CAD and the Geodatabase

An ESRI® White Paper • July 2001

CAD and the Geodatabase

An ESRI White Paper

Contents   **Page**

Contents   **Page**

CAD and the Geodatabase

Introduction

The focus of this document is to discuss the methods and issues involved in using computer-aided design (CAD) data as a data source in the ArcGIS™ family of software and other ESRI® geographic information system (GIS) software. This document will define the various data formats involved in CAD/GIS integration and conversion while outlining some common work flow and conversion methods.

CAD files often serve as excellent sources of GIS data sets. CAD files contain some textual and extended attribution and a good amount of vector data that can be used to populate GIS data sets. However, because of the different data models and the different intended uses of the CAD drawings, the process of using CAD data in a GIS or converting CAD data into a GIS data set warrants special attention.

Definitions

It may be useful to briefly describe the data types that will be used in the discussions that follow.

CAD Files

For the scope of this document we shall limit the discussion of CAD files to AutoCAD® drawing files (.dwg), MicroStation® design files (.dgn), and Autodesk's drawing exchange format (DXF). The file extension for a MicroStation file can vary.

In general terms CAD files contain a collection of autonomous geometric objects that are defined by static graphic properties, such as color, line style, and line weight, and are loosely organized by level or layer. A CAD layer does not hold the same organizational strictness that a layer has in a GIS. Although a CAD layer is often used to organize and define default symbology for a system of data in a CAD file, the CAD layer can be used to hold objects from many different data systems, or CAD objects in a single system can be spread out over many CAD layers. In some sense the layers of a CAD file are no different than the graphic properties of color or line style.

In a CAD file, enforcement of data organization is the sole responsibility of the CAD standards implemented by the organization and prudently adhered to by the CAD operator. Sometimes data organization is dictated by a CAD application that creates the data. Different physical real-world systems modeled by CAD are only differentiated by arbitrary agreed-on selection of similar graphic properties such as color or a CAD layer. For example, a water system may be drawn in CAD as a collection of blue lines on a layer called water. The user may infer that the line represents a water line because it is blue and resides on the water layer. This freedom of data creation and graphic description makes the CAD environment very powerful, without a lot of data schema overhead. The use of well-defined data standards is essential in avoiding ambiguity and poor data quality in a CAD file.

The CAD files are collections of drawing entities/elements contained in a single file. All the information necessary to establish the graphic and geometric definitions of all the objects is contained primarily in that one file. (Some external files are used to define various symbology definitions or fonts.) Different CAD software versions have implemented different versions of these files' formats (for example, the drawing formats are different between AutoCAD releases 13, 14, and 2000). It is necessary to make note of which version of an AutoCAD drawing file you are dealing with in certain situations to ensure that the various conversion or viewing tools will work properly with your drawings.

The DXF file has become the de facto intermediate file format for vector data. DXF files are supported by a wide variety of vector-based geometric editing software. The DXF file is generally an ASCII file. A binary version of the DXF file also exists, but it is less popular. The DXF file contains various geometric and graphic property descriptions that are tailored to the AutoCAD environment. However, the vector and symbology data sets contained in the file are general enough to be imported and exported in and out of a variety of graphic-based software environments such as MicroStation.

CAD files have the ability to store attribution in addition to the static graphic properties and are used as a form of coded attribution. There are a variety of methods used within AutoCAD and MicroStation to maintain descriptive attributes on CAD objects. Some of these methods are relatively standard, and others can be rather tenuous customized schemas. ESRI software has limited access to these user-customized attribution schemas. ArcView® and ArcMap™ have tools to read the extended attributes stored in MicroStation MSLINKs and tags and AutoCAD block names and block attributes as well as the standard graphic CAD properties. Other less popular extended attribution methods are not supported by ArcView and ArcMap and may include AutoCAD Object data, user-defined x-data, AutoCAD MAP data, ASE-linked external data, and LPN values or some forms of MicroStation DMRS data and schemas involving catalog values and external databases.

ArcSDE™ CAD Client can extract the graphic properties, block name, and MSLINK in addition to MicroStation catalog values but does not record the cell values or block attributes using the standard interface. Any attribution including AutoCAD Object data, block attributes, cell tags, user-defined x-data, AutoCAD MAP data, ASE-linked external data, LPN values, and MicroStation DMRS data can be recorded on entities and elements using the ArcSDE CAD Client application programming interface (API).

Coverages

A coverage is a file-based vector data storage format for storing the location, shape, and attributes of geographic features. A coverage usually represents a single theme such as soils, streams, roads, or land use. It is one of the primary vector data storage formats for ArcInfo™. Coverages store geographic features as primary features (such as arcs, nodes, polygons, and label points) and secondary features (such as tics, map extent, links, and annotation). Associated feature attribute tables describe and store attributes of the geographic features.

Coverages use a set of feature classes to represent geographic features. Each feature class stores a set of points, lines (arcs), polygons, or annotation (text). A feature class is a

classification describing the format of geographic features and supporting data in a coverage. Coverage feature classes for representing geographic features include point, arc, node, route system, route, section, polygon, and region. One or more coverage features are used to model geographic features; for example, arcs and nodes can be used to model linear features such as street centerlines. The tic, annotation, link, and boundary feature classes provide supporting data for coverage data management and viewing.

Feature classes can have topology, which determines the relationships between features. In coverages, topology is the spatial relationships between connecting or adjacent features (for example, arcs, nodes, polygons, and points). Topological relationships are built from simple elements into complex elements: points (simplest elements), arcs (sets of connected points), areas (sets of connected arcs), and routes (sets of sections, which are arcs or portions of arcs). Redundant data (coordinates) is eliminated because an arc may represent a linear feature, part of the boundary of an area feature, or both.

To define the features, more than one feature class is often required. For example, line and polygon feature classes both exist in a coverage representing polygon features. Polygon features also have label points, which appear as a separate feature class. Every coverage has a feature class containing tic points, which represent known real-world coordinates.

