Reutlingen University

DBKDA 2010, Les Menuires

„I have a Dream“ – a Vision on Database Technology

Fritz Laux

Reutlingen University
Dept. of Informatics
Reutlingen, Germany

fritz.laux@reutlingen-university.de

Reutlingen University

Aim of the Talk

Aim
Challenges
1. Vision
2. Vision
3. Vision

- *This is not a prediction of future database systems*
  - "Prediction is very difficult, especially if it's about the future." [Mark Twain/Niels Bohr]
- *It is not a trend analysis*
  - No statistically provable results
- *It's only a personal view on what kind of functionality future databases should provide*
  - **Research challenges**
  - Not comprehensive, only some examples
  - I have more questions than answers

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The 3 Challenges are …

to overcome the following limitations

- Schema first requirement
- Data integration problem
- Data access/retrieval is for experts only

3 /28

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

1. Challenge: Schema first requirement

- *Data can be stored only after the schema design but …*
- *Database modeling is a challenge*
  - Some data comes with a description (metadata) others not
  - Difficult to understand other peoples' data
  - How about the structure (content, linguistic, format) of textual data?
- *Data model evolves over time*
  - semantic drift (meaning of data changes)
  - schema growing process is manual
- *How do we get rid of the schema?*
  - Or at least automate schema design and management

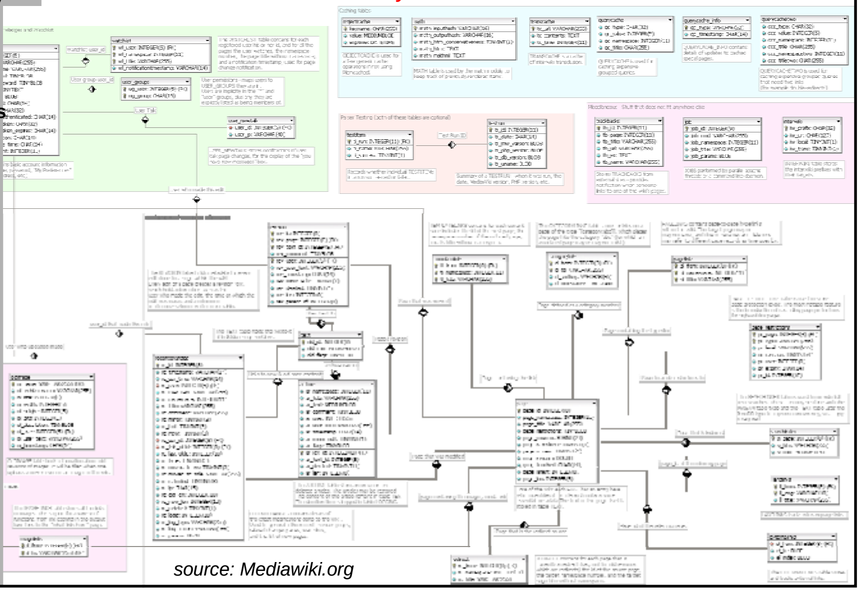
4 /28

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

1. Challenge: Example

- *Do you understand this schema?*
  - And this schema has only 36 annotated tables



source: Mediawiki.org

5 /28
© F. Laux

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

- *No manual modeling or schema design required*

- *DBMS has*
  - no Schema or
  - defines and manages Schema automatically
    - ⇨ Schema derived from (example) data stored
    - ⇨ Automatic schema evolution

- *Research so far*
  - Cassandra Phipps, Karen C. Davis: Automating data warehouse conceptual schema design and evaluation. 23-32, in Laks V. S. Lakshmanan (Ed.): Proceedings of the 4th Intl. Workshop DMDW'2002, Toronto, Canada, May 27, 2002
    → deals only with structured data (from OLTP Systems)
  - B. Howe, K. Tanna, P. Turner, D. Maier. Emergent Semantics: Towards Self-Organizing Scientific Metadata. In Proceedings of Semantics for a Networked World: Semantics for Grid Databases, Volume 3226 of Lecture Notes In Computer Science. Springer, 2004
    → uses triples (id, property, value) for storing data
  - D. Maier. Profiling Dataspaces:Understanding (and Using) Other People's Data, Klaus Dittrich Memorial Symposium, Zurich, CH, 2008
    → reports on a study to find the schema for a medication list RxList and related standards like NDCD, RxNorm with help of Quarry metadata explorer (RDF-like data model) and other data profiler tools

