Modelling Data Warehouses with Multiversion and Temporal Functionality

Waqas Ahmed
waqas.ahmed@ulb.ac.be

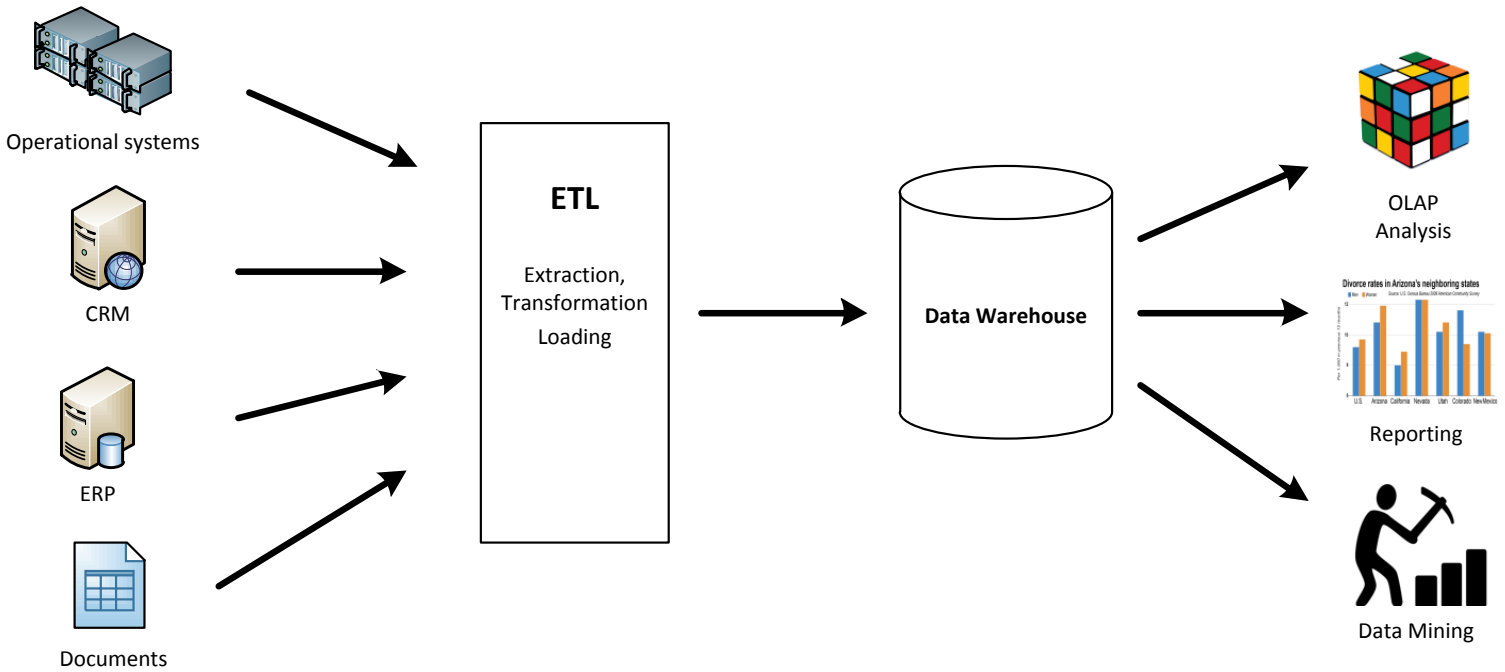
Université Libre de Bruxelles
Poznan University of Technology

