Data structure for the Arizona Geological Survey Geologic Information System- Basic Geologic Map Data
v. 1.0

by

Stephen M. Richard and Tim R. Orr

Arizona Geological Survey
Open-File Report 01-09
52 pages, one plate

December, 2001

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

INTRODUCTION AND PURPOSE

Geoscience data are used for land-management decision-making, for engineering design, in the search for mineral resources, and for scientific research. Traditionally, geologic information has been stored and disseminated using geologic maps and written reports [Bernknopf et al., 1993]. Because of the complexity of the earth, much of the information included in a geologic map is buried in several layers of abstraction. Specific applied use of geologic data typically requires preparation of a derivative map designed to show a particular aspect of the geologic data. Such maps might be designed to show rock lithology without respect to age, the orientation of bedding or foliation in layered rocks, the acid buffering capacity of the rocks, or to show rocks of a particular age. Production of such derivative maps designed for a specific purpose commonly requires a geologically sophisticated analysis of the original map, as well as cartographic design and drafting of the derivative map.

Computer-based geographic information systems allow the manipulation and analysis of much larger and more sophisticated geographic data sets than was possible using paper maps and physical overlays. These systems provide tools to manipulate and integrate geologic data with other geographic data to a greater extent than ever before possible. A well designed data-rich information system could automate much of the process of producing derivative maps designed for specific applications. This would free the data user to explore the data in more ways, and experiment with different representations of the data. Providers of geoscience data, like the Arizona Geological Survey, must redesign their information delivery systems to facilitate the integration of their geologic data resources into these automated systems in order to maximize its usefulness.

To this end, the Arizona Geological Survey is developing a computer-based geologic information system designed to meet the needs of mineral exploration geologists, researchers in search of detailed technical information, land managers or planners requiring information pertinent to regulatory, planning, and development functions, and curiosity-driven users from the general public. Many of these users may not be expert geologists, but still need to be able to query the system to obtain information. The underlying data model must be flexible enough to encompass a wide range of earth science information, storing it in such a fashion that it does not become obsolete with advances in geologic science.

Based on several years of development and discussion with other database developers [See papers in Soller, 1997; 1998; 1999; 2000], this system has evolved into a structure with a variety of inter-related components, summarized in **Table 1**. This document defines a relational database implementation of the metadata, cartography, geologic map, and geoscience infrastructure parts of the Arizona geologic information system necessary to represent the basic geologic information and cartography recorded on a typical geologic map. This information includes the assignment of map units to regions on the map, the classification of boundaries between the map units as faults or contacts (here referring to depositional or intrusive contacts), the recording of basic point-referenced structural data, and the cartographic representation of these features. Subsequent documents will describe the detailed geoscience description tables (map units, lithology, age dates, stratigraphic relationships, etc.). The implementation is based on Microsoft Access (currently using the Access2000 version; datasets are distributed with Access97 tables for wider accessibility) as the relational database, and ESRI ArcInfo (v.8.0.1) and ArcView (v.3.2) as the geographic data system. and metadata tables.

Table 1. Components of Arizona Geological Survey Information System.

Component Name	Function	Status (12/27/01)
MetaData	Stores basic information about people, projects, organizations, the DataSet catalog, system development metadata	implemented
AZRockUnitLexicon	Stores definition and default descriptions of rock units used for geologic mapping in Arizona. Based on USGS geologic name lexicon and AZGS AzStrat [Trapp, in preparation]	Designed, implemented, not populated
Arizona Geologic Bibliography	Stores bibliography of published literature concerning Arizona Geology [Trapp et al., 1994]	Implemented, populated, in maintainence
Geologic Map	Stores map legend definitions, and map view definitions. This component may have multiple instances specific to particular geologic data sets or projects.	Default map visualization implemented in DI-19.
Cartography Infrastructure	Stores definitions and descriptions of graphical elements used to construct geologic maps, along with default legend for symbolizing standard map units and features.	Implemented, partially populated
Rock Samples	Stores information locating and describing rock samples collected in the field for geochronology, geochemistry, representative lithology, etc.	Implemented, partially populated
Geochronology	Stores detailed analytical information for isotopic age dates.	Implemented, populated based on Reynolds et al. [1987], data structure not finalized
Geochemistry	Stores analytical data for whole rock, trace element, and isotopic analyses of rocks.	Planned
Geoscience Infrastructure	Stores basic geoscience terminology Classification Concepts, definitions and descriptions of standard mineral and lithology terms, and the standard geologic time scale used by AZGS (GSA, DNAG, Palmer, 1983). Data in this component database is not specific to a location, and applies to all geologic data sets.	Designed and implemented, partially populated
Geoscience Descriptions	Set of table templates for description of geologic features specific to individual geologic datasets.	Designed, implemented, not populated

