

# Spatial data models (types)

Not taught yet

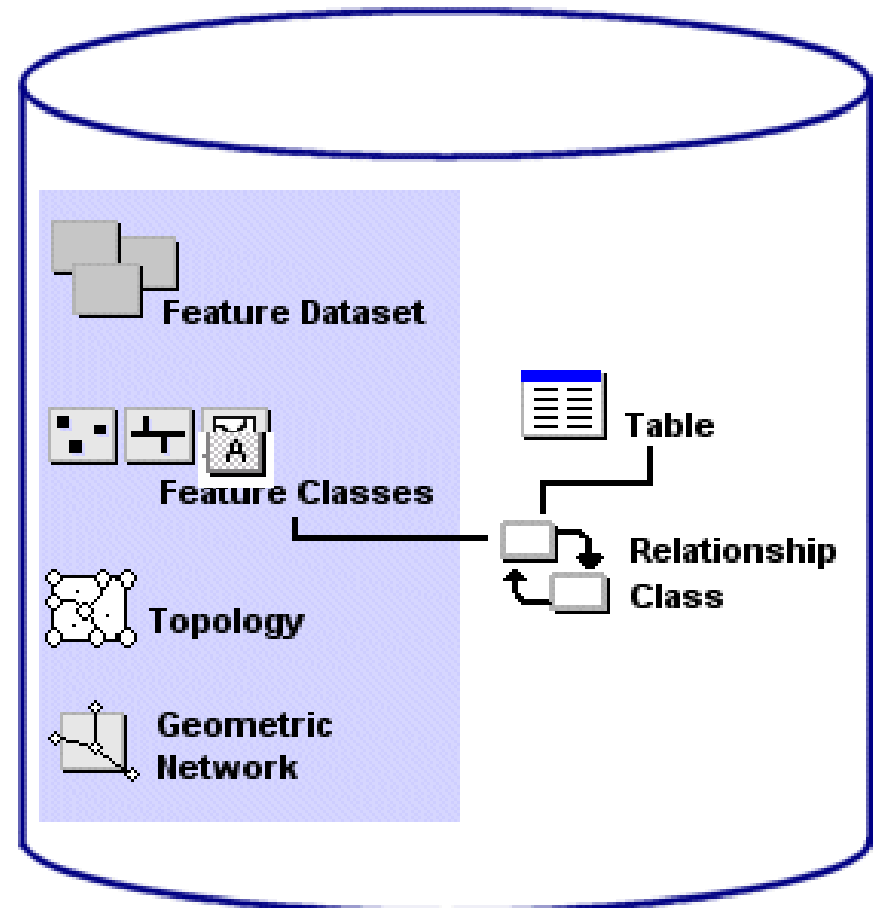


# A new data model in ArcGIS

- Geodatabase data model
  - Use a relational database that stores geographic data
    - A type of database in which the data is organized across several tables. Tables are associated with each other through common fields. Data items can be recombined from different files.
  - A container for storing spatial and attribute data and the relationships that exist among them
  - And their associated attributes can be structured to work together as an integrated system using rules, relationships, and topological associations

