User's Guide

# ArcCAD®

*The GIS for AutoCAD®*

# Contents

## Chapter 4 Display and query

## Chapter 5 Spatial analysis



## Chapter 6 ArcCAD utilities



## Appendices

Chapter 1

# Introduction to ArcCAD

ArcCAD® software is a geographic information system (GIS) engine for AutoCAD®. It adds powerful database management and spatial analysis tools to the powerful graphic editing and geometric construction capabilities of AutoCAD. These GIS tools are integrated within the AutoCAD environment and provide the ability to create, manipulate, analyze and display topologically correct geographic data in digital form. ArcCAD is characterized by a topological and georeferenced database linked to an AutoCAD drawing and the GIS functions that operate on those georeferenced data. ArcCAD also provides extensions to the AutoCAD programming interface through both AutoLISP® and the AutoCAD Development System (ADS), providing a comprehensive set of powerful developer tools for complete customization of the software.

## Functions and capabilities

ArcCAD provides many commands for creating, managing, analyzing and displaying geographic data. These capabilities are described here in more detail.

### Topology creation

ArcCAD provides commands to create and maintain correct topology for ArcCAD databases and to create the feature attribute tables (database files) that store thematic data about the map features contained in those databases. Polygon, line and point features with attributes can be created from either AutoCAD drawing entities or from ARC/INFO® coverages. In addition to building polygon and line topology, the commands used to create topology can perform geometric analysis on linear features to identify new intersections, perform line generalization, and automatically resolve overshoot and undershoot errors after digitizing using AutoCAD commands. Topology creation and maintenance are discussed in more detail in the chapter ‘Database automation’.

## Associating attributes with map features

Using the relational database management system data model, ArcCAD provides tools that allow you to associate descriptive tabular data with your map databases. Database tables that hold all descriptive data about all points, lines, or areas on a map can be built and maintained. Database tables can be edited and new fields added to these tables; related tables can be merged; or a subset of fields can be extracted from the tables. New field values for any feature (point, line or polygon) can be calculated or changed, and a set of features can be selected based upon their attributes. Additionally, new data files can be created that contain descriptive information about map features. Attribute management commands are discussed in more detail in the chapter 'Database automation'.

## Display and query

ArcCAD includes an extensive set of commands used to facilitate and enhance the display of map features. Point, line and polygon features can be displayed using a wide variety of standard and user-defined point, line and area symbols. Features may be symbolized by using their attribute values to control their symbolization. Features may be displayed as entities using any combination of AutoCAD entity properties. Displaying features using entity properties may also be controlled through attributes. By combining these methods with the AutoCAD graphic design tools, ArcCAD can be used to create high-quality cartographic products. Interactive query functions put maps to work as graphic windows into your geographic database. The cursor can be used to select any set of features for both graphic and attribute display. You can also ask to see only those features that match specified criteria; for example, you can show all parcels whose zoning characteristics are inconsistent with the current land use map. ArcCAD also has functions that allow you to create or select drawing entities as a result of a GIS query and create or select GIS features as a result of a drawing query. Display and query commands are discussed in more detail in the chapter 'Display and query'.

## Spatial analysis

ArcCAD provides advanced capabilities for manipulating and analyzing the spatial relationships in geographic information. Many commands are available to give you flexibility in data analysis including polygon overlay, line-in-polygon and point-in-polygon overlay, buffer creation, and dissolve functionality. Spatial analysis functions can be combined to provide an almost unlimited variety of analysis and modeling capabilities. These functions allow you to combine multiple input data sets to create new data containing attributes derived from the input data sets. Using this derived data, you can see new relationships between your input data sets, allowing you to make informed

decisions pertaining to your application. Performing spatial analysis with ArcCAD is discussed in more detail in the chapter 'Spatial analysis'.

## Utility tools

Additional utility commands are provided to help you manage your ArcCAD databases. Database management commands allow you to copy, rename and describe ArcCAD databases. Database transfer utilities allow ArcCAD databases to be transferred to other computer systems running ArcCAD or ARC/INFO. The ArcCAD utility tools are discussed in more detail in the chapter 'ArcCAD utilities'.

## AutoLISP and ADS interface

ArcCAD provides significant extensions to both the ADS and AutoLISP application development environments that allow developers to access and directly manipulate features in the geographic database. ArcCAD user commands can also be invoked from either the ADS or the AutoLISP program environment. This combination of programming features greatly facilitates either customizing ArcCAD through the AutoLISP macro tools or developing new geographic application systems using the ADS programming tools. Please refer to the *ArcCAD Programmer's Guide* for complete information about ArcCAD software's AutoLISP and ADS extensions.

# Getting started

The *ArcCAD User's Guide* assumes that you already know how to use AutoCAD. It is not intended to instruct you on using AutoCAD commands or procedures. Please refer to the *AutoCAD Command Reference* for complete information on using AutoCAD and AutoCAD command syntax.

ArcCAD is a set of application programs that run within AutoCAD. Both AutoCAD and ArcCAD must be installed and configured for your computer before you can run ArcCAD. Please refer to your *AutoCAD Installation Guide*, included with your AutoCAD documentation, for instructions on installing AutoCAD.

Refer to the *ArcCAD Installation Guide* for information on configuring your system and properly installing ArcCAD.

## Command interface

ArcCAD is designed to run from within the AutoCAD drawing editor environment. ArcCAD commands menus and dialog boxes look and behave just like AutoCAD commands, menus and dialog boxes.

There are several ways to enter ArcCAD commands:

- From the Command: prompt—Both AutoCAD and ArcCAD commands can be entered at the Command: prompt.

- From a menu—A menu file is provided with ArcCAD that provides pull-down menu access to all ArcCAD commands. If you have configured AutoCAD for screen menus, AutoCAD commands will be available in the screen menu when using the ArcCAD menu file. This way, you can use ArcCAD commands from the pull-down menus and AutoCAD commands from the screen menu.

- From a dialog box—Some ArcCAD commands are incorporated into a dialog box interface. These dialog boxes were designed to have the same look and feel as AutoCAD dialog boxes.

## Getting help

To get online help for any ArcCAD command, use the **archelp** command or select the **archelp** choice from any ArcCAD menu. The help information displayed on the screen provides information from the command reference for that command. Refer to **archelp** in the *ArcCAD Command Reference* for more information.

## ArcCAD command conventions

The ArcCAD command interface adheres to several useful conventions that can be invoked at certain command prompts as part of a command dialog sequence. These command conventions are listed below:

- Any ArcCAD command may be canceled by pressing the AutoCAD cancel key (**ESC** on some platforms, **control-C** on others).

- Whenever a command prompts for a measurable distance, the value may be provided using any AutoCAD method for specifying the distance between two points.

## Themes

■ Whenever a theme must be created to store the output of an ArcCAD operation on a GIS data set (e.g., the output theme created by the **buffer** or **clean** commands), if the output theme does not exist, the theme will be defined during the command. The following message will be printed on the screen, and you will be placed into a subset of the **defthm** command that allows the specification of a GIS data set and symbol number. Other theme definition parameters can be assumed from the context of the GIS operation and need not be provided. After the output theme is defined, the ArcCAD operation will resume.

```
Theme output_theme is not defined.
Defining output_theme...
```

■ Whenever a command prompts for a theme name, specifying a question mark (?) at the **Theme name** prompt will execute the **listthm** command. This allows you to get a list of available themes when an ArcCAD command requires the name of an existing theme.

■ Whenever a theme name is required, the default value will be the last theme name that was specified. If no previous theme name was specified, the default theme name will be 'THEME'.

■ A theme name can contain alphabetic characters, numbers and the underscore character, and it can be up to 31 characters in length.

## Items

■ Whenever a command requires the specification of an item name, entering a question mark (?) at the **Item name:** prompt will execute the **items** command. This allows you to get the list of available items in a given theme when an ArcCAD command requires the name of an existing item.

■ To access an item in a related theme (as defined using the **relate** command), preface the item name in the related theme with a pound sign (#):

```
#ITEM_NAME
```

■ Wherever a character item can be specified at an item prompt, ArcCAD will also accept the concatenation of two or more consecutive items. Item concatenation is achieved by using the following syntax:

```
ITEM1+ITEM2
```

This syntax convention indicates that ITEM1 and ITEM2, and any items between ITEM1 and ITEM2, should be used as if they were a single CHARACTER item. Note that the concatenation is temporary and only in effect

during the execution of the command with which it was invoked, and that only consecutive items can be concatenated.

You can also use this syntax convention for items in a related theme. Note that you only need to specify the # symbol once, at the beginning of the concatenated item parameter; for example,

```
#ITEM1+ITEM2
```

You cannot combine or concatenate items across the current theme and a related theme.

■ An item range can be specified wherever a list of item names can be entered on an ArcCAD command line. The syntax for a range item is

```
ITEM1:ITEM2
```

This syntax indicates that ITEM1 and ITEM2, and all items between ITEM1 and ITEM2, form the list of item names for the command. Use range items only for specifying consecutive item names.

■ Blank characters in CHARACTER item types are valid characters and have to be accounted for during item comparison and query operations.

## Symbols

■ Whenever a command requires the specification of an ArcCAD symbol number, entering a question mark (?) at any Symbol: prompt will execute the **syminfo** command for the appropriate symbol type (point, line, shade or text). This allows you to get a list of available symbols in the current symbolset file when an ArcCAD command requires a symbol number. At any Symbol: prompt, selecting the symbol index command for the appropriate symbol type (**linindex**, **mrkindex**, **shdindex** or **txtindex**) from any ArcCAD menu will display an icon menu of available symbols from the default symbolset of the appropriate type. Selecting a symbol from the icon menu will input that symbol number into the current ArcCAD command.

## Database file pathnames

Database files can be accessed in several ways in ArcCAD. If a database file is part of a theme (i.e., the database file associated with a point, line, polygon, tic or record theme), using the theme name accesses the theme's associated database file. To access a database file independently of a theme, any of the following methods may be used:

■ Any database file from any directory can be accessed via its pathname. The general form of a data file pathname is

```
DRIVE:  \PATH\WORKSPACE_DIR\DATA_FILE
```

e.g., `C:\DIRA\WORK1\INDEX.PAT`

■ Coverage database files (feature attribute tables) can be accessed in ArcCAD by specifying

```
coverage.data_file
```
or
```
coverage\data_file
```

For example, specifying either ROADS.AAT or ROADS\AAT accesses the arc attribute table associated with the ROADS coverage.

■ In ArcCAD, each of the following names represents the same database file:

```
C:\DIRA\WORK1\PARCELS.PAT

C:\DIRA\WORK1\PARCELS\PAT

PARCELS.PAT          Assumes your current directory is:  C:\DIRA\WORK1
```

The convention of specifying a database file name as `name.fil` implies that '`name`' is a subdirectory name and that '`fil`' is a .DBF file called '`fil.DBF`' under the directory name '`name`'. Another way of specifying `name.fil` is '`name\fil.DBF`'.

## Undo

■ Any ArcCAD command that modifies the current drawing can be undone by using the **undo group** command. This can be abbreviated to the letter '**u**'. **Undo** cannot be used to undo ArcCAD commands that modify a GIS data set or theme (e.g., you cannot undo the **buffer** command).

Chapter 2

# ArcCAD concepts

A geographic information system (GIS) is an organized collection of computer hardware, software and geographic data designed to efficiently capture, store, integrate, update, manipulate, analyze, create and display all forms of geographically referenced information.

Finding the answers to complex spatial questions is possible using a GIS. Geographic information systems excel at providing tools that can be used to solve problems that require sophisticated spatial analysis. Solutions to the same problems could be more difficult, time consuming, or potentially even impractical using a CAD system. CAD systems have tools optimized for display, editing and geometric construction of graphic objects. These tools are not designed to create or manipulate and analyze topological relationships between these graphic objects. Because the CAD data model is limited for geographic analysis, an extended data model is required to provide a geographically referenced spatial database on which GIS operations may be performed. The GIS data model uses a spatial database to provide answers to geographical questions.

ArcCAD® software integrates a geographically referenced spatial database on which GIS operations may be performed into the AutoCAD® environment. Links to the CAD database are provided that manage the relationship between the ArcCAD geographic features and AutoCAD entities. In ArcCAD, the CAD database is used to store graphical representations (drawings) of the geographically referenced spatial database (coverages) stored in the GIS database. This GIS database is optimized to support the storage and analysis of all types of geographic features (point, lines, polygons) and topological relationships between them (linear connectivity and contiguity). ArcCAD software's GIS tools provide the ability to associate and manage tabular attribute data that describes those geographic features. ArcCAD also provides a powerful set of GIS operators that can be used to automate, manipulate, analyze and display the GIS database. This synergistic relationship between the CAD and GIS data models, and the inclusion of GIS operations into the AutoCAD toolkit, greatly facilitates the geographic design and analytic capabilities of AutoCAD.

# Spatial data concepts

The GIS database consists of geographic information. The geographic information is the engine as well as the limiting factor in a GIS. The design and structure of your geographic database lie at the heart of your day-to-day operations. Because this chapter describes the ArcCAD database concepts and capabilities on which your GIS is built, it serves as the foundation for the work you will do with ArcCAD.

Gaining an understanding of how ArcCAD organizes and models geographic data will facilitate your use of the GIS and support critical design decisions you must make regarding the investment required to build and maintain your geographic database.

The concepts presented in this chapter build logically upon each other beginning with the set of fundamental map concepts required to learn and apply GIS. If you are new to GIS and ArcCAD or feel that you need to review some fundamental concepts about maps, start here. If you understand maps well and have a grasp of vector data structures and feature representation using points, lines, and areas, if you understand how map symbols and labels represent descriptive data about features, and if you know what topology is, then skip to the section ‘Themes’.

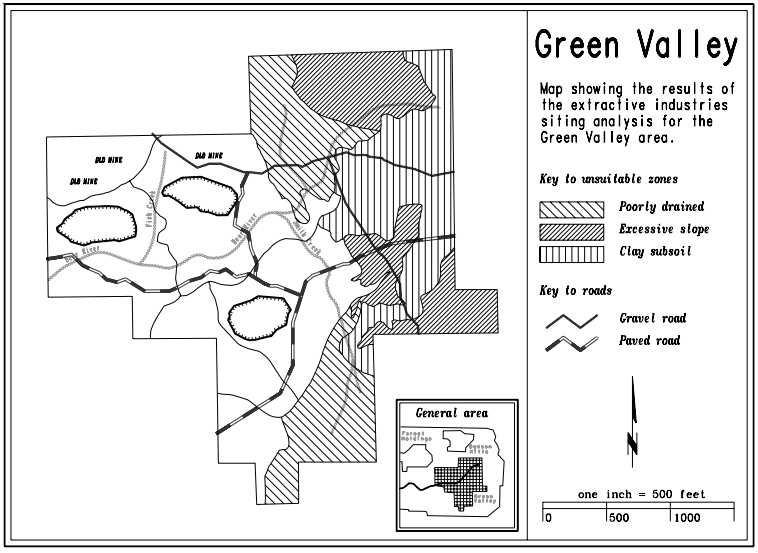
## Representing geographic features

Many geographic data types are supported by ArcCAD. The purpose of this section is to introduce the key spatial data concepts underlying ArcCAD—particularly those related to the geographic data formats supported by ArcCAD.

There are two basic types of geographic information:

- Spatial information describing the location and shape of geographic features
- Descriptive information about features

Maps are the most common method used for presenting spatial data, a method with which we are all familiar. However, there are a few key map concepts that provide an excellent foundation for understanding how a GIS stores geographic information.




## How maps convey spatial information

The spatial information contained within a map is described graphically as a set of points, lines and areas.

**Points** Points define discrete locations on a map for geographic phenomena that are too small to be depicted as lines or areas, such as well locations, telephone poles and buildings. Points can also represent locations that have no area, such as mountain peaks.

**Lines** Lines represent the shapes of geographic features that are too narrow to depict as areas (e.g., streets and streams) or linear features that have length but no area, such as elevation contours.

**Areas** Areas are closed figures that represent the shape and location of homogeneous features such as states, counties, parcels, soil types or land use zones.

## How maps convey descriptive information

Maps present descriptive information about geographic features using map symbols and labels. Here are some common ways that maps depict descriptive information about geographic features:

- Roads are drawn with various line widths, patterns and colors to represent different road classes (e.g., interstate highways are wide, solid red lines).
- Streams and water bodies are typically drawn in blue to indicate water.
- Special point symbols are used to denote locations such as airports; for example, .
- Mountain peaks are labeled with their elevations.
- City streets are labeled with names and often address range information.

AutoCAD users are probably already familiar with how maps depict descriptive information through symbology and feature labeling. You may have already developed a standard set of symbols and labeling standards for your own maps.

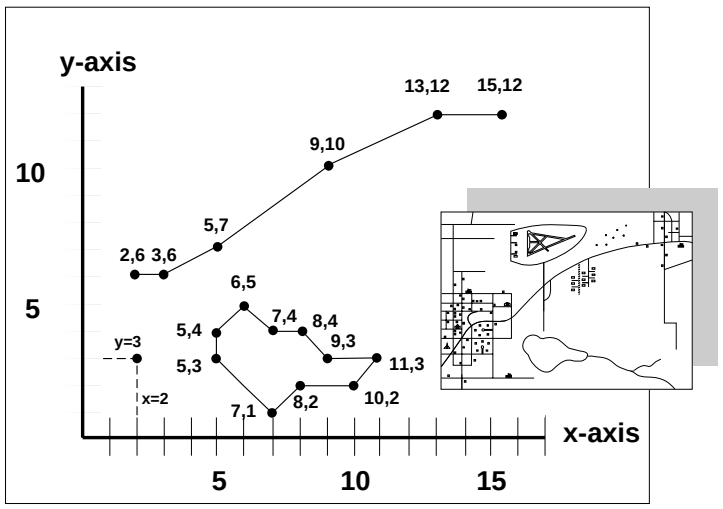
## Spatial relationships

Spatial relationships between various geographic features are implicit on map sheets, but they depend upon you as the map reader to interpret them. You can look at a map and easily interpret many spatial relationships among the features. For example, you can find a route to the airport on a street map (i.e., calculate a path). You can tell mountainous from uniform terrain by looking at contours. You can see where parks are next to roads and determine what features fall within a city boundary. And, you can estimate the elevation of a mountain lake from the surrounding contours. Such information is not represented explicitly on a map. Instead, your mind interprets these relationships from the mapped objects—the streets, contours and other base map features.

A GIS database must depict or store such spatial relationships or provide tools to derive them. These must be explicit or easily derived so that this information is readily available for the spatial operations typically performed with the GIS. *Topology* is a concept used to represent or model such spatial relationships in geographic databases. Topology is used to define areas, describe how linear features connect and identify contiguous areas. Later in this chapter, we'll clarify what topology is and what role it plays in your GIS database.

# Representing spatial information

Features on the earth's surface are mapped onto flat, two-dimensional maps as points, lines and areas. An x,y (Cartesian) coordinate system is used to reference map locations to ground locations.




*Representing points, lines and areas using a vector data structure*

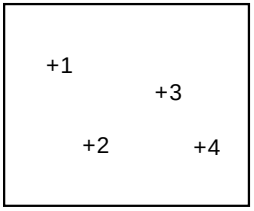
In the example above, a building location is represented as a single x,y coordinate pair, a road becomes a line represented by a series of connected coordinate pairs, and a lake is represented as a series of coordinates defining an enclosed polygon.

Each point is recorded as a single x,y location. Lines are recorded as a series of ordered x,y coordinates. Areas are recorded as a series of x,y coordinates defining line segments that enclose an area; this is the origin of the term *polygon*, meaning 'many-sided figure'. With x,y coordinates, you can represent points, lines and polygons as a list of coordinates instead of as a picture or graph. In the figure above, for example, the coordinate pair 2,3 represents a point location; the coordinate pairs 2,6 3,6 5,7 9,10 13,12 and 15,12 represent a line; and the coordinate pairs 6,5 7,4 8,4 9,3 11,3 10,2 8,2 7,1 5,3 5,4 and 6,5 represent a polygon. Notice how the first and last coordinates are the same; a polygon always closes. Conceptually, these coordinate lists represent how map features are stored in a computer as sets of x,y digits.

## Representing multiple features

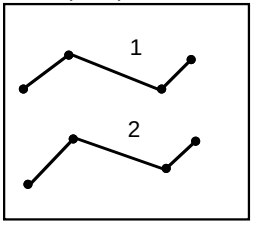
It's easy to see how the coordinates for one feature can be stored. If you have many features, each can be assigned a sequence number. Then coordinates can be recorded for each feature by keeping a sequence number with the list of coordinates for each feature; for example,

Points



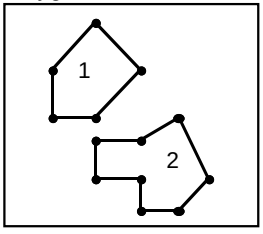

| Point number | x,y coordinates |
|---|---|
| 1 | 2,4 |
| 2 | 3,2 |
| 3 | 5,3 |
| 4 | 6,2 |

Lines (Arcs)




| Line number | x,y coordinates |
|---|---|
| 1 | 1,5 3,6 6,5 7,6 |
| 2 | 1,1 3,3 6,2 7,3 |

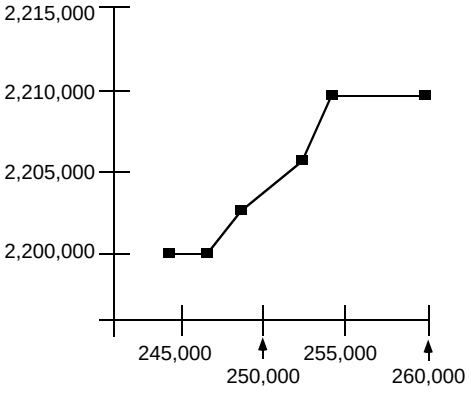
Polygons




| Polygon number | x,y coordinates |
|---|---|
| 1 | 2,4 2,5 3,6 4,5 3,4 2,4 |
| 2 | 3,2 3,3 4,3 5,4 6,2 5,1 4,1 4,2 3,2 |

# Map projections

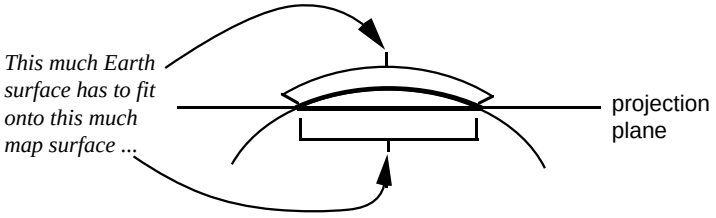
The coordinates shown so far represent *page units*, such as inches or centimeters, and can be used to measure a distance on the map or find an x,y location using a ruler. But maps usually represent real-world coordinates that have been projected onto a flat surface. These coordinates represent a real location on the earth's surface in one of several coordinate systems.




| X | Y |
|---|---|
| 244,674 | 2,200,000 |
| 247,874 | 2,200,000 |
| 248,362 | 2,203,118 |
| 252,932 | 2,206,176 |
| 254,683 | 2,209,897 |
| 261,102 | 2,209,897 |

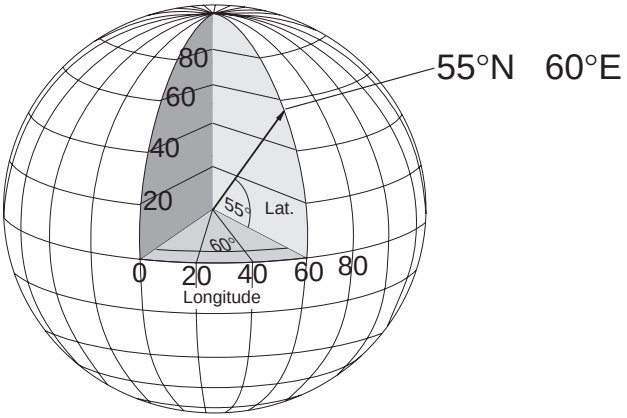
In this example, the real-world coordinates are projected into a coordinate system called *Universal Transverse Mercator* (or UTM). The units are meters. Some other commonly used coordinate systems for mapping include Lambert Conic Conformal and Albers Conic Equal-Area, with x,y coordinates measured in feet or meters. These coordinate systems are based upon a concept called a *map projection.*

Since the earth is a sphere, some method must be used to depict a map in two dimensions. Map projections are used to represent all or part of the earth's surface as a flat map.




*... consequently, map projections distort some parameter of the Earth's surface be it distance, area, shape, or direction.*

People often consider latitude–longitude measures to be x,y coordinates; however, they are not. Latitude–longitude is a unit of measure for a spherical coordinate system.




*Latitude–longitude are angles measured from the earth's center point to locations on the earth's surface. Latitude measures angles in a north–south direction; longitude measures angles in an east–west direction.*

It is important to stress that the latitude–longitude geographic reference system is not a map projection. For example, length is constant anywhere within a Cartesian coordinate system, but it cannot be consistently measured in degrees of latitude–longitude. In the figure above, note how the meridians (vertical reference lines) converge at the poles, but separate or bulge as they get closer to the equator. One degree of longitude at the equator is about 111 km in length while the length of one degree of longitude converges to 0 at the North and South poles.

Latitude and longitude measures for any location can be projected into a planar coordinate system using a map projection. Thus, latitude–longitude values play an important role in map feature registration.

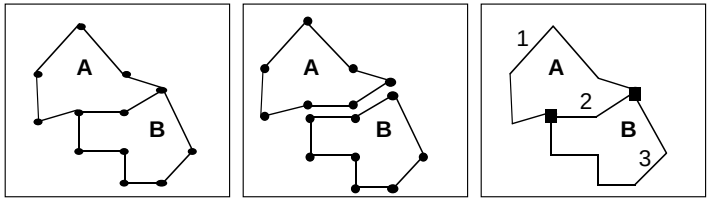
The ArcCAD data structures utilize map projections to store data in an x,y planar coordinate system. Refer to the *Map Projections Guide* for an introduction to map projections.

# Maps as graphs

So far, you've seen how geographic features can be stored as a series of coordinates. Many features are made up of a combination of other features. For example, adjacent polygons share a common boundary. Storing each polygon as a closed loop of coordinates is inefficient because the sides between adjacent polygons are stored twice. There is a more efficient way to store polygons using an *arc–node* data structure. In fact, this data structure also facilitates many types of analyses.

Note that arcs in a graph are not the same as AutoCAD arc entities. Arcs in a graph are always line segments, whereas AutoCAD arc entities are curved drawing elements.

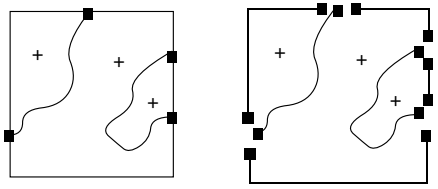
Arcs are the linear elements of a graph—the smallest segments defined by splitting lines where they intersect each other. Nodes are the intersection locations where arcs meet.




*Polygons A and B depicted as a closed loop of coordinates (middle) and using an arc–node data structure (far right).*

Two alternative methods for representing polygons are shown on the middle and right graphs above. The middle depicts each polygon as a closed loop of coordinates and the far right as a set of lines (or arcs) that defines each polygon boundary. Notice how arcs are numbered 1, 2 and 3 in the right graph. Polygon A is defined by arcs 1 and 2; polygon B is defined by arcs 2 and 3. Coordinates can be stored for each arc, reducing coordinate redundancy. Polygons are thus defined as a list of arcs.

One way to conceive of a map is as a graph containing basic elements of points, arcs and nodes. Other features such as areas and networks are composed of these basic elements. Such features are defined *topologically*; that is, as objects composed of other features.



*Areas in the left map can be graphically defined as a collection of arcs and nodes as shown in the exploded view to the right.*

## Topology

Topology is a procedure for explicitly defining spatial relationships. The principle in practice is quite simple. Different types of spatial relationships are expressed as lists of features (e.g., an area is defined by the arcs comprising its border).

Creating and storing topological relationships have a number of advantages. As you will see later in this section, data are stored more efficiently when topology is used. Therefore, you can process data faster and process larger data sets. Topology facilitates analytical functions, such as modeling flow through the connecting lines in a network, combining adjacent polygons with similar characteristics, identifying adjacent features, and overlaying geographic features.

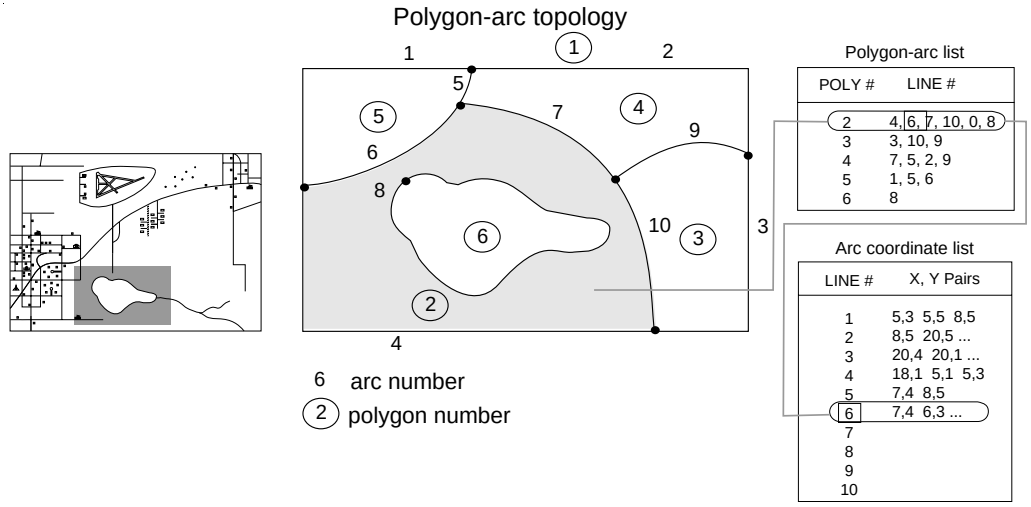
The three major topological concepts in ArcCAD are

- Arcs that connect to surround an area define a polygon (area definition).
- Arcs have direction, and left and right sides (contiguity).
- Arcs connect to each other at nodes (connectivity).

## Area definition

Polygons can be represented topologically as the arcs defining the polygon, rather than as a closed loop of x,y coordinates. A list of the arcs that make up each polygon is stored and used to construct the polygon when necessary (for example, when drawing polygons). In the following illustration, arcs 4, 6, 7, 10 and 8 comprise polygon 2 (the 0 before the 8 indicates that this arc creates an island in the polygon).

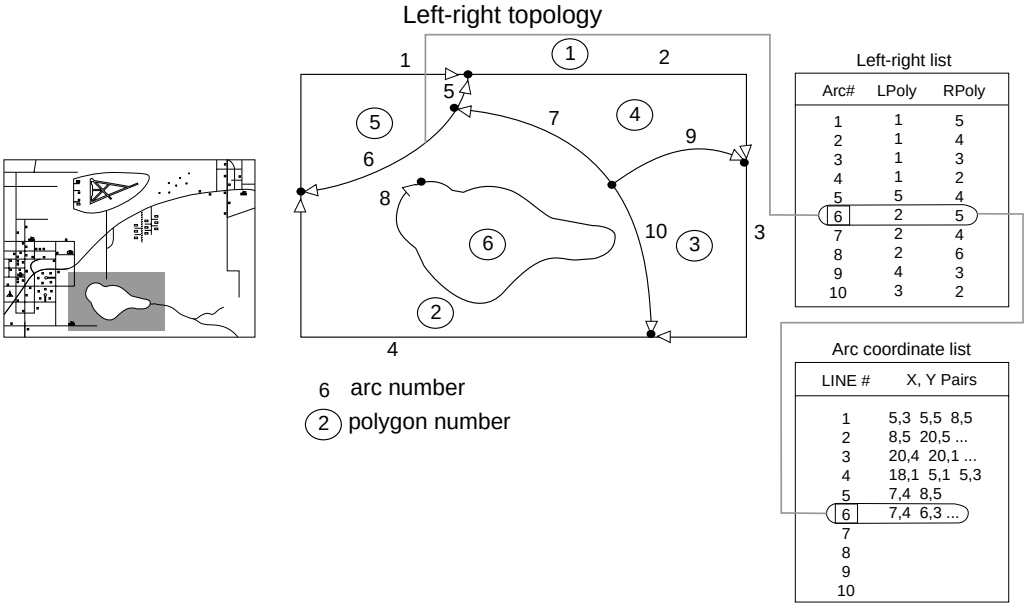
Each arc may appear in the list of arcs for each polygon (in the illustration below, arc 6 appears in the list for polygons 2 and 5); arc coordinates need be stored only once. Storing each arc only once reduces the amount of data and also ensures that the boundaries of adjacent polygons don't overlap.




*Vegetation type boundaries surrounding a lake are represented as polygons.*

## Contiguity

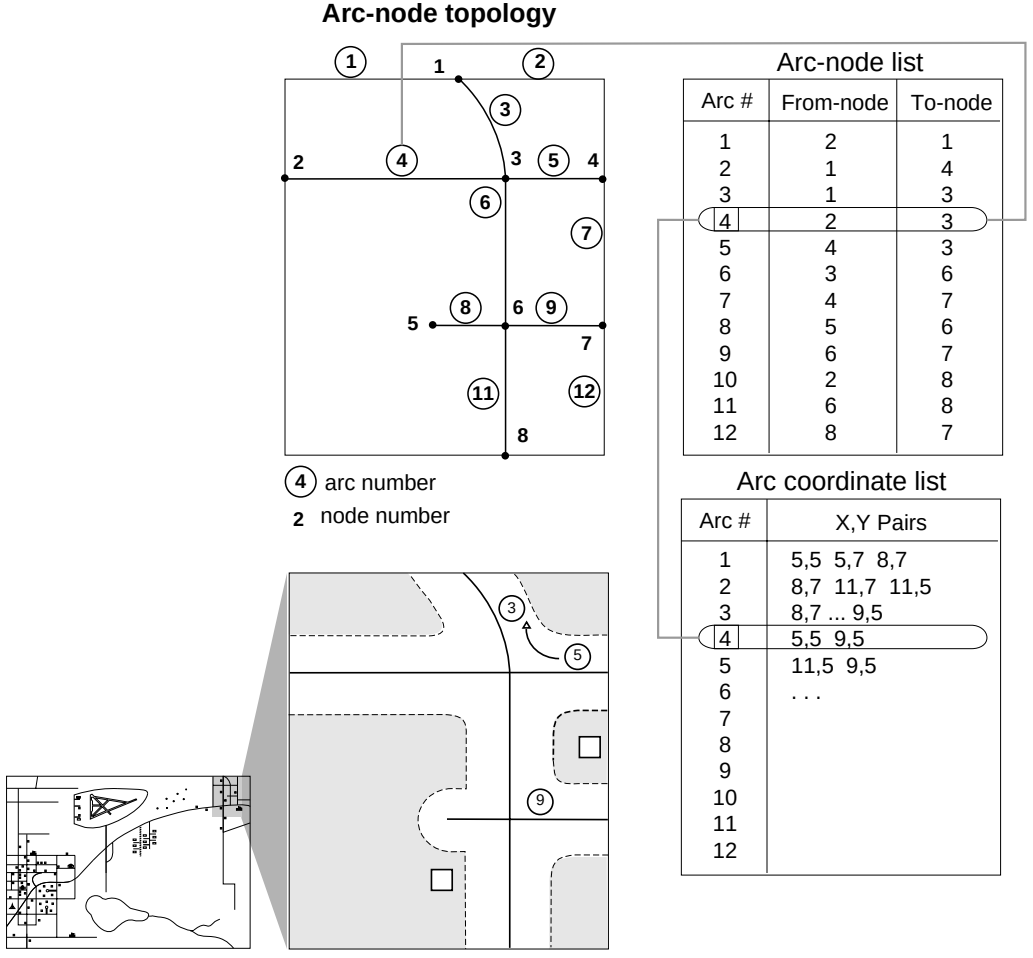
Any polygons sharing a common arc are adjacent. Because every arc has direction (a from-node and a to-node), a list of the polygons on the left and right sides can be maintained. In the illustration below, polygon 2 is on the left of arc 6, and polygon 5 is on the right. Thus, we know that polygons 2 and 5 are adjacent. Notice that the label for polygon 1 is outside the boundary of the area. This polygon is called the external or *universe* polygon and represents the area outside all of the polygons in your map. This outer polygon is necessary so that each arc always has a polygon on its left side and a polygon on its right side.




*Contiguity allows the computer to recognize, for example, that forest land is adjacent to the lake.*

## Connectivity

The x,y pairs along each arc, called *vertices*, define the shape of the arc. The endpoints of an arc are called *nodes*. Each arc has two nodes: a from-node and a to-node. Arcs can join only at their endpoints, or nodes. By tracking all the arcs that meet at any node, ArcCAD knows which arcs connect to each other. In the illustration below, arcs 3, 4, 5 and 6 all join at node 3. The computer now knows that it is possible to travel along arc 5 and turn onto arc 3 because they share a common node (3), but it's not possible to turn directly from arc 5 onto arc 9 because they don't share a common node.




*Arc features are used to represent street centerlines. Nodes are located at street intersections.*

# Methods for representing descriptive information

Descriptive attributes associated with map features are stored in the computer in a manner similar to how coordinates are stored. Attributes are stored as sets of numbers and characters. For example, the attributes for a set of lines representing roads might include

- Road type — 1 = divided highway<br>2 = arterial or collector roads<br>3 = major roads<br>4 = residential streets<br>5 = unpaved
- Surface material — concrete<br>asphalt<br>gravel
- Width — measured in feet
- Number of lanes — count of lanes
- Name — the name of each road

The descriptors for each road segment are stored in the computer as a string of values in a predefined format, such as

| ROAD-TYPE | SURFACE | WIDTH | LANES | NAME |
|---|---|---|---|---|
| 2 | Asphalt | 48 | 4 | N Main St |

Given many features, one set of attributes is stored for each feature in a database file.

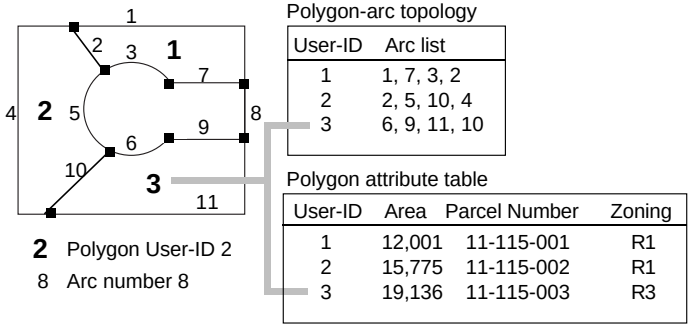
| | ROAD-TYPE | SURFACE | WIDTH | LANES | NAME |
|---|---|---|---|---|---|
| | 2 | Asphalt | 48 | 4 | N Main St |
| | 1 | Concrete | 60 | 4 | Hwy 42 |
| Record | 4 | Asphalt | 32 | 2 | Elm St |

Item

Such a database file is referred to as a *feature attribute table*. Each row in the table is a record and contains descriptive information for a single feature. The same columns or fields appear in each record. These columns are referred to as *items*.

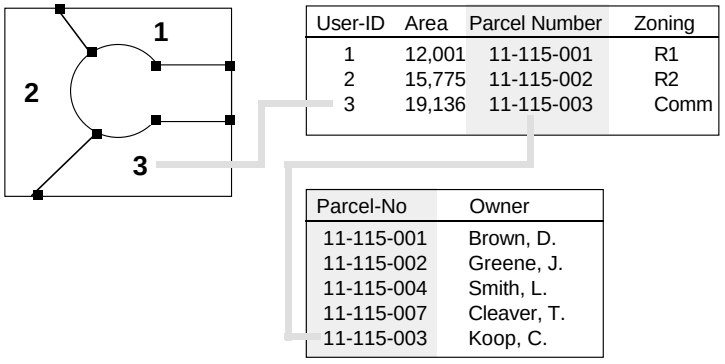
# Linking attributes to features—the georelational model

You have seen how features are represented through coordinates and topology, and how descriptive data are organized as a series of records in tables. The next concept to understand is how a link is created between features and their corresponding attribute records. The answer is quite simple—a feature number associates the attributes with the feature coordinates maintaining a one-to-one correspondence between the spatial records and the attribute records. Once this connection is established, you can query the map to display attribute information or create a map based on the attributes stored in the feature attribute table. This feature number is called the *User-ID*.




One powerful capability of GIS lies in the link between the spatial data and the tabular (descriptive) data. Thus, a hybrid data model, often referred to as the *georelational model*, is used to maintain the connection between features and their descriptive data.

The relational concept can be applied to more than just keeping track of features and their attributes. Any two tables that share a common attribute can be connected. A *relate* uses a common item to establish connections between corresponding records in two tables. In a relate, each record in one table is connected to a record in another table that shares the same value for a common item. The common item is referred to as a *key*. In the example below, the parcel number is used to relate two tables. The key can also be used to permanently merge the two related tables. This type of operation is known as a *relational join.*




*In the georelational model, features and corresponding attribute records are 'tagged' with a common feature number (e.g., polygon numbers 1, 2 and 3).*

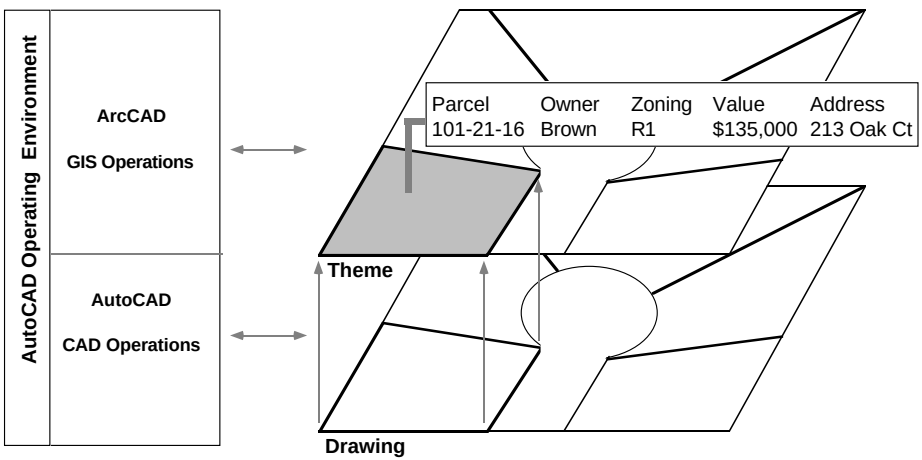
# Themes

So far, you have seen how geographic data are stored as a series of x,y coordinate pairs representing points, lines and polygons. You have also seen how the relationships between these features are made explicit through topology. And you've seen how associated attribute data can be organized in tables and linked to features through a feature number. Now let's take a look at how geographic information is organized in ArcCAD.