GEOLOGIC MAP DATA

A **geologic map image** is a visual representation of a geologic data set for an area designed to communicate information to a user. The map image is defined by the map area extent, the geologic data (both spatial location and classification) used, the choice of symbols for geologic features, the map projection and scale, a specification of the surface represented by the map, and the cultural and physiographic base map. The path from a geologic data set to a geologic map image requires selecting symbols to represent the distribution of the map units, the location and type of map unit boundaries and faults, and the location and relevant data for point observations (orientation measurements). These symbols are placed on a base map that represents the map area by means of a pro-

jection and some elevation model to represent topography on the mapped surface. The base map provides a visual reference frame to depict the spatial relationships between geologic features, and a means of physically locating the features depicted. Design of the base map is an important aspect of cartography. This definition of a map image makes no distinction between a standard geologic map (map surface = earth surface), a mine-level map (map surface = horizontal plane), or a geologic cross section (map surface = vertical plane along section line).

A **digital geologic data set** represents a geologic data set in a georeferenced form using a set of computer files. A digital geologic data set is defined by:

1. The conceptual model that is the basis for the geologic data set (see Richard, in prep.).
2. A logical data schema that is a mapping of the conceptual model underlying the geologic data set to data structures that can be represented by an automated system (e.g. relational tables, described in this report).
3. A physical implementation schema that defines the organization of data into files, the detailed structure of the files, and the representation of data in the files. The file format dictates the software and hardware systems that are compatible with the data.
4. A projection and map horizon specification that describes how the three-dimensional location of features on the Earth is specified.
5. The data instances contained in the files.
6. A set of definitions that specify the meaning of attributes applied to included data instances.

This report describes the logical and physical implementation of a database system for the representation of geologic features represented on geologic maps. It is assumed that the reader is familiar with the basics of the ESRI coverage data model and the use of ESRI ArcView GIS 3.x and Microsoft Access 97-2000 software.

This database implementation is a second-generation effort, and supercedes the data structure outlined in Richard and Thieme [1997]. The design is an outgrowth from a proposed North American standard data model for geologic maps [Johnson et al., 1998]. In the course of implementing this database, the Johnson et al. [1998] model was found inadequate to allow inclusion of information in existing AZGS databases and for a complete representation of geologic information. Focus then shifted to the Cordlink variant model [Brodaric et al., 1999] as a starting point. Various aspects of this model were also found insufficient or unsatisfying. The logical model presented here was evolved to reduce the number of tables and allow greater flexibility and logical consistency. The final implementation resembles the Johnson et al. [1998] NADM 4.3 model only in very general terms. The model builds on the design philosophy laid out in Richard [1998], the conceptual model described in Richard [1999], and the recent parallel development of an object-oriented data model by Brodaric and others [Brodaric and Hastings, 2001; Brodaric and Gahegan, 2000].

CORE COMPONENTS

The core components of the model are:

1. Classification Concept table(s). At the core of the model is a table or group of tables with similar structure that define terminology. The essential elements of these ClassificationConcept tables are a unique identifier, a name, and a definition/description. The unique identifier follows the global unique identifier scheme described below. The name is a string that allows human identification of the concept (e.g. 'basalt'), and the definition/description is a free text field that defines the term or describes its meaning precisely.

2. Relationship tables. These are tables that link data instances. The meaning of the link is defined by a relationship type attribute. Three sorts of relationship tables are included with different structure and application. Hierarchy Relationship tables define parent-child relationships in hierarchies; these may be taxonomic (IsA) or meronymic (Part-Whole). Simple Relationship tables link data instances, which may have a sequence; typically these link description parts (e.g. image to rock description, age date to rock description, chemical analysis to location). The most complex relationships are Attributed Relationships, which allow an attribute value to be associated with the link, along with a sequencing index, and classification confidence and classification basis attributes.

