Schema Coercion: Using Database Meta-Information to Facilitate Data Transfer

A Dissertation Defense

By

Terence Critchlow

THE UNIVERSITY OF UTAH

Outline

- **Problem Statement**
- **Previous Work**
- **My Approach**
- **Results**
- **Conclusions**
- **Questions & Answers**

- **Demo**

Problem Statement

Need to transfer data between databases represented in heterogeneous database systems with minimal user interaction.

This requires identifying corresponding concepts, and mapping between the associated constructs.

Why Is This Difficult?

Concepts are viewed in different ways resulting in:

- **Naming Conflicts**
  - **synonym: two constructs named differently represent the same concept**
  - **homonym: two constructs named the same represent different concepts**
- **Structural Conflicts**
  - **the representation is based on the DBA's view of the concept**
    - **» different views result in different representations**
- **Semantic Conflicts**
  - **related constructs use different semantics**
    - **» different units (US vs. Canadian dollars)**
    - **» different scales (1 vs. 1000)**
    - **» different domains (men vs. people, students vs. CS students)**

Basic Approach

- **Obtain database schemata in usable format**
  - **automatically loaded from database**
- **Compare schemata to identify correspondences**
  - **automatic identification**
  - **additional information about data (hints) may be provided**
- **Resolve conflicts**
  - **arbitrarily complex transformations**
- **Generate translation program**
  - **automatically generated from transformations**
- **Transfer data**

Previous Work

- **Bright: resolve naming conflicts using a thesaurus**
  - **does not work in specialized domains or where terminology is ambiguous**
- **Spaccapietra: resolve complex structural conflicts**
  - **returns the most general schema capable of representing the data**
  - **requires:**
    - **» all correspondences to be identified in advance**
    - **» schemata to be represented in a special format**
- **Sciore: resolve semantic conflicts by schema enrichment**
  - **all databases must contain meta-information**
  - **meta-information specification tailored to relational databases**

Restrictive solutions which do not address entire process

Known Database Systems

Schemata from known systems are automatically read and converted to an ER representation.

- **Known systems:**
  - **relational databases**
    - » **Sybase**
  - **complex file representations**
    - » **ASN.1**
  - **flat file representations**
    - » **GDB flat file format**
    - » **table dumps**
- **Other database systems may be added as required**

Identifying Correspondences

- **A heuristic algorithm determines the likelihood that two constructs represent corresponding concepts**
- **If this confidence is sufficiently high, a *conversion* is created**
  - **conversions are not automatically created between attributes**
- **Conversions define the mapping between constructs**
  - **the confidence in the correspondence**
  - **the correspondence associated with each reference attribute**
- **Absence of a conversion implies the constructs are not related**
- **Conversions are used to identify other correspondences**

Matching Attributes

- **The likelihood two attributes represent the same concept is based on a comparison of several characteristics**
  - **the name**
    - » **comparison made using the Smalltalk string matching method**
  - **the datatype**
  - **the size**
  - **the primary and foreign key characteristics**
- **In order for a correspondence to be identified, this confidence must be above a threshold value**
- **Attribute correspondences form the basis for identifying entity and relationship correspondences**

Matching Entities and Relationships

- **Confidence is based on whether:**
  - **the construct names match**
  - **the construct types are the same (i.e. both entities)**
  - **the reference attributes have high confidence correspondences**
- **Existing conversions may also be taken into account**
  - **allows complex conflicts to be resolved**
  - **the correspondence confidence is based on the confidences of the associated conversions**
  - **for example: A relationship (r1) may correspond to a relationship-entity-relationship (r2-e1-r3) structure if the entities connected by r1 have corresponding entities in the other database, and these entities are related to e1 by r2 and r3**

Annotations

- **Annotations associate meta-information with a database**
  - **independent of the DBMS used**
  - **association lists relate annotations to the appropriate constructs**
- **The following additional information can be specified:**
  - **aliases**
    - » **rename constructs to avoid name conflicts**
  - **datatypes**
    - » **refine type information to reduce type and semantic conflicts**
  - **attribute transformations**
    - » **defaults when correspondences can not be identified**
    - » **mandatory**
    - » **between different datatypes**