MSc in Official Statistics
Statistical Computing: Database Design

Andrew Westlake

Survey & Statistical Computing

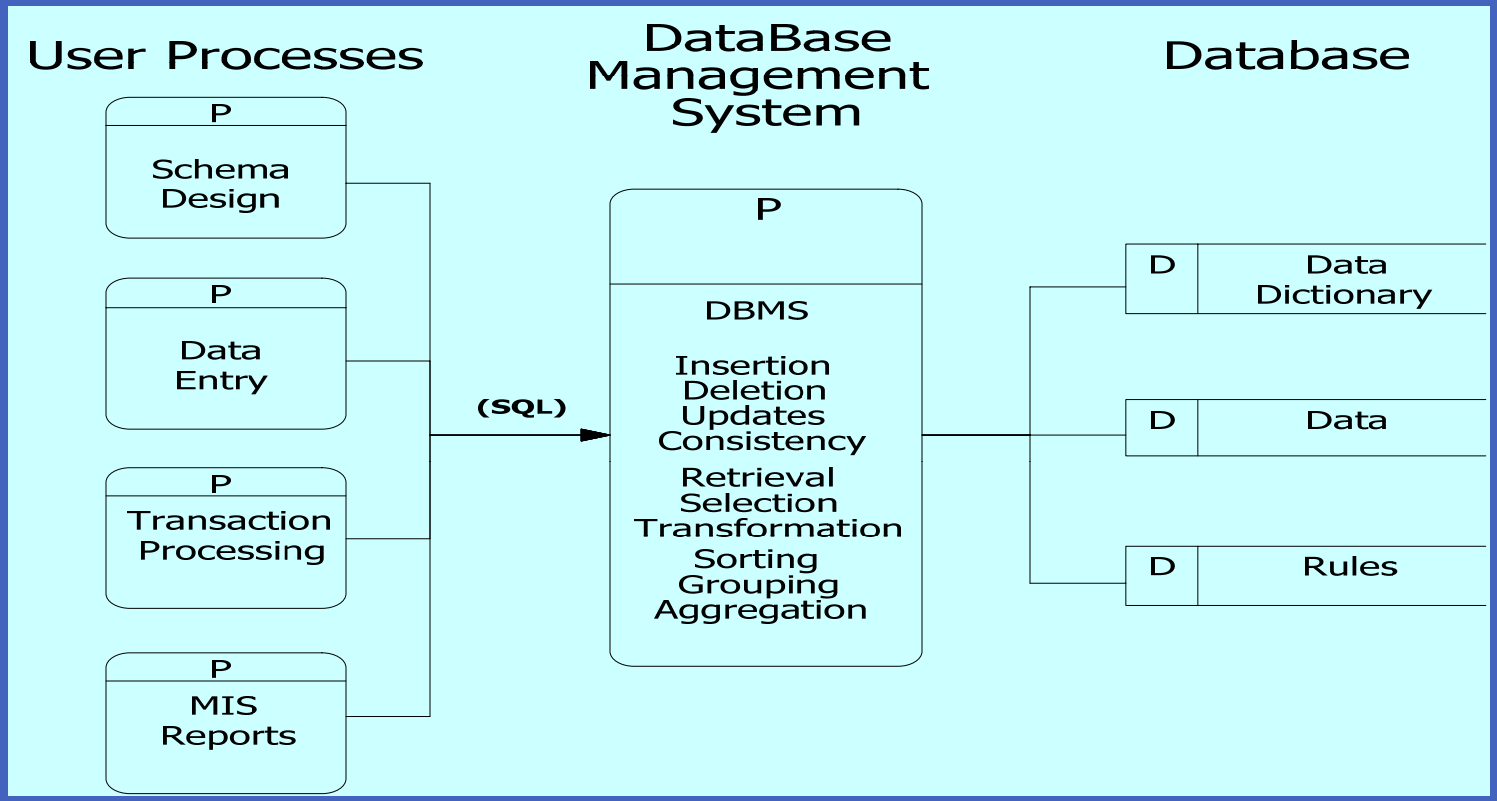
63 Ridge Road, London N8 9NP, UK

+44 (0) 20 8374 4723

AJW@SaSC.co.uk (E-Mail)