6 /28
© F. Laux

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

No manual modeling or schema design required

Some ideas

- Use meta information from objects to get structure info
  - Examples: obj.class(),
  - obj class instanceVariables class
- Use DTD or XML Schema info for XML documents
  - Example: <?xml version="1.0" standalone="no"?>
    <!DOCTYPE hello SYSTEM "hello.dtd">
    <**hello**>**Hello world!**</hello>
- Use layout/linguistic information from sample text/htm/xml documents with known semantics
- Use statistical information to find the most likely data type or coding
  - Example: always ASCII digits → integer
    always ASCII digits plus punctuation → decimal

Reutlingen University

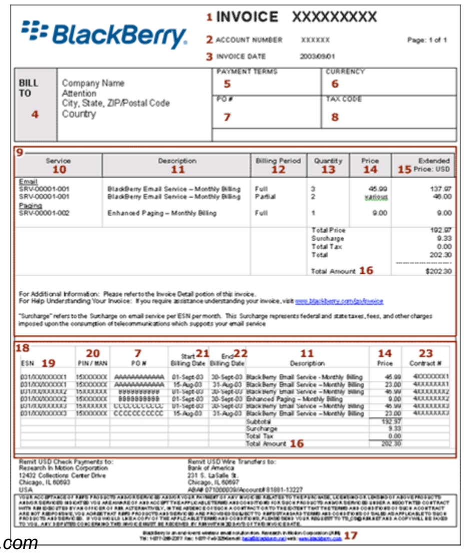
Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

No manual modeling or schema design required

Some ideas

- Use patterns to find structure
- Example: header - detail Relationship
- Generalize: Data structures like x(abc)* suggest
  1:* Relationship
- Use Ontology to find out semantics

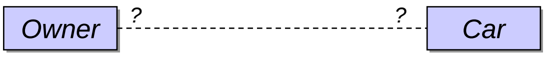


source: blackberry.com

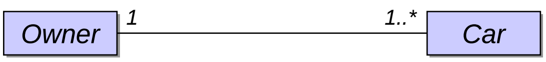
The Vision

- *No manual modeling or schema design required*

- *Some ideas*
  - Stay data type tolerant
    - ⇨ Example: 9/10 digits, 1/10 characters → store as string
  - Stay schema tolerant
    - ⇨ Example: unclear relationship
      then use no relationship and analyze query profile

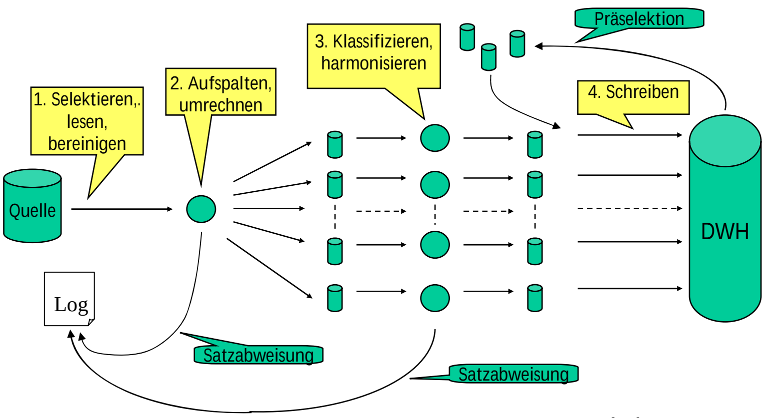


    - ⇨ Example: 9/10 1:* relationship 1/10 *:* relationship
      then allow both until you can rule one out