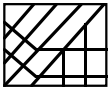
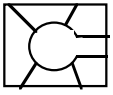
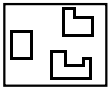
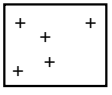
ArcCAD organizes geographic information in *themes*. Themes are logical collections of geographic features with common characteristics defined by the user. Examples of themes include streets, wells, soil, polygons and streams.

An AutoCAD drawing can represent a symbolized map display, but is limited for geographic operations such as defining complex polygons (e.g., those with over 100,000 vertices and many islands), maintaining the link between features and their attributes, and topologically defining linear connectivity and adjacency. These relationships are essential for sophisticated GIS operations.

A theme in ArcCAD extends a drawing by adding support for topologically defined geographic features and their descriptive attributes. Themes are implemented using ARC/INFO® coverages to represent geographic features and database tables stored in dBASE® format to store attributes. Geographic features in a theme can be linked to AutoCAD entities in an AutoCAD drawing. The drawing is used for display and editing. The theme is used for spatial analysis and attribute management. ArcCAD automatically maintains the relationships between entities in a drawing and geographic features in a theme.




*Conceptual organization of the ArcCAD data model*

| Theme | Geographic Features | Representation | Attributes | |
|---|---|---|---|---|
| Soils  | Soil types | Polygons | Materials<br>Texture<br>Slope<br>Depth<br>Drainage | Erosion<br>pH<br>Nitrogen<br>Phosphorus<br>Potassium |
| Land cover  | Land cover | Polygons | Dominant species<br>Canopy closure<br>Stem density | Mid-story<br>DBH |
| Water  | Lakes<br>Ponds<br>Other large water bodies | Polygons | Type<br>Place name | |
| Streams  | Streams | Lines | Name<br>Type<br>Width | Perennial/Intermittent<br>Order |
| Street network  | Streets | Lines | Type<br>Address range<br>Width<br>Pavement | Condition<br>Capacity<br>Speed limit<br>Traffic flow |
| Parcels  | Lots<br>Road casings | Polygons | Site address<br>Assessor parcel number<br>Legal area<br>Owner, address<br>Zoning | |
| Buildings  | Building footprints | Polygons | Building type<br>Scanned photos<br>Permit number | |
| Wells  | Water wells<br>Surface pumps | Points | Depth<br>Pump type<br>Name<br>Gallons per minute | |

*Examples of themes*

# Components of a theme

Themes are composed of the following:

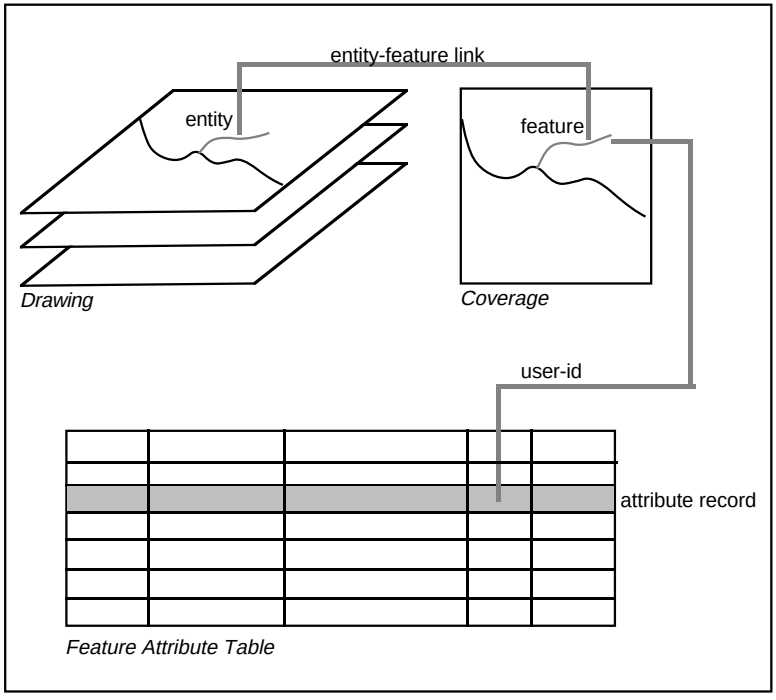
- A collection of features of the same class. ArcCAD supports seven feature classes: point, line, polygon, annotation, tic, record (dBASE and SQL) and image.

- A GIS data set. Point, line, polygon, annotation and tic features are stored in ARC/INFO coverages; record themes are stored in dBASE files or RDBMS tables.

- For a coverage, a feature attribute table with one record per feature. Feature attribute tables store descriptive information about theme features. A record in a feature attribute table is linked to its corresponding feature by the User-ID.

- For each feature, an optional link to a corresponding entity in an AutoCAD drawing. This link is called an *entity-feature link*.

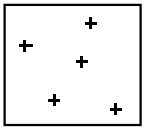
The following figure illustrates the components of a theme:




*Components of a theme*

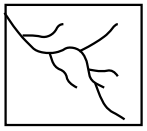
## Theme feature classes

The feature class of a theme determines the type of geographic features that are stored in that theme. ArcCAD supports seven theme feature classes: point, line, polygon, annotation, tic, record (dBASE and SQL), and image.



### Point themes

A point feature is a single x,y coordinate pair that defines the location of a map feature whose boundary or shape is too small to be displayed as an area or line. Examples of point features are wells, mountain peaks, light poles. Point features are stored in coverages. The point feature class links point entities (points, text, attdef and inserts) in the AutoCAD drawing to point features in the coverage.

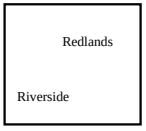


### Line themes

Line themes contain arc features. An arc feature is a set of connected, ordered points and represents a linear feature that is too narrow to be displayed as an area or a map feature that has no width. Examples of arc features are roads, streams, contour lines and faults. Arc features are stored in coverages. The line feature class links linear entities in the AutoCAD drawing (lines, arcs, traces and polylines) to arc features in the coverage.

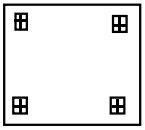
### Polygon themes

A polygon feature is a closed figure whose boundary encloses a homogeneous area. Examples of polygon features are parcels, counties, lakes and census tracts. Polygon features are stored in coverages. Coverages containing polygon features store the boundaries of polygons as a connected set of arc features and a point feature inside each polygon to label that polygon. The polygon feature class links areal entities in the AutoCAD drawing (closed lines and plines, circles, polygons, solids, doughnuts, ellipses) with polygon features in the coverage.




### Annotation themes

Annotation features are text strings used to label features in other theme feature classes. Annotation is not spatially related to any other feature class and is only used for display purposes. The annotation feature class links text entities in the AutoCAD drawing to annotation features stored in the coverage.



### Tic themes

Tic features provide registration or geographic control points for a theme by representing known locations on the earth's surface. Tics allow all theme features to be recorded in a common coordinate system (e.g., Universal Transverse Mercator [UTM] meters or State Plane feet). Tics are used to ensure that all the themes in an ArcCAD database register to each other and to adjacent themes. Tics are stored as part of every ArcCAD coverage. Every theme with a feature class of point, line, polygon or annotation has tics even if a tic theme has not been defined for those tics.

| parcel | date | tax_id |
|---|---|---|
| 1-88234 | 1/4/92 | 44A88734 |
| 1-33665 | 1/11/92 | 86T88721 |
| 1-35088 | 1/7/92 | 93J188236 |

### Record themes (dBASE and SQL)

Record themes store attribute information in database files. Record themes are a special type of theme that does not have a graphical representation or topology. Record themes are primarily used to store additional attributes that can be related to theme features stored in a coverage. Records can also be used to link related database files to a theme. There are two types of record themes: dBASE and SQL. dBASE record themes store attribute data in dBASE-compatible database files. SQL record themes access attribute information stored in external RDBMS tables.

### Image themes

Image themes allow the display of raster images as a backdrop for ArcCAD themes. Image themes support many types of color and monochrome images.

The following table shows the characteristics of the above seven feature classes:

| Theme Feature Class | Graphical Data (AutoCAD ) | Geometric Data (coverage) | Topological Data (coverage) | Attribute Data (database file) |
|---|---|---|---|---|
| Line | color, line-type, width, thickness, elevation | – 2 nodes (begin and endpoint of arc) – one or more arcs each up to 500 vertices | – arc-node topology | AAT (Arc Attribute Table) |
| Polygon | none* (polygon boundaries are represented as lines) | – one or more lines | – polygon topology (list of arcs defining each polygon) | PAT (Polygon Attribute Table) |
| Point | color, thickness, elevation, size, mode | X,Y coordinate pair | none | PAT (Point Attribute Table) ** |
| Tic | color (green) | X,Y coordinate pair | none | none |
| Annotation | color, style, slant, angle, height. | 1 or 2 X,Y coordinate pairs. | none | none |
| Record dBASE | none | none | none | application defined attribute data |
| Record SQL | none | none | none | application defined attribute data |
| Image | none | none | none | none |

* Polygon areas cannot be represented in AutoCAD, only their boundaries can be delineated.

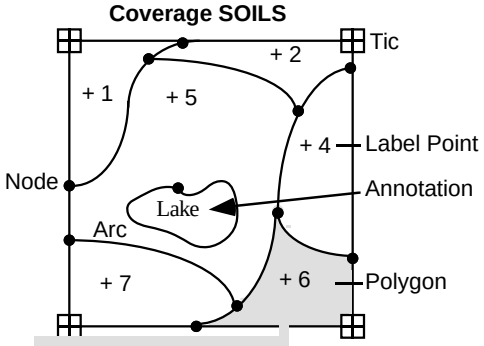
** Feature attributes for points and polygons are stored in a PAT. However, a PAT for any single ArcCAD theme can only represent points or polygons, not both.

## GIS data sets for a theme

Themes store features and attribute data in GIS data sets. A GIS data set can be either an ARC/INFO coverage or a database file. The GIS data set for a theme depends on the theme feature class. Point, line, polygon, annotation and tic themes store data in coverages. SQL and dBASE record themes store information in RDBMS tables and database files, respectively. Images store data in raster bitmap files. Each of these GIS data sets is described here.

### Coverages

A *coverage* is the basic unit of data storage in an ArcCAD GIS data set. A coverage typically represents a homogeneous class of data within a map, such as roads or streams. A coverage is georelational; it contains both the locational data and thematic attributes for geographic features in a given area. Coverages store map features as points, lines or polygons. Map feature attributes such as feature name, symbol, classification and any other desired attributes for each feature are described and stored in associated feature attribute tables. Feature attribute tables are special database files. When these attributes are needed to create maps (e.g., to label or symbolize features), they are accessed from the feature attribute tables (e.g., a coverage of contour lines can be drawn and labeled with their surface elevations, all forest coverage types could be shaded green, street names could be placed along arterial and feeder roads, and so on). Thus a coverage contains both locational data (which defines points, lines and polygons) and attribute data (which describes points, lines and polygons) for each feature in the coverage. Locations of map objects are described by coverage features. Descriptive information about features is stored in feature attribute tables. The figure below shows some of the features that may be present in a coverage:




**SOILS Polygon attribute table (PAT)**

| AREA | PERIMETER | SOILS_ | SOILS_ID | SOIL | CLASS | SUITABILITY |
|---|---|---|---|---|---|---|
| -36.0 | 24.0 | 1 | 0 | — | — | — |
| 3.0 | 9.0 | 2 | 1 | A3 | 113 | HIGH |
| 2.5 | 8.5 | 3 | 2 | C6 | 95 | LOW |
| 15.0 | 15.0 | 4 | 3 | B7 | 212 | MODERATE |
| 4.0 | 8.5 | 5 | 4 | B13 | 201 | MODERATE |
| 2.0 | 4.5 | 6 | 5 | Z22 | 86 | LOW |
| 5.5 | 12.0 | 7 | 6 | A6 | 77 | HIGH |
| 4.0 | 7.0 | 8 | 7 | A1 | 117 | LOW |

For more information about coverages and feature classes stored in a coverage, refer to Appendix C, ‘More on ArcCAD coverage features’.

### RDBMS tables

The GIS data set for an SQL record theme is an RDBMS table and allows access to standard databases supported by AutoCAD. While the database file for a dBASE record theme is a dBASE file on disk (.DBF file extension), the database table for an SQL record theme often resides on an external RDBMS. ArcCAD SQL record themes are designed to function similarly to dBASE record themes except that the tabular data in an SQL theme cannot be changed.

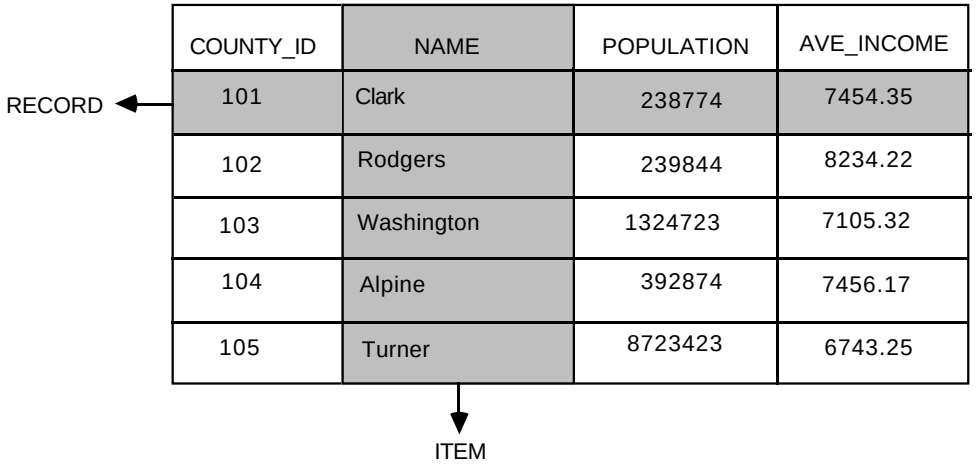
When an SQL record theme is defined, ArcCAD verifies the database connection information and then fetches the queried records as specified in the theme’s definition. When ArcCAD has fetched all the table records and stored them in a temporary file, you can then use the SQL record theme much like you would use a dBASE record theme.

### Database files

The GIS data set for a dBASE record theme is a dBASE-compatible .DBF file. A database file is a 'flat' logical table where columns represent attribute fields and rows contain attribute values. The rows of a table are called *records*, and the columns or fields are called *items*. Each record in the table always has the same items, and each set of columns always represents the same item. For example, columns 1 through 3 might hold the county ID number, columns 4 through 18 might hold the county name, columns 19 through 26 the population, and so on. This means that each record always has the same format or template and is always the same length. In ArcCAD, coverage feature attribute tables (FAT) files are also .DBF files. The format for the table just described above might be:

| Beginning Column | Item Name | Width | Data Type |
|---|---|---|---|
| 1 | COUNTY_ID | 3 | Number |
| 4 | NAME | 15 | Character |
| 19 | POPULATION | 8 | Number |
| 27 | AVG_INCOME | 8 | Number |

A database file with this format might appear as follows:




*Tabular structure of a database file*

In addition to managing feature attribute tables, ArcCAD also provides commands to access data in related database files.

### Item definitions

In ArcCAD, the specification of the record format for a database file is referred to as its item definitions. Each record can be up to 128 items or 4,000 characters (i.e., bytes) long. Items are defined by the following conventions:

- **Item Name**—any name with up to 10 alphanumeric characters. Item names must begin with a character.
- **Item Width**—number of spaces (or bytes) required to store item values.
- **Item Type**—the data type of the item.
- **No. of Decimals**—the number of digits to the right of the decimal place for item types which hold decimal numbers.

| ITEM TYPE | VALID WIDTHS | DESCRIPTION |
|---|---|---|
| C (Character) | 1 - 254 characters | Character strings are fixed length. The item width defines the number of spaces reserved for a character string. |
| D (Date) | 8 digits | Dates are stored as YYYYMMDD. Dates are displayed as: MM/DD/YY. |
| N (Numeric) | 1 - 16 digits | A number with or without a decimal point. If the item represents a real number, then the decimal point takes up a space in the storage space (thus you can have up to 15 digits). |

*ArcCAD-supported item types*

Other item types available in dBASE but not fully supported by ArcCAD include:

- Logical logical data type
- Memo MEMO field

If you have dBASE or a dBASE-compatible DBMS, consult your documentation for more information regarding item definitions.

## Feature attribute tables for themes

Theme feature attribute tables are database files that hold all thematic or descriptive data for theme features. They are managed as an integral part of an ARC/INFO coverage. These database files can be created or updated using the ArcCAD **clean** or **build** commands for point, line and polygon themes. These attribute files can also be accessed using several ArcCAD attribute manipulation commands. Each record in a feature attribute table contains all of the descriptors about one map feature.

The first time **clean** or **build** is executed, a minimal PAT or AAT is created (see the following section for the items contained in each table). Once the table has been created, additional items and values can be added to the tables using the **additem** or **joinitem** commands.

A major feature of ArcCAD is that feature attribute tables are automatically maintained. For example, a subsequent **clean** or **build** will always recreate or update topology and rebuild the theme's feature attribute table (PAT or AAT).

Another important characteristic of feature attribute tables is that they are stored in the dBASE file format. This allows other application software that can read dBASE files to operate on ArcCAD feature attribute tables.

### Point theme attributes

A point theme has an associated point attribute table (PAT). At a minimum, the PAT contains four items:

| PAT Items | Description |
|---|---|
| AREA | Holds the area of a polygon. The value is 0 for points. |
| PERIMETER | Holds the perimeter of a polygon. The value is 0 for points. |
| Cover_ | Internal sequence number (i.e., the record number) of the point feature. |
| Cover_ID | User-assigned feature ID for each point. |

*Additional items can be added to the PAT by the user. They must be added after the Cover-ID item, not inserted before it.*

The PAT is used to hold feature attributes for either point features or polygon features. However, any single theme can only have a PAT for points or polygons, not both.

### Line theme attributes

A line theme has an associated arc attribute table (AAT). The AAT contains descriptive data about arc features. There is one record in the AAT for each arc feature in the theme. The record is related to the feature by the User-ID number stored for each arc feature. At a minimum, the following items are contained in the AAT:

| AAT Items | Description |
|---|---|
| FNODE_ | Internal sequence number of the from-node. |
| TNODE_ | Internal sequence number of the to-node. |
| LPOLY_ | Internal sequence number of the left polygon; set to 0 if the coverage does not contain polygons. |
| RPOLY_ | Internal sequence number of the right polygon; set to 0 if the coverage does not contain polygons. |
| LENGTH | Length in coverage units. |
| Cover_ | Internal sequence number (i.e., the record number) of the arc feature in the ARC file. |
| Cover_ID | User-assigned feature ID. |

*Additional items can be added to the AAT by the user. They must be added after the Cover-ID item, not inserted before it.*

### Polygon theme attributes

Descriptive data about polygon themes is stored in a polygon attribute table (PAT). There is one record in the PAT for each polygon. The record is related to the polygon by the polygon's User-ID.

| PAT Items | Description |
|---|---|
| AREA | Holds the area of a polygon. |
| PERIMETER | Holds the perimeter of a polygon. |
| Cover_ | Internal sequence number (i.e., the record number) of the polygon. |
| Cover_ID | User-assigned feature ID for each polygon. |

*Additional items can be added to the PAT by the user. They must be added after the Cover-ID item, not inserted before it.*

A PAT is also created for point features; however, the AREA and PERIMETER items are assigned values of 0 for point features. A single coverage cannot contain both point and polygon features.

### Annotation theme attributes

Annotation themes do not have associated attributes.

### Tic theme attributes

All tic information for a tic theme is stored in the TIC file, a database file stored with each theme's coverage. The TIC file contains the following items:

| TIC Items | Description |
|---|---|
| IDTIC | The User-ID for each tic |
| XTIC | The tic's x-coordinate |
| YTIC | The tic's y-coordinate |

Although the x,y coordinates for tics are available in the TIC file, editing these values does not change the coordinate system in which other features are stored.

### Record theme attributes

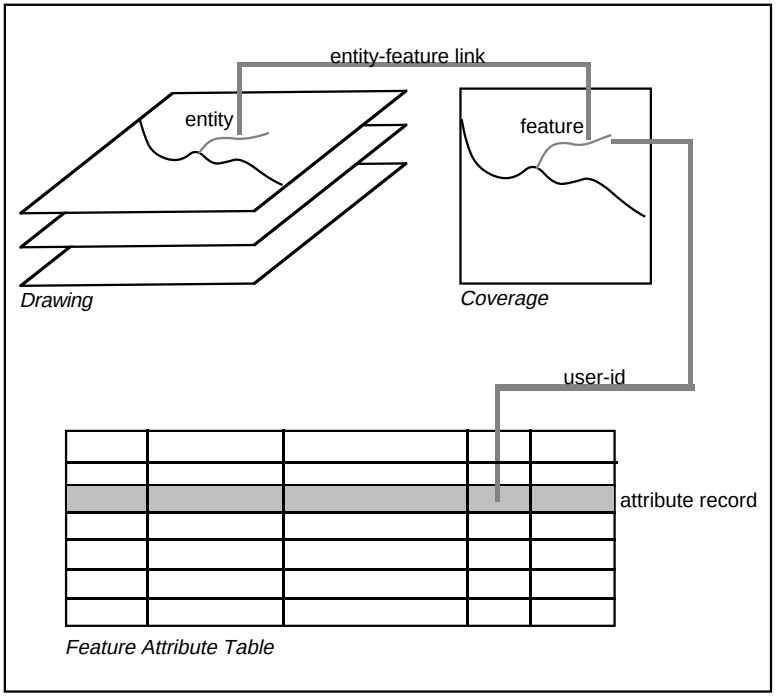
There are two types of record themes: dBASE and SQL. dBASE record themes store user-defined attribute information in dBASE-compatible .DBF files. They do not have a standard record format. SQL record themes store user-defined attribute information in RDBMS tables queried from industry-standard databases such as FoxPro®, PARADOX® and other RDBMS products supported by AutoCAD.

### Image themes

Image themes do not have associated attributes.

# Entity-feature links

In order for drawing entities to represent geographic features in ArcCAD, there must be a relationship or link between a feature and its graphic representation as an entity in an AutoCAD drawing. These entity-feature relationships are known as *entity-feature links*. Entity-feature links are important when creating features from entities (database creation) as well as when creating entities from features (display). They allow a one-to-one correspondence between an entity and its corresponding feature.



# Database design

The GIS database design step is of major importance. It is essential to apply a well-structured, thoughtful approach to design and automation. It is valuable to investigate the database decisions and designs of experienced users with similar applications. For example, the data model for an urban GIS is quite similar for municipalities across the United States. The major database designs of experienced users can help guide your design efforts.

This section shows examples of how geographic database design concepts can be applied to represent a series of typical geographic themes. The goal here is not to explain the steps for database design for an application, but to demonstrate some important design concepts of how a design, once derived, is implemented in ArcCAD.

## Design concepts: Features and themes

Before you read about specific examples, there are some important concepts that provide a foundation for understanding how themes and GIS data sets are designed and organized. There are three main concepts that drive geographic database design.

- *Geographic features* are the real-world phenomena you wish to represent (such as wells, streets, lots, lakes, soil types, and so on).

- A *theme* is a collection of common geographic features defined by the user.

- A *GIS data set* models themes and features using feature classes and attributes.

### Geographic features

It is useful to construct a list or enumerate the geographic features that you are trying to represent in your database. These are the set of geographic features you wish to portray, such as streets, buildings, parcels, land use zones, blocks, streams, sewers, wells, and so on. These are features that you can see, draw on a map and give names to.

For each logical collection of features, you must decide how it will be portrayed (e.g., as points, lines, or areas). This will help to determine how themes are used to represent each set of geographic features.

### Themes

Geographic features are organized logically into groups of layers or *themes*. Themes are logical collections of geographic features with common characteristics defined by the user. Examples of themes include streets, wells, soils polygons and streams.

Typically, you will identify a set of attributes to associate with the geographic features in each theme. For example, for each street, you may want to identify the street class, its name, and perhaps address range information for geocoding applications. In addition, you may want to know the number of lanes, each street's speed limit, the directions you can travel along it, and so on. The public works department may have other needs for descriptive information about streets, such as pavement type, width, a maintenance schedule for each stretch of road, number of lanes, curb type, and so on.

### GIS data sets

The GIS data set stores theme data. A GIS data set can be either a coverage, a database file, or an RDBMS table. Coverages store data for point, line, polygon, tic, and annotation themes while database files store tabular data for dBASE record themes. A coverage is used to model a theme by specifying the theme feature class and items that will depict the theme. For example, streets might be represented by arc features and have an item named STREET_TYPE that can be used to classify streets (e.g., residential streets, major roads, highways, and so on).

## Three steps in database design

Although the task of designing a geographic database is quite extensive, the design process consists of three fundamental steps:

**Step 1.** Identify geographic features and their attributes.

**Step 2.** Organize features into themes or data layers.

**Step 3.** Design themes to be automated.

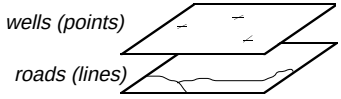
#### Step 1 Identify geographic features and their attributes

Deciding what geographic features are to be represented in your GIS database will be a function of the system objectives. The set of applications to be performed by the GIS will drive the database contents. Most often, the actual database design is a product of a detailed user-needs assessment and pilot study that supports the various departments involved. One major goal of the database

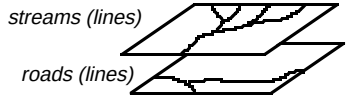
design project should be to prepare a list of the geographic features to be included along with their attributes.

#### Step 2 Organize features into themes or data layers

Once you have identified the geographic features to be depicted and their attributes, you can begin organizing them into themes. A number of factors influence layer organization in a geographic database; many of these factors differ with each application. However, two of the most common considerations for organizing themes include organization by feature type (point, line, or polygon) and thematic grouping of features.




Typically, layers are organized so that points, lines and polygons are stored in separate layers. For example, well sites represented by points might be stored in one layer, while roads represented by lines are organized in another.




Features may also be organized thematically by what they represent. For example, streams might be organized in one layer and roads in another. Although streams and roads are both line features, it makes sense to store them separately because the attributes associated with a stream might include a name, stream class and rate of flow, while attributes for roads might include a name, surface type, and number of lanes. Because their associated attributes differ significantly, the streams and roads should be stored in separate themes referencing the same geographic area.

#### Step 3 Design themes to be automated

A GIS data set is used to model the theme. Coverages store the coordinate and topological information which represent the real-world locations of the geographic features in the theme and a set of attributes about those features. For example, city streets can be represented as a set of arc features whose feature attributes such as street type, speed limit, address range, and so on, are stored in a corresponding Arc Attribute Table (AAT).

A single coverage can represent multiple themes either through alternative items for the same feature class or through different feature classes. For example, the same arc features representing streets could form the boundaries of polygons representing census tracts. Thus, the arc features store attributes related to

streets in an AAT file and the polygons store census tract information in a Polygon Attribute Table (PAT). Although different feature classes and their corresponding feature attributes are stored in a single coverage, the information is accessed through different themes based on the respective feature class.

## Feature attribute code descriptions

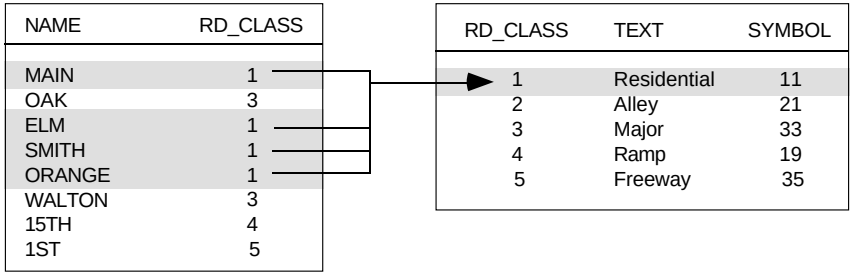
Items in the feature attribute table contain values such as area measurements, text strings and codes. When codes are stored, a special table can be used to list code value interpretations. For example, street types in an urban database might be coded as:

1 = divided highway
2 = arterial or collector road
3 = major road
4 = residential street
5 = unpaved road
9 = unknown or not specified

It is useful to include a table in the attribute database for each theme. In the sample database designs that follow, a code table is often included. This table contains three items: the code item, a text description and a symbol for drawing that class (such as a solid red line to draw divided highways).

## Relates and relate items

Attribute tables can be related to records in a theme's feature attribute table. A common item contained in both themes is used to associate records from one table to records in the other. This item is called the *relate item*. Relating different attribute tables allows you to organize your data sets. For example, related themes let you access attribute data from different sources without physically joining the two data sets. A typical goal in relational database design is to use many-to-one relates to reduce data storage and redundancy as shown in the following illustration:




*Example of a many-to-one relate.*

### Drawing organization

The organization of your AutoCAD drawing is important when creating themes from that drawing. For example, if you have an AutoCAD drawing created from the data obtained from the field survey, you should use the drawing to generate accurate areas and measurements as well as for map display purposes. The themes created from the drawing are used for geographic modeling and spatial analysis where coordinate precision is less important. You must always maintain your original drawing database separately from the geographic database. For example, if you want to add a new subdivision into your parcels theme, first update the information in your original drawing using the survey data, save the original drawing and then update the corresponding changes to your parcels theme.

Much of the drawing you do with AutoCAD consists of positioning entities at coordinate locations. You can assign various portions of your drawing to different layers, and you can define as many layers as you like. You can also assign a color and a linetype to each entity. ArcCAD does not impose any restrictions in the way you organize your drawing. However, we strongly suggest that you organize the entities on appropriate layers based on certain common characteristics before creating GIS data sets.

## Sample database designs

A number of example database designs are presented in the remainder of this chapter. However, before proceeding any further, it is important to understand the relationships between various AutoCAD entity types and the corresponding GIS features. The table on the next page lists the themes and geographic features modeled by each database. In addition, the table also describes the AutoCAD entity types that are used to create geographic features.

The database designs presented in the following sections are not designed to represent the contents of a single geographic database. Instead, real-world objects were selected to illustrate various classes of geographic features. They come from many applications.

An attempt was made to simplify these database designs to cover basic concepts and design issues. A number of critical issues are not presented, but should be addressed in the design of a geographic database; for example,

- How to design feature codes and represent these in your database
- How to assign drawing symbols to features and how to build a symbol library for your GIS
- Choosing a map projection and coordinate system
- How to specify map resolution and accuracy
- How to maintain a data dictionary for your geographic database and what information to store for each geographic data set

Even though these issues are not presented, they are critical considerations for your database design.

| Theme Type | AutoCAD DRAWINGS | | ArcCAD data sets | | | |
|---|---|---|---|---|---|---|
| | **Entity properties** | **Entity types** | **Class** | **Coordinate data** | **Topological data** | **Attribute data** |
| Line | Layer, color, linetype, elevation, thickness, curve, entity type, etc. | Line, Polygon, circle, arc, 3Dface, Trace, Solid | Line | *Arc features*<br>• Unique Identifier<br>• X,Y locations of nodes and vertices | *Arc-Node Topology*<br>• From node<br>• to node<br>• Left poly<br>• Right poly<br>• Internal-ID<br>• User-ID | *Arc Attribute Table*<br>• Entity properties<br>• Street attributes:<br>condition<br>capacity<br>speed limit<br>traffic flow<br>type<br>etc. |
| Point | Layer, color, elevation, thickness, entity type, block attributes, etc. | Point, Shape, Text, Insert (blocks) | Point | *Point features*<br>• Unique identifier<br>• X,Y location pairs | --none-- | *Point Attribute Table*<br>• Entity properties<br>• AutoCAD block attributes<br>• Well Attributes<br>depth<br>pump type<br>name<br>etc. |
| Polygon | Layer, color, elevation, thickness, entity type, block attributes, etc. | All entity types (linear entities define polygon boundaries and location entity types define label points) | Polygon | *Polygon features*<br>• Arc features<br>• Label points | *Polygon-arc-topology*<br>• List of *arc* features defining polygon boundaries<br>• Internal-ID<br>• User-ID<br>• List of label points in each polygon used to assign User-ID | *Polygon Attribute Table*<br>• Entity properties<br>• AutoCAD block attributes<br>• Parcel attributes:<br>parcel number<br>legal area<br>zoning<br>etc. |
| Annotation | Layer, color, entity type, text height, etc. | Text, Attdef | Annotation | • X,Y location of insertion point<br>• text height<br>• text string | --none-- | --none-- |
| Tics | Standard color<br>Standard layer<br>Standard size | Shape | Tic | • Tic-ID<br>• X,Y location | --none-- | --none-- |
| Record (dBASE) | --none-- | --none-- | Record (dBASE) | --none-- | --none-- | *dBASE files*<br>(item specific to each table)<br>• property tables<br>• lookup tables<br>• buffer tables, etc. |
| Record (SQL) | --none-- | --none-- | Record (SQL) | --none-- | --none-- | *RDBMS tables*<br>• user-defined tables |
| Image | --none-- | --none-- | Image | • X,Y Insertion point | --none-- | --none-- |

## Theme: Wells

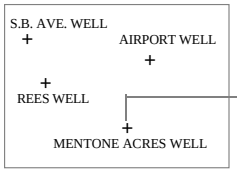
The WELLS theme records the location of each well within the city. It contains both public and private wells, and surface pumps for reservoirs. The city's water supply comes primarily from a series of public water wells and a single reservoir.

Wells are typically designated as *point* features. In an AutoCAD drawing, well locations are generally represented as points, shapes or block entities. These entity types are used to create point features in a coverage. If you have a block entity with associated attributes, you can capture the associated block attributes into a property table (refer to the **addfeat** command reference).

A number of attributes are recorded for well sites, including the pump type (gas or electric), well depth, pipe diameter, output capacity in gallons per minute, and so on. Each of these attributes can be used to draw and symbolize wells. Only one of these, well depth, is illustrated. A dBASE *record* theme, DEPTH, is used to show how values for well-depth can be classified and symbolized. Depth can be classified as 0 to 250 feet, 250 to 500 feet, and so on. The lookup values for DEPTH indicate a maximum class value and are described by the item named TEXT. Special point symbols based on DEPTH values are used to draw well locations; however, they are not illustrated in this summarized database design.

The techniques for classifying well depth in this illustration can be applied to other items such as the output capacity measured in gallons per minute. A lookup table (record theme) to classify item values could potentially be built for each code item: DEPTH, DIAMETER, PUMP_TYPE, HP and GPM.

**Theme: WELLS**




GIS data set: WELLS
Feature class: Point
Feature Attribute Table: PAT
Label item: LOCATION

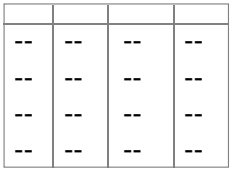
Drawing: WELLS

Entities: point, shape, attdef, block and text insertion points.

Data source: City Public Works Department records

Input scale: 1:24,000
(1 inch = 2,000 feet)

**Theme: DEPTH**



GIS data set: DEPTH
Feature class: Record

Drawing: WELLS
Entities: none
Item definitions:
DEPTH,4,N,0
TEXT,25,C
SYMBOL,3,N,0

**WELLS** theme Point Attribute Table (PAT)

| Item | Data type | Description |
|---|---|---|
| AREA | 13,N,6 | Not used for points, AREA=0 |
| PERIMETER | 13,N,6 | Not used for points, PERIMETER=0 |
| WELLS_ | 11,N,0 | Sequence number of point |
| WELLS_ID | 11,N,0 | User ID |
| LOCATION | 25,C | Location description and well name |
| DEPTH | 3,N,0 | Depth in feet |
| DIAMETER | 7,C | Pipe diameter and units (e.g., " for inches) |
| PUMP_TYPE | 5,C | Type of pump |
| HP | 3,N,0 | Horsepower of pump |
| GPM | 4,N,0 | Gallons-per-minute |
| DISCHARGE | 20,C | Discharge or contract name |

**Contents of table named DEPTH**

| DEPTH | TEXT | SYMBOL |
|---|---|---|
| 0 | Surface (e.g., reservoir) pump | 65 |
| 250 | Up to 250 feet | 18 |
| 500 | 251 to 500 feet | 19 |
| 750 | 501 to 750 feet | 20 |
| 1000 | 751 to 1,000 feet | 21 |
| 1500 | 1,001 to 1,500 feet | 22 |
| 9999 | More than 1,500 feet | 2 |

## Theme: Streets

Many of the most common themes in geographic databases in the United States are based on street centerlines. The street centerlines can be used to classify streets according to street type (e.g., residential, freeway, major road, alley, and so on).

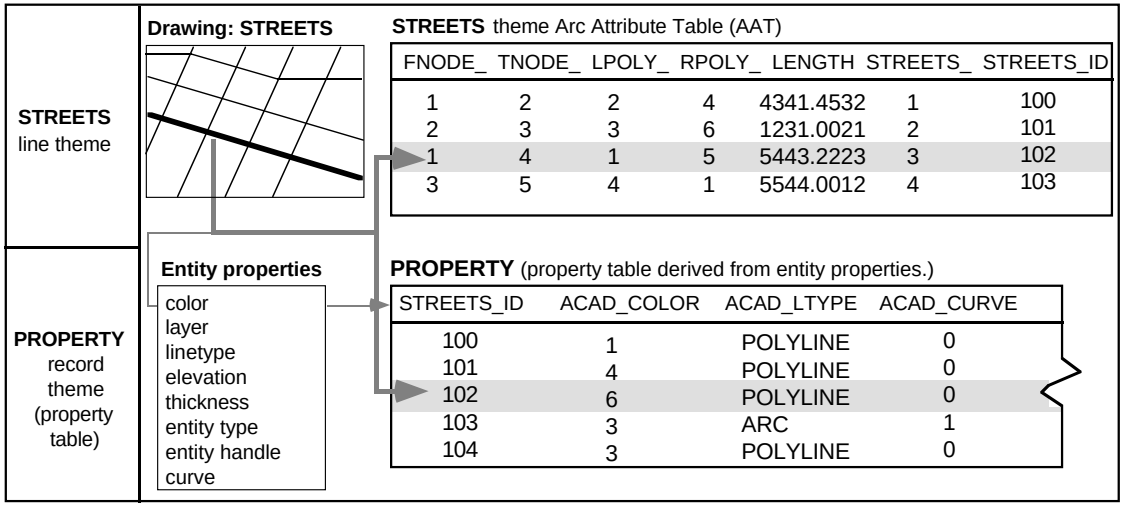
It is important to understand that in ArcCAD you can display theme features in two ways:

**Property driven display** ArcCAD allows you to store entity properties in a property table (record theme) and recreate theme features based on the original AutoCAD entity properties including true curves.

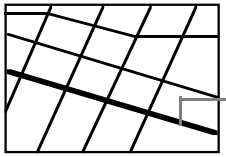
**Attribute driven display** Theme features can be displayed using feature attributes. For example, you can code the streets based on the street type and display theme features using ArcCAD symbology. In addition, you can create lookup tables to display features using desired symbols.

Lookup tables are special kinds of record themes that describe the way theme features are displayed. For example, the Planning Department would like to display theme features based on traffic volumes and the Public Works Department would like to display the same theme by pavement type. Therefore, design your STREETS theme with appropriate relate items to allow individual departments to access the same theme. This kind of design allows various departments to maintain their own tabular databases and relate to the same theme using a common related item such as STREET_TYPE.

The following figure illustrates how the AutoCAD entities (including entity properties) represent the components of the STREETS theme. The opposite page illustrates a comprehensive summary of the STREETS theme.


STREETS
line theme
PROPERTY
record
theme
(property
table)
Drawing: STREETS
STREETS theme Arc Attribute Table (AAT)
FNODE_ TNODE_ LPOLY_ RPOLY_ LENGTH STREETS_ STREETS_ID
1 2 2 4 4341.4532 1 100
2 3 3 6 1231.0021 2 101
1 4 1 5 5443.2223 3 102
3 5 4 1 5544.0012 4 103
Entity properties
color
layer
linetype
elevation
thickness
entity type
entity handle
curve
PROPERTY (property table derived from entity properties.)
STREETS_ID ACAD_COLOR ACAD_LTYPE ACAD_CURVE
100 1 POLYLINE 0
101 4 POLYLINE 0
102 6 POLYLINE 0
103 3 ARC 1
104 3 POLYLINE 0

**Theme: STREETS**



GIS data set: STREETS
Feature class: Line
Feature Attribute Table: AAT

Drawing: STREETS

Entities: line, pline, arc, circle, trace, 3Dface, etc.

Input scale: 1:24,000
(1 inch = 2,000 feet)

**STREETS** theme Arc Attribute Table (AAT)

| Items | Data Type | Description |
|---|---|---|
| FNODE_ | 11,N,0 | Sequence number of the from-node |
| TNODE_ | 11,N,0 | Sequence number of the to-node |
| LPOLY_ | 11,N,0 | Sequence number of the left-polygon |
| RPOLY_ | 11,N,0 | Sequence number of the right-polygon |
| LENGTH | 13,N,6 | Length in feet calculated from State Plane coordinate system |
| STREETS_ | 11,N,0 | Arc sequence number |
| STREETS_ID | 11,N,0 | Arc ID |
| TYPE | 1,N,0 | Type of street. (for values, see STREETTYPE theme) |

**Theme: STREET _PROPERTY**

| | | | |
|---|---|---|---|
| -- | -- | -- | -- |
| -- | -- | -- | -- |
| -- | -- | -- | -- |
| -- | -- | -- | -- |

GIS data set: PROPERTY
Feature class: Record (dBASE)

Drawing: STREETS

Entities: none

Properties: color, layer, linetype, elevation, thickness, entity type, etc.

