ETL Anatomy 101

Tom Miron
Systems Seminar Consultants
Madison, WI

1

What Is ETL?

Extract, Transform, Load

ETL is often associated with Data Warehouse/Mart

But…
An ETL-like process is key for many reporting and analysis systems…

…even if these are not part of a Data Warehouse.

2

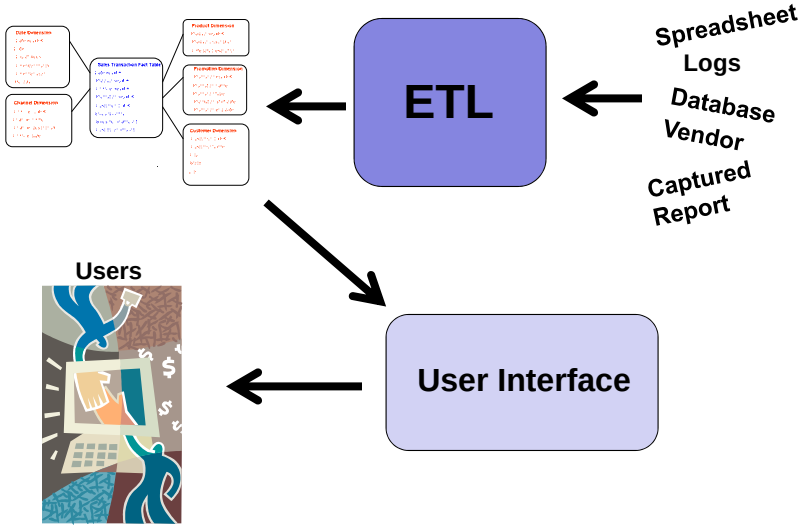
What is ETL?

Key features of ETL:

- Bring data under the control of your application from an outside world over which you have little or no control.
- Reliably publish that data to the end user community.

3

What is ETL?



4

What we'll cover

- Input and output
- Basic functions
- Design consideratons
- Inside ETL
- Wrap up

ETL In

- Operational data
- Periodic data feeds
- Streaming data feeds

6

ETL in: operational data

Operational data

- Operational data originates with business events.
- An operational data item is often the record of a business event.

Examples:

- customer order
- patient visit
- banner ad click through
- account opened

7

ETL in: operational data

- Insert efficiency – get info into the database
- Up time
- Integrity

Operational data systems are not concerned with…

- Query efficiency – get info out of the database
- Optimization for reporting and analysis

ETL copies data out of the operational system into an end user area optimized for query and reporting.

8

ETL in: periodic data feeds

Data feeds come from external sources…

- financial data subscriptions
- off line extracts
- Bob's month-end sales spreadsheet

Data feeds may represent operational data but don't come directly from the operational systems.

9

ETL in: periodic data feeds

Data feeds are problematic for any system…

- data and file formats can differ
- timing
- reliability
- data element names

If you have data feeds you will need ETL functions.

10

ETL in: streaming feeds

Most ETL systems gather data on a periodic basis…

- update overnight
- update weekly

ETL principles can be applied to live or streaming feeds.

We'll just acknowledge such systems but concentrate on periodic update systems.

11

ETL out

ETL output can vary as much as input.

The key feature of ETL output is it's value-added nature…

…inputs have been transformed in some way that makes the data more useful.

12

ETL out

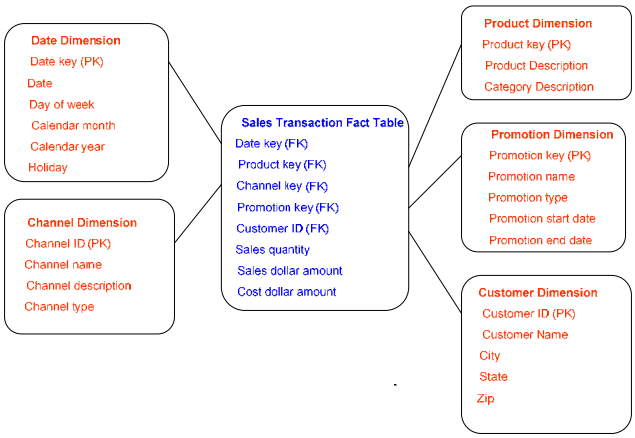
Common ETL outputs…

- dimensional database
- reports
- graphics

13

ETL out: dimensional database

When working with a formal data warehouse, ETL output is often a series of relational tables arranged as a dimensional model.



