Basics on Geodatabases

GIS Data Management

File and Folder System

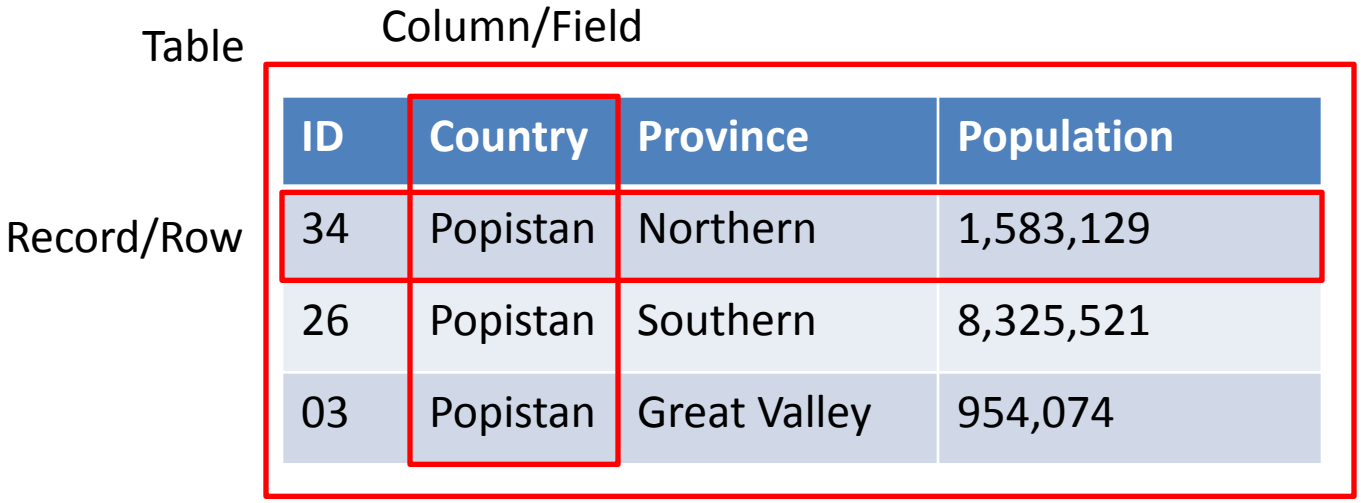
- A **storage system** which uses the **default file and folder structure** found in operating systems.
- Uses the **non-DB formats** we mentioned previously (shapefiles, text/Excel files).
- Data stored on individual computers or shared over a local network.

Database (DB)

- A **storage system** designed to manage large datasets efficiently.
- Users can **query** and **manipulate** data using **joins**, **relates**, and a **Structured Query Language (SQL)**.
- A database can exist on your computer, on a private network (such as your office), or on a server connected to the Internet.