3. Description tables. These are tables tailored to particular kinds of descriptions. The core model includes tables for structural measurements, text, geochronologic ages, chemical substances, lithologic description, stratigraphic time, spatial objects, and measured quantity. Some of these tables are linked to the ClassificationConcept tables directly through the sharing of a unique identifier, and provide a structured description to characterize the classification concept. Others provide descriptions of 'real world' instances (a particular rock sample, a particular contact, a particular fault….).

4. Map Visualization tables. These are a set of tables used to define map visualizations. This group includes three tables:
   a) Map View Definition table – specifies a title, author, design scale, map extent, symbolization scheme and classification scheme for the map;
   b) Map Legend – relates each symbol used in the map visualization to a classification concept;
   c) Cartographic Object table – defines the symbols used on the map in implementation-independent terms.

Three modes of defining assignment of symbols to spatial objects represented on the map are used. First, in this database, all spatial objects have a default classification attribute and a default cartographic object attribute. This default classification/symbolization corresponds to that assigned by the original author of the map visualization. Second, symbols may be associated with spatial objects through the map legend, (symbol – classification link) and a spatial classification attributed relationship (spatial object – classification link). This approach corresponds to the NADM 4.3 and Cordlink Variant approach. Finally, spatial objects may be linked to symbols through an attributed relationship link whose type is the identifier for the map view definition. This final approach corresponds most closely to how map visualizations are actually generated from spatial data. The relationship attribute is the rotation to apply to structure measurement symbols, or, in the case of purely cartographic annotation symbols, the text string to display.

Identification Scheme

Unique identification of data instances in an internationally distributed data warehouse is achieved by partitioning responsibility for maintenance of unique identifiers. The Arizona geological Survey uses a 3-component composite key, consisting of 3 long (4 byte) integers. At the top level, each organization providing data to the system must be assigned a NameSpace by the overall system manager. Note that a NameSpace is a ClassificationConcept. The name string and an integer identifier for the NameSpace must be globally unique. Within each NameSpace, every data file must have a unique integer identifier, and should have a unique name string. The system manager for the NameSpace must assign a unique identifier number to each data table, geographic data set (coverage, shapefile, etc.), image, text file, etc. that will be used by the system. Information about each data file (called a DataSet here) is stored in a central DataSet table maintained within each NameSpace. This information must include a physical address (url) for each DataSet so that it can

be located automatically when accessed. Within each DataSet, every data instance has a unique integer identifier number. The field containing this identifier is generally named with a string in the form 'DataSetName' & "ID". In summary, the unique, global identifier for any data instance is a tuple consisting of 3 integers: {NameSpaceID, DataSetID, ObjectID}. Because this system has not been adopted outside the Arizona Geological Survey at present, the NameSpaceID is not explicitly included in tables here. Because some database software cannot joint on multiple fields, implementation considerations require generating a single UniqueID from the DataSetID and ObjectID under some conditions. This is done using the formula ID = (DataSetID * 10000000) + ObjectID.

Metadata

Feature level metadata is implemented by linking every data instance with an origin TrackingRecord, either as an attribute of the instance, or by inheriting origin tracking from the DataSet that contains the instance. The TrackingRecord defines a person, organization, and project (an 'activity) that generated the data instance, along with a link to a data processing description for how the information was obtained and introduced to the database. Each TrackingRecord may be linked (through a SimpleRelationship) to one or more bibliographic citations. The metadata scheme will be described further in a document in preparation.

Table naming conventions

Tables and fields are named following the conventions used by international standards efforts such as UML [OMG, 1999] and the Open GIS Consortium. Names are strings with no spaces. The first letter of separate words in the name is capitalized, and no underscore separates words in the name. Typing underscores is error-prone, and under many display conditions, the underscores may be difficult to see. Because of limitations in ArcInfo (v8.0.1) and ArcView (v.3.2) software, field names in spatial data native tables are limited to 10 characters.

DATA OVERVIEW AND ORGANIZATION

Two schema at the end of this document are presented to assist in understanding the data structure. **Figure 1** is a simplified schema showing representation of a 'Default Visualization', which uses the geologic classification and symbolization of the original map author, which are included in the native GIS data tables (AAT and PAT in ESRI terminology). This schema includes some representation of description—spatial objects, sample locations and structural measurements are included, and the major elements of the feature-level metadata implementation. It does not include the correlation tables necessary for building general relationships between objects. **Figure 2** is a simplified schema showing the general relationship structure, and the MapView and MapLegend tables that define different visualizations based on the same data. The metadata representation is very schematic (only one table shown). This schema includes some explanatory text. All the tables shown on these schemata are described in this text, and the figures should be referenced throughout the following discussion.