**PROPERTY** (property table to store entity properties)

| Item | Data type | Description |
|---|---|---|
| STREETS_ID | 11,N,0 | Relate item |
| ACAD_COLOR | 3,N,0 | Entity color value |
| ACAD_LAYER | 31,C | The layer name that entity is on |
| ACAD_LTYPE | 16,C | Entity's line type |
| ACAD_ELEV | 13,N,6 | Entity's elevation value |
| ACAD_THICK | 13,N,6 | Entity's thickness (extrusion ht.) |
| ACAD_ETYPE | 10,C | Entity type (polyline, arc, etc.) |
| ACAD_CURVE | 1,N,0 | Flags entity's curve information |

*Note: Property table is optional. Use only appropriate properties.*

**Theme: STREETTYPE**

| | | |
|---|---|---|
| -- | -- | -- |
| -- | -- | -- |

GIS data set: ST_TYPE
Feature class: Record (dBASE)
Drawing: STREETS
Entities: none
Item definitions: TYPE,3,N,0
LABEL,25,C
SYMBOL,3,N,0

**Contents of table named ST_TYPE**

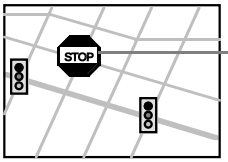
| TYPE | LABEL | SYMBOL |
|---|---|---|
| 1 | Divided highway | 35 |
| 2 | Arterial or collector road | 30 |
| 3 | Off- or On-ramp | 36 |
| 4 | Major road | 12 |
| 5 | Residential street | 11 |
| 6 | Alley | 1 |
| 7 | Unpaved | 17 |
| 9 | Not specified | 2 |

## Theme: Intersections

The STREETS theme, described earlier, can be used to create a new point theme using the **nodepoint** command. The **nodepoint** command takes the STREETS theme as input and creates point features for the endpoints (nodes) of that theme's arc features. This resulting point theme can be used for street intersections and can be coded based on intersection type (e.g., overpass, railroad crossing, street intersection, and so on), the type of traffic control device present, if any, and the name of the cross streets of the intersection.

Since two types of intersection features are represented (i.e., intersection types, and traffic control devices), it may be more appropriate to think of this example as representing two record themes to symbolize intersection features. The contents of each theme are defined by separate lookup tables containing descriptions and point symbols for each intersection type and traffic control device. The figure on the next page illustrates these relationships. The record theme TRAFCONTROL describes traffic control devices, and the other record theme INTERSECTION describes types of intersections and assigns a marker symbol to draw each.

**Theme: INTERSECTIONS**




GIS data set: STREETS
Feature class: Point
Feature Attribute Table: PAT

Drawing: STREETS
Entities: none

Input scale: 1:24,000
(1 inch = 2,000 feet)

**INTERSECTIONS** theme Point Attribute Table (PAT)

| Items | Data Type | Description |
|---|---|---|
| AREA | 13,N,6 | Not used for points; AREA=0 |
| PERIMETER | 13,N,6 | Not used for points; PERIMETER=0 |
| ST_INT_ | 11,N,0 | Sequence number of point |
| ST_INT_ID | 11,N,0 | Point ID |
| INT_TYPE | 2,N,0 | Code for intersection type |
| INT_NAME | 60,C | Names of cross streets with a slash (/) between street names |
| DEVICE | 2,N,0 | Code for traffic control device |

**Theme: TRAFCONTROL**

| | | |
|---|---|---|
| -- | -- | -- |
| -- | -- | -- |

GIS data set: DEVICE
Feature class: Record (dBASE)
Drawing: STREETS
Entities: none
Relate item: DEVICE
Item definitions: DEVICE,2,N,0
LABEL,20,C
SYMBOL,3,N,0

**Contents of table named DEVICE**

| DEVICE | LABEL | SYMBOL |
|---|---|---|
| 1 | Stop sign | 3 |
| 2 | Stoplight | 16 |
| 3 | Yield sign | 6 |
| 4 | Railroad signal | 17 |
| 5 | No device | 0 |
| 99 | Not specified | 0 |

**Theme: INTERSECTION**

| | | |
|---|---|---|
| -- | -- | -- |
| -- | -- | -- |

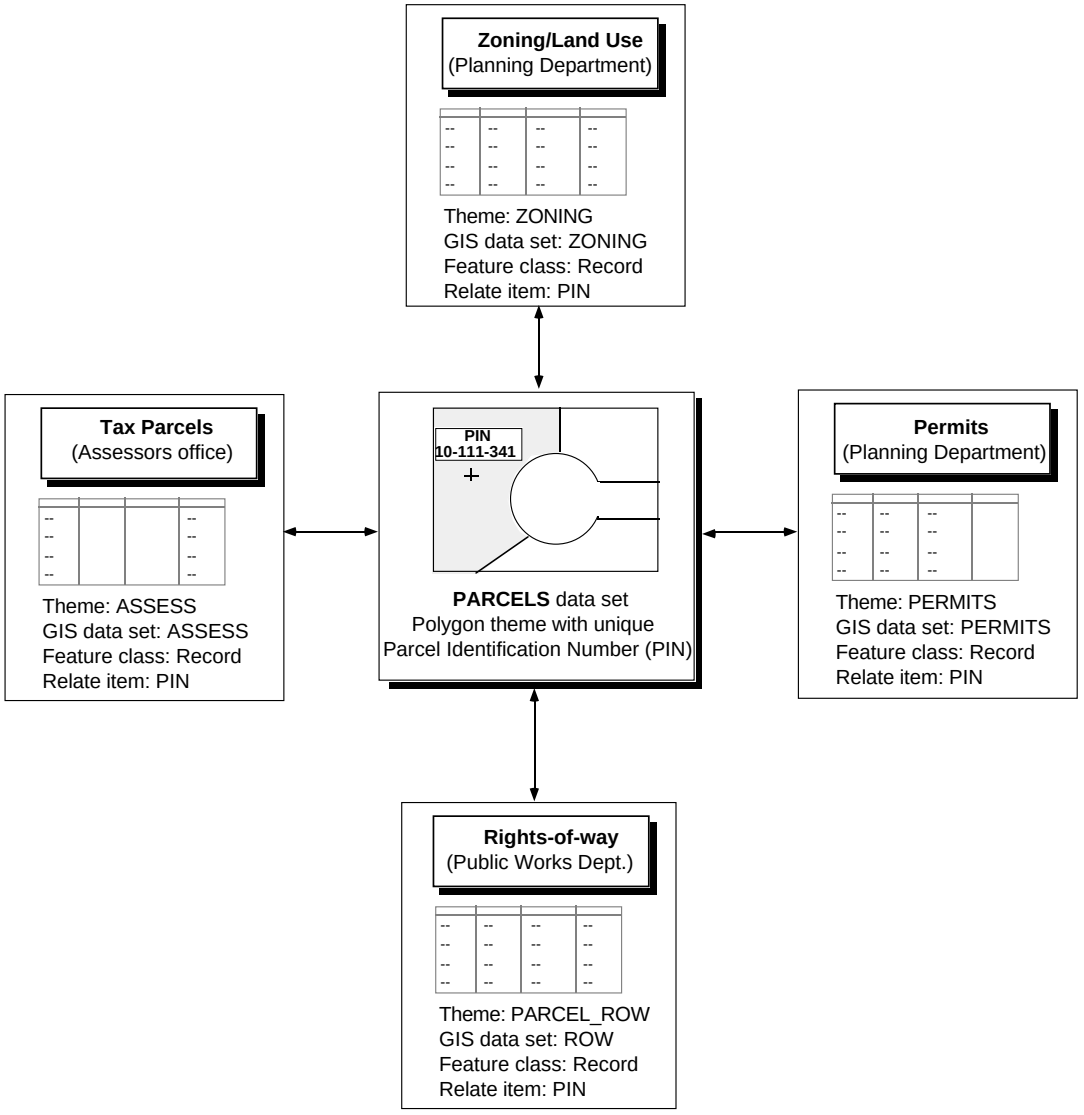
GIS data set: JUNCTION
Feature class: Record (dBASE)
Drawing: STREETS
Entities: none
Relate item: INT_TYPE
Item definitions: INT_TYPE,2,N,0
LABEL,20,C
SYMBOL,3,N,0

**Contents of table named JUNCTION**

| INT_TYPE | LABEL | SYMBOL |
|---|---|---|
| 1 | Street intersection | 46 |
| 2 | Overpass | 88 |
| 3 | Railroad crossing | 34 |
| 4 | Bike path crossing | 77 |
| 5 | Bridge over railroad | 87 |
| 6 | Bridge over stream | 87 |
| 99 | Not specified | 0 |

## Theme: Parcels

It is important that you organize parcel data in such a way that you can use the same database for different parcel-based applications. For example, you can use a single parcel coverage to store the coordinate and feature attribute data along with the parcel identification number (PIN). A variety of tabular data sets can be stored in separate record themes. These record themes can then be used to relate to the parcel theme when required.

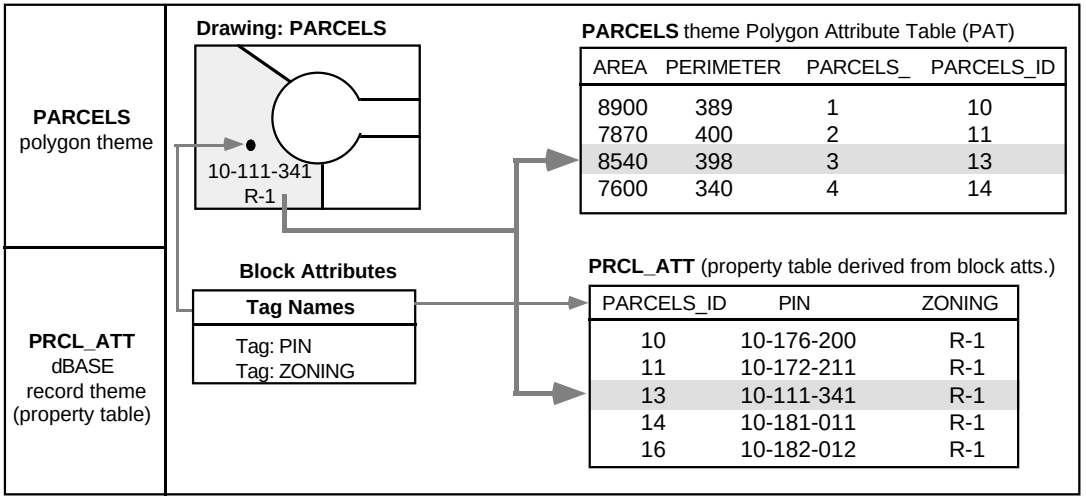



The Assessor's Office stores attributes such as individual owner's address and assessed property values for tax purposes. The City Planning Department, on the other hand, maintains the building and demolition permits, the building permit types, and the current status of the permits. The Planning Department also manages zoning and land use information (e.g., new zoning cases initiated in different parts of the city based upon the General Plan). The Public Works

Department manages the information such as rights-of-way, easements for utilities, sidewalks, and other public facilities. These facilities generally fall along the boundaries of each parcel.

Given these scenarios, it is appropriate to design the PARCELS theme such that it can be used by different departments. These individual departments can access the PARCELS theme and relate individual department's attributes using the parcel identification number. Some of these applications are illustrated here to help you understand how to organize your parcel theme.

The following figure illustrates how the AutoCAD entities (including block attributes) are used to represent the components of a parcel theme.

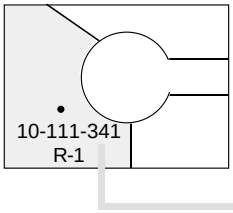



There is always one record in the feature attribute table that corresponds to each feature in the theme. Both the spatial information used to define each parcel and the corresponding record in the polygon attribute table (PAT) contain the feature number so that a one-to-one correspondence is maintained between the parcel and its record in the PAT file.

***Note:*** AutoCAD block attribute values or a text entity within a parcel can be captured into a property table using the **addfeat** command. See the **addfeat** command reference for further details.

In the above example, you have created two separate themes: a polygon theme named PARCELS and a record theme PRCL_ATT (a property table) containing AutoCAD block attributes that are associated with individual parcels in the drawing database. Use the **joinitem** command to physically join these two

themes to append the property table to the standard Polygon Attribute Table as shown below:




**PARCELS** theme Polygon Attribute Table (PAT)

| AREA | PERIMETER | PARCELS_ | PARCELS_ID | PIN | ZONING |
|---|---|---|---|---|---|
| 8900 | 389 | 1 | 10 | 10-176-200 | R-1 |
| 7870 | 400 | 2 | 11 | 10-172-211 | R-1 |
| 8540 | 398 | 3 | 13 | 10-111-341 | R-1 |
| 7600 | 340 | 4 | 14 | 10-181-011 | R-1 |
| 7980 | 365 | 5 | 16 | 10-182-012 | R-1 |

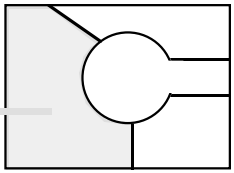
The parcel identification number (PIN) can be used to relate to other record themes maintained by different departments.

It is also worth noting that the AutoCAD drawing is used to store the accurate survey information (traverses, curves, etc.) for parcels while the ArcCAD theme is used to represent parcel polygons and to associate related attributes. Each system would be used for appropriate applications of this integrated data model.

## Example: Tax parcels

Tax parcels within the county are used for numerous applications. Municipal revenue is collected for each parcel within the county by the Tax Assessor's Office. These applications are typically run independently of the GIS. However, the parcel identification number (PIN) in the PARCELS theme can be used to relate these tabular databases to individual parcels. Thus, the role of the polygon attribute table of the PARCELS theme is to act as a relate table between theme features and additional attributes that are specifically designed for various applications.

**PARCELS**



**PARCELS** theme Polygon Attribute Table (PAT)

| AREA | PERIMETER | PARCELS_ | PARCELS_ID | PIN | ZONING |
|---|---|---|---|---|---|
| 8900 | 389 | 1 | 10 | 10-176-200 | R-1 |
| 7870 | 400 | 2 | 11 | 10-172-211 | R-1 |
| 8540 | 398 | 3 | 13 | 10-111-341 | R-1 |
| 7600 | 340 | 4 | 14 | 10-181-011 | R-1 |
| 7980 | 365 | 5 | 16 | 10-182-012 | R-1 |

**OWNERS** (dBASE record theme)

| PIN | OWNER | MAIL_ADD |
|---|---|---|
| 10-176-200 | Smith, M | 121 W Main |
| 10-172-211 | Jones, C | 125 W Main |
| 10-111-341 | Brown, D | 129 W Main |
| 10-181-011 | George, B | 120 W Main |
| 10-182-012 | West, R | 124 W Main |

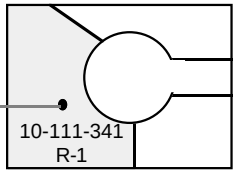
**ASSESS** (SQL record theme)

| PIN | LEGAL_AREA | VALUE | ZONING |
|---|---|---|---|
| 10-176-200 | 8912 | 67,000 | R-1 |
| 10-172-211 | 7871 | 64,500 | R-1 |
| 10-111-341 | 8540 | 73,000 | R-1 |
| 10-181-011 | 7610 | 71,900 | R-1 |
| 10-182-012 | 7985 | 81,250 | R-1 |

In the above example, the polygon attribute table of the PARCELS theme and two record themes containing information about owners and property assessment can be related to access appropriate information for each parcel. There is one record in the OWNERS record theme for each owner of each parcel in the city. This record theme is used by the Assessor's Office to mail tax bills to individual owners.

The second record theme, ASSESS, contains assessment information such as appraised values for land, improvements, exemptions, and so on. Tax rates are applied to these values to calculate taxes due each year.

**Theme: PARCELS**




GIS data set: PARCELS
Feature class: Polygon
Feature Attribute Table: PAT

Drawing: PARCELS

Entities: line, pline, arc, circle, trace, 3Dface, etc.

**Theme: PARCEL_ATTRIBUTES**

| | | |
|---|---|---|
| -- | -- | -- |
| -- | -- | -- |

GIS data set: PRCL_ATT
Feature class: Record
Drawing: PARCELS
Entities: blocks with attributes

TAG: PIN
TAG: ZONING

**Theme: PARCEL_OWNER**

| | | |
|---|---|---|
| -- | -- | -- |
| -- | -- | -- |

GIS data set: OWNER
Feature class: Record
Relate item: PIN
Drawing: PARCELS
Entities: none

**Theme: PARCEL_ASSESS**

| | | |
|---|---|---|
| -- | -- | -- |
| -- | -- | -- |

GIS data set: ASSESS
Feature class: Record (SQL)
Drawing: PARCELS
Entities: none

**PARCELS** theme Polygon Attribute Table (PAT)

| Items | Data Type | Description |
|---|---|---|
| AREA | 13,N,6 | Area in sq. feet calculated from State Plane coordinates |
| PERIMETER | 13,N,6 | Perimeter in feet |
| PARCELS_ | 11,N,0 | Polygon sequence number |
| PARCELS_ID | 11,N,0 | Polygon ID |

**PRCL_ATT** (property table, initially defined as a dBASE record theme, then joined with PARCELS theme's PAT)

| Items | Data Type | Description |
|---|---|---|
| PARCELS_ID | 11,N,0 | Polygon's User-ID |
| PIN | 11,C | Tax Assessor number for each parcel (e.g., 10-111-341). Contains Assessor's book, page, and lot number. |
| ZONING | 3,C | Zoning code |

**OWNER** (dBASE record theme)

| Item | Data type | Description |
|---|---|---|
| PIN | 11,C | Assessor number for parcel |
| OWNER_NAME | 36,C | Last-name first |
| STREET_NO | 7,C | Mailing address number |
| STREET_DIR | 1,C | Street direction |
| STREET_NAME | 20,C | Street name |
| CITY | 20,C | City name |
| STATE | 2,C | Two letter state code |
| ZIP_CODE | 5,N,0 | ZIP code number |

**ASSESS** (SQL record theme)

| Item | Data type | Description |
|---|---|---|
| PIN | 11,C | Assessor number for parcel |
| LAND_VALUE | 13,N,2 | Appraised land value |
| IMP_VALUE | 13,N,2 | Appraised improvement value |
| PROP_VALUE | 13,N,2 | Appraised property value |
| EXEM_VALUE | 13,N,2 | Exemptions |
| LEGAL_AREA | 13,N,6 | Legal area from survey |

Chapter 3

# Database automation

As a Geographic Design System, ArcCAD™ brings powerful GIS database features and operators to the AutoCAD environment. ArcCAD extends your AutoCAD® environment by providing tools that facilitate geographic data automation, coordinate (locational) and attribute (thematic) data management and manipulation, map generation and data analysis. The following chapters will help you use some of ArcCAD software's major features to meet these needs.

## Creating a theme

Themes define collections of geographic features modeled as relationships between AutoCAD drawing entities and GIS data sets. This data model implies that there is a number of important aspects and considerations involved in the creation and maintenance of themes and ArcCAD databases: digitizing coordinates, creating themes, creating theme topology, creating entity-feature links, building feature attribute tables and connecting feature attributes to a corresponding theme feature.

The next section identifies the steps used to create ArcCAD themes and provides guidance on how to perform many of these steps. It is not intended to be an absolute, step-by-step procedure on how to automate your geographic information. Rather, it will help guide your use of ArcCAD tools in preparing your own data automation procedures.

### Data automation steps

One of the powerful features of ArcCAD is its ability to integrate data from two sources. You can begin building themes using either AutoCAD drawings or ARC/INFO® coverages, or you can build your database from scratch using AutoCAD design tools and ArcCAD data automation tools.

Different data automation procedures apply to each source of data. Even existing drawings need slightly different handling procedures as opposed to newly created drawings when used to create themes. The following section contains important information that should be read before attempting to build

themes regardless of the source of the data. Procedures used to create themes from each data source will also be discussed in detail in the following sections.

## Creating themes from drawings

Any AutoCAD drawing can be used to create a theme. The drawing entities used to create the theme features remain as the graphical representation of the features. Subsequent GIS operations are performed on theme features. Themes are used for GIS operations, and drawings are used to view, query and edit the theme data.

There are seven basic steps used to create themes from drawings with ArcCAD. They are:

**Step 1.** Define themes.

**Step 2.** Create features.

**Step 3.** Build feature topology.

**Step 4.** Identify and correct coordinate errors.

**Step 5.** Identify and correct topology errors.

**Step 6.** Assign attributes to features.

**Step 7.** Identify and correct attribute data errors.

AutoCAD data-input methods are used to create your drawing. Your drawing may be created by digitizing a manuscript on your digitizer or you can use other tools to create the drawing. Coordinates can be entered manually in AutoCAD using the keyboard or the mouse. Refer to the *AutoCAD User's Guide* for information on entering coordinates using the keyboard or mouse. Scanners can also be used to scan maps into a format that AutoCAD can read. Consult your scanner's documentation for information about scanning maps into AutoCAD. The method you use to create your drawings should be one that is most appropriate for the type of data you are using.

When creating drawings, the following considerations are important for building topologically sound and error-free themes.

- Ensure that all drafted features meet your specific mapping standards.

- When creating polygon features for a polygon theme, all polygons should contain a label point with a User-ID. Ensure that all features are assigned unique User-IDs. All lines should connect correctly, and all polygons should close.

■ Entities in your drawing can be placed on any layer and can have any AutoCAD property. Since entity properties can be transferred as attributes to the GIS data set when features are created (Step 2), you can organize your drawing to simplify later attribute coding. For example, if you were creating water lines, you could place all water lines with similar attributes (e.g., diameter, material, and so on) on the same layer whose name is indicative of the value of the attribute.

The seven steps used to create themes from AutoCAD drawings are discussed in more detail in the following section.

#### Step 1 Define themes

Themes are created using the **defthm** command. The **defthm** command requires the following information:

| | |
|---|---|
| Theme | - the name of the theme to create. |
| Feature class | - the type of features this theme will contain. Possible feature classes are point, line, polygon, annotation, tic, image, and record. |
| GIS data set | - the name of the coverage that will store this theme data. |
| Symbol | - a default symbol number used to draw theme features. |

For more information about defining themes, refer to the **defthm** command reference.

#### Step 2 Create features

This step will create features from the entities in the drawing.

The first step in creating features is to establish the link mode. The current setting of link mode (established with the **link** command) is important when creating features from entities. The recommended setting when creating features is to set link mode to 2. This setting establishes a one-to-one correspondence between entities and their features allowing only one feature to be linked to one entity. This allows you to delete and modify theme features (using **delfeat** and **modfeat** or **savefeat**, respectively) later on if required. (Please refer to the section 'Entity-feature links and editing' later in this chapter for more information on entity-feature links and the use of the **link** command.)

After the link mode has been established, features that correspond to your drawing entities can be created using the **addfeat** command. The **addfeat** command will prompt for the following:

| | |
|---|---|
| Theme name | - the name of the theme to add features to. |
| Select objects | - select a set of drawing entities that will be made into features. AutoCAD selection tools along with the |

ArcCAD **xselect** command can be very useful in creating selection sets and filtering AutoCAD entities by property to simplify feature creation. Refer to the following table for a list of entities and their resulting features.

Optional property table - give the name of a property table. The entity properties and block attributes of entities in the selected set can optionally be saved to this property table. Refer to the following section, 'Saving entity properties and block attributes', for information about using property tables.

**Addfeat** will process each entity in the selected set and create a feature from it. The entities selected must be consistent with the feature class of the theme. If certain entity types inappropriate for a theme feature class are selected, **addfeat** (**savefeat** and **modfeat** also) will filter them out (e.g., if text entities are in the selection set when arc features are being created, the text entities will be filtered out).

When creating features from entities, the following table indicates the relationship between entity types and the resulting feature classes:

| AutoCAD entity types | ArcCAD feature classes |
|---|---|
| **Shape**<br>**Point**<br>**Insert**<br><br><br><br><br>**Attdef**<br>**Text** | **Point**<br><br>Since hatch patterns are a special case of insert, avoid including hatches in your selection set when using the feature creation commands or you will create point features at the hatch's insertion point of 0,0. Block attributes can be saved to a property table. |
| **3Dface**<br>**Solid**<br>**Line**<br>**Polyline**<br><br><br><br><br>**Arc**<br><br><br><br><br>**Circle**<br>**Trace** | **Line**<br><br>Results in a two point arc feature.<br>Results in multiple arc features if the polyline contains more than 500 vertices. If the polyline contains a bulge, that bulge is added as a separate feature just as a circle or arc entity would be.<br>Converted to an arc feature that approximates the curved entity by dividing the curve into straight line segments and placing one vertex per degree of arc. Entity properties can be saved to a property table to recreate true curves and arcs.<br>Same as arc.<br>Creates an arc feature that follows the centerline of the trace. |
| **Point, shape, attdef, insert and text**<br><br>**Line, polyline, circle, arc, solid, trace and 3Dface** | **Polygon label points**<br><br>**Polygon boundaries (arc features)** |
| **Text** | **Annotation** |
| **none** | **Tic** |
| **none** | **Record** |
| **none** | **Image** |

*Entity to feature conversion table*

Note that ArcCAD does not support the following entity types: 3dsolid, body, ellipse, leader, mline, mtext, ray, region, spline, tolerance and xline. If **addfeat**, **savefeat** or **modfeat** encounter any of these entity types, they are skipped.

Creating features using **addfeat** writes points, lines, polygon and annotation coordinate information to a GIS data set (a coverage) of the specified theme. It does not create topology for point, line or polygon features. This is done by running **build** or **clean**. Topology does not need to be created for annotation features.

### Saving entity properties and block attributes

Most AutoCAD entity properties and block attributes can be saved to a record theme as features are created. This record theme is known as a *property table.* The **addfeat**, **savefeat** and **modfeat** commands can write the following entity properties into a property table that can then be related back to the theme's feature attribute table using the **relate** or **joinitem** commands:

- Color: The AutoCAD color number for this entity. Color is written to an item named *acad_color*.
- Layer name: The layer name that the entity is on. Layer name is written to an item named *acad_layer*.
- Linetype: The linetype of the entity. Linetype is written to an item named *acad_ltype*.
- Elevation: The elevation of the entity. Elevation is written to an item named *acad_elev*.
- Thickness: The thickness of the entity. Thickness is written to an item named *acad_thick*.
- Entity type: A character string describing the entity type. Entity type is written to an item named *acad_etype*.
- Entity handle: A character string containing the entity handle. Entity handle is written to an item named *acad_hand*.
- Curve information: A number indicating whether the entity is a true curve. A '1' indicates a true curve, a '0' indicates this entity does not have curve information. Curve information is written to an item named *acad_curve*.
- Text: The text of a text entity. Text is written to an item named *acad_text*.
- Block: The block name of an insert. The block name is written to an item named *acad_block*.
- Angle: The angle value of the entity. Angle is written to an item named *acad_angle*.

Any or all of the following corresponding items can exist in the property table and are used to save entity properties:

| Item name | Item width | Item type | Number of decimal places |
|---|---|---|---|
| *cover*_id | 11 | n | 0 |
| acad_color | 3 | n | 0 |
| acad_layer | 31 | c | - |
| acad_ltype | 16 | c | - |
| acad_elev | 13 | n | 6 |
| acad_thick | 13 | n | 6 |
| acad_etype | 10 | c | - |
| acad_hand | 16 | c | - |
| acad_curve | 1 | n | 0 |
| acad_text | 254 | c | - |
| acad_block | 32 | c | - |
| acad_angle | 13 | n | 9 |

If the property table contains the item called *COVER*_ID *(COVER* is the name of this theme's coverage) this item will contain the User-ID of the corresponding feature after feature creation.

During the process of adding features, the **addfeat** command will ask for an optional property table. If the theme name entered at this prompt does not exist, ArcCAD will prompt to create it. If 'yes' is entered, ArcCAD will create a property table with the name entered at the Optional Property Table prompt. The theme will contain all the AutoCAD entity property items listed in the previous section. Values are added for those features by the **addfeat** command.

A property table template called PROPERTY is located in the ARCAD\TEMPLATE subdirectory. This empty record theme contains all the above item definitions for entity properties. You can use **copythm** or the DOS COPY command to copy this template file to your current directory, define a record theme for it and then write entity properties to that property table when using the feature creation commands. Alternatively, you can create your own property table (or copy the supplied template and use **dropitem** to remove any items you do not need) that has a subset of items for saving entity properties.

If the drawing contains blocks that have attributes, the block attributes can also be saved to the property table. The property table must contain defined items that correspond to the block tag names. The item definitions must be appropriate for the type of information stored in that block. As features are created from blocks, any blocks with attributes will have their attributes written to the property table. Appropriate items must exist in the property table before block attributes can be written to the table. The **additem** command can be used to add items with appropriate block tag names to a copy of the supplied property table template (PROPERTY). This way, you can save both entity properties and block attributes to the same record theme.

***Note:*** ArcCAD item names are limited to 10 characters. Since attribute tag names can be longer than 10 characters, only the first 10 characters of a tag name are used to match item names in a property table. Attribute tag names should therefore be unique within the first 10 characters.

Entity property information and block attributes are written to the property table in the same order as features are written to the theme. Therefore, to link properties and block attributes with features through a feature attribute table, you can use **relate** or **joinitem** with the User-ID as the relate item. The User-ID of the theme's coverage must have been added to the property table in order to relate the property table to the theme's feature attribute table. Refer to the **relate** and **joinitem** command referenced, as well as Step 7, for more information on merging attributes to features.

#### Step 3 Build topology

To build features for point, line and polygon themes, the ArcCAD **build**, **clean** and **renode** commands are used to create feature topology and feature attribute tables.

Point themes are built using the **build** command. This creates a Point Attribute Table (PAT) with one record per feature.

To build arc features in a line theme, arc-node topology and an Arc Attribute Table (AAT) need to be created. Use **renode** to generate arc-node topology. Then use **build** to create the AAT.

To build polygon features in a polygon theme, either **clean** or **build** can be used; however, knowing when to use **build** or **clean** can be confusing. Both **build** and **clean** perform many of the same functions by defining feature topology and creating feature attribute tables. Here is a quick description of some of their important differences.

- **Clean** performs coordinate analysis on arc feature coordinates to split overlapping arc features, calculate new intersections, remove extremely small sliver arc features (using the fuzzy tolerance), and remove dangling arc features shorter than the dangle length. This means that **clean** does not execute as fast as **build** does. It also means that **clean** may move arc feature coordinates, whereas **build** does not. **Clean** will always create or update arc-node topology. **Clean** manipulates coordinates based on specified tolerance values. Refer to the **clean** command reference for guidance on setting and using tolerance values.

■ **Build** does not process overlapping arc features when creating polygons; **clean** does. For line themes, **build** will not split the overlapping arc features at their intersections, but will still build an AAT for the theme (this capability may be important to implement geographic features such as overpasses). For polygon themes, **build** will stop execution if overlapping arc features are encountered in the theme.

■ The **renode** command can be used to update arc-node topology after building a line theme with the **build** command.

| Capabilities | BUILD | CLEAN |
|---|---|---|
| Processes: | | |
| Polygons | Yes | Yes |
| Lines | Yes | Yes |
| Points | Yes | No |
| Numbers features | Partially † | Yes |
| Calculates spatial measurements | Yes | Yes |
| Creates intersections | No | Yes |
| Processing speed | Faster | Slower |

*Clean vs. build*

† BUILD will renumber arc features and polygons, but will not renumber nodes.

**Build** or **clean** creates a feature attribute table for the theme with the following standard set of items:

**Items included in a theme PAT:**

**Cover** is the coverage name. **Cover**_ refers to the Internal-ID number of each polygon or point; **Cover_ID** refers to the User-ID. When the PAT is used for point attributes in a point theme, AREA and PERIMETER values are zero for each point.

| Item name | Description |
|---|---|
| AREA | Area of polygon. |
| PERIMETER | Perimeter of polygon. |
| **Cover**_ | Polygon Internal-ID number. |
| **Cover_ID** | Polygon User-ID. |

**Items included in a theme AAT:**

**Cover** is the coverage name. **Cover**_ refers to the Internal-ID number of each arc feature; **Cover_ID** refers to each arc feature's User-ID.

| Item name | Description |
|---|---|
| FNODE_ | Internal-ID number of the from-node. |
| TNODE_ | Internal-ID number of the to-node. |
| LPOLY_ | Left polygon Internal-ID number. |
| RPOLY_ | Right polygon Internal-ID number. |
| LENGTH | Length of arc feature. |
| **Cover**_ | Arc feature Internal-ID number. |
| **Cover_ID** | Arc feature User-ID. |

*Default feature attribute table items generated when topology is initially built.*

#### Step 4 Identify and correct coordinate errors

This step is used to verify how accurately arc features, polygon boundaries and label point positions were created. If the drawing was digitized from a manuscript, a verification plot should be made at the exact scale of the original map sheet. The AutoCAD **plot** command can be used to quickly and easily create a verification plot from which you can identify digitizing errors. Use AutoCAD editing commands to correct the digitizing errors you discovered. Compare the map sheet and verification plot visually, and check the following:

- Were arc features accurately traced during digitizing (do the maps overlay well on top of each other)?

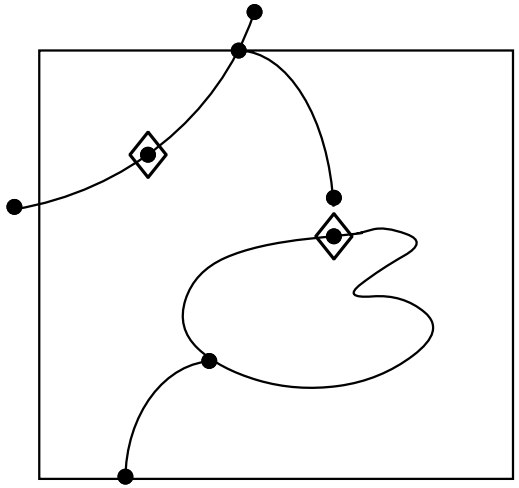
- Are any arc features or label points missing?

- Do arc feature endpoints (nodes) match correctly or are dangling nodes present where arc features should meet?

When identifying digitizing errors in line or polygon features, the following node errors may be present:

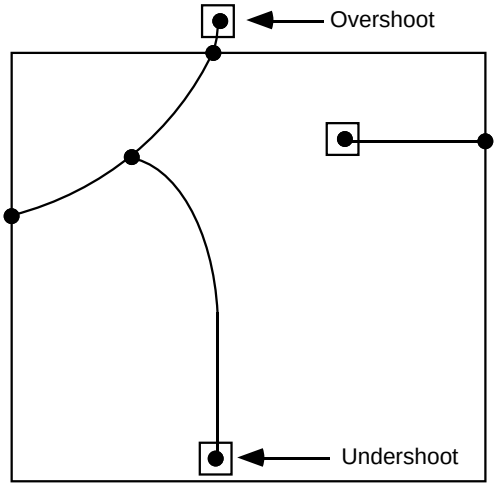
**Pseudo nodes** A pseudo node is one at which two, and only two, arc features intersect (or a single arc feature connects with itself). Another way to think of pseudo nodes is that they identify locations where an otherwise contiguous arc feature is actually 'split' into smaller discrete arc features. Pseudo nodes do not necessarily indicate an error or problem. In fact, some pseudo nodes may be desired—especially if you have an Arc Attribute Table (AAT) that contains different attributes about the two arc features that connect at the pseudo node. However, if two arcs 'split' by a pseudo node were intended to represent a contiguous line feature (e.g., a county boundary), then it may be necessary to remove the features and reenter them as one. Alternatively, you can use the **pedit** command to join together two linear entities and create one.

Other pseudo nodes which are not errors are nodes that separate arc features created from polylines that have more than 500 vertices.



*Pseudo nodes can be displayed using the **noderror** command. Pseudo nodes are always drawn using a diamond symbol.*

**Dangling nodes** A dangling node is the unconnected node of a dangling arc feature. This usually identifies that a polygon does not close properly (undershoot), arc features that do not connect properly, or an arc feature that was digitized past its intersection with another arc feature (overshoot).




*Dangling nodes can be displayed using the **noderror** command. Dangling nodes are always drawn with a square box symbol.*

A dangling node may be intentional. For example, in a street centerline map, cul-de-sacs are often represented by dangling arc features.

***Important:*** As all errors are fixed by editing the drawing, the edits made to the drawing must be written back to the coverage as features using the **savefeat** or **modfeat** command.

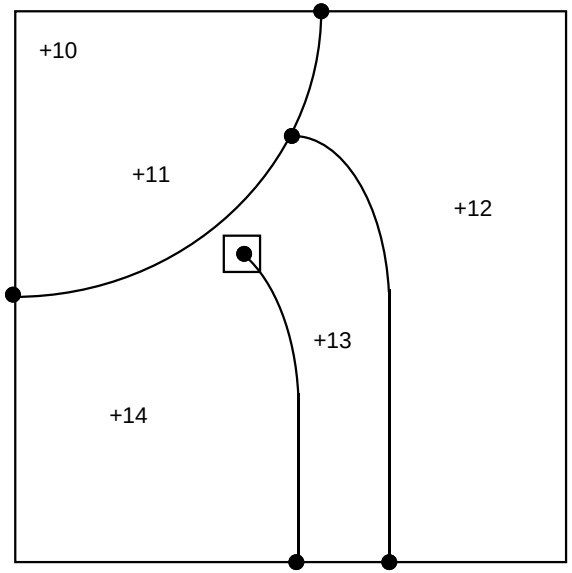
#### Step 5 Identify and correct topology errors

Since the existing drawings used to create ArcCAD databases may contain many different entity types, it is possible that features may be written to the theme that creates topological errors. For example, when creating a polygon theme, if several AutoCAD entities that could be made into point features (refer to the table in Step 3) were selected in **addfeat**, all these entities would create label points in the polygon theme resulting in polygon features with multiple label points. Topological errors like this must be identified and corrected. Then topology can be recreated using **clean** or **build**.

Once feature topology is created, two potential error types can be evaluated. Node errors can be plotted to identify unclosed polygons and unconnected lines

(undershoots and overshoots). This is described in Step 3 above. Also, potential label point errors in polygon themes can be identified: polygons with more than one label point and polygons that don't have a label point.

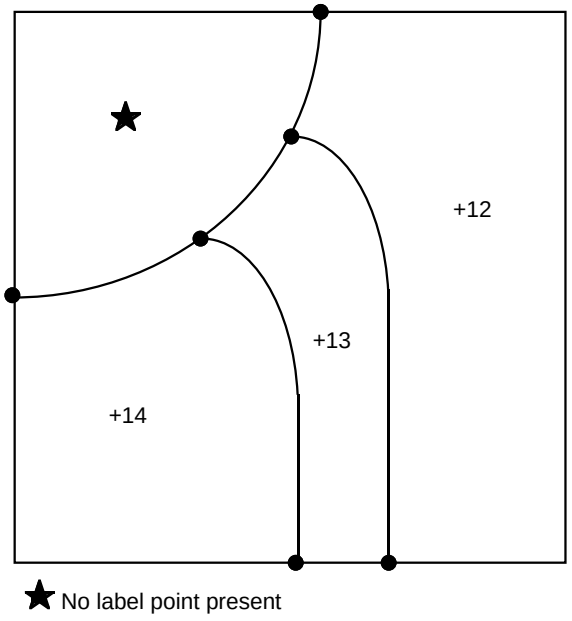
**More than one label point in a polygon** Polygons are assigned User-IDs by locating a single label point within each polygon. The label point's User-ID becomes the polygon's User-ID. If a polygon contains more than one label point, there is no rule for knowing which label point User-ID will be assigned to the polygon (unless both label points have the same User-ID). Multiple labels may also indicate unclosed polygons (e.g., label points 13 and 14 in the figure below).




*Example of too many label points per polygon*

Label errors can be displayed using the **laberror** command. You must use **clean** or **build** to construct polygon topology before you can identify potential label errors.

**No label points in a polygon** A polygon that does not contain a label point will be given a User-ID of 0, and unless a polygon contains a label point, that polygon's User-ID can never be changed from 0. PAT attributes cannot be maintained for that polygon because **clean** and **build** use the User-ID to update the PAT.




*Example of no label point in a polygon*

Polygons without label points can also be displayed using the **laberror** command. A star symbol is placed inside of any polygon that does not have a label. You must use **clean** or **build** to create polygon topology before you can identify potential label errors.

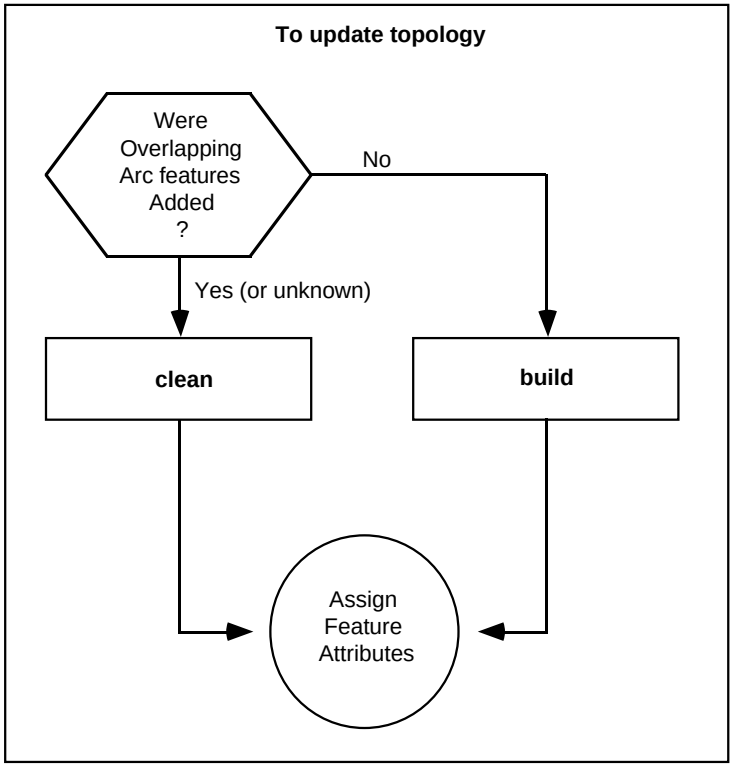
When errors are discovered, use the appropriate AutoCAD commands to correct the problems. Be sure to write the drawing changes back to the coverage using the ArcCAD **savefeat** or **modfeat** commands.

