CSPP 53017: Data Warehousing
Winter 2013

Lecture 6
Svetlozar Nestorov

Class News

- Homework 4 is online
  - Due by Tuesday, Feb 26.
- Second 15 minute in-class quiz today at 6:30pm
  - Open book/notes
- Last 15 minute in-class quiz will be on Mar 5.
ETL Design and Development

ETL Systems

- **Source Systems**
  - Operational Databases, Flat Files, ODS, ERP systems (divided into modules that cover major functional areas of the business, such as HR, manufacturing, etc.), Reporting Instances, Archives, External Data.

- **ETL Area**
  - Place where mapping (from source systems to data warehouse) takes place.
  - Assembly plant, not intended to be seen by users.

- **Presentation Server**
  - Target platform where data warehouse data is stored.
ETL Processes

• ETL: Extraction/Transformation/Load
  – Must result in data that is relevant, useful, high-quality, accurate, and accessible

ETL Processes: Extraction

• Pulling selected data (that pertains to the subject areas of the data warehouse) from the source systems
• Often the largest single effort in the data warehouse project (rule of thumb: 60% of the data warehouse development hours are spent on the extract process), especially if the source systems are legacy, old, mainframe-based, etc.
• Challenge: determining what data to extract and what kinds of filters to apply.
ETL Processes: Transformation

- Transforming data from source systems into data suitable for end user query and analysis application.
- Transformation cleans-up, standardizes, and restructures (as subject-oriented) operational data.
- Quality data is the key to a successful DW; it is better to have no data at all than bad data.

ETL Processes: Load

- Loading data into the warehouse and refreshing the warehouse with updated data.
- Complications:
  - System or network failure may result in partial loads
  - Load auditing and verification
  - Data type mismatches
  - Rejected data
- Test load in a development (duplicate) environment before running in production.
Examining Data Sources

• Production Data
  – Flat files, database systems (e.g. Oracle, IBM DB2, …),
    vertical applications (e.g. Oracle Financials), other (e.g.
    spreadsheets, word documents, …)

• Archive Data
  – Supplies historical data
  – Used for the initial DW implementation (first-time load).
  – Not used for regular data refreshes

• External Data
  – Information form outside the organization (e.g. periodicals
    and reports, syndicated data feeds, competitive analysis
    information, purchased marketing-competitive-customer
    related data, free web-based data, weather reports, etc.)
  – Issues of frequency, format, and predictability.

Extraction and Mapping

• Extraction Techniques
  – Programming (C, C++, Java, PL/SQL, etc.)
  – Tools
    • High initial cost, but a benefit of ongoing automation as well.
    • Functionalities
      – Storing a physical definition of the source and DW data
      – Generate data conversion programs
      – Clean and transform data
      – Allow selective retrieval
      – Maintain metadata
    • Two options
      – In-house developed tools
      – Vendor tools

• Mapping
  – Defines which operational attributes to use and how.
  – Mapping tools are available
Source-to-Target Mapping

- Source-to-target data map is the foundation for the development of the data staging process
- Source-to-target data map contains
  - Target Table Name
  - Target Column Name
  - Target Column Data Type
  - Target Column Length
  - Source System
  - Source Table/File
  - Source Table/File Column/Field
  - Data Transform Notes
  - Dimension/Data Mart
  - Attribute/Fact

### Source-to-Target Data Map

<table>
<thead>
<tr>
<th>Target Table</th>
<th>Target Column</th>
<th>Data Type</th>
<th>Len</th>
<th>Target Column Description</th>
<th>Src Systm</th>
<th>Src Table / File</th>
<th>Src Col / Field</th>
<th>Data Txfm Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Dimension</td>
<td>CUST_Key</td>
<td>Num</td>
<td>8</td>
<td>PK. for Cust. Dimension</td>
<td>New</td>
<td>New</td>
<td>New</td>
<td>Create</td>
</tr>
<tr>
<td>Customer Dimension</td>
<td>CUST_ID</td>
<td>Char</td>
<td>11</td>
<td>Operational Key for Cust.</td>
<td>OPS10</td>
<td>CUST_MAST</td>
<td>Cid</td>
<td>Direct</td>
</tr>
<tr>
<td>Customer Dimension</td>
<td>CUST_FNAME</td>
<td>Char</td>
<td>15</td>
<td>Customer First Name</td>
<td>OPS10</td>
<td>CUST_MAST</td>
<td>CFull Name</td>
<td>ParseOut Before 1st Space</td>
</tr>
<tr>
<td>Customer Dimension</td>
<td>CUST_LNAME</td>
<td>Char</td>
<td>25</td>
<td>Customer Last Name</td>
<td>OPS10</td>
<td>CUST_MAST</td>
<td>CFull Name</td>
<td>ParseOut After Last Space</td>
</tr>
</tbody>
</table>

...
Transformation Guidelines

• Quality (Clean) Data essential for:
  – Targeting customers, determining buying patterns, matching customers, identifying householders (private and commercial), identifying history, etc, …

• Guidelines
  – Operational data should not be used directly in the warehouse.
  – Operational data must be cleaned for EACH increment of the DW.
  – Operational data is not simply fixed by modifying operational systems.

