Relational Database Fundamentals

The what, why, (when), how and where of Relational Database Management Systems.

MWGLAIUG 99 - Introductory Session
Peter C.G. Veenstra - M.J. Harden Associates Inc.

Outline

- Relational Database Management System (RDBMS)
- A Definition and the concept
- WHAT is it?: Software and Process
- WHY?: Advantages and Challenges
- HOW?: Design, Entity-Relationship Model, Modeling, Set Theory, Primary Key, Foreign Key, and Relationships
- WHERE?: Hybrid Systems - ARC/INFO, ShapeFile
- WHERE?: Spatially extending the database - SDE

What is an RDBMS?: Definition

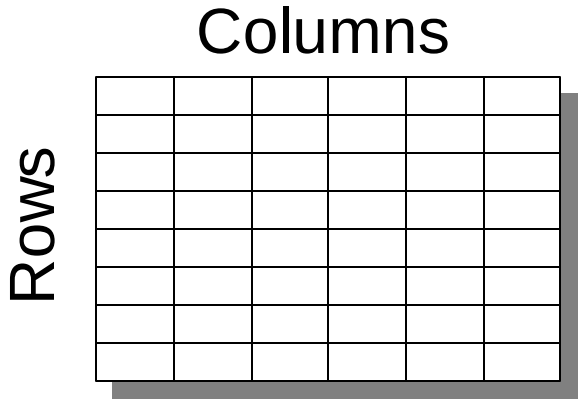
- RBDMS: Relational Database Management System
- RDBMS: a mathematical theory and a software tool
- **<u>The theory ...</u>**
  - A method for structuring data in the form of sets of records or tuples so that relations between different entities and attributes can be used for data access and transformation - Burroughs, 1986
  - A database structure commonly used in GIS in which data is stored based on 2 dimensional tables where multiple relationships between data elements can be defined and established in an ad-hoc manner - Croswell, 1991

What is an RDBMS?: Definition

Entities and Relationships

Each and every crew belongs to one and only one nodes

Two Dimensional Tables



What is an RDBMS?: Definition

- **The software tool ...**
  - Relational Database Management System - a database system made up of files with data elements in two-dimensional array (rows and columns). This database management system has the capability to recombine data elements to form different relations resulting in a great flexibility of data usage. (after Martin, 1976)
  - RDBMS: A database tool for storing data in a logical, easy to understand fashion so that users can query the database in an effort to form meaningful and logical relationships.Outline
  - INFORMIX, DB2, Oracle, Sybase, SQL Server, SQL Anywhere, Microsoft Access

WHAT is an RDBMS?: Concept

- The basis for the relational database management system (RDBMS), the relational model was proposed by Dr. E.F. Codd in 1970

- The relational model contains the following components:
  - *Collection of objects or relations (Tables)*
  - *Set of operations to act on the relations*
  - *Data integrity for accuracy and consistency*

- A relational database is a database that is perceived by the user as a collection of relations or two-dimensional tables.
- Relational databases are manipulated a set at a time rather than a record at a time. SQL is used to manipulate relational databases.

WHAT is an RDBMS?: Software

- *DBMS* - Software package for storage, manipulation and retrieval of data from a database. Serves many users simultaneously, built on the idea of set theory and relations, a means to eliminate redundancy.

- *Kernel* - Core software component, controls query processing, access paths to data, user access management, storage management, indexing, transaction processing and read/update information.

- *Query language Interface* - Wrapped around the Kernel. Receives ad-hoc requests, interprets, issues data requests and presents results back to the users of the database - Screen Forms Manager.

- *Interactive Query Tool* – Access, edit and update of one or more linked data tables using screen based forms.

WHAT is an RDBMS?: Software

- *Utilities* - Import/export/backup tuning tools, Parameterization tools and report writers.

- Database (Physical)
  - *Physical Storage of the data objects within data files*
  - *Contains the system catalogues (Data Dictionary)*

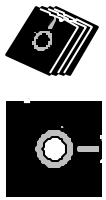
- User, query, processing, input, tables, results, onion drawing

WHAT is an RDBMS?: Process

Internet Applications, Custom Applications, & Interactive Query Tool (SQL)

Queries
User Access

Results
Reports
Web Pages



Physical Implementation
Operating System Files

RDBMS

Structure
Tables, Data Dictionary

Query Language Interface

Processes
Kernel
User Security
Transactions
Processing

Tools
Import, Export, Tuning

WHY an RDBMS?: Advantages

- Rigorous design methodology (Normalization, Set Theory)
- All other database structures can be reduced to a set of relational tables
  - *Mainframe databases use Network and Hierarchical methods to store and retrieve data.*
  - *Access to the data is hard-coded.*
  - *It is very difficult to extract data from this type of database without some pre-defined access path.*
  - *Extremely fast retrieval times for multi-user, transactional environment.*
- Ease of use compared to other database systems

WHY an RDBMS?: Advantages

- Modifiable - new tables and rows can be added easily
- The relational join mechanism
  - *Based on algebraic set theory (A Set is a group of common elements where each member has some unique aspect or attribute)*
  - *Very flexible and powerful*
- Fast Processing
  - *Faster processors, multi-threaded operating systems and parallel servers*
  - *Indexes, fast networks and clustered disk arrays*
  - *2 Million simultaneous users (Microsoft SQl Server)*