July 9, 2015

Outline

Outline

Data Warehouse Architecture



Data Warehouse Architecture

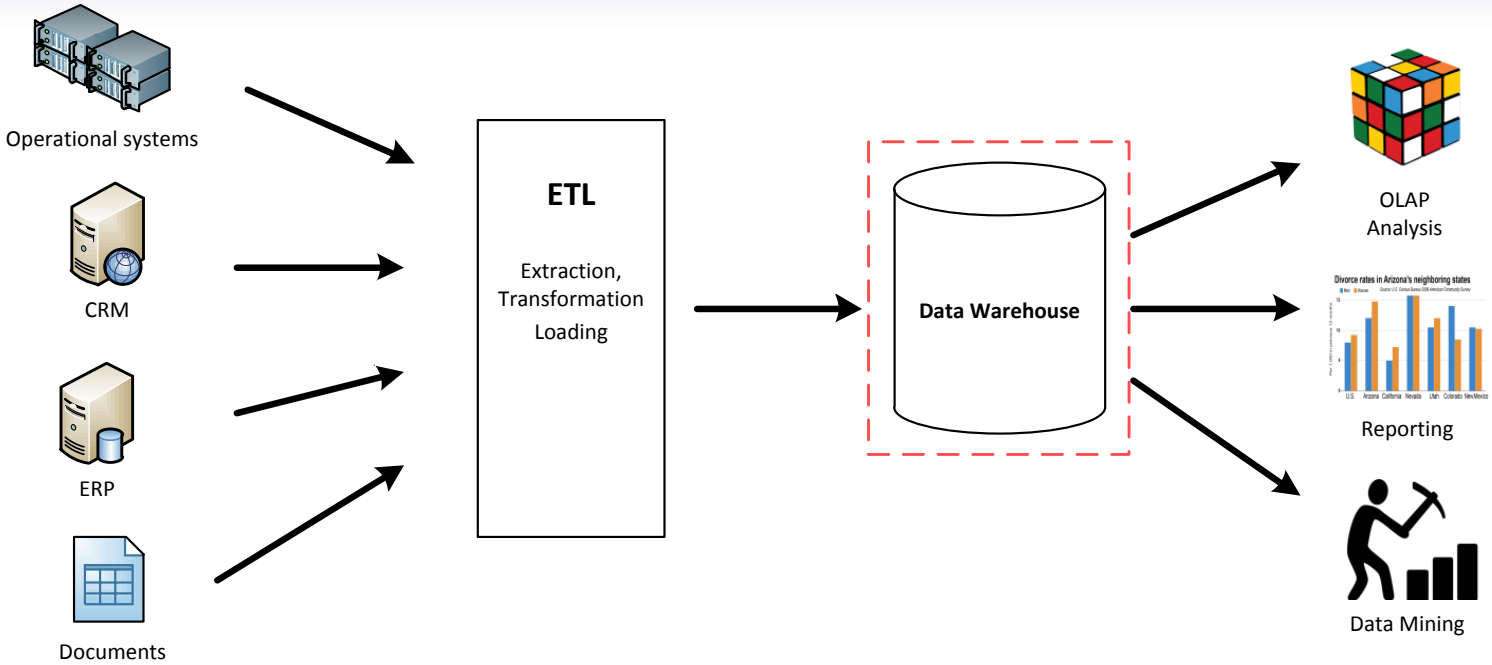
Type of Changes in External Data Sources

External data sources (EDSs) change in their:

1. Content
   - Daily operations
2. Structure
   - Changes in the business requirements
   - Changes in the modeled reality
   - Adaptation of the new technologies
   - Changes in legislation

Propagating Changes in EDS into Data Warehouses

- As a result of changes in EDSs, DWs also change in their:
  1. Content
     - changes in the product prices
  2. Structure
     - changes in the geographical hierarchies
- The user may be interested in keeping the history of changes in the content and structure
  - to reconstruct the state of the business world in the past
  - to simulate the effect of the future changes
  - for the audit and accountability purposes



Data Warehouse Architecture

- What if the user wants to keep the history of changes in the content and structure of the data warehouse?