www.SaSC.co.uk

Relational Database System Structure



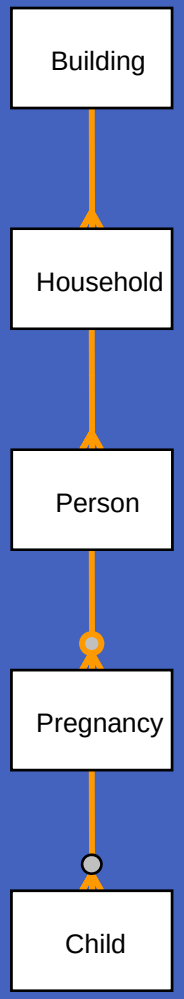
User Processes	***Data Modelling***	SQL	Relational Model	Storage & Access Methods
Applications & Tools, include functionality, semantics appropriate to application	Expressing the real task in terms of the logical model	Standard interface to an RDBMS, syntax and embedding	Specification of functionality, behaviour and scope	Implementation issue, affects performance

Design Processes

- Data Analysis or Modelling
  - » This is the identification of the underlying entities and relationships needed to implement the external views of the database
  - » Makes use of Entity-Relationship diagrams
- Normalisation
  - » move towards a form in which all data values are atomic and redundancy is minimised
- Physical design
  - » This involves the selection of indexes, choice of storage method (where available), introduction of redundancy to speed operations

