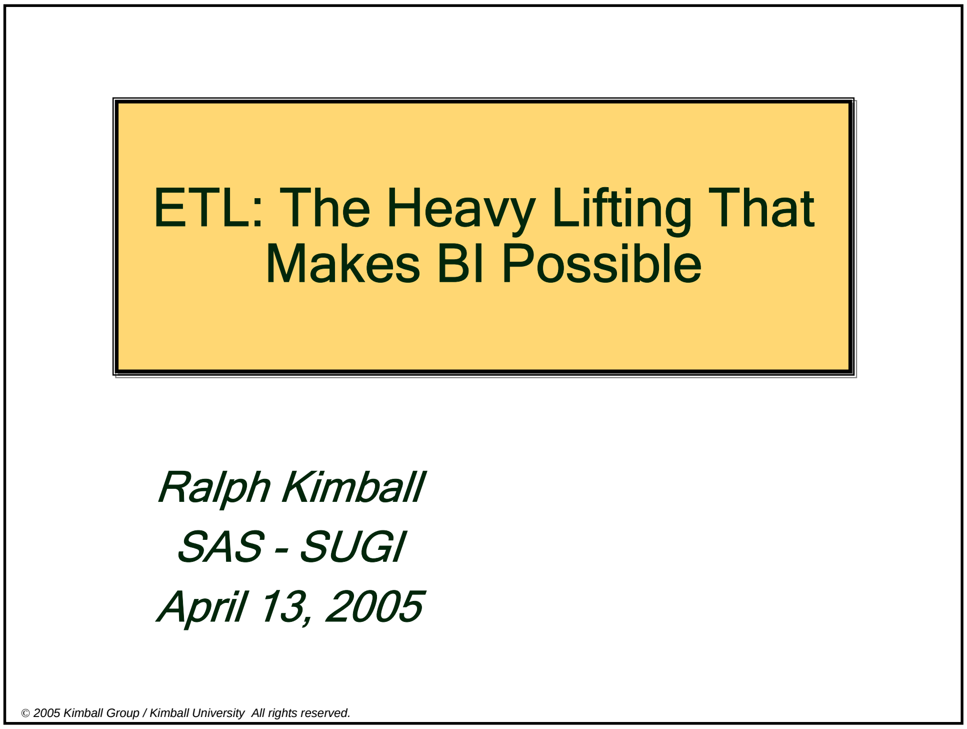
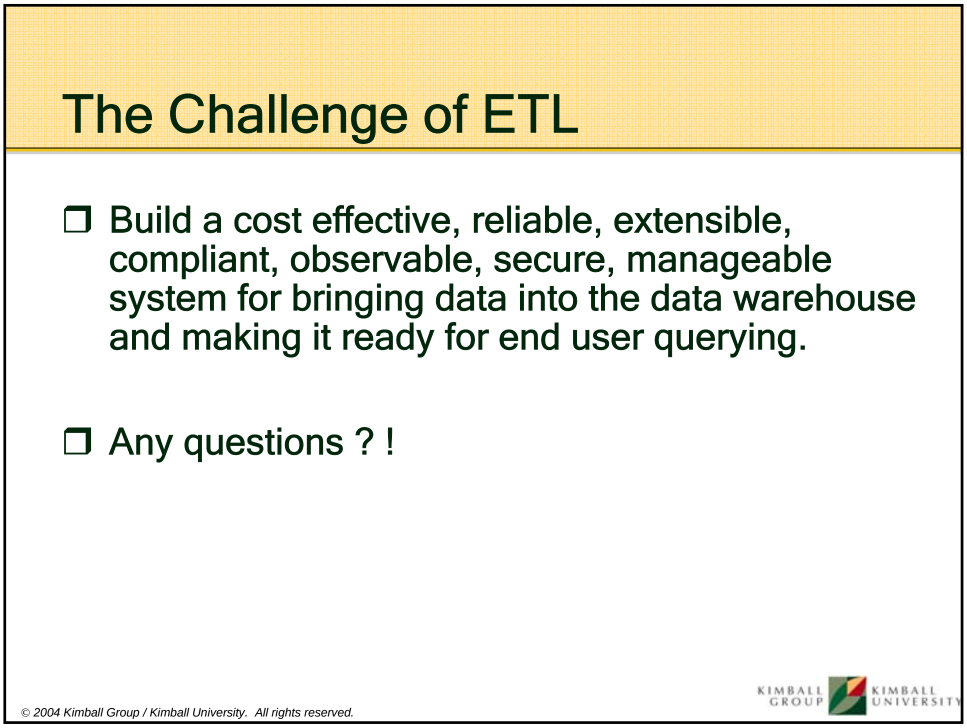
Paper 123-30



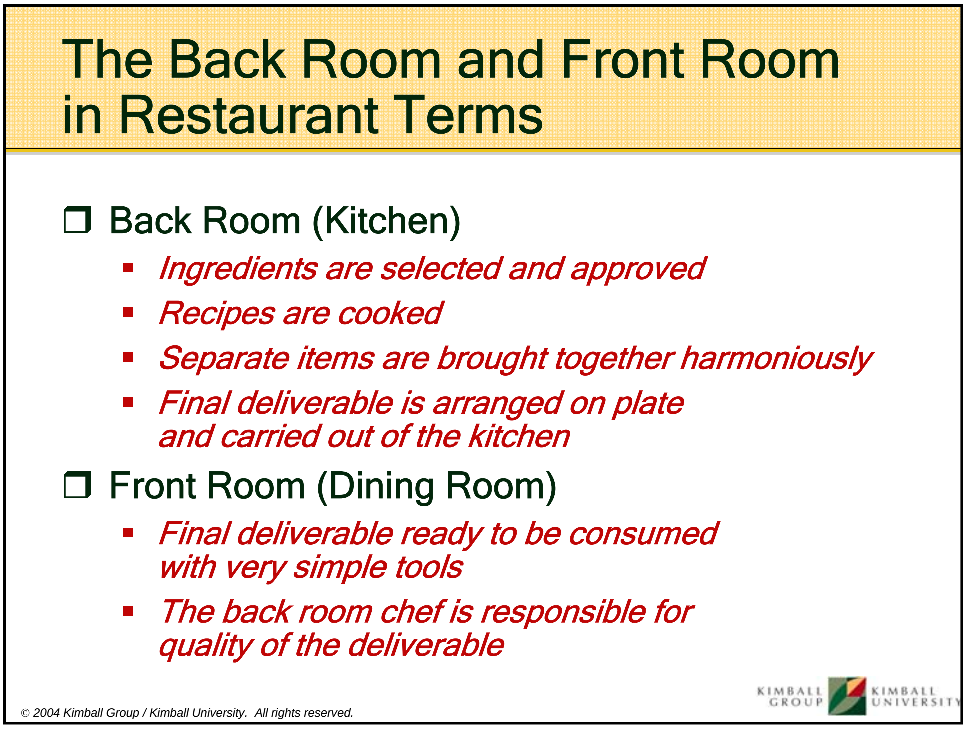
ETL: The Heavy Lifting That Makes BI Possible

