Data Warehouses: Many of the Common Failures

3 May 1999

Patricia L. Carbone
carbone@mitre.org
Overview

- Transaction databases
- Data warehouses
- Data marts
- Comparison of the three
- Examples of data warehousing failures
- OLTP versus OLAP
- Examples of relational and multidimensional modeling
- Summary: Classic mistakes in data warehousing
Transaction Databases

Data Warehouse Architecture

Getting Data In

IT Users

Operational Users

Data Transformation

Metadata

Data Analyst
Database Admin
Operations Manager
Network Admin
Applications Developer

Columbus ECPN
Ogden ECPN
Slidell ECPN
CCR

Accessing
Capturing
Extracting
Filtering
Scrubbing
Reconciling
Conditioning
Condensing
Householding
Loading
Transaction Database Definition

A transaction-oriented, non-integrated, time-invariant, and volatile collection of data in support of business operations.

- Focus on a business transaction, not the users
- Stable design because business processes more stable than user needs
- No data correlation across systems
- May have duplicates
- Quality assurance not fundamental
- A snapshot of a moment in time, rather than a history of data over time
- Repetitions of the same query can give different results
- Constantly updated as transactions occur
- Focus on day-to-day operations, not long-term planning
Transaction Database (continued)

- **Application Types:** On-Line Transaction Processing (OLTP)

- **Examples**
  - **Existing**
    - CCR
    - ECI
  - **Potential**
    - ECI Transaction Archive
    - Transaction Billing System

- **Sample Usages**
  - Query for failed transaction
  - Issue bill to government agency
Data Warehouse Definition

A subject-oriented, focus on a subject as defined by users
• Contains all data needed by the users to understand the subject
• Users change requirements rapidly

integrated,
• Data combined across systems and transactions
• No duplicates
• Quality assured.

time-variant,
• A history of the subject over time, not a single moment in time
• Doesn’t change while a query is running

and non-volatile
• Focus on planning for the future, not on day-to-day operations

collection of data in support of management’s decision-making process.
Data Warehouse (continued)

- **Application Types**
  - On-Line Analytical Processing (OLAP) e.g. purchasing analyzer
  - Data Mining e.g. AT&T Worldnet’s problem predictor

- **Examples**
  - Existing
    - MCI Network Usage Data Warehouse
    - Wal-Mart
    - Citibank
  - Potential
    - DOD Procurement Data Warehouse
Data Warehouse (continued)

● Sample Usage
  – Alert for channel nearing capacity
  – Identify causes of network problems
  – Allocate budget underrun
  – Alert for budget overrun
Process Flow

- Data flow from transaction database to data warehouse
  - Input transaction data into transaction database
  - Access, capture, extract, filter, scrub, reconcile, condition, and condense data
  - Aggregate data
  - Load sanitized and aggregated data into data warehouse
  - Load applicable data into a data mart

- Metadata guides the complete process

- 80% of effort for data warehouse is for loading data (the full extract/clean/load process)
Data Mart Definition

A single department

- Small, cohesive group of users

data warehouse

- <See previous slides>

containing a small number of subject areas

- Very limited data set
Data Mart (continued)

- **Scope**
  - Very limited
  - Frequently defined using time-box techniques. *(Time boxing is a project management techniques focusing on 3 key project dimensions (time, resources and scope), only one of which can vary for a project. For data marts, time and resources are invariant, while scope can vary.)*

- **Examples**
  - Customer Service Data Mart
  - Quotes Data Mart

- **Application Types**
  - OLAP
  - Data Mining

- **Can be precursor or successor to data warehouse**
### Transaction Database, Data Warehouse & Data Mart