***Note:*** In order to use AutoCAD commands to edit a theme's coverage through the drawing, you must have created the features when link mode was set to 1 or 2. These **link** settings establish a one-to-one correspondence between a coverage feature and an entity. Refer to the **link** command reference for more information about link mode settings.

Often, the only way errors can be corrected is to remove some theme features, recreate entities correctly and then regenerate the theme features. This can be done by using the following steps:

1) Use the **delfeat** command to delete the feature.
2) Use the AutoCAD **erase** command to erase the feature's corresponding entity.
3) Recreate the entity using AutoCAD tools.
4) Recreate the feature using the **addfeat** command.
5) In most cases, feature topology must also be updated after such edits are made to a theme. If only label points are added, moved or deleted, you can use **build**. If arc features are added, moved or deleted, creating intersections with other arc features or new polygons, you should use **clean**. (If you used **createlab**, it may be desirable to use **idedit** instead of **build**. Refer to the **idedit** command reference.)

If arc features are edited to correct topology errors, you must decide whether to use **clean** or **build** to update topology. The following flowchart will help guide you.



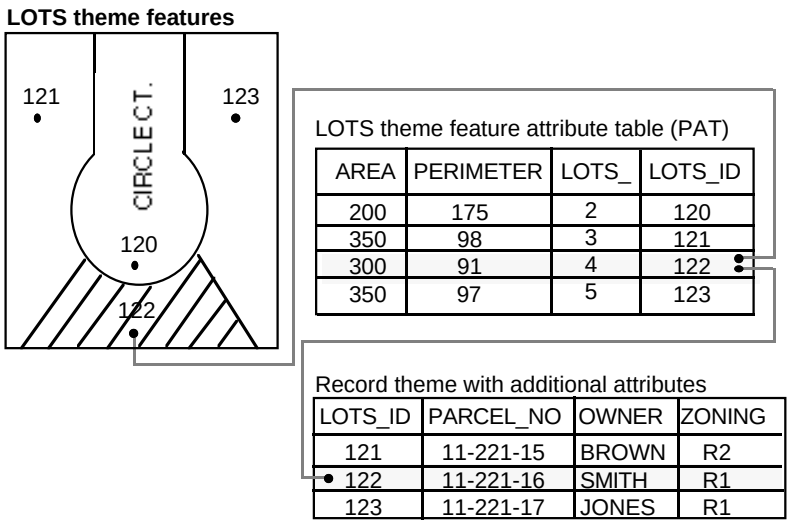
**Step 6 Assign attributes to features**

Additional attributes can be added to the feature attribute table or stored in related tables. Typically, a few key items are added to the PAT or AAT that summarize features or are used to link to attributes in related tables.

To add user-defined items directly to a theme's feature attribute table, use the **additem** command. You can then use the ArcCAD commands **modrec** or **modone** to add values to these new fields. Please refer to the section 'Managing attributes' for more information.

To store attributes in a related table, create a record theme. This record theme must have a common item type with an item in the feature attribute table of the theme its attributes relate to. The items that are related do not have to be defined exactly the same. The item name and width may be different, but types should be the same. Refer to the section 'Creating a record theme' for information about creating record themes. The record theme must then be related to the PAT or AAT using the common item. The relate between the record theme and the feature attribute table is accomplished using the **relate** or **joinitem** command. **Relate** establishes a temporary connection between each feature and its description in the record theme. **Joinitem** establishes a permanent connection by physically joining the record theme items to the feature attribute table.

The following example shows how additional attributes in a record theme can be linked to a feature attribute table:




*Linking records by a common item*

The records in the feature attribute table and the record theme can be 'linked' because a common item (LOTS_ID) is present in both. Most often, the User-ID is used to 'link' or 'merge' attribute information from a record theme with theme features. For example, **joinitem** could be used on the tables above to permanently merge the PAT with the record theme. This would create a new PAT that would look like this:

| AREA | PERIMETER | LOTS_ | LOTS_ID | PARCEL_NO | OWNER | ZONING |
|---|---|---|---|---|---|---|
| 200 | 175 | 2 | 120 | — | — | — |
| 350 | 98 | 3 | 121 | 111-221-15 | Brown | R2 |
| 300 | 91 | 4 | 122 | 111-221-16 | Smith | R1 |
| 350 | 97 | 5 | 123 | 111-221-17 | Jones | R1 |

In this example, the feature User-ID (LOTS_ID) is the key item used to establish this connection. Unique User-IDs will ensure a one-to-one connection. In other words, a record of descriptive data will usually be matched to one, and only one, feature. Only if features share all of the same set of attributes should they have the same User-ID.

A related record theme can also be temporarily linked to a feature attribute table so that those attributes can be accessed only when they are needed. For example, more specific information about each parcel in the previous example could be kept in a related theme. The parcel number (PARCEL_NO) could then be used to relate these data records with their associated PAT records:

| PARCEL_NO | STRUCTURES | VALUE | DUE | STATUS |
|---|---|---|---|---|
| 11-221-15 | Duplex, 3 Br | 151000 | 12-1 | Paid 11-3 |
| 11-221-16 | Single Fam, 4 Br | 121000 | 6-3 | Due |
| 11-221-17 | Single Fam, 3 Br | 115750 | 9-2 | Paid 8-16 |

The relationship between the PAT and this record theme through the common item PARCEL_NO could be established using the **relate** command. This allows access to this additional information only when required.

The User-ID for each feature is often used as the common item used to relate additional attributes to theme features. As features are created or modified using **addfeat**, **savefeat** or **modfeat**, User-IDs (Cover_ID) are sequentially assigned to each feature based on the current value of the User-ID of the theme (refer to the **modthm** command reference for more information). You do not have control over which entities in the selection set are assigned a particular User-ID. Two methods can be used to assign a User-ID to a particular feature.

- Modify the User-IDs after features have been created:

  1) Use the **modone** command to interactively select the features whose User-IDs you wish to modify.
  2) Run **idedit** to write the new User-IDs to the theme's GIS data set.

- As each map feature is created, set the User-ID base using the **user-id** command and then run **addfeat** to create the feature. The feature will be created with the specified User-ID. Repeat this procedure for all map features.

If entity properties and block attributes were saved to a property table when features were created, you can join these attributes to the theme's feature attribute table using the **joinitem** command. If the User-ID (Cover_ID) of each feature was written to the property table, use the linear relate option for **joinitem** and the User-ID as the relate item. Alternatively, you can establish a temporary relate between the feature attribute table and the property table using the **relate** command when you wish to access these attributes.

***Note:*** ArcCAD will automatically maintain the feature attribute table if theme features are modified or updated. However, the relational join operation that occurs during a **clean** or **build** may result in lost attribute data if the User-IDs for theme features are not unique. This is because the join process looks for the first occurrence of a record in the related data file with a matching User-ID. The attribute values of the matching record from the old PAT are used to update the attributes of the records in the new PAT. Thus, records with duplicated User-IDs will always match with the first occurrence of the User-ID in the old PAT file and will be updated with the attribute values of that record. This may be an undesirable effect, potentially resulting in lost attribute information.

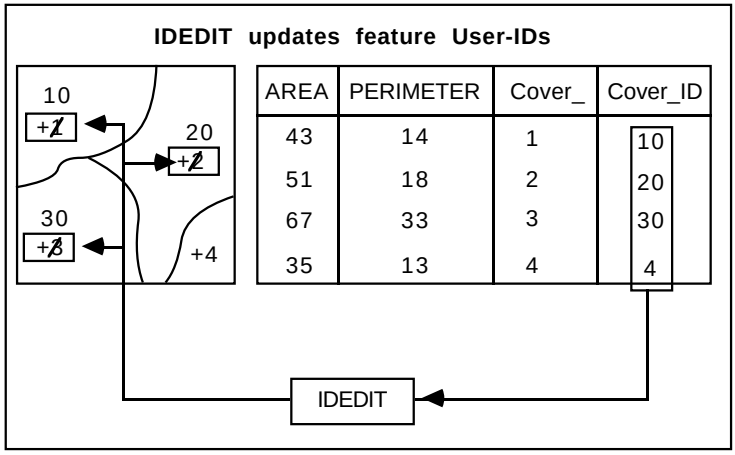
#### Step 7 Identify and correct attribute data errors

The process of verifying feature attribute data must occur at three levels. First, you must check for errors that were made while entering the attribute values (e.g., was a '1' entered instead of a '2'?). Second, the correct values must have been assigned to each feature during map interpretation and compilation (e.g., does a polygon whose Land Cover code is 'LAKE' also have a Slope of 0?). Third, the attributes must have been associated with the correct theme feature. Maintaining unique User-IDs for each feature will help minimize this last type of error.

The typographic error is the easiest to identify and solve, and many approaches can be taken. For example, each attribute table could be entered twice. Then, the two tables could be compared against each other to identify and correct typographic errors. Once these types of errors are corrected, you should check for incorrectly assigned and illogical attribute values.

Once attribute values are corrected, a number of verification plots can be produced using the AutoCAD **plot** command to check coverage feature code assignments against those coded on the manuscript.

Assuring that feature attributes are associated with the correct feature involves confirming that the same User-ID value for each feature is stored with the feature geometry and in its feature attribute records. User-IDs for features are stored in two places in a theme's coverage: once in the theme's feature attribute table as the User-ID item and again in the locational data which define arcs, label points and polygons. Updating feature User-IDs in the feature attribute table can be performed by using **modone**, **calculate**, **modrec** or **wbrowse**. After User-ID values are updated, use the **idedit** command to copy the new User-ID values in the feature attribute table to the User-ID values stored with each feature's locational data in the coverage. Care should be taken to ensure **idedit** is used on a theme whenever a feature User-ID is updated in the PAT or AAT.




Only if User-ID values are changed in the feature attribute table will you need to update the locational information in a coverage using **idedit**. Edits or updates to other attribute information do not require coverage topology adjustment through **build** or **clean**.

***Caution:*** Never alter the Internal-ID attribute in the PAT or AAT files of a coverage. These are the internal feature identifiers maintained by ArcCAD.

Sometimes, code verification within ArcCAD isn't necessary—especially when existing data sets are used with ArcCAD. An example might be accessing a county's parcel database. Such databases are usually maintained outside of ArcCAD and are only accessed by ArcCAD for query and analysis. In these

cases, it is only important to verify that you can correctly read the data file(s) and that you can associate attribute records with the appropriate locational information stored for a feature.

## Creating themes from a coverage

Existing ARC/INFO coverages are another data source that can be used to create ArcCAD databases. Creating an ArcCAD database from a coverage is much simpler than creating the database from a drawing because all the work in creating the GIS data set has already been done. You only need to define themes for this GIS data set.

There are two steps for creating an ArcCAD database from a coverage:

#### Step 1 Define themes

Define themes of the appropriate feature class using **defthm** or **ddtheme** to access features stored in this coverage. Remember that more than one theme can represent geographic features in the same coverage. Use the **describe** command to list information about the features in a coverage.

#### Step 2 Create drawing entities

Create drawing entities from the coverage features by using any of the appropriate ArcCAD display commands to draw a graphic representation of theme features as entities into the current drawing. This allows you to view, query and edit theme features through the drawing. Refer to the 'Data display' section of this chapter for more information on displaying features.

The following notes include important information that will help you use existing coverages to create themes:

- When drawing features as entities into the current drawing, set link mode to 2 using the **link** command. This links each entity to its appropriate feature and ensures that only one entity is linked to each feature.

- The User-IDs for theme features will be the same User-IDs that are stored in the coverage's feature attribute table.

- When editing the theme using AutoCAD software's drawing tools, be sure to save the edits to the theme's coverage using **addfeat**, **savefeat** or **modfeat**. Then create topology using **build** or **clean**. Refer to the steps above for further information about modifying features and recreating topology.

## Creating a dBASE record theme

dBASE record themes are themes that access information stored in dBASE-compatible database files. dBASE record themes can be created from existing database files or from scratch using ArcCAD commands. If you wish to create a record theme using other types of database files, see the section on creating SQL record themes.

### Creating a new dBASE record theme

There are three steps in creating a new dBASE record theme.

#### Step 1 Define a theme

To create a new dBASE record theme, define a dBASE record theme using the **defthm** or **ddtheme** commands. These commands will ask you for the following information.

Theme name - the name of the dBASE record theme to create.
Feature class - the type of features this theme will contain. Specify 'Record, dBASE' to create a dBASE record theme.
GIS data set - the name of a database file that will store this theme data.

For more information about defining themes, refer to the **defthm** or **ddtheme** command reference.

#### Step 2 Define items

Define the record format of this record theme. Defining the record format involves defining items to contain whatever attribute information you wish to store in this dBASE record theme. A theme can contain up to 128 items. Refer to the chapter 'ArcCAD concepts' for complete information about item definitions. Items are defined using the **additem** command. Issue **additem** for each item you wish to define.

#### Step 3 Add records

Use ArcCAD commands such as **addrec** or your tabular database management system to add records to your dBASE record theme.

### Creating a dBASE record theme from a database file

ArcCAD can access information in any existing database file stored in dBASE III and higher compatible file format. These files may have been created using other DBMS or spreadsheet software packages. To access this data in ArcCAD, you must create a dBASE record theme.

To create a record theme from an existing database file, you only need to define a theme to access this data. Define a record theme using the **defthm** or **ddtheme** commands. These commands will ask you for the following information.

Theme - the name of the dBASE record theme.
Feature class - the type of features this theme will contain. Specify 'Record, dBASE' to create a record theme.
GIS data set - the name of the existing database file that contains information you wish to access.

The data in the database file can now be accessed using the dBASE record theme.

For more information about defining themes, refer to either the **defthm** or **ddtheme** command references.

## Creating an SQL record theme

SQL record themes allow access to industry-standard databases supported by AutoCAD. While the database table for a dBASE record theme is a dBASE file on disk (.DBF file extension), the database table for an SQL record theme often resides in an external RDBMS. The ArcCAD SQL themes are designed to function similarly to dBASE themes except that the tabular data in an SQL theme cannot be changed.

When an SQL theme is defined, ArcCAD verifies the database connection information and then fetches the queried records as specified in the theme's definition. When ArcCAD has fetched all the table records and stored them in a temporary file, you can then use the SQL theme much like you would use a dBASE theme.

#### Step 1 Setting up the SQL environment

To access an SQL database, you must first define the AutoCAD SQL environment. Within this environment, the DBMS driver controls the connection between the AutoCAD SQL Interface (ASI) and the data in the external database. Establishing a connection between the DBMS driver and the SQL environment is done within the ASI.INI file. ASI.INI is divided into sections that provide information required by the database drivers. Some

sections contain information pertinent to all DBMS drivers, while other sections supply information for the individual environments. Since ArcCAD uses the same AutoCAD SQL interface used by the AutoCAD SQL Extension (ASE), setting up ASE also sets up the ArcCAD SQL environment. Databases defined to work with ASE also work with ArcCAD SQL themes. For detailed information on setting up an SQL environment, refer to the *ArcCAD Installation Guide* and the *AutoCAD Installation and Customization Guides*.

#### Step 2 Defining an SQL theme

An SQL theme can be defined from the command line, the **defthm** dialog box, or the **ddtheme** function panel. When defining an SQL theme you will be prompted for the following information.

Theme - The name of the SQL record theme.

Feature Class - The type of features this theme will contain. Specify 'Record, SQL' to create an SQL record theme.

Connections - The SQL environment ArcCAD will connect to. This environment determines the RDBMS from which to query.

Login - The user login required for connection to the RDBMS. (For some RDBMSs this is not required.)

Password - The user password required for connection to the RDBMS. (For some RDBMSs this is not required.)

Expression - The SQL expression used to query records from the RDBMS. A discussion of SQL syntax may be found in your AutoCAD documentation.

For more information on defining an SQL record theme, refer to the **defthm** or **ddtheme** command reference.

***Note:*** to view the SQL record theme, use the **wbrowse** table browser.

## Creating an image theme

Image themes allow ArcCAD to support basic raster functionality, such as the display of images as a backdrop to the AutoCAD drawing. Image themes display the following raster formats.

Monochrome - HRF, PCX, TIFF, RLE, IMG, RLC, BIN, RNL, RL9, CG4, CCRF

Color - TIFF, PCX, GIF

#### Step 1 Define theme

To define an Image theme use the **defthm** or **ddtheme** commands. Enter a theme name and select the image theme type. At the GIS data set prompt enter the full pathname to the image file you want to display. Be sure to include the file extension with the full pathname.

#### Step 2 Display theme

To display an image theme use the **image** command. The image is displayed according to the information contained in an associated .RAT or World image configuration file, if one exists. In order for the **image** command to associate the image configuration file with the image, the files must follow the convention of *'image_name'.rat* and *'image_name'.wld*, where *'image_name'* is the name of the image to be displayed and .wld is any World file extension. See the **image** command reference. For example, an image called ROADWAY.TIF would have an associated image configuration file called ROADWAY.RAT or ROADWAY.TFW. The following information is stored in the .RAT and World files.

| | |
|---|---|
| Insertion point | X,Y location of the lower-left corner of the image. |
| Scale | A value by which the image will be increased or decreased. |
| Rotation angle | The angle by which the image will rotate about the insertion point. |
| DPI | The number of pixels per inch. |
| Skipping scan lines | The number of scanlines to be skipped between drawn lines. This affects drawing speed. |

The .RAT and World file store the same information. The .RAT file is a standard image configuration file used by many image processing packages, while the World file is the standard image configuration file used by ESRI for ArcView and ARC/INFO image display. ArcCAD will read either one of these files, depending on what is present.

If a configuration file does not exist for the image theme, default parameters are applied that set the image insertion point at the coordinates (0,0). To modify the existing configuration of an image or to create a configuration file, if one does not exist, use the **iconfig** or the **iregister** commands. These commands will allow you to manipulate the image for placement as a backdrop for coverages and AutoCAD entities. The **iconfig** command will prompt for each of the configuration parameters, whereas the **iregister** command will prompt for the location of two registration points on the image.

***Note:*** Only one image may be displayed at a time. To remove an image from the screen, use the **iclose** command.

# Important processing guidelines

Listed below are some guidelines that will help in using ArcCAD.

1) Be aware of the topology defined for a theme and how it is modified by ArcCAD commands. Many commands automatically update topology (e.g., **build**) while others do not (e.g., **addfeat**). **Describe** can be used to determine if a coverage has topology for various feature classes.

2) Do not edit database files with the system text editor. Special control characters required for editing could be added to the file, rendering it useless. Database files are dBASE-compatible database files and should only be edited using appropriate ArcCAD commands or a dBASE-compatible database management system.

3) Do not delete, change or add new items between the following standard items in theme feature attribute tables:

| Attribute file name | Items not to be altered |
|---|---|
| Theme TIC | IDTIC, XTIC, YTIC |
| Theme BND | XMIN, YMIN, XMAX, YMAX |
| Theme PAT | AREA, PERIMETER, COVER_, COVER_ID |
| Theme AAT | FNODE_, TNODE_, LPOLY_, RPOLY_, LENGTH, COVER_, COVER_ID |

ArcCAD expects a standard item format for these files and will not recognize changes made to them.

After the User-ID, items in the PAT and AAT can be added or modified. Items should not be added to the TIC or BND files of a coverage.

4) Never change the Internal-ID values for any record in the PAT or AAT. The Internal-ID is the internal feature number and is used as a direct access key by ArcCAD to theme features.

5) A theme's PAT and AAT files must be sorted on the Internal-ID. For each record, the Internal-ID and $RECNO must be equal. If you sort one of these attribute tables on some other item, you must re-sort it on the Internal-ID before using any other ArcCAD command on that theme.

6) Always run **idedit** after altering User-IDs in a feature attribute table. User-IDs for theme features are stored in two coverage files. If the User-ID is not the same in both files, other feature attributes can be lost.

7) Use unique User-IDs for each feature to ensure that additional item values maintained as part of a theme are modified or updated correctly.

8) Always save the results to the theme's GIS data set after editing the drawing using the **addfeat**, **savefeat** or **modfeat** commands.

9) After running **addfeat**, **savefeat** or **modfeat** to update changes in feature topology to the theme's GIS data set (adding or deleting label points and arc features) use **clean** or **build** to recreate topology.

10) Do not store point and polygon features together in the same coverage. Since label points are used to represent point features and assign User-IDs to polygons, these two features cannot be maintained in one coverage.

11) Use your directory structure to organize your work. Remember, you can access coverages from more than one workspace with an ArcCAD command by using pathnames.

## Theme management

ArcCAD includes several commands to assist in the management of your digital database. To obtain information about theme coverage features and characteristics, use the commands **frequency** and **describe**. **Frequency** produces a list of the unique code occurrences and their frequency for a specified set of items and **describe** provides details about the feature contents of a coverage. **Listthm** will display the themes in the current drawing.

The **append** command is another useful function allowing up to 100 coverages or themes to be merged into one.

## Coordinate transformation

Once a coverage is created, it can be manipulated to fit specific output requirements (e.g., to convert the coverage coordinates from one coordinate system to another or from one map projection to another). The ArcCAD command **transform** uses an affine or a projective transformation function to shift, scale or rotate theme coordinates. The transformation function is calculated by comparing the locations of control points (tic features) in a theme's coverage to their desired new locations (tic values in a new empty theme). The transformation function is then applied uniformly to all theme coordinates. Refer to the **transform** command reference for information on transforming theme coordinates. The ArcCAD utility command **project** is used to convert coverage coordinates from one map projection to another. The **project** command reference in the appendix of the *ArcCAD Command Reference* describes the process of projecting coverage coordinates.

## Coordinate precision

ArcCAD coverages store coordinate values in single precision with approximately 7 digits of accuracy. AutoCAD maintains drawing entity information in double precision with approximately 13 digits of accuracy. In order to maintain the same level of accuracy between the theme data stored in the coverage and the theme feature's representation in the drawing, coordinates can be shifted using the **xyshift** command. **Xyshift** is used to offset coverage coordinates by a fixed value to maintain coordinate precision. The xyshift for a theme is applied to coordinates when entities are created from features or when features are created from entities.

ArcCAD also provides commands for importing and exporting double-precision interchange files. The **esri_import** command will read a single- or double-precision interchange file, optionally create an xyshift offset and create a single-precision ArcCAD coverage. The **esri_export** command will give you the option of creating a single- or double-precision interchange file. Refer to the **xyshift**, **esri_import**, and **esri_export** command references for further information.

## Coordinate generalization

Line and polygon themes may sometimes contain too many vertices for your particular application. An arc feature may contain more vertices than necessary to represent the actual shape of that feature at the scale of your database. This redundant data can be automatically removed using the **generalize** command. **Generalize** removes arc feature vertices within a specified tolerance. Refer to the **generalize** command reference for more information.

## Transferring ArcCAD databases

ArcCAD includes utilities for transferring coverage data between your PC and other computer platforms running ARC/INFO®. The ArcCAD **esri_export** command can be used to convert coverage data files to an interchange-file format. The **esri_export** command produces either single- or double-precision interchange files. This interchange file can then be transferred to other ARC/INFO platforms. The **esri_import** command converts an interchange file, which can be in either single or double precision, to the ArcCAD coverage data file format. The **shapein** command converts an ArcView shape file to an ArcCAD coverage. If coverage data is sent between ArcCAD sites, you do not need to use **esri_export** and **esri_import** since the coverage file format on a PC is the same.

When transferring themes be sure to include all theme components: the drawing, the theme's GIS data sets (coverages and database files) and the drawing's link file directory.

Drawings can be directly sent to ArcCAD sites but cannot be directly sent to ARC/INFO sites. Use the **dxfout** command to create a DXF file and provide this DXF file to ARC/INFO sites.

# Managing ArcCAD databases

This section will describe some guidelines that you can use to help you maintain and edit theme features, and manage attributes in ArcCAD databases.

## Using an existing drawing

When ArcCAD is loaded, it must perform several checks on the current drawing to verify link information and database integrity.

In ArcCAD, entity-feature links are implemented using AutoCAD entity handles. Handles must be turned on in the current drawing for entity-feature links to operate correctly. If the current drawing does not have handles turned on, ArcCAD will automatically turn them on and print the message:

```
Turning HANDLES ON...
```

If themes have been previously defined in the current drawing, each theme is checked for the existence of its GIS data set and valid entity-feature links. If the current drawing is a new drawing or an existing drawing that has no themes, this step is skipped. If the drawing does contain themes and if the GIS data set of any themes cannot be found, ArcCAD must either locate the data set or drop the theme. This process is called theme recovery. When a theme's data set needs to be recovered, by default, ArcCAD will present a dialog box into which the user can type the pathname to the missing GIS data set (or the SQL login and query for SQL record themes). By entering the correct GIS data set pathname and selecting ok, links for the theme are reestablished. Selecting the CANCEL button drops the links between the specified theme and its GIS data set. If you tell ArcCAD to drop the links, the theme definition remains but it will no longer be connected to its GIS data set. You may wish to drop a theme's links if you know its GIS data set no longer exists and you do not wish to use that theme anymore. After ArcCAD has loaded, you may want to delete that theme using the **kill** command.

After the GIS data set for a theme has been either verified or reestablished, entity-feature link verification is performed to ensure that if entity-feature links exist for entities, then the corresponding features can be located. If the appropriate features cannot be matched with the entities for a given theme, then the entity-feature links must be dropped. In other words, entity-feature links are invalid if the features they relate to cannot be found. When entity-feature links are dropped, the following message is displayed:

```
Dropping links for theme theme_name
```

If the current drawing contains themes, ArcCAD must reestablish internal data structures that manage entity-feature link information for those themes. ArcCAD stores this entity-feature link information in link files. Link files are stored in a directory that has the same name as the current drawing with the

extension of '.LNK'. The entity-feature link information for each theme defined in the current drawing is stored in a separate link file in the drawing's link file directory. The link file directory must exist in the same directory as the current drawing. If ArcCAD cannot find the current drawing's corresponding link file directory, or one of the link files for any theme is missing, the drawing's link files must be reconstructed. ArcCAD issues the following message:

```
Link information missing.
```

Entity-feature links are then recreated for each entity in the drawing that has a corresponding theme feature. The process of establishing entity-feature links for all themes can take some time to complete. ArcCAD prints the messages:

```
Establishing links...
Please wait...
```

when link data structures are being rebuilt. Refer to the 'ArcCAD database structure' appendix for complete information about link files.

If the current drawing does not contain themes, this step is skipped.

Since there are no links for tic and record themes, the entity-feature link verification step is skipped for these theme feature classes.

# Editing theme coordinates

When an ArcCAD database has been created from AutoCAD drawings, there are two ways to edit theme coordinates:

1) Edit the original entities that were used to create the theme features.

2) Recreate the entities (preferably on a different layer) from the theme features using ArcCAD display commands and edit these entities.

Each method has advantages and disadvantages. For example, if you have a polyline containing 700 vertices and edit the polyline using **pedit**, the entire polyline will be selected and modified if method 1 is used. You can then modify the corresponding feature stored in the coverage using **modfeat**. The single original polyline represents two arc features in the theme (recall that an arc feature can contain a maximum of 500 vertices) both of which will be updated. If you used **pedit** on the recreated entities using method 2, you will need to potentially modify two polylines and then write both of them back to the coverage using **modfeat**. Method 1 is simpler in that a single entity is edited. But the procedure in method 2 more closely parallels how the features are actually stored in the coverage. The recommended approach is to use method 2 as this manipulates entities that actually represent individual features in the GIS data set.

Regardless of which editing method is chosen, edits to the theme's coverage are made through the drawing using AutoCAD software's geometric construction tools. In order to use the drawing to edit the theme coordinate data, the drawing must contain the most recent graphic representation of the theme as discussed above. If the theme to be edited already has a graphic representation of the entities in the current drawing, you can use those entities to edit the theme's coverage. If the theme does not yet have a graphic representation of its features, this can be done by using appropriate ArcCAD display commands to draw coverage features into the current drawing as AutoCAD entities. Please refer to the chapter 'Display and query' for complete information on using ArcCAD display commands.

The **xselect** command can be very useful during ArcCAD editing sessions to create entity selection sets. AutoCAD editing functions can then be applied to the selected set of entities. Refer to the **xselect** command reference for complete information on using the **xselect** command.

## Entity-feature links and editing

In ArcCAD, drawing entities are the graphic representation of theme features. As such, they can be freely manipulated independently of the features they represent. For example, you can delete an entity using the AutoCAD **erase** command without affecting that entity's associated feature. Entities and features can also be linked together. For example, the feature associated with an entity can be deleted using the **delfeat** command. This does not delete the entity, however. The entity must be deleted using the AutoCAD **erase** command. This concurrent independence and linkage between entities and features is managed by the **link** command. ArcCAD feature management and display commands look for entity-feature links. The link (if any) to the feature is only referenced when an ArcCAD feature management or display command is used on that entity. Thus you can move an entity in a drawing, but that entity's new location is only written to the theme's coverage if the **modfeat** or **savefeat** command is used on it. If the entity is moved but not written to the theme's coverage, then the graphic representation (the entity) would not reflect the true location of that feature.

Entity-feature links are established when entities are created from features using the ArcCAD display commands. The behavior of these links is dependent on the current link mode setting which is set by the **link** command. There are three possible link mode settings. If link mode is 0, then there is no link established between entities and features. If link mode is 1 or 2, links will be established between entities and features, but the way entities are managed during display operations will be different. For link mode 1, every display command will generate a new entity in the drawing. Previously generated entities will remain in the drawing, but only the most current instance of an entity will be linked to a feature in the coverage. This means that display commands could generate multiple entities, all representing the same feature, one on top of another if the entities have not been moved, but only the most

recent entity would be linked to a feature. If link mode is 2, a previous instance of an entity is erased before a new entity is generated. This option keeps the drawing 'clean' by assuring that there are no entities in the drawing that are not linked to features.

When entities are created to represent features for editing purposes, it is strongly recommended that link mode be set to 2. This will eliminate any confusion as to which entities are linked to which features and the possibility of corrupting your theme by inadvertently updating features from undesired or obsolete entities.

If entities have been created with link mode set to 0 or 1, you can change the link mode to 2 by erasing all of the entities in the drawing, setting link mode to 2 using the **link** command and then recreating the entities using ArcCAD display commands. This situation can always be avoided by setting the link mode to 2 when creating entities or features in ArcCAD.

The following notes pertaining to entity-feature links will help you make efficient use of ArcCAD to edit theme features.

- Entity-feature links for editing are only important when editing point, line, polygon and annotation features. The current link mode setting does not apply to record, tic, or image features since there are no entity-feature links for these feature classes.

- Entity-feature links are established when features are created using the **addfeat** or **savefeat** commands and when entities are first created by ArcCAD display commands. These commands establish a link based on the current setting of link mode as set with the **link** command. All ArcCAD commands that can access an entity and its corresponding feature use the existing link as established using **addfeat**, **savefeat** and the display commands.

- The **modfeat** and **delfeat** commands modify and delete features linked to entities. If there is no entity-feature link for a feature, these commands will have no effect.

- **Delfeat** will only delete an entity's corresponding feature stored in the theme's coverage. It does not delete the entity itself.

- The AutoCAD **erase** command can be used to delete entities. **Erase** will delete entities, and any entity-feature links associated with the deleted entities, since entity-feature links are stored as part of an entity's extended entity data. **Erase** does not delete features stored in a GIS data set.

- Entity-feature links are stored with the extended entity data of an entity. This means that care should be taken when using the AutoCAD **explode** command because all of a block's extended entity data is lost when a block is exploded. For example, if you display point features or polygon label points with ArcCAD symbology, the marker symbols are displayed as blocks. If you

explode the blocks, the links to the corresponding theme feature will become corrupted.

■ ***Warning:*** Do not use the AutoCAD **insert** command to insert a drawing that contains theme definitions into the current drawing. This will corrupt entity-feature links in the current drawing. If you need to do this, you must remove all themes from the current drawing prior to inserting the new drawing. Xrefs are not supported in ArcCAD.

■ In order to edit the polygon boundaries in a polygon theme, you must create a line theme that has the same GIS data set as the polygon theme. This is necessary because entity-feature links are created for polygon label points only; links are not created for the arc features that comprise the polygon boundaries. After edits have been made to the linear entities, use **addfeat**, **savefeat** or **modfeat** to write these coordinate changes to the line theme and then **clean** or **build** the polygon theme to recreate topology.

■ When all edits are complete, you must write the coordinate changes back to the theme using **addfeat**, **savefeat** or **modfeat** and rebuild topology using **build** or **clean**. The following table will help you decide which feature management command to use:

| If you... | Use... |
|---|---|
| Created new entities in the drawing that represent new features in the GIS data set. | **addfeat** - adds new features to the GIS data set. |
| Modified entities that represent existing features in the GIS data set. | **modfeat** - modifies existing features in the GIS data set. |
| Created new features and modified existing features in the GIS data set. | **savefeat** - adds new features and modifies existing features in the GIS data set. |

*Theme management commands*

# Managing attributes

ArcCAD creates and maintains attribute information in dBASE-compatible database files. Every theme feature class (except annotation and image themes) has an explicit database file. Point, line and polygon themes have an associated feature attribute table that is a database file that stores information about each feature in the theme. There is one entry or record in the feature attribute table for each feature. Tic themes have an associated TIC file that stores control

points. Record themes allow any type of data to be stored in a database file. Their primary use is to relate additional data to feature attribute tables.

You can use either the ArcCAD attribute management commands or a relational DBMS compatible with dBASE file formats for managing and viewing your attribute data. Both ArcCAD commands and the RDBMS can assist in creating, manipulating, listing and merging feature attribute tables and record themes. ArcCAD keeps track of and updates feature attribute tables, and manages relationships between features and their corresponding tabular records. You can create record themes to contain any data you wish and relate them to other record themes or feature attribute tables. ArcCAD also provides commands that can be used to perform logical and arithmetic operations on the rows and columns of any database file.

There may be occasions when you require the advanced functions of a complete relational database management system such as dBASE. Some of the features supported by relational database management systems include:

**Report Generation**—create reports with column headings, sort and break on various item values, total item fields and average fields.

**Screen Management/Forms**—these are user-designed forms that appear on your screen. A form may be designed that allows you to enter values to prompts on your screen (e.g., 'Enter Last Name:', 'Address:', 'ZIP Code:', etc.). Forms can include rules for entry such as defining a set of legal values or ranges (e.g., AGE > 0) or illegal values; rules for forcing values (e.g., if COVER = 'WATER', then SLOPE = 0). Forms can be used for data entry, query and update.

**Programs**—you can write special programs to perform your specific analysis operations.

Following is a brief description of some of the most commonly used ArcCAD database management commands that operate on the database files of any theme feature class. This description is only meant to introduce each command. Refer to the appropriate command references for more details on each one.

## Building and deleting database files

The GIS data set for a record theme is a database file. To create a record theme, define a record theme using **defthm** and then define its items using the **additem** command. To delete a record theme, use the **kill** command.

Several commands are introduced here that help you create, delete and modify the structure of database files:

**Additem**—adds an item to a theme's database file.

**Pullitem**—copies a subset of items and values from a theme's database file to a new theme. The pulled item values for all records are saved in the new data file.

**Joinitem**—merges two theme database files together using a relational join. Each file has a common item. Records between each file are matched together when the item value in the first file matches an item value in the second file. Matched records are then merged into one new record.

**Dropitem**—deletes items from a theme's database file.

**Items**—lists the item definitions for a specified theme's database file.

**Kill**—deletes a record theme's database file. When used on other theme feature classes, data other than the associated database file may be deleted. Refer to the **kill** command reference for complete information on the **kill** command.

**Sortdb**—arranges the records in the theme's database file in any order. Remember that if you sort a theme's PAT or AAT file, sort the file on its Internal-ID item before performing any ArcCAD commands that may access the feature attribute table.

**Moditem**—modifies a theme database file's item names, definitions and widths.

**Sizeitem**—resizes numeric items in a theme's database file to be smaller, based on the maximum width of data within the item, or larger to the default width of the item.

## Displaying database file information

**Listdb**—allows the examination of data in the theme's database file by listing items and their values. If a **relate** is active, then the related items and values are also listed.

**Wbrowse**—allows the examination and editing of dBASE and SQL record themes, as well as the feature attribute tables of point, line, polygon, and tic themes. **Wbrowse** is designed for use only within the Windows and Windows NT environments.

**Items**—lists the item definitions for a theme's database file.

**Frequency**—produces a list of unique code occurrences and their frequency for a specified set of items in a theme's database file.

**Statistics**—generates summary statistics for an item in a theme's database file.

## Adding and deleting records

Once a record theme is defined, data can be added using the following commands. Note that you cannot add or delete records from an SQL theme (refer to the *ArcCAD Programmer's Guide* for more information). Remember that you should not add or delete records in a feature attribute table. ArcCAD maintains feature attribute tables, and manages relationships between features and their corresponding tabular records.

**Addrec**—allows you to interactively add new item values to a theme.

**Copydb**—copies database file records from any theme's database file to a record theme.

**Loaddb**—appends data from a fixed-length ASCII file to a theme.

**Delrec**—deletes the currently selected records of a theme's database file. Deleting data from a database file usually occurs in two steps: first select the set of records you wish to delete and then delete them. Record selection is performed using the query operations described in the next section.

## Querying a database file

Whenever you work with a theme's database file, you may wish to operate on only a subset of the records without changing the actual number of records in the database file. Whenever a theme is accessed, all of its records are immediately available to you for changing, listing, deleting, and so on. ArcCAD provides three record-selection commands that can be used to select a subset of records from the theme. You indicate which records you wish to use by specifying selection criteria that the item values must meet (e.g., AGE greater than 40 or LAST_NAME is JONES). To help you define these search criteria, ArcCAD provides a set of logical operators. These logical and arithmetic operators are described below in the section 'Logical expressions' and are available for use with the following record-selection commands:

**Reselect**—allows you to select a subset of records from a theme. Those records whose item values match the selection criteria become the subset.

**Aselect**—allows you to add records to a previously selected set.

**Nselect**—replaces the currently selected records with those not selected and unselects the currently selected records.

**Relate**—temporarily relates a theme with a record theme using a common item.

## Logical expressions

Logical expressions in ArcCAD commands have three components: operands, logical operators and logical connectors.

### Logical operands

- The name of an item in a theme's database file (e.g., `STREAMS_ID`)
- A constant numerical value (e.g., `10`)
- A character string in single quotation marks (e.g., `'HIGH'`)
- An internal variable (e.g., `$RECNO`)

### Logical operators

- `EQ` or `=` Operand-1 is equal to Operand-2.
- `NE` or `<>` Operand-1 is not equal to Operand-2.
- `GE` or `>=` Operand-1 is greater than or equal to Operand-2.
- `LE` or `<=` Operand-1 is less than or equal to Operand-2.
- `GT` or `>` Operand-1 is greater than Operand-2.
- `LT` or `<` Operand-1 is less than Operand-2.
- `CN` Operand-1 contains the character expression Operand-2. Character operands only (e.g., `NAME CN 'MAIN'`).
- `NC` Operand-1 does not contain the character expression Operand-2. Character operands only (e.g., `NAME NC 'MAIN'`).
- `IN` Operand-1 is contained in the set of numeric constants or character strings specified in Operand-2. This set of constants or character strings must be enclosed in { } brackets. The individuals in the set must be separated by commas, unless they are being used to express a range, in which case, -> is used to separate the individuals forming the lower- and upper-inclusive limits of the range. A range defined between two character strings is based on the ASCII number sequence, which is alphabetical. No blank spaces should separate any of the elements within the brackets.

***Note:*** Computer roundoff can alter the values of real numbers. This can cause a problem when specifying real numbers in a [logical expression] that require equality. When using expressions of equality, the operands must match exactly for a match to be found. For example, the value .01139 does not equal .0114. In such cases, use an expression that includes a range of real values (i.e., "HEIGHT GT .01139 AND HEIGHT LT .01141").

### Logical connectors

- AND — For the condition to be evaluated as true, the logical expressions on both sides of the AND must be true.
- OR — For the condition to be evaluated as true, the logical expression on one or the other side of the OR must be true. The condition will also be evaluated as true if both logical expressions are true.
- XOR — For the condition to be evaluated as true, the logical condition on one and only one side of the XOR must be true. If both logical expressions are true or both are false, the condition will be evaluated as false.

The simplest logical expressions take the following form:

[operand-1] [logical-operator] [operand-2]

For example,

```
CLASS LT 8
```

Up to eight logical expressions of this simple form can be combined to form more complex expressions by using logical connectors. For example,

```
CLASS GE 2 AND CLASS LT 8 OR SUIT = 5
```

There is no specific limit to the number of [operand-1] [logical-operator] [operand-2] combinations and logical connectors that can be used in a single expression. However, commands that have logical expressions as arguments are limited to 254 characters in length.

All logical operators and connectors have equal precedence. Operations are performed in sequence from left to right. However, parentheses can be used to request that logic within parentheses be performed first. Operations inside the innermost set of parentheses have the highest precedence.

Each element of a logical expression (i.e., operand, logical operator, logical connector, or parenthesis) must be separated by blanks, except when using the IN operator.

Logical expressions can only be used on point, line, polygon and record themes. They are not supported on tic and annotation themes.

In order to perform selection on point, line or polygon themes, you must have run **build** or **clean** to create topology and feature attribute tables.

