

---

# Creating a GIS Data Schema for Information Resource Management

D. Scott Denkers

---

Scott Denkers received B.S. degrees in geography and applied biology from Utah State University. He later received his M.S. in geography with emphasis in digital cartography and remote sensing. After working with the Arizona State Land Department and University of Vermont's GIS programs, he moved to Washington state where he is currently data administrator for the Washington State Department of Natural Resources. He is interested in digital information theory and its application to GIS.

**Abstract:** *Washington State Department of Natural Resources' GIS was developed to help manage a wide array of land-based resources and support activities. Other traditional information system environments exist concurrently within the department. The GIS and other systems often share similar data requirements. A methodology is developed to logically model the agency's data resource without the imposition of artificial administrative or application boundaries. Entity-relationship formalism is used as the construct for this schema. This paper describes the "reverse-engineered" application of this methodology to GIS attribute data sets, and the advantages of doing so. Specific benefits derived from using a logical data schema in database development include: 1) better physical design; 2) establishment of common terminology throughout the department; and 3) increased data transfer opportunities.*

The geographic information system (GIS) developed at the Washington State Department of Natural Resources has expanded considerably since its installation in the early 1980s. Its data component, measured by volume and complexity, has also increased dramatically. In January 1987, 150 data items were supported by GIS staff in Olympia. Currently there are approximately 400. Within the next few years this figure is expected to double with the development of several large applications. Ties with existing, non-GIS, multi-user computer systems are also being established to help provide answers to increasingly complex resource management issues.

Increasing data complexity and issues of system-to-system

data sharing calls for effective data documentation that can readily be used by a wide range of users from application development staff to casual non-departmental personnel. With the help of our GIS software vendor, a menu-driven data dictionary system was set up during initial GIS development. Information about data (metadata) has subsequently been entered into the dictionary. Metadata are now retrieved through three physical data structure themes: data layer, data file and data item. The data dictionary has proved to be an invaluable information resource tool for users wishing to access metadata based on current system architecture.

Concurrently with early GIS development, the department

formally recognized data as an actual resource through policy and guidelines. In effect, this meant that data was to be managed as any other more tangible resource such as timber and real estate. Guidelines also stated that new system development would be driven by data, as an agency resource, rather than on an application by application basis.

This strategic direction was accompanied by a relatively new concept being explored within Washington state government at that time; that of modeling data based on business subject (data entity) and associations, rather than on application-driven file structures. The entity-relationship (E-R) model, put forth by Chen as "a unified view of data" (1976, p. 10), has

been the foundation for this approach. The methodology has been used successfully in many traditional (non-spatial) information processing settings (for example, see Margaronis 1990 and Plotkin 1991). Nyerges (1980, 1989), Calkins and Marble (1987), and Bedard and Paquette (1989), among others, have also referred to the applicability of the E-R modeling approach in geographic databases. Starting with contents of the established data dictionary, the new approach has been tested against the department's GIS attribute data environment in a "reverse-engineering" fashion. A Computer-Aided Systems Engineering (CASE)<sup>1</sup> tool was found to be very useful in creating the logical data model and accompanying documentation.

Benefits resulting from the use of a documented logical data schema include better database design, establishment of common data terminology used throughout the whole organization, and increased data sharing capability. This paper describes the steps involved in developing a logical data schema and the benefits of its application. The relationship between the GIS data schema and the agency-wide data schema will also be discussed. Definitions of key concepts used in the paper are provided in Table 1.

## Defining a Logical Data Schema

This section describes how a logical data schema was derived from the existing GIS operational environment. A series of

TABLE 1.  
Definition of data modeling terminology.