<table>
<thead>
<tr>
<th>Objective</th>
<th>Transaction Database</th>
<th>Data Warehouse</th>
<th>Data Mart</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pull data in for transaction processing</td>
<td>Push data out to decision makers</td>
<td>Push data out to decision makers</td>
</tr>
<tr>
<td>Focus</td>
<td>Transactions</td>
<td>Subjects of interest to an enterprise</td>
<td>Subjects of interest to a department</td>
</tr>
<tr>
<td>Ownership</td>
<td>Fiefdom</td>
<td>Enterprise</td>
<td>Enterprise</td>
</tr>
<tr>
<td>Consistency</td>
<td>Microscopic (transaction level)</td>
<td>Global (enterprise level)</td>
<td>Global (department level)</td>
</tr>
<tr>
<td>Transaction Scale</td>
<td>Atomic (record level)</td>
<td>Summary of enterprise’s data</td>
<td>Summary of department’s data</td>
</tr>
<tr>
<td>Users</td>
<td>“Turn the wheels of the organization” e.g. sysops</td>
<td>“Watch the wheels of the organization” e.g. upper management</td>
<td>“Watch the wheels of the organization” e.g. middle management</td>
</tr>
</tbody>
</table>
### Transaction Database, Data Warehouse & Data Mart (continued)

<table>
<thead>
<tr>
<th></th>
<th>Transaction Database</th>
<th>Data Warehouse</th>
<th>Data Mart</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number of Users</strong></td>
<td>Many</td>
<td>Few</td>
<td>Very few per data mart</td>
</tr>
<tr>
<td><strong>Type of Usage</strong></td>
<td>Data entry; single record lookup</td>
<td>Periodically changing summaries of multiple records</td>
<td>Periodically changing summaries of multiple records</td>
</tr>
<tr>
<td><strong>Time Dimension</strong></td>
<td>Instantaneous snapshot, constantly updated</td>
<td>Static snapshot, regularly but infrequently updated</td>
<td>Static snapshot, regularly but infrequently updated</td>
</tr>
<tr>
<td><strong>Data Modeling Technique</strong></td>
<td>Entity relationship</td>
<td>Dimensional</td>
<td>Dimensional</td>
</tr>
</tbody>
</table>
Definition of Warehouse “Failure” (or “Disappointment”)

- Warehouse does not meet the expectations of those involved
- Warehouse was completed, but was severely over budget in relation to time, money, or both
- Warehouse failed one or more times but eventually was completed
- Warehouse failed with no effort to revive it

Excellent article: “Data Warehousing Failures: Case Studies and Findings” in Journal of Data Warehousing, Vol. 4, No. 1, Spring 1999
Example 1: Large Retailer (1 of 2)

● Version 1
  – Warehouse project begun
  – Became apparent that project would take much more time than originally planned
  – Hardware was not able to handle the volume of data
  – Software could not handle the data; vendor dropped support for the software
  – Upper management became disillusioned and halted the project

● Version 2
  – Now focusing on subject area data mart
  – Have plans to add additional subject areas until create the enterprise-wide warehouse
Observations from project manager

- Management of expectations is critical to any sizeable data warehouse
- Proven technology makes the project easier (but is not essential)
- Construction of a sizeable data warehouse should be treated more like an R&D project, or go with a data mart approach
Example 2: Government Research Laboratory (1 of 2)

● Description
  – 15 laboratories each have finance department reporting to national office
  – All data stored via COBOL
  – If reports differed from standard, would need IS support to generate new report

● Solution 1
  – Construct data warehouse oriented to finance department
  – Assigned 2 people full time to build warehouse in 4 months
  – In timeframe, passed summary data to warehouse - access via PowerBuilder
  – Simultaneously, mainframe system was drastically modified - not in alignment with data warehouse project
  – Data warehouse became end goal - modifications and extensions after initial version were not allocated for
  – No solution to original problems
Example 2: Government Research Laboratory (2 of 2)

- **Solution 2**
  - Began 3 years after first attempt
  - Project manager lined up funding to enable solving multiple problems
  - Access to data warehouse via web-based reports