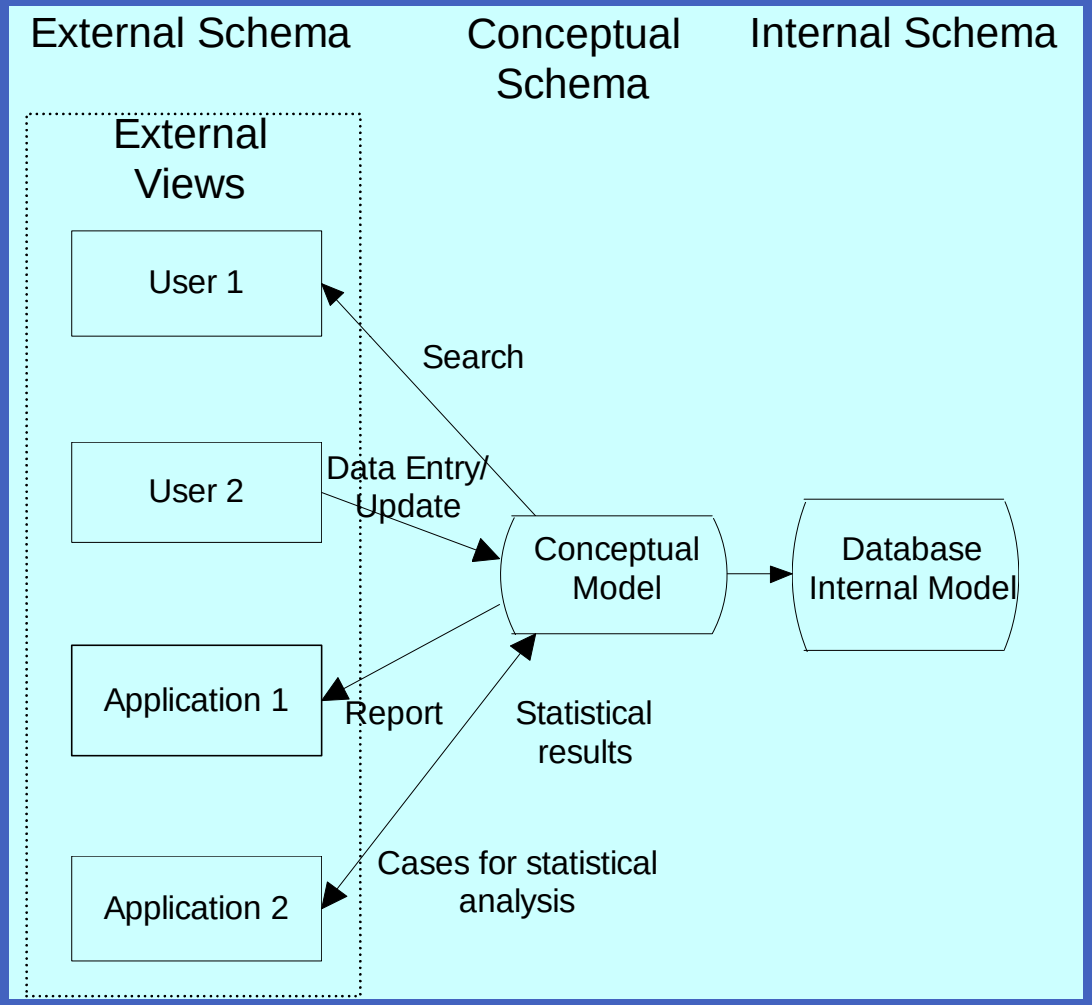
Data Modelling

- Entity-Relationship Modelling
  - » Useful tool for analysing the structure of data
  - » Much observational data has (potentially) complex structure.
    - Statisticians are good at reducing structure to simple forms (rectangular or hierarchical) when designing data collection procedures (for studies, surveys, experiments)
    - This can involve some loss of information
    - ER Modelling helps to identify the structures
  - » Implementation should be easy with a relational database system, or the loss of information from simplification of structure can easily be seen
- Data-Flow Diagrams
  - » Useful for identifying processing requirements
- Tools available for modelling and design
  - » Diagram templates in Visio, e.g.:
    - Database – Database Diagram
    - Flowchart – Data Flow Diagram

