Welcome to this Panel Session at DBKDA 2020, Lisbon

I'm Malcolm Crowe, a retired academic from University of the West of Scotland. Our panellists for this session are Lisa Ehrlinger, from Johannes Kepler University, Fritz Laux, from Reutlingen University, and Andreas Schmidt from Karlsruhe Institute of Technology.

Our topic: Mobile, Volatile, Incomplete Data leads us to consider how to provide suitable database technology if data is being supplied from mobile devices, if the data keeps changing, or if the data is aggregated from many samples, surveys and such things. And obviously we assume the use of the Web (or at least TCP/IP) as the platform for servers and communications.

Major issues

- Data Integration
  - Data warehouses tend to be static snapshots
  - But most important data evolves from many sources
- Lisa Ehrlinger: How to assure DATA QUALITY?
- Fritz Laux: How to SUPPORT integration?
- Andreas Schmidt: How to MANAGE data models?
- Malcolm Crowe: Real time data integration?

Common to all these topics is a concern with Data Integration. The current state of the art is mostly data warehouses built from static snapshots of data, but most important data sets evolve from many sources. So our panellists will deal with the issues of how to assure data quality, how to support integration, and how to create and manage data models from the evolving data. After this I will briefly reflect on the concept of real-time data integration.

Lisa Ehrlinger

- Topic: Automating Data Quality Measurement for Mobile and Volatile Web Data
- Measuring Characteristics of Data Quality:
  - Knowledge graphs
  - Reference data profiles
  - Traceability of changes to the knowledge graph
- Aims to achieve a higher degree of automation than the manual creation of rules by domain experts

Automating Data Quality Measurement for Mobile and Volatile Web Data

Lisa Ehrlinger
Johannes Kepler University Linz, Austria — lisa.ehrlinger@jku.at
Software Competence Center Hagenberg, Austria — lisa.ehrlinger@scch.at

JOHANNES KEPLER UNIVERSITÄT LINZ
JOHANNES KEPLER UNIVERSITY LINZ
Altenberger Straße 69
4040 Linz, Austria
jku.at

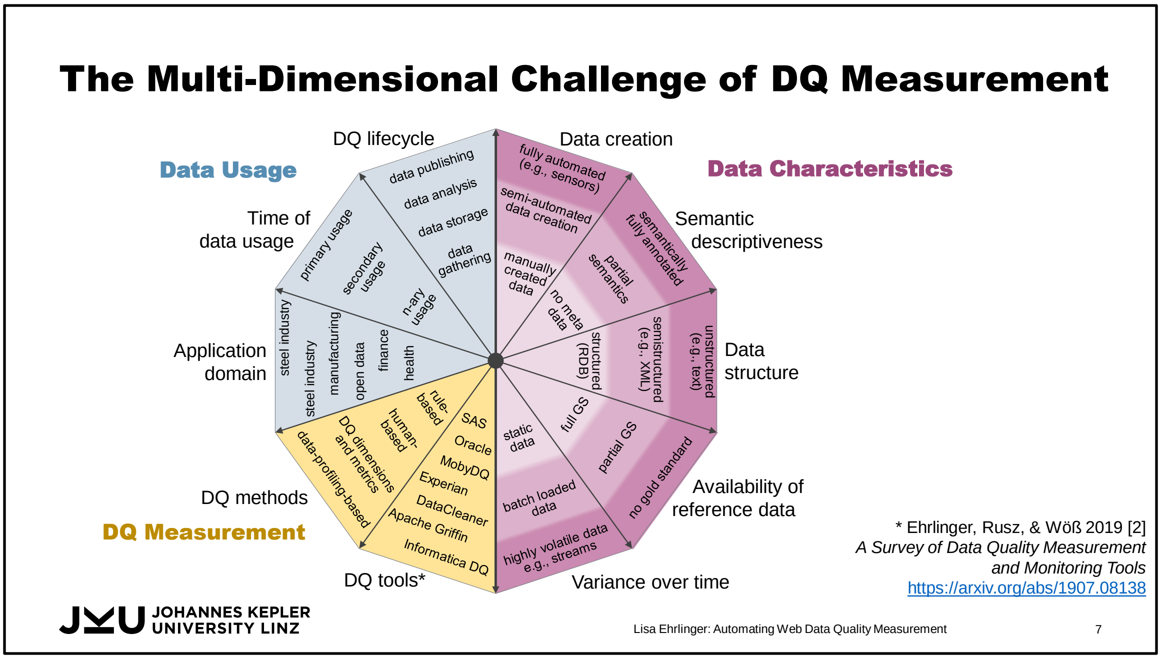
Data, especially data on the web underlies constant change: values are inserted, deleted, or updated, and the meaning of metadata changes over time. To ensure a sufficient level of quality (consistency, conformance) of volatile web data, it is necessary to continuously monitor it and to inform a user in case of abnormal behavior.

Data Quality Researcher at JKU and SCCH

- Johannes Kepler University (JKU) Linz
  - Senior researcher in the research group of Prof. Dr. Wolfram Wöß
  - Data Quality (DQ) tool DQ-MeeRKat: https://github.com/lisehr/dq-meerkat
  - DQ tool survey: https://arxiv.org/abs/1907.08138 (Ehrlinger et al. 2019)
  - Talks at MIT Chief Data Officer and Information Quality Symposium 2019 and 2020
    https://www.youtube.com/watch?v=InFTjhtpp94

- Software Competence Center Hagenberg GmbH (SCCH)
  - Senior researcher, currently responsible for R&D area “Data Management and Data Quality”
  - Research on DQ issues with industrial companies (e.g., KTM motorbikes)
  - DQ tool: A DaQL to Monitor Data Quality in Machine Learning Applications
    International Conference on Database and Expert Systems Applications. Springer, Cham (Ehrlinger et al. 2019)

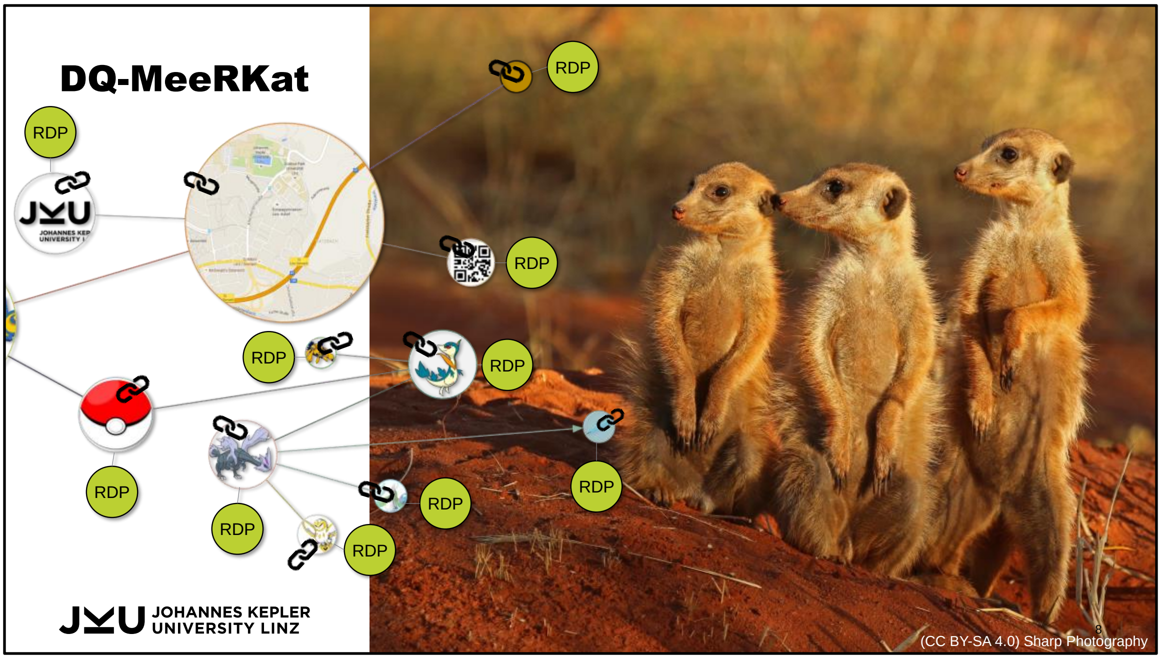
This slide shows a short resume about myself and my current roles and research at Johannes Kepler University Linz (JKU) and the Software Competence Center Hagenberg (SCCH).