14

ETL out: reports and graphics

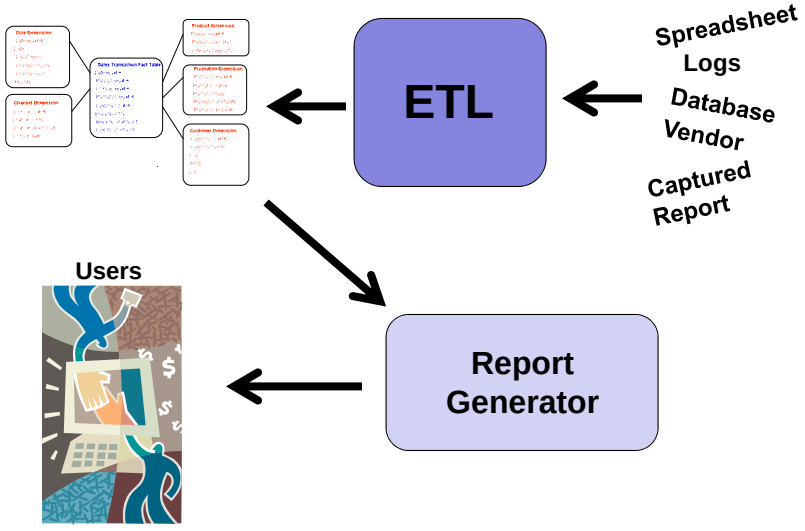
Reporting, analysis, or graphics systems may have no component named “ETL” but…

…most do include the basic ETL functions:

- get outside data into the application
- validate
- transform data into to something that can be represented on the report

15

ETL out: reports and graphics



16

Basic ETL functions

- Outside data interface
- Data formatting
- Handling time
- Data conformation
- Process management

17

Basic ETL functions: Outside data interface

ETL is the interface between the wild, uncontrolled world outside of your application and…

… your perfectly controlled application.

Since you're designing the application you control what's going on…most of the time.

18

Basic ETL functions: Outside data interface

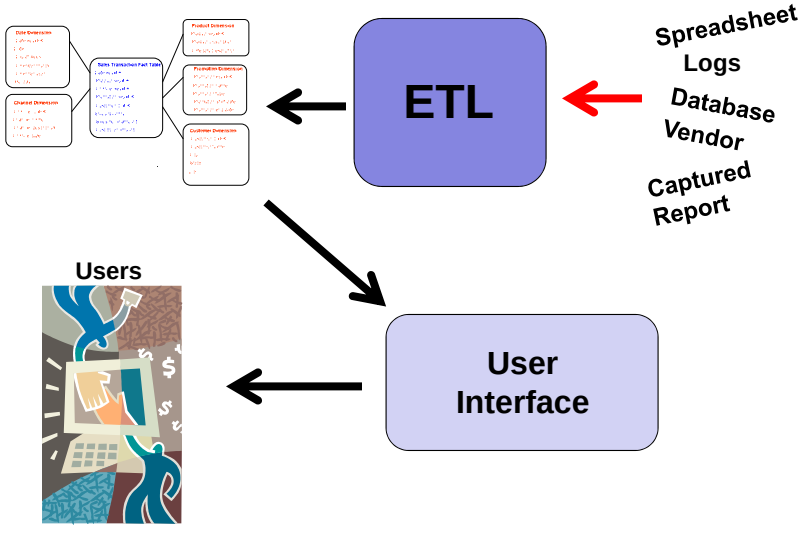
ETL must have the technical means to access outside data…

- FTP
- database access methods
- spreadsheet translation
- cross operating system transfers

A first task is to match up these technical means with the data you need to input.

19

Basic ETL functions: Outside data interface



20

Basic ETL functions: Data formatting

The ETL must be able transform the feed into a common format, typically, relational tables.

The ETL must be able to carry out any validation logic.