- **Observations**
  - Warehouse initiative should have been done with the mainframe restructuring
  - Planning and resourcing needed to be projected further into the future
  - A pilot might have identified a number of technical problems
  - Reasonable deadlines
  - “It could have been done right ... for the right reasons”
Example 3: North American Federal Government (1 of 2)

● Description
  – Proposal put forth for data warehouse at a cost of $800,000 taking 8 months to build
  – IT department assumed proposal was accepted, but did not wait for concurrence from business unit (who was supposed to provide $ and manpower)
  – Actual time spent: 2 years

● Problems
  – Business unit stretched the detailed data analysis from 1.5 months to 9 months
  – Scope creep - planned users for system grew from 250 to 2500
  – Acquiring correct technological tools took formal approval process exceeding 1 year
  – 3 weeks prior to delivery, IT director canceled the project
Example 3: North American Federal Government (1 of 2)

- **Problems (concluded)**
  - 6 weeks after cancellation, new interest in populating the warehouse was generated - nothing delivered
  - Final cost - $2.5 million

- **Observations**
  - Lack of focus of project - business unit could not identify scope of project
  - Milestones were pushed back, implying that project was not urgent or important
  - Negative internal politics - business leader did not allow project analysts to talk to end users; business leader reassigned IT staff without telling IT project lead
### Reasons for Data Warehousing Failures

<table>
<thead>
<tr>
<th>Reason for failure</th>
<th>Large Retailer</th>
<th>Government Research Lab</th>
<th>N.A. Federal Government</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weak sponsorship</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poor choice of technology</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wrong project scope</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data problems</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Problems with end user tools</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Insufficient funding</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Scope creep</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Organization politics</td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
Some Basic Questions

- Why are you building your warehouse?
- Who will use the warehouse?
  - The entire enterprise
  - One particular department
- What is the goal of the warehouse?
  - To provide a historical perspective of the aggregated data
- What kind of a data model do you expect to use?
  - Relational
  - Multidimensional
- What kind of analysis do you expect your users will need?
  - OLAP
  - Data mining
Techniques for Using Data

- **Reporting**: repeatable, preferably through agents and alerts
- **OLAP**: On-Line Analytical Processing - exploratory and hypothesis testing
- **Data mining**: hypothesis formation
- **Groupware**: publish, discuss, share
- **Group Decision Support Systems**: collect divergent thought, categorize for convergence, generate consensus
Processing OLAP data (1 of 2)

- **Relational database**
  - Not the obvious choice to perform complex multidimensional calculations
  - Complex multi-pass SQL is necessary to achieve more than the most trivial functionality
  - Tools can have limited range of calculations in SQL, with results being used as input by a multidimensional engine on the client or mid-tier server

- **Multidimensional server engine**
  - Most obvious and popular place to perform calculations
  - Good performance - engine and database can be optimized to work together
  - Plenty of memory on a server enables large scale array calculations to be performed efficiently
Processing OLAP data (2 of 2)

- Client
  - Vendors aiming to take advantage of desktop PC power to perform multidimensional calculations
  - Popularity of thin clients is requiring that vendors move most of the client-based processing to new Web application servers
Comparison to OLTP (Transaction Processing)

- OLTP applications typically have many users creating, updating, or retrieving *individual* records
- OLAP applications are used by analysts and managers wanting historical, aggregated views of the data

<table>
<thead>
<tr>
<th>OLTP (Relational)</th>
<th>OLAP (Multidimensional)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Atomized</td>
<td>• Summarized</td>
</tr>
<tr>
<td>• Present</td>
<td>• Historical</td>
</tr>
<tr>
<td>• Record-at-a-time</td>
<td>• Many records at a time</td>
</tr>
<tr>
<td>• Process oriented</td>
<td>• Subject oriented</td>
</tr>
</tbody>
</table>
More OLAP vs OLTP