With this slide, I would like to highlight the challenges of holistic DQ measurement, with a specific focus on web data quality. In order to measure DQ in practice and to build a DQ tool, it is crucial to understand these challenges for the data to be observed.

The right hand side of the chart (in purple) shows the intrinsic data characteristics and on the left upper hand, external data usage is illustrated. In alignment to Rich Wangs definition of "fitness for use", it is important to consider both and to also take into account the context and the usage of the data. Data on the web is often un- or semistructured without a gold standard at hand and might change very quickly.

At the MIT Chief Data Officer and Information Quality Symposium 2019, I presented a survey on DQ measurement and monitoring tools, where we found that most tools support rule-based DQ measurement. But DQ can also be measured manually by humans. In this talk I would like to present a new method: data-profiling-based DQ measurement, which achieves a higher degree of automation than the manual creation of rules by domain experts.



At JKU, we developed DQ-MeeRKat, a DQ monitoring tool that achieves a higher degree of automation than existing tools. DQ-MeeRKat is based in 3 concepts:

(1) it exploits the power of knowledge graphs (KGs) to provide a global, homogenized view of data schemas,

(2) it introduces "reference-data-profiles", which serve as quasi-gold-standard to verify modified data, and

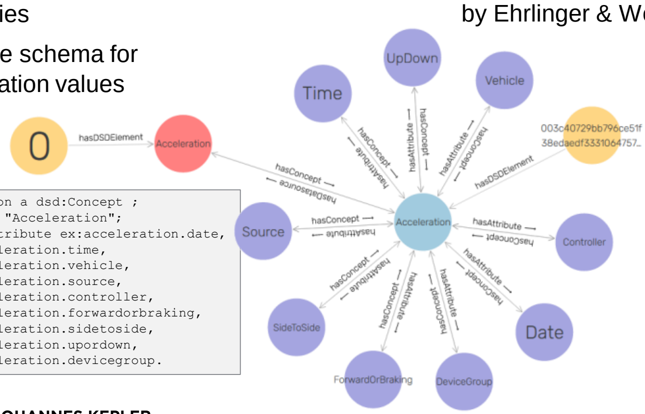
(3) optionally utilizes a blockchain to make changes in the graph globally visible, traceable, and tamper-proof.

Each of the three concepts is explained in the following slides.

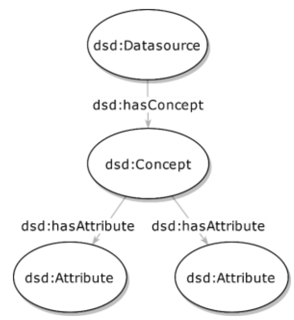
(1) Knowledge Graph (KG): A Global View on Data and its Semantics

- Explicit semantic data modeling with ontologies
- Example schema for acceleration values

- Data source description (DSD) vocabulary by Ehrlinger & Wöß 2015




```
ex:acceleration a dsd:Concept ;
   rdfs:label "Acceleration";
   dsd:has Attribute ex:acceleration.date,
      ex:acceleration.time,
      ex:acceleration.vehicle,
      ex:acceleration.source,
      ex:acceleration.controller,
      ex:acceleration.forwardorbraking,
      ex:acceleration.sidetoside,
      ex:acceleration.updown,
      ex:acceleration.devicegroup.
```

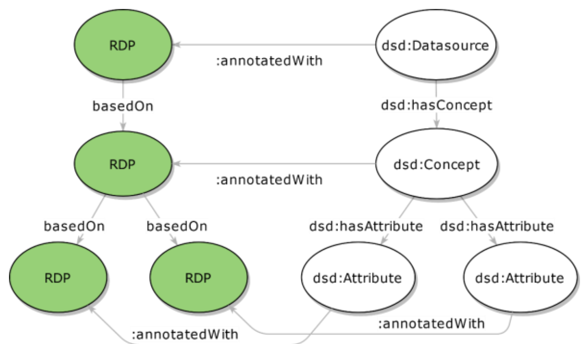


A knowledge graph allows to provide a global view on data. In the backend, we use the DSD vocabulary, originally published by Ehrlinger and Wöß in 2015 to automatically create a semantic description of each local data source in the system. The DSD vocabulary allows to represent each data source in a standardized form. An example is shown on the left hand side of the slide, where a streaming data set is visualized as a graph as well as in its machine-readable form.

(2) Reference Data Profiles

DP Category	DP Metric
Cardinalities	# Rows
	# Nulls
	% Nulls
	# Distinct
	Uniqueness
Data type info	Data type
	Minimum
	Maximum
	Average
	Median
	Standard deviation
	# Digits
	# Decimals
Histogram	# Classes
	Class range
	Values
Dependencies	Key candidate
Patterns	Pattern recognition

Each element in the graph is **automatically annotated** with a reference data profile (RDP)



After the initialization of the semantic graph, which contains the schema descriptions, each element in the graph is automatically annotated with a RDP. This means that each element can have a RDP and these have dependencies between each other. An example would be if an attribute is not allowed to have null values, its comprising concept might still allow a specific percentage of null values for an entire table. In summary, a RDP can be seen as quasi-gold-standard, where manipulated data (inserted, updated, or deleted data) can be checked if it still adheres to the constraints stored in the RDP.