21

Basic ETL functions: Dealing with time

Most reporting systems deal with events at a discrete moment in time:

- sales last month
- impressions per hour
- average account balance first quarter

22

Basic ETL functions: Dealing with time

ETL must be able to:

- normalize the various input data to a common time frame
- normalize to a common time grain (precision)

ETL requires strong date and time handling features.

23

Basic ETL functions: Data conformation

Conformation means that an attribute or measure has the same meaning no matter where it's used and…

…an entity is represented in only one "official" way.

This ensures that you can join tables by common attributes.

ETL must have some means of normalizing or looking up the official version of an attribute.

24

Basic ETL functions: Process management

Process management is a catch all term for:

- Metadata
- Logging
- Scheduling
- Auditing
- Recovery
- Backup

Every system will have its own requirements.

The effort required for process management is easy to under estimate.

25

ETL Design Considerations: Write it yourself?

Small ETL (of ETL-like) systems are probably less expensive and require less time to create in house.

If skills or time are not available in house an ETL product may be considered.

26

ETL Design Considerations: Naming conventions

Common, published names for tables, files, columns, and business entities save…

- time
- frustration
- debugging effort
- support effort

- *Is it "transaction", "trans", "trn", "trnsact"…???*

27

ETL Design Considerations: Storage space

Disk space required for ETL intermediate processes and output storage must be anticipated.

Use a spreadsheet calculation of anticipated rows times bytes per row for all tables.

Be sure to account for intermediate and temporary data, indexes, backup, and growth.

28

ETL Design Considerations: Time

There are two time related issues:

- update cycle time
- time precision

29

ETL Design Considerations: Time

Cycle time relates to how often you update the data.

- Cycle time is dependent on the frequency of your data feeds.
- You can't do a complete update of the output database or reports any more frequently than your least frequent feed.
- Your update cannot be more current than the end date of the oldest feed data in the update cycle.

30

ETL Design Considerations: Time

Precision relates to the snap shot frequency of your feeds…

…regardless of update frequency.

Time precision cannot be any better than your least precise feed.

31

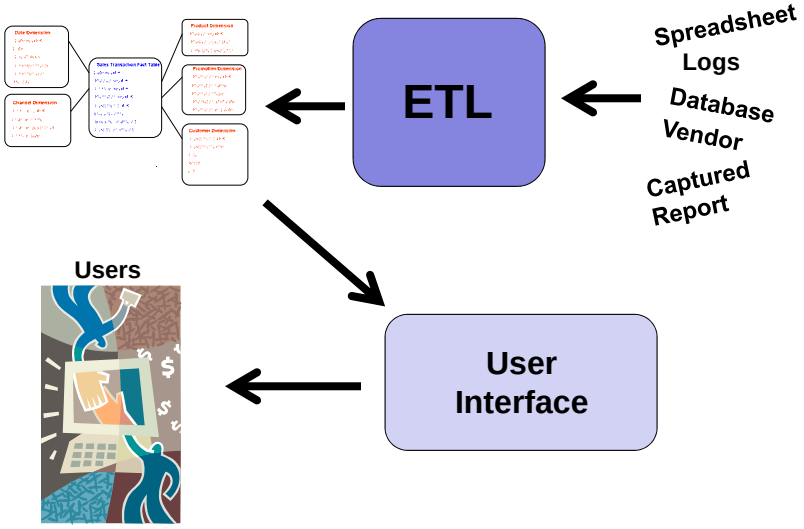
ETL Design Considerations: Quality Commitment

- Nail down who's responsible for what level of data quality.
- Push that responsibility as far upstream as possible…

…hopefully outside the realm of your ETL.

32

ETL Design Considerations: Quality Commitment



33

ETL Design Considerations: Metadata

Metadata can be:

- column name lists
- row counts
- update and insert counts
- version stamps

…any information about the information itself.

Generating and maintaining comprehensive metadata can be a complex task so follow the rule: *Every solution has to have a problem.*

34

ETL Design Considerations: Audits

Do you have compliance and auditing requirements?

- Identify audit points and data tracking requirements.
- Audits may require various levels of intermediate data storage. Factor these into disk space requirements.
- You may be able to push some audit responsibilities upstream along with quality commitments.

35

ETL Design Considerations: Recovery and backup

- Can you use the system back up or do you need to write your own?
- Recovery may depend on your service commitments to downstream users.