Outline

Available Approaches

1. Slowly Changing Dimensions
2. Temporal Data Warehouses
3. Multiversion Data Warehouses

Slowly Changing Dimensions (SCDs)

- Idea
  - Three basic and four hybrid responses to changes in the contents
  - Basic types are supported by commercial tools such as SSIS, SSAS
- Drawbacks
  - Not suitable for temporal data – performance issues
  - Unable to handle structural changes
- References
  - R. Kimball and M. Ross. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. 2013
  - Faisal, S. et M. Sarwar. Handling slowly changing dimensions in data warehouses. JSS, 2014

Temporal Data Warehouses

- Idea
  - Systems which provide built in support for *storing* and *querying* time varying data
  - Temporal features are supported by commercial and open source systems such as Teradata, PostgreSQL
- Drawbacks
  - Limited support for temporal operations such as coalescing
  - Unable to handle the structural changes
- References
  - M. Golfarelli, and S. Rizzi. A survey on temporal data warehousing. IJDWM, 2009
  - E. Malinowski, and E. Zimányi. A conceptual model for temporal data warehouses and its transformation to the ER and the object-relational models. DaWak 2008.
  - J. Eder, C. Koncilia, and T. Morzy. The COMET metamodel for temporal data warehouses. In Proc. of CAiSE 2006

Temporal Data Warehouses

- Capture the evolution of attribute values by associating independent time dimensions
  - Valid time – Time when a fact is valid in the modeled reality
  - Transaction time – Time when a fact is registered in the system

Store ID	Address	Manager	VS	VE	TS	TE
s1	ABC	John	1-Jan-2010	now	3-Jan-2010	UC

Table before change

Store ID	Address	Manager	VS	VE	TS	TE
s1	ABC	John	1-Jan-2010	now	3-Jan-2010	1-Dec-2012
s1	ABC	John	1-Jan-2010	1-Jan-2013	1-Dec-2012	UC
s1	ABC	Doe	1-Jan-2013	now	1-Dec-2012	UC

Table after change

Temporal Data Warehouses

- Capture the evolution of attribute values by associating independent time dimensions
  - Valid time – Time when a fact is valid in the modeled reality
  - Transaction time – Time when a fact is registered in the system
- Temporal DWs cannot handle the structural changes

Store ID	Address	Manager	VS	VE	TS	TE
s1	ABC	John	1-Jan-2010	now	3-Jan-2010	UC

Table before change

Store ID	Address	Manager	VS	VE	TS	TE
s1	ABC	John	1-Jan-2010	now	3-Jan-2010	1-Dec-2012
s1	ABC	John	1-Jan-2010	1-Jan-2013	1-Dec-2012	UC
s1	ABC	Doe	1-Jan-2013	now	1-Dec-2012	UC

Table after change

Multiversion Data Warehouses (MVDW)

- Idea
  - Consists of the sequence of DW versions
  - Each DW version is composed of *schema version* and *data version*
- Drawbacks
  - Inter and intra-version querying is complex
  - Storage and performance issues
  - No support for creating temporal data versions in a schema version
- References
  - S. Rizzi and M. Golfarelli. X-time: Schema versioning and cross-version querying in data warehouses. ICDE, 2007
  - R. Wrembel, B. Bebel. Metadata management in a multiversion data warehouse. JODS, 2007