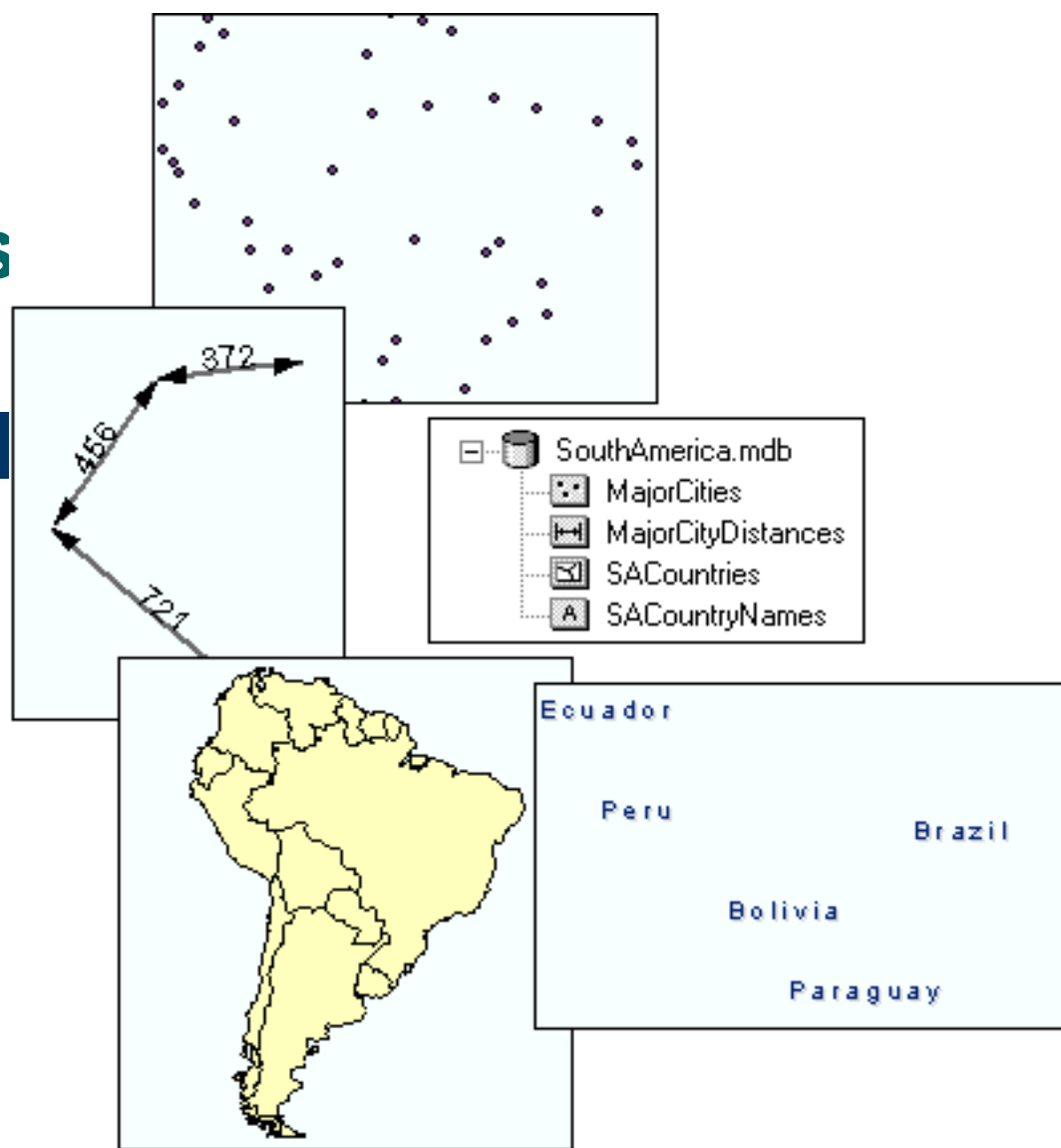
# Geodatabase components- vector data and table

- **Primary (basic) components**
  - feature classes,
  - feature datasets,
  - nonspatial tables.
- **complex components building on the basic components:**
  - topology,
  - relationship classes,
  - geometric networks