Note on recovery:
It can be difficult or impossible to reproduce data feeds.

At a minimum you should capture and store your raw data feeds through one successful ETL cycle.

This allows you to reproduce the cycle without tapping your feed sources for resends.

36

ETL Design Considerations: Attribute changes

aka “slowly changing dimensions”

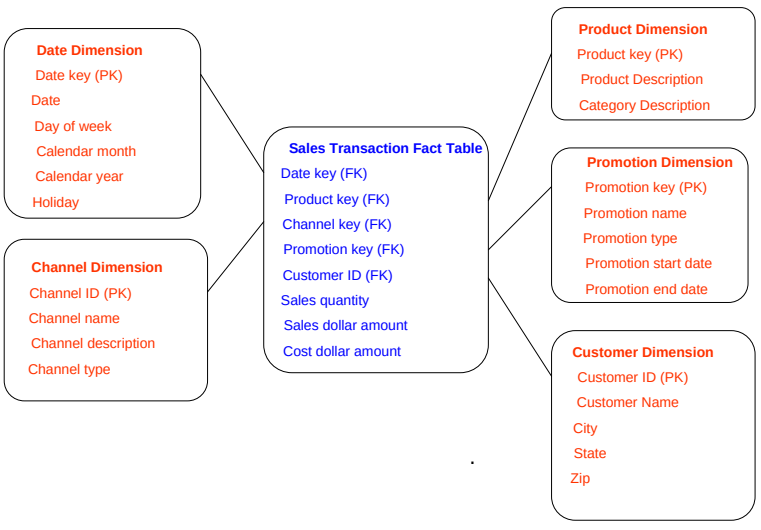
Example:

Acme Bearings Incorporated changes their name to “Bearings R Us.” What do you do?

This problem has been around as long as attribute tables themselves and there are three well defined strategies or “types” for handling it.

37

ETL Design Considerations: Attribute changes



38

ETL Design Considerations: Attribute changes

Type 1

Simply change “Acme Bearings Incorporated” to “Bearings R Us” in the customer table.

Now all past transactions keyed to the original name refer to “Bearings R Us.”

The fact that transactions before the change referred to a different company name is lost.

39

ETL Design Considerations: Attribute changes

Type 2

Insert a new customer row, with a new key for “Bearings R Us” and use that from the date of change onward.

Past transactions with the company original name are preserved and the new name is used going forward.

The fact that “Acme Bearings Incorporated” and “Bearings R Us” represent the same entity is lost.

40

ETL Design Considerations: Attribute changes

Type 3

Incorporate history attributes (columns) in the original customer table.

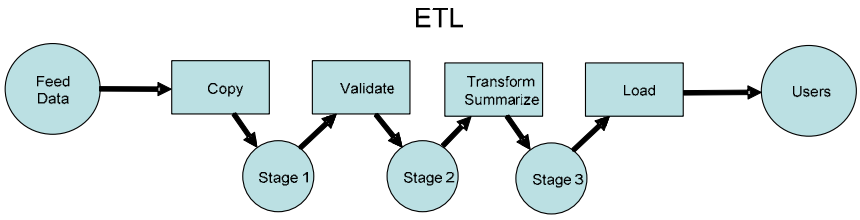
custID	customerName	previousName01	previousName02
4813	Pauls Pencils	Pauls Smartphones	
4814	**Bearings R Us**	**Acme Bearings Incorporated**	**Wayne's Machine Shop**
4815	Home Again Cleaning	Maid For You	Septic Solutions

Typically an effective date is also included.

41

ETL Design Considerations: Staging

Staging refers to storing intermediate copies of data as it moves through the ETL.



42

ETL Design Considerations: Staging

- Staging facilitates recovery without having to revert to the original feeds.
- Stage data is useful for audits.
- Staged data is useful during system development because it allows you break up the system and examine the output of each piece.
- Staging promotes modular systems.

43

ETL Design Considerations: Key look up

If you are assigning surrogate (meaningless) keys to attributes from the feed files…

… you will need some way of translating an operational key or text description to the surrogate key.

Example: Three feeds with differing names for a customer:

Acme Bearing, Co.

Acme Bearings

Acme Bearings, Inc.

