Extraction, Transformation and Loading Process

Extraction, Transformation and Loading

- Data is extracted from the operational systems
- Before data is loaded to the DW it needs to pass a Transformation process.
- ETL tools are commonly used to extract, cleanse, transform and load data into the DW.
- ETL operate as the central hub of incoming data to the DW

ETL and Metadata

- Metada is created by and updated from the load programs that move data into the DW or DM
- ELT tools generate and maintain centralized metadata
  - Some ETL tools provide an open Metadata Exchange Architecture that can be used to integrate all components of a DW architecture with central metadata maintained by the ETL tool.

Role of Metadata

- Metadata allows the decision support user to find out where data is in the architected DW environment.
- Metadata contains the following components:
  - Identification of the source of data
  - Description of the customization that has occurred as the data passes from DW to DM
  - Descriptive information about the tables, attributes and relationships
  - Definitions

Role of Metadata

- Three main layers of metadata exist in the DW
  - Application-level (operational) metadata
  - Core warehouse metadata - catalog of the data in the warehouse. I s based on abstractions of real world entities like project, customer
  - User-level metadata - maps the core warehouse metadata to useful business concepts

Mistakes to Avoid

- "Garbage in = Garbage out " Avoid Loading dirty data into the DW
- Avoid Building stovepipe data marts that do not integrate with central metadata definitions

ETL- Tools for Data Extraction, Transformation and Load

- Techniques Available:
  - Commercial off-the-shelf ETL (+)
  - Write ETL using a procedural language (+-)
  - Data replication of source data to the DW (-)

Generation of ETL Tools

- **First Generation - Source Code-Generation**
  - Prism Executive Suite from Ardent Software , Inc
  - Passport from Carleton Corporation
  - ETI-extract tool suite from Evolutionary Techn., Inc
  - Copy Manager from Information Builders, Inc.
  - SAS/Warehouse Administrator from SAS Institute, Inc.
- **Second Generation - Executable Code-Generation**
  - PowerMart from Informatica Corporation
  - Ardent DataStage from Ardent Software , Inc.
  - Data Mart Solution from Sagent Technology, Inc.
  - Tapestry from D2K , Inc.
  - Ab Initio from Ab Initio Software Corporation
  - Genio from LeoLogic

Strengths and Limitations of First Generation ETL tools

- **Strengths**
  - Tools are mature
  - Good at extracting data from legacy systems
- **Limitations**
  - High cost products and complex training requirements
  - Extract programs must be compiled from source code
  - Single-threaded applications do not use parallel processor
  - Use of intermediate files limits throughput
  - Requirements to manage code, files, programs, JCL
  - Many transformations must be coded by hand
  - Significant amount of metadata must be manually generated

Strengths and Limitations of Second Generation ETL tools

- **Strengths**
  - Lower cost
  - Products are easy to learn
  - Generate extensible metadata
  - Generate directly executable code (speed up the extraction process)
  - Parallel architecture
- **Limitations**
  - Many tools are not mature
  - Some tools are limited to RDBMS database sources
  - Problems with capacity when apply to large enteprise DW architectures

ELT process

- Data transformation can be very complex and can include:
  - Field translations (such as from mainframe to PC formats)
  - Data formatting (such as decimal to binary data formats)
  - Field formatting (to truncate or pad field data when loaded into DW)
  - Reformatting data structure (such as reordering table columns)
  - Replacing field data through table lookups (such as changing alpha codes to numeric Ids)
  - Remapping transation codes to summary reporting codes
  - Applying logical data transformations based on user-defined business rules
  - Aggregating transaction level values into “roll-up” balances

Steps of Daily Production Extract

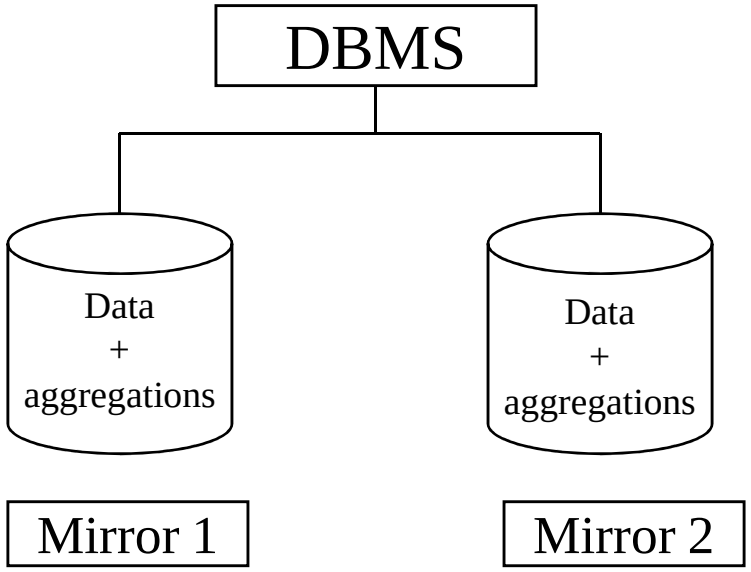
- Primary extraction (read the legacy format)
- Identifying the changed records
- Generalizing keys for changing dimensions
- Transforming into load records images
- Migration from the legacy system to DW system
- Sorting and Building Aggregates
- Generalizing keys for aggregates
- Loading
- Processing exceptions
- Quality assurance
- Publishing

Extraction Phase - Snapshots

- Periodically Snapshots
- Identify What Has Changed since the last time a snapshot was built. The amount of effort is determined by the type of scanning technique
  - Scan data that has been time stamped
  - Scan a "Delta" file
  - Scan an Audit Log
  - Modify the application code
  - Compare "Before" and "After" image files

Loading Phase

- All or part of the DW is taken off-line while new data is loaded.
- One way to minimize the downtime due to loading is to MIRROR the DW.