Additional Value of Reference Data Profiles

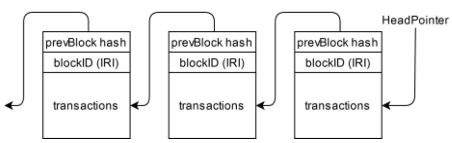
- **Improved automation through RDPs**
  - Creation of RDPs and annotation to the KG can be done fully automatically
  - RDPs replace the high manual effort of specifying and maintaining *DQ rules*
- **Data quality dimensions not required**
  - Different (partially ambiguous) definitions and classifications of DQ dimensions exist
  - To practitioners it is often unclear which DQ dimension to choose (Sebastian-Coleman [3])
  - RDPs = measure-centric approach where *DQ dimensions / metrics* are not required
- **DQ-MeeRKat provides a data and domain-independent solution**
  - Most existing DQ tools support DQ measurement on single tabular files only (cf. Ehrlinger, Rusz, & Wöß 2019)
  - The KG in DQ-MeeRKat allows to investigate multiple heterogeneous data sources at once

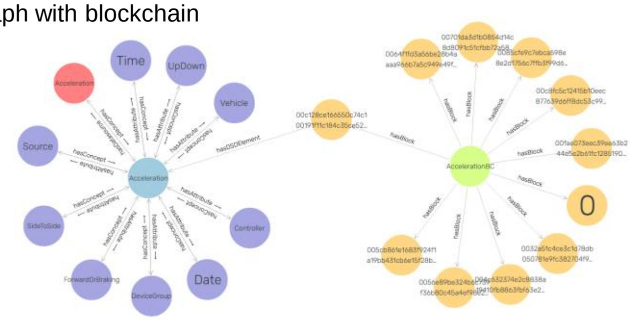
JKU JOHANNES KEPLER UNIVERSITY LINZ

This slide justifies the additional value of reference data profiles in comparison to state-of-the-art methods for DQ measurement.

(3) Blockchain for Tracking Schema Development

- **Challenge**: different departments *modify schema elements or data profiles*
  - Changes are not always reported
  - Domain knowledge is not kept up-to-date
- **Solution:** track changes in knowledge graph with blockchain
  - Each schema modification is persisted as globally visible tamper-proof state





The third concept implemented in DQ-MeeRKat is the blockchain. Although optional, the aim of the blockchain is to track schema development over time. While private or permissioned blockchains (e.g., in closed company settings) have the problem that they are not tamper-proof (i.e., no global consensus, no Proof of Work), public blockchains on the web are more beneficial for this use case.

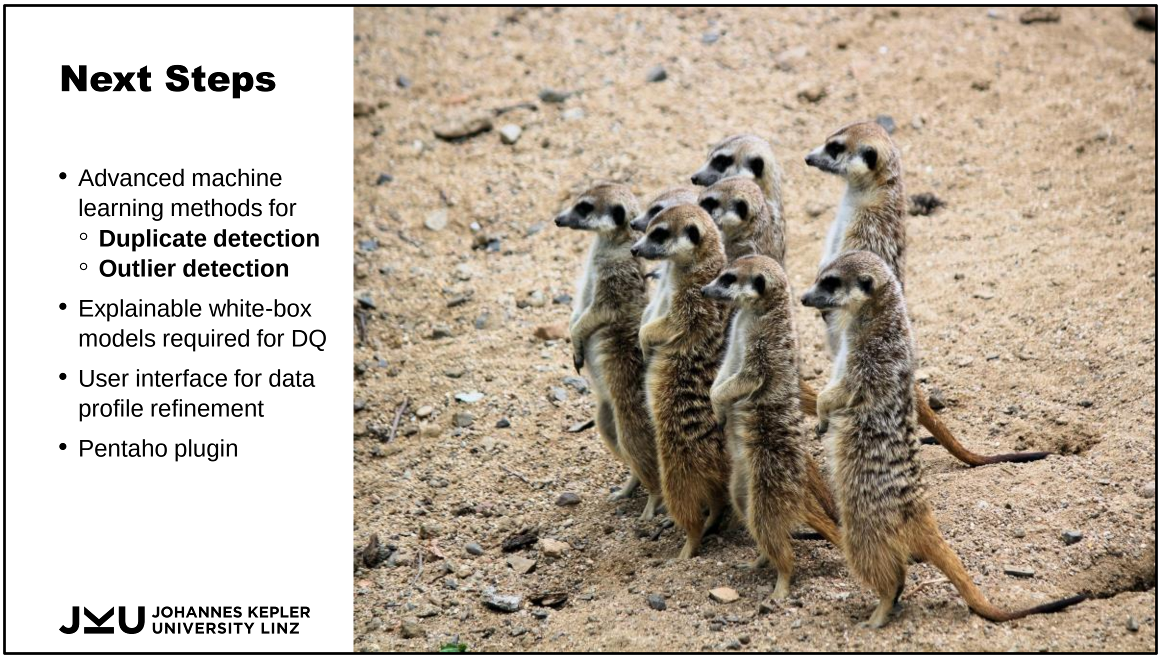
DQ Monitoring Demo with Data Streams

- **Tributech Solutions GmbH**
  https://www.tributech.io/
- Evaluating the performance of streaming values to their RDPs
  - Forward or braking (99.99 %)
  - Side to side (100 %)
  - Up or down (99.99 %)

tributech

JOHANNES KEPLER UNIVERSITY LINZ

This slide shows an example how to monitor real-world data streams. The data streams are provided by Tributech Solutions GmbH, an Austrian start-up that offers cloud-based solutions for the auditability of provisioned data streams. The highly volatile data streams comprise data on acceleration values (e.g., forward or braking, side to side, up and down), engine information, and device voltage, and have been collected from a mobile device assembled in an Audi A4 that reads the CAN bus.



The last slide of my short presentation provides an outlook on our ongoing and future work. Currently, our major aim is to extend the current data-profiling-based statistics with advanced solutions using machine learning (ML). We will focus on white-box models only (no neural networks) since it is crucial that statements about DQ are always explainable. Examples are regression analysis or time-series analysis. Further, in order to seamlessly use DQ-MeeRKat for DQ monitoring in integration scenarios, we are currently working to adjust it as Pentaho plugin.

If you are interested to attend our early adopters program, please contact me via lisa.ehrlinger@jku.at.

Fritz Laux

- Topic: Data Preparation for Integration and Analysis
- Mobile, Volatile, and Incomplete Data on the Web need a well-designed data preparation to be useful for Integration and Analysis.
- This is possible by steps of careful selection, adjustments, harmonization, grouping, correction and amendment.

Reutlingen University

IARIA

Panel Discussion at InfoSys 2020, Lisbon, 27.09. – 1.10.2020

Mobile, Volatile, and Incomplete Data on the Web

Data Preparation for Integration and Analysis

Fritz Laux
Prof. emeritus
Reutlingen University
Dept. of Informatics
Reutlingen, Germany

fritz.laux@reutlingen-university.de

© F. Laux

My name is Fritz Laux. I'm a retired professor from Reutlingen University where I was responsible for the database teaching and research since the start of our department in 1984.

My contribution to the panel focusses on Data Preparation for Integration and Analysis.
Data preparation is important to make data on the Web ready to use for Analysis or Integration.

© F. Laux, Reutlingen University

Reutlingen University

IARIA

Panel Discussion At InforSys 2020 Lisbon Sept 2020

Problems and Scope of Topic

- *Mobile, volatile, and incomplete data on the Web need preparation to be useful*
  - Data might be unreliable, inconsistent, faulty, incomplete, …
- *What are the Problems*
  - Traditional Extract-Transform-Load (ETL) is not suitable because data can be volatile, outdated, and scattered on changing Web pages → live data needed [Crowe2017]
  - Data is "buried" somewhere in the Web → data integration needed [Crowe2017]
  - Data may have different syntax, units, semantics, and may be wrong or missing → data preparation needed
- *My topic*
  - preparation steps and actions to make data ready for use?