DATA ENTITY	Person, place, thing, concept or event about which there is enough interest in the organization to collect and store data. Often referred to as data subject or business subject. Example: An organization may collect varied information about <i>watercourses</i> (length, depth, flow rate, etc.). "Watercourse" is the data entity.
RELATIONSHIP	A business connection or logical association that exists between data entities. Example: <i>watercourses flow across</i> geologic formations. "Flow across" is the relationship between the data entities, "watercourse" and "geologic formation."
CARDINALITY	The upper and lower bounds on the number of possible instances of one entity in relationship to another entity. Example: <i>one</i> watercourse flows across <i>one to many</i> geologic formations.
LOGICAL DATA SCHEMA	Data conceptually organized by data entity with the imposition of no application or physical system constraints.
PHYSICAL DATA SCHEMA	Data structures as carried in the database.
METADATA	Information about data.

steps describe how data entities, and then data relationships, were discovered and defined. This "reverse-engineering" or "bottom-up" approach began by analyzing the data item component of the database.

### Establish Tentative Entity List

Entity types are implicit within any information representation. In discovering the department's GIS data entities, the inventory of supported data items was examined, one item definition at a time. A sample output from the GIS data dictionary showing data item names, their definitions, and files of occurrence is provided in Table 2.

Even though file (and spatial layer) names often provided clues as to which entities were

described in the structures, they were ignored to ensure objective entity analysis based on actual information content of data items. The key question asked about each data item was, "What is this data describing?" The first data item in Table 2, STAND.NO, describes an entity called (FOREST) STAND. It is the unique identifier for each instance of a STAND. The second data item, LAND.USE, at first appearance seems to describe STAND. However, land use does not solely describe forest stands. Land use may describe any type of resource unit, whether it is forested, agricultural, urban or water. In this case, defining an entity called RESOURCE UNIT would seem appropriate. In Table 2, another

TABLE 2.  
GIS data dictionary sample output.

FILE NAME	ITEM NAME	ITEM DESCRIPTION
LULC.COM	STAND.NO	Unique statewide stand number
	LAND.USE	Code describing activity on DNR-managed land
	LAND.COV	Natural or artificial land surface cover code
	INSPEC.DATE	Date ALL detailed forest inventory was updated
	EXAM.TYPE	The method of inventory data gathering
	PRI.SPEC	Primary tree or agricultural species code
	SEC.SPEC	Secondary tree or agricultural species code
	TER.SPEC	Tertiary tree or agricultural species code
	TREAT.DECADE	Future decade of stand conversion or treat.
	DAMAGE	Type of damage seen in a forest stand
	INTENSITY	Level of forest stand damage
	INPUT.FORM	Form type used to record stand mgmt. activities
	DATE	See detailed item description
	TWP.CODE	Township code
LULC.EVEN	STAND.NO	Unique statewide stand number
	PRI.DBH	Average DBH of primary species
	PRI.BA	Basal area/acre of primary species
	PRI.7.STEM	Stems/acre of primary species < 7"
	PRI.ORIG	Year of origin of primary species
	SEC.DBH	Average DBH of secondary species
	SEC.BA	Basal area/acre of secondary species
	SEC.7.STEM	Stems/acre of secondary species < 7"
	SEC.ORIG	Year of origin of the secondary species
	TER.DBH	Average DBH of tertiary species
	TER.BA	Basal area/acre of tertiary species
	TER.7.STEM	Stems/acre of tertiary species < 7"
	TER.ORIG	Year of origin of tertiary species
	CON.BA	Stand's conifer basal area/acre
	HWD.BA	Stand's hardwood basal area/acre
	CON.7.STEM	Total conifer stems/acre < 7"
	HWD.7.STEM	Total hardwood stems/acre < 7"
TWP.CODE	Township code	

entity represented within the files LULC.COM (Land Use/Land Cover COMMON forest stand data) and LULC.EVEN (Land Use/Land Cover EVEN-aged forest stand data) is PUBLIC LAND SURVEY TOWNSHIP. TWP.CODE (township code) uniquely identifies instances of each one.

### Verify Entities

After data entities were tentatively identified, they were tested against a series of questions developed to validate their usefulness as entities (Table 3). These questions were synthesized from various writings on data entity analysis (see, for ex-

ample, Fleming and Von Halle, p. 88). The analysis of approximately 400 data items led to the identification of 34 major data entities within the database (Table 4).