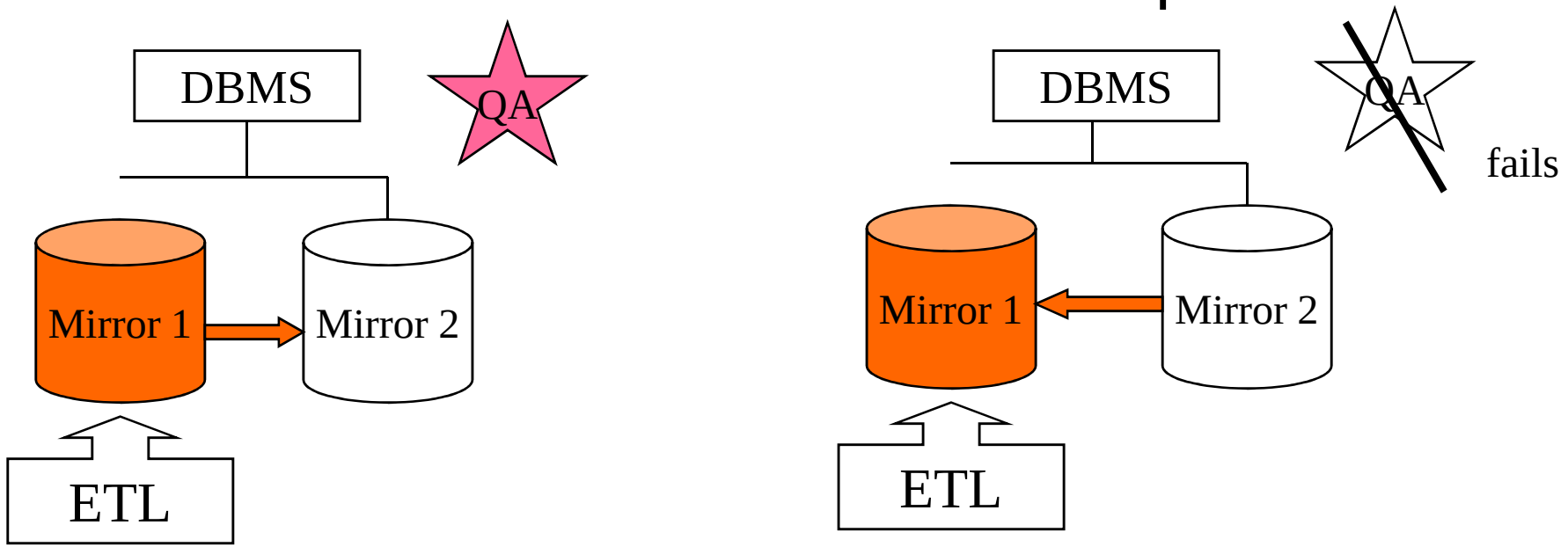
What are the benefits of using a mirrored DW?

Mirrored Data Warehouse

- High reliability during the day
- During loading the mirror is broken
- Both production and loading processes run in parallel in different mirrors

Data Quality Assurance

- At the end of the loading process a Quality Assurance check is made on the data in the mirror that has been changed.
- All-disk to all-disk data transfer takes place



Advantages Techniques

- Use of 3 mirrors
  - 2 for data availability and redundancy
  - 1 for loading
- Use of Segmentable Fact Table index
  - Drop the most recent section (segment) of the master index of the Fact table, rather than the whole table
  - This technique provides speed and allows the portion of the index that is dropped to be rebuilt quickly once the loading is complete

Loading, Indexing and Exception Processing

- Loading data into a fact table should be done as a bulk loading operation with the master index turned off. It will be much faster to perform a bulk load rather than process the records one at a time with INSERT or UPDATE statements.
- The ability during a bulk data load to insert or overwrite , and the abilityto insert or add to values simplifies the logic of data loading.
- Loading and Indexing should be able to gain enourmous benefits from parallelization.
- Loading data into a dimension table is quite different from loading data into a fact table.
  - The fact table typically has one or two large indexes built on a combination of dimension keys.
  - A dimension tables, typically has many keys built on its textual attributes.

Conforming Dimensions

- Incompatible Granularity when loading data from multiple sources
- Conforming the Dimensions means:
  - forcing the two data sources to share identical dimensions
  - expressing both data sources at the lowest common level of granularity