Project for the Strengthening of Spatial Data Infrastructures in Member States and Territories of the Association of Caribbean States

Capacity Building Program

Geographic Information Systems

Dr Earl Edwards
Ms. Valrie Grant, MSc.
Topic 5:
Geospatial Data Management
The GIS Process

Geospatial Information processing begins and ends with the real world

Source: Aronoff (1991)
Topic Outline

• Data and Information
• Database Management Systems (DBMS)
• Relational Database Model
• Spatial Data Storage Formats
• Single User vs Multiuser Geodatabases
• Versioning and Replication
• Distributed Geodatabases
Data and Information

• Simply put, data is what goes into the GIS; information is what comes out.

• Information is data, which when processed, would remove the level of uncertainty in decision making.
Database

• An organized collection related data items
• Stored in a highly structured way
• Represents a model of reality
• Data can be stored in a single location or HDD
• Or distributed across large networks
Database Management Systems

• A DBMS is a computer software application that interacts with the user, other applications, and the database itself to capture and analyze data.
• An application that allows users to interact with data.
• A general-purpose DBMS is designed to allow the definition, creation, querying, update, and administration of databases.
Examples of DBMS

- **Server DBMS**
  - Microsoft SQL Server
  - Oracle
  - DB2
  - PostgreSQL
  - MySQL

- **Desktop DBMS**
  - Microsoft Access
## Database Types

<table>
<thead>
<tr>
<th>Type</th>
<th>Typical number of users</th>
<th>Typical architecture</th>
<th>Typical size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal</td>
<td>1</td>
<td>Desktop/Laptop/PDA</td>
<td>MB</td>
</tr>
<tr>
<td>Workgroup</td>
<td>5-25</td>
<td>Client/server: 2 tier</td>
<td>MB-GB</td>
</tr>
<tr>
<td>Department</td>
<td>25-100</td>
<td>Client/server: 3 tier</td>
<td>GB</td>
</tr>
<tr>
<td>Enterprise</td>
<td>&gt;100</td>
<td>Client/server: distributed</td>
<td>GB-TB</td>
</tr>
<tr>
<td>Internet</td>
<td>&gt;1000</td>
<td>Web server &amp; application servers</td>
<td>MB-GB</td>
</tr>
</tbody>
</table>
Database Models

- Flat files  ‘60
- Hierarchical  ‘60
- Network  ‘70
- Relational  ‘80
- Object oriented  ‘90
- Object relational  ‘90
- Web enabled  ‘90
Relational Database Model

- A method of structuring data in the form of sets of records so that relations between different entities and attributes can be used for data access and transformation.

- Proposed by E.F. Codd in 1970 and is based on formal mathematical theory (relational algebra).

- Allows definition of relationships between data tables through common attributes.
Relational Terms and Concepts

Relation Name

Relation Table

STUDENT

<table>
<thead>
<tr>
<th>sid</th>
<th>name</th>
<th>login</th>
<th>age</th>
<th>gpa</th>
</tr>
</thead>
<tbody>
<tr>
<td>53214</td>
<td>Jones</td>
<td><a href="mailto:jones@cs.edu">jones@cs.edu</a></td>
<td>25</td>
<td>3.2</td>
</tr>
<tr>
<td>52748</td>
<td>Smith</td>
<td><a href="mailto:smith@gs.edu">smith@gs.edu</a></td>
<td>32</td>
<td>3.4</td>
</tr>
<tr>
<td>58328</td>
<td>Charles</td>
<td><a href="mailto:charles@ns.edu">charles@ns.edu</a></td>
<td>20</td>
<td>3.8</td>
</tr>
</tbody>
</table>

Attributes

Columns or Fields

Tuples

Rows or Records
### Monitoring Wells

<table>
<thead>
<tr>
<th>Well ID</th>
<th>Date Sampled</th>
<th>Concentration</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-6A</td>
<td>5/8/94</td>
<td>300</td>
</tr>
<tr>
<td>C-8A</td>
<td>5/8/94</td>
<td>20</td>
</tr>
<tr>
<td>C-13A</td>
<td>5/8/94</td>
<td>120</td>
</tr>
<tr>
<td>C-17A</td>
<td>5/8/94</td>
<td>560</td>
</tr>
</tbody>
</table>

### Industries

<table>
<thead>
<tr>
<th>Facility</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acme</td>
<td>3029 Convington Dr.</td>
</tr>
<tr>
<td>Fox</td>
<td>742 West Lake St.</td>
</tr>
<tr>
<td>TPC</td>
<td>90 Aspen Dr.</td>
</tr>
</tbody>
</table>