## Arithmetic expressions

Arithmetic expressions in ArcCAD have the following components:

### Numeric operands

- An item name
- A constant (e.g., `10`)
- An internal variable (e.g., `$RECNO`)

### Arithmetic operators

- **+** Addition
- **-** Subtraction
- **/** Division
- ***** Multiplication
- ****** Exponentiation
- **LN** Logarithm
  Calculates the natural logarithm of the operand it precedes. The operand must be a positive number.
- **WD** Width computation
  Calculates the width in characters of the operand it precedes, excluding trailing blanks. The operand must be a character item or a literal string.

Arithmetic operators have the following precedence from highest to lowest:

1) LN, WD
2) **
3) *, /
4) +, -

Operands of equal precedence are performed as they are encountered, moving from left to right through the expression. Parentheses can be used to override inherent precedence. Operations within the innermost set of parentheses are performed first.

***Note:*** There is no unary minus operator for negating an operand in ArcCAD. For example, the expression -AGE evokes an error message (instead, specify -1 * AGE). Also, all arithmetic operations in ArcCAD are performed in double precision. As a result, an expression involving integer operands may be evaluated as having a fractional part.

Examples of arithmetic expressions:

```
SUIT = ( SOIL + 2 * TERRAIN ) / 12
LAB_WIDTH = ( WD ( LABEL ) + 4 ) * 0.22
```

## Display width limitations

A number assigned to a NUMERIC type item with 0 decimal places that exceeds the defined width for its item definition is displayed as asterisks (**). The item value is lost.

The values of a NUMERIC type item defined with 1 or more decimal places that exceed the defined width for that item will be displayed in scientific notation. If this occurs, you can either increase the size of the item or add a new item and use the **calculate** command to move the values to the new item.

Character items are left justified in their output display field and truncated if they contain more characters than the defined output widths of their items.

## Internal variables usable in logical and arithmetic expressions

ArcCAD provides three internal variables that can be used in logical and arithmetic expressions.

**$RECNO**—the record number of a record in the selected data file.

**$PI**—the value for *pi* (3.14159…), which is the ratio of a circle's circumference to its diameter.

**$E**—the value for *e* (approximately 2.71828), which is the base of the number system for natural logarithms.

These internal variables can be used as operands anywhere within a logical or arithmetic expression. For example, with the **reselect** command, you can specify:

```
$RECNO GT 100
```

This selection expression will find all records from the currently selected set whose record number is greater than 100.

## Modifying item values

There are four commands that can be used to change item values in a theme's database file. If a subset of records has been selected using the selection commands, any of the following commands can be used to update values in a database file:

**Calculate**—assigns new values to a numeric item using arithmetic expressions for all selected records. These expressions are described below in the section 'Arithmetic expressions'.

**Modone**—allows interactive modification of a theme's database file or to a related theme using an input form. **Modone** is very useful for modifying feature User-IDs. When changing User-IDs in the feature attribute table, always remember to run **idedit** immediately afterwards to update User-IDs in the coverage geometry files.

**Moveitem**—updates a character item for all selected records.

**Modrec**—changes item values record by record and item by item. You specify a record number, view item values for that record, and then change desired item values.

## Saving data outside of a database file

It may often be desirable to save the selected records of a theme as a disk file. For example, this may be used for loading the selected records into another system such as a spreadsheet or statistical package. The following commands can be used to create new database files or ASCII text files from database files:

**Copydb**—saves the selected records of a theme's database file as a disk file. The saved disk file is still a dBASE-compatible database file. It is often loaded back into ArcCAD using **loaddb**.

**Dumpdb**—creates ASCII text files of selected database file records.

Chapter 4

# Display and query

ArcCAD® software's display and query commands provide facilities for interactively creating and previewing maps and for using maps as graphic windows into themes for interactive query and update of feature attribute information. All these display operations are carried out from within the AutoCAD® graphic environment. In order to manipulate entities and features you must be able to indicate which objects are to be considered. This process involves the selection of objects using combinations of spatial, graphical, and attribute selection criteria that are specified as a series of logical expressions. ArcCAD software's query commands allow you to perform feature and entity selection, maintain feature and entity selection sets, and retrieve information related to the objects in these selection sets.

## Displaying theme features

The ArcCAD display operators create entities in the current AutoCAD drawing that are the graphical representations of the theme's geographic features. In this way, drawings may be created from theme features. Features can be drawn using one symbol, or you can use a lookup table to specify different symbols for different classes of theme features.

### Using entity-feature links with display commands

The link mode setting is important when using ArcCAD software's display commands. Entity-feature links establish one-to-one relationships between an entity and its corresponding feature. If a theme only has entities or only has features, there can be no entity-feature links. An entity-feature link is created either when the feature is created from the entity or when the entity is created from the feature. The section 'Entity-feature links and editing' in the chapter 'Database automation' describes how entity-feature links are managed when creating features from entities. This section will describe how entity-feature links are managed when creating entities from features.

In ArcCAD, drawing entities are the graphic representation of theme features. As such, they can be freely manipulated independently of the features they represent. For example, you can move an entity using the AutoCAD **move**

command without affecting that entity's associated feature. Entities and features can also be linked together. For example, the feature associated with an entity can be deleted using the **delfeat** command. This concurrent independence and linkage between entities and features is managed by the **link** command. ArcCAD display commands look for entity-feature links. Even if entity-feature links are established for a particular entity-feature pair, you can manipulate the entity independently of the feature using AutoCAD commands. The link (if any) to the feature is only referenced when an ArcCAD display command is used on that entity.

When drawing entities are initially created from theme features, an entity-feature link can be established based on the current link mode setting as set with the **link** command.

## Link mode 0

When link mode is set to 0 and entities are created from features using ArcCAD display commands, no entity-feature link is established for those features. This means that a feature or entity does not know that it has a corresponding entity or feature. You cannot query the feature attributes of entities created with link mode 0. Every time you issue a display command that draws a particular feature, a new entity that represents that feature will be created in the drawing (i.e., if you issue the **arcs** command three times for a theme when link mode is set to 0, you will create three entities for each arc feature).

## Link modes 1 and 2

When link mode is set to 1 and entities are created from features using ArcCAD display commands, a one-to-one correspondence is established between each entity and feature. This means that a feature or entity knows that it has a corresponding entity or feature. Link mode 1 is most useful for editing rather than display purposes (refer to 'Entity-feature links and editing' in the 'Database automation' chapter). When an entity with link mode 1 is recreated using an ArcCAD display command, the entity-feature link on the original entity is moved to the newly created entity. The original entity is not erased but remains in the drawing.

Link mode 2 also establishes a one-to-one entity-feature link but has an additional feature. Every time you issue a display command that draws a particular feature, if an entity linked to that feature already exists in the drawing, that entity is first erased and a new entity that represents that feature is created. This means that if you issue the **arcs** command three times for a theme when link mode is set to 2, you will still only have one entity that represents that feature. Setting link mode to 2 is the recommended setting when displaying features using ArcCAD display commands. By default, link mode is always set to 2 unless it is explicitly changed.

To change the entity-feature links of any set of entities, first erase those entities, then reset link mode to the desired value (using the **link** command) and finally, recreate the entities using ArcCAD display commands. The new entities will have entity-feature links current with the current value of link mode.

# Using display commands

ArcCAD provides many different display options for drawing theme features. Features can be displayed in several different ways:

- Using the current AutoCAD settings
- Using the default symbol defined for a theme
- Using a specified symbol number
- Using attribute values as symbol numbers
- Using attribute values to look up symbol numbers in a lookup table
- Using entity property constant values or items

Each of these display options will be discussed below.

***Note:*** Before using an ArcCAD display command to display theme features, use the **setext** command on that theme.

## Using the current AutoCAD settings

To use the current AutoCAD symbol settings for feature display, specify symbol number 0 for any symbol number prompt. The properties of entities created to represent theme features are assumed from the current AutoCAD environment. Refer to the *AutoCAD User's Guide* for more information about AutoCAD symbol settings.

## Using the default symbol defined for a theme

A default symbol number must be specified whenever a point, line or polygon theme is defined. This symbol number is stored with the theme definition. ArcCAD display commands can use this symbol whenever those theme features are displayed. The **points** command displays point features using the default symbol number. The **arcs** command displays arc features using the default line symbol. To shade polygon features using the default symbol number, use the **polyshd** command. The annotation feature class stores the default symbol number with each annotation feature. The **annotxt** command displays annotation with these symbols. When a display command prompts for a symbol number (e.g., 'Line symbol number (0-100) (?/<5>)') the default value displayed in the prompt will always be the default symbol number for that theme.

## Using a specified symbol number

To specify a symbol number other than the default, enter the desired symbol number whenever an ArcCAD display command requests a symbol number (e.g., 'Line symbol number (0-100) (?/<5>)'). This symbol number will always override the default symbol number.

## Using attribute values as symbol numbers

The **arclines**, **labelmrk**, **pointmrk** and **polyshd** commands let you specify different symbols to represent different features. Symbols are assigned to features according to attribute values stored for the features in feature attribute tables. For example, you can use the values of an attribute representing classes of roads to determine the symbology used to display the different road classes. Then, if you are making a road map, the **arclines** command will let you draw the arc features from the roads line theme using different line symbols to represent different classes of roads. If you are making an urban land use map, you could use the **polyshd** command to shade polygons from a land use polygon theme using values from an attribute representing land use classification to determine the appropriate shade symbols required to represent different land uses.

The values of any numeric item from the theme's feature attribute table can be used to access the symbol numbers used to draw the features. The item named in the command can be of any numeric type.

Item values with decimals are truncated when used to determine symbol numbers, so a value of 23.9 would access symbol number 23. Features with negative item values are not drawn. Features with item values of 0 are displayed using the current AutoCAD symbol settings. Features with item values greater than the highest symbol number in the appropriate symbol file are also not drawn, so a feature with an item value of 105 will not be drawn if the largest symbol number in the symbol file you are using is 100.

For example, the command:

```
Command: polyshd
Theme name: parcel
Display using? item
Item: tax_val
```

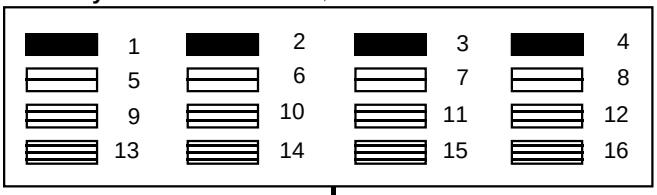
could be used to shade the polygons in the PARCEL theme using their values for the TAX_VAL item as shade symbol numbers:

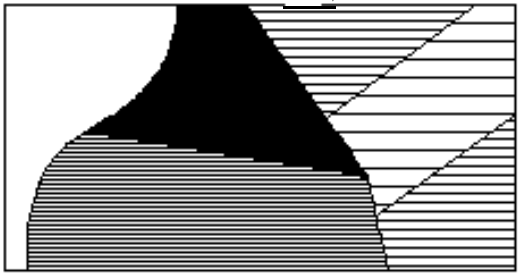
**PARCEL PAT**

| AREA | PERIMETER | PARCEL_ | PARCEL_ID | TAX_VAL |
|---|---|---|---|---|
| 4.391 | 4.509 | 2 | 1 | 0 |
| 1.854 | 3.625 | 3 | 2 | 10 |
| 1.957 | 4.212 | 4 | 3 | 6 |
| 5.231 | 11.967 | 5 | 4 | 15 |
| 2.744 | 8.912 | 6 | 5 | 3 |
| 1.294 | 3.107 | 7 | 6 | 11 |

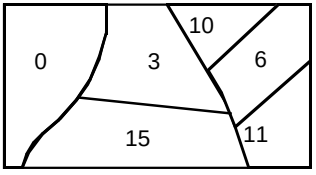
***Polyshd** uses values from the TAX_VAL item as symbol numbers. The map below shows what these values are for each polygon. (Note that each column in the symbol file represents a different color for the same set of symbols.)*

**Shade symbol file**



**PARCEL THEME**



## Using attribute values to look up symbol numbers in a lookup table

ArcCAD also lets you use a lookup table to assign symbols to features. This method is more flexible than using item values because the values of your attributes will not usually result in desirable symbols.

A lookup table is a type of record theme. The record theme's geographic information system (GIS) data set is a database file that contains information needed to symbolize features. Any item from a theme's feature attribute table can be used as the lookup item to a lookup table. The lookup table stores values from this lookup item. The lookup table also stores symbol numbers. To display each theme feature, ArcCAD reads its value for the lookup item in the feature attribute table and then finds this value in the lookup table to obtain the assigned symbol number. Lookup tables can contain a value for every possible value for the attribute in the feature attribute table, or more typically, will contain representative values for the attribute which define a range of values in the lookup table (see below).

### How to create a lookup table

As mentioned earlier, a lookup table is a record theme which can be defined using **defthm**. A lookup table consists of at least two items. The first item is the lookup item. This item is used to relate the lookup table to the theme AAT or PAT. It stores values from the corresponding item in the theme feature attribute table. This item must be the same item type as it is in the theme attribute table.

The second item in the lookup table must be named SYMBOL. Commands like **polyshd**, **arcmrk** and **arclines** require an item named SYMBOL. This item stores the symbol numbers used for drawing the desired coverage features. This item may be defined as any numeric item type.

The lookup table must be sorted in ascending order on the lookup item.

***Note:*** You cannot use a lookup table on an item in a related file.

### How to use lookup tables to define ranges

Successive records in a lookup table define ranges of values for the lookup item. This makes it easy to assign symbols to features based on a simple classification. Ranges are defined for numeric lookup items in numeric order. Ranges are defined for character lookup items in ASCII number order for each text string (same as alphabetical order).

When ArcCAD display commands read a value for the lookup item in a feature attribute table, they look for a match in the lookup table. If a matching value is not found, the record with the next greater value for the lookup item will be used. For example, this lookup table assigns symbols to features based on their value for a lookup item called RAINFALL:

| RAINFALL | SYMBOL |
|---|---|
| 10 | 17 |
| 25 | 3 |
| 100 | 26 |
| 200 | 2 |

- Features with RAINFALL values less than or equal to 10 are assigned symbol 17.

- Features with RAINFALL values greater than 10 and less than or equal to 25 are assigned symbol 3.

- Features with RAINFALL values greater than 25 and less than or equal to 100 are assigned symbol 26.

- Features with RAINFALL values greater than 100 and less than or equal to 200 are assigned symbol 2.

- Features with lookup item values greater than the largest value in the lookup table will use that record, so in this example, features with RAINFALL values greater than 200 are assigned symbol 2.

Note that the item in the lookup table that stores values from the lookup item must be named and defined as the same item type as the lookup item. In the example above, the item called RAINFALL must be defined as the same item type as the RAINFALL item in the feature attribute table of the theme being displayed. The item storing the symbol numbers must be called SYMBOL but can be defined as any numeric item type. When lookup tables are used to assign text labels to features (see the section 'Labeling features'), the item storing the text labels must be called LABEL but can be defined as any item type. Remember that lookup tables must always be sorted in ascending order on the lookup item.

In this example, the command:

Command: **polyshd**
Theme name: **newzone**
Display using? **item**
Item: **zone**
Optional lookup table name: **zonelut**

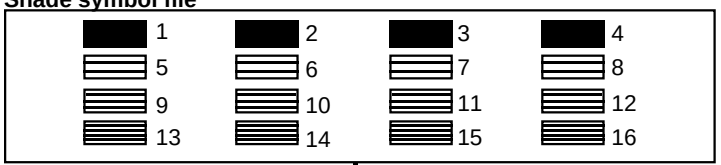
could be used to shade the polygons from the NEWZONE theme using the ZONE item as a lookup item to a lookup table called ZONELUT:

| AREA | PERIMETER | NEWZONE_ | NEWZONE_ID | ZONE |
|---|---|---|---|---|
| 17.463 | 35.625 | 2 | 2 | RA-1 |
| 24.617 | 78.892 | 3 | 3 | RS-1 |
| 12.032 | 23.097 | 4 | 4 | A2-1 |
| 8.920 | 15.901 | 5 | 5 | M2-1 |
| 25.561 | 94.873 | 6 | 6 | RA-1 |
| 20.432 | 54.542 | 7 | 7 | M2-1 |
| 15.117 | 28.859 | 8 | 8 | M2-1 |

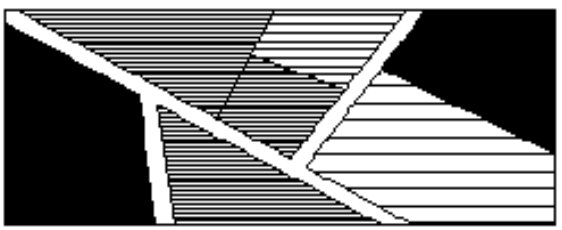
**ZONELUT**

| ZONE | SYMBOL |
|---|---|
| A2-1 | 9 |
| M2-1 | 14 |
| RA-1 | 2 |
| RS-1 | 6 |

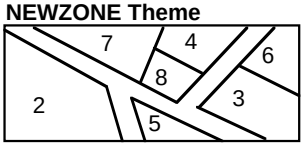
**Shade symbol file**




*Here, **polyshd** uses values from the ZONE item to obtain symbols from a lookup table. The map below shows the NEWZONE_ID for each polygon. (Note that each column in the symbol file represents a different color for the same set of symbols.)*



**NEWZONE Theme**



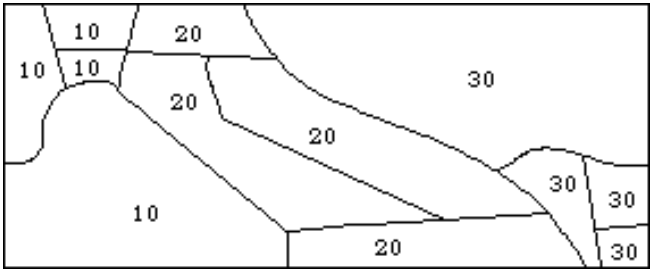
## Using entity property constant values or items

ArcCAD also has the capability of displaying features as entities and explicitly using constants or item values to control the properties of those entities. The **pointmrk**, **labelmrk**, **arclines**, **polyshd** and **polys3d** commands all have a 'properties' option. This allows you to control the properties of entities that are created to represent features by specifying constant values for entity properties. For example, when displaying arc features using **arclines**, all linear entities will be generated on a specified layer if that layer is specified as a property constant. If the layer did not already exist in the drawing, it will be created. Item values can also be used to determine entity properties. For example, if you use **arclines** to display arc features and have an item in the line theme's AAT that contains a value representing the thickness of each feature, the thickness property for linear entities will be set by the value of this item. The specified item can be in the theme's feature attribute table or can be in a related file (as established with **relate**). If your GIS data set was created from a drawing, using items in a related file allows you to use the property table saved by any of the feature creation commands (**addfeat**, **savefeat** or **modfeat**) to control entity creation. This way, entities representing features will be created with the same entity properties they had in the original drawing.

The entity properties that you can control differ for each feature class. For example, since **pointmrk** displays marker symbols (as blocks) for point features, you can control the angle and x and y scale factors for the marker symbol blocks. Refer to the command references for each command for specific information on which entity properties can be controlled through constants or item values. A special case is the **arclines** command. **Arclines** allows you to generate true curves from features. When specifying entity properties in the **arclines** command, you can specify an item that controls whether a feature will be generated as an AutoCAD arc entity (or circle entity) or as a polyline. This way you can generate a true curve entity from a curved feature rather than generating a polyline that approximates that curved feature. Refer to the **arclines** command reference for more information.

## The dropline command

**Dropline** is a special display command which suppresses the drawing of arc features that separate polygons having the same value for a specified attribute. Polygon boundaries (arc features) between polygons that have the same value for the attribute are not drawn.




*A theme called STAND is shown labeled with the item values for an item called CODE. Boundaries between polygons having the same values for CODE will be dropped using the* ***dropline*** *command.*

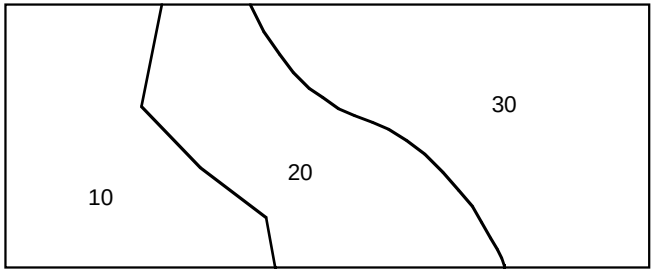
The command is given as

Command: **dropline**
Theme name: **stand**
Item: **code**

Notice that only bounding polygons with different CODE values are displayed.

**STAND PAT**

| AREA | PERIMETER | STAND_ | STAND_ID | CODE |
|---|---|---|---|---|
| 4.854 | 8.625 | 2 | 2 | 30 |
| 3.675 | 8.212 | 3 | 3 | 10 |
| 5.231 | 11.967 | 4 | 4 | 30 |
| 3.744 | 8.912 | 5 | 5 | 10 |
| 2.626 | 3.107 | 6 | 6 | 30 |
| 8.126 | 15.908 | 7 | 7 | 20 |
| 21.393 | 43.665 | 8 | 8 | 20 |
| 34.756 | 57.492 | 9 | 9 | 10 |
| 41.671 | 67.453 | 10 | 10 | 30 |
| 19.873 | 45.610 | 11 | 11 | 20 |
| 6.661 | 12.029 | 12 | 12 | 10 |
| 13.123 | 24.781 | 13 | 13 | 20 |




*After displaying the polygon features with **dropline**, polygon boundaries are displayed only between polygons with different CODE values.*

# Labeling features

Any item from theme feature attribute tables or related record themes can be used to supply text labels for features that have been displayed. A lookup table can also be used to assign text labels to different features.

The **arctxt**, **labeltxt**, **pointtxt** and **polytxt** commands are used to label theme features.

## Using constant values to label features

**Arctxt** adds labels to arc features. Various options are provided for positioning the text relative to the arc features. **Labeltxt** adds text labels to label points. **Pointtxt** labels point features. Various options are provided for positioning the text relative to the label point or point feature. **Polytxt** labels polygons. These commands can be used in two ways.

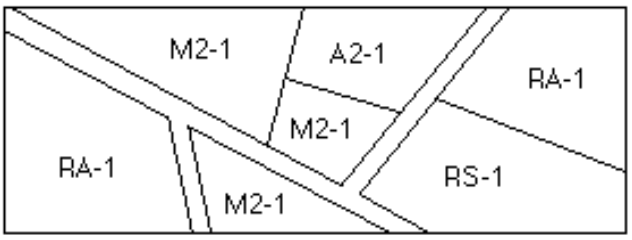
In the first method, you specify any item from a theme's feature attribute table. The values of this item will be used as text to label the features. The item used to supply text labels can be of either character or numeric item type. The command

Command: **polyshd**
Theme name: **newzone**
Item: **zone**

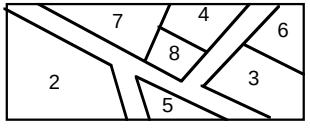
can be used to label each polygon in the NEWZONE theme with its value from the ZONE item in the PAT. The labels will be drawn inside each polygon:

**NEWZONE PAT**

| AREA | PERIMETER | NEWZONE_ | NEWZONE_ID | ZONE |
|---|---|---|---|---|
| 9.354 | 11.509 | 1 | 1 | |
| 17.463 | 35.625 | 2 | 2 | RA-1 |
| 24.617 | 78.892 | 3 | 3 | RS-1 |
| 12.032 | 23.097 | 4 | 4 | A2-1 |
| 8.920 | 15.901 | 5 | 5 | M2-1 |
| 25.561 | 94.873 | 6 | 6 | RA-1 |
| 20.432 | 54.542 | 7 | 7 | M2-1 |
| 15.117 | 28.859 | 8 | 8 | M2-1 |




**NEWZONE theme**




*In this example, **polytxt** uses values from the ZONE item to label polygons. The map shows the NEWZONE_ID for each polygon.*

# Using lookup tables to label features

The second method uses a lookup table to assign text labels to features. This method is more flexible than the first because you can assign any text labels in the lookup table, rather than being limited to using the data values already stored in the attribute tables.

A lookup table is a type of record theme. The record theme's GIS data set is a database file which contains information needed to symbolize features. Any item from a theme's feature attribute table can be used as the lookup item to a lookup table. The lookup table stores values from this lookup item. The lookup table also stores text labels. To label each theme feature, ArcCAD reads its value for the lookup item in the feature attribute table and then finds this value in the lookup table to obtain the associated text label.

Lookup tables can contain a value for every possible value for the theme's feature attribute table or, more typically, will contain representative values for this attribute that define a range of values in the lookup table. Successive records in the lookup table can be used to specify ranges of values from the lookup item. Thus, the lookup table can be defined with explicit ranges. When ArcCAD text display commands read the value of the lookup item in the feature attribute table, they will look for an exact match in the lookup table. If they do not find a match, they will use the next value greater than the lookup item value to determine the appropriate record in the lookup table. Lookup tables must be sorted in ascending order on the lookup item value for this feature to work. If the lookup item value is greater than the largest value in the lookup table, it will use the last record in the table. (The section 'Displaying theme features' in this chapter contains more information on using lookup tables to define ranges.)

## How to create a lookup table

As mentioned earlier, a lookup table is a type of record theme. A lookup table consists of at least two items, the first item being the lookup item. This item is used to relate the lookup table to the theme AAT or PAT. It stores values from the corresponding item in the theme feature attribute table. This item must be defined as the same item type as it is in the feature attribute table and is commonly the coverage User-ID.

The second item must be named LABEL. This item stores the label text to be used for each value in the lookup item. LABEL can be defined as any ArcCAD supported item type.

***Note:*** Always keep the lookup table sorted in ascending order by the lookup item.

In this example, the command

Command: **polytxt**
Theme name: **zone**
Item: **cost**
Optional lookup table name: **costlut**

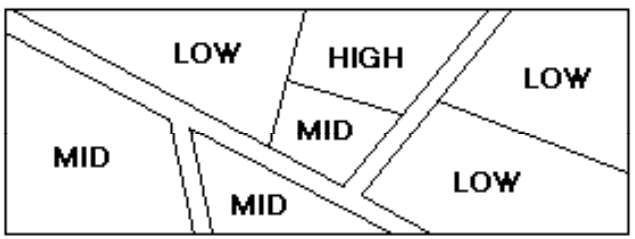
will label each polygon in the ZONE coverage using the COST item in the PAT as a lookup item to a lookup table, called COSTLUT, containing the label text for the polygons. See the following diagram.

**NEWZONE PAT**

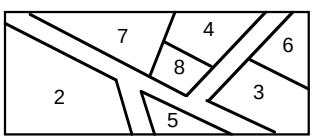
| AREA | PERIMETER | ZONE_ | ZONE_ID | COST |
|---|---|---|---|---|
| 9.354 | 11.509 | 1 | 1 | |
| 17.463 | 35.625 | 2 | 2 | 325 |
| 24.617 | 78.892 | 3 | 3 | 25 |
| 12.032 | 23.097 | 4 | 4 | 890 |
| 8.920 | 15.901 | 5 | 5 | 230 |
| 25.561 | 94.873 | 6 | 6 | 95 |
| 20.432 | 54.542 | 7 | 7 | 40 |
| 15.117 | 28.859 | 8 | 8 | 210 |

**COSTLUT**

| COST | LABEL |
|---|---|
| 100 | LOW |
| 500 | MID |
| 1000 | HIGH |



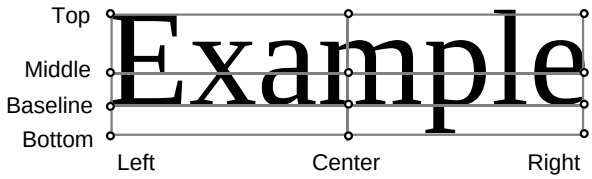

**NEWZONE theme**




*Here, **polytxt** uses values from the COST item to obtain labels from a lookup table. Note that if the COST value does not match a value in the lookup table, the next greater value is used. The map shows the ZONE_ID for each polygon.*

**Pointtxt**, by default, draws the lower-left point of text labels to the upper right of each point feature. You may also position text around the point feature by specifying a positioning parameter. Text positioning parameters follow the same standard as the justification option in the AutoCAD **text** command.

**Labeltxt** draws the lower-left point of text labels to the upper right of each label point inside a polygon. You may also position text labels around the polygon label point by specifying a positioning parameter. Text positioning parameters follow the same standard as the justification option in the AutoCAD **text** command. Polygons that have more than one label point will have text labels displayed at each label point.




*AutoCAD text alignment positions*

**Polytxt** labels polygons by finding the best position inside each polygon to neatly fit the text label. Text labels that do not fit neatly inside a polygon are drawn to the upper right of the polygon's label point.

By default, **arctxt** positions text labels to the upper right of the arc feature midpoints. However, **arctxt** features a set of options for controlling the placement and orientation of text labels. Refer to the **arctxt** command reference for more information.

# Annotation

Annotation is a special feature class that stores text labels. Annotation themes store the text strings used in the labels, the text symbol numbers used to draw them, and their locations. Annotation also stores the height of the text labels. This means that when you display text annotation, their height can be recalculated as a function of their original text size and the current zoom window. In this way, when you draw annotation, it is automatically scaled, positioned and proportioned correctly in relation to other features.

You can organize annotation into annotation levels. For example, an annotation theme storing text labels for roads may have street names in one annotation level, highway names in another level, and place-names in a third level. For different map products based on this theme, only one of these annotation levels may have to be drawn.

Any text entities can be turned into annotation including text created using the ArcCAD text labeling commands. To create annotation, define an annotation theme using **defthm** and use the feature creation commands (**addfeat** or **savefeat**) to make annotation features from the selected text entities. The

annotation symbol and annotation level are established during feature creation. Refer to the section 'Creating themes from drawings' in the 'Database automation' chapter for more information on creating annotation.

***Note:*** ArcCAD ignores the text justification and always stores the lower-left corner of the text string as the insertion point.

Annotation is displayed using the **annotxt** command. The text height and angle of the annotation is stored with the GIS data set and overrides the current AutoCAD **textsize** and the HEIGHT and ANGLE text symbol parameters from the ArcCAD symbolset. The STYLE, COLOR and SCALE parameters for the text are obtained from the current ArcCAD symbolset file (refer to the section 'Text symbols' below for more information on ArcCAD text symbol parameters). The insertion point of the text string will always be the lower-left corner of the annotation feature. If you want to change the insertion point of the text string for an annotation feature, you can use the AutoCAD **change** command to change these properties and then update the corresponding features using **modfeat**.

# Displaying images

Use the **image** command to display an image theme. By entering a valid image theme name, the image is displayed at the coordinates 0,0 if a RAT or World file does not exist. Only one image theme can be displayed at a time. If you display a new image theme while one is already displayed, the current image will be dismissed before the new image theme is displayed. If you want to remove an image from the backdrop, use the **iclose** command to dismiss the image from the screen.

## Using image configuration files

You can apply a number of basic operations to image themes. The command **iconfig** controls these operations. A file that has the same name as the image with an extension .RAT saves the information obtained by this command. An ARC/INFO-compatible World file stores the same information. For more details, refer to the **iconfig** command in the *ArcCAD Command Reference*.

# ArcCAD symbol management

Displaying maps with ArcCAD involves specifying symbols. There are four groups of symbols: marker symbols, line symbols, shade symbols, and text symbols. Whenever you display features using an ArcCAD display command, ArcCAD automatically uses the appropriate symbol group. Point features and polygon label points are displayed using marker symbols. Arc features are displayed using line symbols. Polygon features are filled in using shade symbols. Annotation features and text labels are displayed with text symbols.

Features are rendered using a combination of both the ArcCAD and AutoCAD symbology systems. AutoCAD commands use the AutoCAD symbology system (color, linetype, etc.). ArcCAD display commands use an independent method which combines the standard AutoCAD symbology components into symbolsets. Refer to the *AutoCAD User's Guide* for complete information about the AutoCAD entity property management.

The symbols in each group are defined by a number of characteristics, such as pattern, color and size, which control the appearance of the symbols. These characteristics are stored in database files called symbolset files. Each symbolset file is a dBASE® database file containing up to 100 records. Each record in the symbolset file defines a symbol. The record number of each symbol in the symbolset file is that symbol's symbol number. A default symbolset is provided with ArcCAD. The default symbolset (called ARCAD) is stored in the ARCAD\SYMBOLS subdirectory. You can create your own customized sets of symbols and save them for use in later ArcCAD sessions.

While there are only four groups of symbols to think about while you are working with ArcCAD, the number of different symbol designs you can create for your maps is practically unlimited.

Each symbol type will be discussed in detail in the following sections.

## Marker symbols

Each ArcCAD marker symbol is an AutoCAD block. Marker symbol definitions are stored in a markerset file. A markerset file is a database file that stores characteristics of marker symbols. Each markerset file contains 100 symbols. The default markerset file is called ARCAD.MRK. This file is stored as the file MRK.DBF under the ARCAD\SYMBOLS\ARCAD subdirectory.

Markerset files contain the following items that define the characteristics of each symbol:

| Item name | Item width | Item type | Number of decimals |
|---|---|---|---|
| NAME | 31 | C | 0 |
| COLOR | 4 | N | 0 |
| XSIZE | 16 | N | 6 |
| YSIZE | 16 | N | 6 |
| ANGLE | 16 | N | 6 |
| SCALE | 1 | C | 0 |

■ NAME is an AutoCAD block name. This block must exist in the current AutoCAD drawing. The ArcCAD prototype drawing (ARCAD.DWG) contains all the block definitions (ESRI_M1–ESRI_M16) for the default markerset file. A drawing file called MARKERS.DWG located in the ARCAD\SYMBOLS directory also contains the block definitions for the default markerset file. You can insert this drawing into the current drawing if you are not using ARCAD.DWG as your prototype drawing.

■ COLOR is the AutoCAD color number of this marker symbol. COLOR must be a number between 1 and 255. Colors 0 and 256 have special meanings in ArcCAD symbology as follows:

- Color 0 sets the marker symbol color to BYBLOCK. This means that the marker symbol block will inherit its color from the current AutoCAD settings (as set with the AutoCAD COLOR command) when it is inserted.

- Color 256 sets the marker symbol color to BYLAYER. This means that the marker symbol block will inherit its color from the layer on which it is inserted.

It is not recommended to use colors 0 and 256 as this may produce confusing and undesirable results.

■ XSIZE is the X scale factor for this marker symbol block. The effect of this value depends on the value of SCALE for this symbol. If SCALE is ‘S’ (static), then the XSIZE value is taken as the X scale factor for this block. If SCALE is ‘D’ (dynamic), then the X scale factor of this block is determined using the following equation:

XSIZE = 1/VIEWSIZE

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units.

This allows the X scale factor for the marker symbols to be scaled to the current zoom window.

■ YSIZE is the Y scale factor for this marker symbol block. The effect of this value depends on the value of SCALE for this symbol. If SCALE is 'S' (static), then the YSIZE value is taken as the Y scale factor for this block. If SCALE is 'D' (dynamic), then the Y scale factor of this block is determined using the following equation:

YSIZE = 1/VIEWSIZE

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units.

This allows the Y scale factor for the marker symbols to be scaled to the current zoom window.

■ ANGLE is the rotation angle for this marker symbol block. The marker symbol will be rotated by this angle when displayed.

■ SCALE is scaling type applied to this marker symbol block. SCALE is a single character and can either be dynamic ('D') or static ('S'). If SCALE is dynamic, the size of the marker symbol block is determined as follows:

XSIZE = 1/VIEWSIZE, YSIZE = 1/VIEWSIZE

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units. When the block is inserted into the current drawing, the block's X and Y scale factors are changed based on the above equation.

If SCALE is static, the XSIZE and YSIZE values are used as the block's X and Y scale factors.

If the SCALE is dynamic, the block's X and Y SCALE factors are automatically adjusted (based on the VIEWSIZE) such that the block size is always proportional to the screen size. To achieve this, it is very important to specify the correct (proportional) value in the XSIZE and YSIZE items in the symbolset file. To calculate the XSIZE and YSIZE values, follow the procedure described below:

1) Insert the block (in the original units that were used to define the block) with X and Y SCALE factors set to 1.

2) Adjust the zoom window (using the **zoom** command) such that the block is proportional to the screen size.

3) Use the **viewsize** command to determine the current window height.

4) Calculate XSIZE = 1/VIEWSIZE.

5) Calculate YSIZE = 1/VIEWSIZE.

***Note:*** If the original block is defined in a different coordinate system, you must insert the block with the appropriate X and Y SCALE factors. Explode the block and redefine it before calculating the VIEWSIZE.

## Modifying existing marker symbol parameters

Any of the items in the current markerset file may be modified at any time in an ArcCAD session. This can be done by defining a record theme to point to the markerset file and using **modrec** to modify item values. To use the updated markerset file in the current ArcCAD session, use **symload** to reload that markerset file into memory.

## Creating your own marker symbols

The following steps can be used to create your own marker symbols for symbolizing point features and polygon label points:

#### Step 1 Create an AutoCAD block that represents the desired symbol

- The block may contain any number of entities.

- All the AutoCAD block creation rules apply to marker symbols. Refer to the *AutoCAD User's Guide* for further details on blocks. The best way to create multicolored marker symbols is to explicitly define the entity colors of each component of the marker symbol block.

- The block's insertion point will become the marker's position location.

- If the marker symbol (or symbols) created using this block will have dynamic scaling (item SCALE = 'D'), the size of the block can be any size as long as it looks proportional to the size of the screen.

- If the marker symbol (or symbols) created using this block will have static scaling (item SCALE = 'S'), the block should be created in proportion to the scale of the working (current) drawing units. For example, if you are working at a scale of 1 inch to 100 feet and you want to plot a symbol 0.25 inches in size, you have two methods of creating the marker symbol block: (1) Make the block 25 feet in size, set XSIZE and YSIZE to 1 and SCALE to 'S' or (2) Make the block 1 foot in size, set XSIZE and YSIZE to 25 and scale to 'S'.

#### Step 2 Create a new markerset file

A database file template containing the markerset file item definitions is located in the ARCAD\TEMPLATE\SYMBOLS directory. The markerset template file is called MRK.DBF. You need to create a copy of this template in your current directory to store the new markerset parameters:

1) Define a record theme that points to the markerset template database file in the ARCAD\TEMPLATE\SYMBOLS directory using the **defthm** command. For example:

Command: **defthm**
Theme name: **mrk_template**
Feature class: **record**
GIS data set: **\arcad\template\symbols\mrk**

2) Use the **copythm** command to copy this theme to your current directory. You will use the output theme created by **copythm** to access your new markerset file. For example:

First define the output theme:

Command: **defthm**
Theme name: **my_sym**
Feature class: **record**
GIS data set: **my_sym\mrk**

Then copy the template to your current directory:

Command: **copythm**
Input theme : **mrk_template**
Output theme: **my_sym**

***Note:*** By default, ArcCAD searches the \ARCAD\SYMBOLS directory for symbolset files. You can also place your symbolset files under this directory so that they will be found by **symload**.

#### Step 3 Add a new record to your markerset file

This record will contain the characteristics of your new marker symbol. Use the **addrec** command to add records to your new markerset file. The markerset file can be accessed using the record theme created by the **copythm** operation in Step 2. The symbol number used to access your new symbol is the same as the record number in the markerset file. Be sure to fill in all item values for the new record.

**Step 4 Load the new markerset file using the symload command**

Use the **symload** command to load your new markerset file into memory. To display features using the new marker symbol, display features using that symbol number.

***Note:*** You must have the appropriate block definitions in your current drawing.

# Line symbols

Each line symbol uses standard AutoCAD line symbology. Line symbol definitions are stored in a lineset file. A lineset file is a database file that stores characteristics of line symbols. Each lineset file contains 100 symbols. The default lineset file is called ARCAD. This file is stored as the file LIN.DBF under the ARCAD\SYMBOLS\ARCAD subdirectory. Lineset files contain the following items that define the characteristics of each symbol:

| Item name | Item width | Item type | Number of decimals |
|---|---|---|---|
| TYPE | 31 | C | 0 |
| COLOR | 4 | N | 0 |
| WEIGHT | 16 | N | 6 |
| SCALE | 1 | C | 0 |

■ TYPE is an AutoCAD linetype name. This linetype can either be a standard or custom linetype and must be loaded in the current AutoCAD drawing. The AutoCAD **linetype** command can be used to load linetypes. Refer to the *AutoCAD User's Guide* for information on linetypes.

■ COLOR is the AutoCAD color number of this line symbol. COLOR must be a number between 1 and 255. Colors 0 and 256 have special meanings in ArcCAD symbology as follows:

- Color 0 sets the color to BYBLOCK. This means that the line will inherit its color from the current AutoCAD settings.

- Color 256 sets the color to BYLAYER. This means that the line will inherit its color from the layer on which it is drawn.

It is not recommended to use colors 0 and 256 as this may produce confusing and undesirable results.

■ WEIGHT is the line width used in polylines. AutoCAD linetypes do not store the width of a line. You must explicitly change the width factor to display thick lines. Using ArcCAD symbology, you can set the WEIGHT to a value greater than 0 to display polylines with appropriate width factors.

■ SCALE is scaling type applied to this line symbol. SCALE is a single character and can either be dynamic ('D') or static ('S'). If SCALE is dynamic, the line width of polylines is determined as follows:

VIEWSIZE * WEIGHT

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units.

If SCALE is set to dynamic ('D'), it is advisable to use **setext** on the theme whose features you wish to display before displaying those features using an ArcCAD display command.

If SCALE is static, the WEIGHT factor is used as the width of polylines.

## Modifying existing line symbol parameters

Any of the items in the current lineset file may be modified at any time in an ArcCAD session. This can be done by defining a record theme to point to the lineset file and using **modrec** to modify item values. To use the updated lineset file in the current ArcCAD session, use **symload** to reload that lineset file into memory.

## Creating your own line symbols

The following steps can be used to create your own line symbols for use in symbolizing arc features:

#### Step 1 Create a new lineset file

A database file template containing the lineset file item definitions is located in the ARCAD\TEMPLATE\SYMBOLS directory. The lineset template file is called LIN.DBF. You need to create a copy of this template in your current directory to store the new lineset parameters:

1) Define a record theme that points to the lineset template database file in the ARCAD\TEMPLATE\SYMBOLS directory using the **defthm** command.