The geologic and cartographic information in the database is organized into several ArcInfo coverages and ArcView shapefiles. The Geo polygon and arc coverage contains the lines that represent geologic contacts and faults, and the associated polygons based on those lines that define the outcrop area of map units. The Pnt point coverage contains the field observation stations that record things such as structural measurements and collected rock samples. The GeoLines line shapefile contains the geologic lines that do not define boundaries between rock units, such as concealed

faults and fold hinge surface traces. The CartoLines line shapefile contains cartographic lines, such as text lead-in lines. Last, the CartoPnts point shapefile locates the cartographic point features used in the default map layout, such as text labels. These coverages and shapefiles, and the user-defined features included in their feature attribute tables, are summarized in Table 2, Table 9, and Table 12.

Every spatial object (point, line, or polygon) is uniquely identified by a compound primary key consisting of a source-file identifier, DatasetID, and a unique identifier within that file, 'DataSetName'&ID (referred to as ObjectID here). The ArcInfo-assigned Coverage-ID field, a seemingly good candidate for unique identifiers, is apparently not immutable under build and clean operations on the data set. Therefore, ObjectID was added as a user-defined attribute, and the uniqueness constraint must be enforced by the user. The ObjectID values in the tables in this database should not be edited unless the user fully understands the data structure and the ramifications of editing the primary key in a relational database table. All points, lines, and polygons have a TrackingID attribute that joins with the *TrackingRecord* table to show the source origination and tracking information for each object. Geologic points and lines also have an Accuracy attribute that defines the location uncertainty for the point or line in meters. The compound object key, ObjectID and DatasetID, and the compound source tracking key, TrackingID and TrackingDS, plus the Accuracy attribute for geologic points and lines, are the minimal set of attributes fundamental to each spatial object.

A number of other attributes are also included in the coverage and shapefile tables to facilitate visualization of the geologic data in a default layout, and to allow querying against a default classification scheme equivalent to the original source map. These default values also make simple analyses of the map possible in non-relational database environments required by some users of AZGS data. The compound classification concept attribute, ConceptID and ConceptDS, defines the default classification of every object (Fault; Bedding; Surficial Map Unit…); the classification confidence attribute, CConf, provides a subjective measure of the confidence level for the classification of the object (Low; Standard...); and the compound cartographic object attribute, CartoObjID and CartoObjDS, defines the cartographic object used to symbolize each feature in the default visualization (0.35pt. solid black line (24K); Inclined bedding symbol – color black (24K); PMS-1205…). There is also a Label attribute used to store any specific labels or names associated with an object, such as unit names for geologic polygons, and a Name attribute that contains a brief description of each object for simplification purposes. Polygon features have a map unit confidence attribute, MConf, that provides a subjective measure of the identification confidence of a polygon to a particular map unit (Low; Standard...). Point features also have a Rotate attribute, measured anticlockwise, starting from a compass azimuth of 90º, that defines the degree of rotation of graphical elements used for feature symbolization in the ArcView project. The rotation magnitude is specific to the graphical environment of ArcView 3.2 using the AZGSgeofont true type font. Use of these geographic data sets with a different GIS platform and/or font may require that the rotation values in the Rotate attribute be recalculated.

GEOLOGIC SPATIAL DATA

Geologic Coverages

The coverages below are specific to the geospatial database for a particular geologic data set. These coverages, and the user-defined features included in their feature attribute tables, are summarized in Table 2.

Table 2. Summary of coverages showing fields, field definitions, and associated database tables. The ObjectID field, along with the DataSetID field, is the compound primary key for each coverage. If a field is a foreign key to a lookup table, the table name is shown adjacent to that field in the last column.