Feature attributes are stored in a separate INFO table for each feature class in the coverage. Other attributes can be stored in INFO tables or tables in a relational database management system (RDBMS), then joined to the features with a relationship class.

PC ARC/INFO® coverages are like ArcInfo coverages, except that their attributes are stored in dBASE® tables. PC ARC/INFO coverages can be previewed in the catalog, and you can create metadata for them, but other data management operations, such as copy, paste, and delete, are unavailable.

Shapefiles

A shapefile stores nontopological geometry and attribute information for the spatial features in a data set. The geometry for a feature is stored as a shape comprising a set of vector coordinates.

Because shapefiles do not have the processing overhead of a topological data structure, they have advantages over other data sources such as faster drawing speed and editability. Shapefiles handle single features that overlap or that are noncontiguous. They also typically require less disk space and are easier to read and write.

Shapefiles can support point, line, and area features. Area features are represented as closed loop, double-digitized polygons. Attributes are held in a dBASE format file. Each attribute record has a one-to-one relationship with the associated shape record.

A shapefile consists of a main file, an index file, and a dBASE table. The main file is a direct access, variable record length file in which each record describes a shape with a list of its vertices. In the index file, each record contains the offset of the corresponding main file record from the beginning of the main file. The dBASE table contains feature attributes with one record per feature. The one-to-one relationship between geometry and

attributes is based on record number. Attribute records in the dBASE file must be in the same order as records in the main file.

ArcSDE Simple Feature Layers

ArcSDE organizes features in feature classes. A feature class is a collection of one or more features of one geometric type and is synonymous with the older term "layer," used with Spatial Database Engine™ (SDE®) 3.x. A feature is a geometric representation of a spatial object (e.g., a road), defined as a sequence of one or more *x,y* coordinates and the attributes for that geometry. Features are stored so that one row in a table equals one feature.

ArcSDE implements a feature class as one or more tables, depending on the database management system (DBMS) and column type used for storing the geometry.

ArcSDE does not change existing DBMSs or affect current applications. It simply adds a spatial column to tables and provides tools (an API) for a client application, such as ArcMap, to manage and access the geometry data referenced by that column.

When you add a spatial column to a table (sometimes referred to as a business table), you spatially enable it. ArcSDE software manages spatially enabled tables by storing information such as the name of a feature class; its owner; *x,y* extent; type of geometry allowed in the layer; and many other pieces of information in ArcSDE metatables. These ArcSDE metatables are stored in the host DBMS and are created when ArcSDE is installed. They are owned, managed, and populated exclusively by ArcSDE.

All data is stored in standard DBMS tables using data types available for the host database. ArcSDE does not supersede or replace any existing DBMS functionality. Instead, it uses and complements the base DBMS capability by adding a spatial component to the database.

In a relational database management system model, data is stored in tables consisting of rows and columns. The cell defined by the intersection of a row and column is called a field; the data contained in the field is a value. A row represents a particular occurrence, or instance, of a feature, while the columns contain the attributes of the feature such as owner name for a parcel. Attributes can have many types such as dates, text strings, or numbers. A geometric shape of a feature is another type of value, stored in a column that defines an abstract geometric data type.

Every geometry type in ArcSDE has a set of strict verification rules that determine whether a feature is geometrically correct before it is stored. Verification rules for each shape type are described in the ArcSDE Developer Help (in the documentation folder of your ArcSDE installation).

These simple features that ArcSDE stores in a DBMS are the building blocks on which the ArcGIS geodatabase is built.

Geodatabases

A geodatabase is a repository of spatial data inside a DBMS. It contains vector data, raster data, tables, and other GIS objects. The term geodatabase is short for geographic database, a relational database containing geographic information.

Geodatabases come in many sizes and have any number of users. They can scale from small, single user databases to large work group and enterprise geodatabases used simultaneously by many users. ArcSDE lets you implement a multiuser geodatabase of any size in the DBMS of your choice—Oracle®, Microsoft® SQL Server™, IBM® DB2®, and Informix®.

The geodatabase model supports an object-relational vector data model. In this model, entities are represented as objects with properties, behavior, and relationships. Support for a variety of different geographic object types is built into the system.

These object types include simple objects, geographic features (objects with location), networks and topology (objects having spatial relationships with other features), annotation features, and other more specialized feature types.

The geodatabase model lets you define relationships between objects, together with rules for maintaining their referential integrity.

The simplest geodatabase is one that contains a number of independent feature layers. Each feature layer typically contains points, lines, polygons, or annotation. This is analogous to how SDE layers are implemented in SDE 3 and to ArcView shapefiles. In these data models, each layer contains a single, stand-alone feature class.

A geodatabase may be comprised of one or more feature classes. A feature class is a homogeneous collection of geometry. These feature classes are the underlying geometry structures of a geodatabase. The extended rules and behaviors are stored as additional tables that are also managed by ArcSDE. ArcSDE manages both simple feature layers as well as ArcGIS geodatabases. The difference between the two is the additional extended rules and behaviors being managed with geodatabases that are not included with ArcSDE simple feature layers.

Within the geodatabase there is a data collection container called a feature data set, which itself is made up of one or more feature classes. A geodatabase is a container of one or more feature classes and/or one or more feature data sets that are themselves made of one or more feature classes.

A geodatabase feature data set is a collection of related feature classes and is used to define a system of objects with the same or different geometric types such as the pipes, streams, manholes, and ponds of a drainage system. One might have a geodatabase to contain the drainage system and another geodatabase to maintain parcel data. Contained within the water system geodatabase may be a feature data set that contains a pipe feature class, a streams feature class, and a manholes feature class as well as additional feature classes of floodplains that are not directly part of the drainage system feature data set. The parcel geodatabase, on the other hand, may simply contain a single feature class of areas that describe the parcel boundaries.