### Define Entities

Rigorous, business-oriented, definitions were then developed for each data entity. Definitions included:

- Description of the reasons, rules, agreements, and conventions that justified existence of the entity within the department;
- Estimated volume of entity instances and potential growth rate within the department's area of interest;
- Party(ies) responsible for entity instance creation, update and deletion;
- Retention time of each entity instance in the database; and
- Formal entity name and short identifier to be used throughout the department.

### Determine Relationships

Data relationships are the "glue" that hold the schema together. They are found by asking the question, "What is the business, or logical, association between entity A and entity B?" For example, an obvious relationship between the entities TIMBER CRUISER and TIMBER SALE CRUISE would be a "performing" relationship. That is, "a TIMBER CRUISER performs a TIMBER SALE CRUISE." Entity-relationship modeling requires the ability to read the relationship in both directions—and be understandable to the user. Therefore the converse would read, a "TIMBER SALE CRUISE

TABLE 3.  
Data entity verification test.

- 1) Is it a person, place, or thing, concept or event about which there is enough interest to collect and store data?
- 2) Does an identifier exist which uniquely identifies each instance of the data entity? If not, can one be developed which will meet the needs of the organization?
- 3) Does the entity definition apply to all instances of the entity?
- 4) Is it stable, non-controversial business concept?
- 5) Is it relatively equal in importance, or at the same level of detail, to other organizational entities?
- 6) Does the entity have a unique set of data items?
- 7) Is it more appropriately described as a relationship or data item of another entity?
- 8) Is the entity definition devoid of conditional language (e.g., if, or, except)?
- 9) Can it be named with a singular noun or noun phrase, which describes one instance of the entity?

TABLE 4.  
Major data entities identified in the agency's geographic information system.

AGRICULTURAL UNIT	TIMBER CRUISER
BLOCK PLANNING UNIT	TIMBER HARVEST ACTIVITY
COORDINATE POINT	TIMBER PURCHASER
COUNTY	TIMBER SALE
CROSSING STRUCTURE	TIMBER SALE AGREEMENT
FOREST STAND	TIMBER SALE CRUISE
HYDROLOGIC UNIT	TIMBER SALE EVENT
LAND EXCHANGE	TIMBER SALE UNIT
LAND PARCEL	TRANSPORTATION ROUTE
LAND SURVEY CORNER	TREE SEED LOT
LAND SURVEY UNIT	TREE SPECIES
REGIONAL FIELD UNIT	TRUST PARCEL
RESOURCE UNIT	WATER BODY
ROAD SEGMENT	WATERCOURSE
SEED DRUM	WATER RESOURCE INVENTORY
	AREA
SILVICULTURAL ACTIVITY	
SOIL MAPPING UNIT	
SOIL TYPE	
SURVEY LINE	

is performed by a TIMBER CRUISER" (Figure 1).