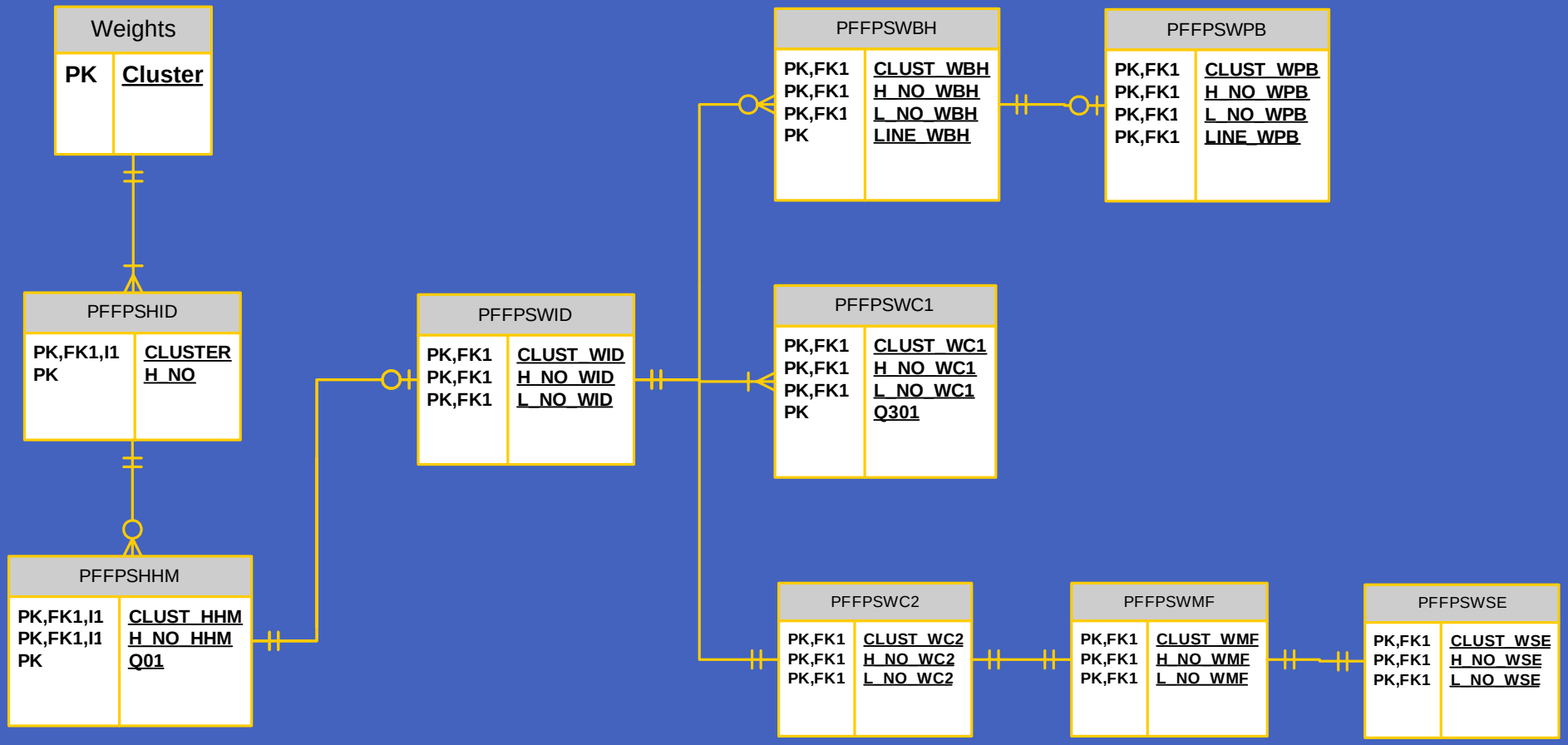


Building the Conceptual Schema

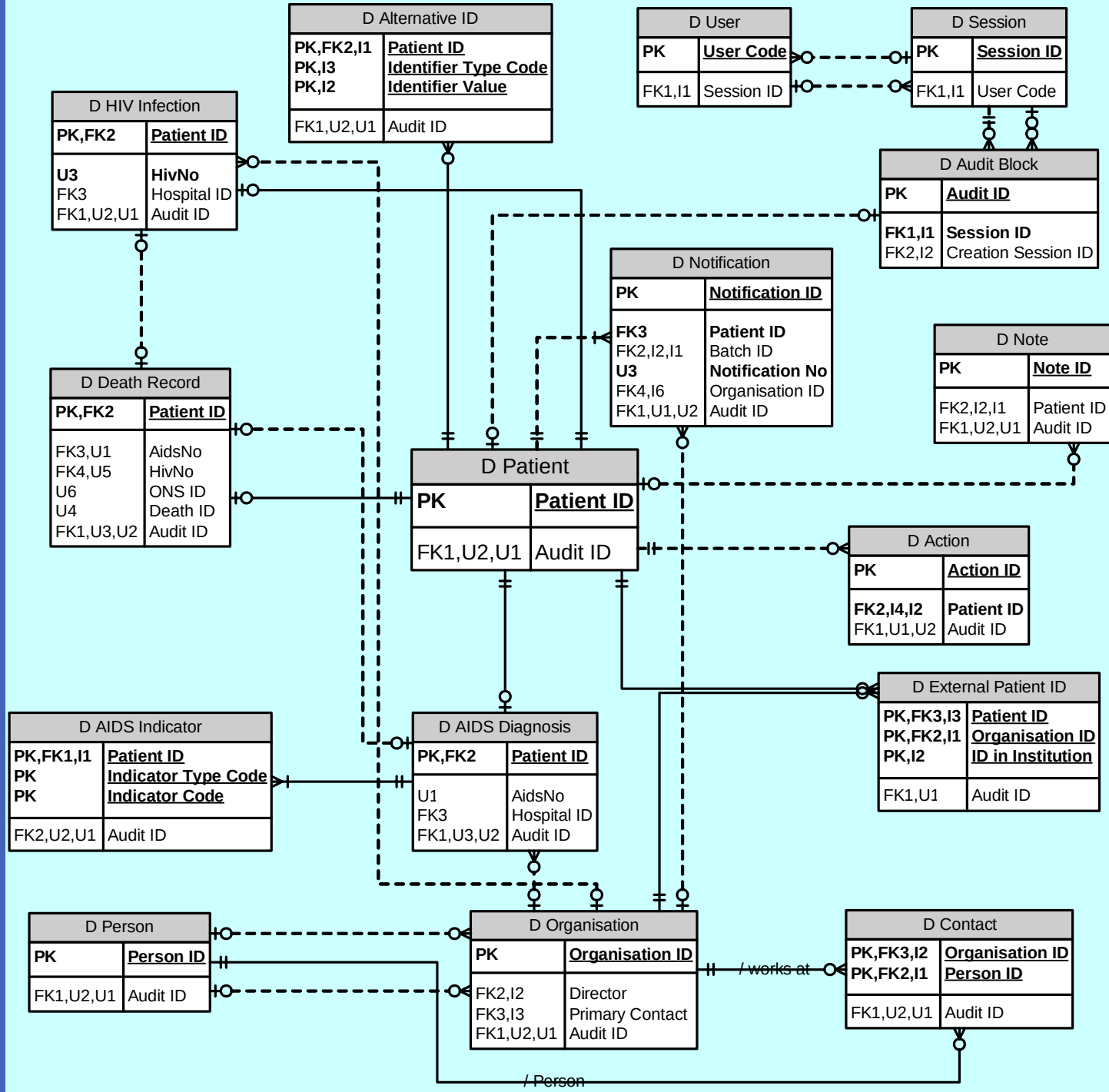
- Represented by ER Model
- Based on external requirements
- Uses understanding of relational model
- Can include physical DB details in most ER diagram software
- Can generate much of the Internal Schema from a detailed ER model