# Feature classes

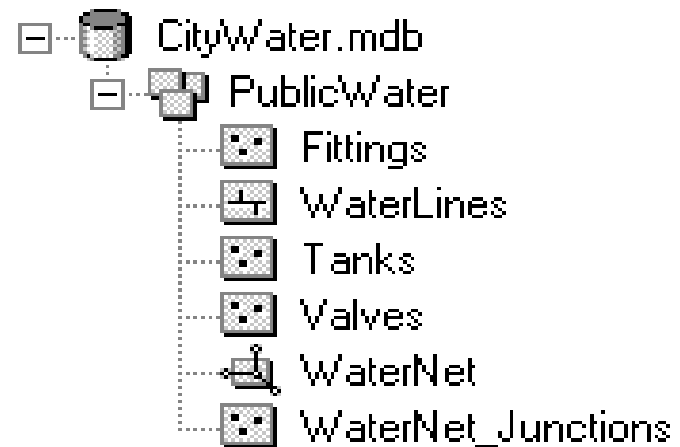
- A feature class is a geographic feature include points, lines, polygons, and annotation feature class.
- Feature classes may exist independently in a geodatabase as stand-alone feature classes or you can group them into feature datasets



**The SouthAmerica geodatabase contains four stand-alone feature classes:**  
a point feature class of cities,  
a dimension feature class of distances between cities,  
a polygon feature class of countries, and  
an annotation feature class of country names

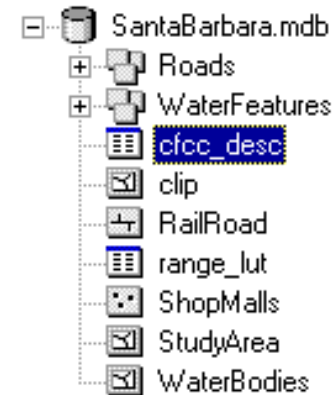
# Feature datasets

- A feature dataset is composed of feature classes that have been grouped together so they can participate in topological relationships with each other. All the feature classes in a feature dataset must share the same spatial reference (or coordinate system)
- Edits you make to one feature class may result in edits being made automatically to some or all of the other feature classes in the feature dataset




In the CityWater geodatabase, three point feature classes and one line feature class were grouped into the PublicWater feature dataset to create a geometric network called WaterNet.

# Tables



OBJECTID	CFCC	CFCC_DESC	CFCC_ID
1	A00	Road, classification unknown or not elsewhere	1
2	A01	Road, undivided	2
3	A02	Road, undivided, in tunnel	3
4	A03	Road, undivided, underpassing	4
5	A04	Road, undivided, rail line in center	5
6	A05	Road, divided	6
7	A06	Road, divided, in tunnel	7
8	A07	Road, divided, underpassing	8
9	A08	Road, divided, rail line in center	9
10	A10	Primary road, interstate highway and limited ac	10

- Feature class tables and nonspatial attribute tables.
- Both types of tables are created and managed in ArcCatalog and edited in ArcMap. Both display in the traditional row-and-column format. The difference is that feature class tables have one or more columns that store feature geometry.
- Nonspatial tables contain only attribute data (no feature geometry) and display in ArcCatalog with the table icon . They can exist in a geodatabase as stand-alone tables, or they can be related to other tables or feature classes.

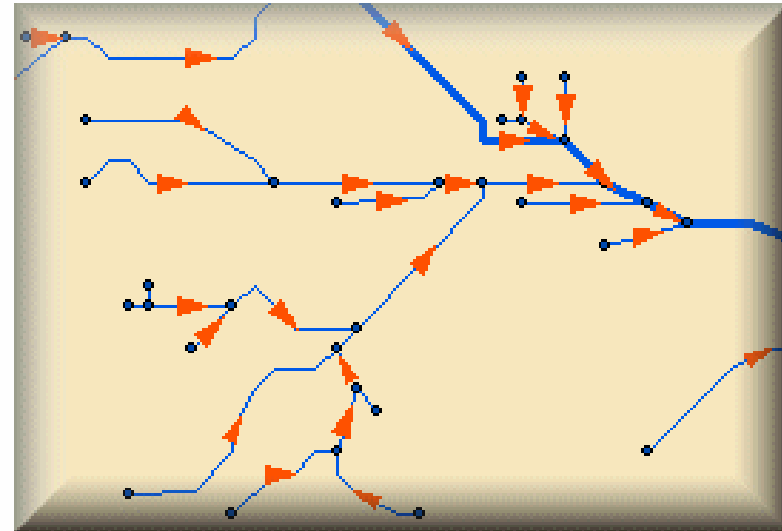
The cfcc\_desc table in the SantaBarbara geodatabase contains attribute data for the Roads feature class (stored inside the Roads feature dataset).