44

ETL Design Considerations: Key look up

The ETL needs to assign the official key and name from the customer table:

custID	customerName	Region	State
4813	Pauls Pencils	MW	KS
4814	**Acme Bearings Incorporated**	NE	NY
4815	Home Again Cleaning	NE	ME

- As with the data quality commitments you may be able to enforce name conformity onto your feed providers.
- If names will be variable you'll have to develop a strategy to normalize them.

45

ETL Design Considerations: Key look up

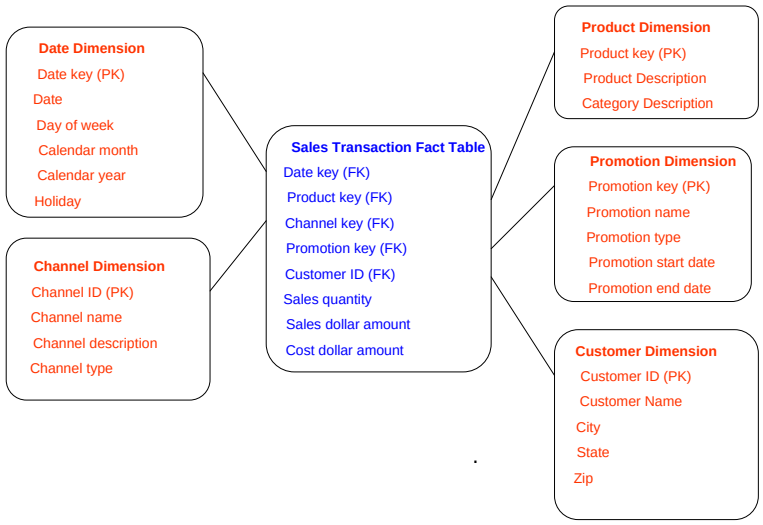
Even if company name never varies you'll have to decide how to look up the custID:

- You could query the customer table directly on name to return the key or…
- Maintain a key table within the ETL "back room" that holds only the operational key, name in this example, and the surrogate key. The operational key can be indexed.

custID	customerName
4813	Pauls Pencils
4814	**Acme Bearings Incorporated**
4815	Home Again Cleaning

46

ETL Design Considerations: Key look up



47

ETL Checklist

- Define data quality responsibilities.
- Technical means to read all anticipated input file and data formats and write output formats.
- Ability to reformat data
- Time and date handling
- Staging
- Backup, recovery
- Internal reporting, metadata, and audit
- Anticipate and provide for changes and corrections
- System support

48

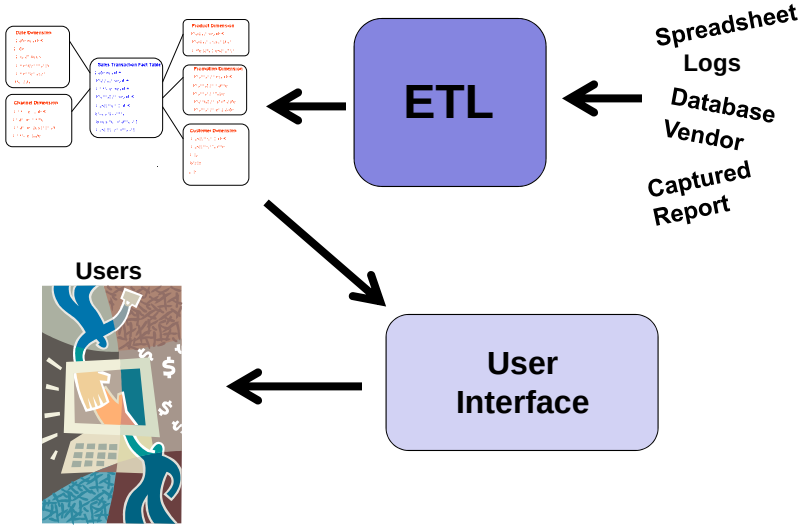
Finally

ETL is not necessarily for data warehouse:

ETL system considerations apply any time you're handling data feeds from outside sources and must publish that data to the end user environment.

49

Finally



50

Contact Information

Contact the author at:

Tom Miron
Systems Seminar Consultants
2997 Yarmouth Greenway Dr.
Madison, WI 53711
608 625-4541
608 278-9964
tmiron@sys-seminar.com