Integration   ESRI provides different strategies for working with and sharing CAD data. One form of CAD/GIS integration is to simply use CAD data as a GIS data set. Different from data

conversion, this form of integration seeks to read CAD drawings directly from their native files and treat them as a valid GIS data source.

ArcView GIS 3.x CAD Reader Extension

ArcView GIS contains tools to read CAD files as valid GIS data sources. These tools directly read a CAD file off of the disk. When these tools view data, a CAD file is treated as a GIS data source inside these GIS software environments. In ArcView the CAD file is treated like a shapefile, using the CAD object's geometry as the shape and its graphic properties as a descriptive database table. The CAD reader extension supports the following CAD graphic properties and extended CAD attribution: color, line style, level/layer, block name/cell name, text value, file position/handle, MSLINK, cell or block attributes, text style thickness, line width, elevation, and entity type.

A single drawing can be treated as one or more themes in ArcView GIS. Because ArcView GIS works only with a theme that has homogeneous geometric types, it is necessary to divide a single CAD file into one of four valid geometric types to be viewed as themes in ArcView GIS. Each CAD drawing can be opened multiple times in ArcView GIS as a source of one or more themes of either lines, points, areas, or annotation.

It may be useful to make several references to a single CAD file to group various levels or layers of CAD data into separate ArcView GIS themes. This is common practice to manipulate a selected group of objects independently. For example, a CAD file may contain linear objects that define roads, water lines, sewers, and property boundaries. It is advantageous in the ArcView GIS environment to control the symbology and visibility of each of these linear systems separately. This is similar to the way a CAD operator would turn on an off the visibility of CAD layers inside of AutoCAD or MicroStation.

AutoCAD blocks and MicroStation cells can be viewed as points or as their component geometry of points, lines, and/or areas. The AutoCAD block attributes and MicroStation cell tag values are automatically included in the virtual table for blocks or cells when viewed as points or as their component geometries.

ArcGIS CAD Feature Classes and CAD Drawing Layers

ArcMap also has the ability to directly view CAD files. ArcGIS has two different ways to treat this information. The first way is as CAD Feature Data Objects, similar to the way they are handled in ArcView GIS 3.x software's CAD Reader Extension, or one could choose to view the CAD drawing as a single layer. With the ArcGIS Drawing Layer you can control layer visibility and identify objects. A virtual attribute table is created for the feature data objects that include the CAD graphic properties as well as block/cell attributes, unique identifiers such as file position and handle, and MicroStation MSLINKS.

AutoCAD blocks and MicroStation cells can be viewed as points or as their component geometry of points, lines, and/or areas. The AutoCAD blocks attributes and MicroStation cell tag values are automatically included in the virtual table for blocks or cells when viewed as points or as their component geometries.

When a CAD drawing is viewed as Feature Data Objects, ArcMap creates a CAD feature class. One can use this CAD feature class for query and analysis and to modify the symbology of the features using the standard symbology editing tools of ArcGIS.

When a CAD drawing is opened in ArcGIS as a CAD drawing layer, the entire CAD drawing is treated as one mapping layer, and its symbology is static. A virtual table is also created for the CAD objects when viewed as a drawing layer. Individual CAD objects in the drawing layer can be identified and the graphic CAD properties displayed, but the drawing layer cannot be used for ArcGIS analysis or query.

In the catalog tree you will also find a CAD drawing item, which represents all features in all layers of the drawing. In Geography view, the CAD drawing item's features appear with the symbology defined in the drawing itself.

In ArcMap, geographic data is represented on a map as a layer. When a layer is created from the CAD drawing item, you can choose which of the drawing's layers it represents. For example, you might want to see only the streets, traffic lights, and street names on your map.

Several CAD formats can be directly accessed in ArcGIS. Supported CAD drawing files include AutoCAD drawing files (.dwg) up to AutoCAD 2000; all ASCII, binary, and partial drawing interchange files (.dxf) that comply with DXF standards; and MicroStation design files (.dgn) up to version 7.

The same CAD viewing functionality of ArcMap is also available in ArcGIS software's three-dimensional ArcScene™ environment to view three-dimensional CAD data along with three-dimensional GIS data.

ArcSDE CAD Client

ArcSDE CAD Client is a spatial data client of ArcSDE. ArcSDE CAD Client allows a user in AutoCAD or MicroStation to both store and retrieve and edit CAD data and ArcSDE simple features managed by ArcSDE. CAD Client can simultaneously access data from many ArcSDE servers in various DBMSs. One single session of CAD Client could be accessing data from ArcInfo coverages managed by ArcSDE for Coverages, an Oracle Spatial database managed by ArcSDE for Oracle, and SQL Server databases managed by ArcSDE for SQL Server all at the same time. CAD Client stores and retrieves data from ArcSDE simple feature layers. Furthermore, CAD Client can accesses data from the simple feature layers of an ArcGIS geodatabase in ArcSDE.

Conversely, when CAD data is stored into a DBMS, GIS users have access to the geometry of the CAD data from within the other ESRI ArcSDE software clients without conversion. When CAD Client stores CAD data into an ArcSDE-managed DBMS, each CAD object has its geometry interpreted and stored as a valid ArcSDE geometric feature. Sophisticated parametric CAD geometries, such as ellipses, circles, and blocks/cells, are interpreted so as to allow access to other ArcSDE client applications.

CAD Client can store a binary copy of the CAD object. This binary copy of the CAD object contains everything about the CAD object including its exact parametric geometry, graphic properties, x-data, object data, block definitions and block attributes, tags, cell

descriptions, and other custom user data. A CAD Client user also has access to all other spatial data managed by ArcSDE regardless of its source. Likewise, GIS users have direct access to the geometry of any CAD data that is stored by a CAD Client user. The CAD object stored in ArcSDE is only accessible through ArcSDE CAD Client and the originating host CAD application. CAD objects are linked directly to their associated ArcSDE features.