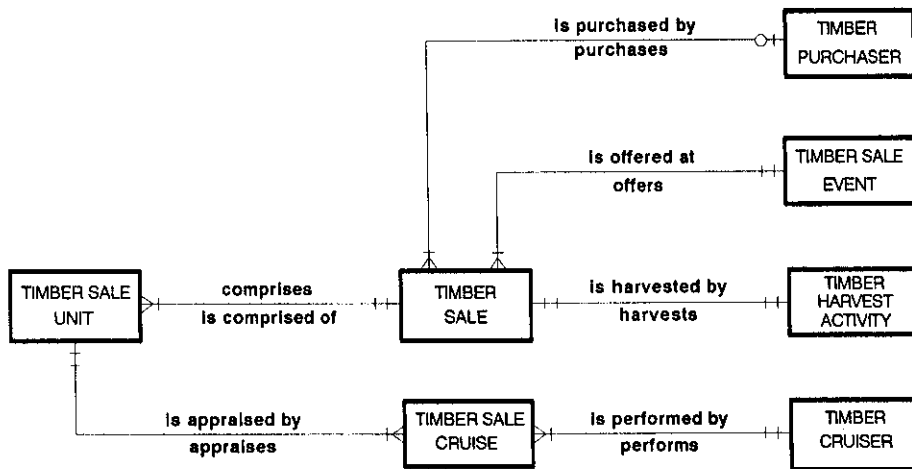
### Define Relationships

Note the symbols at each end of the relationship lines. These graphically represent an important part of the relationship definition called cardinality. Three symbols are used: 1) a circle represents zero entity instances, 2) a line represents one entity instance, and 3) a "crow's-foot" represents many entity instances. The symbol pair represents the smallest and largest number of possible instances. The cardinality constraint is always expressed after the relationship word(s) for that particular relationship direction. Taking the example of TIMBER SALE to TIMBER PURCHASER relationship, the full relationship phrase set would read, "a TIMBER SALE is purchased by zero to one TIMBER PURCHASERS" and "a TIMBER PURCHASER purchases one to many TIMBER SALES." In this example, analysis pertains to entities in the GIS timber sale data domain. Note that data entities are represented as rectangular boxes and relationships are read from box to box in a clockwise fashion.

A good data relationship definition should also include aspects of time dependency (past, present, and future variations in the relationship role) and what conditional situations exist for the relationship (the business procedures that precede the relationship event to make it possible and/or valid).

Good relationship documentation is critical during database

FIGURE 1.  
Portion of GIS data schema showing the TIMBER SALE subject area.



design. It determines how data files are optimally associated and individually structured based on solid business rules (such as those based around the TIMBER SALE to TIMBER PURCHASER relationship). Note that "optimally" here refers to the optimal logical data configuration. Conversion from the logical to physical data schema may include optimization based on physical system or access constraints. This process is beyond the scope of this paper.

### Create Data Schema

When the preceding steps were accomplished, the complete documentation was incorporated into a Computer-Aided Systems Engineering (CASE) package. This was in the form of logically connected text and graphic objects representing the pertinent data entities and relationships. Creation and formalization of the data schema within the CASE tool facilitated the ex-

amination of the information in a complete business—or enterprise—context.

### Incorporation Into Enterprise Schema

A geographic information system logical data schema by itself is very useful as a documentation and applications development tool. The real power of this approach, however, is that it can be combined with a complete organizational data schema so that enterprise information management can be accomplished (Ezizbalike, Coleman, Cooper and McLaughlin 1988). Concurrent data modeling representing other functional areas within the agency have been, and are continuing to be, completed. The logical data schema for a part of the agency's Contract Management System is shown in Figure 2.

For the most part, there is little subject content in common

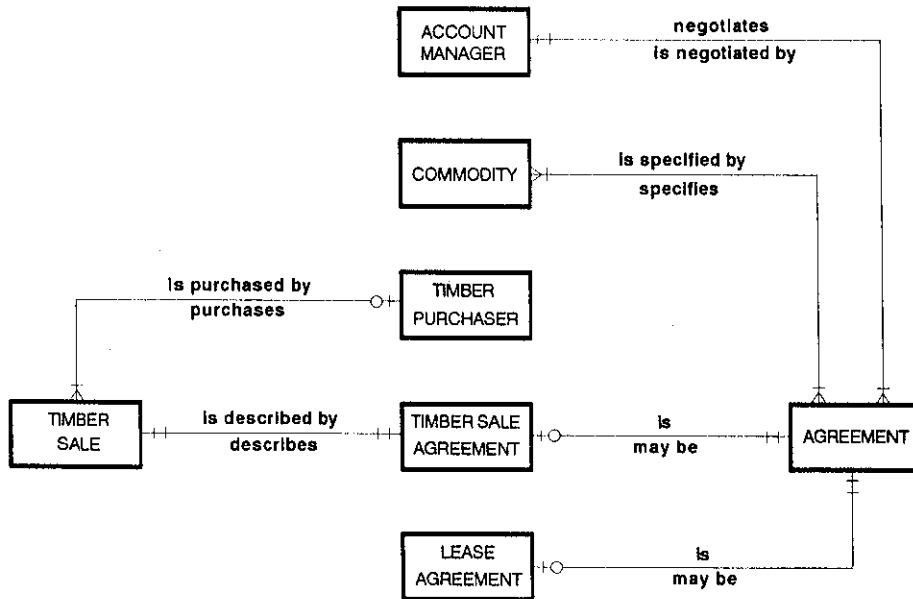
with the GIS logical data schema. However, there are some notable data areas in common. The entities TIMBER SALE and TIMBER PURCHASER occur in both. This indicates the existence of an information interface that could be taken advantage of between two different functional areas within the agency—that of natural resources management and contract management. Detection of this data overlap without the use of an enterprise data model would be difficult. Additionally, the fact that there are common entities emphasizes the need, and provides a basis, for common enterprise data definition. At a high level, these data schemata, along with others, will be combined to provide a complete view of the department's data resource.