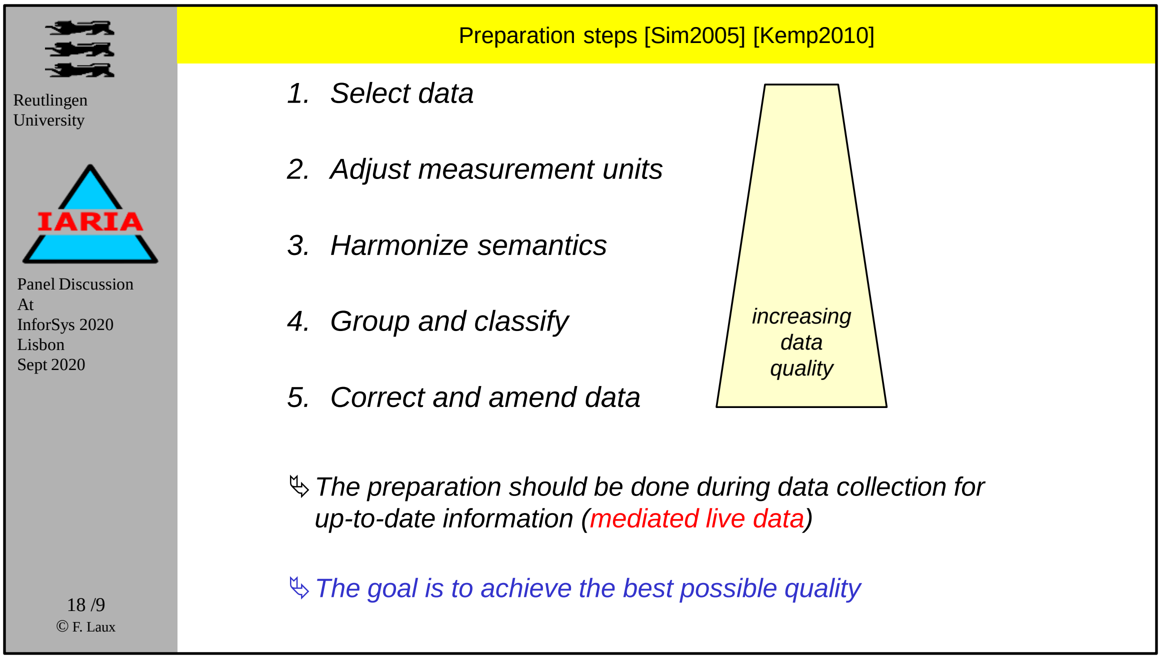
17 /9

© F. Laux

Mobile, volatile, and incomplete data on the Web suffer from various deficits: data might be unreliable, inconsistent, faulty, incomplete, …

To overcome the shortcomings data need some preparation, but the traditional Extract-Transform-Load (ETL) process is not suitable because data can be volatile, outdated, and scattered on changing Web pages. We need the latest data, called live data.
Data is residing somewhere in the Web and data owners need to agree to provide a live data views for processing.
The data may have different syntax, units, semantics, and defects, therefore a careful data preparation is needed.

In the following, I will only focus on the preparation steps.



Inspired by the book of Kemper/Baars/Mehanna and the paper of Simitsis et al. we distinguish 5 data preparation steps:

In Step 1 the required data will be identified and retrieved.

Step 2 cares about measurement units and other meta information from HTML-tags or schema information depending on the data origin.

In Step 3 synonyms and homonyms are identified and renamed according its semantic.

With Step 4 aggregates are built, this can improve speed and make analysis tasks possible on devices with limited storage and power.

Step 5 corrects apparently incorrect, like age < 0 or age > 130.

Reutlingen University

IARIA

Panel Discussion
At
InforSys 2020
Lisbon
Sept 2020

Select Data

- *Define what data is needed, select sources, and extract data*
  - Find quality sources (reliability, timeliness, data type, value range)
  - Get access permission
  - Select data fields (see [Caf2009] for Text extraction from Web Pages)
  - Cleanse syntax (Remove syntactical faults and decoration)
- *Examples*
  - Quality sources (official web pages, statistical offices, Wikipedia, …
  - Data cleansing: remove special and control characters ,e. g. HTLM tags, but keep the semantics it carries, unify rendering of decimal point, measure units, etc.
- *It is crucial to have high quality sources because the remaining steps depend on the selected data*

19 /9
© F. Laux

First of all find quality sources in terms of reliability, timeliness, data type, and value range. Promising sources are official web pages, statistical offices, Wikipedia, …

Then get access permission, select data and cleanse the syntax, this means, Remove syntactical faults and decoration. This first data preparation step includes the removal of special and control characters while keeping the necessary semantics.

Reutlingen University

IARIA

Panel Discussion
At
InforSys 2020
Lisbon
Sept 2020

Adjust Measurement Units

- *Adjust coding and convert measurements to the same unit*
  - Transform different coding to make it uniform
  - Convert measurements to the same unit
  - Make sure that the data has the same granularity
- *Examples*
  - Adjust coding: Transform different gender coding into a uniform one, uniform money coding, e.g. exact numeric coding like SQL decimal(12,2), gender, e.g. M=male, F=female, D=diverse
  - All prices in € or US$, weight in kg, length in m, etc.
  - Same granularity: Price per part, sales per customer, sales per day, …
- *This makes data ready to process:*
  - Quantitative data allow arithmetic calculations
  - Categorical data allow comparison

20 /9
© F. Laux

In step 2 the measure units and other meta-data provided by the data owner are used to adjust all measures to the same unit and granularity.
For instance, use the same numeric coding like SQL decimal(12,2) for all currencies;
Code the gender in a uniform way, e.g. M=male, F=female, D=diverse.
With this step the data is ready for arithmetic calculations if the data is quantitative.
For categorical data comparison is possible.

© F. Laux, Reutlingen University

Reutlingen University

IARIA

Panel Discussion
At
InforSys 2020
Lisbon
Sept 2020

Harmonize Semantics

- *Identify homonyms and synonyms and unify names*
  - Create meaningful names for homonyms
  - Choose best name for synonyms
- *Examples*
  - Homonyms: name could be a person's first name, last name, part name, object name.
    Order could mean a customer order or an internal order
  - Synonyms: part no, material id, EAN (EU article no) could mean all the same.
    Order, customer order, customer booking, subscription could mean all the same.
- *Goal is to clarify the information a data item carries*
  - This is necessary to map and merge data correctly

© F. Laux

The next step harmonizes all synonyms and homonyms providing unambiguous naming and data description (meta-data). When dealing with homonyms we need to introduce new distinct names. In contrast, synonyms must agree to one name, usually the most precise one. For instance, part no, material id, EAN (EU article no) could mean all the same.
The Goal is to clarify the information a data item carries. This is necessary to map and merge data correctly during the integration process.