One highly productive by-product of using CAD Client is that all of the stored data becomes available to the entire GIS enterprise including any Internet or Intranet user via ArcIMS® software. ArcIMS is also an ArcSDE software client. Access can be given from an Internet browser to a central repository of spatial data made up of CAD and/or GIS data by deploying ArcIMS on an ArcSDE data server.

CAD Client is provided with read-only access to ArcSDE for Coverages and geodatabases.

Conversion Tools

Uses of CAD and GIS data are often different; however, the need to share and reuse data across these different environments is important. Reorganization and manipulation of CAD data must often be performed to make the data suitable for GIS applications. There are a variety of different conversion tools provided by ESRI to help users of both GIS and CAD make the best use of the available data.

ArcView GIS 3.2: Save as Shapefile

The CAD Reader extension of ArcView GIS has already been mentioned in the previous sections as an integration tool. It can also be seen as a conversion tool. In ArcView GIS 3.2, one can select a group of spatial features from a CAD file using attribute and/or spatial selection criteria. This selected set of objects can be exported out to a shapefile. Using this feature, ArcView GIS has converted the CAD objects from CAD file into an ArcView GIS shapefile format.

ArcGIS ArcCatalog: Simple Data Loader

The simple data loader is the preferred method of adding data to an existing feature class. Once in ArcCatalog™ simply select a feature class and right-click to access the Load Data option to invoke the simple data loader. The simple data loader is a wizard that guides you through the process of loading data. Simply select a CAD drawing as your data source, and the wizard will allow you to map CAD graphic property values to existing geodatabase fields or omit certain fields. The simple data loader provides a query capability to filter out unwanted entities. Finally all of the selected information is appended to the target geodatabase feature class. The simple data loader wizard in ArcCatalog has the advantage over other conversion methods that will be presented in that it allows data to be appended to existing data, and it is the only tool that provides mapping of table fields. However, this tool requires that a geodatabase feature class already exists before you can add data to it.

ArcGIS: ArcMap— Export Data

Using the Export Data tool from ArcMap, a selected set of CAD features can be exported to an empty geodatabase feature class or a shapefile. To access the Export Data function, right-click on the data source in the table of contents and expand the Data option and then pick the Export Data selection from that submenu. The default file type to save the data will be a shapefile; however, if you select the browse button there is a Save as type selection box from which you can choose a geodatabase feature class format.

CAD text conversion using ArcMap is handled a little differently. Because the text of a CAD drawing is viewed in ArcMap as a hybrid coverage annotation type, one must use the convert coverage annotation function to get CAD text into the geodatabase as geodatabase annotation. This tool is included with ArcMap but does not appear on the standard ArcMap toolbar and can be added to the interface by customizing the Labels menu.

ArcGIS: ArcMap— Cut and Paste

Yet another way to load data into a geodatabase feature class is to use the cut and paste feature of an ArcMap editing session. When editing a geodatabase feature class you can select data from a CAD feature source in ArcMap and paste it onto a geodatabase feature class of the same geometric type. This method and the simple data loader method in ArcCatalog both allow you to append data to an existing feature class.

ArcGIS ArcToolbox: CAD to Geodatabase Conversion Tool

ArcGIS provides a CAD to Geodatabase conversion tool that converts CAD data into new geodatabase feature classes. This conversion tool is available from ArcCatalog and ArcToolbox™. The ArcToolbox conversion tool allows you to define a query to select the CAD objects you want to convert and store them into a new geodatabase feature class. Unlike the methods used in ArcMap to save data to an existing empty geodatabase feature class, the ArcToolbox conversion tool will create the new geodatabase feature classes in an existing geodatabase as part of the process of conversion. One can perform multiple queries against a single CAD file to pull out various line, point, and area data systems from within a CAD drawing to create multiple geodatabase feature classes.

When using the CAD to Geodatabase Translator function, CAD text is converted into a point feature class.

Parametric curve information is retained on the CAD features as it is converted into the geodatabase. CAD geometric properties and block and cell attributes can also be retained during the conversion process.

Converting multiple drawings into a single geodatabase feature class requires combining the resulting geodatabase feature classes after each of the conversions have been processed. One could then use the Geoprocessing Wizard in ArcMap to combine the separate feature classes created from each drawing into a single geodatabase feature class. One could also merge these feature classes into more sophisticated geodatabases where domain constraints and editing rules have been defined. Once the data is in a geodatabase format, validation can be performed on the data to enforce these rules and constraints or establish new rules, relationships, and constraints.

ArcSDE CAD Client

CAD Client was mentioned above as an integration tool; however, it can also be used for conversion. During the process of storing CAD data into an ArcSDE-managed DBMS, CAD Client will interpret the geometry of every CAD object and generate an ArcSDE geometric feature for it. This in itself is a form of conversion. Other GIS software clients of ArcSDE can view these ArcSDE features without the need for further conversion.

It should be noted that once an ArcSDE simple feature layer is registered as a geodatabase, it becomes read-only to CAD Client. Generally this is acceptable and even

desirable since one usually stores data in a geodatabase for the purpose of extending the database to include rule-based editing and/or extended object relationship. These are GIS functions that can only be managed within the ArcGIS environment. The correct work flow in such situations is truly data conversion rather than data sharing or integration. In such situations CAD data is converted solely to populate a GIS database designed to support GIS applications and geodatabase editing.

Moving CAD Data into the Geodatabase

Using the previous discussion and information as a foundation, this section will outline the work flow and software tools used to successfully move a variety of CAD data into a typical geodatabase. Included in this discussion will be an outline of some of the processes and benefits of undertaking the effort of conversion.

Benefits of Conversion

Although the data models of CAD and an ArcGIS geodatabase can be very different, they also share many things in common. It can be very useful to use the geometry of a CAD file, for example, to populate the geometry of a geodatabase. Various forms of CAD attribution and graphic properties can also be valuable sources of information. Not all of the attribution in a CAD file is readily available to the geodatabase, and for the most part, the decision to undergo the conversion process is usually with the understanding that even a partial conversion is worthwhile.