Ralph Kimball
SAS - SUGI
April 13, 2005



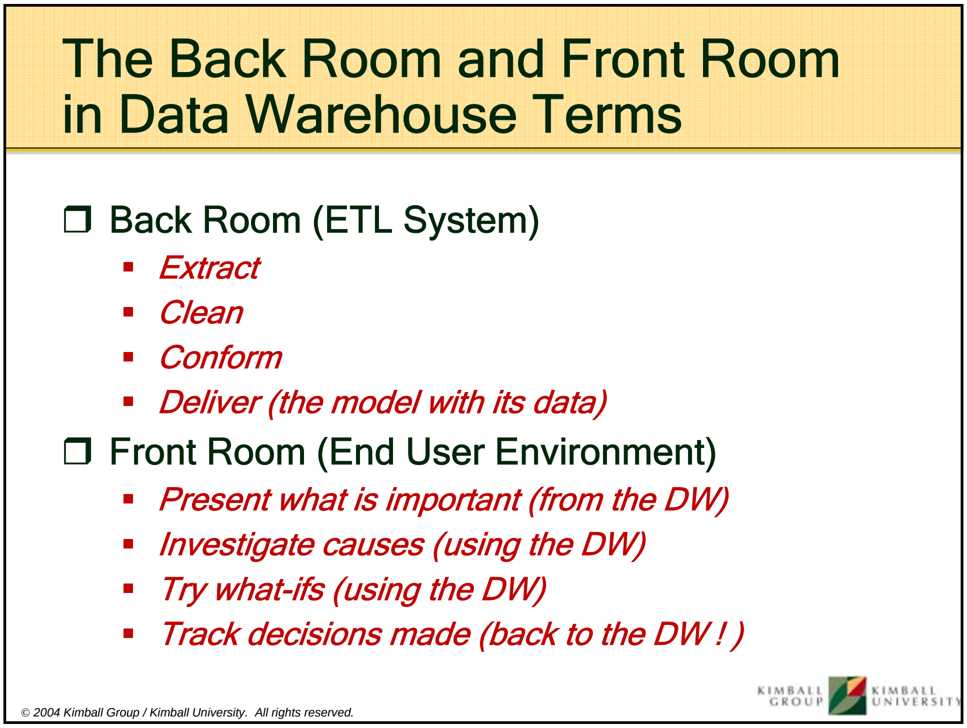
The Challenge of ETL

- Build a cost effective, reliable, extensible, compliant, observable, secure, manageable system for bringing data into the data warehouse and making it ready for end user querying.
- Any questions ? !



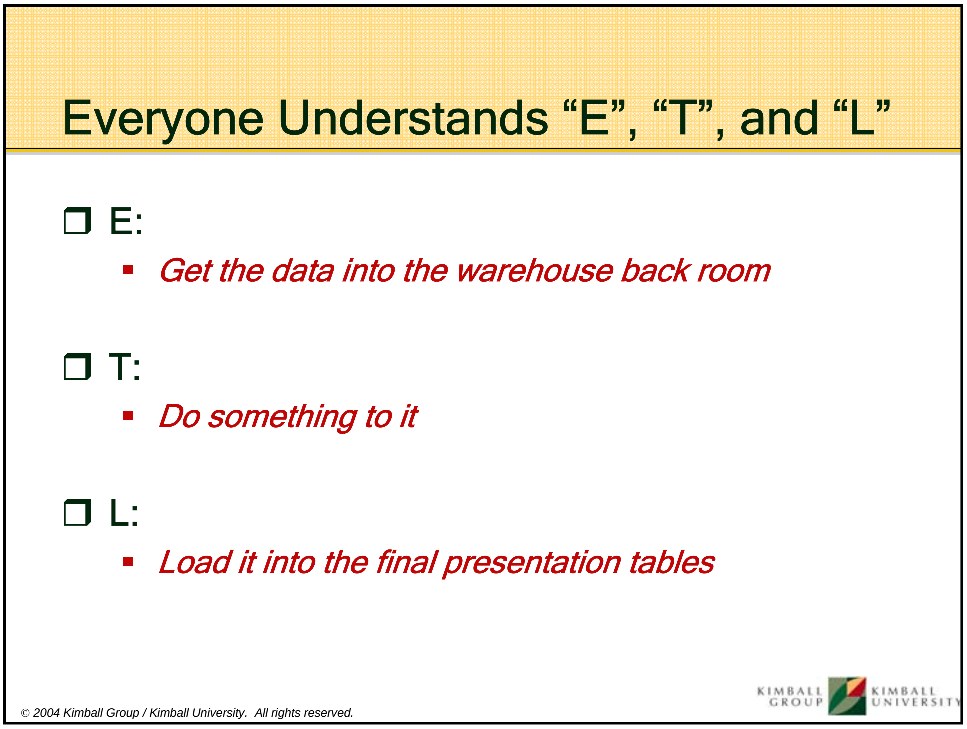
The Back Room and Front Room in Restaurant Terms

- ❒ Back Room (Kitchen)
  - *Ingredients are selected and approved*
  - *Recipes are cooked*
  - *Separate items are brought together harmoniously*
  - *Final deliverable is arranged on plate and carried out of the kitchen*
- ❒ Front Room (Dining Room)
  - *Final deliverable ready to be consumed with very simple tools*
  - *The back room chef is responsible for quality of the deliverable*