# Topology

- In a GIS, spatial relationships among feature classes in a feature dataset are defined by topology. You can choose whether or not to create topology for features.
- The primary spatial relationships that you can model using topology are adjacency, coincidence, and connectivity
- There are three types of topology available in the geodatabase: [geodatabase topology](#) (over 20 topology rules), [map topology](#), and [geometric network topology](#). Each type of topology is created from feature classes that are stored within a feature dataset. A feature class can participate in only one topology at a time
  - Geodatabase topology defines the spatial relationships of your data. All feature classes must be in the same feature dataset
  - Map topology creates temporary topological relationships among features in one or more feature classes during an edit session
  - Geometric network topology established between point and line feature classes stored in the same feature dataset

# Geometric Networks

- In the real world, examples of networks abound: streams joining together to form larger streams, pipes carrying water to homes and businesses throughout a city, and power lines carrying electricity.
- In a geodatabase, you can model each of these real-world networks with a geometric network. Starting with simple point and line feature classes, you use ArcCatalog to create a geometric network that will enable you to answer questions such as: Which streams will be affected by a proposed dam? Which areas will be affected by a water main repair? What is the quickest route between two points in the network?



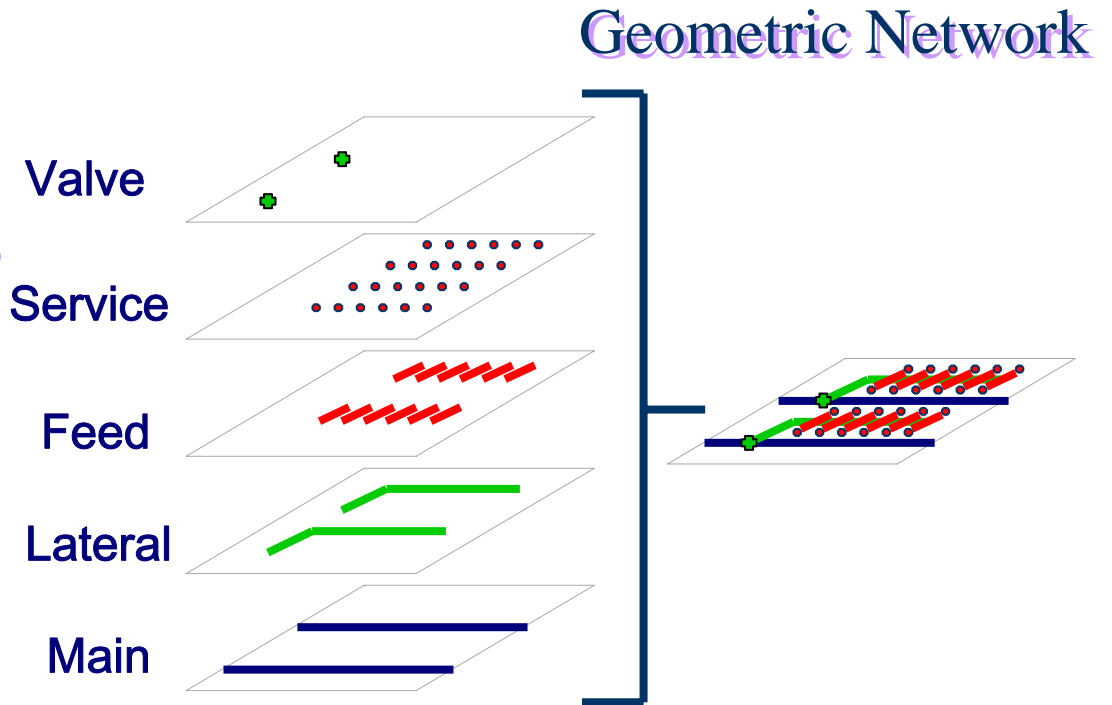
In a geodatabase, you can model the flow of resources using a geometric network. A geometric network combines line and point feature classes into a single database entity and maintains the explicit topological relationships between features.

