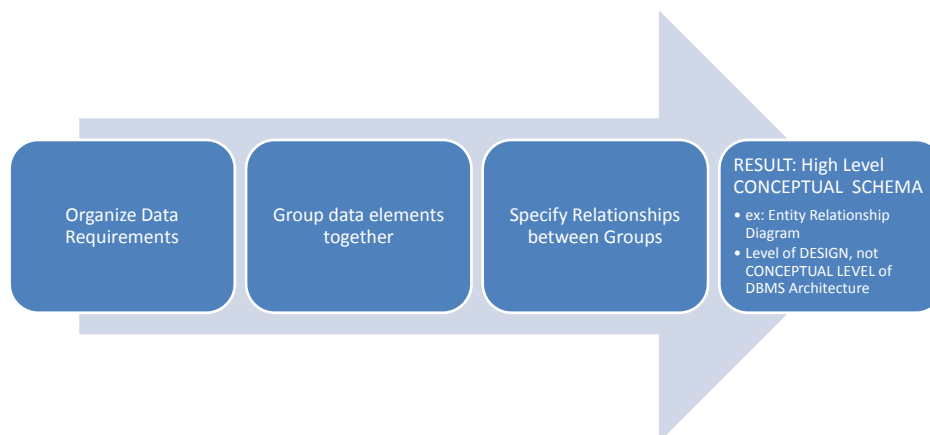# Database Design Process

2-fold Process

- Model some part of the Real World (Miniworld) as DATA
- Determine the OPERATIONS to be used on this model.
- Both have DBMS independent and DBMS specific aspects.

REQUIREMENTS COLLECTION & ANALYSIS

- Discover DATA and OPERATIONS requirements
  - Interaction with the customer
- We will discuss DATA now, OPERATIONS later
- Questions
  - what data must be available?
  - How are data elements to be related?
- RESULT: DATABASE REQUIREMENTS

CONCEPTUAL DESIGN PROCESS

| Organize Data Requirements | Group data elements together | Specify Relationships between Groups | RESULT: High Level CONCEPTUAL SCHEMA
• ex: Entity Relationship Diagram
• Level of DESIGN, not CONCEPTUAL LEVEL of DBMS Architecture |

LOGICAL DESIGN for DATA MODEL

- Implement High Level CONCEPTUAL SCHEMA in some Database
- Use DATA MODEL of that DB: Relational, Network, OO…
- RESULT: Database CONCEPTUAL SCHEMA
  - In RDB, Table Schemas and Constraints

PHYSICAL DESIGN

- **Incorporates knowledge of how the data will be used**:
  - from the operations analysis
- **RESULT: INTERNAL SCHEMA**
  - Layout
  - Clustering (what tables near other tables for faster disk access)
  - Access methods: B-Tree, Hash Table, Indexes

Database Design Process

Requirements
Collection

Logical
Design

High Level
Conceptual Sch.

Database
Internal Sch.

Database
Requirements

Conceptual
Design

Database
Conceptual Sch.

Physical
Design
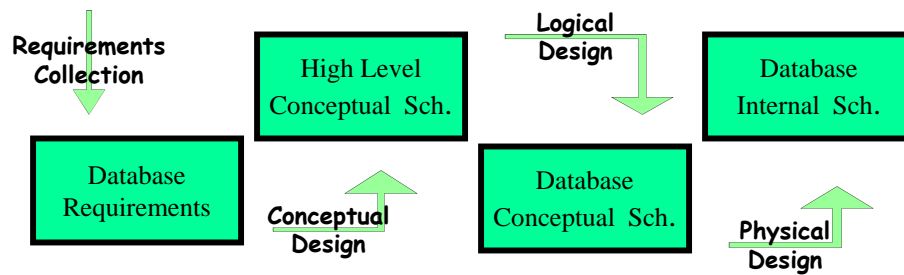
OPERATIONS DESIGN PROCESS

Parallels DATA Design Process

REQUIREMENTS COLLECTION & ANALYSIS

- Discover OPERATIONS requirements
  - Interaction with the customer
- Questions
  - How will data be used?
  - Estimated Frequency of Operations
- RESULT: FUNCTIONAL REQUIREMENTS