Coverage Name	Type	Field Name	Data Type	Width	Lookup Tables
Geo.pat (field definitions start on page 7)	Poly	ObjectID DataSetID TrackingID TrackingDS ConceptID ConceptDS CConf CartoObjID CartoObjDS MapUnitID MapUnitDS MConf Label Name	Integer Integer Integer Integer Integer Integer Character Integer Integer Integer Integer Character Character Character	16 16 16 16 16 16 16 16 16 16 16 16 50 255	 DataSetAZ TrackingRecord DataSetAZ ClassificationConcept DataSetAZ CartographicObject DataSetAZ MapUnitsRoskruge DataSetAZ
Geo.aat (field definitions start on page 10)	Line	ObjectID DataSetID Accuracy TrackingID TrackingDS ConceptID ConceptDS CConf CartoObjID CartoObjDS Label Name	Integer Integer Float Integer Integer Integer Integer Character Integer Integer Character Character	16 16 8 16 16 16 16 16 16 16 50 255	 DataSetAZ TrackingRecord DataSetAZ ClassificationConcept DataSetAZ CartographicObject DataSetAZ
Pnt.pat (field definitions start on page 11)	Point	ObjectID DataSetID Accuracy TrackingID TrackingDS ConceptID ConceptDS CConf CartoObjID CartoObjDS Label Rotate Name	Integer Integer Float Integer Integer Integer Integer Character Integer Integer Character Integer Character	16 16 8 16 16 16 16 16 16 16 50 4 255	 DataSetAZ TrackingRecord DataSetAZ ClassificationConcept DataSetAZ CartographicObject DataSetAZ

Geology Coverage

The **Geo** coverage is a polygon and arc coverage that contains geologic lines that bound polygons (contacts, faults, mapping boundaries...), or represent surfaces that are discontinuous within polygons (faults that become buried or die out). The polygon topology defined by the lines in this coverage identifies the mapped distribution of rock units.

Polygon Attributes
- **ObjectID**: Integer, width 16. First part of the compound primary key. Uniquely identifies each feature in the Geo polygon coverage. Each feature has a different value. Domain: Integers >0 and <10^{16}, no duplicates.

- **DataSetID**: Integer, width 16. Second part of the compound primary key. Uniquely identifies the Geo polygon coverage. All features in the data set have the same value. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **TrackingID**: Integer, width 16. Uniquely identifies the origin tracking for each object. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **TrackingDS**: Integer, width 16. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **ConceptID**: Integer, width 16. Specifies the concept used to classify the kind of unit a particular polygon represents. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 3.

Table 3. Example classification concept ID codes used in the Geo.pat table.

ConceptID	Name
2405	Surficial Map Unit
2406	Rock Volume Map Unit
2424	Not Defined

- **ConceptDS**: Integer, width 16. Specifies the data set that contains the data object identified by ConceptID for each record. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **CConf**: Character, width 16. Assigns a qualitative confidence level to the classification of the kind of unit represented by the polygon. Domain: 'low', 'standard', or 'high'.
- **CartoObjID**: Integer, width 16. Identifies the cartographic symbolization used for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: See **Table 4**.

Table 4. Example cartographic object codes used in the Geo.pat table

CartoObjID	Seq.	Name
999	1	Transparent
1008	1	PMS-100
1036	1	PMS-1205
1941	1	PMS-728
2200	1	Blue (R135,G207,B254)
2216	1	LtGreen (R120,G254,B185)
2228	1	Red (R240,G128,B128)
2230	1	Tan (R244,G213,B158)
2231	1	Transparent background
2231	2	Blue (R39,G146,B182) 1st pattern layer
2232	1	Transparent background
2232	2	Blue (R69,G228,B236) 1st pattern layer
2240	1	Transparent background
2240	2	LtGreen (R130,G250,B183) 1st pattern layer

CartoObjID	Seq.	Name
2241	1	Transparent background
2241	2	LtPurple (R179,G175,B213) 1st pattern layer
2254	1	PMS-454 background
2254	2	PMS-1205 1st pattern layer

- **CartoObjDS**: Integer, width 16. Specifies the data set that contains the data object identified by CartoObjID for each record. Domain: Single value (typically), the DataSetID for the *CartographicObject* table.
- **MapUnitID**: Integer, width 16. Identifies the geologic map unit associated with each polygon of type "Rock Volume Map Unit" (ID = 2406) or "Surficial Map Unit" (ID = 2405). It is a foreign key that joins to the ObjectID field of the *MapUnits* table. Domain: See Table 5.

Table 5. Example rock unit identification codes used in the Geo.pat table.