2) Use the **copythm** command to copy this theme to your current directory. You will use the output theme created by **copythm** to access your new lineset file.

#### Step 2 Add a new record to your lineset file

This record will contain the characteristics of your new line symbol. Use the **addrec** command to add records to your new lineset file. The lineset file can be accessed using the record theme created by the **copythm** operation in Step 2. The symbol number used to access your new symbol is the same as the record number in the lineset file. Be sure to fill in all item values for the new record.

#### Step 3 Load the new lineset file using the symload command

Use the **symload** command to load your new lineset file into memory. To display features using the new line symbol, display features using that symbol number.

***Note:*** Be sure that the AutoCAD linetype referenced by your symbol is loaded in the current drawing.

## Shade symbols

Each shade symbol is an AutoCAD hatch pattern. Shade symbol definitions are stored in a shadeset file. A shadeset file is a database file that stores characteristics of shade symbols. Each shadeset file contains 100 symbols. The default shadeset file is called ARCAD. This file is stored as the file SHD.DBF under the ARCAD\SYMBOLS\ARCAD subdirectory. Shadeset files contain the following items that define the characteristics of each shade symbol:

| Item name | Item width | Item type | Number of decimals |
|---|---|---|---|
| PATTERN | 31 | C | 0 |
| COLOR | 4 | N | 0 |
| SIZE | 16 | N | 6 |
| ANGLE | 16 | N | 6 |
| SCALE | 1 | C | 0 |

■ PATTERN is an AutoCAD hatch pattern name. This hatch pattern can either be a standard or custom pattern. ArcCAD symbology ignores any AutoCAD hatching styles.

- COLOR is the AutoCAD color number of this shade symbol. COLOR must be a number between 1 and 255. Colors 0 and 256 have special meanings in ArcCAD symbology as follows:

  - Color 0 sets the color to BYBLOCK. This means that the hatch will inherit its color from the current AutoCAD settings.

  - Color 256 sets the color to BYLAYER. This means that the hatch will inherit its color from the layer on which it is drawn.

It is not recommended to use colors 0 and 256 as this may produce confusing and undesirable results.

- SIZE is the scale factor for this shade symbol. The effect of this value depends on the value of SCALE for this symbol. If SCALE is 'S', then the SIZE value is taken as the scale factor for hatching this shade symbol. If SCALE is 'D', then the scale factor for hatching this shade symbol is determined using the following equation:

VIEWSIZE * SIZE

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units. This allows the scale factor for hatching a shade symbol to be scaled to the current zoom window.

- ANGLE is the rotation angle for the hatch pattern.

- SCALE is the scaling type applied to hatching this shade symbol. SCALE is a single character and can either be dynamic ('D') or static ('S'). If SCALE is dynamic, the scale for hatching this shade symbol is dependent on the current zoom window and is determined as follows:

VIEWSIZE * SIZE

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units.

If SCALE is static, the value of SIZE is used as the scale for hatching this symbol. Note that you must be careful using static scaling for shade symbols. For example, if you specify a SIZE of 0.1, this will display correctly in page coordinates but will take a very long time to draw when displayed in real-world coordinates.

## Modifying existing symbol parameters

Any of the items in the current shadeset file may be modified at any time in an ArcCAD session. This can be done by defining a record theme to point to the shadeset file and using **modrec** to modify item values. To use the updated shadeset file in the current ArcCAD session, use **symload** to reload that shadeset file into memory.

## Creating your own shade symbols

The following steps can be used to create your own shade symbols for use in symbolizing polygon features:

#### Step 1 Create a new shadeset file

A database file template containing the shadeset file item definitions is located in the ARCAD\TEMPLATE\SYMBOLS directory. The shadeset template file is called SHD.DBF. You need to create a copy of this template in your current directory to store the new shadeset parameters:

1) Define a record theme that points to the shadeset template database file in the ARCAD\TEMPLATE\SYMBOLS directory using the **defthm** command.

2) Use the **copythm** command to copy this theme to your current directory. You will use the output theme created by **copythm** to access your new shadeset file.

#### Step 2 Add a new record to your shadeset file

This record will contain the characteristics of your new shade symbol. Use the **addrec** command to add records to your new shadeset file. The shadeset file can be accessed using the record theme created by the **copythm** operation in Step 2. The symbol number used to access your new symbol is the same as the record number in the shadeset file. Be sure to fill in all item values for the new record.

#### Step 3 Load the new shadeset file using the symload command

Use the **symload** command to load your new shadeset file into memory. To display features using the new line symbol, display features using that symbol number.

# Text symbols

Text symbols in ArcCAD access AutoCAD text styles to display annotation and label features. Text symbol definitions are stored in a textset file. A textset file is a database file that stores characteristics of text symbols. Each textset file contains 100 symbols. The default textset file is called ARCAD. This file is stored as the file TXT.DBF under the ARCAD\SYMBOLS\ARCAD subdirectory. Textset files contain the following items that define the characteristics of each text symbol:

| Item name | Item width | Item type | Number of decimals |
|---|---|---|---|
| STYLE | 31 | C | 0 |
| COLOR | 4 | N | 0 |
| HEIGHT | 16 | N | 6 |
| ANGLE | 16 | N | 6 |
| SCALE | 1 | C | 0 |

■ STYLE is an AutoCAD text style definition. This style can either be a standard or custom style and must be loaded in the current AutoCAD session. A component of an AutoCAD text style is text font. Text symbols use the text font associated with the specified STYLE.

■ COLOR is the AutoCAD color number of this text symbol. COLOR must be a number between 1 and 255. Colors 0 and 256 have special meanings in ArcCAD symbology as follows:

- Color 0 sets the color to BYBLOCK. This means that the text will inherit its color from the current AutoCAD settings.

- Color 256 sets the color to BYLAYER. This means that the text will inherit its color from the layer on which it is drawn.

It is not recommended to use colors 0 and 256 as this may produce confusing and undesirable results.

■ HEIGHT is the height of the text entity. The effect of this value depends on the value of SCALE for this symbol. If SCALE is ‘S’, then the HEIGHT value is taken as the height of the text. If SCALE is ‘D’, then the height for this text entity is determined using the following equation:

VIEWSIZE * HEIGHT

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units. This allows the height of text symbols to be scaled to the current zoom window.

■ ANGLE is the rotation angle of the text symbol. The rotation angle specifies the orientation of the text baseline with respect to the starting point of the text entity.

■ SCALE is scaling type applied to this text symbol. SCALE is a single character and can either be dynamic ('D') or static ('S'). If SCALE is dynamic, the scale for this text symbol is dependent on the current zoom window and is determined as follows:

VIEWSIZE * HEIGHT

where VIEWSIZE is the current height of the AutoCAD display screen in drawing units.

If SCALE is static, the value of HEIGHT is used as the text height for this symbol. HEIGHT overrides the current AutoCAD textsize.

## Modifying existing symbol parameters

Any of the items in the current textset file may be modified at any time in an ArcCAD session. This can be done by defining a record theme to point to the textset file and using **modrec** to modify item values. To use the updated textset file in the current ArcCAD session, use **symload** to reload that textset file into memory.

## Creating your own text symbols

The following steps can be used to create your own text symbols for use in symbolizing annotation features and labeling theme features:

#### Step 1 Create a new textset file

A database file template containing the textset file item definitions is located in the ARCAD\TEMPLATE\SYMBOLS directory. The textset template file is called TXT.DBF. You need to create a copy of this template in your current directory to store the new textset parameters:

1) Define a record theme that points to the textset template database file in the ARCAD\TEMPLATE\SYMBOLS directory using the **defthm** command.

2) Use the **copythm** command to copy this theme to your current directory. You will use the output theme created by **copythm** to access your new textset file.

#### Step 2 Add a new record to your textset file

This record will contain the characteristics of your new text symbol. Use the **addrec** command to add records to your new textset file. The textset file can be accessed using the record theme created by the **copythm** operation in Step 2. The symbol number used to access your new symbol is the same as the record number in the textset file. Be sure to fill in all item values for the new record.

#### Step 3 Load the new textset file using the symload command

Use the **symload** command to load your new textset file into memory. To display text using the new text symbol, display features or label theme features using that symbol number.

The following general notes apply to ArcCAD symbology:

- Symbol number 0 is the default for each of the symbol types with the exception of text symbols. Symbol 0 is a special symbol that uses the most current AutoCAD settings. Note that ArcCAD symbolization does not permanently affect the current AutoCAD default settings.

- The current symbol environment is stored as part of a drawing.

- Tic themes and nodes have fixed symbology. You cannot change the symbols that are used to display tics or nodes.

- If any symbolset parameter in a symbolset file is modified, you must run the **symload** command on that symbolset file. This will load the modified version of the symbolset file into memory so that the modified parameters can be used.

- If you plan to combine both static and dynamically scaled symbols in a symbolset file, you must be cautious in determining the value for the size of the symbol (SIZE, HEIGHT, etc.). The size of the symbol must be very small (i.e., around 0.01) if the symbol is dynamically scaled. The size of the symbol must be larger (depending on whether you are working in page coordinates or real-world coordinates) if you are using static symbols.

- Dynamically scaled symbols are most useful if you are not interested in plotting the symbols to a specific scale and if you are only using the symbol for query and display purposes. The size of dynamically scaled symbols varies with the zoom extent.

■ Static symbols are most useful for plotting symbols to a specific scale. This enables you to specify a value for that symbol's size in the symbolset's size item (item SIZE, HEIGHT, etc.) based on the scale of the drawing. For example, to plot a drawing at 1:100 scale and have text symbols be 0.25 inches in size, specify a value of 25 for HEIGHT for that text symbol by modifying the appropriate record in the textset file. Symbolset files can be customized by using certain symbol size values for plotting at specific scales. In this way, the symbolset file need only be changed when a change in the scale of symbols is required in a drawing.

## Symbol management commands

Two commands are provided to load and display symbols. The **symload** command loads symbol definitions from a symbolset file into memory for use in the current ArcCAD session. The default symbolset (ARCAD) is loaded automatically when ArcCAD starts. You should use **symload** to load custom symbolset files or to reload a current symbolset file after changes have been made to it.

The **syminfo** command can be used to display records from symbolset files. This allows you to see the symbol characteristics of a particular symbol.

## Symbol index commands

ArcCAD has four commands that display icon menus of the four symbol types. The commands **mrkindex**, **linindex**, **shdindex** and **txtindex** display icon menus containing all 100 of the default marker, line, shade and text symbols, respectively, from the default (ARCAD) symbolset files. These commands are only active from the ArcCAD pull-down menu; they cannot be typed in at the Command: prompt. Selecting one of these commands and picking a symbol at an ArcCAD Symbol: prompt will input that symbol number into the current ArcCAD command. This allows you to visually preview and select an appropriate symbol when ArcCAD prompts for a symbol. Refer to the appropriate command references for more information.

# Querying features

In order to manipulate entities and features for both editing and display, you must be able to indicate which objects are to be considered. ArcCAD provides commands that allow you to combine spatial, graphical, and attribute criteria specified as a series of logical expressions to create entity and feature selection sets. You can then display, edit or retrieve information related to these objects.

## Selecting features to draw

ArcCAD provides several commands for creating *feature selection sets*. A feature selection set is a subset of features in a particular theme. You can have one feature selection set for a given theme at any one time. The commands **reselect**, **aselect**, **nselect** and **clearsel** are used to create and manipulate feature selection sets. Only those features in the selected set will be displayed or labeled using the ArcCAD display commands.

You can select features according to their attributes or according to their location. Selecting features for display lets you control precisely what is shown on your maps. The commands used for selecting features look similar to those used for querying a database file (refer to the section 'Querying a database file' in the 'Database automation' chapter). In fact, when theme features are selected using logical expressions, you are applying those logical expressions against that theme's associated database file. You can then display only those features that meet the selection criteria.

The **reselect** command is used for the initial selection of features. The **aselect** command is used for adding more features to the selected set. The **nselect** command replaces the selected set with all the features that have not been selected. The **clearsel** command clears all the feature selection sets created during the ArcCAD session and makes all the features in each theme accessible for display again. If you do not use any of these feature selection commands, all the features of a theme will be accessible for display.

## Attribute selection of features

Let's say the ROADS theme has an arc attribute table item called WIDTH that stores the width of each road feature in meters. If we want to make a map showing only those road features representing roads wider than 10 meters, the following commands could be used to accomplish this task:

Command: **reselect**
Theme name: **roads**
Reselect by? **expression**
Logical expression: **width gt 10**

Command: **arcs**
Theme name: **roads**

This selection includes only those arc features in the ROADS theme that have a value greater than (abbreviated as 'gt') 10 for the item called WIDTH in the arc attribute table. ArcCAD carries out the selection and tells you how many features fall into the selected set of arc features from this theme based on your selection criterion. The **arcs** command will only draw those arc features from the ROADS theme that fall into this selected set.

We may have a theme called LANDUSE featuring a character item called TYPE that stores a text string describing what land use type is represented by each polygon. To make a map showing just those polygons that represent industrial land use, we could give these commands:

Command: **reselect**
Theme name: **landuse**
Reselect by? **expression**
Logical expression: **type eq 'INDUSTRIAL'**

Command: **polyshd**
Theme name: **landuse**
Display using?: **symbol**
Shade symbol number: **12**

Only those polygons in the LANDUSE theme that have a text string 'INDUSTRIAL' for the item called TYPE in the polygon attribute table are selected. **Polyshd** will only shade those polygons in this selected set. We could add to this map by giving these commands:

Command: **nselect**
Theme name: **landuse**

Command: **polyshd**
Theme name: **landuse**
Display using?: **symbol**
Shade symbol number: **4**

**Nselect** is used to swap the selected and unselected sets. The selected set for this theme now contains all those polygons that do not represent industrial land use. Polygons in this newly selected set are then shaded with a different shade symbol. This gives us a map showing whether the polygons in the LANDUSE theme represent industrial land use or not.

In these examples, the selection criteria WIDTH GT 10 and TYPE EQ 'INDUSTRIAL' are known as logical expressions. Those features for which the logical expression is true become part of the selected set.

## Logical expressions

Logical expressions in ArcCAD commands have three components: operands, logical operators and logical connectors.

### Logical operands

- The name of an item in a data file (e.g., `STREAMS_ID`)
- A constant numerical value (e.g., `10`)
- A character string in single quotation marks (e.g., `'HIGH'`)
- An internal variable (e.g., `$RECNO`)

### Logical operators

- `EQ` or `=` Operand-1 is equal to Operand-2.
- `NE` or `<>` Operand-1 is not equal to Operand-2.
- `GE` or `>=` Operand-1 is greater than or equal to Operand-2.
- `LE` or `<=` Operand-1 is less than or equal to Operand-2.
- `GT` or `>` Operand-1 is greater than Operand-2.
- `LT` or `<` Operand-1 is less than Operand-2.
- `CN` Operand-1 contains the character expression Operand-2. Character operands only (e.g., `NAME CN 'MAIN'`).
- `NC` Operand-1 does not contain the character expression Operand-2. Character operands only (e.g., `NAME NC 'MAIN'`).
- `IN` Operand-1 is contained in the set of numeric constants or character strings specified in Operand-2. This set of constants or character strings must be enclosed in { } brackets. The individuals in the set must be separated by commas, unless they are being used to express a range, in which case, -> is used to separate the values forming the lower- and upper-inclusive limits of the range. A range defined between two character strings is based on the ASCII

number sequence, which is alphabetical. No blank spaces should separate any of the elements within the brackets.

***Note:*** Computer roundoff can alter the values of real numbers. This can cause a problem when specifying real numbers in a [logical expression] that require equality. When using expressions of equality, the operands must match exactly for a match to be found. For example, the value .01139 does not equal .0114. In such cases, use an expression that includes a range of real values (i.e., "HEIGHT GT .01139 AND HEIGHT LT .01141").

### Logical connectors

- AND — For the condition to be evaluated as true, the logical expressions on both sides of the AND must be true.
- OR — For the condition to be evaluated as true, the logical expression on one or the other side of the OR must be true. The condition will also be evaluated as true if both logical expressions are true.
- XOR — For the condition to be evaluated as true, the logical condition on one and only one side of the XOR must be true. If both logical expressions are true or both are false, the condition will be evaluated as false.

The simplest logical expressions take the following form:

[operand-1] [logical-operator] [operand-2]

For example,

```
CLASS LT 8
```

Up to eight logical expressions of this simple form can be combined to form more complex expressions by using logical connectors. For example,

```
CLASS GE 2 AND CLASS LT 8 OR SUIT = 5
```

There is no specific limit to the number of [operand-1] [logical-operator] [operand-2] combinations and logical connectors that can be used in a single expression. However, commands that have logical expressions as arguments are limited to 254 characters in length.

All logical operators and connectors have equal precedence. Operations are performed in sequence from left to right. However, parentheses can be used to request that logic within parentheses be performed first. Operations inside the innermost set of parentheses have the highest precedence.

Each element of a logical expression (i.e., operand, logical operator, logical connector, parenthesis) must be separated by blanks, except when using the IN operator.

Logical expressions can only be used on point, line, polygon and record themes. They are not supported on tic and annotation themes.

In order to perform selection on point, line or polygon themes, you must have run **build** or **clean** to create topology and feature attribute tables.

## Arithmetic expressions

Arithmetic expressions in ArcCAD have the following components:

### Numeric operands

- An item name
- A constant (e.g., `10`)
- An internal variable (e.g., `$RECNO`)

### Arithmetic operators

- **+** Addition
- **-** Subtraction
- **/** Division
- ***** Multiplication
- ****** Exponentiation
- **LN** Logarithm
  Calculates the natural logarithm of the operand it precedes. The operand must be a positive number.
- **WD** Width computation
  Calculates the width in characters of the operand it precedes excluding trailing blanks. The operand must be a character item or a literal string.

Arithmetic operators have the following precedence from highest to lowest:

1) LN, WD
2) **
3) *, /
4) +, -

Operands of equal precedence are performed as they are encountered, moving from left to right through the expression. Parentheses can be used to override inherent precedence. Operations within the innermost set of parentheses are performed first.

***Note:*** There is no unary minus operator for negating an operand in ArcCAD. For example, the expression -AGE evokes an error message (instead, specify -1 * AGE). Also, all arithmetic operations in ArcCAD are performed in double precision. As a result, an expression involving integer operands may be evaluated as having a fractional part.

Examples of arithmetic expressions:

```
SUIT = ( SOIL + 2 * TERRAIN ) / 12
LAB_WIDTH = ( WD ( LABEL ) + 4 ) * 0.22
```

## Internal variables usable in logical and arithmetic expressions

ArcCAD provides three internal variables that can be used in logical and arithmetic expressions.

**$RECNO**—the record number of a record in the selected data file.

**$PI**—the value for *pi* (3.14159…), which is the ratio of a circle's circumference to its diameter.

**$E**—the value for *e* (approximately 2.71828), which is the base of the number system for natural logarithms.

These internal variables can be used as operands anywhere within a logical or arithmetic expression. For example, with the **reselect** command you can specify

```
$RECNO GT 100
```

This logical expression will find all records from the currently selected set whose record number is greater than 100. Selection using the pseudo item $RECNO will be much faster using the 'IN {LIST}' syntax as follows:

```
$RECNO IN {100->99999}
```

This logical expression will also find all records from the currently selected set whose record number is greater than 100 (and less than 99999). This technique will select records much faster than using the 'gt' expression above.

## Using indexed items

Item indexes can be used to increase the efficiency by which features are selected when selecting features using attributes. An item index is created using the **indexitem** command. To use an item index when selecting features, you must specify a logical expression. The logical expression must contain the name of the item index and must use the 'IN {LIST}' syntax. For example, to create an item index called LEN_NDX on the LENGTH item in the AAT of a theme called ROADS use **indexitem** as follows:

Command: **indexitem**
Theme name: **roads**
Item to index: **length**
Index file name: **len_ndx**

Then to select all features in the ROADS theme using the index LEN_NDX, whose length is greater than 300 but less than 700, use the following command line:

Command: **reselect**
Theme name: **roads**
Reselect by? **expression**
Logical expression: **len_ndx in {300->700}**

In the logical expression, the index name (LEN_NDX) is used where an item would be used for nonindexed selections. Selections made using the index will operate much faster than selections made without an index.

## Using related items

Anywhere an item can be specified in a logical expression, ArcCAD will accept an item in a related file. This means that selections can be performed not only using attribute values in the theme's feature attribute table, but also using attributes stored in a related record theme. A relate is established using the **relate** command. To access an item in a related file, precede the related item name with a pound sign (#). For more information on using related items, refer to the **relate** command reference.

## Spatial selection of features

Features can be selected by defining a rectangular area from a theme. All the features of the specified theme that fall at least partly inside this area can be selected. Optionally, you could specify that only the features contained entirely within the area will be selected. Using **reselect** or **aselect** with the **window** option lets you define this area by giving the coordinates of the opposing corners of a rectangle. Other options, such as **circle** and **polygon**, allow the selection of features by specifying a circle or arbitrarily shaped polygon.

For example, suppose you want to make a map using two themes called STREAMS and PRCLS that cover the same area. You want your map to show all the features from the first line theme, and only those polygons in the center of the second polygon theme. You can't use the **setext** command to window in on the polygons in the center of the PRCLS theme because this zoom window would not cover all of the features from the STREAMS theme, which you also want to display. Instead, you can give these commands:

```
Command: setext
Theme name: streams

Command: arcs
Theme name: streams

Command: reselect
Theme name: prcls
Reselect by? window
First corner: 34,1002
Second corner: 189,1034

Command: polys
Theme name: prcls
Line symbol number: 9
```

in which **reselect** is used with the **window** option to select only those polygons that fall within an area in the center of the PRCLS theme. The coordinates 34, 1002 and 189, 1034 that specify the lower-left and upper-right corners of this area are given in the same units that the GIS data set is stored in.

## Display commands that operate on selected sets

When a set of arc features has been selected from a line theme, the following commands draw or label only those arc features in the selected set:

**arclines**
**arcmrk**
**arcs**
**arctxt**

When a set of polygons has been selected for a polygon theme, the following commands draw or label only those polygons in the selected set:

**labelerr**
**labelmrk**
**labeltxt**
**polyshd**
**polytxt**
**polys**
**polys3d**

When a set of point features has been selected for a point theme, the following commands draw or label only those points in the selected set:

**pointmrk**
**points**
**pointtxt**

The **annotxt** and **tics** commands do not operate on the selected set of features since feature selection cannot be performed against annotation and tic features. Also, the **nodes** and **noderror** commands do not operate on the selected set of arc or polygon features. For example, **nodes** will draw all the nodes in a specified theme regardless of the selected sets of arc or polygon features.

When you use the **arcs** command to draw arc features from a polygon theme, all the arc features will be drawn, regardless of the selected set of polygons. To draw just the outlines of polygons in the selected set, use the **polys** command.

## Entity-feature links and selection

If entity-feature links exist between entities in the current drawing and theme features, then ArcCAD can translate one type of selection set to the other. Entity-feature links are established when features are created from entities or when entities are created from features (refer to the sections ‘Entity-feature links and editing’ and ‘Using entity-feature links with display commands’ for more information on establishing links).

The ArcCAD feature selection commands (**reselect** and **aselect**) generate feature selection sets. If entities in the current drawing are linked to their corresponding features, then those entities can be used to create a feature selection set. The **entity** option on the **reselect** and **aselect** commands will prompt for an AutoCAD entity selection set. This entity selection set can be created using any AutoCAD method. ArcCAD reads the entity-feature links for each entity in the selected set and creates a feature selection set containing the features linked to those entities. This creates a feature selection set for the specified theme that can be used like any other ArcCAD feature selection set. Refer to the **reselect** and **aselect** command references for more information on the **entity** option of these commands.

The reverse of this operation can be performed using the ArcCAD **xselect** command. The **xselect** command (along with the AutoCAD standard selection commands) generate entity selection sets. If features in the current drawing are linked to their corresponding entities, then those features can be used to create an entity selection set. The **theme** option on the **xselect** command processes each feature in the feature selection set for a theme. The entity-feature links are read to create an entity selection set containing the entities linked to those features. This creates an entity selection set that can be used like any other AutoCAD entity selection set. Refer to the **xselect** command reference for more information on the **theme** option of the **xselect** command.

# Adding key legends

Key legends are displayed using a special set of commands to design and draw key legends. There are three types of key legends in ArcCAD. Key legends can be created for each of the following feature classes: point, line and polygon.

Key legends show samples of symbols drawn inside boxes along with associated descriptive text.

Key legends are defined by key legend files. Key legend files are ASCII text files created with any text editor. A key legend file contains the ArcCAD symbol numbers of the symbols used on the map and associated text describing what each symbol represents. Each symbol number must be on its own line in the key file and must be preceded with a period (.). The lines following each symbol number line contain the text that will be drawn next to that symbol. When the text is drawn, it appears exactly as it is given in the file so that indents, spaces and line returns are preserved.

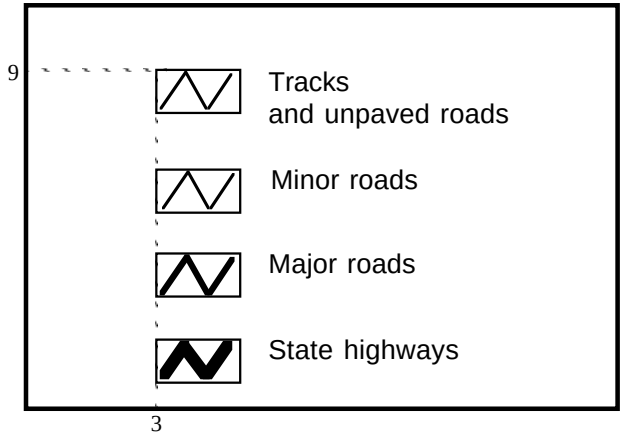
Here is a sample key legend file that defines a legend describing what the line symbols on a map represent:

```
.1
Tracks
and unpaved roads
.3
Minor roads
.5
Major roads
.15
State highways
```

When a legend is displayed with **key**, the type of legend (i.e., whether it contains marker, line or shade symbols) must also be specified. The text in the legend is drawn using the current AutoCAD text settings. You must also specify the size of the boxes around each symbol, the separation between the key boxes and the position where the legend will be drawn. The legend is drawn so that the top-left corner of its first key box is positioned at the point specified.

In this example, the key legend file listed above is used to define the legend:

Command: **key**
Symbol class?: **line**
Key file name: **roads.key**
Width of key box: **0.50**
Height of key box: **0.25**
Horizontal separation: **0.15**
Vertical separation: **0.25**
Draw key box outline?: **yes**
Key upper left corner (X,Y): **3.0,9.0**



When a legend is drawn, a solid line is drawn around each key box using the current AutoCAD line settings unless you answer **no** to the Draw key box outline?: prompt.

When a legend for line symbols is drawn, each key box has a line symbol drawn inside it as a zigzag. If you would prefer each line symbol sample to be a straight horizontal line instead of a zigzag, specify the size of the key box equal to the desired length of the straight line and a height of 0. Then do not draw the key box outline.

When a legend for marker symbols is drawn, each key box has a marker symbol drawn in its center.

When a legend for shade symbols is drawn, each key box is filled in with a shade symbol.

# Using the identify command

The **identify** command lets you use the cursor to point at a feature displayed on the screen and obtain a list of that feature's attributes. To use **identify**, name the theme that is to be queried and select an entity (for point or line themes) or place the cursor anywhere inside a polygon (for polygon themes) and press any key.

**Identify** lists the attributes stored in the theme's feature attribute table for the identified feature. For example,

```
Command: identify
Theme name: parcel
Select a polygon:

$RECNO     AREA    PERIMETER  PARCEL_  PARCEL_ID  NAME
  17    2432.473  901.877      23        23       SMITH
```

By default, **identify** lists all the item values for the record in the theme's feature attribute table for the feature you selected. If you don't want **identify** to list all of the item values, you can specify which items you want to list. When the 'Use ArcCAD dialog boxes' setting in **esri_prefs** is **on**, the attribute listing for the selected feature will be displayed in a dialog box.

**Identify** can also be used to list the values of items in a related file.

Chapter 5

# Spatial analysis

The major advantage of a geographic information system (GIS) is that it allows you to identify, maintain and manipulate the spatial relationships between map features representing geographic phenomena. Essentially, a GIS gives you the ability to create new relationships, associate new attributes to map features, and then store these attributes in the theme's feature attribute table.

For example, suppose you wanted to identify the soil type within a particular land parcel. One quick way to do this is to manually overlay a soils map and a parcel map. This is an easy operation if the soil type for only one parcel needs to be identified. But what if you need to find out the soil types for all of the parcels in a county? And what if some parcels contain more than one soil type? These questions are more difficult to answer, but such an operation is easily performed in a GIS using topological overlay. Topological overlay is one of many spatial operations that can be performed with a GIS to create new spatial relationships—relationships like the soil type(s) of each land parcel.

The results of these analytical operations add new descriptive data about features to the feature attribute table for a theme. Once these relationships are recorded, arithmetic and logical operations can be performed on the tabular attributes. With tabular analyses, you can determine the suitability of various sites for development, evaluate environmental impacts, calculate harvest volumes and develop plans for timber management, determine optimum routes for emergency response, identify the best location for a new facility, and so on.

The following example illustrates how you can analyze the descriptive attributes for map features.

**Objective:** Calculate expected dollar loss to residents from a major flood.

**Analysis criteria:** If zoned as RESIDENTIAL (R1 or R2) and within 30-year floodplain with unstable soils.

**Operation:** Value multiplied by loss factor.

| Lot_ID | Zoning | Value | Floodplain | Soils | Loss |
|---|---|---|---|---|---|
| 1 | R1 | 10,000 | No | A | 0 |
| 2 | R2 | 50,000 | Yes | C | 25,000 |
| 3 | C | 30,000 | Yes | B | 0 |
| 4 | C | 90,000 | No | A | 0 |
| 5 | R1 | 100,000 | No | C | 0 |
| 6 | R1 | 115,000 | Yes | A | 86,250 |
| 7 | R2 | 100,000 | Yes | C | 50,000 |

$161,250 = Total

| Soils | Compaction | Stability | Factor |
|---|---|---|---|
| A | 50 | Low | .75 |
| B | 95 | Good | .25 |
| C | 83 | Moderate | .50 |

ArcCAD® software's analysis commands provide you with the tools necessary to build additional feature attributes that can then be analyzed using tabular analysis commands (such as **class**, **frequency**, **statistics**, etc.) or your database management software. This chapter will introduce some of the most important spatial operations performed by ArcCAD that allow you to create new spatial relationships for your mapping applications.

# Spatial analysis concepts

Geographic information systems are used to perform a number of fundamental spatial analysis operations. For example, in a forestry application, a set of buffer zones can be generated around stream boundaries. These buffer zones can then be used to identify environmentally sensitive areas that are closed to timber harvesting. In a municipality, a similar operation can be used to generate a 300-foot buffer around a specified parcel whose owners are requesting a zoning change. All of the property owners living within the 300-foot zone can then be notified about the zoning change request.

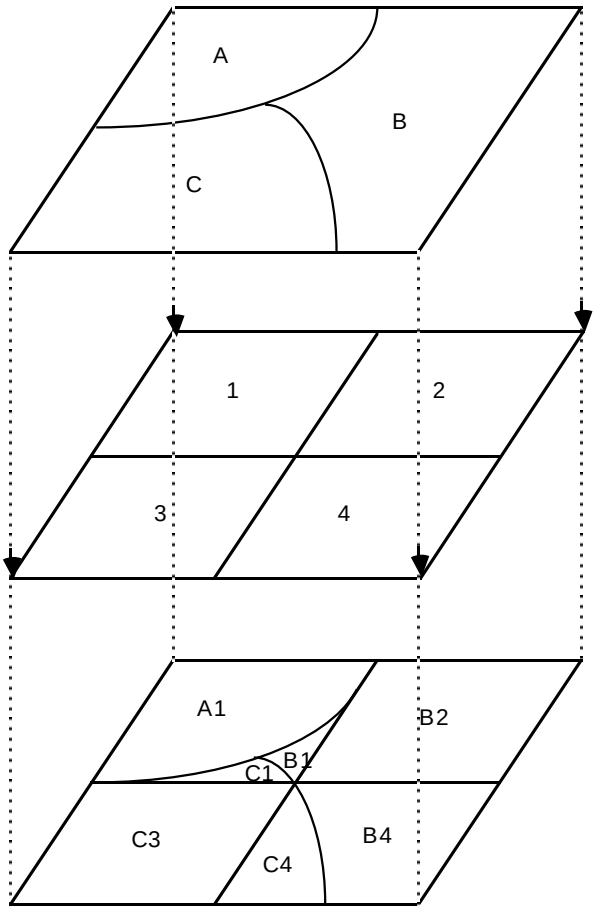
## Fundamental GIS operations

Such operations can use any number of analytical commands; however, these and almost all other spatial analyses utilize six fundamental operations: topological map overlay, buffer generation, feature extraction, feature merging and two relational database operations—relate and join.

## Topological overlay

New map features can be created by overlaying features from two themes referencing different data sets. Features from each data set are intersected to create new output theme features. Attributes of each input theme feature are combined from the two data sets to describe each new output theme feature.

**Conceptual View of Map Overlay**




*Theme features are merged spatially and their attributes are joined together. The resulting theme contains the features and attributes of both inputs.*

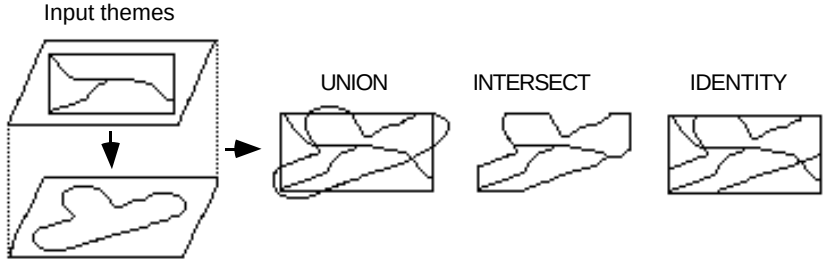
Topological overlay produces a spatial join of the input theme feature attributes. Notice how the output polygons in the preceding figure have attributes from both input theme features. Topological overlay involves not only the creation of new spatial features, but also the creation of new attribute relationships. The overlay operation merges attributes from the input theme features and creates a new theme.

Topological overlay can be used for a number of different objectives including theme updating, feature extraction, merging adjacent themes and merging feature attributes. To meet these various objectives, ArcCAD contains six commands: **esri_union**, **esri_intersect**, **identity**, **clip**, **erasecov** and **update**. Each is described in more detail throughout this chapter.

In topological overlay, polygon features of one theme's data set can be overlaid on polygon, point, or line features of a different data set. Depending on the objectives of the overlay, different output theme features can result.

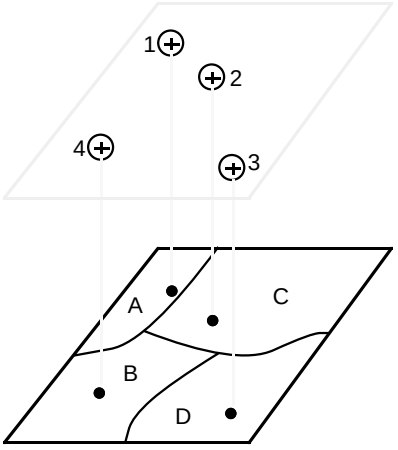
## Polygon overlay

This process merges overlapping polygons from two themes referencing different data sets to create new polygons in an output theme. Polygon attributes are also merged. This makes polygon overlay useful for modeling purposes. For example, a polygon overlay of zoning boundaries on soil boundaries might be used to show all areas within a city's commercial zones that have unstable soils. Commands available for performing a polygon-on-polygon overlay include **esri_union**, **esri_intersect** and **identity**. Each differs in the subset of features saved in the output theme. **Esri_union** keeps all polygons from both input themes; **esri_intersect** keeps only those areas covered by both input themes; and **identity** uses one theme as the template into which the other theme's polygons are merged and clipped along the edges.



## Point-in-polygon

Point theme features can be overlaid on polygon theme features to identify the polygon within which each point falls.

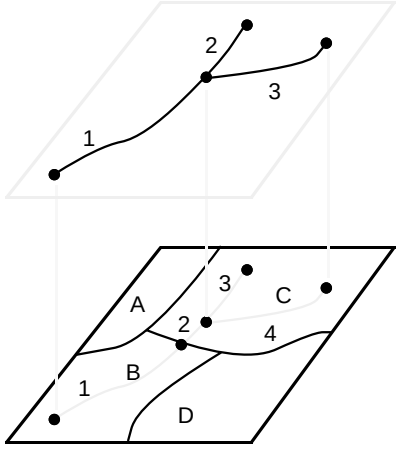



| Point ID | V1 | V2 | V3 | Poly | V6 |
|---|---|---|---|---|---|
| 1 | | | | A | |
| 2 | | | | C | |
| 3 | | | | D | |
| 4 | | | | B | |

The result of a point-in-polygon overlay is a set of points with additional attributes. The new attributes are those of the polygon within which each point is located. For example, if well sites are represented as points in a point theme, and leaseholdings are represented as polygons in a polygon theme, a point-in-polygon overlay could be used to determine the well sites contained within each leasehold.

## Line-in-polygon

Polygon theme features can be overlaid on arc theme features to identify which polygon, if any, contains each arc or part of an arc. The resulting output of a line-in-polygon overlay has arc features.




| Line ID | Old Line | Poly |
|---|---|---|
| 1 | 1 | B |
| 2 | 1 | C |
| 3 | 2 | C |
| 4 | 3 | C |

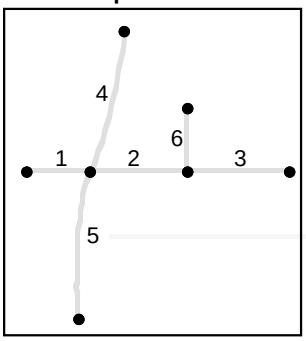
Part of an arc feature may also fall directly on a polygon boundary instead of within a polygon. When this occurs, the attributes of one of the polygons are assigned to the arc. In all cases, polygons that are to the left and right sides of each output arc feature are identified as part of the arc's topology.

The resulting arc features of a theme also contain the attributes for polygons within which each arc falls. For example, roads stored as arcs in one theme can be merged with county boundary polygons from another theme using a line-in-polygon overlay. The results can be used to determine the number of miles of various road types that fall within each county.

## The relate and relational join operators

Geographic information contains two data types, spatial and descriptive, which are used to define each feature. To connect or 'relate' the spatial data to the descriptive data for each feature, a common feature number is stored with both record types. For example, the coordinates and attributes for a set of features could be stored as follows:

**Roads map**




**Coordinates**

| FEATURE_NO | X,Y pairs | | | | |
|---|---|---|---|---|---|
| 1 | 1,5 | 5,5 | | | |
| 2 | 5,5 | 8,5 | | | |
| 3 | 8,5 | 10,5 | | | |
| 4 | 6,9 | 5,8 | 5,7 | 5,6 | 5,5 |
| 5 | 5,5 | 4,4 | 4,3 | 4,2 | 4,1 |
| 6 | 8,5 | 8,7 | | | |

**Attributes**

| FEATURE_NO | ROAD_TYPE | PAVEMT_TYP | WIDTH | LANES | NAME |
|---|---|---|---|---|---|
| 1 | 2 | Asphalt | 48 | 4 | N Main ST |
| 2 | 2 | Asphalt | 48 | 4 | N Main ST |
| 3 | 2 | Asphalt | 48 | 4 | N Main ST |
| 4 | 1 | Concrete | 60 | 4 | Hiway 42 |
| 5 | 1 | Concrete | 60 | 4 | Hiway 42 |
| 6 | 4 | Asphalt | 32 | 2 | Elm ST |

Notice that the unique feature ID number can be used to relate the coordinates and attributes for each feature.

This relational concept can be applied to more than just keeping track of feature coordinates and their attributes. Any two feature attribute tables, database files or record themes can be 'connected' if they share a common attribute; for example,

| PARCEL_NO | OWNER |
|---|---|
| 11-115-001 | BROWN, D. |
| 11-115-002 | GREENE, J. |
| 11-115-003 | SMITH, L. |

| PARCEL_NO | ZONING | LEGAL_AREA |
|---|---|---|
| 11-115-001 | R1 | 12,001 |
| 11-115-002 | R2 | 15,775 |
| 11-115-003 | COMM | 19,136 |

A *relate* uses a common item type to establish connections between corresponding records in two database files. Each record in one file is connected to a record in the other file that shares the same value for the common item. A relate has the effect of making a feature attribute table 'wider' by giving you access to additional feature attributes that aren't actually stored in the feature attribute table. The **relate** command is used to relate feature attribute tables, database files or record themes to each other.

A *relational join* is the operation of relating and physically merging two database files using their common item:

| PARCEL_NO | OWNER | ZONING | LEGAL_AREA |
|---|---|---|---|
| 11-115-001 | BROWN, D. | R1 | 12,001 |
| 11-115-002 | GREENE, J. | R2 | 15,775 |
| 11-115-003 | SMITH, L. | COMM | 19,136 |

The **joinitem** command is used to perform a relational join of two database files.

As seen above, relates and relational joins are conceptually simple operations; yet they are fundamental GIS operations that are used frequently. For example, when a theme overlay is performed, each new output feature has attributes from both sets of input features used to create it (such as soils and land parcels). In essence, topological overlay is a 'spatial join'.

Here are some other situations where the use of relates and relational joins are important:

- Existing attribute databases can be related to theme features to further describe those features. For example, records in tax assessor files can be related to a parcels theme that contains unique numbers for each parcel. For themes that contain information extracted from Census Bureau TIGER and DIME files, census data about blocks can be related to polygons using the block and tract numbers contained in both.