Something More about Cross-Version Queries

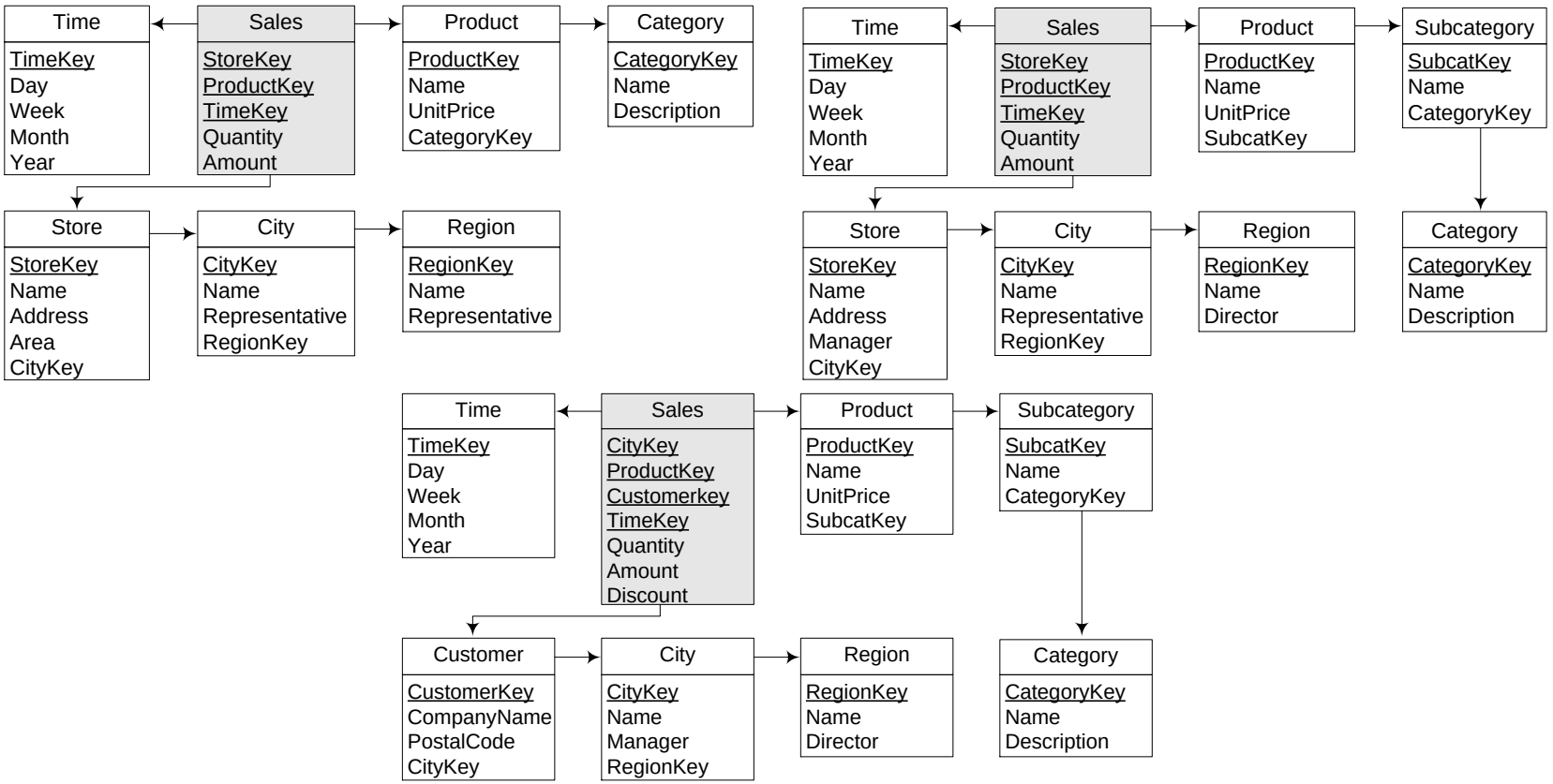


Figure: Schema versions in the example MVDW

Something More about Cross-Version Queries

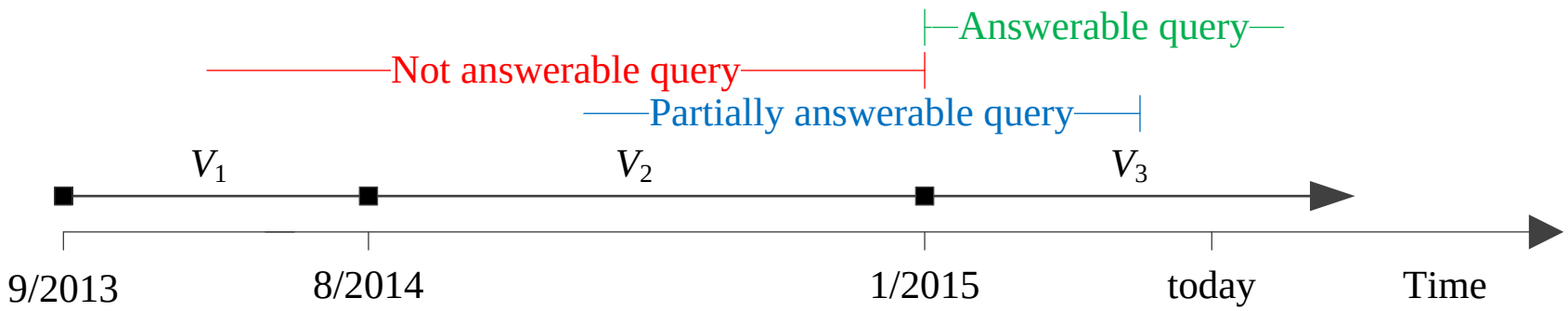