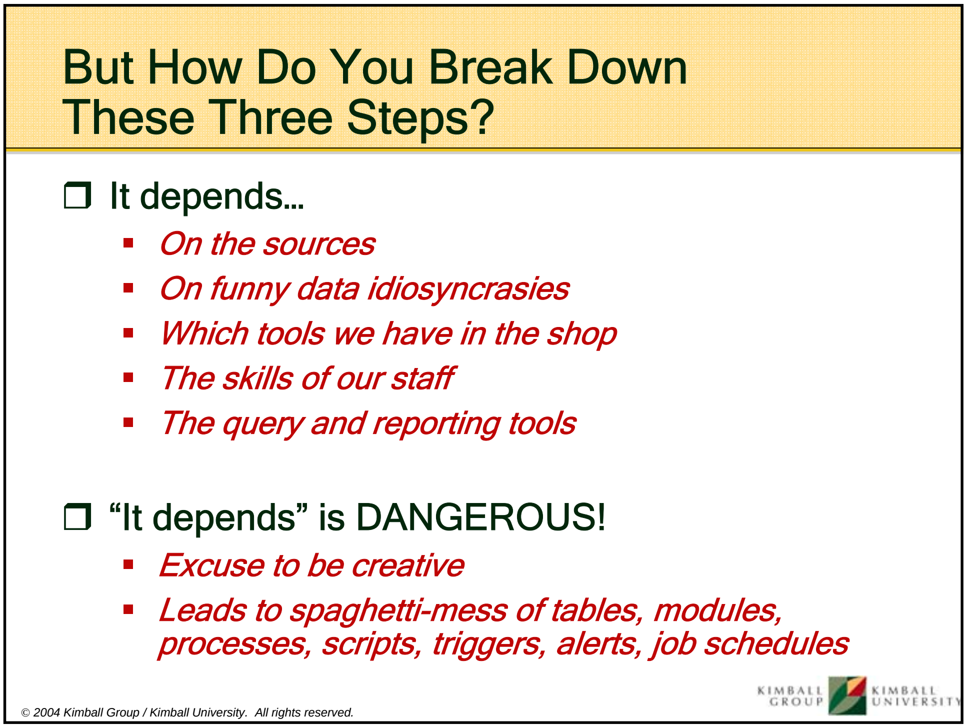
The Back Room and Front Room in Data Warehouse Terms

- ❒ Back Room (ETL System)
  - *Extract*
  - *Clean*
  - *Conform*
  - *Deliver (the model with its data)*
- ❒ Front Room (End User Environment)
  - *Present what is important (from the DW)*
  - *Investigate causes (using the DW)*
  - *Try what-ifs (using the DW)*
  - *Track decisions made (back to the DW !)*



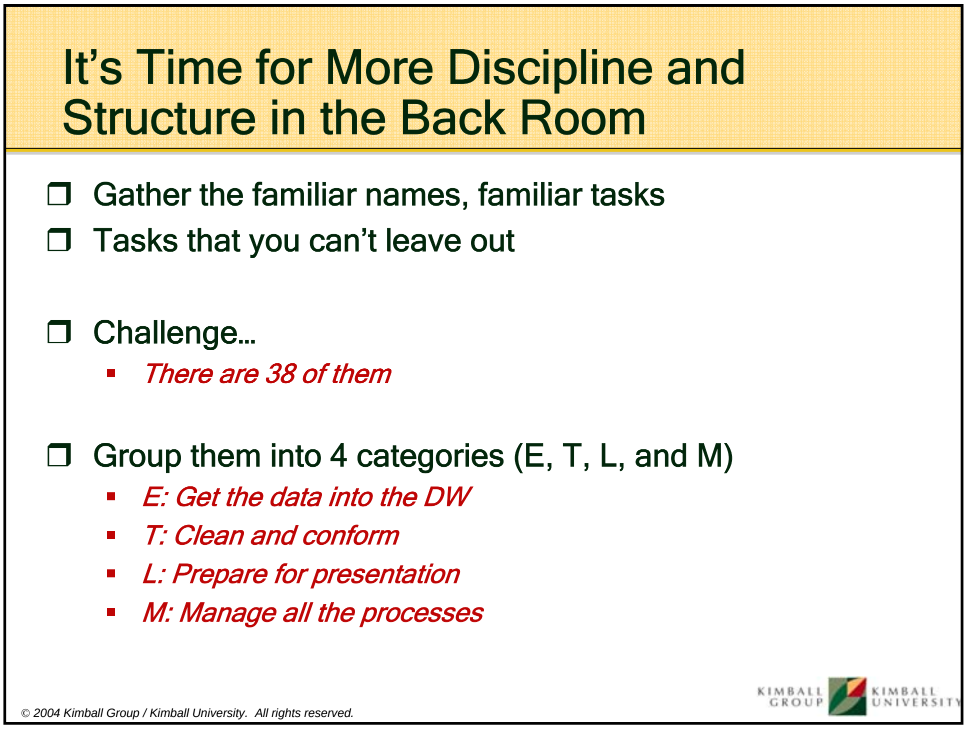
Everyone Understands “E”, “T”, and “L”

- ❒ E:
  - *Get the data into the warehouse back room*
- ❒ T:
  - *Do something to it*
- ❒ L:
  - *Load it into the final presentation tables*



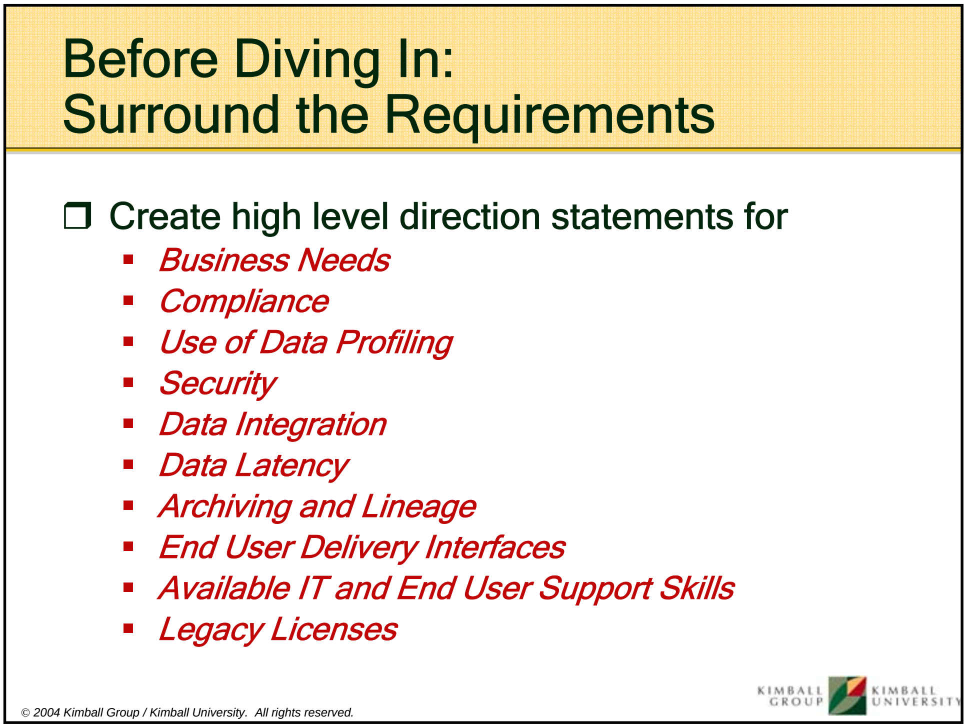
But How Do You Break Down These Three Steps?

