Introduction to Data Warehousing / Business Intelligence
Your Presenter

- Robert C. Cain
- Senior BI Architect COMFRAME
  - Offices in Birmingham, Nashville & Beijing
  - Microsoft Partner, 1 of 37 Nationally Managed
  - Systems Integrator
  - App Dev, SharePoint, BI, EPM
- 5 years BI at Nuclear Power Co.
- 10 years as a consultant in the B’ham Market
- Wide range of .Net applications, ASP & Win
- SQL Server Data Warehouse
- [http://arcanecode.com](http://arcanecode.com)
- [http://twitter.com/arcanecode](http://twitter.com/arcanecode)
Why learn about Data Warehousing?

- **DBA**
  - Implement new Data Warehouse project
  - Install and configure BI tools (SSIS, SSRS, SSAS)
- **DB Designer / Developer**
  - Design and script a DW
  - DW significantly different from traditional database designs.
- **Software Developer**
  - Interact with warehouses for additional data or reports
  - Data mining results into your applications
Business Intelligence is HOT

- According to Computerworld, BI is the 5\textsuperscript{th} hottest IT Skill for 2009
- Dice.com over 2,800 job openings
What is a Data Warehouse

- A giant storehouse for your data
- ALL of your data
- Aggregation of data from multiple systems
What is Business Intelligence

• Leveraging data you already have to convert knowledge into informed actions
• Providing ways to measure the health of your business
• Examining the data in your warehouse to look for three main areas of interest.
  • Aggregations
  • Trends
  • Correlations (Data Mining)
Why Have a Data Warehouse?

• Combine data from multiple systems and resolve inconsistencies between those systems
• Make reporting easier
• Reduce the load on production systems
• Provide for long term storage of data
• Provide consistency among system transitions
Some More Reasons for a Data Warehouse

- Make the data available for analysis
- Ability to apply advanced data mining tools
- To extract further value from the data you already own
  Business Intelligence
What’s wrong with reporting from a Transactional System?

- OLTP – On Line Transaction Processing
- Designed for working with single record at a time.
- Data is highly “normalized”, i.e. duplicate values have been removed.
- Getting all data for a record can involve many table joins
- Can be quite confusing for ‘ad-hoc’ reporting
- Can also be slow, having an impact on the OLTP system
What’s different about a Data Warehouse?

- Data Warehouses typically use a design called OLAP
- On-Line Analytical Processing
- Data is de-normalized into structures easier to work with.
- Number of tables are reduced, reducing number of joins and increasing simplicity
- Often a Star Schema or Snowflake Schema
Star Schema

Fact Table

Dimension
Dimension
Dimension
Dimension
Dimension
Dimension
Types of Tables in a Warehouse

- Facts
- Dimensions
- Both require the concept of Surrogate Keys
- A new key, typically some type of INT, that is used in place of any other key as the Primary Key
Reasons for Surrogate Keys

• Preserve data in case of source system change
• Combine data from multiple sources into a single table
• Source System keys can be multi-column and complex, slowing response time
• Often the key is not needed for many data warehousing functions such as aggregations
Fact Tables

- A Fact marks an event, a discrete happening in time.
- Facts join dimensions, “who” did an action (SoldBy, SoldTo) to the “what”, what object was acted upon (Product).
- Facts also hold numeric measures to quantify the fact: Quantity, SaleAmt, SaleDate.
Fact Table Example - Sales

- Employee Dimension
- Customer Dimension
- Product Dimension

<table>
<thead>
<tr>
<th>ID</th>
<th>SoldById</th>
<th>SoldToID</th>
<th>ProductID</th>
<th>Qty</th>
<th>SaleAmt</th>
<th>SaleDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>3456</td>
<td>1234</td>
<td>6789</td>
<td>987</td>
<td>3</td>
<td>156.00</td>
<td>7/17/2009</td>
</tr>
</tbody>
</table>
Dimensions

• Dimensions hold the values that describe facts
• “Look Up Values”
• Some examples: Time, Geography, Employees, Products, Customers
• When a Dimension can change over time, it’s known as a Slowly Changing Dimension
• Many types of Dimensions
Type 0 Dimension (Fixed)

- Type 0 Dimensions are also called Fixed
- For data that will not change. Ever.
- Best used for static data like colors, sizes, etc.