WHY an RDBMS?: Challenges

- Expensive solutions that require thorough planning
- Easy to create badly designed and inefficient database designs if there isn't any proper data analysis prior to implementation

The HOW of an RDBMS: Design

- Objective is to define and model the things of significance about which the business needs to know or hold information, and the relationships between them.
- High level look at the database.The art of distilling a business requirements statement into a conceptual diagram.
- Business requirements are determined from users needs assessments.
- Is high level abstraction and occurs before database design and implementation.
- Is independent of hardware or software.
- The goal is to develop an Entity-Relationship model representing the information requirements of a business.

The HOW: Entity-Relationship

- An Entity-Relationship Data Model should accurately model the organization's information needs and support the functions of the business.
  - *Entities - the things of significance about which information needs to be held.*
  - *Relationships - how the things of significance are related*
  - *Attributes - the specific information that describes, quantifies or qualifies an entity.*

- An entity-relationship model is an effective means for collecting and documenting an organization's information requirements.
- Robust Syntax. User Communication. Ease of Development. Definition of Scope. Integration of Multiple Applications.

The HOW of an RDBMS: Modeling

- Identify and model entities.
- Analyze and model the relationships between the entities.
- Analyze and model the attributes that describe the entities.
- Identify unique identifiers for each entity.
- Develop a complete entity-relationship model from the statement of information requirements.
- Normalize the entities and relationships between them.

The HOW ... Set Theory

- The power of a relational database comes from its ability to relate significant data together.
- Database tables are related to each through columns of data sharing identical data (called keys).
- Each table is based on mathematical set theory (each element in the set must be unique).
- Relational databases are usually manipulated a set at a time rather than a record at a time.
- The Structured Query Language (SQL) is used to manipulate relational databases.

The HOW ... Set Theory

A = {1, 2, 4, 5, 6, 7, 9 } Is a set!

B = {1, 2, 3, 3, 4, 5, 2 } Is not a set!

C = {3, 4, 5, 6, 2 } Is a set!

The HOW ... Set Theory

- A SET is a series of unique elements
- A relationship is formed when an Intersection occurs between Set A and Set B on a common element

C = {3, 5, 6, 2}

A = {1, 2, 4, 5, 6, 7, 9}

A = {1, 2, 4, 5, 6, 7, 9}

C = {3, 4, 5, 6, 2}

The HOW of an RDBMS: Tables

- A table describes or models phenomena that are of significance to the business and are abstractions of reality.
- Tables consist of rows of data that are uniquely identified from other rows of data. Each row represents or corresponds to an instance of the phenomena being modeled.
- Each table is made of columns or attributes that describe the phenomena being modeled.
- A table is often the implementation of an entity
- Tables are the logical and perceived data structure not the physical data structure, in a relational system.

The HOW of an RDBMS: Tables

Personnel Table

Columns (Fields)

Rows (Tuples)

ID	First	Last	Salary	DeptNO
3	James	Kirk	50000	1
4	Monty	Scott	35000	2
5	Spock		45000	1
9	Ensign	Goner	15000	1
6	Bones	McCoy	42500	3

The HOW … Primary Keys

- A Primary key (PK) is a column or set of columns that uniquely identifies each row in a table.
- Each table must have a primary key and a primary key must be unique.
- A PK consisting of multiple columns is called a Composite or Compound Primary Key.
- No part of the PK can be null.
- Tips for Identifying Primary Keys
  - *Must be a unique value.*
  - *The value in the primary key for each tuple or row should never change.*
  - *The primary key is best auto-generated – should not contain business information*

The HOW ... Primary Keys

Personnel Table

Columns (Fields)

Rows (Tuples)

ID	First	Last	Salary	DeptNO
3	James	Kirk	50000	1
4	Monty	Scott	35000	2
5	Spock		45000	1
9	Ensign	Goner	15000	1
6	Bones	McCoy	42500	3

Primary Key

The HOW ... Foreign Keys

- A Foreign Key (FK) is a column or combination of columns in one table that refers to a primary key in the same or another table.
- A FK must match an existing primary key value (or else be null).
- If a FK is part of a Primary Key, that FK cannot be NULL.

The HOW ... Foreign Keys

Personnel Table

ID	First	Last	Salary	DeptNO
3	James	Kirk	50000	1
4	Monty	Scott	35000	2
5	Spock		45000	1
9	Ensign	Goner	15000	1
6	Bones	McCoy	42500	3

Foreign Key

Primary Key

Department Table

DeptNO	Department
1	Command
2	Engineering
3	Medical

The HOW … Relationships

- In order for a relation to be established between two tables, they both must contain a common data element
  (e.g. a field that has been defined the same in both tables).

Each (and every)
ENTITY1 (must be /or/ may be)
RELATIONSHIP NAME (related to…)
(one or more /or/ one and only one) ENTITY2

Each crew member must belong to one and only one department.