FUNCTIONAL ANALYSIS

- Requirements are broken down into operations and sequences
- List of the transactions known to be required
- RESULT: High Level TRANSACTION SPECS
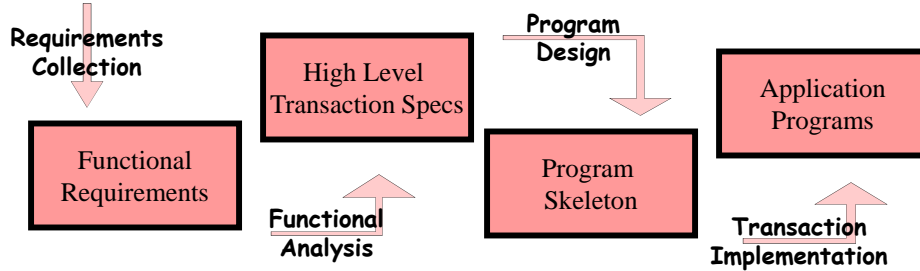 - Info needed for PHYSICAL DESIGN of DB

APPLICATION PROGRAM DESIGN

- Can plan LOGIC of programs without full knowledge of final DB design
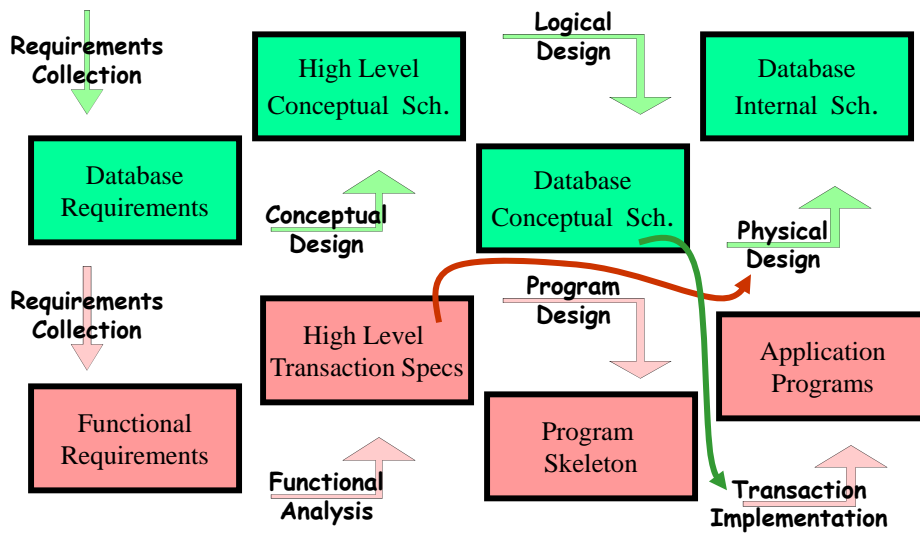- RESULT: Program Skeleton

TRANSACTION IMPLEMENTATION

- Requires at least CONCEPTUAL DB DESIGN
- RESULT: APPLICATION PROGRAMS

Operations Design Process

Requirements Collection

Functional Requirements

High Level Transaction Specs

Functional Analysis

Program Design

Program Skeleton

Application Programs

Transaction Implementation

Combined Design Process

Requirements Collection

High Level Conceptual Sch.

Logical Design

Database Internal Sch.

Database Requirements

Conceptual Design

Database Conceptual Sch.

Physical Design

Requirements Collection

High Level Transaction Specs

Program Design

Functional Requirements

Functional Analysis

Program Skeleton

Application Programs

Transaction Implementation

Oracle Designer

- CASE Tool
  - Computer Assisted S/W Engineering
- Builds Database from ERDs
- Builds Beginnings of Applications for DB
  - Same look & feel
  - Finish with Oracle Developer

High Level Diagrams $\Rightarrow$ DB & Apps.

- Build ERD (in Oracle Dialect)
- **Build Model of business operations using one of several tools**
  - Business Process Modeller
  - Data Flow Diagrammer
  - Hierarchical Input Output (HIPO) chart
- **Cross check**
  - Is all data mentioned in model found in ERD?
  - Is all data collected in ERD needed in model?

ERD to DB Diagram

- **"Click" → Database diagram**
- **Fill-out-forms for DB fields**
  - All caps
  - Limited set of values
  - Always show as Dropdown List
- **"Click" → DDL for DB**
  - Create Table statements
  - Package of triggers enforcing constraints for each table

Model to Applications

- "Click": Model + DB Diag → Raw Apps
- Overall styling, look & feel with Designer
- Final perfection with Oracle Developer, a tool like Visual Basic.

This Course

Entity Relationship Diagram

Database Diagram

DDL

Go to Biz School for modeling.