### Benefits in Using a Logical Data Schema

#### Better Physical Design

A logical database model is described as being useful if it meets the practical considerations of being usable, implementable, modifiable and general enough to accommodate geographic concepts (Feuchtwanger 1989, p. 607). Specifically in the area of model implementation, it is our contention that database physical design is enhanced by the use of a logical data schema. The following describes some of the benefits from the use of the entity-relationship modeling technique in developing the schema.

FIGURE 2.  
Portion of contract management system data schema showing the AGREEMENT subject area.



- Data integrity is promoted with the establishment of an architecture for data. As an example, data "mutation" can be controlled. That is, one data item should always have only one discrete definition. Data content should exactly match that definition. As applications mature and business requirements dictate change, the understandable tendency of the system maintainer is to add new meaning to pre-existing data items, e.g., adding age categories to tree species codes. It's easier and quicker to do this rather than incorporating a new data item (tree age category) to the physical file structure. The schema, and accompanying documentation, provides a clear record showing which data should, and should not, be present for a particular information object.
- Database flexibility is ensured by design based on sound logical data modeling techniques as de-

scribed in detail by Chen (1976) and Van Roessel (1987). This frees the user to address the database using any number of query combinations. Structural independence from original requirements definition is important in allowing this.

- Database stability is enhanced when the logical data schema is carried through to physical implementation. In general, data organized by entity will be longer lived than those data structures based only on the original "time-frozen" application requirement. For example, where data are representing coordinate values, the initial application requirement may show the need for the data items, X-COORDINATE and Y-COORDINATE. "Value" would describe each of these. This works fine until a change in business requirements dictate the need for elevation values to be tied to the same coordinate points on the ground. There is no

Z-COORDINATE data item. Adding new data items to the database every time new requirements surface is expensive and can prove disruptive to the user community. A more stable way of structuring the database is to identify the correct entity in the first place. In this case COORDINATE POINT would be a likely candidate. "Value" and "axis" would describe the entity. The data items in this scenario would be COORDINATE.POINT.VALUE and COORDINATE.POINT.AXIS. Adding "Z" coordinates in response to a change in business requirements merely entails an additional code in the axis code table.

- Data redundancy is not always undesirable. In a relational database environment, redundant data exist in the form of relational keys. However, unnecessary redundancy can be controlled with the help of an established logical data schema. With a logical schema in place, it is clear where each data item should reside in a complex, subject-oriented, database. For example, customer information such as name, mailing address and phone number may be stored in several functional areas of an organization. The schema, however, will express this information once under the CUSTOMER entity.

### Establishment of Common Terminology

Data naming has been traditionally *ad hoc*, names often conveying no meaning, or worse, false meaning. The logical data schema provides a beginning point for a standard naming structure. That is, an entity abbreviation can serve as the first component of every data item

name. This ensures that confusion is avoided over such data names as NAME, especially where the name occurs several times with different meanings (homonyms). For example, NAME could be describing a wide range of entities including CUSTOMER, TIMBER SALE, COUNTY, STREAM, and so on. Conversely, with one data entity represented by one formalized entity name in the enterprise logical data schema, multiple data names for the same data (synonyms) can be avoided.

### Data Transfer

Compared to traditional computer program-based applications, the database environment has made data sharing relatively easy. Data naming and structuring based on a logical data schema make the transfer of data not only possible, but practical and desirable. The schema provides a common frame of reference for parties involved on both ends of the transfer. Databases constructed from the logical, entity-oriented, data schema will be more open to receiving and providing data since the data will have been designed according to what it is—not how it is used in a specific application.