Conceptual Schema for PFFPS



HAP Internal Schema



Components

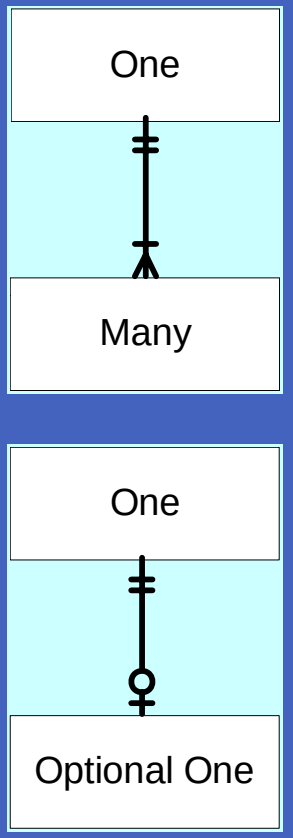
- Entities
  - » The components that have some existence in the system being modelled
  - » Usually lots of Instances of each Entity
- Relationships
  - » Links between Entities (that associate instances of different entities)
    - Author writes books
    - Publisher distributes books
    - Patient consults Doctor
    - Household contains Members, some are Respondents, who have Children
  - » Usually associated with a role or verb
    - Part of, Special case of, Component of
    - Writes, distributes, consults

