SQL and SSQl
and GIS Data Architecture

IT4GIS
Keith T. Weber, GISP
GIS Director
ISU-GIS Training and Research Center

Today's Road Map

• We will be making some connections and tying up some threads...
• To accomplish this, the exercise will ask you first to revisit some GIS fundamentals
  – Data Structure (vector and raster)
  – Objects in a Geodatabase
  – Topology
• And Finally, spatial SQL
• Our presentation/discussion focuses on SSQl

Definitions to get started

• SQL = Structured Query Language
• SSQl = Spatial SQL
• GPL = Graphical Presentation Language
SQL - A Review

• SQL is a simple language used to query (question) an ODBC-compliant database and retrieve data.
  – SQL is not simple or standard
  – S = structured

Structure

• The most basic SQL statement is:
  – SELECT * from database.table
• Let’s dissect this statement
  – SELECT is the command
  – * is a wildcard = i.e., everything and anything
  – Database.table is the target of the query

A Little More…

• The previous SQL statement selected everything from a table
• But, how do we select only a portion of a table?
  – The WHERE CLAUSE
WHERE CLAUSE

• WHERE conditional operator
• For example:
  – SELECT * from database.table WHERE
    CITY_NAME = 'Pocatello'

Types of Conditional Ops

• Simple (as in the previous example)
• Compound
  – Let’s say we want to select and work with all records describing
    Pocatello and Blackfoot
  – We could select and work with them individually using two
    discrete Simple statements or use Conditional operators

Combining Statements using Conditional Operator Expressions

• Instead of:
  – SELECT * from database.table WHERE
    CITY_NAME = 'Pocatello'
  – …do some work, and then
  – SELECT * from database.table WHERE
    CITY_NAME = 'Blackfoot'
  – …do some more work
We Can Use…

• A **Compound** expression combining two or more single expressions using either:
  – AND
  – OR
• In our example, which shall we use?

**OR**

• SELECT * from database.table WHERE
  CITY_NAME = ‘Pocatello’
  OR
  CITY_NAME = ‘Blackfoot’

Why **OR**?

• Before a record (entity) is returned as a result of a query, the record must satisfy EACH WHERE clause if **AND** is used.
• When **OR** is used, a record must satisfy only one of the **WHERE** clauses.
This is SQL

• What is SQL?
  – Spatial Structured Query Language
  – Or SQL for Spatially-enabled relational databases (i.e., object-relational databases)
    • Informix
    • Oracle
    • IBM DB2
    • MS SQL Server
    • PostGreSQL

An Example

• SELECT residence.geometry
  FROM residence
  WHERE Type = 'single family'

What is different about this expression?
residence.geometry

Why is *.geometry important?

• Until now, we have been returning all fields
  – (SELECT * FROM…)
• *.geometry returns the geographic feature(s) as objects
• SSQl is used to select the geometry (.geometry) of the TABLE of interest (residence) from a spatially-enabled object-relational database
GIS Layers are Tables?

Data type for Geometry

Spatial Grid Extent

Object Relational

- OBJECTID inherited from Object class
- SHAPE inherited from a class called Feature

This could be “Boundary”

Geometry Data Type

- We have talked a lot about the data types used to store traditional attributes (e.g., long integer, text, etc.)
- Recall, an ORDBMS can store OBJECTS natively
- What data type is used to store OBJECTS?
Key Concepts

- SQL is highly structured
- **Spatial SQL** builds upon SQL but remains within the same general framework
- SSQL requires an object relational, spatially-enabled database
- The *.geometry table is queried to return features...
  - Objects are stored in the table as LOB data
  - Along with other attributes

Questions?