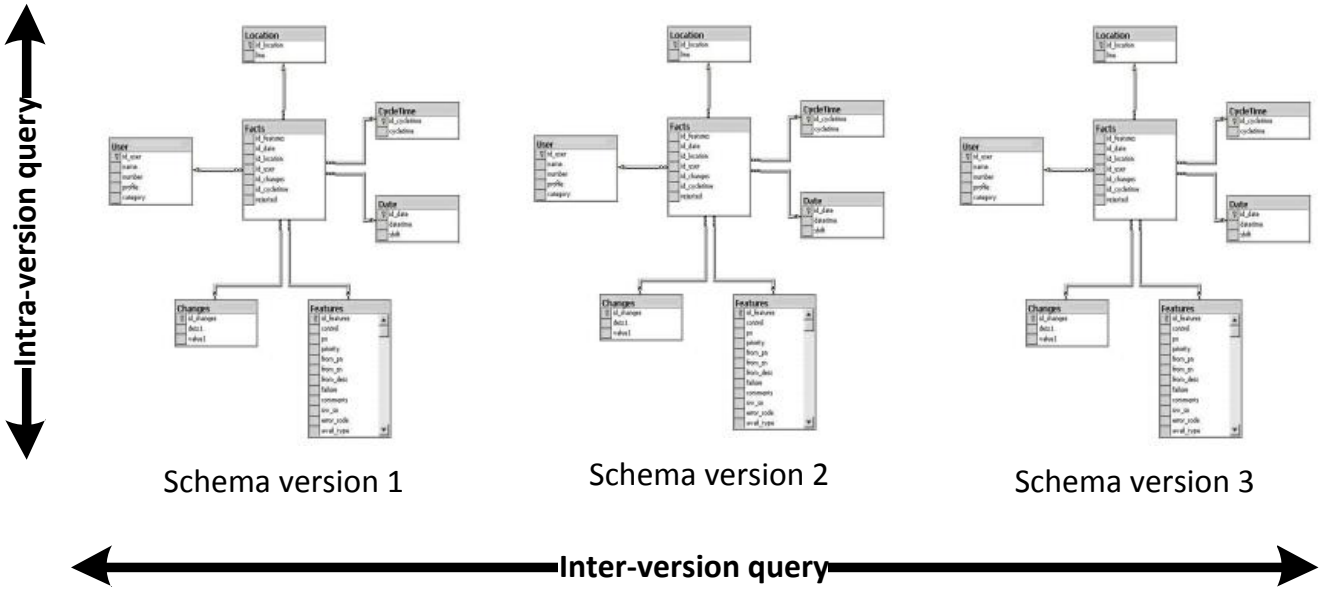
Figure: Three versions of a DW and possible cases for a query computing the value of a schema element present in the current version only