**OLTP**
- Real time, read/write to corporate data stores
- Many simultaneous internal and external users
- Short, repetitive, simple processing tasks
- Supports commerce and monitoring
- Integrity and guaranteed completion of tasks
- Fixed, well defined processes with few if any exceptions

**OLAP**
- As long as it takes, read-only access to corporate data stores
- Small number of primarily internal users
- Long, often unique, process intensive tasks
- Supports decision making and discovery
- Accuracy and completeness of information and results
- Ad hoc explorations as well as fixed reports
Relational versus 2-Dimensional: A Simple Example

**Relational Representation**

<table>
<thead>
<tr>
<th>Cargo</th>
<th>Port</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hogs</td>
<td>Singapore</td>
<td>50</td>
</tr>
<tr>
<td>Hogs</td>
<td>New Orleans</td>
<td>60</td>
</tr>
<tr>
<td>Hogs</td>
<td>Perth</td>
<td>100</td>
</tr>
<tr>
<td>Cars</td>
<td>Singapore</td>
<td>40</td>
</tr>
<tr>
<td>Cars</td>
<td>New Orleans</td>
<td>70</td>
</tr>
<tr>
<td>Cars</td>
<td>Perth</td>
<td>80</td>
</tr>
<tr>
<td>Oil</td>
<td>Singapore</td>
<td>90</td>
</tr>
<tr>
<td>Oil</td>
<td>New Orleans</td>
<td>120</td>
</tr>
<tr>
<td>Oil</td>
<td>Perth</td>
<td>140</td>
</tr>
<tr>
<td>Corn</td>
<td>Singapore</td>
<td>20</td>
</tr>
<tr>
<td>Corn</td>
<td>New Orleans</td>
<td>10</td>
</tr>
<tr>
<td>Corn</td>
<td>Perth</td>
<td>30</td>
</tr>
</tbody>
</table>

**2-Dimensional Representation**

<table>
<thead>
<tr>
<th></th>
<th>Singapore</th>
<th>New Orleans</th>
<th>Perth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hogs</td>
<td>50</td>
<td>60</td>
<td>100</td>
</tr>
<tr>
<td>Cars</td>
<td>40</td>
<td>70</td>
<td>80</td>
</tr>
<tr>
<td>Oil</td>
<td>90</td>
<td>120</td>
<td>140</td>
</tr>
<tr>
<td>Corn</td>
<td>20</td>
<td>10</td>
<td>30</td>
</tr>
</tbody>
</table>

Query 1: How much oil is shipped from Singapore?  
Query 2: What is the total weight shipped from Perth?
Consolidation (or Pre-Aggregation): Relational versus 2-Dimensional

<table>
<thead>
<tr>
<th>Cargo</th>
<th>Port</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hogs</td>
<td>Singapore</td>
<td>50</td>
</tr>
<tr>
<td>Hogs</td>
<td>New Orleans</td>
<td>60</td>
</tr>
<tr>
<td>Hogs</td>
<td>Perth</td>
<td>100</td>
</tr>
<tr>
<td>Hogs</td>
<td>Total</td>
<td>210</td>
</tr>
<tr>
<td>Cars</td>
<td>Singapore</td>
<td>40</td>
</tr>
<tr>
<td>Cars</td>
<td>New Orleans</td>
<td>70</td>
</tr>
<tr>
<td>Cars</td>
<td>Perth</td>
<td>80</td>
</tr>
<tr>
<td>Cars</td>
<td>Total</td>
<td>190</td>
</tr>
<tr>
<td>Oil</td>
<td>Singapore</td>
<td>90</td>
</tr>
<tr>
<td>Oil</td>
<td>New Orleans</td>
<td>120</td>
</tr>
<tr>
<td>Oil</td>
<td>Perth</td>
<td>140</td>
</tr>
<tr>
<td>Oil</td>
<td>Total</td>
<td>350</td>
</tr>
<tr>
<td>Corn</td>
<td>Singapore</td>
<td>20</td>
</tr>
<tr>
<td>Corn</td>
<td>New Orleans</td>
<td>10</td>
</tr>
<tr>
<td>Corn</td>
<td>Perth</td>
<td>30</td>
</tr>
<tr>
<td>Corn</td>
<td>Total</td>
<td>60</td>
</tr>
<tr>
<td>Total</td>
<td>Singapore</td>
<td>200</td>
</tr>
<tr>
<td>Total</td>
<td>New Orleans</td>
<td>260</td>
</tr>
<tr>
<td>Total</td>
<td>Perth</td>
<td>350</td>
</tr>
<tr>
<td>Total</td>
<td>Total</td>
<td>810</td>
</tr>
</tbody>
</table>