Relationship Properties

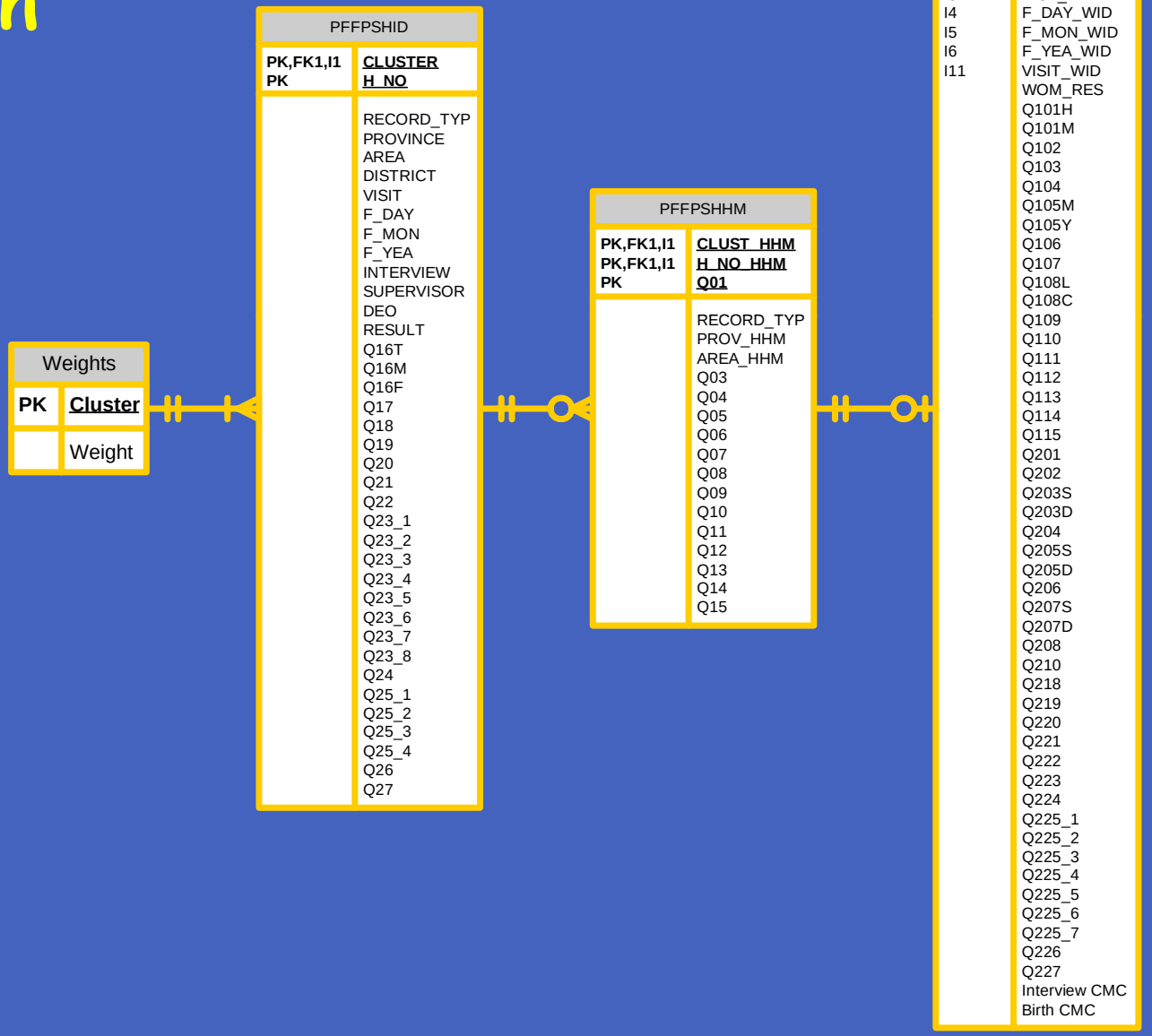
- Relationship has cardinality and status
  - » How many can be associated with each
    - One to Many – e.g. Mother to Child, Household to Member, Patient to Disease Notification
    - One to One – e.g. Birth record to Death record
    - Many to Many – e.g. Patient to Consultant, Product to Purchase Order, Author to Publisher
  - » Many to Many relationships have to be resolved
    - Convert to the One to Many form, by introducing a new entity that represents the link between the two main entities
    - e.g. Consultation between Patient and Consultant, Order Item of Purchase Order, Contract between Author and Publisher
  - » Does the association have to exist?
    - Does the Mother of a Child have to exist in the database?
    - Does the Household have to exist for a Member
    - Related to Referential Integrity

Diagram Conventions

- Two main conventions
  - » Crow's-feet (as shown)
  - » UML associations (arrows)
- Diagram Software
  - » MS Visio – general diagram system, special ERD (and UML) facilities in Professional version
  - » Rational Rose, Together, Poseidon, ...
    - full UML systems with code generation



PFFPS - Main structure



Keys implement Relationships

- A **Candidate Key** is a set of attributes which, taken together, uniquely identify each tuple
  - » Several such Keys may exist, and at least one must always exist.
- The **Primary Key** for a relation is arbitrarily nominated from among these
  - » The selection of a Key should be based on the conceptual uniqueness of the attributes (i.e. on the Domains), not on the actual (subset of possible) values in a relation at any particular time
- A **Foreign Key** is defined over the same domain as a Primary Key, and so can provide a link between tuples
  - » The Cluster and Household number in a Household Member record together form the Foreign Key into the Household record

Properties of Keys

- Keys are usually implemented through **Indexes**
  - » An Index is a physical structure which stores information about the order and location of data values for a set of attributes, and which speeds up retrieval of subsets of records.
- A relationship can be **Identifying**
  - » If the Primary Key of the parent table is part of the Primary Key of the child table