- ❒ It depends…
  - *On the sources*
  - *On funny data idiosyncrasies*
  - *Which tools we have in the shop*
  - *The skills of our staff*
  - *The query and reporting tools*
- ❒ “It depends” is DANGEROUS!
  - *Excuse to be creative*
  - *Leads to spaghetti-mess of tables, modules, processes, scripts, triggers, alerts, job schedules*



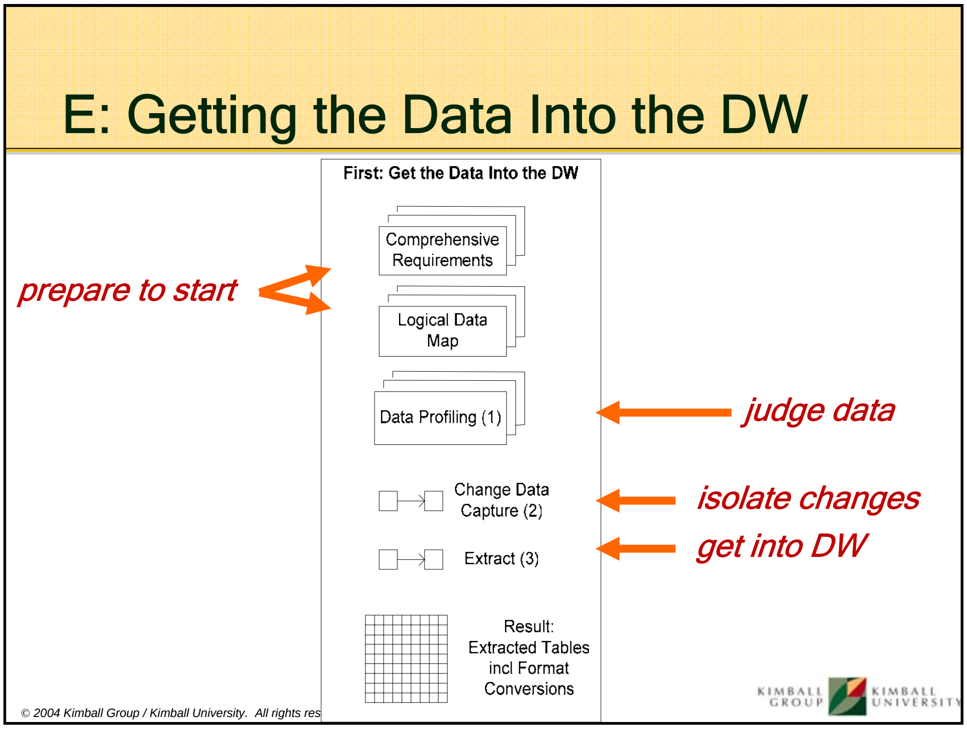
It’s Time for More Discipline and Structure in the Back Room

- ❐ Gather the familiar names, familiar tasks
- ❐ Tasks that you can’t leave out
- ❐ Challenge…
  - *There are 38 of them*
- ❐ Group them into 4 categories (E, T, L, and M)
  - *E: Get the data into the DW*
  - *T: Clean and conform*
  - *L: Prepare for presentation*
  - *M: Manage all the processes*



Before Diving In: Surround the Requirements

- ❐ Create high level direction statements for
  - *Business Needs*
  - *Compliance*
  - *Use of Data Profiling*
  - *Security*
  - *Data Integration*
  - *Data Latency*
  - *Archiving and Lineage*
  - *End User Delivery Interfaces*
  - *Available IT and End User Support Skills*
  - *Legacy Licenses*



E: Getting the Data Into the DW

First: Get the Data Into the DW

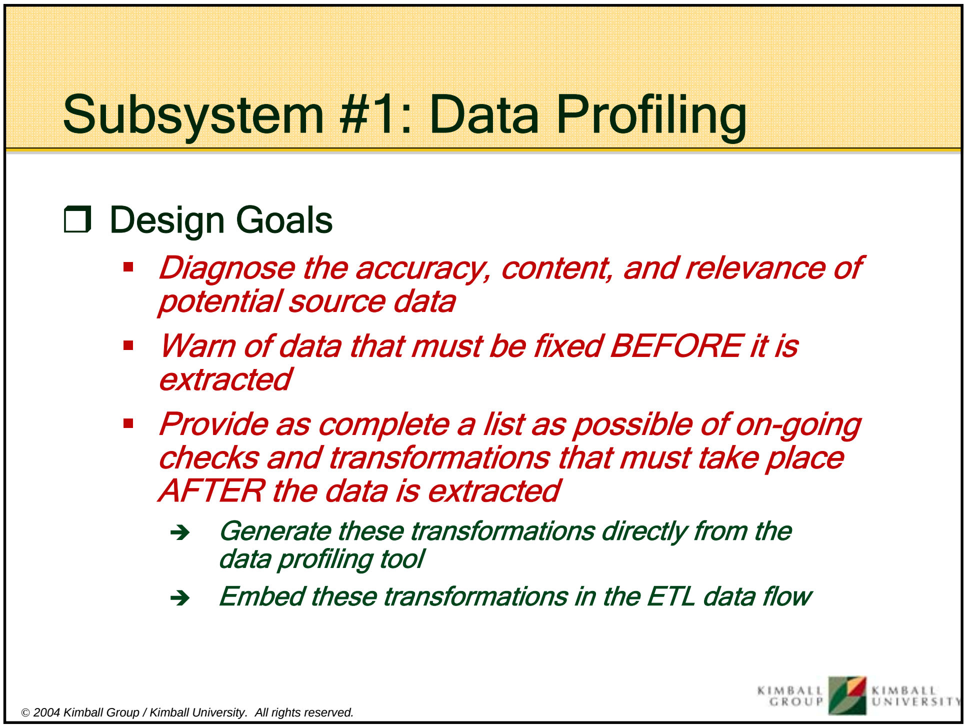
Comprehensive Requirements
Logical Data Map
Data Profiling (1)
Change Data Capture (2)
Extract (3)
Result: Extracted Tables incl Format Conversions