Database Fundamentals

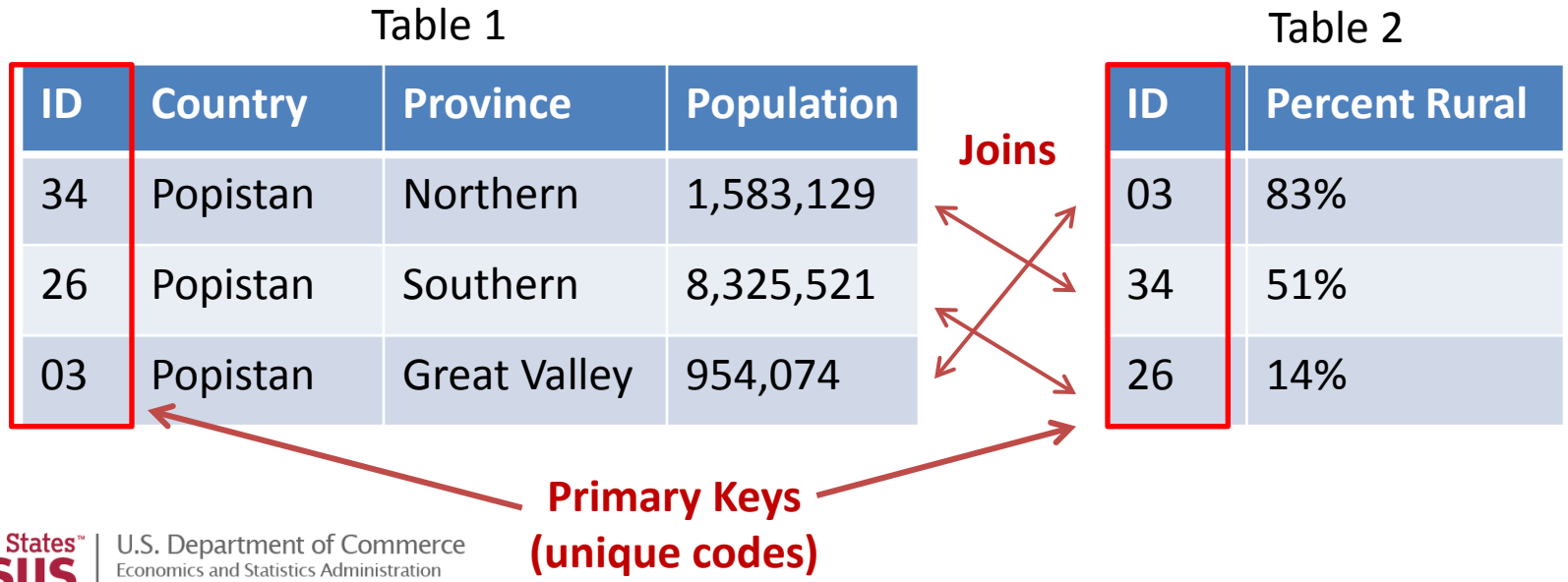
- Databases are a collection of **tables**.
- Each table contains **columns (fields)** and **rows (records)**.



ID	Country	Province	Population
34	Popistan	Northern	1,583,129
26	Popistan	Southern	8,325,521
03	Popistan	Great Valley	954,074

Database Fundamentals

- Tables can be **joined** to each other using a unique identifier or code (a **primary key**).
- It is good practice to assign a primary key to every unit of census geography (including administrative and statistical).
  - Use an **alphanumeric code**.



Database Management System (DBMS)

- Software designed to efficiently **administer** one or more database(s).
- In action, users rarely distinguish between a "DBMS" and "database".
- Examples you may be familiar with include **Microsoft Access** and **ArcGIS**.
  - Note: A Microsoft Excel spreadsheet is not a database, though it does share many features.
- You may also be familiar with more advanced DBMS software, such as **PostgreSQL**, **MySQL**, and **Microsoft SQL Server**.

Single User Databases

- Single user means **only one user at a time** can access the database.
  - There may be minimal support for multiple users.
- Useful for managing data for small projects with few participants.
- These databases are generally stored **locally** (i.e., on your computer) or on a **shared drive** for minimal collaboration.
- Both **Microsoft Access** and **ArcGIS** can create single user databases.

Multi-User Databases

- Nearly always a **DBMS** such as PostgreSQL, Microsoft SQL Server, or MySQL.
- Designed to handle **multiple users** retrieving from and updating the **same database simultaneously**.
- Often just called an **enterprise database**.
  - "Enterprise" refers to an office or organization.

Multi-User Databases

- Managing the edits of multiple users at once is called **versioning** or **deconflicting**.
  - Versioning stores a record of every user's transaction.
  - Thus, each database edit is reversible.
- The databases are **stored remotely** and accessed over a private network or the Internet.
- **Data processing** can either occur on the network or on your local computer.

Diagram of a Multi-User Database

Database Administrator (DBA)

Data Tables

User 1

User 2

User 3

Database Management System (DBMS)

Database 1

Database 2

Database 3

Servers hosting DBMS and DBs

Spatial Databases

- Also called geospatial databases, geographic databases, or geodatabases.
- Possess all of the same features of other databases, plus the ability to **store location data**.