Demonstration

- Entities in Visio
  - » PFFPS

Specialisation

- Some instances of an Entity may have additional attributes
  - » A Manager may have more information than other Employees
  - » In surveys, make conditional sections into separate entities
- Create additional Entities, with One to Optional One relationship to the original
  - » As in PFFPS – selected women interviewed
- Where there are a set of alternatives, create a Category link
  - » E.g. an employee's Job Type may determine which additional information record is needed (but only one is allowed)

Normalisation

- All data values are Atomic
  - » Simple values, codes or measurements, no overloading
    - But complex values in Object-Relational DB, SQL:1999
- Remove redundant information
  - » Do not repeat things
    - Don't put household information with the members
    - Don't repeat information about people in different contexts
  - » Targeted at integrity
    - If something changes you only have to change the database in one place
    - Important for rapidly changing data, e.g. transaction processing
  - » If use requires redundancy, then achieve through Views
  - » Can be inefficient, so may de-normalise for implementation
    - Code may be needed to enforce integrity – not an issue with static (statistical) data
    - Provide procedures to reconstruct derived tables when data changes

Example

- Design for Statistical Metadata
  - » Can see value labels as a normalisation problem
  - » Or as requiring Entity design

Data with labels

Cluster	HID	HH	Q06	Gender	Q07	Q08	Status
77	10	1	1	Male	46	1	Currently Married
77	10	2	2	Female	46	1	Currently Married
77	10	3	2	Female	27	3	Divorced
77	10	4	1	Male	2	6	Neve Married
77	10	5	2	Female	21	6	Neve Married
77	10	6	2	Female	19	6	Neve Married
77	10	7	2	Female	17	6	Neve Married
77	10	8	1	Male	13	6	Neve Married

- Clearly wrong to store labels with the data
  - » Inefficient, wastes storage through repeated strings
  - » Integrity issues, wrong to change individual occurrences
- Labels are needed when information is displayed

Normalisation Approach

Cluster	HID	HH	Q06	Q07	Q08
77	10	1	1	46	1
77	10	2	2	46	1
77	10	3	2	27	3
77	10	4	1	2	6
77	10	5	2	21	6
77	10	6	2	19	6
77	10	7	2	17	6
77	10	8	1	13	6

Code	Label
1	Male
2	Female

Code	Label
1	Currently Married
2	Widowed
3	Divorced
4	Separated
5	Marria. Contract Not Lived Tog
6	Neve Married

- Take each type of label into a separate table
  - » Reconstitute the labelled data (for analysis) with views
- Lots of separate tables for labels
  - » Scaling problem
  - » Same functionality for all such tables (use and management)