# Geometric Networks cont.

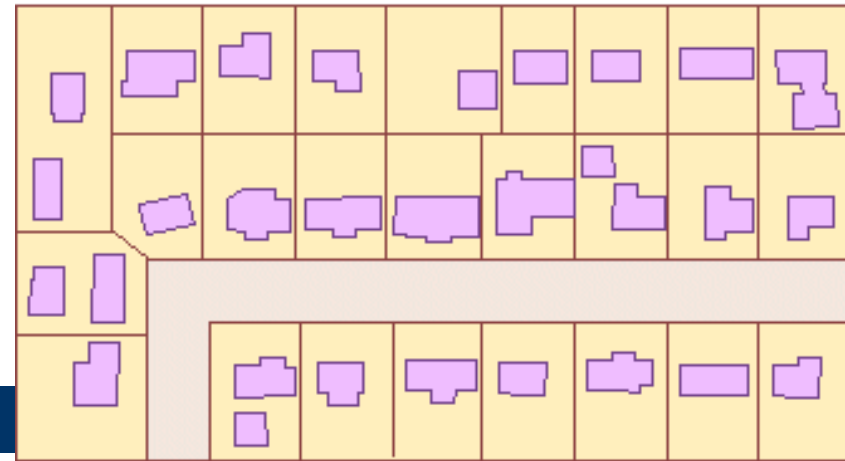
- Feature classes that participate in the network are automatically converted from simple feature classes to network feature classes, and one or more attribute fields containing network information are added to the feature class table.
- There are more restrictions involved with managing network feature classes than with managing simple feature classes. You cannot rename, delete, or copy a network feature class. To perform any of these actions, you must convert the network feature class back to a simple feature class by deleting the geometric network.
- When you build a geometric network, there are a number of options you can choose from to make your network model more realistic. For example, you can:
  - set the direction that resources will flow through the network
  - assign weights that control the speed of flow through different parts of the network
  - specify rules that control how each element in the network connects to the others

# Geometric Networks example

## Feature Classes



# Relationship classes



- In a geodatabase, relationship classes provide a way to model real-world relationships that exist between objects such as parcels and buildings or streams and water sample data. By using relationship classes, you can make your GIS database more accurately reflect the real world and facilitate data maintenance.

OBJECTID	NAME	TEMP	PH	DO
1	Jones Creek	22	7	4
2	Fern Creek	20	7	7
3	Running Spring	20	7	2
4	Longtown Creek	20	7	7
5	Arroyo Canal	21	7	6
6	Tornado Creek	9	8	6
7	Allen Creek	10	7	8

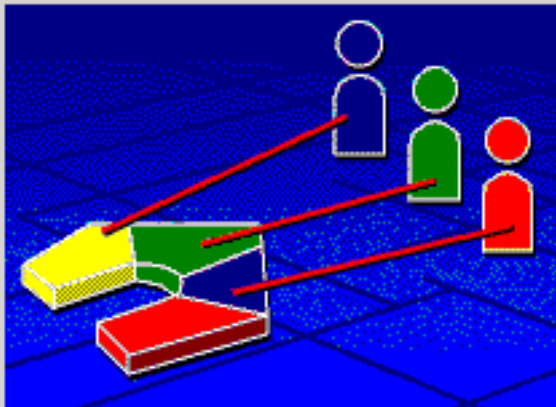
The relationships stored in a relationship class can be between two feature classes (such as buildings and parcels, [top](#)) or between a feature class and a nonspatial attribute table (such as streams and water quality sampling data, [bottom](#)).