Database Advantages

- The file/folder system is easier for **quick projects**.
- However, storing geographic data in a database provides numerous **advantages** over the file/folder system:
  - Data are stored more efficiently.
  - Can separate geographic features and attribute data.
  - Larger datasets are easier to access and manage.
  - Specific data can be retrieved using queries.
  - The quality of geographic features can be managed more effectively with **topology**.
    - Much more on topology later!

Realities

- Databases require extra **knowledge** of administration, querying, managing joins/relates, etc.
- A multi-user enterprise database, while advantageous, requires **expensive hardware**, persistent network **connectivity**, and **highly skilled IT support**.
- May not be feasible for all organizations.

Group Discussion

- How do you store data in your organization?

- Would you like to use a different method?

Geodatabases in ArcGIS

Geodatabases in ArcGIS

- As we discussed previously:
  - Geographic data can be stored in many different **formats**.
  - Data can also be stored using standalone **files and folders** or a **database**.
- ArcGIS includes functionality to create **geodatabases** for storing, editing, and managing your data.
- These geodatabases include a number of useful tools which we will explore in detail.
  - Note: the terms "**geodatabase**", "**spatial database**", and "**database**" are used interchangeably.

Fundamental Concepts

- The preferred format for storing data in the ArcGIS environment is the **File Geodatabase**.
  - Another format, the **Personal Geodatabase**, is obsolete.
- You can store all types of data in the file geodatabase, including **vector**, **raster**, and **non-spatial tables**.

Important Caveats

- The file geodatabase is designed as a **single user database**.
  - Very limited multi-user support.
- The file geodatabase is a **proprietary format**.
  - Not easily **compatible** with other GIS software.

File Geodatabase vs. Shapefile

Feature	Shapefile	File Geodatabase
File size limits	2GB ✗	Unlimited (TBs) ✓
Storage efficiency	Less efficient ✗	More efficient ✓
Performance	Slower ✗	Faster ✓
Raster support	No ✗	Yes ✓
Enforcing consistency within and between files	No ✗	Yes (topology, schemas, projections) ✓
Compatibility	Most GIS software ✓	ArcGIS Only ✗
Portability	"Zip and ship" ✓	"Zip and ship" ✓
Multi-User	No ✗	Limited ⚠

Structure of File Geodatabases

- Several file types can exist in the file geodatabase:
  - **Feature dataset:** A geospatial "container" which stores projection information and topology for vector data.
  - **Feature class:** Geospatial point, line, or polygon (vector) data.
  - **Non-spatial table:** A set of attributes which are commonly linked to a feature class.
  - **Raster:** Sits independently within a file geodatabase and cannot be stored in a feature dataset.
  - **Topology file:** Stores the rules which enforce data integrity within the database.
  - **Relationship file:** Creates a join between multiple feature classes and/or non-spatial tables.
  - **Others:** Raster mosaic/catalog, schematic dataset, toolbox, parcel fabric, annotation, network, terrain.
- We will work with feature datasets, feature classes, and tables.

Good Practices: In General

- Keep spatial data and non-critical attributes **separate**.
  - Exception: attributes critical to the geographic definition of the features in the dataset (e.g., place names, identification codes).
- Use feature datasets to store related geography.
  - Can be structured many different ways.
  - E.g., province-by-province, grouped by type of feature.

Good Practices: Data Preservation

- Maintain **separate** databases for **production** files and **working** files.
  - E.g., one database of data approved for field staff to use and another for data being edited by head office staff.
  - Data from the working DB feeds into the production DB.
  - If you want to experiment, **export** your feature data to a **scratch database** or a **shapefile**.
  - **Never edit production data directly!**
- Produce daily or weekly **backups** of your database and store in a safe place.
  - E.g., a removable hard drive locked in a storage room.

Other Options for geo-enabled RDBMs

Options have increased dramatically, here are some of the major players:

Proprietary

- Oracle Spatial
- Microsoft SQL (post 2008)

Open Source

- PostGIS/Postgre
- SpatialLite