DWBI : Unit 4 Imp Questions

5. Explain Extract System in ETL process

9. What are the various requirements that impact the design and development of the ETL system?

13. How Business needs, Data Quality, Data Integration and Available skills requirements can impact the design and development of the ETL system?

Ans:

extracting data from the source systems is a fundamental component of the ETL architecture.

A] file oriented extract, stream oriented extract : There are two primary methods for getting data from a source system: as a file or a stream. If the source is an aging mainframe system, it is often easier to extract into files and then move those files to the ETL server.If you are using an ETL tool and your data is in a database you may be able to set up the extract as a stream.an extract to file approach consists of three or four discrete steps: extract to file, move file to ETL server, transform file contents, and load transformed data into the staging database.

B] data compression: Data compression is important if you need to transfer large amounts of data over a significant distance or through a public network. In this case, the communications link is often the bottleneck. If too much time is spent transmitting the data, compression can reduce the transmission time by 30 to 50 percent or more, depending on the nature of the original data file.

C]data encryption: Data encryption is important if you are transferring data through a public network, or even internally in some situations. If this is the case, it is best to send everything through an encrypted link and not worry about what needs to be secure and what doesn't. Remember to compress before encrypting.

6. What is the role of error event schema subsystem in ETL process?

15. How Error event schema is used in ETL process?

Ans:

A] The error event schema is a centralized dimensional schema whose purpose is to record every error event thrown by a quality screen anywhere in the ETL pipeline. Although we are focusing on data warehouse processing, this approach can be used in generic data integration (DI) applications where data is being transferred between legacy applications. The error event schema is shown in Figure

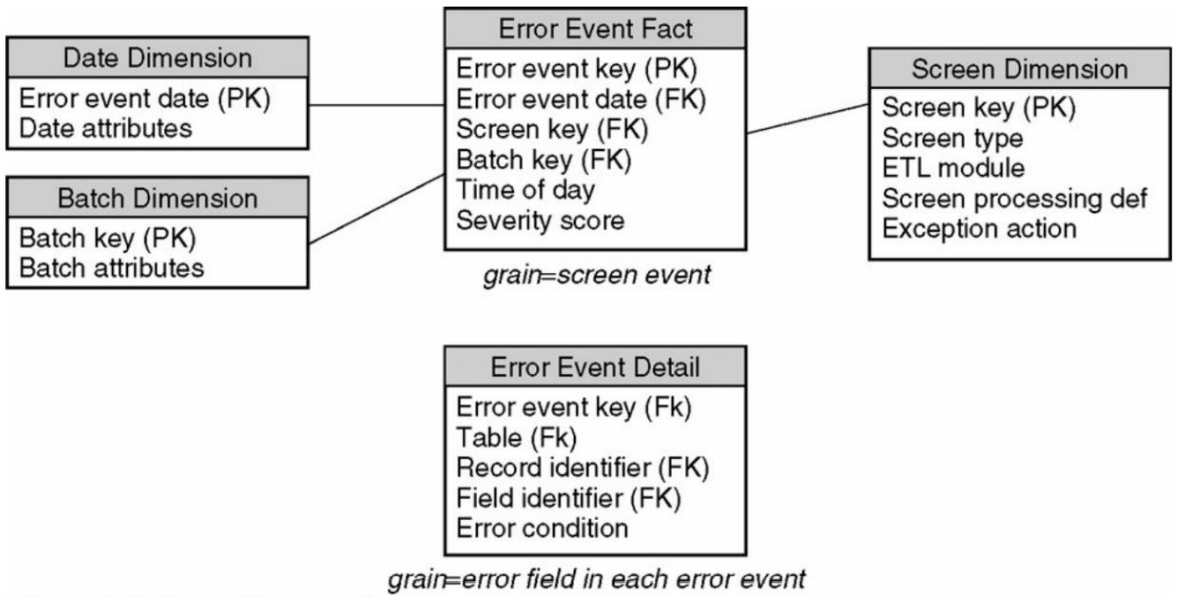

Figure 9-1: Error Event schema

The main table is the error event fact table. Its grain is every error thrown (produced) by a quality screen anywhere in the ETL system.

The dimensions of the error event fact table include the calendar date of the error, the batch job in which the error occurred, and the screen that produced the error. The calendar date is not a minute and second timestamp of the error, but rather provides a way to constrain and summarize error events by the usual

attributes of the calendar, such as a weekday or the last day of a fiscal period. The time-of-day fact is a full relational date/timestamp that specifies precisely when the error occurred. This format is useful for calculating the time interval between error events because you can take the difference between two date/timestamps to get the number of seconds separating events.