# Three types of relationship

In a 1-1 (one to one) relationship, each object of the origin table/feature class can be related to zero or one object of the destination table/feature class.

Parcels  
Table/Feature  
Class

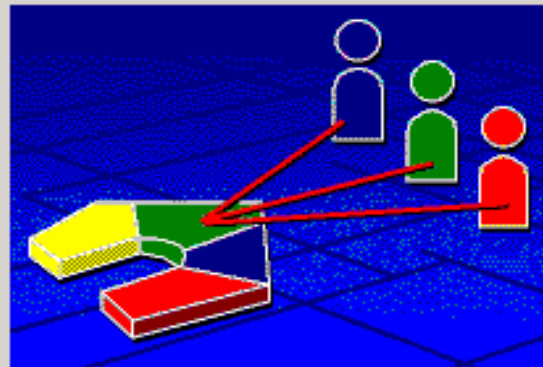
Owners  
Table/Feature  
Class



In a 1-M (one to many) relationship, each object in the origin table/feature class can be related to multiple objects in the destination table/feature class.

Parcels  
Table/Feature  
Class

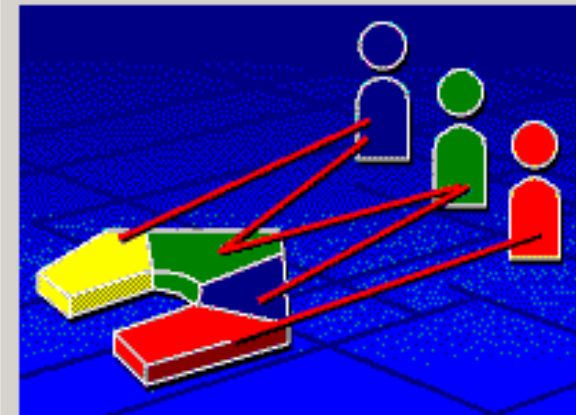
Owners  
Table/Feature  
Class



In a M-N (many to many) relationship, multiple objects of the origin table/feature class can be related to multiple objects of the destination table/feature class.

Parcels  
Table/Feature  
Class

Owners  
Table/Feature  
Class



# Geodatabase components-

## Raster data

- Raster data referenced only in personal geodatabase
- Raster data physically stored in multiuser geodatabase
- Raster datasets and raster catalogs
  - A raster dataset is created from one or more individual rasters. When creating a raster dataset from multiple rasters, the data is mosaicked, or aggregated, into a single, seamless dataset in which areas of overlap have been removed. The input rasters must be contiguous (adjacent) and have the same properties, including the same coordinate system, cell size, and data format. For each raster dataset (.img, grid, JPEG, MrSID, TIFF), ArcGIS creates an ERDAS IMAGINE file (.img).
  - A raster catalog is defined as a table in the geodatabase which you can view like any other table in ArcCatalog. Each raster in the catalog is represented by a row in the table. It contains a collection of rasters that can be noncontiguous, stored in different formats, and have other different properties. In order to view all the rasters in the catalog, they must have the same coordinate system and a common geographic extent

**Table 2. Comparison of raster datasets and raster catalogs**

<b>Raster Dataset</b>	<b>Raster Catalog</b>
Single dataset built from one or more rasters	Collection of rasters
Homogeneous source data: same resolution, format, coordinate system	Heterogeneous source data: different resolutions, formats, data types, file sizes
Faster display	Slower display. The more rasters, the slower the display
Loss of overlapping pixels when mosaicked	Overlapping pixels are preserved
One metadata file	Metadata files for the catalog and for each raster