Outline

Project Goal

Design, *implement*, and *evaluate* a multiversion data warehouse that can:

- Store temporal data versions in a each schema version
- Efficiently query the data stored within a schema version (intra-version)
- Efficiently query the data stored across multiple schema versions (inter-version)



Outline

Challenges

- Modelling
  - Schema modification operators (SMOs) to derive schema versions
  - A meta-model to store the details of the multiple schema versions
  - Integrity constraints to ensure inter-schema and intra-schema data consistency
- Performance
  - No comprehensive study of temporal algebraic operators for multidimensional model with temporal and multiversion functionality
- Storage
  - Time varying aggregates are complex e.g. coalescing
  - Data redundancy overhead in case of multiple versions
  - An efficient storage model for temporal and multiversion data

Challenges (Cont.)

- Querying
  - The temporal data stored in multiple schema versions is not trivial
  - Missing data and data with different structure across multiple schema versions
- Technological challenge
  - Limited support for the temporal features in the SQL-Standards and thus in commercial systems

Outline

Our Approach

1. Model that supports
   - Temporal data in schema versions
2. Query mechanism that supports
   - Inter-schema and intra-schema querying of data
3. Performance
   - Benchmark
4. Prototype
   - System as a proof of concept

Our Approach

1. Model that supports
   - Temporal data in schema versions
   - W.Ahmed, E.Zimányi, and R.Wrembel. A Logical Model for Multiversion Data Warehouses. DaWaK 2014
   - W.Ahmed, E.Zimányi, and R.Wrembel. Temporal Data Warehouses: Logical Models and Querying. EDA 2015
2. Query mechanism that supports
   - Inter-schema and intra-schema querying of data
   - W.Ahmed, E.Zimányi. On Querying Multiversion Data Warehouses. ADBIS 2015
3. Performance
   - Benchmark
4. Prototype
   - System as a proof of concept

Ongoing Work

- Extension of the multiversion DW model with SMOs and temporal features
- Extension of TPC-DS to use it as the benchmark
- Query rewriting algorithm for the cross-version queries
- Prototype

Planned Publications

1. A Query Mechanism for Multiversion Datawarehoues
   The paper will present a system capable of storing multiple schema versions of a DW and querying these versions.
   Outlet: Bulliten of the Polish Academy of Science. The proposal has already been accepted by the committee.
2. A Data Warehouse with Multiversion and Temporal Functionality: A Prototype Implementation
   The paper will present the details of prototype system implementation and empirical results obtained from running this application on the reference data set.
   Outlet: ACM SIGMOD (SIGMOD 2016).
   Expected Submission Deadline: December 2015.

Planned Publications (cont.)

3. Modelling Data Warehouses with Multiversion and Temporal Functionality

   The paper, with an aim to be published in a journal, will detail all aspects of the doctoral project.

   1. **Outlet**: Information Systems Journal (IS).
   2. **Outlet**: Information and Software Technology (IST).
      Expected Submission Deadline: March 2016.

Plan for Fall 2015 - Spring 2016

#	Activity	Time
1	Return to the home university (ULB)	September 2015
2	Paper presentation in ADBIS 2015	September 2015
2	Submission of the first journal paper	October 2015
3	Preparation and submission of conference paper	December 2015
4	Preparation and submission of the second journal paper	March/April 2015
5	Thesis writing	May, June, July 2015
6	Thesis defense	August/September 2015

Outline

Conclusions

- Maintaining the history of changes in the content and structure is an important, yet a challenging issue in the field of data management
- The temporal and multiversion features allow the user to re-create the state of the business retrospectively or prospectively
- A natural solution to the challenge of the evolution of content and structure is to combine the both, temporal and multiversion features as a single solution
- For this, a model, efficient data storage, retrieval and querying mechanism are needed
- This project aims to address the challenges and present its outcome as a proof of the concept application