Tools and Techniques for Understanding Data Exchange Systems

Wang-Chiew Tan
UC Santa Cruz
Information Integration Systems

- Integrating data or converting data from one representation to another typically requires a huge investment of time and effort.
- Not easy to understand.
A simple example: Mapping two university schemas
One interpretation of the value correspondences

The underlying XQuery implementation of the value correspondences
An alternative interpretation of the value correspondences

```xml
$xll in $xoll/courses
where
  $xl/cid/text() = $xll/cid/text() and
  $x0/university/text() = $xll/university/text()
return
  <students>
    <sid> ($xoll/sid/text()) </sid>
    <grade> ($xll/grade/text()) </grade>
  </students>
</courses>

{ distinct-values 
  for 
  $x0 in $doc0/students-info/students
  return
  <courses-info>
    <cid/>
    <cname/>
    <university> ($x0/university/text()) </university>
    { distinct-values 
      for 
      $x0l in $doc0/students-info/students
      where
        $x0/university/text() = $x0l/university/text()
      return
        <students>
          <sid> ($x0l/sid/text()) </sid>
          <grade/>
          <students>
        </students>
      </students-info>
```
XQuery

|-- Input XML file path should be the parameter of document() --|

```xml
let $doc0 := document("input XML file goes here")
return
<mondial>
  |
  |  distinct-values
  |  for
  |  | $x0 in $doc0/mondial/country,
  |  | $x1 in $x0/encompassed,
  |  | $x2 in $x0/mondial/continent,
  |  | $x3 in $doc0/mondial/organization
  |  where
  |  | $x0/membership/text() = $x3/id/text() and
  |  | $x2/id/text() = $x1/continent/text()
  |  return
  |  <CONTINENT>
  |  | <NAME> ($x2/name/text())</NAME>
  |  | <AREA> ($x2/area/text())</AREA>
  |  | </CONTINENT>
  |
  |  distinct-values
  for
  | $x0 in $doc0/mondial/organization,
  | $x1 in $x0/members,
  | $x2 in $x0/mondial/country,
  | $x3 in $doc0/mondial/organization
  where
  | $x2/membership/text() = $x3/id/text() and
  | $x1/id/text() = $x0/country/text()
  return
  <ORGANIZATION>
  | <ABBREVIATION> ($x0/abbreviation/text())</ABBREVIATION>
  | <NAME> ($x0/name/text())</NAME>
  | <CITY> ($x0/city/text())</CITY>
  | <COUNTRY/>
```

File: mondial/mondial-small_map_vcs.xsml
Not peculiar to Clio

Commercial integration/exchange tools
- Altova’s Mapforce, Data Direct’s Stylus Studio, Microsoft Biztalk mapper, BEA’s Aqualogic Data Services Platform, IBM’s Rational Data Architect

Value correspondences / schema mappings form a fundamental component in integration tools

Tools/Techniques for understanding them
- Bypass implementation details as much as possible
Value correspondences / Schema mappings

- **Value correspondences**: the associations between elements of the source and target schema
- **Schema mapping**: a specification of how data structured under the source schema is to be converted into data structured under the target schema
First example

- Data-driven understanding and refinement of schema mappings
  - L. Yan, R. Miller, L. Haas, R. Fagin (SIGMOD 2001)

- Goal: to help a user understand different schema mappings and choose the desired one

- Use selective samples of source data to illustrate the different behaviors of different schema mappings
Second example

SPIDER: a Schema mapPIng DEbuggeR

- B. Alexe, L. Chiticariu, (VLDB 2006)

Goal: to allow a user to explore and understand a schema mapping.

One functionality of SPIDER

- Understand the specification by examining the relationship between input (source data) and output (target data)
Challenges

- During debugging, the schema mapping and source instance is often modified
- Reflect the “cause-and-effect” of modifications
  - Adapt the target instance when changes are made to the schema mapping
  - Adapt the target instance when changes are made to the source instance
Challenges

- Tools for understanding peer data management systems, Extract-Transform-Load (ETL) tools
  - Chains of mappings, different models

- How can we leverage techniques for computing provenance to make II systems more user-friendly?

- What can we do to make information integration systems more user-friendly?