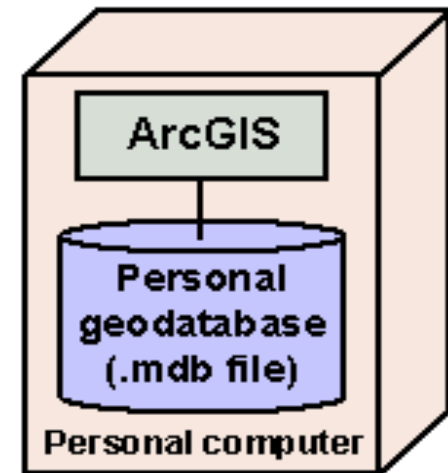
# Types of geodatabases

---

- personal
- enterprise

# Personal Geodatabase

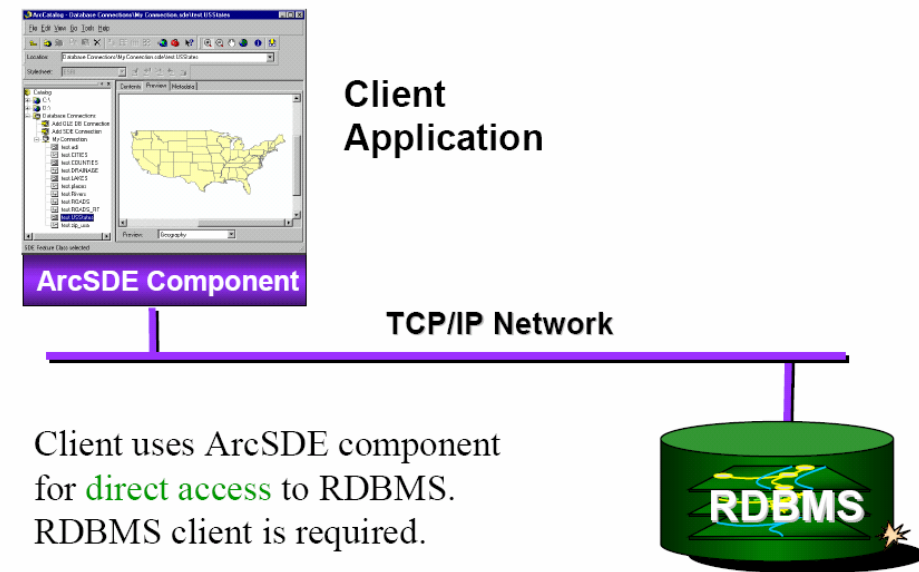
- The personal geodatabase is given a name of *filename.mdb* that is browsable and editable by the ArcGIS, and it can also be opened with Microsoft Access. It can be read by multiple people at the same time, but edited by only one person at a time. maximum size is 2 GB. no support of raster