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Blue</td>
</tr>
<tr>
<td>2</td>
<td>Black</td>
</tr>
<tr>
<td>3</td>
<td>Green</td>
</tr>
<tr>
<td>4</td>
<td>Yellow</td>
</tr>
</tbody>
</table>
Type 1 Dimension

- When a dimensions value is updated, the old one is simply overwritten

Original Value

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>McGillicutty</td>
<td>Hortence</td>
</tr>
</tbody>
</table>

New Value

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>Hollywoger</td>
<td>Hortence</td>
</tr>
</tbody>
</table>
Type 2 Dimension

- When a dimension is changed, a new record is inserted and old one dated

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
<th>FromDate</th>
<th>ThruDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>McGillicuty</td>
<td>Hortence</td>
<td>12/1/1998</td>
<td>&lt;NULL&gt;</td>
</tr>
</tbody>
</table>

New Value

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
<th>FromDate</th>
<th>ThruDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>2468</td>
<td>PQ1894958</td>
<td>Hollywoger</td>
<td>Hortence</td>
<td>7/6/2008</td>
<td>&lt;NULL&gt;</td>
</tr>
</tbody>
</table>
Type 3 – Just Say NO

• When a dimensions value is updated, a new column is added

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last1</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>McGillicutty</td>
<td>Hortence</td>
</tr>
</tbody>
</table>

Original Value

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last1</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>Hollywoger</td>
<td>McGillicutty</td>
</tr>
</tbody>
</table>

• Almost never used
Type 4 Dimension

- When a dimension is changed, an old record is copied to the history table and the current one is updated.

Original Value in DimEmployee

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>McGillicuty</td>
<td>Hortence</td>
</tr>
</tbody>
</table>

New Value in DimEmployee

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>Hollywoger</td>
<td>Hortence</td>
</tr>
</tbody>
</table>

New Value in DimEmployee_History

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
<th>FromDate</th>
<th>ThruDate</th>
</tr>
</thead>
</table>
Type 4 Dimension (Another Way)

- When a dimension is changed, old record is updated in history table, current one copied in

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>Hollywoger</td>
<td>Hortence</td>
</tr>
</tbody>
</table>

New Value in DimEmployee

<table>
<thead>
<tr>
<th>ID</th>
<th>EmployeeID</th>
<th>Last</th>
<th>First</th>
<th>FromDate</th>
<th>ThruDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1234</td>
<td>PQ1894958</td>
<td>Hollywger</td>
<td>Hortence</td>
<td>7/6/2008</td>
<td>&lt;NULL&gt;</td>
</tr>
</tbody>
</table>
Conformed Dimensions

- When pulling in data from multiple systems, you often have to reconcile different primary keys.
- This process is known as conforming your dimensions.

<table>
<thead>
<tr>
<th>ID</th>
<th>Product</th>
<th>InventoryID</th>
<th>PurchasingID</th>
<th>WorkMgtID</th>
</tr>
</thead>
<tbody>
<tr>
<td>9876</td>
<td>Widget</td>
<td>459684932</td>
<td>Wid45968</td>
<td>602X56VV1</td>
</tr>
</tbody>
</table>
### Dimensions in a Star Schema

<table>
<thead>
<tr>
<th>ID</th>
<th>SoldByID</th>
<th>SoldToID</th>
<th>ProductID</th>
<th>Qty</th>
<th>SaleAmt</th>
<th>SaleDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>3456</td>
<td>1234</td>
<td>6789</td>
<td>987</td>
<td>3</td>
<td>156.00</td>
<td>7/17/2009</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProductID</td>
<td>987</td>
</tr>
<tr>
<td>BusinessID</td>
<td>SHBL4X</td>
</tr>
<tr>
<td>Description</td>
<td>Knit Shirt</td>
</tr>
<tr>
<td>Color</td>
<td>Blue</td>
</tr>
<tr>
<td>Size</td>
<td>4XL</td>
</tr>
<tr>
<td>Sleeve</td>
<td>Long</td>
</tr>
</tbody>
</table>

- Flattened model easier to link
- Simpler for ad-hoc reporting
- Takes more database space
- Updates take more work (4XL to XXXXL requires 1 update per record)
### Dimensions in a Snowflake Schema

<table>
<thead>
<tr>
<th>ID</th>
<th>SoldByID</th>
<th>SoldToID</th>
<th>ProductID</th>
<th>Qty</th>
<th>SaleAmt</th>
<th>SaleDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>3456</td>
<td>1234</td>
<td>6789</td>
<td>987</td>
<td>3</td>
<td>156.00</td>
<td>7/17/2009</td>
</tr>
</tbody>
</table>

#### Columns

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProductID</td>
<td>987</td>
</tr>
<tr>
<td>BusinessID</td>
<td>SHBL4X</td>
</tr>
<tr>
<td>Description</td>
<td>Knit Shirt</td>
</tr>
<tr>
<td>Color</td>
<td>2</td>
</tr>
<tr>
<td>Size</td>
<td>7</td>
</tr>
<tr>
<td>Sleeve</td>
<td>2</td>
</tr>
</tbody>
</table>

#### Value Tables

<table>
<thead>
<tr>
<th>ID</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Red</td>
</tr>
<tr>
<td>2</td>
<td>Blue</td>
</tr>
<tr>
<td>3</td>
<td>Green</td>
</tr>
<tr>
<td>6</td>
<td>3XL</td>
</tr>
<tr>
<td>7</td>
<td>4XL</td>
</tr>
<tr>
<td>8</td>
<td>5XL</td>
</tr>
<tr>
<td>1</td>
<td>Short</td>
</tr>
<tr>
<td>2</td>
<td>Long</td>
</tr>
</tbody>
</table>