2-Dimensional Representation

<table>
<thead>
<tr>
<th></th>
<th>Singapore</th>
<th>New Orleans</th>
<th>Perth</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hogs</td>
<td>50</td>
<td>60</td>
<td>100</td>
<td>210</td>
</tr>
<tr>
<td>Cars</td>
<td>40</td>
<td>70</td>
<td>80</td>
<td>190</td>
</tr>
<tr>
<td>Oil</td>
<td>90</td>
<td>120</td>
<td>140</td>
<td>350</td>
</tr>
<tr>
<td>Corn</td>
<td>20</td>
<td>10</td>
<td>30</td>
<td>60</td>
</tr>
<tr>
<td>Total</td>
<td>200</td>
<td>260</td>
<td>350</td>
<td>810</td>
</tr>
</tbody>
</table>
Moving to Multiple Dimensions with Hierarchy

Region Total

Indochina
- Indonesia
- Singapore
- Thailand

Australia

North America
- Canada
- United States
  - New Orleans
  - New York
  - Oakland

Canada

United States

Indonesia

Singapore

Thailand

New Orleans

New York

Oakland
Multidimensional with Hierarchical (and Drill Down)

Cargo Dimension

Region Dimension

Indochina
- Indonesia
- Singapore
- Thailand

... North America
- Canada
- United States
- New Orleans
- New York
- Oakland

Time Dimension

Can now query on cities, countries, or regions
Standard SQL Approach

```sql
SELECT sum(Event.Weight) 
FROM Event, Port, Cargo 
```

- Hard to formulate
  - Who is going to write this query?
- Time consuming to compute
The OLAP Data Cube Approach

“How many tons of corn left New Orleans in April?”

“How many tons of corn left New Orleans in April?”

“21 tons”
Summary

- Problems are often organizational, not technical, in nature
- Bad warehouses are often poorly conceived, planned, and executed
- Ensure that the sponsor understands the concept of warehousing and has the necessary long-term view
- Data mart “pilots” have been successful
  - “quick hit” benefit
  - more adaptable
  - However, at some point, multiple marts should be “integrated” into a warehouse
Classic Mistakes in Data Warehousing (1 of 3)

- Up-keep of technology
- Managing multiple users with various needs
- Lack of integration/integrating data marts into data warehouses, after the fact
- Unclear business objectives; not knowing the information requirements
- Lack of effective project sponsorship
- Lack of data quality
- Lack of user input
- Unrealistic expectations - cost
Classic Mistakes in Data Warehousing (2 of 3)

- Incentives for using data marts instead of data warehouses
- Inexperienced / untrained / inadequate number of personnel
- Corporate politics
- Access to data manipulation (users should not have access) / Security
- Lack of stewardship
- Inappropriate format of information - not a single, standard format
- Up-keep of information (keeping information current)
Classic Mistakes in Data Warehousing (3 of 3)

- Unrealistic expectations - overly optimistic time schedule
- Inappropriate architecture
- Vendors overselling capabilities of products
- Lack of training and support for users
- Omitted information
- Lack of coordination (requires too much coordination)
- Cultural issues
- Using the warehouse for operational, not informational, purposes
- Not enough summarization of data