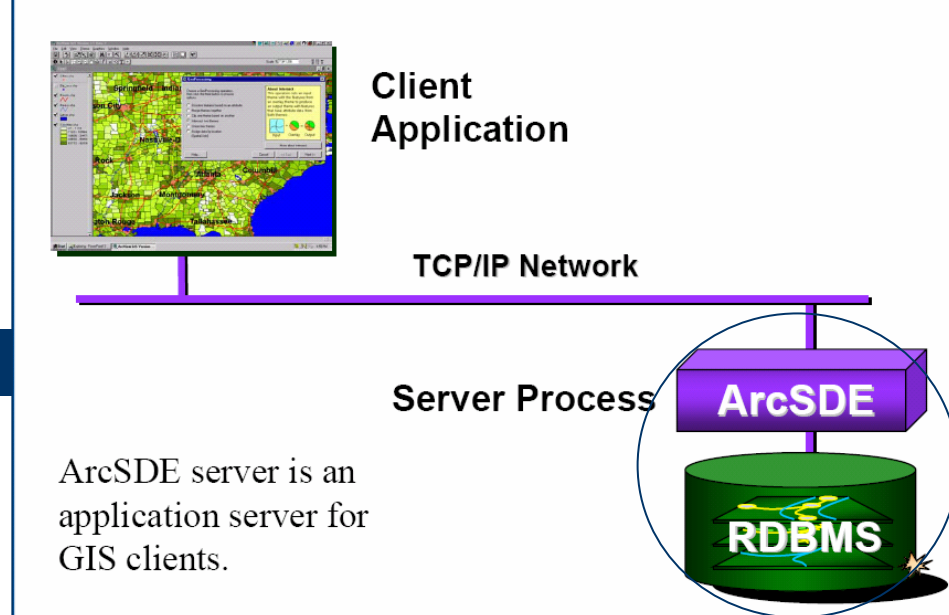
# Multiuser Geodatabase

- Multiuser (ArcSDE or enterprise) geodatabase are stored in IBM DB2, Informix, Oracle, or Microsoft SQL Server.
- It can be edited through ArcSDE by many users at the same time, is suitable for large workgroups and enterprise GIS implementations. no limit of size. support raster data.

# ArcSDE 2-tier Client/Server Architecture



# ArcSDE 3-tier Client/Server Architecture



Twentieth Annual ESRI International User Conference • June 26-30, 2000

Twentieth Annual ESRI International User Conference • June 26-30, 2000

3-tier ArcSDE client/server architecture with both the ArcSDE and Oracle RDBMS running on the same server, which minimizes network traffic and client load while increasing the server load compared to 2-tier system, in which the clients directly connect to the RDBMS

# Personal and Multiuser Geodatabase Comparison

If a project becomes too large for a personal geodatabase, you can easily move the data stored in a personal geodatabase to a multiuser geodatabase using ArcCatalog.

**Table 1. Comparison of multiuser and personal geodatabases**

	<b>DBMS</b>	<b>Client / Server</b>	<b>Long Transactions*</b>	<b>Disconnected Editing</b>	<b>Editors</b>	<b>Raster Data</b>	<b>Size</b>
<b>Multiuser</b>	IBM DB2, Informix, Microsoft SQL Server, Oracle	Yes	Yes	Yes	1 or more at the same time	Stored	Unlimited
<b>Personal</b>	Microsoft Access	No	No	No	1 at a time	Referenced	Up to 2 GB

\*Database transactions spanning multiple edit sessions

# Differences in Geodatabase among ArcGIS family

	ArcView/Map	ArcEditor	ArcInfo
<b>Read, Display, Query, &amp; Use</b>	Any geodatabase	Any geodatabase	Any geodatabase
<b>Editing</b>	<p>Edit simple personal geodatabases<sup>1</sup></p> <p>Can edit simple feature classes, annotation, tables, and simple feature datasets</p>	<p>Edit all geodatabases</p> <p>Multiuser geodatabase editing requires ArcSDE™</p>	<p>Edit all geodatabases</p> <p>Multiuser geodatabase editing requires ArcSDE</p>
<b>Schema management and database design</b>	<p>Create simple schemas for personal geodatabases<sup>1</sup></p> <p>Supports: points, lines, polygons, annotation, and tables</p> <p>No topology No geometric networks No relationship classes No feature-linked annotation No dimension classes No raster support No custom feature classes</p>	<p>Create all schemas for any geodatabase</p> <p>Multiuser geodatabase support requires ArcSDE</p> <p><a href="http://ag.arizona.edu/art/kb/geodb/geo_info.html">http://ag.arizona.edu/art/kb/geodb/geo_info.html</a></p>	<p>Create all schemas for any geodatabase</p> <p>Multiuser geodatabase support requires ArcSDE</p>

<sup>1</sup> Simple personal geodatabases model spatial features that are very much like ESRI® shapefiles: points, lines, and polygons without topology. Personal geodatabases also allow you to store multiple feature classes together in a single feature dataset, as well as provide annotation support.

# Problems caused by the simplified features may still exist, but let's live on it

- Dynamic nature (not static)
  - Forest grow
  - River channel change
  - City expand or decline
- Identification of discrete and continuous features
  - Road to be a line or a area?
- Scale
- Some may not fit to any type of features: fuzzy boundaries
  - Transition area between woodland and grassland

Lets do not worry about these problems now!!! Just keep in mind