2. Challenge: The Data Integration Problem

- *Example: Complex ETL process in Data Warehousing*



source: U. Knuplesch, Forum WI, 2000

… and this is only one data source

Reutlingen University

Aim

Challenges

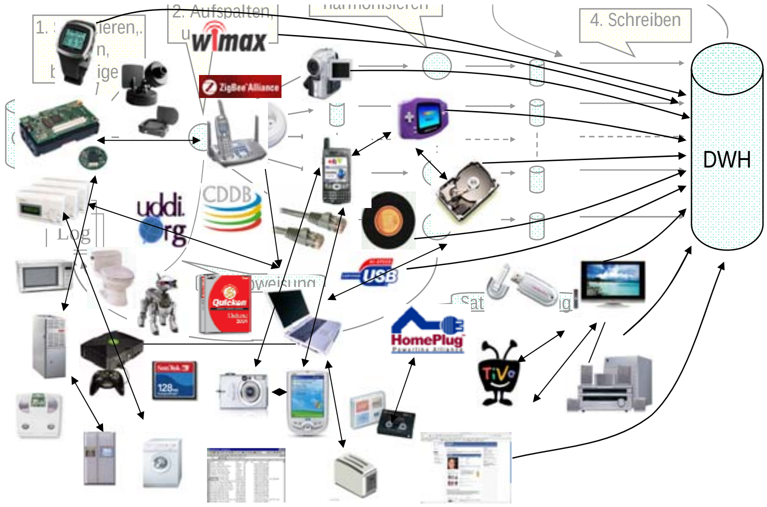
1. Vision

2. Vision

3. Vision

2. Challenge: The Data Integration Problem

- *Example: Complex ETL process in Data Warehousing*
  - What you really need to integrate …
  - … Your data sources are everywhere!



Reutlingen University

Aim

Challenges

1. Vision

2. Vision

3. Vision

2. Challenge: The Data Integration Problem

- *Data must be copied and integrated into a schema (periodical ETL-process)*
  - Data is always out of date
  - Much of the data is never used
- *Data provenance and ownership is lost*
  - Control over data is lost
  - Data quality unknown
- *How can we avoid the integration of highly scattered but interrelated data?*

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

13 /28

The Vision

- *Database is virtual, but manages an interrelation schema or at its best an integration schema*
- *In situ Storage*
  - Data remains where it is created
  - Ownership and provenance preserving
- *Data may be cached for performance*
  - Trade off between performance and consistency, resp. data freshness

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

14 /28

The Vision

- *Database is virtual, but manages an interrelation schema or at its best an integration schema*
- *Research so far*
  - Michael Franklin, Alon Halevy, David Maier: "From Databases to Dataspaces: A New Abstraction for Information Management", SIGMOD Record, December 2005.
    → introduces dataspace concepts, in situ data, collection of relationships
  - Rudolf Munz, "Datenmanagement für SAP Applikationen", in A. Kemper et al (Eds.): Proceedings BTW 2007, 12. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme" (DBIS), Proceedings, 7.-9. März 2007, Aachen, Germany
    → reports on experiments with object caches, in situ queries, column-wise storage and memory blades, incremental data loads

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

Database is virtual, but manages an interrelation schema or at its best an integration schema

Some ideas for storage:

- Data stays at its source location
  - Preserve data provenance
- Select as early as possible
  - Requires query decomposition
- Move data only for performance (caching, hoarding)
  - Needs high bandwidth with low latency

15 /28

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

Database is virtual, but manages an interrelation schema or at its best an integration schema

Some ideas on performance:

- how to accelerate access beyond caching data and high bandwidth?
  - Work in parallel
  - Smart indexes, e.g.
    - index for metadata identifies redundant data
    - Index knows cost function for data access
    - index on stored query results
  - Relaxed consistency requirements with regard to time dimension
    - Store and forward
    - Semantically equivalent query rewrite

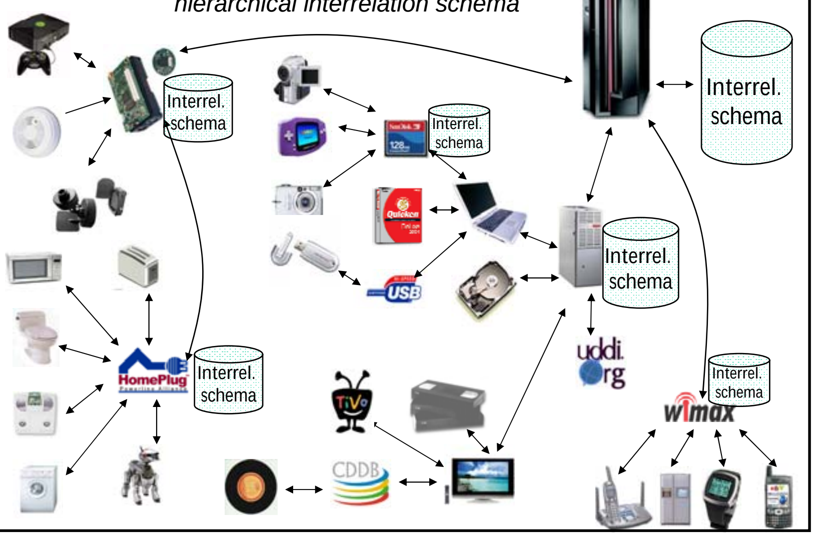
16 /28

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

- *Database is virtual, but manages an interrelation schema or at its best an integration schema*
- *Some ideas: taming the data flood with a hierarchical interrelation schema*



Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

3. Challenge: data access and retrieval only for experts

- *Only syntactic queries possible*
  - Absurd join operations possible
  - Need to learn SQL and know the schema
- *How about information retrieval in text documents?*
  - Only syntactic pattern matching
- *ACID transaction model is not adequate*
  - Long, nested transactions [Korth/Speegle, Wang/Peng]
  - Serializability is often too restrictive
    - Semantic transaction steps [Garcia-Molina, Farrag/Özsu]
    - Multiversion reconciliation [Phatak/Nath]
    - Escrow serializability [Laux/Lessner]
- *How can we make the database more "intelligent"?*

Reutlingen University

3. Challenge: data retrieval

- *Example: What is the average width for images of Mark Twain?*
- *Select avg(width) from ???*
  - What is SQL?
- *Keyword search*
  - Average width image Mark Twain
- *What you expect, is a number*

But Google returns …

Aim
Challenges
1. Vision
2. Vision
3. Vision

Reutlingen

3. Challenge: data retrieval

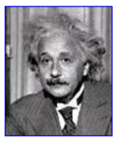
- *Example: What is the average width for images of Mark Twain?*

Web Bilder Video Lokale Suche Shopping Nachrichten Mehr

YAHOO! DEUTSCHLAND SUCHE Niels Bohr Average Suche

Familienfilter: An

? Niels Bohr Average = Albert Einstein ?

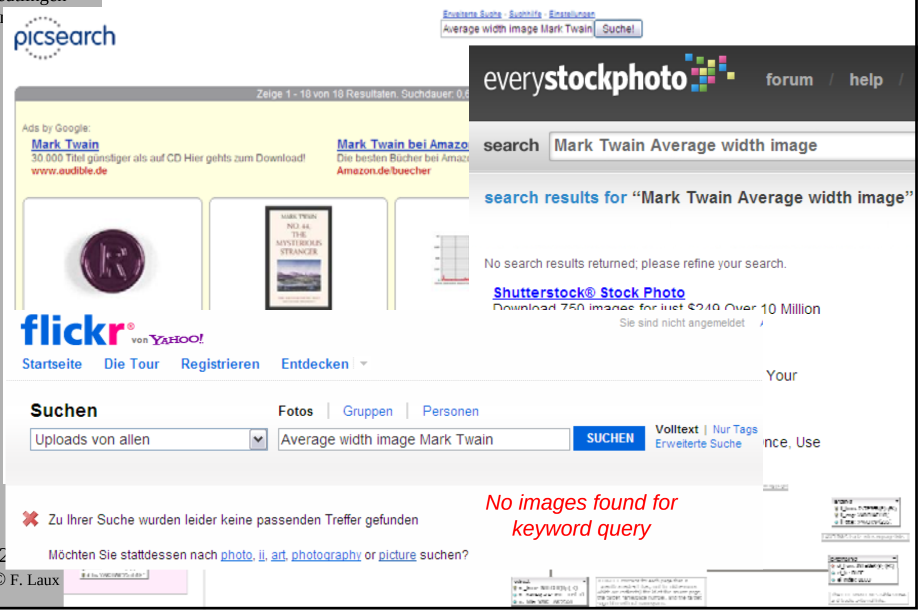


Albert Einstein JPG
104 x 126 | 3k
qenergyspa.com

Reutlingen University

3. Challenge: data retrieval

Example: What is the average width for images of Mark Twain?