prepare to start

judge data

isolate changes

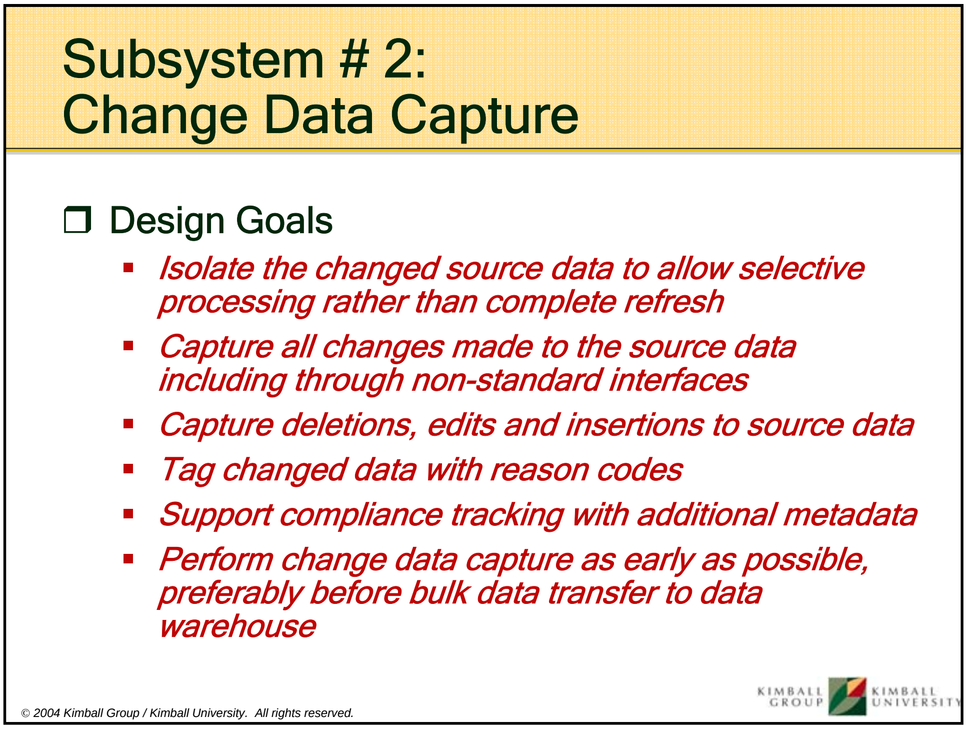
get into DW



Subsystem #1: Data Profiling

❒ Design Goals

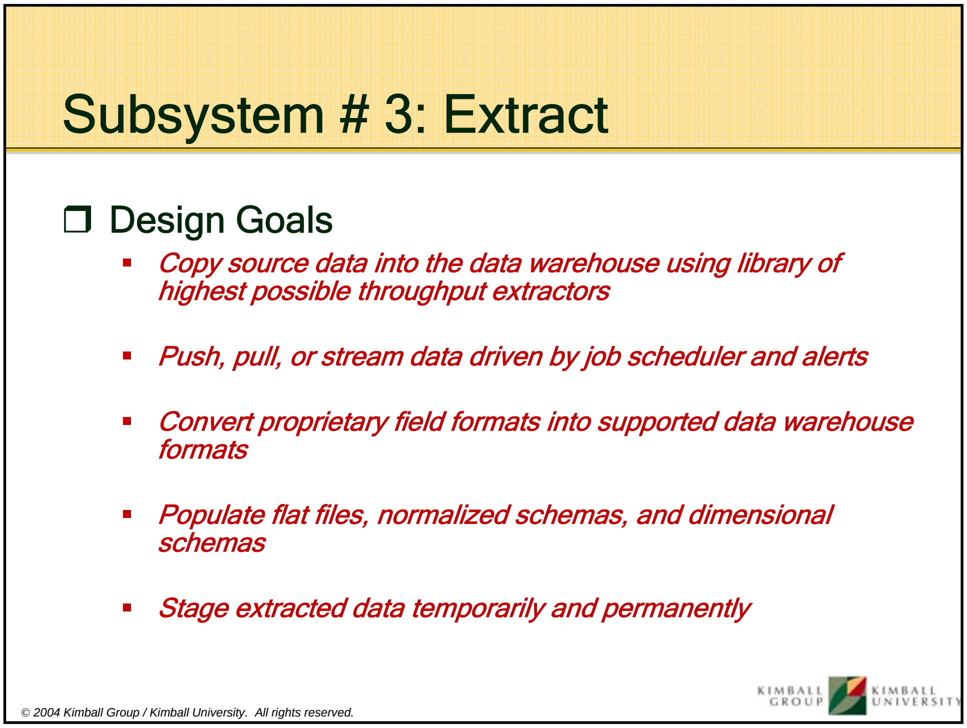
- *Diagnose the accuracy, content, and relevance of potential source data*
- *Warn of data that must be fixed BEFORE it is extracted*
- *Provide as complete a list as possible of on-going checks and transformations that must take place AFTER the data is extracted*
  - ➔ *Generate these transformations directly from the data profiling tool*
  - ➔ *Embed these transformations in the ETL data flow*



Subsystem # 2: Change Data Capture

❐ Design Goals

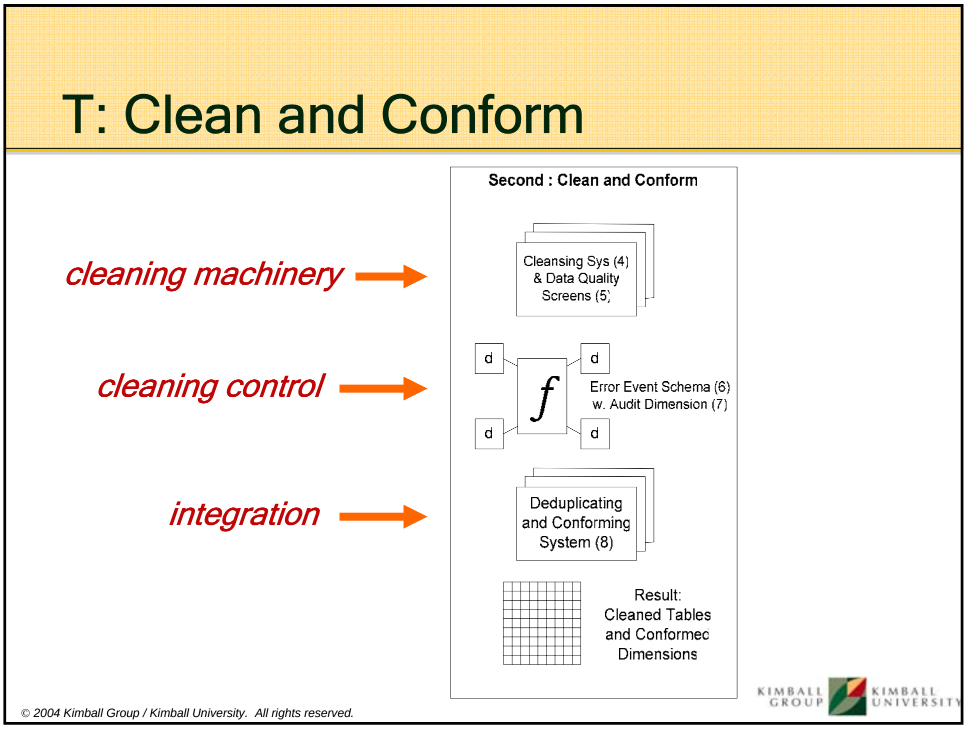
- *Isolate the changed source data to allow selective processing rather than complete refresh*
- *Capture all changes made to the source data including through non-standard interfaces*
- *Capture deletions, edits and insertions to source data*
- *Tag changed data with reason codes*
- *Support compliance tracking with additional metadata*
- *Perform change data capture as early as possible, preferably before bulk data transfer to data warehouse*