Usefulness of the entity concept in data sharing is explained in detail in the proposed Spatial Data Transfer Standard (SDTS). It is considered a key component in facilitating data interchange. This standard will contribute to the task of exchanging data throughout federal (and hopefully other) organizations.

The SDTS, at the time of this writing, has been submitted to the National Institute of Standards and Technology for approval as a Federal Information Processing Standard (Spatial Data Transfer Standard Technical Review Board 1990).

### Conclusion

The development of a logical data schema from an existing database environment entailed the following steps.

- 1) Establish a tentative entity list by examining data items and determining what they are describing;
- 2) Scrutinize the list of tentative data entities and apply the entity verification test;
- 3) Define the entities;
- 4) Determine the business relationships between entities;
- 5) Define the relationships and connect the appropriate entities;
- 6) Create the data schema and formalize.

Adhering to these data analysis steps will provide a data architecture from which future application enhancement or system development can take place. When expanded, it can also serve as a data documentation tool which can be used to identify critical data connections throughout the organization. The results of this applied structure and associated documentation will greatly increase information connectivity within the organization, user and analyst comprehension of the data resource, database stability, and application/system development efficiency.

In reference to the entity-relationship construct, Calkins and

Marble (1987, p. 119) have stated that "... the risks involved in the development of a large, expensive database can be substantially reduced . . . , the useful life of the cartographic database should be extended, again contributing to the economic viability of the . . . system." In recent months, this view has been substantiated at the Washington State Department of Natural Resources.

---

### Acknowledgements

The author wishes to thank all those who took time to review and offer suggestions on this paper's content. Joy Denkers, Mary Smith, and Larry Sugarbaker provided continual encouragement for this paper. Their support ensured its successful completion.

---

### Note

1. The definition for the acronym "CASE" is seen in some literature as "Computer-Aided Software Engineering." This author believes the use of "Systems" (rather than "Software") to be more descriptive of the tool's capabilities.

---

### References

- Bedard, Y. and Paquette, F. 1989. "Extending Entity/Relationship Formalism for Spatial Information Systems." *Auto Carto 9*: 818-827.
- Calkins, H.W. and Marble, D.F. 1987. "The Transition to Automated Production Cartography: Design of the Master Cartographic Database." *The American Cartographer* 14, No. 2: 105-119.
- Chen, P.P. 1976. "The Entity-Relationship Model—Toward a Unified View of Data." *Association for Computing Machinery Transactions on Database Systems* 1, March: 9-36.
- Ezizbalike, I.F.C., Coleman, D., Cooper, R.H. and McLaughlin, J.D. 1988. "Managing a Multi-User Land Information System." In *Proceedings, 26th Annual Meeting of the Urban and Regional Information Systems Association*, Vol. 3, 55-64.
- Feuchtwanger, M. 1989. "Geographic Logical Database Model Requirements." *Auto Carto 9*: 599-609.

- Fleming, C.C. and Von Halle, B. 1989. *Handbook of Relational Database Design*. Reading, Massachusetts: Addison-Wesley, Inc. 605 pp.
- Margaronis, J.S. 1990. "Beyond Experimentation." In *Proceedings, Third Annual KnowledgeWare International User Conference*. Atlanta, Georgia. Vol. 1, B31-B83.
- Nyerges, T.L. 1989. "Information Integration for Multipurpose Land Information Systems." *Journal of the Urban and Regional Information Systems Association* 1, No. 1: 27-38.
- Nyerges, T.L. 1980. "Representing Spatial Properties in Cartographic Data Bases." *Technical Papers of the American Congress on Surveying and Mapping*: 29-41.
- Plotkin, D.N. 1991. "Designing a Human Resources System on Chevron." In *Proceedings, Fourth Annual KnowledgeWare International User Conference*. Atlanta, Georgia. Vol. 1, 335-363.
- Spatial Data Transfer Standard Technical Review Board. 1990. *Spatial Data Transfer Standard, version 12/90*. U.S. Department of the Interior, USGS National Mapping Division, Washington, DC.
- Van Roessel, J.W. 1987. "Design of a Spatial Data Structure Using the Relational Normal Forms." *International Journal of Geographical Information Systems* 1, No. 1: 33-50.