Transformation Techniques

• Transformation Techniques
  – Programming (C, C++, PL/SQL, etc.)
  – Tools (In-house developed and/or specialized vendor tools)

• Transformation Routines
  – Cleaning data (A.K.A. data cleansing or scrubbing)
  – Adding elements
  – Merging rows or records in files
  – Integrating data into files and formats to be loaded into the DW
Source Data Anomalies

- No unique key, data naming and coding anomalies, data meaning anomalies, spelling and text inconsistencies, etc.
- Examples

<table>
<thead>
<tr>
<th>CUSNUM</th>
<th>NAME</th>
<th>ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>9575</td>
<td>Oracle Corp.</td>
<td>100 NE 1st Street, Tampa</td>
</tr>
<tr>
<td>9575</td>
<td>Oracle</td>
<td>100 NE. First St., Tampa</td>
</tr>
<tr>
<td>9475</td>
<td>Oracle Services</td>
<td>100 North East 1 St., Tampa FL</td>
</tr>
</tbody>
</table>

- Reasons: data and systems developed over many years, no consistent policies, ...

Source Data Problems 1

- Multipart Keys e.g. Product Code = 12M654141232
  - Solution: program or tool capable of identifying on a position-by-position basis the individual values, length of value, and meaning of resulting information.

- Multiple encoding e.g. male, female or m,f or 0.1
  - Solution: program or tool capable of identifying all the distinct possibilities, e.g.: if field in (‘male’, ‘m’, 0) then new value = ‘m’;

- Erroneous data e.g. mle, female or null, 1
  - Solution: program or tool capable of identifying spurious and bad entries and changing them into appropriate values.

- Multiple local standards: metric/USA, currencies
  - Solution: tools and filters that preprocess data into a suitable format
Source Data Problems 2

• **Missing Values**
  Solutions: ignore the missing data, wait until entered, …

• **Duplicate Values**
  Solution: duplicate values must be eliminated by e.g. using standard SQL UNION operator.

• **Element Names Problems**
  Solution: agree on standardization and re-name.

• **Element Meaning Problem**:
  Solution: Document the meaning in metadata.

• **Referential Integrity Problem**:
  Solution: Clean data and enforce referential integrity constraints.

Example: Name and Address

• No unique key
• Missing values
• Personal and commercial names mixed
• Different address for same customer
• Different names and spelling for same customer
• One name on multiple lines
• Many names on one line, e.g.

<table>
<thead>
<tr>
<th>Name</th>
<th>Location_id</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe Smith</td>
<td>N100</td>
</tr>
<tr>
<td>Tina Lewis</td>
<td>F101</td>
</tr>
<tr>
<td>Andy and Ann Jones</td>
<td>M300</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>

• Single vs. Multiple Field format
e.g. Name, Location Vs. Name, Street, City, Zipcode, County
Solutions

- Create atomic values
- Standardize formats
- Verify data accuracy
- Match with other records
- Identify private and commercial addresses and inhabitants
- Document in metadata
- May require sophisticated tools and techniques

Merging Data

- Operational transactions usually do not map one-to-one with warehouse data
- Data for the warehouse is merged to provide information for analysis

<table>
<thead>
<tr>
<th>Type</th>
<th>Date</th>
<th>Time</th>
<th>Item</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:01</td>
<td>Ham Pizza</td>
<td>$12.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:02</td>
<td>Cheese Pizza</td>
<td>$10.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:03</td>
<td>Veggie Pizza</td>
<td>$120.00</td>
</tr>
<tr>
<td>Cancel</td>
<td>10/2/2001</td>
<td>12:00:04</td>
<td>Veggie Pizza</td>
<td>-$120.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:05</td>
<td>Veggie Pizza</td>
<td>$12.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:06</td>
<td>Cheese Pizza</td>
<td>$10.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Date</th>
<th>Time</th>
<th>Item</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:01</td>
<td>Ham Pizza</td>
<td>$12.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:02</td>
<td>Cheese Pizza</td>
<td>$10.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:05</td>
<td>Veggie Pizza</td>
<td>$12.00</td>
</tr>
<tr>
<td>Sale</td>
<td>10/2/2001</td>
<td>12:00:06</td>
<td>Cheese Pizza</td>
<td>$10.00</td>
</tr>
</tbody>
</table>
More Transformation Details

- Adding a Date (Time) Stamp
- Adding (DWH) Keys to Data
- Summarizing Data
- Maintaining Transformation Metadata
  - Information on how to perform key restructuring
  - Logic to eliminate different coding methods and data values, parsing rules
  - Logic to detect multiple source files
  - Logic and exception rules to handle null, negative, and default values and to eliminate and consolidate duplicate values
  - Input or language formats, conversion algorithms, data standardization rules
  - Logic and programs used to create summary data
  - Transformation frequency, program name, location
  - Temporary extraction storage name and location.

Load (Transportation)

- Loading moves the data into the warehouse
- Can be time-consuming
  - Time period for load (load window) should be known
  - All load processes should be automated
  - Loading should be scheduled and prioritized
- DW Processing Environment
  - Build a new database
  - After each time interval, add changes to database
  - Archive or purge oldest data
- First-Time Load
  - Initial load moves large volumes
- Refresh
  - Less data to load
  - Business determines the refresh cycle (refreshing is often done overnight)