Subsystem # 3: Extract

❐ Design Goals

- *Copy source data into the data warehouse using library of highest possible throughput extractors*
- *Push, pull, or stream data driven by job scheduler and alerts*
- *Convert proprietary field formats into supported data warehouse formats*
- *Populate flat files, normalized schemas, and dimensional schemas*
- *Stage extracted data temporarily and permanently*



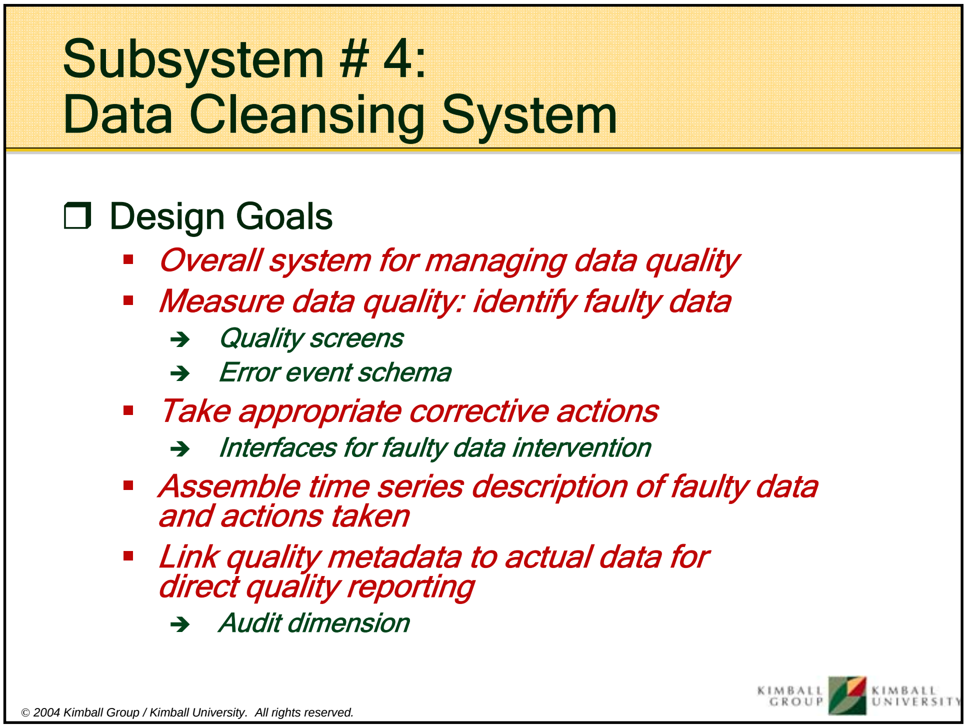
T: Clean and Conform

cleaning machinery →

cleaning control →

integration →

Second : Clean and Conform
Cleansing Sys (4) & Data Quality Screens (5)
d
d
ƒ
d
d
Error Event Schema (6) w. Audit Dimension (7)
Deduplicating and Conforming System (8)
Result: Cleaned Tables and Conformed Dimensions



Subsystem # 4: Data Cleansing System

❒ Design Goals

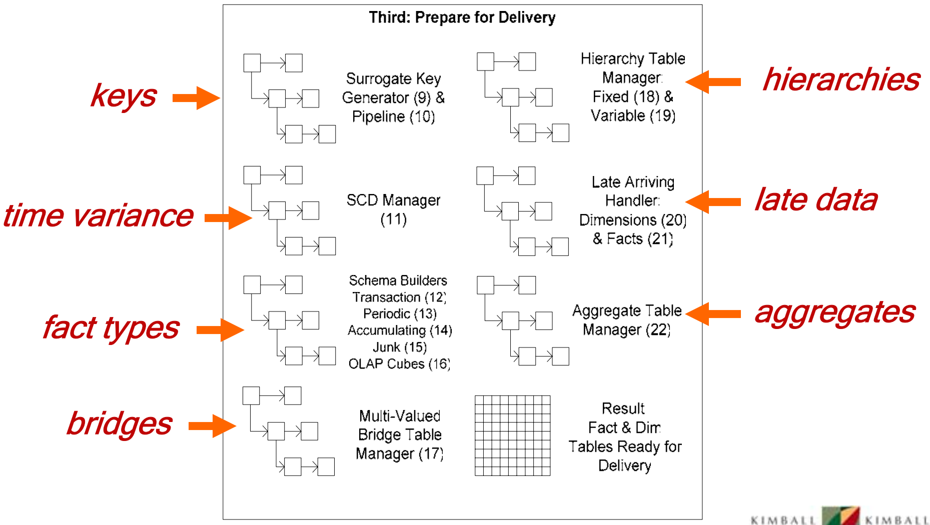
- *Overall system for managing data quality*
- *Measure data quality: identify faulty data*
  - ➔ *Quality screens*
  - ➔ *Error event schema*
- *Take appropriate corrective actions*
  - ➔ *Interfaces for faulty data intervention*
- *Assemble time series description of faulty data and actions taken*
- *Link quality metadata to actual data for direct quality reporting*
  - ➔ *Audit dimension*