The error event schema includes a second error event detail fact table at a lower grain. Each record in this table identifies an individual field in a specific data record that participated in an error. Thus a complex structure or business rule error that triggers a single error event record in the higher level error event fact table may generate many records in this error event detail fact table. The two tables are tied together by the error event key, which is a foreign key in this lower grain table.

The error event detail table could also contain a precise date/timestamp to provide a full description of aggregate threshold error events where many records generate an error condition over a period of time.

7. Explain Deduplication and Conforming system in ETL process .

Ans:

A] Deduplication System :

Often dimensions are derived from several sources. This is a common situation for organizations that have many customer facing source systems that create and manage separate customer master tables. Customer information may need to be merged from several lines of business and outside sources. Sometimes the data can be matched through identical values in some key column. However, even when a definitive match occurs, other columns in the data might contradict one another, requiring a decision on which data should survive. there is seldom a universal column that makes the merge operation easy.The different sets of data being integrated and the existing dimension table data may need to be evaluated on different fields to attempt a match. Sometimes a match may be based on fuzzy criteria, such as names and addresses that may nearly match except for minor spelling differences

Survivorship is the process of combining a set of matched records into a unified image that combines the highest quality columns from the matched records into a conformed row. There are a variety of data integration and data standardization tools to consider if you have difficult deduplicating, matching, and survivorship data issues. These tools are quite mature and in widespread use

B] Conforming system :

1) Conforming consists of all the steps required to align the content of some or all of the columns in a dimension with columns in similar or identical dimensions in other parts of the data warehouse. For instance, in a large organization you may have fact tables capturing invoices and customer service calls that both utilize the customer dimension. It is highly likely that the source system for invoices and customer service have separate customer databases. It is likely there will be little guaranteed consistency between the two sources of customer information. The data from these two customer sources needs to be conformed to make some or all of the columns describing customer share the same domains.

2)The conforming subsystem is responsible for creating and maintaining the conformed dimensions and conformed facts .incoming data from multiple systems needs to be combined and integrated so that it is structurally identical, deduplicated, filtered of invalid data, and standardized in terms of content rows in a conformed image. A large part of the conforming process is the deduplicating, matching, and survivorship processes described previously. The conforming process flow combining the deduplicating and survivorship processing is shown in Figure 9-3.

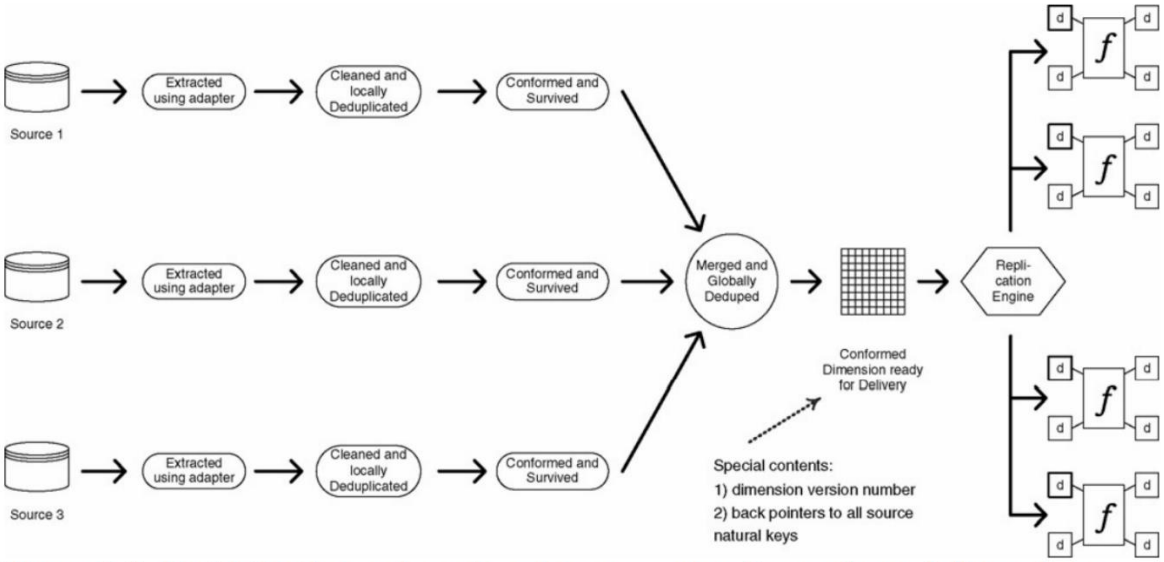

Figure 9-3: Deduplicating and survivorship processing for conformed dimensions

To implement conformed dimensions and facts, the conforming subsystem needs domain mappings that are the reference metadata for capturing the relationship between explicitly valid values from source systems to the conformed dimension and conformed fact values.