No images found for keyword query

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

"Intelligent" data access

Some ideas

- Exact results, where possible
  - Regular structure
  - Irregular structure
    - Keyword query
    - Semantic query
- Intentional query (approximate results)
  - Query is not bound to a schema
    - No SQL, but declarative
  - Query must "understand" your intention
    - Ontology, query profile needed
- Active database, agents
  - Signal/show new information

Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

Some innovative examples

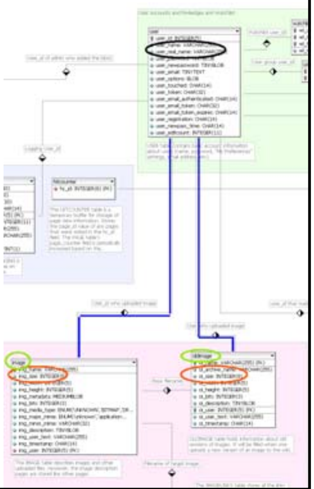
Keywords query on structured data

- Example: SQAK [Tata, Lohman] builds SQL aggregates using keyword query like:

  Mark Twain average **image** **width**

  results in a SQL query derived from the schema similar to:

  select avg(width) from
  (select img_width as width
    from image join user …
    where user like ‚Mark Twain%'
  union
  select oi_width as width
  from oldimage join user …
  where user like ‚Mark Twain%')



Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

Some innovative examples

Linguistic search

- TextRunner [Banko, Etzioni] extracts linguistic triples like [Mark Twain, born on, November 30 , 1835] from text/web documents.

TextRunner Search

TextRunner took 2 seconds.

Retrieved **8** results for **Mark Twain** in argument 1 and **born on** in the predicate.

Grouping results by predicate. Group by: argument 1 | argument 2

was born on - 4 results

Mark Twain was **born on** November 30 , 1835 (16), the day of the appearance of Halley 's Comet (10),
American satirist and writer **Mark Twain** was **born on** November 30 , 1835 (2)
Halley s Comet **Mark Twain** was **born on** the day of the appearance of Halley s Comet (2)
Halley 's Comet **Mark Twain** was **born on** the day of the appearance of Halley 's Comet (2)

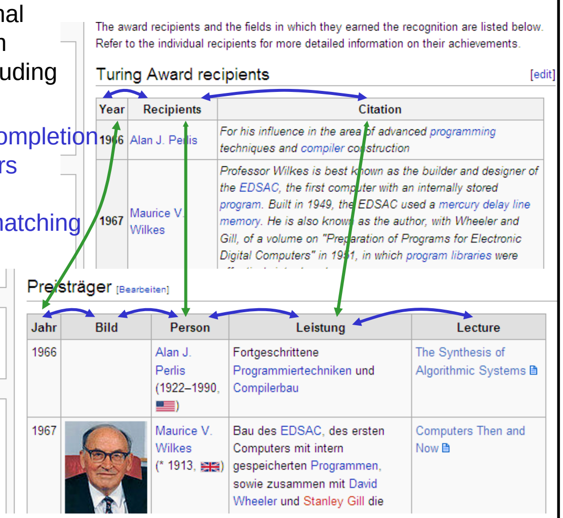
Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

The Vision

Some innovative examples

- *WebTable search [Cafarella et. Al]*
  - Extracts relational information from Web-pages including Metadata
  - Schema auto-completion via table-headers and column name matching
  - Synonym finder or translator via correlated tables



Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

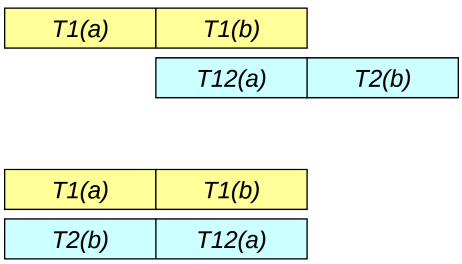
Vision

Ideas for concurrent data access

- *Splitting transaction into compatible atomic steps [Gracia-Molina, Farrag/Özsu]*
- *Example: Money transfer from account a to account b. T1 and T2 are in conflict, but T1(a) and T2(b), T1(b) and T2(a) are „compatible“*

Interleave compatible portions

Reorder portions for compatibility

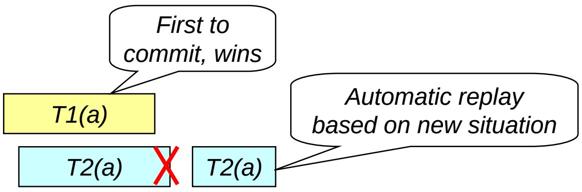


Reutlingen University

Aim
Challenges
1. Vision
2. Vision
3. Vision

Vision

- *Ideas for semantic transaction processing*
- *Semantic classification of transactions that allow escrow serialization [Laux/Lessner]*
- *Example: Money withdraw from account a.*
  T1 and T2 are in conflict, but T1 and T2 are escrow serializable (reconcilable).



- *Open question: getting performance* ***and*** *semantics?*

Reutlingen University

Discussion