COMPLIANCE COMPONENT

Last Updated: 6/07/06

DEFINITION	
Name	Extract Transform and Load (ETL) Administration
Description	This document will present the necessary concepts needed to establish an ETL system which can be effectively administered from the start of the extract process to the final data presented in the data load process. What administrative tasks are performed in each ETL system will depend on the size of the ETL system and the amount and complexity of the data processed through the system.
Rationale	An effective administrative process to monitor the ETL process is necessary in order to document the flow of data from its source to its final output.
Benefits	Implementing an effective administrative process will ensure that the ETL system can do the following. • Deliver data most effectively to end user tools • Add value to data in the cleaning and conforming steps • Protect and document the lineage of data

ASSOCIATED ARCHITECTURE LEVELS	
Specify the Domain Name	Information
Specify the Discipline Name	Knowledge Management
Specify the Technology Area Name	Extract Transform and Load (ETL)
Specify the Product Component Name	

COMPLIANCE COMPONENT TYPE	
Document the Compliance Component Type	Guideline
Component Sub-type	

COMPLIANCE DETAIL	
State the Guideline, Standard or Legislation	The extent to which administrative tasks can be performed during the ETL process is dependent on the frequency that the data being processed is 'staged'. Staged in this context means written to disk or other storage device. It is recommended that data be staged at the four major checkpoints of the ETL data flow (Extraction, Cleansing, Conforming and Data Load). During the ETL process, data can either be staged or processed in memory. Being able to develop an efficient ETL process is partly dependent on being able to determine the right balance between physical input and output (I/O) and in-memory processing. The decision as to whether to stage the data or not depends on which of the following objectives is chosen: 1. With limited staging, data will move from the originating source to the ultimate target as fast as possible. 2. With extensive staging, you will have the ability to recover from failure without restarting from the beginning of the process but will reduce the speed of the process.

The decision to stage data varies depending on the environment and business requirements within which the ETL system operates. Consider the following reasons for staging data before it is transferred out of the ETL system to its final destination.

1. Recoverability – To optimize recoverability, it is a good practice to stage data as soon as it has been extracted from the source system and then again after each major transformation, assuming that the transformation steps are significant. These staging steps (tables in a database or file system) serve as recovery points.
2. Backup – Frequently, massive volume prevents the data warehouse (or other final output destination) from being reliably backed up at the database level. Significant difficulties and loss of time can be avoided if the data load files are saved, compressed and archived.
3. Auditing- Many times the data lineage between the source and target is lost in the ETL code. When it comes time to audit the ETL process, having staged data makes auditing between different portions of the ETL process much more straightforward because auditors can simply compare the original input file with the logical transformation rules against the output file. Staging for auditing purposes is especially useful when the source system overwrites its history.

If the decision is made to stage some of the data at some point in the ETL process, an appropriate design is required for the staging area. Serious thought needs to be given to the various roles that data staging can play in the overall data flow operations. There is more to staging than just building temporary files to support the execution of the next job. A given staging file can also be used for restarting the job flow if a serious problem develops downstream, and the staging file can be a form of audit or proof that the data had specific content when it was processed. Some of the basic rules to follow when designing and implementing the staging areas are:

1. *The data-staging area must be owned by the ETL team responsible for the process.* The data-staging area and its content are off limits to anyone other than the ETL team. The data-staging area is not designed for presentation. There are no indexes or aggregations to support querying in the staging area. There are no service-level agreements for data access in the staging area.
2. *Users are not allowed in the staging area for any reason*.
3. *Reports cannot access data from the staging area*
4. *Only ETL processes can write to and read from the staging area*.

The staging area normally consists of both DBMS tables and flat text files on the file system. Flat files are especially important when using a dedicated ETL tool. Most of the tools utilize an area in the file system for placing data down to optimize its workflow. You may need to stage data outside the DBMS in flat files for fast sequential processing.

When the staging area(s) is initially set up, the ETL team must supply the database and OS administrator with an overall data storage measure for the staging area to be used. They must estimate the space allocations and parameter settings for the staging database, file systems and directory structures. In order to do this, a volumetric worksheet should be prepared. At a minimum, this worksheet will list each table in the staging area with the following information:

1. **Table Name** – Name of table or file in staging area. There is one row in the worksheet for each table or file.
2. **Update Strategy** – How the table is maintained. If it is a persistent

	staging table, it will have data appended, updated and perhaps deleted. Transient staging tables are truncated and reloaded with each process. 3. **Load Frequency** – Reveals how often the table is loaded or changed by the ETL process. 4. **ETL Job(s)** – Staging tables are populated or updated through ETL jobs. 5. **Initial Row Count** – Estimate how many rows each table in the staging area initially contains. 6. **Average Row Length** – For size estimation purposes, you must provide the DBA with the average row length in each staging table in bytes. 7. **Increases with** – You must define when each table in the staging area increases with additional records. These would be based on business rules. A table does not have records added to itself each time it is accessed. Data such as status codes may not change for long periods of time, while new accounts would increase each time a new account is added to the business. 8. **Expected Monthly Rows** – Estimate is based on history and business rules. 9. **Expected Monthly Bytes** – A calculation of average row length times expected monthly rows. 10. **Initial Table Size** – Table size is usually represented in bytes or megabytes. It is a calculation of average row length times initial row count. 11. **Table Size in 6 Months** – An estimation of table sizes after 6 months of activity. It is a calculation of (average row length times initial row count) + (average row length times expected monthly rows times 6) / 1,048,576. Another on-going administrative function is impact analysis. Impact analysis examines the metadata associated to a table or file and determines what is affected by a change to its structure or content. Changing data-staging tables can break processes that are critical to the delivery of data to the final destination. Once a table is created in the staging area, you must perform impact analysis before any changes are made to it. Impact analysis, an ETL function, is a serious responsibility since changes to the source systems and the target data can be continuous and only the ETL process knows exactly which of these disparate elements are connected.
Document Source Reference #	*The Data Warehouse ETL Toolkit* by Ralph Kimball, Wiley Publishing Inc, 2004

Compliance Sources			
Name		*Website*	
Contact Information			
Name		*Website*	
Contact Information			

KEYWORDS	
List Keywords	extraction, cleansing, conforming, data load, volumetric worksheet, impact analysis, recoverability, backup, auditing, administration, staging, Extract Transform and Load (ETL)

COMPONENT CLASSIFICATION				
Provide the Classification	☐ *Emerging*	☒ *Current*	☐ *Twilight*	☐ *Sunset*
Sunset Date				

COMPONENT SUB-CLASSIFICATION		
Sub-Classification	Date	Additional Sub-Classification Information
☐ *Technology Watch*		
☐ *Variance*		
☐ *Conditional Use*		

Rationale for Component Classification	
Document the Rationale for Component Classification	

Migration Strategy	
Document the Migration Strategy	

Impact Position Statement	
Document the Position Statement on Impact	

CURRENT STATUS				
Provide the Current Status	☐ *In Development*	☒ *Under Review*	☐ *Approved*	☐ *Rejected*

AUDIT TRAIL			
Creation Date	4/21/2006	*Date Approved / Rejected*	6/13/06
Reason for Rejection			
Last Date Reviewed		*Last Date Updated*	
Reason for Update			