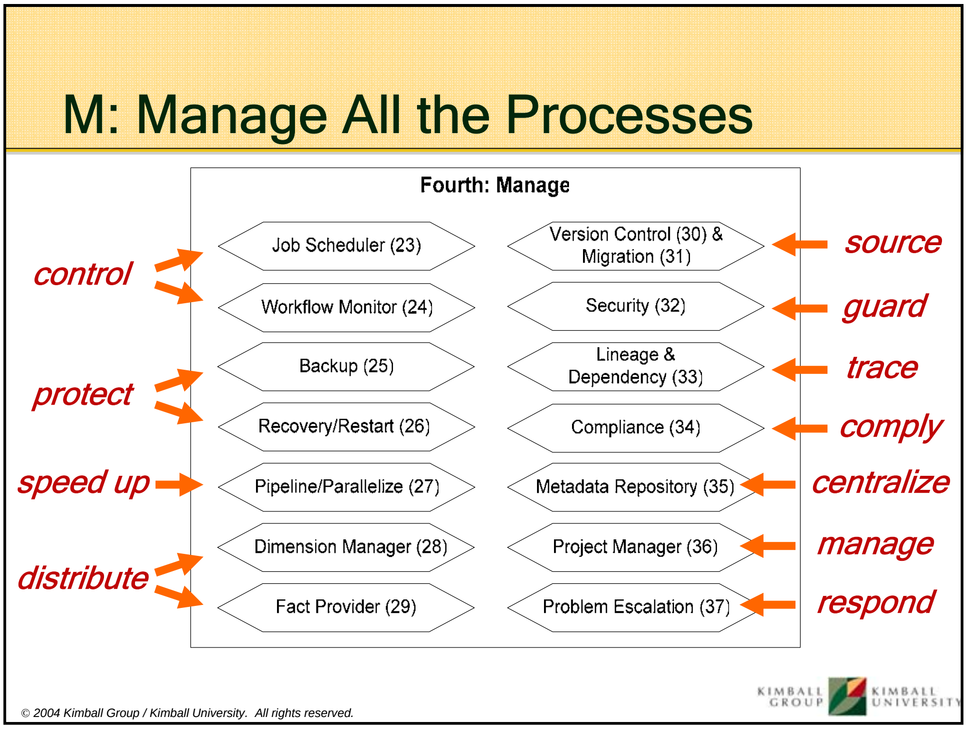
Subsystem #8: Data Conforming

❒ Design Goals

- *Enable drill across applications in multi fact table environments*
- *Enforce common data domains for designated fields in conformed dimension tables*
- *Enforce common business rules for designated fields in conformed fact tables*
- *De-duplicate dimension members within and across dimension tables*
- *Implement survivorship procedure for integrating data from multiple sources*

L: Prepare for Presentation


Third: Prepare for Delivery
keys
Surrogate Key Generator (9) & Pipeline (10)
Hierarchy Table Manager: Fixed (18) & Variable (19)
hierarchies
time variance
SCD Manager (11)
Late Arriving Handler: Dimensions (20) & Facts (21)
late data
fact types
Schema Builders Transaction (12) Periodic (13) Accumulating (14) Junk (15) OLAP Cubes (16)
Aggregate Table Manager (22)
aggregates
bridges
Multi-Valued Bridge Table Manager (17)
Result Fact & Dim Tables Ready for Delivery



M: Manage All the Processes

Fourth: Manage

control → Job Scheduler (23)

control → Workflow Monitor (24)

protect → Backup (25)

protect → Recovery/Restart (26)

speed up → Pipeline/Parallelize (27)

distribute → Dimension Manager (28)

distribute → Fact Provider (29)

Version Control (30) & Migration (31) ← source

Security (32) ← guard

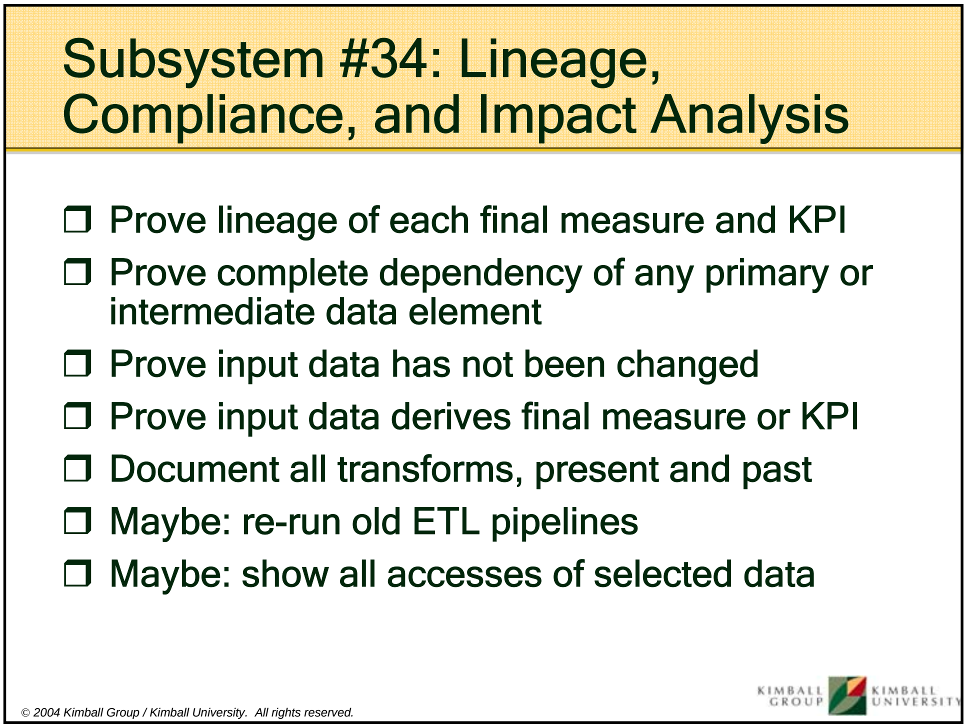
Lineage & Dependency (33) ← trace

Compliance (34) ← comply

Metadata Repository (35) ← centralize

Project Manager (36) ← manage

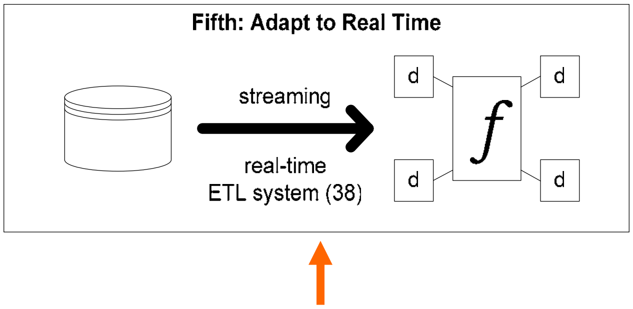
Problem Escalation (37) ← respond



Subsystem #34: Lineage, Compliance, and Impact Analysis

- ❒ Prove lineage of each final measure and KPI
- ❒ Prove complete dependency of any primary or intermediate data element
- ❒ Prove input data has not been changed
- ❒ Prove input data derives final measure or KPI
- ❒ Document all transforms, present and past
- ❒ Maybe: re-run old ETL pipelines
- ❒ Maybe: show all accesses of selected data