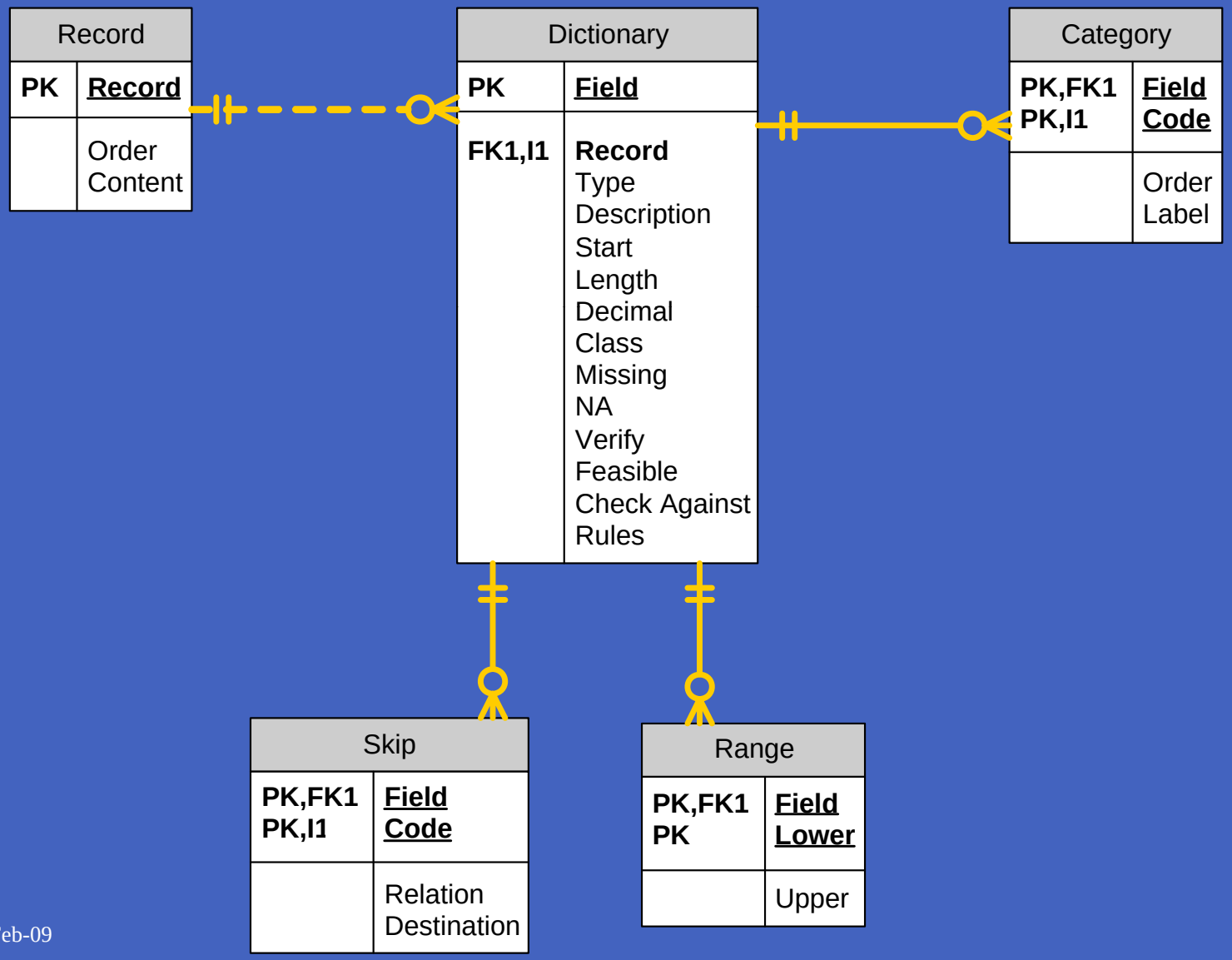
Entity Approach

- At the metadata level
  - » Variable and Code are both entities
  - » Build metadata management functionality at this level
  - » Link to data in views
  - » Cf. Statistical Packages
- Example from PFFPS
  - » Will see more in HAP and other contexts

Field	Description	Missing	NA
Q06	Sex	9	
Q07	Age	99	
Q08	Marital Status	9	6
Q09	Education Level	9	3
Q10	Highest Class Passed	99	97
Q101H	Start Time (Hour)	99	
Q101M	Start Time (Minutes)	99	
Q102	Early life area	9	

Field	Code	Order	Label
Q06	1	1	Male
Q06	2	2	Female
Q08	1	1	Currently Married
Q08	2	2	Widowed
Q08	3	3	Divorced
Q08	4	4	Separated
Q08	5	5	Marria. Contract Not Lived Tog
Q08	6	6	Neve Married
Q09	1	1	Formal Schooling
Q09	2	2	Only Informal or Quranic Edu.
Q09	3	3	On Formal or Informal Educatio
Q09	8	4	Don't Know
Q102	1	1	City
Q102	2	2	Town
Q102	3	3	Village

Metadata tables in PFFPS



PFFPS - Metadata

- The metadata structure is not ideal
  - » Derived from and also supports data entry programme
  - » Includes physical layout for DE records
  - » Relates to records, not tables
- Used to produce printed Dictionary
  - » No data entry in Access, so constraints not used in this form
  - » Can be used to label outputs

Demonstration

- Entities in Visio
  - » PFFPS Variable labels and codes (metadata)
- Forms in MS Access
  - » PFFPS Metadata