- Takes less database space. Linked Dimensions reusable.
- Easier to update (change 4XL to XXXXL requires 1 Update)
- More difficult as it requires many links
- More links makes it difficult for ad-hoc reporting (views can help with this)
Getting Data Into A Warehouse

- ETL
- Extract
- Transform
- Load
- SSIS – SQL Server Integration Services
Getting Data Out of Your Warehouse

- SSRS – SQL Server Reporting Services
- SSAS – SQL Server Analysis Services
• Key Performance Indicators
• Dashboards
• Quick, at a glance indicator of system health
• Inmon – Bill Inmon - Top down
• Kimball – Ralph Kimball - Bottom up
• Demo Site - 
http://www.codeplex.com/Wiki/View.aspx?ProjectName=SqlServerSamples

• Sample DBs -  http://www.codeplex.com/MSFTDBProdSamples/

• SSAS -  

• SSIS - http://www.codeplex.com/MSFTISProdSamples/

• SSRS - http://www.codeplex.com/MSFTRSProdSamples/
Resources

The Data Warehouse Toolkit
and
The Microsoft Data Warehouse Toolkit
by the Kimball Group

- [The Data Warehouse Toolkit](http://www.amazon.com/Data-Warehouse-Toolkit-Complete-Dimensional/dp/0471200247/ref=pd_bbs_sr_1?ie=UTF8&s=books&qid=1239580212&sr=8-1)
- [The Microsoft Data Warehouse Toolkit](http://www.amazon.com/MicrosoftData-Warehouse-Toolkit-MicrosoftBusiness-Intelligence/dp/0471267155/ref=sr_1_fkmr0_1?ie=UTF8&qid=1264636802&sr=8-1-fkmr0)
Resources

Smart Business Intelligence Solutions with Microsoft SQL Server 2008

- [http://www.amazon.com/Business-Intelligence-Solutions-Microsoft%C2%AE-PRO-Developer/dp/0735625808/ref=sr_1_1?ie=UTF8&s=books&qid=123958654654&sr=1-1](http://www.amazon.com/Business-Intelligence-Solutions-Microsoft%C2%AE-PRO-Developer/dp/0735625808/ref=sr_1_1?ie=UTF8&s=books&qid=123958654654&sr=1-1)
Resources

Programming Microsoft SQL Server 2008

- [Link](http://www.amazon.com/Programming-Microsoft-Server-2008-PRO-Developer/dp/0735625999/ref=sr_1_1?ie=UTF8&s=books&qid=1239580376&sr=1-1)
Resources - SSIS

- **Erik Veerman / Brian Knight Books**
  
  
  - [http://www.amazon.com/Professional-Microsoft-Integration-Services-Programmer/dp/0470247959/ref=sr_1_1?ie=UTF8&s=books&qid=1239833324&sr=8-1](http://www.amazon.com/Professional-Microsoft-Integration-Services-Programmer/dp/0470247959/ref=sr_1_1?ie=UTF8&s=books&qid=1239833324&sr=8-1)
  
  
  
Resources - SSAS

- [http://www.amazon.com/Microsoft%C2%AE-Server%C2%AE-Analysis-Services-Microsoft/dp/0735626200/ref=sr_1_5?ie=UTF8&s=books&qid=1252100419&sr=1-5](http://www.amazon.com/Microsoft%C2%AE-Server%C2%AE-Analysis-Services-Microsoft/dp/0735626200/ref=sr_1_5?ie=UTF8&s=books&qid=1252100419&sr=1-5)
Resources - SSRS

- [Pro SQL Server 2008 Reporting Services](http://www.amazon.com/Pro-Server-2008-Reporting-Services/dp/1590599926/ref=sr_1_11?ie=UTF8&s=books&qid=1252100749&sr=1-11)
- [Microsoft Server Reporting Services](http://www.amazon.com/Microsoft%C2%AE-Server%C2%AE-Reporting-Services-Microsoft/dp/0735626472/ref=sr_1_10?ie=UTF8&s=books&qid=1252100749&sr=1-10)
- [Professional Microsoft Reporting Services - Programmer](http://www.amazon.com/Professional-Microsoft-Reporting-Services-Programmer/dp/0470242019/ref=sr_1_15?ie=UTF8&s=books&qid=1252100793&sr=1-15)
Resources

**Blogs**
Brian Knight - [http://pragmaticworks.com/community/blogs/brianknight/default.aspx](http://pragmaticworks.com/community/blogs/brianknight/default.aspx)

**Podcast**
SQL Share (formerly JumpstartTV) – [http://sqlshare.com](http://sqlshare.com)

**Forums**

**Other**
Microsoft BI Site - [http://www.microsoft.com/events/series/bi.aspx](http://www.microsoft.com/events/series/bi.aspx)
SQL Serverpedia - [http://sqlserverpedia.com/](http://sqlserverpedia.com/)
Thanks Again!

- Questions?
- All material available at http://arcaneencode.com
- rcain@comframe.com