And (maybe)
R: Adapt to Real Time



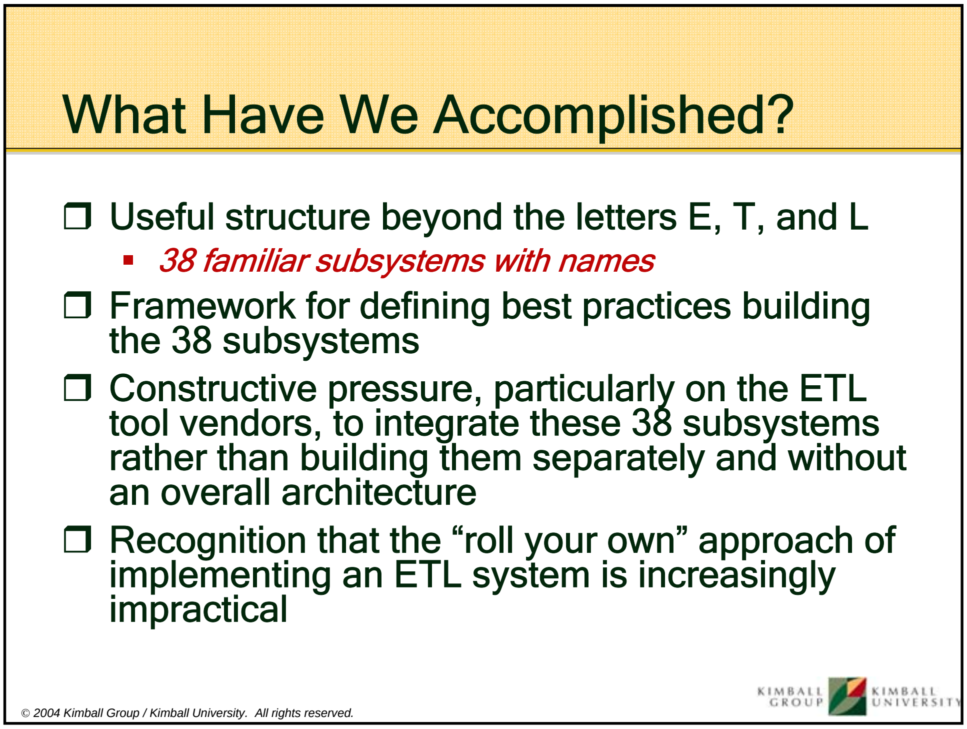
convert

KIMBALL GROUP KIMBALL UNIVERSITY

Subsystem 38: Real-Time ETL

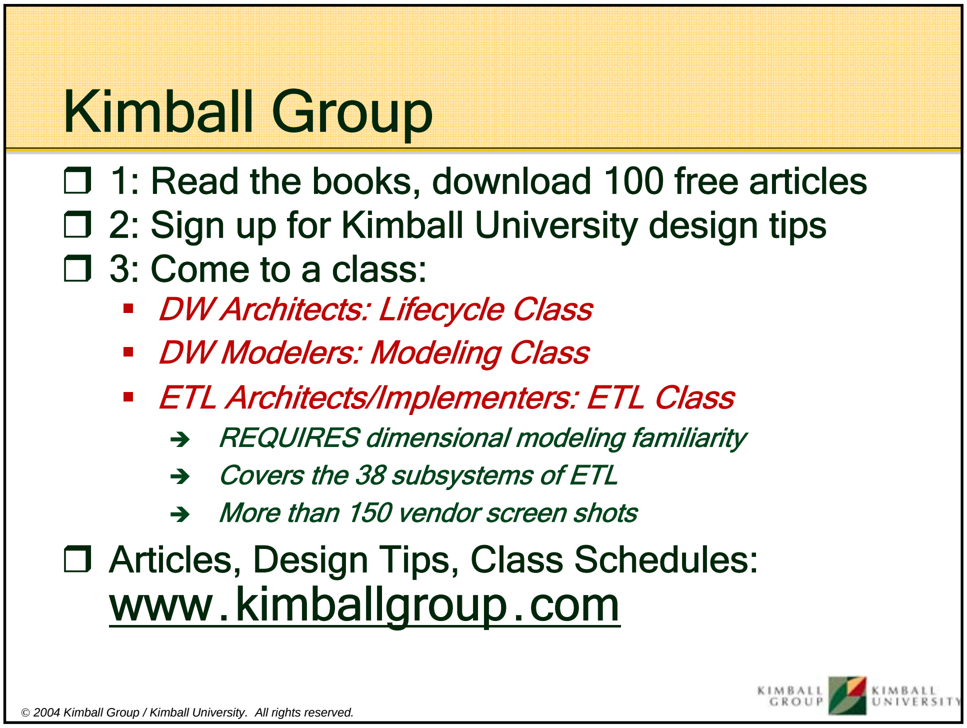
- ❒ "Anything that is too fast for your current ETL"
- ❒ "Change from batch ETL to streaming ETL"
- ❒ Generation 1–The Operational Data Store
  - *Physically separate system between OLTP and DW*
- ❒ Generation 2–The Real-Time Partition
  - *Physically separate extension of existing fact table(s) containing only new activity since the last load of static tables*
  - *In memory, no indexes, no aggregations*

KIMBALL GROUP KIMBALL UNIVERSITY



What Have We Accomplished?

- ❒ Useful structure beyond the letters E, T, and L
  - *38 familiar subsystems with names*
- ❒ Framework for defining best practices building the 38 subsystems
- ❒ Constructive pressure, particularly on the ETL tool vendors, to integrate these 38 subsystems rather than building them separately and without an overall architecture
- ❒ Recognition that the “roll your own” approach of implementing an ETL system is increasingly impractical



Kimball Group

- ❒ 1: Read the books, download 100 free articles
- ❒ 2: Sign up for Kimball University design tips
- ❒ 3: Come to a class:
  - *DW Architects: Lifecycle Class*
  - *DW Modelers: Modeling Class*
  - *ETL Architects/Implementers: ETL Class*
    - ➔ *REQUIRES dimensional modeling familiarity*
    - ➔ *Covers the 38 subsystems of ETL*
    - ➔ *More than 150 vendor screen shots*
- ❒ Articles, Design Tips, Class Schedules: www.kimballgroup.com