### Population

<table>
<thead>
<tr>
<th>Family Name</th>
<th>Occupants</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blake</td>
<td>6</td>
<td>79 Circuit St.</td>
</tr>
<tr>
<td>Hernandez</td>
<td>2</td>
<td>148 Plain St.</td>
</tr>
<tr>
<td>Joy</td>
<td>4</td>
<td>18 Webster St.</td>
</tr>
<tr>
<td>Smith</td>
<td>5</td>
<td>4321 Tecumseh Dr.</td>
</tr>
</tbody>
</table>
## Linking Tables Using Common Fields

### One to Many Relationship

![Diagram showing a one to many relationship between two tables](image)

<table>
<thead>
<tr>
<th>Parcel-No.</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-115-001</td>
<td>Brown, D</td>
</tr>
<tr>
<td>11-115-002</td>
<td>Greene, J</td>
</tr>
<tr>
<td>11-115-003</td>
<td>Smith, L</td>
</tr>
<tr>
<td>11-115-004</td>
<td>Hester, D</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parcel-No.</th>
<th>Zoning</th>
<th>Legal-Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-115-001</td>
<td>R1</td>
<td>12,001</td>
</tr>
<tr>
<td>11-115-002</td>
<td>R2</td>
<td>15,775</td>
</tr>
<tr>
<td>11-115-003</td>
<td>COM</td>
<td>19,136</td>
</tr>
<tr>
<td>11-115-004</td>
<td>R1</td>
<td>13,005</td>
</tr>
</tbody>
</table>
Linking Tables Using Common Fields

One to Many Relationship
Spatial Data Storage Formats

- ArcGIS can work with spatial data in multiple formats
Shapefile Vector File Format

- Shapefiles are vector composite files made up of 3-13 separate files.
- In Windows Explorer all shapefile components are shown, in ArcCatalog the entire shapefile is shown as one item.
- All components of a shapefile need to be present together, otherwise the shapefile can be defunct or incomplete.
- The projection file is a useful, but not necessary addition to a shapefile. It can be added or changed accordingly.
- All elements will have the same file name but different extensions e.g. Building_poly.
- Shapefiles can only contain a single shape type!

Possible composite file extensions:
- *.dbf - dBase table (database) file, containing attributes
- *.shp - the file which stores feature geometry (x,y coordinates)
- *.shx - file that stores the index connecting .dbf and .shp files
- *.prj - projection file
- *.shp.xml - metadata file
- *.sbn, *.sbx - spatial index files - sometimes present
- *.ain, *.aih - attribute index files
- *.atx - new, ArcGIS, attribute index file
- *.lxs, *.mxs - geocoding index files
- *.cpg - specifies character set code page
Modelling Reality with Geodatabase

Data Model

Building
Lot Line
Lot
Parcel
Imagery

Geodatabase design

Reality
Design process

Geodatabase
A Geodatabase is a store of geographic data implemented with a relational DBMS.
Multidimensional Data

Data Cube

Time Slices

Time = 1

Time = 2

Time = 3
Multidimensional Data
Multidimensional Data Formats

- NetCDF (network Common Data Form)
- HDF (4.x and previous releases), HDF-EOS, HDF5 (Hierarchical Data Format)
- GRIB, GRIB II (GRIdded Binary)
What is NetCDF?

- NetCDF (network Common Data Form) is a data format designed to support the creation, access, and sharing of array-oriented scientific data.

- It is used extensively in the atmospheric and oceanographic communities to store variables, such as temperature, pressure, wind speed, and wave height.
The storage of netCDF data

Data: Temperature, irradiance, precipitation, humidity, incident solar radiation, vapor pressure, elevation, land area, vegetation, water holding capacity of soil, etc.

Three-dimensional data: Data over an area varying with time
Changing Time Slice

(Selecting a different “dimension value”)
Single User vs. Multiuser Geodatabase
ArcSDE technology client-server model

- All data accessed over TCP/IP network
- ArcSDE translates (acts as a gateway to the DBMS)
  - Spatial and attribute filters limit rows returned
- ArcSDE performs spatial filtering
What is a version?

• A version represents a snapshot in time of the entire geodatabase and contains all the datasets in the geodatabase.
• Versions are not separate copies of the geodatabase.
• Instead, versions and the transactions that take place within them are tracked in system tables.
• This isolates a user's work across multiple edit sessions, allowing users to edit without locking features in the production version or immediately impacting other users and without having to make copies of the data.
• Versions are only supported in ArcSDE geodatabases.
Overview of versioned editing

- Method for presenting and tracking changes to tables
- Multiple, alternate versions may co-exist
  - Appears to users as if they have their own copy of a table
- Includes mechanisms for merging changes
  - ArcGIS offers tools to resolve conflicts
Versioning

- Versioning allows multiple users in a multiuser geodatabase to edit the same data without applying feature locks or duplicating data.
- When you start editing, you are working with your own representation of the version.
- Other users who are connected to the same version cannot see any of your changes until you save the edits.
- When you are ready to apply your edits, you will merge your changes through a process of reconciling edits, resolving conflicts, and posting your changes to the parent version of the geodatabase.
Registering object as versioned