© F. Laux, Reutlingen University

Reutlingen University

IARIA

Panel Discussion
At
InforSys 2020
Lisbon
Sept 2020

Group and Classify

- *Group or classify data to meet the desired granularity*
  - Group data and count its cardinality, generate summary values
  - Bin temporal data and provide count, average, sum, etc.
  - Calculate aggregate values for data classes
- *Examples*
  - Group customers into ABC-customers
  - For time series bin data into equidistant intervals
  - Classify products into product groups, product families, product lines
- *Grouping and Classifying helps to compare and process data on the same abstraction level*

22 /9
© F. Laux

The 4th step depends on the intended analysis and can be considered as an analysis-specific data preparation. A common example is the ABC-customers grouping.

For time series processing data should be binned into equidistant intervals. Classification of products into product groups, product families, product lines can help applications to reduce the amount of data for devices with limited processing power and small displays.

Grouping and Classifying in general helps to compare and process data on the same abstraction level

Reutlingen University

Panel Discussion At InforSys 2020 Lisbon Sept 2020

Correct and Amend Data

- *Correct obviously incorrect data and amend missing data*
  - Identify apparently incorrect (impossible data, outliers) or missing data
  - If multiple sources are available use most plausible value or average value
  - If only one source is available use mean value of adjacent data for sequential or temporal data
  - Else use default value or NULL (to signal unknown values)
- *Examples*
  - Apparently incorrect: age < 0 or age > 130,
  - Outlier in sequence: 2, 3, 4, 5, 6, 100, 8, 9, 10, 11, … replace 100 by 7
  - Missing data in time series: 2, 3, 4, 5, 6, , 8, 9, 10, 11, …
    fill empty slot with mean value of adjacent values (6 + 8)/2 = 7.
  - Missing data in nominal values: e.g. gender is missing, use NULL
- *This final step needs special care because the reliability of the data sources and the nature of the data is crucial and determine if and how the data should be corrected or amended.*
  - This process should be manually supervised

23 /9
© F. Laux

Apparently incorrect data could be replaces by default values or by mean values. This highly depends on the nature of the data.
This last step depends very much on human rationale and requires a human decision if data should be corrected or not.
For time series missing values need to be complemented to make the algorithms work correctly. This can be done with interpolation of missing data.
In other cases it could be prohibited to add data because this can invalidate the results. Therefore, this process needs human supervision.

Reutlingen University

Panel Discussion
At
InforSys 2020
Lisbon
Sept 2020

Literature

- [Crowe2017] M. Crowe et al., „*Data Validation for Big Live Data", DBKDA 2017, Barcelona, Spain, ISBN13:* 978-1-61208-558-6
- [Sim2005] A. Simitsis et al., "Extraction-Transformation-Loading Processes", in Encyclopedia of Database Technologies and Applications, 2005, ISBN13: 9781591405603, DOI: 10.4018/978-1-59140-560-3.ch041
- *[Kemp2010] H.-G. Kemper, H. Baars, and W. Mehanna , Business Intelligence – Grundlagen und praktische Anwendungen, Vieweg+Teubner Verlag, 2010, ISBN13: 9783834807199*
- [Caf2009] *Michael J. Cafarella, Extracting and Managing Structured Web Data, PhD-Dissertation, University of Washington, 2009*

24 /9
© F. Laux

In [Crowe2017] the term live data was introduced and a technical implementation for virtual integration was presented.
The contributions of [Sim2005]. [Kemp2010], and [Caf2009] name and propose process steps for preparation and transformation of data.

© F. Laux, Reutlingen University

Andreas Schmidt

- Learned Database Models
- The idea is to learn the characteristic of a dataset (typically using Deep Neural Networks - DNN) and then query this model instead of the original dataset.
- The advantage of this approach is, that the model can queried much faster, compared to the original dataset and is also much smaller (typically multiple magnitudes).
  - This can be very beneficial using mobile devices with less computing and storage capacities as well as unstable connections.
- The challenges of this approach lie in the accuracy of the results, the learning time and the ability to incorporate updates into the model.

Karlsruhe Institute of Technology

Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

A short Resume of the Presenter

Prof. Dr. **Andreas Schmidt** is a professor at the Faculty of Computer Science and Business Information Systems of the Karlsruhe University of Applied Sciences (Germany). He is lecturing in the fields of database information systems, data analytics and model-driven software development. Additionally, he is a senior research fellow in computer science at the Institute for Automation and Applied Informatics (IAI) of the Karlsruhe Institute of Technology (KIT). His research focuses on database technology, knowledge extraction from unstructured data/text, Big Data, and generative programming. Andreas Schmidt was awarded his diploma in computer science by the University of Karlsruhe in 1995 and his PhD in mechanical engineering in 2000. Dr. Schmidt has numerous publications in the field of database technology and information extraction. He regularly gives tutorials on international conferences in the field of Big Data related topics and model driven software development. Prof. Schmidt followed sabbatical invitations from renowned institutions like the Systems-Group at ETH-Zurich in Switzerland, the Database Group at the Max-Planck-Institute for Informatics in Saarbrucken/Germany and the Data-Management-Lab at the University of Darmstadt.

that‘s me … ;-)

Research Interests

- For PIA Group at KIT see: https://www.iai.kit.edu/english/941.php
- Additionally, all sort of database related stuff, like
  - Database implementation
  - Graph databases
  - Semantic Text Analysis
  - Information Retrieval
  - ...

… and my research interests

Learned Database Models: General Idea

- Learn the characteristic of a (potentially huge) dataset
- Small, compared to complete dataset --> can be queried fast
- Useful i.e. for
  - Estimation of missing data values
  - Approximate Query processing
  - Query Optimizer cost model (cardinality estimation)
- Challenges
  - learning speed
  - updates

The underlying idea is to learn the characteristics of a data set in an offline phase. this includes aspects such as correlations between values of a dataset, but also the cardinalities between entities.

The result is a model of your data set, which is typically much smaller than the original data set.

For this reason, requests to the model do not provide the same exact results as requests to the original data set, but in a number of cases the quality is sufficient

This technique can be used for example to determine missing values, for time-critical queries where the exact result is not important, or in the context of a query-optimizer, which tries to keep the number of tuples in the intermediate results as small as possible (keyword catdinality estimation)

Challenges here are the learning time as the ability of the procedures to deal with changes in the data set.

Learned Database Models

- „Traditional“ approach to learn a model from a dataset:
  - Run workload with a big number of Queries (offline, typically > 10K queries)
  - Collect results
  - Train model (Deep Neural Network - DNN) with featurized queries and found results

  - At runtime, not the database, but the model (DNN) is queried
  - Updates force model to be rebuild (expensive)

The typical approach for a learned database model is to capture the structure and behavior by executing a typical set of queries (typically > 10K queries) and using the results to train a machine learning model (i.e. neural network).

This workload-driven approach has two major disadvantages. First, capturing the training data can be very expensive, since all queries must be executed on potentially large databases. Second, training data must be re-entered when the workload or database changes.

Our Approach [DeepDB]

- Model learns data distribution (no queries needed) - learn directly from data
- Model is representated as RSPN (Relational Sum-Product-Network), an extension of Sum Product Networks [SPN] to deal also with aggregations and joins
- RSPNs are updateable, so no retraining is necesary.