2. Explain Change Data Capture system.

10. Explain Change Data Capture System in ETL .

Ans:

The idea behind change data capture is simple enough: Just transfer the data that has been changed since the last load. But building a good change data capture system is not as easy as it sounds. You must carefully evaluate your strategy for each data source. There are several ways to capture source data changes, each effective in the appropriate situation:

1. Audit columns: In some cases, the source system has appended audit columns that store the date and time a record was added or modified. These columns are usually populated via database triggers that are fired off automatically as records are inserted or updated. Sometimes, for performance reasons, the columns are populated by the source application instead of database triggers. When these fields are loaded by any means other than database triggers, you must pay special attention to their integrity, analysing and testing each column to ensure that it's a reliable source to indicate change. If you uncover any NULL values, you must find an alternative approach for detecting change. The most common situation that prevents the ETL system from using audit columns is when the fields are populated by the source application and the DBA team allows back-end scripts to modify data. If this occurs in your environment, you face a high risk of missing changed data during your incremental loads.

2. Timed extracts: With a timed extract, you typically select all rows where the date in the create or modified date fields equal SYSDATE-1, meaning all of yesterday's records. Sounds perfect, right? Wrong. Loading records based purely on time is a common mistake made by most beginning ETL developers. This process is horribly unreliable. Time-based data selection loads duplicate rows when it is restarted from mid-process failures. This means that manual intervention and data clean-up is required if the process fails for any reason. Meanwhile, if the nightly load process fails to run and skips a day, there's a risk that the missed data will never make it into the data warehouse.

3. Full "diff compare.": A full diff compare keeps a full snapshot of yesterday's data, and compares it, record by record against today's data to find what changed. The good news is that this technique is thorough: you are guaranteed to find every change. The obvious bad news is that, in many cases, this technique is very resource intensive. If you must do a full diff compare, try to do the comparison on the source machine so you don't have to transfer the entire table or database into the ETL environment. Of course, the source support folks may have an opinion about this. Also, investigate using cyclic redundancy checksum (CRC) algorithms to quickly tell if a complex record has changed.

4.Database log scraping: Log scraping effectively takes a snapshot of the database redo log at a scheduled point in time (usually midnight) and scours it for transactions that affect the tables of interest for your ETL load. A variant of scraping is "sniffing" which involves monitoring the redo log process, capturing transactions on-the-fly. In any case, scraping the log for transactions is probably the messiest of all techniques. It's not uncommon for transaction logs to get full and prevent new transactions from processing. When this happens in a production transaction environment, the knee-jerk reaction from the responsible DBA may be to empty the log so business operations can resume, but when a log is emptied, all transactions within it are lost. If you've exhausted all other techniques and find log scraping is your last resort for finding new or changed records, persuade the DBA to create a special log to meet your specific needs.

5. Message queue monitoring: In a message-based transaction system, the queue is monitored for all transactions against the tables of interest. The contents of the stream are similar to what you get with log sniffing. One benefit is this process is relatively low overhead, assuming you already have the message queue in place.

4. Role of Data Profiling in ETL.

12. How data profiling is done in ETL Process?

Ans:

1) Data profiling is the technical analysis of data to describe its content, consistency, and structure. In some sense, any time you perform a SELECT DISTINCT investigative query on a database field, you are doing data profiling. There are a variety of tools specifically designed to do powerful profiling.

2)Data profiling plays two distinct roles: strategic and tactical. As soon as a candidate data source is identified, a light profiling assessment should be made to determine its suitability for inclusion in the data warehouse and provide an early go/no go decision. ideally this strategic assessment should occur immediately after identifying a candidate data source during the business requirements analysis. Once the basic strategic decision is made to include a data source in the project, a lengthy tactical data profiling effort should occur to identify as many problems as possible. Usually, this task begins during the data modelling process and extends into the ETL system design process.

3) Issues that show up in this subsystem result in detailed specifications that are either 1) sent back to the originator of the data source as requests for improvement,

or 2) form requirements for your data quality subsystem.

4)The profiling step provides the ETL team with guidance as to how much data cleaning machinery to invoke and protects them from missing major project milestones due to the unexpected diversion of building systems to deal with dirty data.