- Enables versioned edits
  - Feature classes, feature datasets, tables

- Must register entire feature dataset
  - Registers all feature classes

- Creates delta tables
  - Adds (A) and Deletes (D)
  - DBMS statistics created
  - DBTUNE controls storage
Move Edits to Base

- This is an option available when registering data as versioned. It allows edits made to the DEFAULT version of the geodatabase to be immediately moved from the delta tables to the base tables.
- Specifying this option when you register the data as versioned can be useful if the modifications you are making will take only a few minutes to complete and if you are connecting to a versioned geodatabase with a third-party application.
- You cannot use the move edits to base option on datasets that contain a topology or network, are archived, or participate in replication.
Editing Versioned Data

The general workflow for editing a versioned ArcSDE geodatabase is as follows:

1. Establish a connection with the geodatabase.
2. Register the data as versioned.
3. Add the data to ArcMap.
4. Start editing and make your edits.
5. Review and reconcile any conflicts between the version being edited and the target version.
6. Post changes to the parent database.
Creating a database connection
Editing Non-Versioned Data

The general workflow for editing non-versioned data is as follows:

1. Make sure the data is not registered as versioned.
2. Use the Editing Options dialog box to configure ArcMap to perform nonversioned editing.
3. Add the data to ArcMap.
4. Start editing and make your edits.
5. Save edits and stop the edit session.
Delta Tables

• The adds and deletes tables for a dataset are collectively referred to as the delta tables because they store changes made to the dataset.

• Registering a feature class as versioned creates an adds and a deletes table. These tables track edits made to the dataset and allow you to edit a dataset without blocking other users from accessing or editing it.

• When you register a dataset as versioned, you can register it as fully versioned (the default option) or with the option to move edits to base.
How versioned edits are stored

### ROADS

<table>
<thead>
<tr>
<th>Shape</th>
<th>ObjectID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>OAK</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>ELM</td>
</tr>
</tbody>
</table>

### A91

<table>
<thead>
<tr>
<th>Shape</th>
<th>ObjectID</th>
<th>Name</th>
<th>StateID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td>PLUM</td>
<td>1</td>
</tr>
<tr>
<td>✔</td>
<td>1</td>
<td>OAK</td>
<td>3</td>
</tr>
</tbody>
</table>

### D91

<table>
<thead>
<tr>
<th>ObjectID</th>
<th>StateID</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
Overview of geodatabase archiving

• Maintain record of edit transactions
• Edits are preserved in a history class
  – Denoted with FROM and TO dates
    • Transaction time is recorded
• Built on versioning architecture
Distributed Data

• Copies of data distributed in multiple locations

• Can provide
  – Improved data availability with poor networks
  – Load balancing: Separate offices can work on same data
  – Field projects
  – Fail over

• Options
  – Copy/Paste: Hard to synchronize edits
  – Geodatabase replication
  – DBMS replication: Limited support for geodatabases
Fundamentals of geodatabase replication

- User defines data to replicate from source geodatabase
- Replica describes data and how to synchronize changes
  - Parent replica
  - Child replica
- Data edited in versioned environment
- Synchronize changes
  - Send to related geodatabase
Replicating data

- Replicate to another geodatabase or handheld device
- Implement as part of workflow
  - Enables local/remote data access for editing, analysis, or mapping

![Diagram of geodatabase connections]

Connected

Multiuser Geodatabase

Handheld device

Personal for Access and file geodatabase

Multiuser geodatabase
Types of replication

• Single generation
  – One check out/check in operation

  Check out/ check in
  Parent ArcSDE
  Child Any geodatabase

• Multigeneration
  – Changes synchronized multiple times

  One Way
  Parent ArcSDE
  Child Any geodatabase

  Two Way
  Parent ArcSDE
  Child ArcSDE
Synchronization

• Connected
  – All replicas accessible on the network
  – Performed in a single process
    ◆ Example: Synchronize wizard in ArcCatalog

• Disconnected
  – Replicas not on the same network
  – Performed by export, file transfer, and import
    ◆ Example: Export changes to a delta XML file and send via snail mail
Geodatabase Replication Use Cases

Regional Offices

Mobile Users

Hierarchical Levels

Production / Publication
Production-Publication Replication
Distributing Data using Replication

Trinidad

Tobago

Geodata service

Local geodatabase

Check out / Check in

2-way

Internet

Local area network (LAN)

File Geodatabase

Personal SDE

File Geodatabase

Personal SDE
Geodatabase Distribution for Land Management
Geodatabase Replication Design
UP NEXT ..... 

Activity: Managing data using Geodatabase