To avoid these two serious drawbacks, we take a different approach and propose a new data-driven approach to learned DBMS components that directly supports changes in the workload and also data changes without the need to rebuild the model. And as w'll see, this can be done without compromising accuracy compared to the "classic" approach described above.

In contrast to the many approaches based on Deep neural networks (DNN) we use Relational-Sum-Product-networks (RSPN), an extension of the existing Sum-Product-networks (SPN) developed by us, which can also handle aggregations, joins and especially updates.

Karlsruhe Institute of Technology

Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

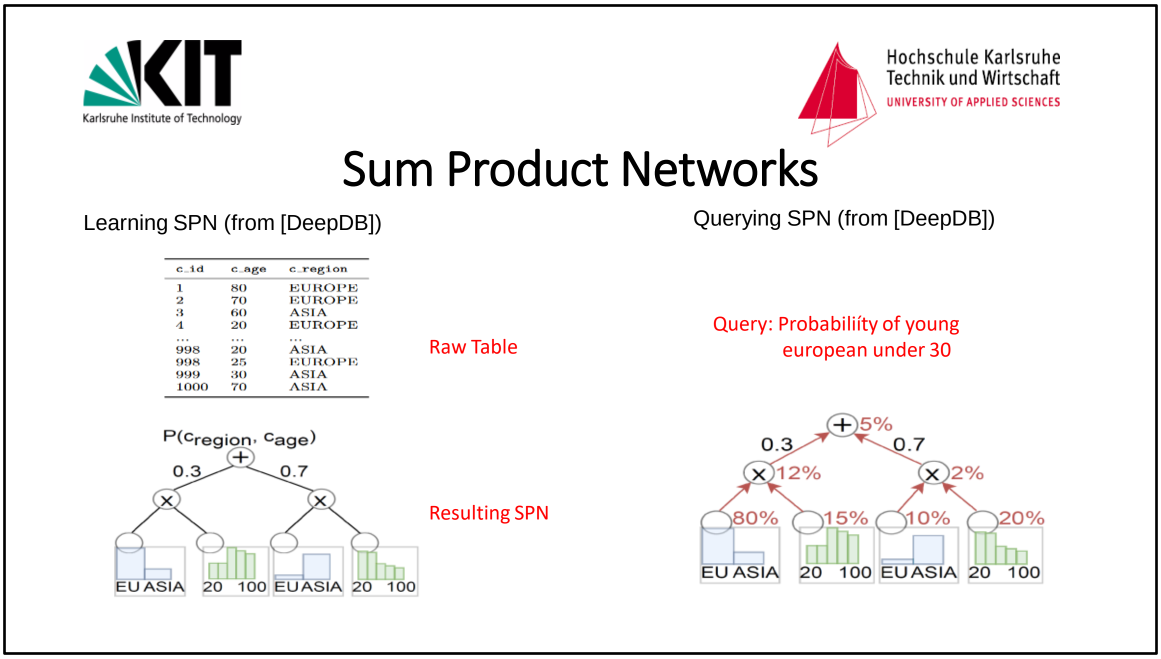
Sum Product Network [SPN]

- SPN learn the joint probability distribution $P(x_1, x_2, ..., x_k)$ of variables $x_1, x_2, ..., x_k$ in a dataset.
- Structure:
  - Tree with Sum, Product and Leaf nodes
  - Product nodes split independent variables
  - Sum nodes splits the datasets into clusters
  - Leaf node reperesent single variable x_i
- Learning phase:
  - Recursively splitting the dataset into clusters of rows (sum-node) or clusters of independent columns (row-node).
  - *Randomized Independence Coefficients* [RDC] are used to test independence between columns (variables X_i), *KMeans* for clustering datasets

A SPN is a tree, consisting of so-called product, sum and leaf nodes. The root of each SPN is a sum node. In the hierarchies below, there are product and sum nodes alternately. the leaf nodes are located on the lowest level.

The SPN divides the entire data set. Sum-nodes split the data set into individual clusters, while product nodes split the data records into independent variables. Leaf nodes then contain information about the value range of a variable. this can be done, for example, by means of a histogram.

During the learning phase the tree is created by alternating horizontal (sum node) and vertical (product node) splitting of the dataset. as criterion for the statistical independence of the values from columns we use RDC.



c_id	c_age	c_region
1	80	EUROPE
2	70	EUROPE
3	60	ASIA
4	20	EUROPE
...	...	...
998	20	ASIA
998	25	EUROPE
999	30	ASIA
1000	70	ASIA

Here we see a simple example …

Starting from the data record (top left image), the SPN (bottom left) is built. For each product node, what proportion of the data records of the sum node above it is represented by it (here 30%, 70%).

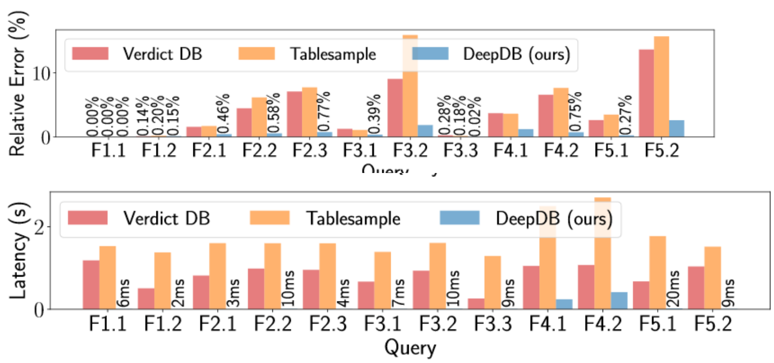
On the left side we see an exemplary query about the percentage of europeans under 30 years of age.

Starting from all affected leaf nodes (which contain the relevant variables) with their value distributions, the tree is then traversed recursively to the root and the overall probability is determined.

here : 0.8 * 0.15 * 0.3 = 0.05

Some Results (from [DeepDB])

Approximate Queries (Flights dataset form [BlinkDB])

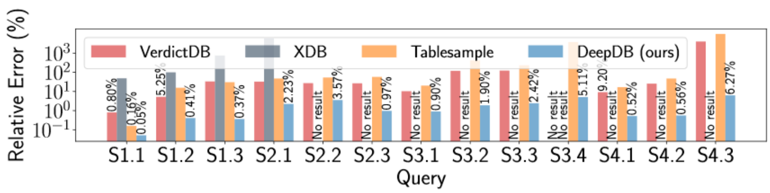


As you can see in the experiment on approximate queries, our approach for the flight data set, as it is also used in BlinkDB, performs significantly better in all queries.