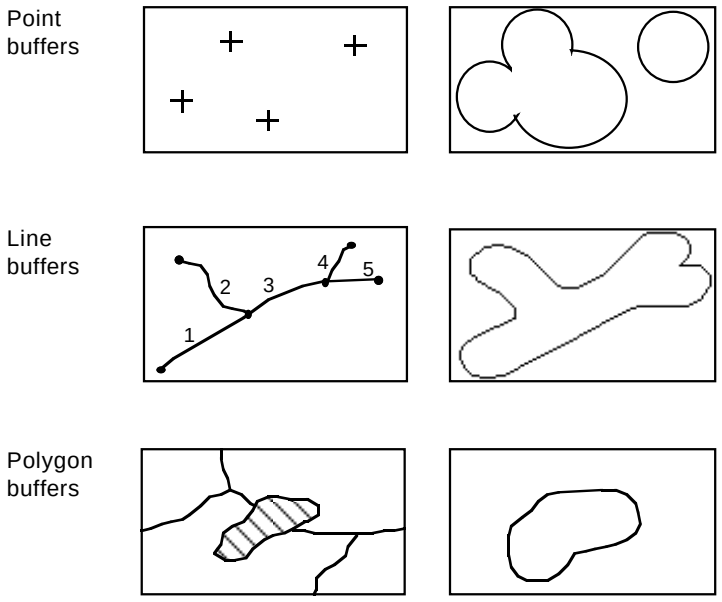
- Various groups of users may be interested in the same theme yet need to use different attributes about that theme's features. For example, city engineers may be interested in the maintenance condition of city streets, while the fire department may be interested in emergency access routes and street attributes that can help evaluate proposed routes. A relate can be used to access only those theme feature attributes that are relevant to a specific application.

- The same descriptive data might apply to a number of theme features. For example, characteristics about soil types such as compaction, moisture content and erosion hazard might be recorded for soil types. Instead of repeating these attributes for every soil polygon, you can use the soil type attribute of each polygon to relate to another table that contains the attributes about each soil type. This type of relate reduces the amount of redundancy in your database.

## Buffer generation

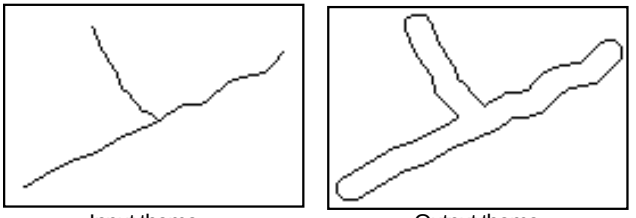
One very important class of spatial operations concerns the determination of spatial proximity or nearness of various geographic features. For example, a notice about a request for a zoning change within a city must be sent to the owners of all surrounding parcels that are within a particular distance (say 500 feet) of the subject property. In forestry, timber harvesting is restricted within a certain distance of streams. In wildlife habitat assessment, many animals only use areas that are within a home range distance of special critical habitats such as water or nesting sites.

In each of these cases, a spatial buffer can be generated to identify the areas and features that fall within the buffer zone. As shown in the figure below, buffer zones can be generated around point, line and polygon theme feature classes:



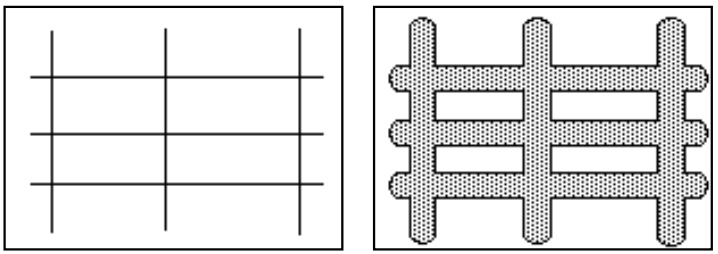

The ArcCAD **buffer** command is used to generate buffer polygons.

You need to be aware of some important considerations when generating buffers. First, the generated buffer zones are always polygons that are created as a separate polygon theme. The new theme never includes any of the input features:



You can use the appropriate ArcCAD display commands if you want to see the original theme features displayed within the buffer polygon theme by simply drawing the themes on top of each other in the same drawing.

All of the output polygons from buffer generation may not necessarily be included within the buffer zone. Below is a buffer map showing areas that are inside and outside the buffer zones. In this case, each polygon output from **buffer** can be identified as being inside or outside of the buffer polygon with a special feature attribute generated by the **buffer** command.



Input theme Output theme

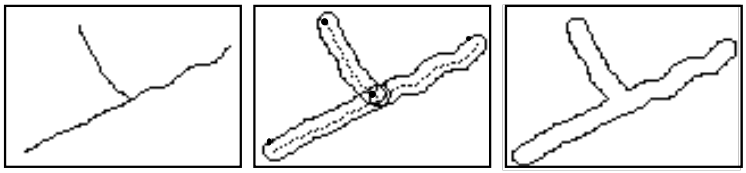
*In this example, the shaded areas in the output theme represent areas within the buffer zone. Unshaded polygons are not within the buffer distance of lines in the input theme.*

An item named INSIDE is included in the output theme's PAT created by **buffer**. INSIDE has two possible values:

- 1 - the polygon is outside the buffer zone.
- 100 - the polygon is within the buffer zone.

### Resolving overlaps between buffer zones

The **buffer** command has the unique capability of performing a topological analysis to automatically resolve overlapping buffers between features. When multiple features are buffered, each of their buffer zones may intersect with each other. **Buffer** automatically identifies the arc features that fall inside a buffer zone and dissolves them.

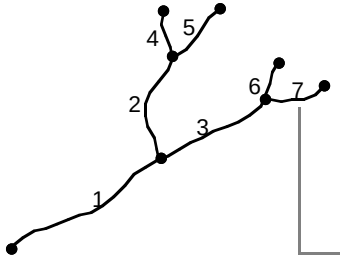


Input theme During **buffer** **Buffer** results

*Buffer line segments interior to buffer zones are automatically removed by **buffer**.*

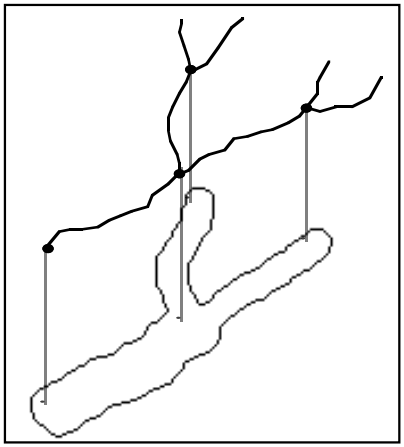
### Variable distance buffering

It is often important to specify different buffer distances for features based upon feature attributes. For example, the width of environmentally sensitive areas to be depicted around streams may be dependent upon the stream type; the area of impact caused by noise pollution from various factories, airports and other facilities will differ with the noise level at each source. Or perhaps you wish to buffer only selected features as opposed to all features. **Buffer** handles these situations by allowing you to specify an item in the feature attribute table from which a buffer distance is determined for each input theme feature.




| STREAMS_ID | ORDER | DIST |
|---|---|---|
| 1 | 3 | 1200 |
| 2 | 2 | 800 |
| 3 | 2 | 800 |
| 4 | 1 | 0 |
| 5 | 1 | 0 |
| 6 | 1 | 0 |
| 7 | 1 | 0 |

Resulting buffer

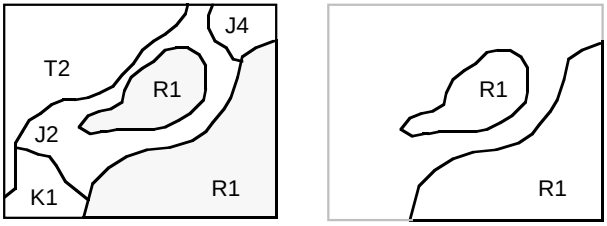


When the distance item value for a feature is zero, it is not buffered and no feature is saved in the output theme. In addition to using an item to specify the distance, you can also use a lookup table with **buffer**. Refer to the **buffer** command in the command reference guide for a description of how to use lookup tables.

## Feature extraction and merging

Another important spatial operation performed with the ArcCAD analysis commands is feature extraction or feature subsetting. Feature extraction is the process of identifying a subset of theme features to be saved and then eliminating the unselected features to create a new output theme. Two types of feature extraction can be performed. One approach is to use a spatial area as a clipping region to clip out a portion of a theme.


A second approach is to identify a subset of theme features to be saved or eliminated if their attributes match specified logical criteria. For example, select and save all polygons whose zoning class is R1.

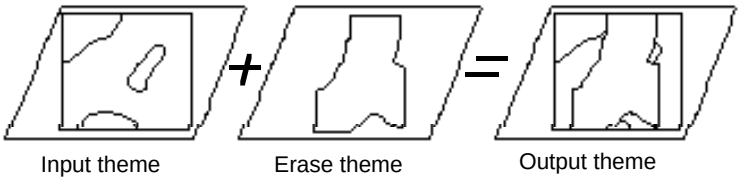



ArcCAD includes a number of commands that can be used to perform both types of theme feature extraction. These include **clip**, **erasecov** and **split** for performing spatial clipping, and **extract** and **eliminate** for performing logical selection of features. The situations in which each command is used will vary with the type of analysis being performed.

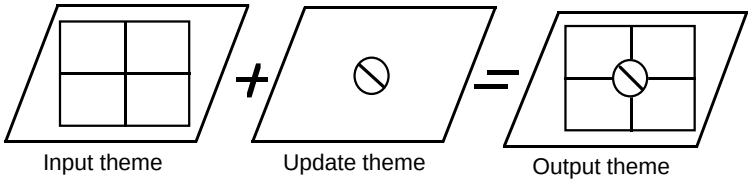
There are four main processes in which theme feature extraction commands are used. These include theme update, study area clipping, sliver removal and feature reduction for further analysis.

### Theme update

Before adding new line or point features to update a theme, you can use the **erasecov** or **extract** command to remove the portion of your theme in which features will be added. Then, you can digitize or append new features into the theme.



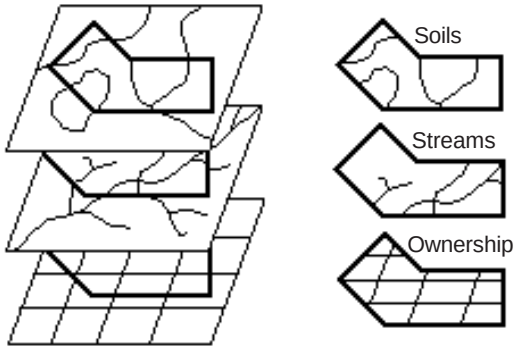

Of special note is the **update** command which can be used to 'cut and paste' a new update into an existing polygon theme.



### Study area clipping

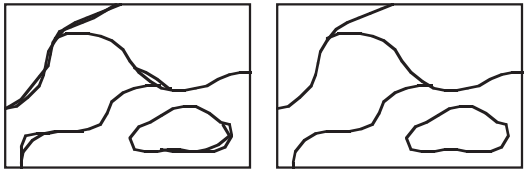
Some studies will not cover the entire area contained in your geographic database. You may wish to extract a portion of each theme from your database using the **clip** command to create themes that are just large enough to cover the study area and its area of influence.

**CLIP of themes in a database**




### Sliver removal

After performing a polygon overlay to create new theme features with attributes from a series of themes, you may find that the output theme contains many small sliver polygons. Sliver polygons often result from the overlay of inaccurate data sets and can have inconsistent attribute combinations. For example, an overlay of land use and vegetation themes might classify a polygon as both 'RESIDENTIAL' for land use and 'WATER' for vegetation.



Before **eliminate** After **eliminate**

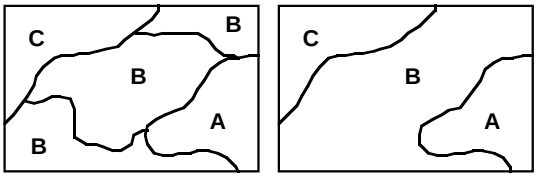
The **eliminate** command is used to remove sliver polygons by merging selected slivers into adjacent polygons. Sliver polygons can be selected for removal using any set of logical criteria. For example, you may wish to identify polygons whose area is less than a specified size. **Eliminate** can be very effective for removing polygons to resolve such inconsistencies.

## Feature reduction before analysis

By defining a set of logical criteria with **extract**, you can identify a subset of features to be placed in another theme for further analysis. For example, for a park siting, you may wish to select only those areas that are within one mile of roads, zoned agricultural, and contain streams of a particular class. Then you can use the subset of areas that match these criteria for additional study.

### Dissolve operation

A special type of feature reduction can be performed using the **dissolve** command, which merges adjacent polygons into a single polygon if they share the same value for a specified item. **Dissolve** is useful for simplifying a theme after performing an analysis.

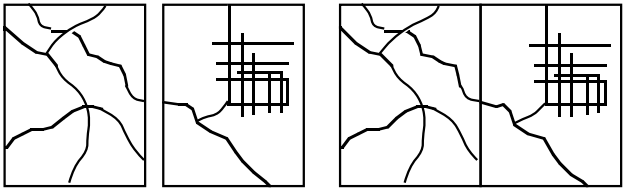



Before **dissolve** After **dissolve**

## Database creation and management

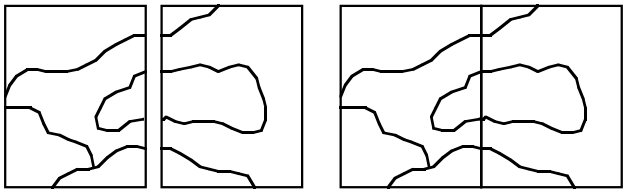
In addition to the spatial operations described previously, ArcCAD contains three commands that aid in the creation and maintenance of geographic databases: **append**, **mapjoin** and **split**.

**Append** combines themes or coverages with the same or different feature classes into a single theme or coverage. **Append** gives you flexibility in managing your database, allowing you to combine either themes or coverages into an output theme or coverage. **Append** is most often used to merge different feature classes into the same theme or coverage.



Before **append** After **append**

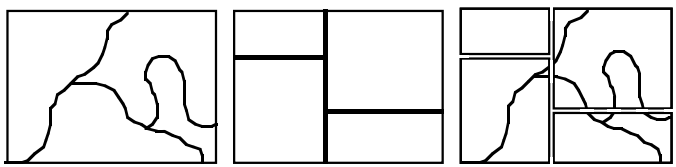
**Mapjoin** merges adjacent polygon themes into a single theme or merges adjacent polygon coverages into a single coverage. **Mapjoin** gives you flexibility in managing your database, allowing you to merge either themes or coverages into an output theme or coverage.



Before **mapjoin** After **mapjoin**

**Mapjoin** allows you to automate themes or coverages as individual map sheets, yet store and use them as joined themes or coverages. After **mapjoin**, the **dissolve** command can be used to remove the original border arcs.

The **split** command partitions one large theme or coverage into many smaller themes or coverages. It uses the polygons of a split coverage to create each individual output coverage. In order to provide flexibility in managing your database, **split** allows you to partition either themes or coverages into many output themes or coverages.



Input theme or coverage | Split theme or coverage | Output themes or coverages

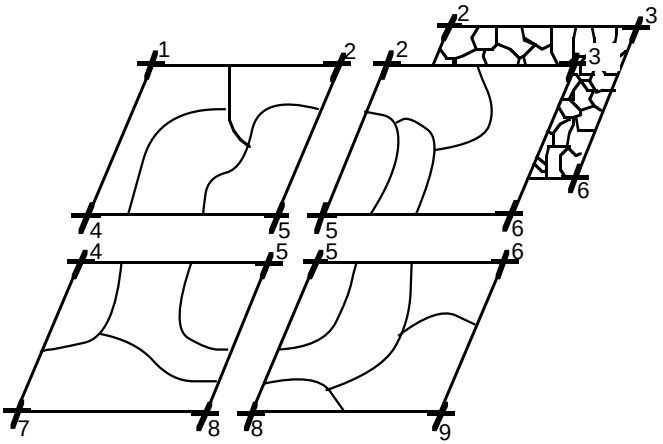
The **mapjoin** and **split** commands are useful for managing large geographic databases. During automation, each layer can be digitized using individual map sheets; then, **mapjoin** can be used to merge adjacent maps when necessary. **Split** can partition each layer into a set of subregions that can be used to store and maintain your database over time.

## Common coordinate systems

A common coordinate system is a prerequisite to performing analysis using topological overlay, buffer and other spatial operations. Themes that represent

separate layers must accurately overlay on top of each other, and adjacent themes must connect along their shared borders.

The coordinates in which a theme is stored are controlled by the coordinates of that theme's coverage tics. Every theme with a feature class of point, line, polygon or annotation has tics even if a tic theme has not been defined for those tics. By default, ArcCAD will create tics at the corner points of the extent of the coverage features. To access coverage tics, define a tic theme for the theme's GIS data set. Tic features serve as the common basis for registering a theme to ground coordinates, as well as for registering themes to each other. Therefore, it is important that the x,y locations of theme tics be accurately recorded and all related themes registered using a common set of tic locations (e.g., UTM meters, State Plane feet, etc.). Each theme within a study area can be spatially related using the same set of tics as geographic control when creating each theme.



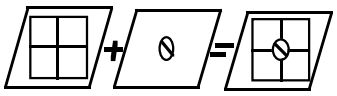
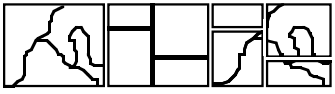
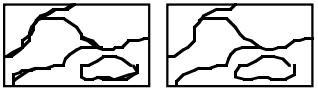
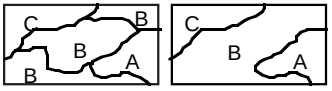

*Common tics can be used to register adjacent themes as well as 'layers' of themes for the same area.*

Once your database is in a common coordinate system, ArcCAD will automatically maintain correct coordinates for you during subsequent spatial operations.

One method that can be used to determine if all of your themes are in a common coordinate system is to draw the various themes of your database into one drawing. If themes do not correctly overlay or join each other, you can use the **transform** command supplied with ArcCAD to transform coverage coordinates into the same units. **Transform** is used when coverages are digitized from the same base map (i.e., the same projection), but their units are not equivalent, for example, when one coverage is in meters but the other is still in digitizer inches.

# Summary of spatial operations

As described above, ArcCAD commands are used to perform a number of spatial operations. This table is provided as a guide for selecting the appropriate command to perform a desired spatial operation and summarize the analytical capabilities of ArcCAD.

| Type of operation | Description | Commands Used | Example |
|---|---|---|---|
| Theme updating | – merge new features via a cut and paste | UPDATE |  |
| | – erase part of a theme before adding update features | ERASECOV |  |
| Feature extraction, subset or reduce a theme | – cut out a piece of a theme using a template | CLIP |  |
| | – split a coverage or theme into number of smaller coverages or themes | SPLIT |  |
| | – remove part of the inside of a theme | ERASECOV |  |
| | – logically select features to be kept | EXTRACT |   |
| Feature merging | – logically select polygons to be merged into a neighboring polygon | ELIMINATE |  |
| | – drop borders between neighboring polygons with equal values for a selected item | DISSOLVE |   |

# Performing spatial analysis

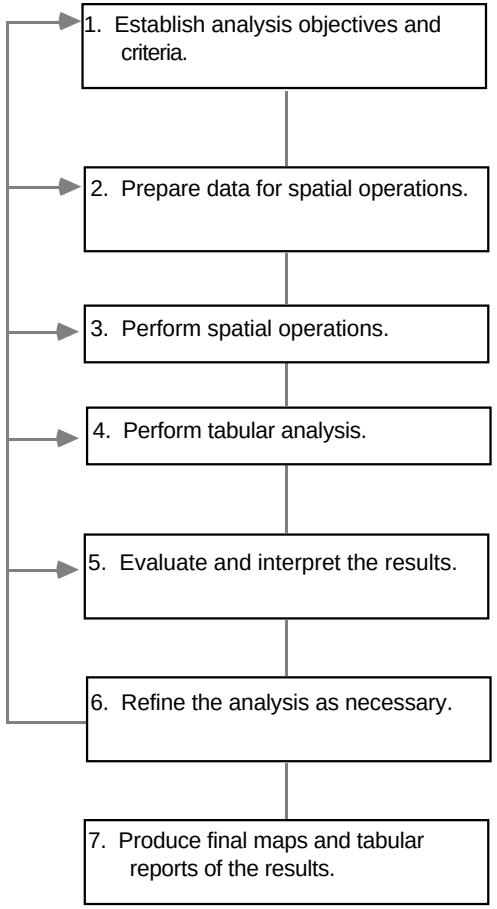
As described previously, ArcCAD provides commands that perform a number of fundamental spatial operations, such as polygon overlay, buffer generation and feature extraction, which is required for performing geographic analyses.

The flexibility of ArcCAD allows you to propose new alternatives and quickly evaluate the results. These capabilities provide you with the tools necessary to address important questions regarding the resources managed by your organization.

For example, by overlaying soils on top of a parcel theme, you can identify the soil type(s) of each parcel. The new polygons created by the overlay contain attributes from both of the original themes. Then you can use ArcCAD software's tabular analysis commands or your database management system to analyze the new attribute combinations using arithmetic and logical expressions.

This section will introduce you to procedures you can use to perform spatial analysis and modeling with ArcCAD. The steps used to perform analysis are presented, along with example applications that demonstrate how to use ArcCAD for spatial analysis.

**Steps for performing geographic analysis**



## Establish analysis objectives and criteria

Analysis objectives are used to define the questions you want to answer with your geographic database. The criteria specify how you can use your GIS to answer the questions you pose. For example, the objectives of a study may be to identify sites that are suitable for locating a new park or to calculate the expected loss due to floods. The criteria used to meet the objectives should be stated such that they can be analyzed using a series of spatial queries. For example, here are some criteria that might be used for park siting:

- The park site must be easily accessible from major highways, yet it must not be located too close to highways in order to minimize noise levels and other disturbances.
- The park should be designed around small, natural streams.
- To maximize the usable area within the park, it should contain little or no marshlands along the streams.

Each of these criteria can be analyzed using spatial operations such as buffering, line-in-polygon overlay and polygon overlay. After these spatial operations are performed, you can evaluate various land areas for their suitability to support a new park.

## Prepare data for spatial operations

This step is used to identify and prepare the data to be used in the analysis. Usually, your geographic database will already exist, but some modifications might be necessary before you can conduct the analysis. Since geographic data have both spatial and descriptive components, each of these must be prepared before continuing with an analysis. For example, you may need to convert the units in which particular attributes are stored (such as converting area measurements stored as square feet to acres). A typical approach is to add a new item to a theme's feature attribute table that will hold the new data. In another example, you may need to clip out a portion of your geographic database to encompass only the area under study.

The requirements for data preparation differ from study to study. Thoughtful consideration for data preparation before you begin an analysis can help you complete the analysis more efficiently.

# Perform spatial operations

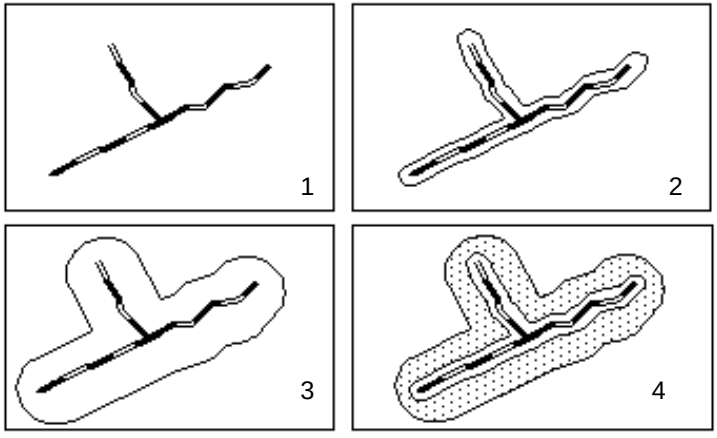
This step is unique to a geographic information system. It is this step that is used to derive the spatial relationships to be analyzed. Spatial operations include feature buffering, topological overlay, feature extraction, feature merging, and so on. Each spatial operation results in new information required for performing the analysis. Many operations may be required to derive the necessary data. For example, first you might create buffer zones around roads, then overlay these buffer zones with other layers such as soils or land use. Then you can analyze the results of this sequence of spatial operations.

The set of spatial operations to be performed is defined by interpreting the analysis criteria outlined in the first step. For the park siting example, each statement can be used to identify a series of spatial operations to be performed as follows.

'The park site must be easily accessible from major highways, yet it must not be located too close to highways in order to minimize noise levels and other disturbances'.

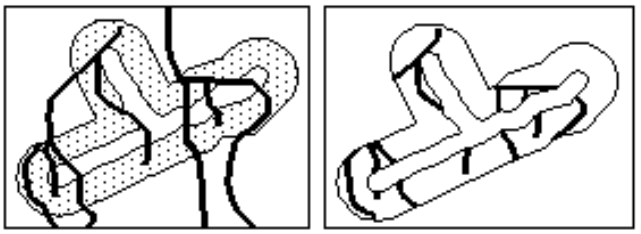
This statement can be translated into the following series of spatial operations:

1. Define major highways in study area.
2. Generate a half-mile buffer (2,640 feet) around major highways.
3. Generate a two-mile buffer (10,560 feet) around major highways.
4. Erase the inside of the wide buffer using the narrow buffer.



'The park should be designed around small, natural streams' can be translated as:

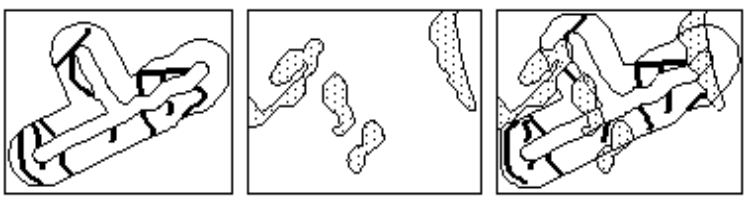
- Intersect a line theme of streams onto the results of the buffer theme generated above to identify only those stream segments that are within the desired distance of the highways.



'To maximize the usable area within the park, it should contain little or no marshlands along the streams'.

This statement can be translated as:

- Perform a line-in-polygon overlay of the remaining streams on a polygon theme that identifies marshes. This will identify stream segments whose entire shore length would be usable for picnicing, hiking, and so on.



The output theme resulting from the spatial operations may also require further spatial preparation. For example, you may wish to extract a subset of features to be analyzed further, or you may wish to eliminate sliver polygons before analysis.

# Perform tabular analysis

Most analyses require the generation of a final theme or set of themes resulting from the spatial operations. Once the final data sets are produced, you must prepare the data for analysis. Both spatial and descriptive data must be prepared.

The feature attribute tables of the derived theme hold information that is used for tabular analysis using logical and arithmetic expressions. It is often necessary to add additional items that are required to perform the analysis to the feature attribute table. For example, you might add an item named VALUE to hold a property value that is calculated as a function of the area, existing structures, and soil types of each lot in a theme of parcels. For park siting, you might add an item named SUITABILITY, which will hold a suitability ranking for siting a park at each output feature. ArcCAD commands such as **additem** and **dropitem** can be used to prepare a theme's feature attribute table for further analysis.

The analysis is performed using logical and arithmetic expressions on the new attribute relationships resulting from the spatial operations performed in Step 3. Use the criteria specified in Step 1 to define a series of logical and arithmetic operations that operate on the derived geographic data. For example, the criteria for park siting from Step 1 can be translated into the following expressions:

> 'The park site must be easily accessible from major highways, yet it must not be located too close to highways in order to minimize noise levels and other disturbances'.

This statement can be translated from the results of the two buffer zones around major highways.

- Only sites within the areas defined by the buffers will be considered for further analysis.

'The park should be designed around small, natural streams' can be translated as follows:

- Select all streams that fall within the buffer zones whose CLASS value is 2. The value 2 for CLASS might represent small, first- and second-order streams that have characteristics desirable for a new park site.

'In order to maximize the usable area within the park, it should contain no marshlands along the streams' can be translated as follows:

- Select all stream segments that do not fall within marsh polygons.

These logical operations result in an identification of stream segments that meets the original criteria for park siting.

## Evaluate and interpret the results

Once you have derived an answer from tabular analysis, you must evaluate your results to determine their validity. Do the results provide an answer that is both reliable as well as meaningful? This is an important verification step in which you may want to involve resource specialists to help you interpret and validate the results. It will be helpful to create a series of verification plots and tabular reports for interpretation.

One goal of this step is to accept the model results. First prepare a statement of the acceptance level that you think the analysis must meet. The statement defines how reliable the analysis results must be for you to accept them. For example,

> 'The results of the analysis match the expectations of resource specialists. The selected sites contain areas that they think are suitable for new park sites'.

Next, evaluate the results. If you cannot accept the results, use this step to identify modifications and enhancements to the analysis that must be performed to make the results acceptable. For the park siting example, you may want to further exclude potential sites that are adjacent to industrial developments, or you may want to add additional sites because you feel that the exclusion of marshland reduces the ecological value of the potential sites.

## Refine the analysis as necessary

After recording the limitations and shortcomings of your analysis, you can make a decision to further refine your analysis. If you choose to do so, then you must return to the appropriate step and perform the analysis again using the modified procedures. Here, a GIS is extremely useful because it will allow you to readily modify and repeat your analysis or to test a number of alternative approaches. Formal documentation of the procedures you follow will facilitate the repeatability and reliability of your studies.

# Produce final maps and tabular reports of the results

It will be important to communicate the results of your analysis in the most effective manner possible. Use the ArcCAD display functions along with AutoCAD software's graphic editing tools to create maps, and ArcCAD database management functions for tabular reports to display your results.

The remainder of this section demonstrates the use of ArcCAD software's analytical capabilities to address two problems. The first example demonstrates the evaluation and selection of potential sites for new parks. In the second example, the expected loss due to major flooding is computed for residential properties that fall within the floodplain. Each example applies ArcCAD tools following the analysis steps outlined previously.

# Example: Siting a picnic park

The following example will demonstrate the use of buffer zone generation, topological overlay and feature extraction for selecting possible sites for a public picnic park.

### Step 1 Establish the objectives and criteria for analysis

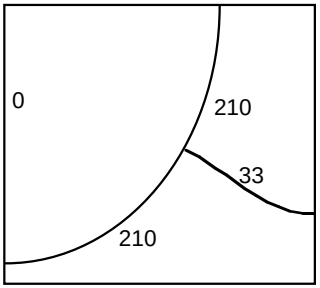
The objective of this example is to identify specific stream segments as possible sites for a public picnic park. The criteria for the park site locations include the following:

- The park site must be at least 0.25 but no farther than 1.25 kilometers from a highway.
- It must be situated next to small, natural streams.
- It must not be located in a marshy area.

### Step 2 Prepare the data for spatial operations

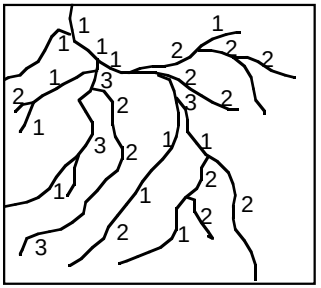
The following themes have been prepared. ROADS represents the highways that cross the study area; STREAMS represents the stream network of the study area; and MARSHES represents marshland areas.

**ROADS**



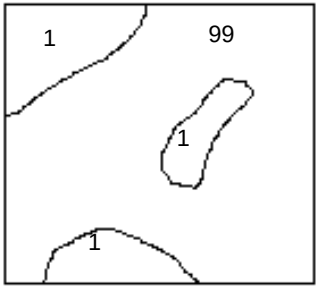

*The highways are identified by the item HWY_NUM. A HWY_NUM of 0 represents a border arc feature.*

**STREAMS**



*STREAMS are annotated with stream segment classification from item CLASS. A CLASS value of 2 indicates streams with characteristics desirable for a park site.*

**MARSH**




*The item called TYPE classifies the land. TYPE 1 represents land that is marshy, and TYPE 99 represents land that is not.*

In order to delimit the portion of the study area that is between 0.25 and 1.25 kilometers from the highways, two buffer polygons will be generated around the highways. One will represent the area that is within 0.25 kilometers of a highway and the other the area within 1.25 kilometers of a highway. Generating these buffer zones will require that **buffer** be executed twice (once for each buffer zone). Each execution of **buffer** will use a separate lookup table to assign buffer distances.

The lookup tables prepared for the **buffer** operations contain the following information. The item HWY_NUM is the same item contained in the ROADS theme feature attribute table and serves as the buffer item. The item DIST contains the buffer distance for each value of the buffer item in meters.

**Near lut**

| HWY_NUM | DIST |
|---|---|
| 0 | 0 |
| 33 | 250 |
| 210 | 250 |

**Far lut**

| HWY_NUM | DIST |
|---|---|
| 0 | 0 |
| 33 | 1250 |
| 210 | 1250 |

*The arc features having a HWY_NUM of 0 (those forming the theme boundary) are not assigned a buffer distance. This prevents them from being buffered.*

Refer to the **buffer** command in the command reference guide for a description of how to use lookup tables.

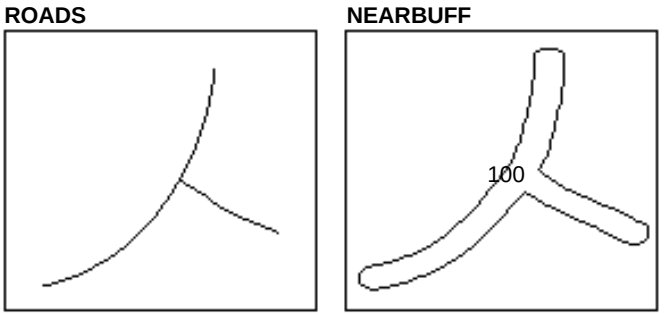
#### Step 3 Perform spatial operations

**Buffer** is a command that will generate polygons around features. In the case of this example, one polygon can be created around the highway features of the ROADS theme to represent the area surrounding the highways that is within the 0.25 kilometers distance from a highway, and another can be created to represent the area that is within the 1.25 kilometer distance from a highway.

The following is the command line used to generate the first buffer zone coverage enclosing the area within 0.25 kilometers of a highway.

Command: **buffer**
Theme name: **roads**
Output theme name: **nearbuff**
Buffer using?: **item**
Buffer item: **hwy_num**
Optional lookup table name: **near.lut**

The input and resulting themes look like this:




Using the **listdb** command to display the attributes of the output polygon theme would list the following:

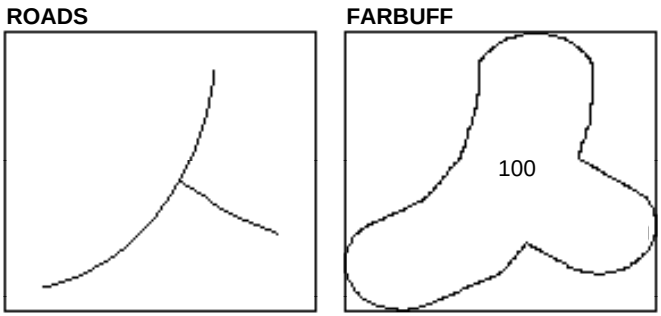
| AREA | PERIMETER | NBUFCOV_ | NBUFCOV_ID | INSIDE |
|---|---|---|---|---|
| -4515036.000 | 18778.590 | 1 | 0 | 1 |
| 4515035.000 | 18778.590 | 2 | 1 | 100 |

The PAT item INSIDE is automatically added by **buffer**. It is used to flag those polygons that represent areas outside a buffer zone (1) and those that are within a buffer zone (100).

The second buffer theme, representing the farthest extent the park site can be from a highway, is created as follows:

Command: **buffer**
Theme name: **roads**
Output theme name: **farbuff**
Buffer using?: **item**
Buffer item: **hwy_num**
Optional lookup table name: **far.lut**

The input and resulting themes appear below:




Input theme Output theme

Using **listdb** on the output theme would list the following:

| AREA | PERIMETER | FBUFCOV_ | FBUFCOV_ID | INSIDE |
|---|---|---|---|---|
| -2.60010E+07 | 24409.500 | 1 | 0 | 1 |
| 26000968.000 | 24409.500 | 2 | 1 | 100 |

The next operation combines these two buffer zone themes into one polygon theme in preparation for overlaying the MARSH theme with the buffer zones. For this analysis, we need only cut and paste NEARBUFF (the polygon theme representing the buffer zone 0.25 of a kilometer from a highway) into the polygon theme representing the 1.25 kilometer buffer zone (FARBUFF). The ArcCAD command **update** performs this operation. It creates a new theme by overlaying the two sets of input features and simply replacing the features in one theme with the features of the second theme that overlap the first one. It is important for this command that both themes contain the same set of items in their feature attribute tables. NEARBUFF and FARBUFF meet this requirement. Before merging their feature attribute tables, the value of INSIDE for NEARBUFF must be recalculated to 1 since, in the output theme, it will represent an area excluded as a possible park location (i.e., the area that is within 0.25 kilometers of a highway).

For example, the following ArcCAD commands will accomplish this:

Command: **reselect**
Theme name: **nearbuff**
Reselect by?: **expression**
Enter logical expression: **inside = 100**

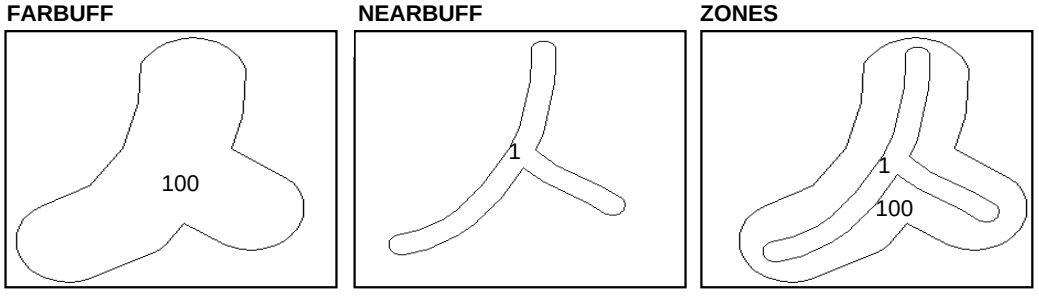
All records in NEARBUFF where INSIDE = 100 are now selected. The **calculate** command can then be used to change the value of INSIDE for these records to 1:

Command: **calculate**
Theme name: **nearbuff**
Target item: **inside**
Equation: **inside = 1**

The following command line is used to initiate the **update** command. The polygon contained in NEARBUFF will be 'pasted' into FARBUFF:

Command: **update**
Theme name: **farbuff**
Update theme name: **nearbuff**
Output theme name: **zones**

The following are the themes used to create ZONES:




Input cover Update cover Output cover

The ZONES feature attribute table appears as follows:

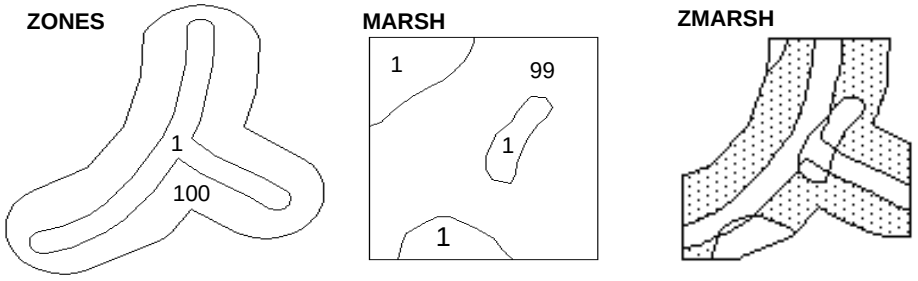
| AREA | PERIMETER | ZONES_ | ZONES_ID | INSIDE |
|---|---|---|---|---|
| -2.60010E+07 | 24409.500 | 1 | 0 | 1 |
| 21485932.000 | 43188.090 | 2 | 1 | 100 |
| 4515035.000 | 18778.590 | 3 | 1 | 1 |

The next step in this model is to overlay the MARSH theme with the newly generated ZONES theme. The desired result is a polygon theme in which each polygon can be identified as representing marsh or nonmarshland, as well as whether it is within the required distance from one of the highways.

As noted in an earlier example, the commands **identity**, **esri_intersect** and **esri_union** can each perform a polygon-on-polygon overlay. In this case, **esri_intersect** will be used since it will also restrict the feature extent of the output theme to the area in common to both the input themes.

