Geog 461 Learning Objective Outline
Part 2 GIS-based Modeling for Decision Support

LOO 07 Data, Data Models, and Database Models
07.1 How can we differentiate data, information, evidence, and knowledge?
Nyerges and Jankowski *GISDS* Chapter 3 Section 3.1 Data, Information, Evidence, and Knowledge – A Comparison

Data – is/are raw observations (as in a measurement) of "some reality", whether past, current, future, in the context of some shared understanding of an "organizational context". Often times we value what we measure and we measure what we value – that is, what is important enough on which to spend human resources to get data.

Information – is/are data placed in a context for use "tells us something" about a world we share. Geographic information is a fundamental basis of decision making, hence information needs to be transparent in groups if people are to share an understanding about a situation.

Evidence – is/are information we use to make reasoned thought (argument) about the world. All professionals, whether they be doctors, lawyers, scientists, GIS analysts, etc. use evidence as a matter of routine in their professions to establish the "shared valid information" in a community. Credible (as in the source is believable) information is the basis of evidence. How we interpret evidence influences how we gain knowledge.

Knowledge – is/are evidence as credible information that has withstood a "long lasting effect" that helps us interpret the world through new information, and of course, data. Knowledge about circumstances is what we use to interpret information, and decide if we have gained new insight or not. It is what we use to tell whether information is useful of not. When we integrate information into our world circumstances we create knowledge.

07.2 What are the three components of every conceptual, logical and physical data model?
Nyerges and Jankowski *GISDS* Chapter 3 Section 3.2 Data Models – the Core of GIS Data Management

Database design process - several levels of database descriptions, some oriented for human communication, while others oriented to computer-based computation. In the database modeling literature, "conceptual, logical, physical" are used to differentiate levels of data modeling abstraction.

- **Data model** - language that can used to describe entity (feature object) classes, plus operations and constraints on those entity classes

- **Database model** - use of that language to specify a specific set of entity (feature object) classes
  See table 3.1 for distinguishing characteristics

No other terms have been proposed in literature to clarify the important nuance. One important thing to remember is that the database model is still an object class expression – it is not the database. The database model makes use of a particular schema language to specify certain object classes that will be used in the creation of a database.

A **conceptual data model** organizes and communicates the meaning of data categories in terms of object/entity classes, attributes and (potential) relationships. This interpretation of the term data model is often credited to James Martin, a world-reknowned information systems consultant, having authored some 25 books as of the mid 1980's.

Conceptual data model language – ER Model and its use See Figure 3.1 ER language
See Figure 3.2 Geospatial Data Constructs that often appear in GIS data models – Conceptual data model constructs

07.3 What are the constructs and processes of the logical data model?
Nyerges and Jankowski *GISDS* Chapter 3 Section 3.2.2 Logical Data Models

A **logical data model** expresses a conceptual data model in terms of computable:
a) data constructs (i.e., entity classes or object classes) for data structures
b) operations (to create relationships), and
c) validity constraints.
This interpretation of the term "data model" is often credited to Edgar Codd, the person who invented the Relational Data Model as the basis of relational database management systems.

A **logical data model is a "formal design" for a data management system** to be implemented as a software system. Hence, the data construct component of the relational data model is the table. The operations component is the relational calculus (later simplified to the relational algebra). The validity constraints component tests for data entry errors in order to keep database "clean"

Geospatial data constructs – the building blocks of geospatial data models (geometry and topology) stored as data structures in software

Topology – study of connectedness, adjacency, and containment among objects embedded in a surface ("topo" actually means surface);

Constructs in Five ESRI Logical Data Models
See Table 3.2 Spatial Data Construct Types Associated with Data Models

Relationships Underlying the Operations of Five ESRI Logical Data Models
See Table 3.3 Spatial, logical, and temporal relationships underlie operation opportunity.

07.4 What are major advantages of a geodatabase data model?

Object-based data models and network structuring - Water, sewer, gas, electric

Geodatabase data model – ESRI's newest data model
- spatial data and attribute data - same "level of precedence" (either stored followed by the other)
    - coverage data model, geometry had to be stored first, then attribute data to follow
- temporal data still stored as an attribute; does not have its own "special domain" of operations

Advantages of geodatabase data model
- Built-in behavior – feature ways of acting (implemented through rules) can be stored with data

- Geodatabase manager – management performed by a single database manager (object relational); Previously, spatial data managed by file manager and attribute data managed by relational data manager
- Large geodatabases – do not need to be tiled (squares of physical space) using file manager
- Customized features are possible – transformers, parcels, pipes (not geometry defined, but attribute defined)

Feature classes for geodatabase model
- feature class is stored as a table; generic feature classes and custom feature classes possible

generic feature classes (feature specification in general) are the following and defined as:

point – single point represented by ID

multipoint – multiple points cluster represented by ID

network junction features:

simple junction feature – like a node storing topology, but can have logical behavior

e.g., valve connecting pipe of same or different diameter

complex junction feature – can contain internal parts, like a transformer or junction box

e.g., switch in electrical network with multiple wires

line:

line segments – straight line from point to point

circular arcs – parameterized by a radius (pixel subpoints needed for shape)

Bezier spines – multiple arcs to fit a series of points

network edge features:

simple edge feature – lines play a topological role (no interior junctions)

can have connectivity rules

complex edge feature – can support one or more junctions along the edge

multiple simple edges all in same feature

07.5 What are the constructs and processes of the physical data model?
Nyerges and Jankowski *GISDS* Chapter 3 Section 3.2.3 Physical Data Models

Constructs, operations, and rules - implemented in the physical data model

Table 3.4 Data Types – detail of the geospatial constructs

Numeric

Integer – positive or negative whole number, usually 32 bits

Long Integer – positive or negative whole number, usually 64 bits

Real (floating point) – single precision decimal number

Double (floating point) – double precision decision number

Character (text string) – alpha-numeric characters

Binary – numbers stored as 0 or 1 expression

Blob/Image – scanned raster data of usually very large size

Geometry shape (Figure 3.2 are all shapes)

Spatial data indexing of fields for performance

R-trees (short for region trees) or

Quad trees (that subdivide space into quadrants)