At the same time, our approach has the smallest latency time (figure below)

Some Results (from [DeepDB])

Average relative error of Star Schema Benchmark (SSB) dataset (q-error)



Also the comparision of the relative errors is quite impressive, compared to our competitors (based on SSB dataset)

Some Results (from [DeepDB])

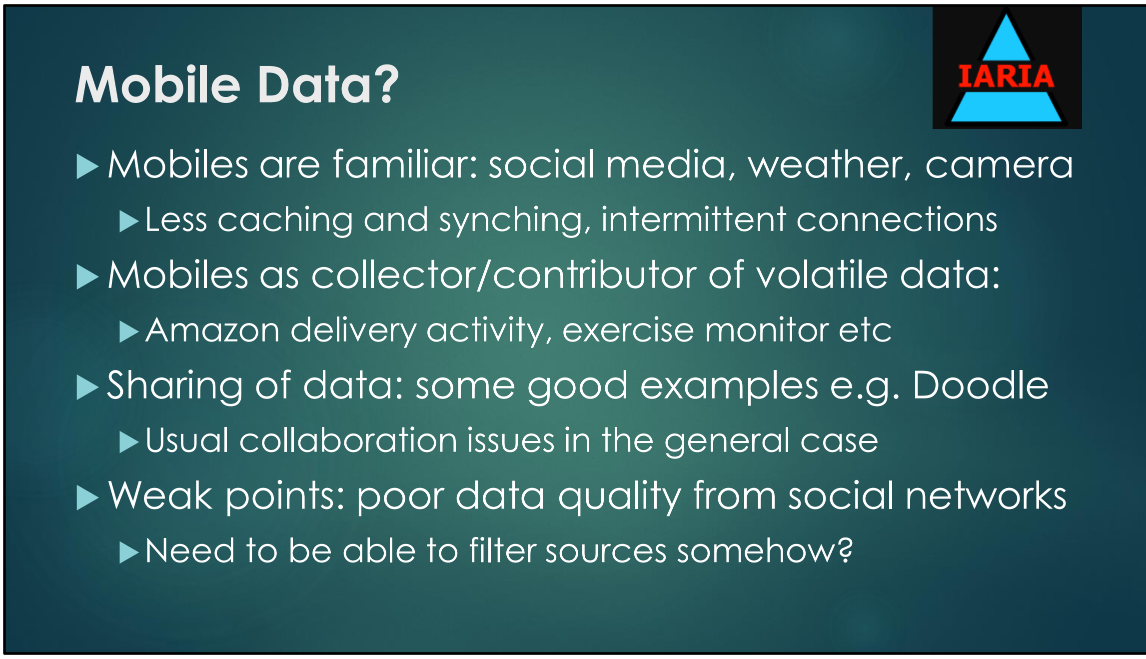
- Estimation errors after updates (q-error)
  - Job Light Dataset
  - Update-rate between 0% and 40%,
  - Splitting criteria: release date of film

Temporal Split	< 2019 (0%)	< 2011 (4.7%)	< 2009 (9.3%)	< 2004 (19.7%)	< 1991 (40.1%)
Median	1.22	1.28	1.31	1.34	1.41
90th	3.45	3.17	3.23	3.60	4.06
95th	4.77	4.30	3.83	4.07	4.35

To show the effects of the accuracy of updates, we learn a certain proportion of the entire IMDb data set and then use the remaining tuples to update the database.
To ensure a realistic setup, we split the IMDb dataset based on the year of production (i.e. newer films are inserted later). As shown in the table , the q-errors are not significantly higher for updated RSPNs, even if the update fraction increases.

Literature

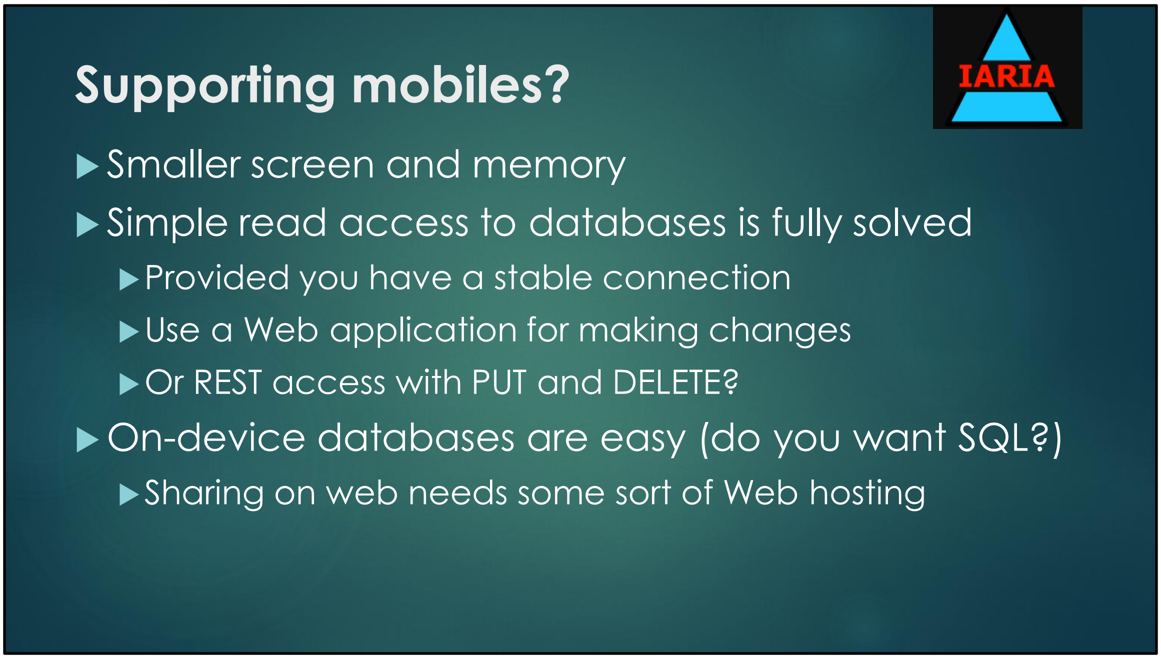
- [DeepDB] Benjamin Hilprecht, Andreas Schmidt, Moritz Kulessa, Alejandro Molina, Kristian Kersting, and Carsten Binnig. DeepDB: Learn from Data, not from Queries. Proceedings of the VLDB Endowment,Vol. 13, No. 7, Tokyo, Japan, 31. August - 4. September 2020.
- [SPN] A. Molina, S. Natarajan, and K. Kersting. Poisson Sum-Product Networks: A Deep Architecture for Tractable Multivariate Poisson Distributions. In AAAI, 2017.‘
- [RDC] D. Lopez-Paz, P. Hennig, and B. Schölkopf. Therandomized dependence coefficient. In Advances in neural information processing systems, pages 1–9,2013.
- [BlinkDB] S. Agarwal, B. Mozafari, A. Panda, H. Milner,S. Madden, and I. Stoica. Blinkdb: Queries with bounded errors and bounded response times on very large data. In Proceedings of the 8th ACM European Conference on Computer Systems, EuroSys ’13, pages 29–42, New York, NY, USA, 2013. ACM.



Almost everybody uses mobile devices for an enormous variety of data – chat, weather, as a camera, for buying things or arranging meetings or holidays. Importantly, many of these activities see the mobile device as a source of new data which is obviously being stored in many databases. A large proportion of such data is obviously volatile: we can think of the current position of a van driver delivering a parcel, the most recent message from our friends, the reading from an exercise monitor.

In many cases, mobile devices play the same role as collaborating desktop clients, for example arranging meetings, or email. Some involve collaborative editing of data and documents, such as with Doodle, or live meetings. It is clear that the technical issues involved in such applications are largely solved, or at least that many ways of managing collaboration have become accepted.