CAD to Geodatabase Conversion Scenario

Description of CAD Data Sources

This scenario assumes that there is a mix of both AutoCAD and MicroStation CAD files. There are 15 AutoCAD drawings that contain as-built drawings of the water system. Each AutoCAD drawing has a unique local coordinate system, but each drawing can be made a cell of a grid.

Each pipe is drawn as either a line or lightweight polyline. Each pipe is colored based on the material type—white for PVC, red for ductile iron, gray for concrete, and blue for asbestos. The pipes reside on one of five layers: main, zone1, zone2, or zone3. The pipe diameter is coded in the graphic property called thickness. There is some extended attribute information contained on each pipe that describes each pipe. This extended information was stored using AutoCAD Map's object data format. There is a concern that many of the attribute values for the pipes are missing, outdated, or have errors.

AutoCAD blocks are used to represent the valves, fittings, and pumps. Each type of fitting, valve, or pump has a different block name and different block symbol. Each block contains several textual attributes that describe the maintenance or descriptions of the fitting, pump, or valve. There is some concern that the wrong fitting blocks were used in some cases, but it is unclear which of the hundreds of fittings may be drawn incorrectly.

The pump facilities are closed polylines on a drawing layer called pump_facilities. Each is intended to be a closed area, but not all of the facilities were drawn with the closed polyline representation. Not all pump houses are closed polylines; some are a series of line segments, and some are drawn on the wrong AutoCAD layer. It is not known specifically which of the 23 pump facilities are drawn incorrectly.

There are street centerlines available as lines in a single DXF file. The centerlines were created using the same State Plane coordinate system that will be used by the target geodatabase. The street names for the streets appear as text near the midpoint of various street centerline line segments. There is no distinction between the centerline values other than the proximity of some of the line segments to text elements.
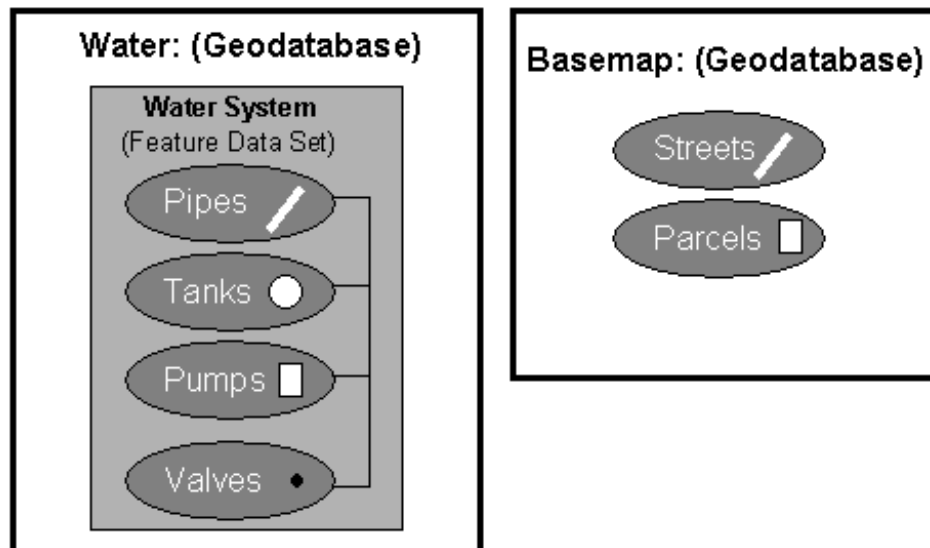
There is one MicroStation design file that was maintained by MGE for the parcels. The MicroStation design file contains closed line strings that represent each parcel. There are MSLINK values and catalog values on these parcel elements that link each parcel element in the design file to an external Open Database Communication (ODBC) database table of parcel attributes.

Describing the Target Geodatabases

There are two target geodatabases used in this scenario, a water system and a basemap. The water geodatabase has a water system feature data set, which contains a feature class of pipes, a feature class of water valves, a feature class of storage tanks, and a feature class of pump facilities. There is an existing geodatabase schema, which includes editing rules, relationships for complex objects, and various domain constraints on columns in various tables. It is the goal to inherit all of these rules, relationships, and domain constraints when you convert the CAD drawings to the geodatabase. The basemap geodatabase contains a parcel feature class and a street centerlines feature class.