The following command line is used for this overlay:

Command: **esri_intersect**
Input theme name: **zones**
Intersect theme name: **marsh**
Output theme name: **zmarsh**




Input theme Intersect theme Output theme

*The shaded areas represent nonmarsh portions of the study area that are between 0.25 and 1.25 kilometers of a highway (i.e., TYPE is 99 and INSIDE is 100).*

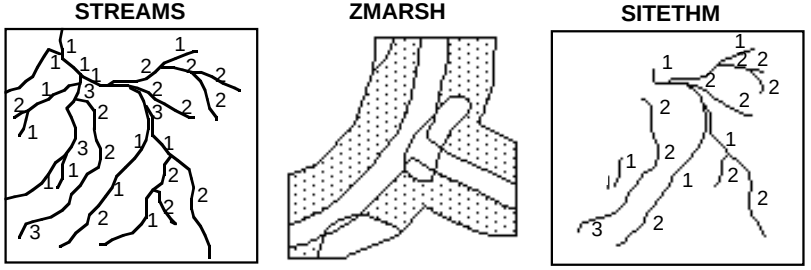
| AREA | PERIMETER | ZMARSH_ | ZMARSH_ID | INSIDE | TYPE |
|---|---|---|---|---|---|
| -2.60010E+07 | 24409.480 | 1 | 0 | 1 | 0 |
| 234022.900 | 2513.122 | 2 | 1 | 100 | 1 |
| 5165234.000 | 12382.630 | 3 | 2 | 100 | 99 |
| 2552426.000 | 14037.810 | 4 | 3 | 1 | 99 |
| 3476536.000 | 10345.510 | 5 | 4 | 100 | 99 |
| 392700.600 | 3113.878 | 6 | 5 | 100 | 1 |
| 633648.400 | 4879.152 | 7 | 6 | 1 | 1 |

*This sample PAT for the ZMARSH theme includes items from the feature attribute tables of both the overlaid themes. Only a few of the actual items and records are listed here.*

There is one more overlay to perform before this model is complete. The **esri_intersect** command will now be used to overlay the STREAMS theme with ZMARSH. This line-in-polygon overlay is initiated as follows:

Command: **esri_intersect**
Input theme name: **streams**
Intersect theme name: **zmarsh**
Output theme name: **sitethm**

The following themes are used to create SITETHM and a portion of its feature attribute table:




Input theme Intersect theme Output theme

Listing of SITETHM feature attribute table: *(Only a few of the records and items contained in the SITETHM Arc Attribute Table are listed here.)*

| LENGTH | SITECOV_ | SITECOV_ID | CLASS | INSIDE | TYPE |
|---|---|---|---|---|---|
| 262.750 | 1 | 1 | 1 | 100 | 1 |
| 730.750 | 2 | 1 | 1 | 100 | 99 |
| 498.250 | 3 | 1 | 1 | 1 | 99 |
| 999.875 | 4 | 1 | 1 | 100 | 99 |
| 241.624 | 5 | 7 | 1 | 100 | 1 |
| 527.090 | 6 | 7 | 1 | 1 | 1 |
| 430.579 | 7 | 8 | 2 | 1 | 99 |
| 161.701 | 8 | 7 | 1 | 1 | 1 |
| 237.342 | 9 | 8 | 2 | 100 | 99 |
| 896.737 | 10 | 7 | 1 | 100 | 1 |

The theme SITETHM contains the information necessary to extract the stream features that meet all the selection criteria for potential park locations (represented by the items CLASS, INSIDE and TYPE). It is now possible to identify those streams of CLASS 2 (small, first- and second-order streams with characteristics desirable for a park site) that are within 0.25 and 1.25 kilometers of a highway, represented by an INSIDE value of 100, and flow through a nonmarshy area (have a TYPE of 99). The ArcCAD command **extract** can be used to extract features from a theme by selecting the desired features through the use of logical expressions. The set of stream segments meeting all selection criteria can therefore be isolated in a new theme as follows:

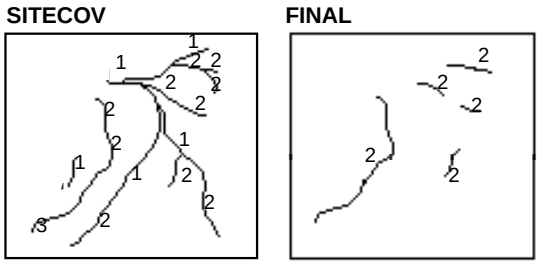
First select the set of features in SITETHM that meet our criteria using the **reselect** command:

Command: **reselect**
Theme name: **sitethm**
Reselect by?: **expression**
Enter logical expression: **class = 2 and inside = 100 and type = 99**

Now extract these features in the current feature selection set for the SITETHM theme and place them in a theme called FINAL:

Command: **extract**
Input theme name: **sitethm**
Output theme name: **final**

The theme FINAL contains these arcs:



#### Step 4 Perform the tabular analysis

In the previous step, the **extract** command was used to create a theme that contains only those streams that meet the criteria for park sites. No further tabular analysis is necessary to identify potential park sites.

As an alternative to performing a spatial **extract** to create a new theme, we could have performed a similar operation by using the ArcCAD commands **reselect** and **calculate** or your database management system to identify the arc features contained in SITETHM that meet the criteria for potential park sites. If we choose this alternative, we would need to add an item at this step to contain a code identifying the arcs that meet the park criteria and those that do not.

#### Step 5 Evaluate and interpret the results

In this model, all stream segments within the study area that are capable of becoming a park location are identified. However, we have not considered factors that limit the suitability of park sites, such as the surrounding zoning classification and the current land use of a potential park site. For example, stream segments in areas zoned for industrial use would probably be disqualified as potential park sites.

A potential park site must also be available for acquisition by the public agency responsible for building the park. Most sites are probably not available. It may be important to identify potential properties available for purchase on which park sites can be located. A number of additional criteria will need to be evaluated at this step. Is the size of potential properties appropriate for the park site, is the surrounding land use inconsistent, is the price reasonable, and so on.

#### Step 6 Refine the analysis as necessary

In this case, no further analysis is performed. Instead, final reports and maps are passed on to the division responsible for evaluating the potential park sites to help identify alternative sites which might be selected and purchased.

#### Step 7 Produce final maps and tabular reports of the results

ArcCAD software's display and database management commands can be used to create final maps and final reports, respectively.

# Example: Calculate expected cost of a major flood

This example demonstrates the use of spatial operations and tabular analysis functions in estimating property loss due to flooding. A polygon overlay is performed to merge the attributes of two separate themes. One contains parcel boundary information and the other delimits the extent of a floodplain. The subset of polygons that represent the portions of the lots that fall within the floodplain is then selected from the overlay theme. A mathematical equation is applied to their feature attributes to estimate a 'loss' value. The example follows the analysis steps outlined previously.

### Step 1 Establish analysis objectives and criteria

The objective is to determine a total expected dollar loss to residents from a major flood. The following criteria will be used to estimate the expected loss:

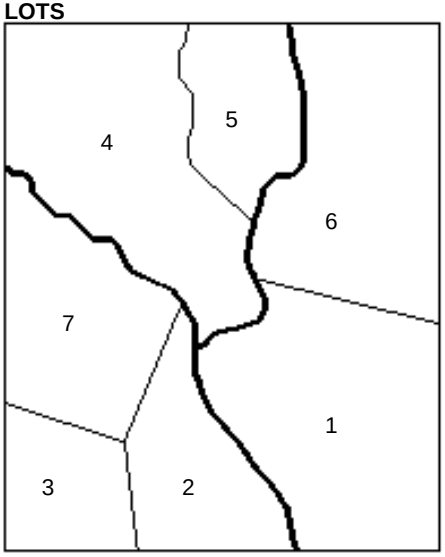
- The land for which losses are calculated must be zoned as residential and be in the floodplain.
- The upper limit of the floodplain is considered to be the 500-meter contour.
- The expected dollar loss is the value of the land in the floodplain multiplied by a loss factor. The loss factor is based upon the stability of the soil type that comprises each land parcel.

### Step 2 Prepare the data for spatial operations

The initial database consists of a polygon theme called LOTS, which identifies the seven lots in the study area and contains attribute information about the zoning, dollar value and soil type of each lot; a polygon theme called CONTOURS, which identifies the elevation range of each part of the study area; and a lookup table called LOOKUP, containing information about each soil type including a loss factor based on the stability of each soil type.

The following themes will be used:

**LOTS**




*Each polygon is numbered with its LOTS_ID. The thicker lines separating the lots show where a river flows through the study area.*

**LOTS theme PAT**

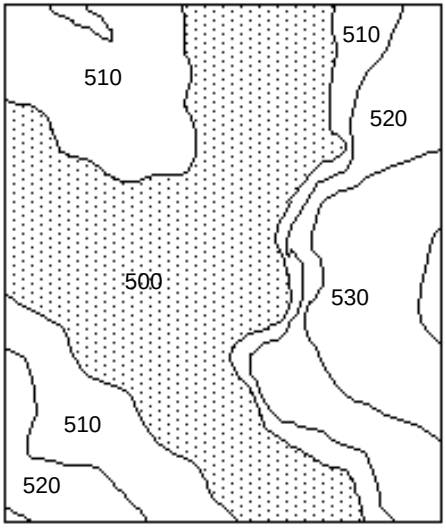
| AREA | PERIMETER | LOTS_ | LOTS_ID | ZONING | VALUE | SOILS |
|---|---|---|---|---|---|---|
| -2.44736E+O7 | 19976.000 | 1 | 0 | | 0 | |
| 5425867.000 | 9692.300 | 2 | 1 | R1 | 10000 | A |
| 2642950.000 | 7692.700 | 3 | 2 | R2 | 50000 | C |
| 2006077.000 | 5799.300 | 4 | 3 | C | 30000 | B |
| 4954105.000 | 10327.200 | 5 | 4 | C | 90000 | A |
| 1872968.000 | 5660.800 | 6 | 5 | R1 | 100000 | C |
| 4504634.000 | 9336.900 | 7 | 6 | R1 | 115000 | A |
| 3066946.000 | 7528.900 | 8 | 7 | R2 | 100000 | C |

**LOOKUP**

| SOILS | COMPACTION | STABILITY | FACTOR |
|---|---|---|---|
| A | 50 | LOW | 0.75 |
| B | 95 | GOOD | 0.25 |
| C | 83 | MODERATE | 0.50 |

*In the LOTS theme PAT, zones containing 'R' are residential. LOOKUP is a record theme that is a lookup table. SOILS is the relate item, and FACTOR contains the loss factor for each soil type.*

**CONTOURS**




*Each polygon represents a 10-meter range of elevation. The value of ELEV is the maximum elevation of each polygon. The floodplain (shaded) is represented by an ELEV of 500 meters.*

**CONTOURS theme PAT**

| AREA | PERIMETER | CONTOURS_ | CONTOURS_ID | ELEV |
|---|---|---|---|---|
| -2.44736E+07 | 19976.000 | 1 | 0 | 0 |
| 2683509.000 | 7569.202 | 2 | 1 | 510 |
| 96275.090 | 1596.552 | 3 | 2 | 520 |
| 10333784.000 | 19549.020 | 4 | 3 | 500 |
| 1453374.000 | 15187.650 | 5 | 4 | 510 |
| 3122228.000 | 15325.770 | 6 | 5 | 520 |
| 2804079.000 | 7.697.589 | 7 | 6 | 530 |
| 180.913.300 | 2531.523 | 8 | 7 | 540 |
| 2140607.000 | 8372.975 | 9 | 8 | 510 |
| 1520184.000 | 7.641.943 | 10 | 9 | 520 |
| 138593.700 | 1539.258 | 11 | 10 | 530 |

It will be important to calculate the value-per-unit area for each polygon of the LOTS theme so that the value of the portion of each lot that falls within the floodplain can be calculated later on. The following sequence of commands will add an additional item to the LOTS theme PAT and calculate a per-unit-area dollar value for each lot. Note that additional decimal places are used to ensure that enough mathematical precision is available to store the value-per-unit-area calculations.

Command: **additem**
Theme name: **lots**
Item name: **val_unit**
Item width: **6**
Item type?: **numeric**
Item decimal places: **4**
Start item: <last> Enter

Command: **calculate**
Theme name: **lots**
Target item: **val_unit**
Equation: **value / area**

Command: **listdb**
Theme name: **lots**

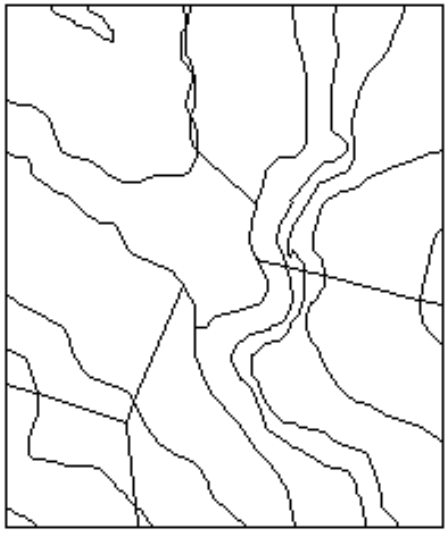
| AREA | PERIMETER | LOTS_ | LOTS_ID | ZONING | VALUE | SOILS | VAL_UNIT |
|---|---|---|---|---|---|---|---|
| -2.44736E+07 | 19976.000 | 1 | 0 | | 0 | | 0.0000 |
| 5425867.000 | 9692.314 | 2 | 1 | R1 | 10000 | A | 0.0018 |
| 2642950.000 | 7894.714 | 3 | 2 | R2 | 50000 | C | 0.0189 |
| 2006077.000 | 5799.319 | 4 | 3 | C | 30000 | B | 0.0150 |
| 4954105.000 | 10327.260 | 5 | 4 | C | 90000 | A | 0.0182 |
| 1872968.000 | 5660.854 | 6 | 5 | R1 | 100000 | C | 0.0534 |
| 4504634.000 | 9336.922 | 7 | 6 | R1 | 115000 | A | 0.0255 |
| 3066946.000 | 7528.945 | 8 | 7 | R2 | 100000 | C | 0.0326 |

### Step 3 Perform the spatial operations

ArcCAD provides three commands that can perform a polygon-on-polygon topological overlay: **identity**, **esri_intersect** and **esri_union**. The primary difference in these commands is the physical extent of the output coverage that they generate. These differences are noted in the previous 'Spatial analysis concepts' section. In this example, **esri_union** will be employed, although any one of these commands could be used since the features of the themes being overlaid cover the same area. The following command lines will generate a new overlay theme called FLDLOTS:

Command: **esri_union**
Theme name: **lots**
Union theme name: **contours**
Output theme name: **fldlots**

**FLDLOTS**



*This is the resulting overlay theme FLDLOTS. It contains 27 polygons, each one having the combined attributes of the original themes.*

**FLDLOTS.PAT**

| AREA | FLDLOTS_ID | ZONING | SOILS | VAL_UNIT | ELEV |
|---|---|---|---|---|---|
| -2.44736E+07 | 0 | | | 0.0000 | 0 |
| 2644575.000 | 1 | C | A | 0.0182 | 510 |
| 96246.190 | 2 | C | A | 0.0182 | 520 |
| 8144.063 | 3 | C | A | 0.0182 | 520 |
| 1834286.000 | 4 | R1 | C | 0.0182 | 500 |
| 912407.500 | 5 | R1 | A | 0.0534 | 500 |
| 750218.900 | 6 | R1 | A | 0.0255 | 510 |
| 1389628.000 | 7 | R1 | A | 0.0255 | 520 |
| 22554.440 | 8 | R1 | C | 0.0534 | 510 |
| 22358.310 | 9 | C | A | 0.0182 | 500 |

*This listing is a subset of selected items from the FLDLOTS theme PAT. Only the first ten records of the PAT are listed.*

As mentioned, the items listed above are only a subset of the total number of items created by the **esri_union** overlay. The following is an items listing of the FLDLOTS theme PAT. Note that the list includes all items from the themes that were overlaid.

Command: **items**
Theme name: **fldlots**
Item display options: **long**

| COLUMN | ITEM NAME | WIDTH | TYPE | N.DEC |
|---|---|---|---|---|
| 1 | AREA | 13 | N | 6 |
| 14 | PERIMETER | 13 | N | 6 |
| 27 | FLDLOTS_ | 11 | N | 0 |
| 38 | FLDLOTS_ID | 11 | N | 0 |
| 49 | LOTS_ | 11 | N | 0 |
| 60 | LOTS_ID | 11 | N | 0 |
| 71 | ZONING | 4 | C | 0 |
| 75 | VALUE | 7 | N | 0 |
| 82 | SOILS | 1 | C | 0 |
| 83 | VAL_UNIT | 6 | N | 4 |
| 89 | CONTOURS_ | 11 | N | 0 |
| 100 | CONTOURS_I | 11 | N | 0 |
| 111 | ELEV | 4 | N | 0 |

**Step 4 Perform the tabular analysis**

Use ArcCAD commands (or your database manager) to calculate the expected dollar loss for each of the land parcels in FLDLOTS. First add an item to FLDLOTS to contain the results of the calculations.

Command: **additem**
Theme name: **fldlots**
Item name: **loss**
Item width: **10**
Item type?: **numeric**
Item decimal places: **2**
Start item: <last> Enter

To calculate the expected loss value for each lot, the following operations will be performed:

1) **Reselect** the polygons of FLDLOTS meeting the selection criteria.

2) **Relate** the lookup table LOOKUP by the relate item SOILS.

3) **Calculate** the loss value. The dollar value that each polygon represents in FLDLOTS (AREA * VAL_UNIT) will be multiplied by a loss factor (FACTOR) from LOOKUP according to its soil type.

You should be familiar with these operations. Refer to the *ArcCAD Command Reference* for information on using the ArcCAD commands employed here.

**FLDLOTS PAT**

| AREA | FDLOTS_ID | ZONING | SOILS | VAL_UNIT | ELEV | LOSS |
|---|---|---|---|---|---|---|
| -2.44736E+07 | 0 | | | 0.0000 | 0 | 0 |
| 2644575.000 | 1 | C | A | 0.0182 | 510 | 0 |
| 96426.190 | 2 | C | A | 0.0182 | 520 | 0 |
| 8144.063 | 3 | C | A | 0.0182 | 500 | 0 |
| 183426.000 | 4 | R1 | C | 0.0534 | 500 | 0 |

**LOOKUP**

| SOILS | COMPACTION | STABILITY | FACTOR |
|---|---|---|---|
| A | 50 | LOW | 0.75 |
| B | 95 | GOOD | 0.25 |
| C | 83 | MODERATE | 0.50 |

A loss value will only be calculated for those polygons representing land that is both in the floodplain (ELEV = 500) and residentially zoned (ZONING CN 'R').

Command: **reselect**
Theme name: **fldlots**
Reselect by?: **expression**
Enter logical expression: **elev = 500 and zoning cn 'R'**

Command: **relate**
Relate?: **on**
From theme: **fldlots**
To theme: **lookup**
Relate item: **soils**
Relate type: **linear**

Command: **calculate**
Theme name: **fldlots**
Target item: **loss**
Equation: **( area * val_unit ) * #factor**

Command: **listdb**
Beginning record number: **1**
Ending record number <last>: Enter
List all items?: **no**
Item: **area**
Item: **lots_id**
Item: **zoning**
Item: **soils**
Item: **val_unit**
Item: **#factor**
Item: **loss**

| AREA | LOTS_ID | ZONING | SOILS | VAL_UNIT | #FACTOR | LOSS |
|---|---|---|---|---|---|---|
| 1834286.250 | 5 | R1 | C | 0.0534 | 0.50 | 48967.36 |
| 912407.500 | 6 | R1 | A | 0.0255 | 0.75 | 17469.82 |
| 2305955.000 | 7 | R2 | C | 0.0326 | 0.50 | 37593.67 |
| 1457468.250 | 1 | R1 | A | 0.0018 | 0.75 | 2014.61 |
| 1610722.500 | 2 | R2 | C | 0.0189 | 0.50 | 15236.03 |

The sum of these records is the total expected dollar loss to residents from a major flood. The **statistics** command will calculate this sum and display the total value of LOSS on the screen; for example,

Command: **statistics**
Theme name: **fldlots**
Item: **loss**

Total statistics for LOSS

| CASE | COUNT | MINIMUM | MAXIMUM | SUM | MEAN |
|---|---|---|---|---|---|
| TOT | 5 | 2014.610000 | 48967.360000 | 121281.490000 | 24256.298000 |

### Step 5 Evaluate and interpret the results

In the calculation of flood loss, the assumption is made that all portions of each property are of equal value. The valuation used is recorded as the dollar value per unit area. This does not consider the locations of structures and buildings. In fact, dollar loss could be higher or lower depending upon whether or not the structures are located within the floodplain.

If a theme of building footprints and structure locations is available, we may wish to perform the analysis using this additional information. However, we would also have to separately determine land value and building value, as well as perform a more detailed evaluation of what kind of losses each would incur from flooding.

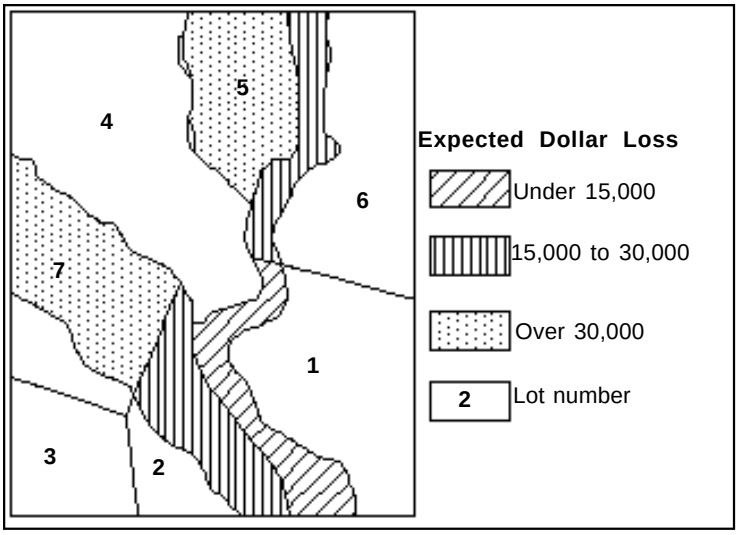
### Step 6 Define the analysis as necessary

Due to the complexity of the analysis and the lack of additional information, the decision was made to forego further analysis for the time being. Instead, these limitations were used as an argument to perform a more detailed evaluation of each residential property located within the floodplain by performing an on-site inspection.

### Step 7 Produce final maps and tabular reports of the results

The ArcCAD and AutoCAD display commands are used to produce final maps while the ArcCAD database commands or your database management system can be used to produce final reports. Following is an example of each for this analysis:

**Final AutoCAD drawing**




**Database report**

| RESIDENTIAL LOT NUMBER | %AREA IN FLOODPLAIN | VALUE | EXPECTED LOSS |
|---|---|---|---|
| 1 | 26.86 | 10,000 | 2,014.61 |
| 2 | 60.94 | 50,000 | 15,236.03 |
| 5 | 97.93 | 100,000 | 48,967.36 |
| 6 | 20.25 | 100,000 | 17,469.82 |
| 7 | 75.19 | 115,000 | 37,593.67 |
| | | $375,000 | $121,281.49 |

# Spatial analysis processing considerations

The approaches used in the previous examples, though basic, demonstrate a number of important principles used in spatial analysis. This section identifies some additional considerations that are also important.

## Multiple approaches

There are a number of methods that can be used to solve each analysis problem, and there may be varying ideas about how to develop a solution. For example, you may have some ideas about how you might have developed a different model for performing the analyses in the examples above. The spatial analysis tools contained in ArcCAD provide you with the flexibility to evaluate different approaches and to select one that best meets your needs. It will often be important to try a number of alternatives so that more than one method, approach, or viewpoint can be tested. These GIS functions give you the ability to evaluate different approaches to solve spatial analysis problems.

## Additional spatial operations

There are a number of spatial operations that were not demonstrated in the examples, yet they are as valuable as the operations used. You should be aware of the operations they perform and be able to apply them when they are appropriate.

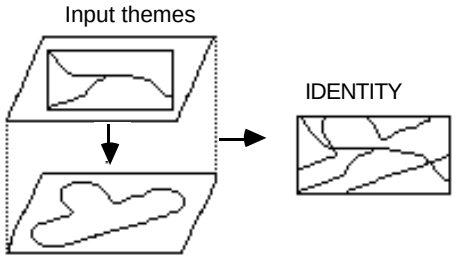
**Eliminate** is used to remove sliver polygons by dissolving them into adjacent polygons. When performing polygon-on-polygon overlays, sliver removal is essential before the results are analyzed (particularly when using natural resource data layers that may have polygons with ‘fuzzy’ boundaries).

**Clip** is used as a template to clip out a portion of a larger theme for detailed analysis. It is one of the most commonly used overlay functions.

**Dissolve** merges adjacent polygons that have the same value for a specified item or group of items. **Dissolve** is often used to remove map sheet borders from a theme that was mapjoined from a series of adjacent themes.

## Using identity on buffer results

Zones created by the **buffer** command often extend beyond the boundaries of the study area. This may occur when the features being buffered fall close to the edge of the study area. In these situations, the **identity** command is often used (instead of **esri_union** or **esri_intersect**) after **buffer** to clip the buffer zones at the edge of the study area, yet still retain portions of the study area that are not within a buffer zone; for example,




## Divisible attributes

Many times, the feature attributes resulting from an overlay operation do not apply to the output feature—they only apply to features from one of the themes used as input to the overlay operation. For example, LEGAL_AREA may be an attribute of parcels that is overlaid with soil types. After the overlay, some parcels may be split into several pieces by the soils polygons. Although each resulting polygon has a value for LEGAL_AREA, the LEGAL_AREA item does not apply to the separate pieces of a parcel. Another example of such an item is one that holds populations of polygons. If the polygon is split after an overlay, the value of the population item is not applicable for the output polygons since it is the value of the entire input polygon.

Prior to the overlay operation, items with such data must be identified and a decision reached about how to manage such attributes, if necessary. For example, population might be allocated to each polygon by multiplying the proportion of the original area in each polygon times the population. In this case, a polygon that is split into two equal parts would receive half of the original population value. In other cases, you would not want to divide the value and assign it to each portion of the original polygon. However, in this situation, you should be aware that such values are not attributes of the output polygons.

## Renaming coverages after overlay

**Identity**, **esri_intersect** and **esri_union** merge all input theme items into the output feature attribute table including the Internal-ID and User-ID items from each input theme. These items can be used to relate the output theme feature attribute table to each of the input themes or to other database files, which can be related by User-ID. Remember that the Internal-ID and the User-ID of a theme's feature attribute table are named for the coverage GIS data set of that theme and not for the theme name. If you rename the input theme's coverage GIS data sets after an overlay operation, their new Internal-ID and User-ID item names do not agree with the corresponding item names in the overlay theme. If relating these tables, it is necessary to rename or create items in the overlay theme that allow you to perform the relate or **joinitem**.

Refer to the **moditem** and **additem** commands as well as the user's guide for your database manager for additional information on modifying items or relating files.

## Item names for overlay coverages

The overlay commands **identity**, **esri_intersect** and **esri_union** merge feature attributes from both input themes. They perform a **joinitem** internally to merge items from each of the two input themes into the output feature attribute table. If both input tables contain an item with the same name, only the item for the theme listed first on the command line is saved in the output theme's table. Therefore, to maintain all attribute values in the output theme, it is important that each input theme contains unique item names before performing the overlay. This is true for all items including and to the right of the Internal-ID and User-ID in the feature attribute table.

# ArcCAD utilities

A set of utility commands is provided with ArcCAD® software that can be used to directly manage coverages and database files. These commands are designed to operate on coverages and database files rather than on themes. Commands are available for copying, renaming and listing coverage database files as well as for converting coverage data sets so that they can be used with ARC/INFO® software on other computer platforms. These utility commands are located under the TOOLS pull-down menu.

## Managing ArcCAD system settings

The **esri_prefs** command controls default directory locations and a number of other ArcCAD system settings.

## Managing database files

Three commands are available for managing database files: **browse**, **wbrowse** and **dbdir**. The **browse** and **wbrowse** commands are useful for displaying or updating records in any database file. Feature attribute tables, record themes and standalone dBASE files can be used with the **browse** and **wbrowse** commands. **Dbdir** can be used to get a list of all database files in a specified directory.

# Managing coverages

Several commands are provided for manipulating coverages. The **copycov** command makes a copy of a coverage, while **renamecov** changes the name of a coverage. **Describe** lists information about a coverage such as the number of features it contains and whether feature topology is current or not. **Rebox** sets the coverage boundary (BND) to the extent of features in a coverage and deletes tic features outside of this feature extent. This command is useful for removing excess tics when features have been removed from a coverage, shrinking its geographic extent. The **listcov** command lists the coverages in a specified directory.

# Online help

ArcCAD provides online help for every command. To get help information about any ArcCAD command, use the **archelp** command.

# ArcCAD data conversion commands

ArcCAD has three commands that will allow you to transfer data between ESRI products. ArcCAD can read and write interchange files using **esri_import** and **esri_export**. ArcCAD can also read ArcView® shape files and convert them to coverages using the **shapein** command. For more information refer to the **esri_import**, **esri_export** and **shapein** command references.

# ArcCAD limitations

The following limitations apply to certain ArcCAD commands. These limitations may affect your ArcCAD applications. Keeping these limitations in mind may save you time and effort when running ArcCAD.

## Features

| | |
|---|---|
| Maximum number of point, arc or polygon features in a coverage | 262,144 |
| Maximum number of tic features in a coverage | 10,000 |
| Maximum number of arc features per polygon | 10,000 |
| Maximum number of vertices per arc feature | 500 |
| Maximum number of concurrent feature selections | 128 |
| Maximum number of arc features that can connect at a single node | 100 |

## Record length

The maximum length of a database record is 128 items or 4,000 characters (bytes). This limit can be avoided by using smaller files and relating them when necessary. The wider a database file is, the slower the execution speed of many ArcCAD commands.

## Coordinate rounding

Themes containing very large coordinate values may experience some coordinate rounding during overlay operations due to single-precision data limits. This is especially true for those themes also having a small range of x or y coordinate values. This limitation can be avoided by using the **xyshift** command to apply a constant offset to the coordinates.

## Disk space

Processing space required by some ArcCAD commands (such as **clean**, **build** and spatial overlay commands) can be from three to fourteen times the input GIS data set size. Processing space is empty disk space that is temporarily filled with files during command execution. The temporary files are deleted when the commands finish executing. A general rule of thumb to follow is this: as the number of coordinates per arc feature increases, the number of intersections increases, or as the record length of the feature attribute table increases, the available processing space will also need to increase. Temporary files are created in the directory specified in **esri_prefs**. If a command fails because it runs out of temporary disk space, you might try starting ArcCAD on a disk drive with more free space.

## dBASE MEMO items

dBASE MEMO items should be used with caution since some operations may result in loss of data in the MEMO field upon completion. In general, MEMO items should never be maintained in a feature attribute table. If you want to use MEMO item types, they should be maintained in separate record themes and accessed through the **relate** command. Specific details are documented in the command references for individual commands.

## Themes

- The maximum length of a theme name is 31 characters.

- The maximum number of themes that can be contained in an ArcCAD database is 511.

- You cannot define more than one theme with the same feature class and GIS data set. The exception to this is record themes. Any number of record themes can point to the same GIS data set.

## GIS data sets

- The maximum length of the name of a GIS data set is 8 characters.
- The maximum length of the full pathname to a GIS data set is 64 characters. This includes the GIS data set name.
- A coverage GIS data set cannot contain both point and polygon features.

## Item names

- The maximum length of an item name is 10 characters.
- The only legal characters in item names are alphabetical characters, numbers and the underscore character. If you attempt to create an item name that contains illegal characters, those illegal characters are automatically converted to the underscore character.
- Item names cannot begin with a number.

# ArcCAD database structure

## Coverage data files

ArcCAD stores coverages as subdirectories having the same name as the coverage. All coverage data files reside in this subdirectory. The directory structure of a coverage is outlined below with a brief description of the most common coverage files:

| | | |
|---|---|---|
| AAT.DBF | - | arc attribute table |
| ARC | - | arc feature coordinates and topology |
| ARF | - | arc feature cross-reference file |
| ARX | - | arc file index |
| BND.DBF | - | coverage minimum and maximum coordinates |
| CNT | - | polygon centroid table |
| LAB | - | label point coordinates and topology |
| MSK | - | edit area masks |
| PAL | - | polygon topology |
| PAT.DBF | - | polygon/point attribute table |
| PFF | - | polygon filter file |
| PRF | - | polygon/point cross-reference file |
| TIC.DBF | - | tic coordinates and IDs |
| TOL | - | coverage processing tolerances |
| TXT | - | coverage annotation features |
| TXX | - | annotation index file |

All files with a .DBF file extension are dBASE-compatible database files. All other files store feature coordinates and topology. These *geometry files* are binary files stored in a proprietary format and are not accessible by the user.

Coverage database files (feature attribute tables) can be accessed in ArcCAD by specifying

```
coverage.data_file
```

or

```
coverage\data_file
```

For example, specifying either ROADS.AAT or ROADS\AAT accesses the arc attribute table associated with the ROADS coverage.

The following names represent the same database file:

```
C:\DIRA\WORK1\PARCELS.PAT

C:\DIRA\WORK1\PARCELS\PAT

PARCELS.PAT          Assumes your current directory is: C:\DIRA\WORK1
```

The convention of specifying a database file name as `name.fil` implies that '`name`' is a subdirectory name and that '`fil`' is a .DBF file called '`fil.DBF`' under the directory name '`name`'. Another way of specifying `name.fil` is '`name\fil.DBF`'.

# Link file directories

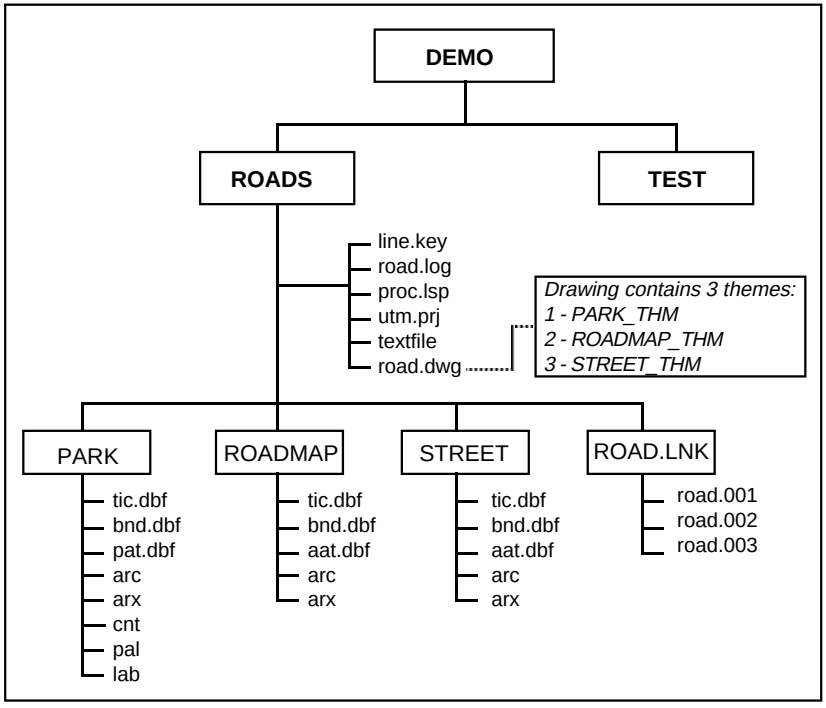
ArcCAD stores internal data structures that manage entity-feature link information in *link files*. Link files are stored in a directory that has the same name as a drawing with the extension of '.LNK'. For example, if a drawing is called ROAD.DWG, ArcCAD will store link files for themes in ROAD.DWG in a directory called ROAD.LNK. The entity-feature link information for each theme defined in the drawing is stored in a separate link file in the drawing's link file directory. Each link file has the same name as the drawing with the file extension '*.nnn*', where *nnn* is the internal theme ID. Since a drawing can contain a maximum of 511 themes, a link directory can contain a maximum of 511 link files. The link file directory must exist in the same directory as its corresponding drawing. When a drawing containing themes is loaded into ArcCAD, ArcCAD reads the entity-feature link data from that theme's link files. If ArcCAD cannot find the current drawing's corresponding link file directory or if one of the link files for any theme is missing, the drawing's link files must be reconstructed. This process can take some time for large drawings as every entity in the drawing must be examined for an entity-feature link. The entity-feature link data structures are then recreated for each entity in the drawing that has a corresponding theme feature.

To avoid rebuilding the link files each time a drawing with themes is loaded into ArcCAD, always keep the drawing and its associated link file directory together when a drawing with themes is copied to another directory.

# ArcCAD workspaces

ArcCAD databases are usually organized into *workspaces*. Each ArcCAD workspace is a directory that contains one or more drawings containing themes and GIS data sets (coverages and database files) associated with those themes.

Here is an example of an ArcCAD workspace.




The workspace used in these examples is named ROADS and is located in a directory named DEMO. Here is a list of the files contained in the workspace:

line.key - a key file for line symbols
road.log - the ArcCAD log file for the ROAD drawing
proc.lsp - a lisp routine
utm.prj - a command file used by the **project** command
textfile - a text file
road.dwg - an AutoCAD drawing containing themes

Here is a list of the directories contained in this workspace:

| | | |
|---|---|---|
| PARK | - | the GIS data set (a coverage) for the polygon theme PARK_THM |
| ROADMAP | - | the GIS data set (a coverage) for the line theme ROADMAP_THM |
| STREET | - | the GIS data set (a coverage) for the line theme STREET_THM |
| ROAD.LNK | - | the link file directory for the ROAD.DWG drawing file |

The feature attribute tables for these coverages are under their respective coverage subdirectory names and include .DBF extensions.

The link file directory for the ROAD.DWG drawing contains the link files for all three themes defined in ROAD.DWG.

# More on ArcCAD coverage features

This appendix summarizes important information about coverages in ArcCAD.

## Coverage feature types

There are five feature types stored in coverages.

### Points

Points can be used for two different purposes in a coverage:

- To represent point features
- To assign User-IDs to polygon features

Any single point cannot be used for both purposes.

When a point is used to assign a User-ID to a polygon, it is known as a *label point.* One label point is placed within each polygon. This is done by placing a point somewhere inside the boundary of the polygon. When polygon topology is created, the label point's User-ID becomes the polygon's User-ID. If a polygon does not contain a label point, that polygon's User-ID will be 0. If a polygon contains more than one label point, there is no set rule to determine which label point will be used to assign a User-ID to the polygon. Polygons should only have one label point, otherwise, unique attributes describing that polygon cannot be attached to that polygon.

Since points are used to represent both point features and polygon label points in ArcCAD, point and polygon features cannot be stored in the same coverage and must be organized into separate coverages.

## Lines (arcs)

Lines represent linear features, the borders of polygons, or both. Lines in ArcCAD are referred to as *arc features*. A linear feature, such as a river, may be made up of several individual arc features. Arc features can be topologically linked at their endpoints (nodes) to other arc features and to the areas (polygons) on each side of them. Note that the ArcCAD arc feature refers to a linear feature or line and not to the AutoCAD arc entity type.

## Polygons

Polygons represent area features. Polygons are defined topologically by the series of arc features that compose its border and by a label point positioned inside its border. The label point ID is used to assign the polygon a User-ID. Since polygon boundaries are defined topologically (i.e., by a list of arc features), a one-to-one relationship cannot be established between an arc feature and a particular polygon (an arc feature always has a polygon on either side). Therefore, entity-feature links cannot be created for polygon boundaries when polygons are displayed using ArcCAD display commands. A line theme with the same GIS data set as the polygon theme must be created to establish entity-feature links for the polygon boundaries if required.

## Tics

Tics are registration or geographic control points for a coverage. They are identified by known coordinates on a map and used to orient a coverage. They allow all coverage features to be registered to a common coordinate system (e.g., UTM meters, State Plane feet, and so on). Additionally, other coverages such as another layer of the same area or adjacent coverages can be spatially related using the same tics as geographic control.

Tics are usually calculated from known reference points found on a base map. For example, known latitude–longitude values may be recorded for a series of points on the map.

It is important to record tic coordinates as accurately as possible. They are used to orient the coverage for map merging and overlay. At least four tics must be created for each coverage. When creating a coverage from a drawing, ArcCAD will automatically create four tics at the corners of the extent of the features. Every coverage has tic features even if there is no tic theme defined for them.

Additional tics can be added at known locations to serve as control points. Tic features cannot be created from AutoCAD entities. To create or modify tic features, you must edit the tic theme's database file using ArcCAD attribute management commands or your database management system.

## Annotation

Annotation is text used to label theme features. Annotation is not topologically linked with any other features. Annotation is used only for display purposes; it is never used in analytical processes. An important characteristic of annotation is the annotation level. A level is a category of annotation within a coverage. Levels are a convenient way of organizing annotation. For example, in a coverage GIS data set containing roads, it might be useful to store annotation for major highways in one level and annotation for secondary roads in another. Annotation levels are identified by number, starting at 1. There is no limit to the number of annotation levels. ArcCAD commands that manipulate annotation allow you to access annotation by level. Annotation is stored in a coverage. The GIS data set of an annotation theme stores the location, size, text symbol and text string of each annotation feature.

Annotation features are often stored in GIS data sets that store other feature types such as points, lines and polygons.

# Feature numbers in PATs and AATs

Polygon, line and point features in ArcCAD have two numbers associated with them:

- Cover_ - referred to as the Internal-ID number (software assigned)
- Cover_ID - referred to as the User-ID (user assigned)

### Internal-ID numbers

An Internal-ID number is actually the record number of a record within another file that contains data about a feature. Internal-ID numbers are important for building topology lists (e.g., an arc feature with sequence number 1 is the arc feature whose coordinates appear in the first record in an ARC file; polygon 1 with sequence number 1 is defined by four arc features whose sequence numbers are 3, 12, 2, 1).

Internal-ID numbers are automatically created and maintained by ArcCAD using **clean** and **build**. You should never manually alter these numbers in the feature attribute tables. Internal-ID numbers are unique for each feature but do not necessarily remain fixed. They can change, especially as coverages are updated or corrected.

Internal-ID numbers are provided in feature attribute tables to allow sophisticated types of GIS analyses and link relates to be performed. You should consider the Internal-ID item in a feature attribute table to be read-only.

## User-IDs

A User-ID is an integer value assigned by the user to each feature (e.g., a polygon). It will not change until you change it. The User-ID can be used to store actual data values or, more often, to relate other items to particular map features. Note that it is most often important to have unique User-IDs so that the items added to a PAT or AAT are not lost during update. For example, **clean** and **build** both use the User-ID to merge in any existing, additional items when updating an existing PAT or AAT. In this case, a relational join (i.e., **joinitem**) is performed using the User-ID as the relate item.

# Differences between ArcCAD and PC ARC/INFO Version 3.5

There are a few functional differences between ArcCAD and PC ARC/INFO® Version 3.5. These are noted below.

## Data conversion routines

ArcCAD does not include the PC ARC/INFO data conversion routines distributed with PC DATA CONVERSION™ software.

## Network analysis

ArcCAD does not include the PC ARC/INFO programs ROUTE and ALLOCATE distributed with PC NETWORK™ software.

## Feature editing and display

Feature editing and display in ArcCAD are performed using a combination of AutoCAD and ArcCAD commands. ArcCAD does not include the PC ARC/INFO programs distributed with PC ARCEDIT™ and PC ARCPLOT™ software.

# Index