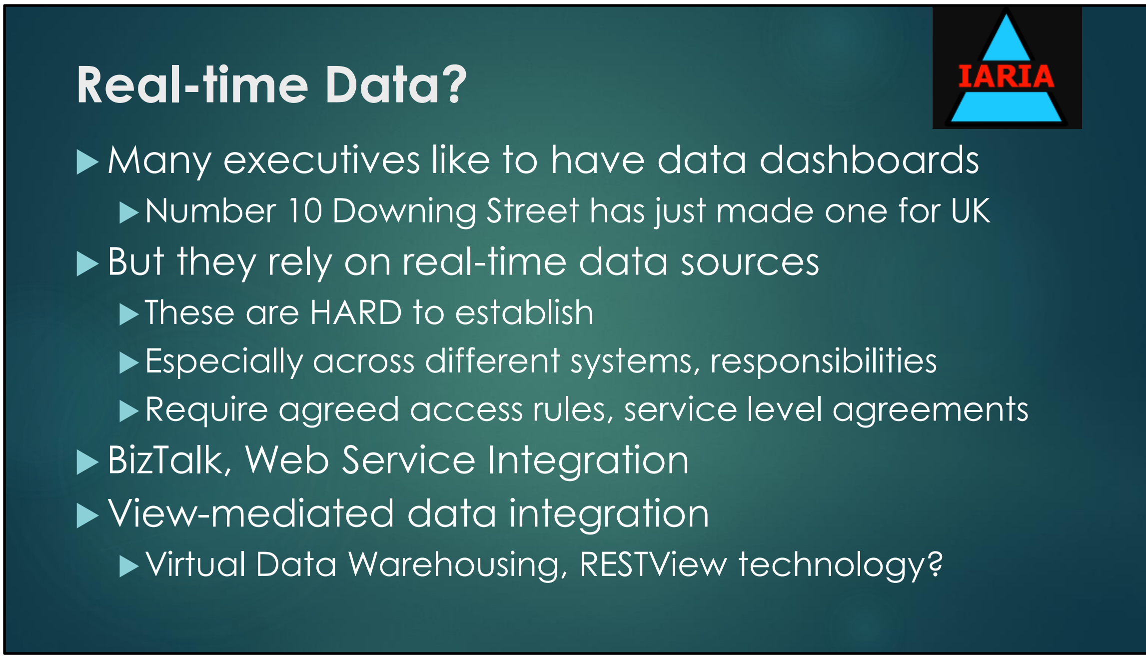
Other issues are more difficult: dealing with fake news, fake reviews, lies and fraud will always be with us, and where facts and accuracy are important there is a need to be able to filter data somehow. Alas, too many business executives insist on being allowed to alter data analytics before publishing them.



While there are similarities between integrating desktop clients and mobile devices, mobiles do bring their own issues. The user interface is different, and the network connection comes and goes.

Nevertheless, for simple read access to online data, the problems of supporting mobiles are pretty much solved by current or Web technology and web services. Web applications now make a great success of making changes to online data, and maybe somewhat round the corner we can expect more adoption of RESTful services for the same purposes.

It is even possible to host a small database on your mobile: many applications effectively do this, though SQL databases are rarely hosted on mobiles.

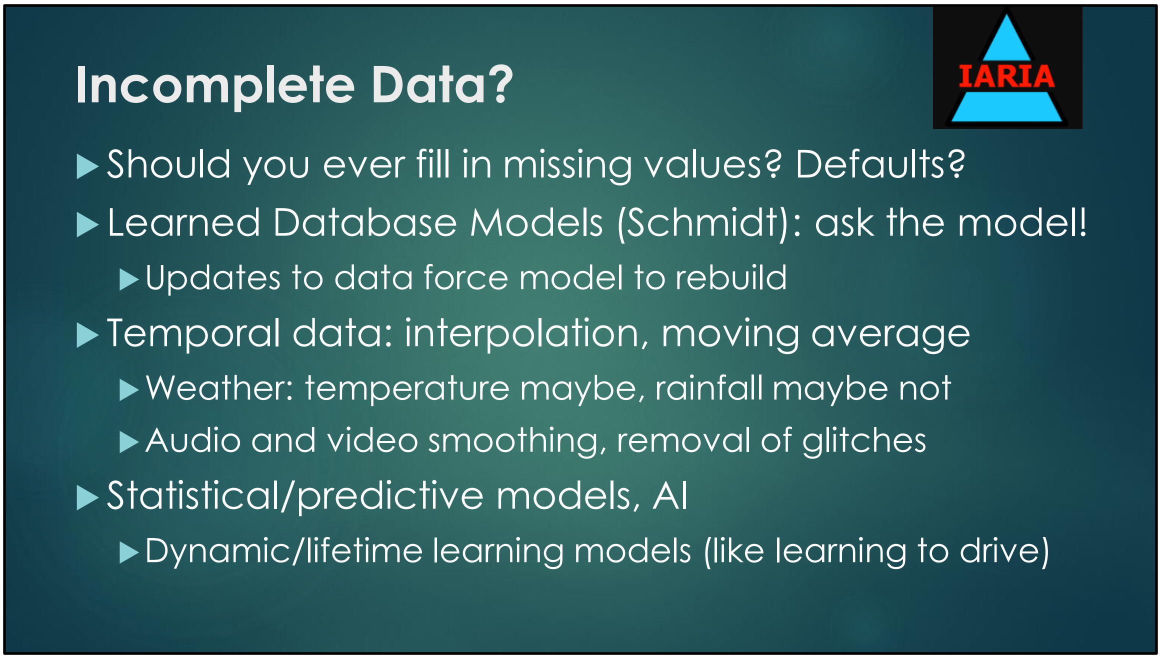


Social media and news feeds can provide real-time data of course, and many businesses dream of having data dashboards to enable them to see in real time how their business is performing. Most universities now have these for student and marketing activity. Newspapers have reported that Number 10 Downing Street has just installed one for monitoring government policy.

But in most cases, the data is very far from real-time. To provide real-time performance data, we need real-time data sources, and except for the simplest cases this are very hard to establish, particularly when the data is integrated or aggregated from different sources with different ownership and responsibilities. In such cases there are always service level agreements to be negotiated to establish the rules of access.

There are intermediate cases where success is currently possible, using web service integration and messaging hubs such as BizTalk. These allow direct interrogation of data sources, and with enough programming effort they can be made to incorporate data from other places. Personally, I believe there is more that can be done to provide tools for the general case by better exploitation of HTTP and particularly

REST, together with the concept of view-mediated virtual data warehousing.



For weather forecasting we are used to displays that immediately follow the weather as gathered from satellites and tracking by a similar-looking video that shows a forecast evolution. Today we have heard a contribution that considers a generalisation of this process to other kinds of data, and there are many applications of this approach in developing neural network models.

Obviously, it is important to distinguish facts from forecasts. Lives have been lost by over-reliance on predictive analytics by governments and police forces. I am told that as a practical matter it is more dangerous to neglect a data source because of some missing values, and there are different mechanisms to resolve these, some of which are more convincing than others.

I have been impressed by recent work in reinforcement learning that solves the problem of continuous or lifetime learning by allowing the agent to resume learning if things change. I look forward to these new ideas finding a place in data integration technology for one or more of the problems considered above.