The pipe feature class has a linear feature type and has four pipe subclasses defined for ductile iron, asbestos, concrete, and PVC pipes. There are domain constraints on the diameter of each of these pipes that limit the possible values to two-inch, four-inch, eight-inch, 12-inch, and 48-inch diameters. Editing relationships have been defined on the pipes to require that the appropriately sized expansion and reduction fittings are included that join two pipes together. Likewise there are relationships defined between the tank, pipe, valve, and pump facilities to ensure that each is properly connected to the network.
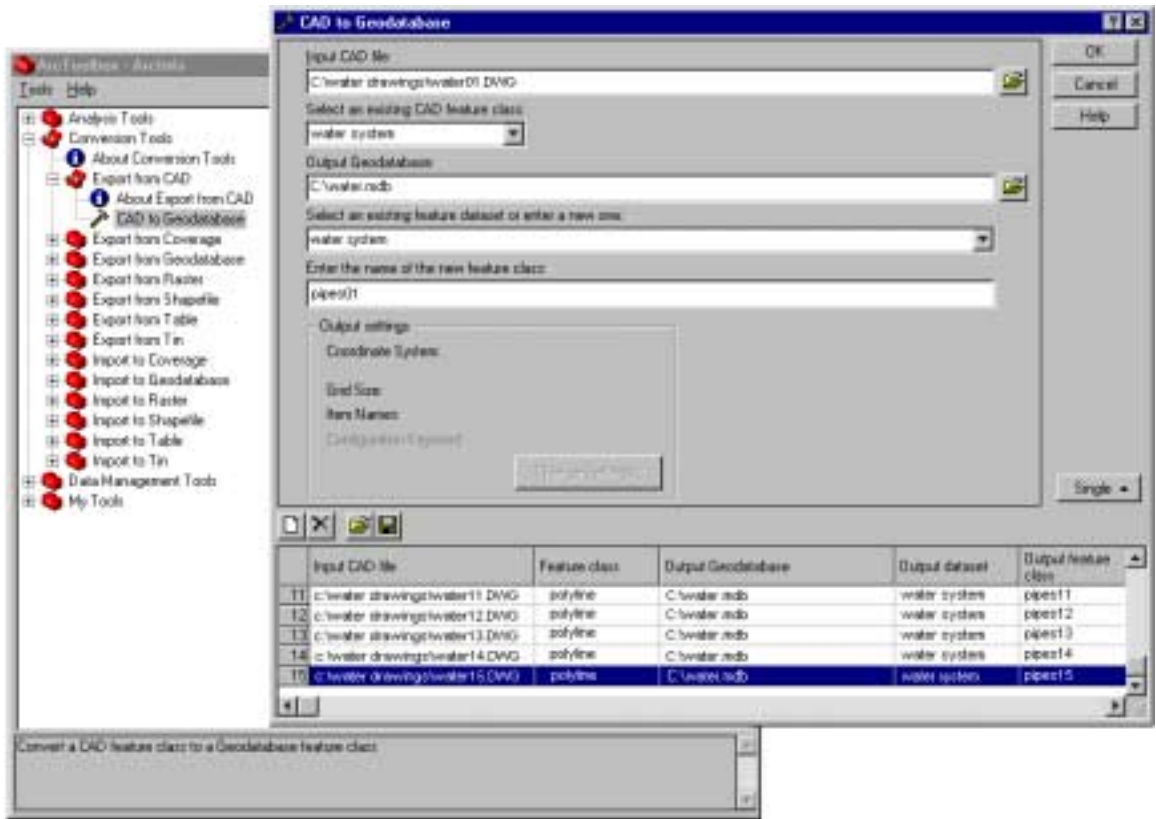
The simple geodatabase of parcels and street centerlines contains a feature class of areas to model the parcels and a feature class of lines that define the street centerlines.

J-8687

Using ArcToolbox CAD to Geodatabase Conversion Tool

The first step will be to extract various entities out of the 15 AutoCAD drawings to define the water system. To accomplish this conversion you will use the ArcToolbox CAD to geodatabase conversion tool. The CAD to Geodatabase translator tool in ArcToolbox is invoked from the Conversion Tools->Export from CAD-> CAD to Geodatabase option. By selecting the Batch button at the bottom of the resulting dialog box, each drawing can be added to perform the 15 drawing conversion all at once. Once the drawing file has been selected and the target geodatabase information has been entered, you can select the change setting buttons to access the Spatial Reference, Grid Size, Item Names, Query, and Geometry settings. Of special interest is the Query tab, which allows you to specify a query against the drawing to select just those entities that qualify as water pipes. To get just the water pipes you will use the knowledge of the drawings and select all the entities where the entity type is either line or polylines and resides on a layer called main, zone1, zone2, or zone3. The result of the conversion will be 15 separate new feature classes of just water lines extracted from the CAD files in a new or existing feature data set of an existing geodatabase.

J-8687

This process will be repeated for each series of feature classes in the water system. The process is repeated for the storage tanks, water valves, and pump facilities for the 15 drawings, using knowledge about the drawings and various queries to select just the desired entities from the CAD drawings to build the appropriate feature classes within the water system feature data set of the water geodatabase.

Not readily accessible on the pipe data will be the AutoCAD Map attributes associated with the pipes. Although some of the data was missing or inaccurate, it may still be worth attempting to convert it. One of the attributes obtainable from the drawings is a value called "HANDLE" that when combined with a drawing name is a unique value for an entity. If the handle field can be added to the AutoCAD Map object data before the conversion, the external database tables referenced by AutoCAD Map can be relinked after the conversion using the key field of handle.

The characteristics that differentiated the pipes in the CAD drawing are coded within the graphic properties of the CAD data. It is useful to retain this information and then use it to populate table attributes once they are successfully moved into the geodatabase. You can move various coded values into more descriptive headings using the attribute matching function of ArcGIS. For example, in the pipes of our water system ArcToolbox has created a column called layer into which the zone information has been extracted from the CAD graphic property called layer. You can use the column-mapping feature of

the schema wizard to push the values from the column called layer into a column you established in the existing geodatabase schema called zone.

The pipe diameter is coded into the graphic property called thickness, and the material type is coded by color. All of these CAD properties *do* become tabular attributes as part of the conversion; however, it would be more useful to take the additional step to decode the attributes and enumerate them more clearly under a column heading such as material rather than color and diameter rather than the ubiquitous thickness label. This step may be undertaken when using the schema wizard or later using the database manipulation tools of ArcGIS to rename columns or merge and join lookup tables of attributes to resolve the coded values.

