instead, integration of new sources into existing data management landscapes is a laborious process that has to be performed ahead-of-time, i.e., before queries on the combined data can be issued.

In this thesis, we introduced the *Query-time Data Integration* paradigm as an alternative concept. It aims at enabling users to express queries on their own data as if all potential other data sources were already integrated, without declaring specific sources and mappings to use. Relevant sources are then automatically retrieved and integrated at query processing-time. The ambiguity resulting from the coarse query specification, as well as the uncertainty introduced by relying on automatically integrated data is compensated by returning a ranked list of possible results. To achieve this goal, we developed and evaluated several new methods, algorithms and systems. Firstly, we introduced a novel method for *Top-k Entity Augmentation*, which is able to construct a top-k list of consistent integration results from a large corpus of heterogeneous data sources. This technique forms the basis for our *DrillBeyond* system, which is able to process Open World SQL queries, i.e., queries referencing arbitrary attributes not defined in the queried database. Finally, we introduced *Publish-time Data Integration* as a new technique for data curation systems, which aims at improving the individual reusability of datasets without forcing ahead-of-time global integration.

BIBLIOGRAPHY

Jeffrey Cohen, Brian Dolan, Mark Dunlap, Joseph M. Hellerstein, and Caleb Welton. MAD skills: new analysis practices for big data. *Proc. VLDB Endow.*, 2:1481–1492, August 2009.

Tamraparni Dasu and Theodore Johnson. *Exploratory Data Mining and Data Cleaning*. John Wiley & Sons, Inc., New York, NY, USA, 1 edition, 2003.

Thomas H Davenport and DJ Patil. Data scientist: the sexiest job of the 21st century. *Harvard business review*, 90(10):70—6, 128, October 2012.

Xin Luna Dong and Divesh Srivastava. Big data integration. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 1245–1248. IEEE, 2013.

Michael Franklin, Alon Halevy, and David Maier. From databases to dataspaces: a new abstraction for information management. *SIGMOD Rec.*, 34:27–33, December 2005.

Alon Halevy, Anand Rajaraman, and Joann Ordille. Data Integration: The Teenage Years. In *Proceedings of the 32Nd International Conference on Very Large Data Bases*, VLDB '06, pages 9–16. VLDB Endowment, 2006.

H. V. Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M. Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big Data and Its Technical Challenges. *Commun. ACM*, 57(7):86–94, July 2014.

Alexandros Labrinidis and H. V. Jagadish. Challenges and Opportunities with Big Data. *Proc. VLDB Endow.*, 5(12):2032–2033, August 2012.

Doug Laney. 3D data management: Controlling data volume, velocity and variety. *META Group Research Note*, 6:70, 2001.

Oliver Lehmberg, Dominique Ritze, Petar Ristoski, Robert Meusel, Heiko Paulheim, and Christian Bizer. The Mannheim Search Join Engine. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2015.

Jayant Madhavan, Shawn R. Jeffery, Shirley Cohen, Xin (luna Dong, David Ko, Cong Yu, Alon Halevy, and Google Inc. Web-scale Data Integration: You Can Only Afford to Pay As You Go. In *In Proc. of CIDR-07*, 2007.

Gary Marchionini. Exploratory Search: From Finding to Understanding. *Commun. ACM*, 49 (4):41–46, April 2006.

Andrew McAfee and Erik Brynjolfsson. Big data: the management revolution. *Harvard business review*, 90(10):60—6, 68, 128, October 2012.

Hrushikesha Mohanty, Prachet Bhuyan, and Deepak Chenthati. *Big Data: A Primer*. Springer India, 2015.

Kristi Morton, Magdalena Balazinska, Dan Grossman, and Jock Mackinlay. Support the Data Enthusiast: Challenges for Next-generation Data-analysis Systems. *Proc. VLDB Endow.*, 7(6): 453–456, February 2014.

Rakesh Pimplikar and Sunita Sarawagi. Answering Table Queries on the Web using Column Keywords. In *Proc. of the 36th Int'l Conference on Very Large Databases (VLDB)*, 2012.

Sunita Sarawagi and Soumen Chakrabarti. Open-domain Quantity Queries on Web Tables: Annotation, Response, and Consensus Models. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 711–720, New York, NY, USA, 2014. ACM.

Michael Stonebraker, George Beskales, Alexander Pagan, Daniel Bruckner, Mitch Cherniack, Shan Xu, Verisk Analytics, Ihab F. Ilyas, and Stan Zdonik. Data Curation at Scale: The Data Tamer System. In *CIDR 2013*, 2013.

Mohamed Yakout, Kris Ganjam, Kaushik Chakrabarti, and Surajit Chaudhuri. InfoGather: entity augmentation and attribute discovery by holistic matching with web tables. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 97–108, New York, NY, USA, 2012. ACM.

Meihui Zhang and Kaushik Chakrabarti. InfoGather+: semantic matching and annotation of numeric and time-varying attributes in web tables. In *Proceedings of the 2013 international conference on Management of data*, SIGMOD '13, pages 145–156, New York, NY, USA, 2013. ACM.