The HOW ... Relationships

Personnel

ID	First	Last	Salary	DeptNO
3	James	Kirk	50000	1
4	Monty	Scott	35000	2
5	Spock		45000	1
9	Ensign	Goner	15000	1
6	Bones	McCoy	42500	3

Department

DeptNO	Department
1	Command
2	Engineering
3	Medical

The HOW ... Relationships

Joined Tables

ID	First	Last	Salary	DeptNO	Department
3	James	Kirk	50000	1	Command
4	Monty	Scott	35000	2	Engineering
5	Spock		45000	1	Command
9	Ensign	Goner	15000	1	Command
6	Bones	McCoy	42500	3	Medical

SQL Statement ...

```
SELECT a.Last, a.Salary, b.Department
FROM Personnel a, Department b WHERE
a.DeptNO = b.DeptNO AND b.Department =
"Command" ORDER BY a.Salary;
```

The HOW ... Relationships

Query Result ...

```
Last        Salary     Department
=================================
Kirk        50000      Command
            45000      Command
Goner       15000      Command
```

WHERE in the RDBMS? Hybrid

- Comprised of two systems - one to handle the spatial elements, another to manage attribute data
- Most hybrid systems use a proprietary data model.
- Separate storage systems complicate database maintenance, increase disk access and network traffic.
- Requires diligence, attention to detail and special applications to maintain feature-attribute linking.
- What happens when a user splits a line segment?

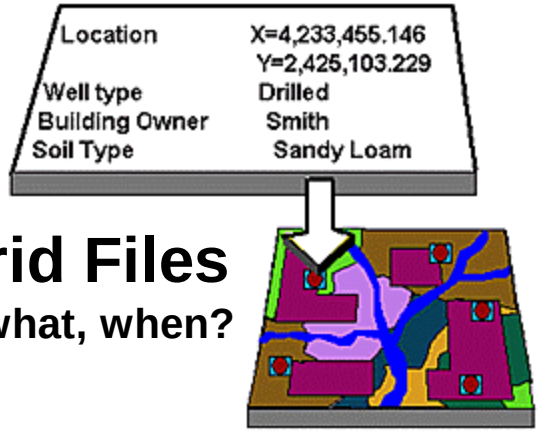
WHERE in the RDBMS? Hybrid

- Where does the original attribute records go?
- How do you maintain a historical record of line splitting?
- How are other GIS layers affected by splitting a pipe? (adding a break in the pipe requires the addition of a node or point feature ...)
- Example of a Hybrid Model? (ARC/INFO, ESRI ShapeFile)
- GIS Database Design

WHERE in the RDBMS? SDE

- Continuous, non-tiled, spatial database for adding spatial data to a relational database management system (RDBMS).
- Database interface that couples spatial data to the RDBMS allowing for high-performance access to all the data in there, spatial and non-spatial.
- No more split system data management - single source editing. Requires special maintenance application to main topology, perform database edits, updates and maintenance (ArcFM)
- Utilize the inherent strengths of commercial RDBMS's ...
- Store the Shape in the same record as the attribute data - BLOBS

WHERE in the RDBMS? SDE



Hybrid Files

Who, what, when?

Shape	BuildID	Owner
	1	Smith
	2	Jones
	3	Clinton

SDE Table

Who, what, when and WHERE!

.shp file

Shape

.dbf file

BuildID	Owner
1	Smith
2	Jones
3	Clinton

	Spatial Server (SDE)	Hybrid Model (Flat File)
User Access	Roles, Users, Built-in Security	No inherent Security, Operating System Security
Access/Data Storage Mechanism	Only accessible by RDBMS software	Operating System files, editable with many applications
Asynchronous Processing	Designed for fast transfer of packets through uniform network protocol	Access both spatial and attribute features within file structure. Many disk head hits.
Multi-User	Row and table level locking. User roles determine editing rights. Trigger and Procedures.	Only one user should edit although many can. No locking or update mechanism.
Open data structure	Use existing database table mechanism.	Proprietary format. ShapeFile is flat file.
Robustness	Roll-backs, Redo Log Files, Back Up tools.	No inherent backup processing. Third party tools. Disk backup.
Data Structuring	Views can be created from existing table as object in database.	A coverage/ShapeFile is a operating system file.
Data Integrity	Enforces referential integrity, date stamping, user access and rights, transactions (rollback)	No enforced integrity checking

Summary

- A Definition and the concept
- WHAT is it?: Software and Process
- WHY?: Advantages and Challenges
- HOW?: Design, Entity-Relationship Model, Modeling, Set Theory, Primary Key, Foreign Key, and Relationships
- WHERE?: Hybrid Systems - ARC/INFO, ShapeFile
- WHERE?: Spatially extending the database - SDE

Thank You!

M.J. HARDEN ASSOCIATES, INC.

PHOTOGRAMMETRY
GIS IMPLEMENTATIONS
COMPUTER TECHNICAL SERVICES

Peter Veenstra
Project Consultant
Government/Utilities Division

1019 Admiral Boulevard
Kansas City, MO 64106

Tel: (816) 889-1153
Fax: (816) 471-1599
E-mail: pveenstra@mjharden.com