To make the best use of the schema that has already been defined in the target database, you will want to be careful to name the resultant feature classes with the same names listed in the target database schema. Before you end up with these four feature classes you must deal with each of the 15 sets of resultant feature classes. Since the ArcToolbox CAD-to-Geodatabase conversion tool was used, the result will be 15 line feature classes representing the pipes of the water system, 15 point feature classes representing valves, 15 more point feature classes representing tanks, and 15 area feature classes representing the pump facilities.
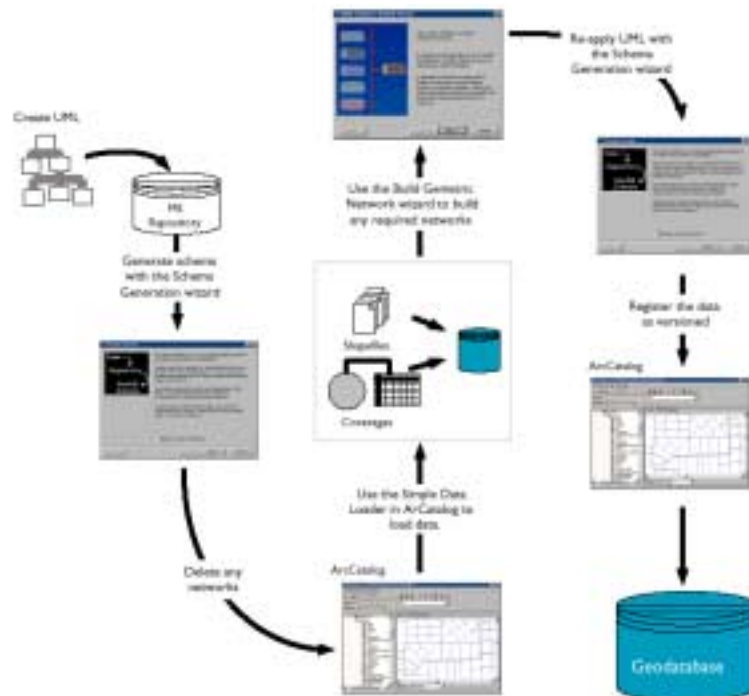
Merge Feature Classes

Each of the conversion processes described above has created separate feature classes in their respective geodatabases and feature data sets. Our next step will be to combine all of the similar features from the 15 separate AutoCAD drawings to become part of a single feature class. Furthermore you would like to apply a complex existing geodatabase schema for each geodatabase. Applying the geodatabase schema rules, relationships, and domain constraints to the newly converted CAD data provides a powerful means of data quality control. Once the geodatabase schema has been applied, tools in ArcGIS can be used to validate the schema to identify and resolve errors as well as inherit all of the intelligence to support future or existing GIS applications.
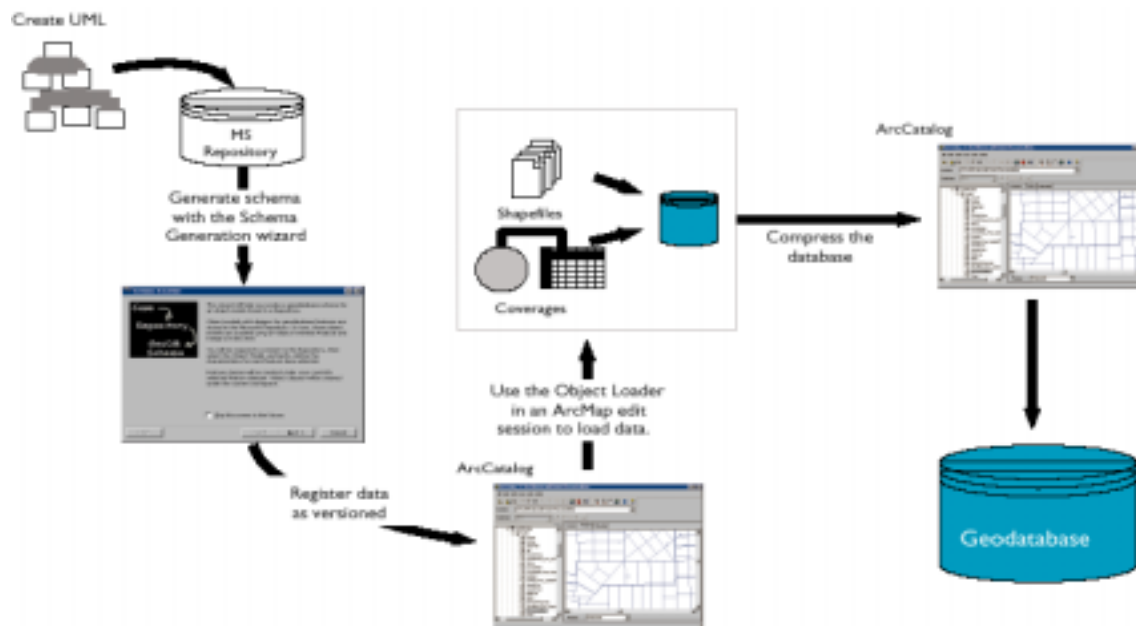
You have choices as to how the 15 feature classes of each data type are assembled into a single feature class of their respective system features. Furthermore, once that is accomplished you have some choices as to how you choose to apply the existing schema to inherit the rules, relationships, and domain constraints developed for our water system feature data set. First you could assemble the 15 pipe feature classes into a single feature class of pipes using the Geoprocessing Wizard in ArcMap to merge all of these similar feature classes together. Or, the 15 feature classes can be used to populate an empty geodatabase where the schema has already been defined using the ArcMap Load Objects function. During the ArcMap Load Objects operation all the effects of the underlying schema will be applied for the incoming data.

At this point you have a choice of two strategies for finishing the loading of the 15 individual sets of feature classes into a single feature class of the appropriate type. The choice is due to performance considerations and hinges on balancing speed of loading data against maintaining certain custom feature behavior.

One strategy is to delete the geometric network from your feature data set and load your data with the ArcCatalog Simple Data Loader, then rebuild the network and reapply the database schema. Another strategy is to keep the network and use the ArcMap Object Loader to load your data. Each has advantages and disadvantages.

The first strategy allows you to load features much more rapidly, though it has the disadvantage of requiring you to reapply your database schema. This will not work for feature classes that have custom objects with custom object creation behavior because the object creation behavior will not be triggered. For example, you might have a custom Water Structure: Pump Facility object with a class extension to create a Network Structure: Pump Facility object and relate it to the Water Structure in a composite relationship whenever a Water Structure feature is created. If you were to load the Water Structure features with strategy 1 as simple features, this behavior would not be triggered, and the Network Structures would not be created or correctly related to the Water Structure features. You would need additional custom scripting tools to create the Network Structures and implement the relationships.

The second strategy is to use the Object Loader within an ArcMap edit session. This strategy will work for these feature classes, but it requires the database to have a geometric network and be registered as versioned before the objects are loaded. This process can be very slow (as much as 7–9 seconds per feature loaded). For more information about the performance trade-offs involved in loading and using network feature data, see the ESRI white paper *Multiuser Geographic Information Systems with ArcInfo 8,* which provides guidelines for designing and implementing multiuser GIS databases in ArcInfo 8. This paper and the latest database performance enhancement tips can be found at ArcOnline (http://arconline.esri.com).

Whichever method you choose, you should be careful when naming columns and feature class names to ensure that all of the rules and relationships can be identified and resolved correctly by the information stored in the schema repository file being used by the schema wizard.

Using ArcMap Export to Geodatabase

For the sake of diversity you will use the ArcMap Export to Geodatabase tool as a way to save the street centerline data into a geodatabase feature class of street centerlines. This process could just as easily have been performed using the ArcToolbox CAD to geodatabase converter like the water system was, but since there is only one drawing it is just as easy to use ArcMap. Using ArcMap software's ability to access a DXF file as feature data objects, the lines from the DXF file can be displayed as a set of linear features. By limiting which DXF file layers are displayed from the drawing, it is easy to isolate just the actual street centerlines. ArcMap can then export the data to a geodatabase feature class using the Data->Export Data tool accessible by right-clicking on this CAD data source in the ArcMap table of contents.

J-8687

Once the CAD data is converted into the geodatabase, you can add various editing rules or domain constraints, but there are none required for this simple street centerlines feature class example.

One additional step that you may want to make would be to also create a coverage annotation feature class from the text. This annotation can be associated with various street line features as annotation or perhaps as a street name attribute in the feature classes database table. One could use a spatial join in ArcMap to support the future gathering of street names for the street centerline database as a short cut to entering street names by hand or some other source. This would be considered an optional step for conversion. This could be an example where some information is not directly usable or easy to access from the CAD files. The text for the street names in the CAD file are sparse and not included near every line segment of the streets, and not every street is labeled even once. Perhaps there is even a better source of street centerline data where the names are already included in a GIS database.

The same procedure used for the street centerlines can be applied to the parcels features from the parcels.DGN MicroStation design file to create the parcels area feature class in our basemap geodatabase using ArcMap.

Using ArcMap to Reestablish MGE Database Links

The parcels information contained in the parcels.DGN MicroStation design file was part of an MGE data set. Included on each of the parcel elements was a link to an external ODBC database. Although ArcMap does not have direct access to the attributes stored externally in the database, there is enough information stored on the elements to re-create the links to the external parcel database tables. This link information is presented in the form of the MSLINK–ODBC and MSCTLG–ODBC fields accessed by ArcMap. These two values combine to identify a reference to a particular record in an external database table. Database location and connection information of the tables must be known to reconnect the database in ArcMap. MSLINK and MSCTLG values are grouped in pairs. The MSCTLG value contains a number that references the record in a MGE catalog table. This MGE catalog table contains information as to which table a particular MSLINK record number references. Simply stated, the MSCTLG value determines in which table to look, and the MSLINK tells in which record in that table to look.

Validation of Converted CAD Data

Now that all of the data has been successfully converted into its respective feature classes and feature data sets, you can use the existing rules defined in the schema of our water system feature data set and its feature classes to validate the newly added data. The validation rules have now been inherited by the incoming CAD data. This very powerful aspect of conversion helps improve data quality and pinpoint possible errors in the incoming data. Now that the data is fully a geodatabase in every respect, the ArcGIS validation tools can be used as with any other geodatabase.

Avoiding the Effects of Large Appends on Versioned Geodatabases

Another important consideration of loading CAD data into a geodatabase is the impact those changes will have on existing versioned geodatabases. Volumes of appended data can cause unnecessary strain on the systems that are monitoring changes. When loading data, especially new data, these changes to a versioned database need not be tracked like other kinds of edits and may cause unnecessary performance problems. To avoid this,

there are two suggested methods of limiting the effects of data loading on the versioned geodatabase.

The first method would be to reconcile all outstanding versions and delete all but the current version. From this state the data can be appended to the geodatabase and its appropriate feature classes. After a process of quality control (QC) and any validation, one can compress the version and continue on as normal.

The second method is more efficient and eliminates the need to compress the version. This second method involves reconciling all outstanding versions and then unregistering the geodatabase as versioned. After the appending process and QC and validation are completed, the geodatabases can be reregistered and brought back online for versioned editing.

One might be quick to realize that either one of these methods may prove to be difficult or even impossible for organizations that rely heavily on distributed editing and multiple ongoing versions. When possible however, these steps can be taken to reduce the impact that data loading has on versioned geodatabases that often occurs when CAD data is periodically appended to existing versioned geodatabases.

Summary of CAD Conversion Scenario

In this example of conversion you have used various methods that make use of both the geometry and attributes of objects stored in various CAD files to populate existing geodatabases. Depending on the desired result, it may be necessary to take several steps in the conversion process.

Making use of the existing CAD data through conversion is worth the effort when faced with the alternatives of re-creating the data. Another very important aspect of this conversion process is the ability to inherit the rules and constraints of existing geodatabases. This data quality validation is a valuable tool to the CAD user and is considered one of the selling points of a GIS system for data management.

Another form of conversion, not discussed here, would be methods that create shapefiles or coverages from CAD data that could then be converted into geodatabases. Sometimes these methods are equally valid but, for the scope of this white paper, were omitted since they would actually be shapefile-to-geodatabase or coverage-to-geodatabase conversions.

Summary

ESRI provides a variety of conversion and integration tools to assist CAD users who wish to perform GIS functions as well as provides tools for GIS users who need to share data between these similar, but often very different, technologies. ESRI will continue to address the needs of CAD and GIS users to share information in the GIS and CAD environments by creating and improving its software applications.