MapUnitID	Name
0	Area not digitized
1	Abrigo Formation (Middle Cambrian)
2	Abrigo Formation (Middle Cambrian) - lower sandstone and mudstone unit
3	Abrigo Formation (Middle Cambrian) - middle mottled carbonate unit
4	Abrigo Formation (Middle Cambrian) - upper sandstone, marl, and limestone unit
5	Bolsa Quartzite (Cambrian)
6	Bolsa and Abrigo Formations, undivided
7	Bolsa, Abrigo and Martin Formations, undivided, photogeologic identification
8	Martin Formation (Devonian)
9	Quartz arenite (Jurassic?)
11	Andesite to dacite volcaniclastic breccia (Jurassic?)
12	Reddish mudstone, siliceous argillite, and quartz-arenite (Jurassic or Triassic)

MapUnitID	Name
13	Light gray to white, feldspathic quartz arenite, and quartzite cobble conglomerate (Jurassic or Triassic)
14	Light greenish-gray siliceous argillite and feldspathic sandstone (Jurassic or Triassic)
15	Red mudstone and volcanic lithic sandstone, with interbedded quartzite in lower part (Jurassic or Triassic)
16	Andesite (Cretaceous or Jurassic); sedimentary and volcanic sequence east of the Recortado Well Fault
17	Intrusive andesite (Cretaceous or Jurassic); sedimentary and volcanic sequence east of the Recortado Well Fault
18	Andesite breccia (Cretaceous); sedimentary and volcanic sequence east of the Recortado Well Fault
24	Mafic to intermediate volcanic and shallow intrusive rocks (Cretaceous or Jurassic)
25	Mafic sill (Cretaceous); sedimentary and volcanic sequence east of the Recortado Well Fault
26	Sandstone photogeologic unit 1 (Cretaceous or Jurassic)

- **MapUnitDS**: Integer, width 16. Specifies the data set that contains the data object identified by MapUnitID for each record. Domain: Single value (typically), the DataSetID for the *MapUnits* table.
- **MConf**: Character, width 16. For polygons of type "Rock Volume Map Unit" (ID = 2406) or "Surficial Map Unit" (ID = 2405), indicates the subjective confidence of the person making the map unit classification in the assignment of the material within the polygon to a particular rock volume or surficial geologic map unit. Otherwise the field does not contain a value. Domain: 'low', 'standard', or 'high'.
- **Label**: Character, width 50. Equivalent to the geologic map unit labels on the default map visualization. This attribute represents the default classification of each polygon to a particular rock unit and is included to make symbolizing and viewing the default map visualization relatively simple. The label is queried if the classification confidence is low. In addition, map

labels with Tertiary, Pennsylvanian, Precambrian, Cambrian, Paleozoic, or Mesozoic geologic age prefixes are shown using their corresponding special font symbols included in the AZGSArial true type font (included with database distribution packages, in a file named Azgsa___.ttf). When shown in the default ArcView font, as in ArcView tables, these special characters are displayed as follows: ₮R = ² (Alt-0178) ; ℙ = ³ (Alt-0179); pℂ = ¹ (Alt-0185); ℂ = º (Alt-0186); ℞z = ¼ (Alt-0188); and Mz = ¾ (Alt-0190). Domain: Free text.

- **Name**: Character, width 255. Equivalent to the geologic map unit names in the map explanation on the default map visualization. This is a redundant field added to simplify the use of the data set in non-relational database environments. Domain: Free text.

Arc Attributes

- **ObjectID**: Integer, width 16. First part of the compound primary key. Uniquely identifies each feature in the Geo arc coverage. Each feature has a different value. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Integer, width 16. Second part of the compound primary key. Uniquely identifies the Geo arc coverage. All features in the data set have the same value. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **Accuracy**: Float (real), single precision. Represents the spatial uncertainty in the location of a feature, in meters. For example, a value of 10 for a line feature indicates that the geologic entity represented by the line on the default map visualization is within 10 meters of the mapped feature's actual location on the ground. At present this uncertainty combines the geologic uncertainty in the accuracy of location (e.g. for a gradational or poorly exposed contact), and the numerical uncertainty in the computer representation of the line location resulting from accumulated calculation and digitizing errors. The uncertainty must be greater than the numerical precision of the X,Y coordinates that locate a point (i.e. the accuracy cannot exceed the precision). This value determines the line style that represents the line by using standard solid, dashed, and dotted lines. For most existing maps, this length will be based on standard map accuracy, i.e. the geologic entity is located within the width of the line shown on a map for a solid line. In this data set, location uncertainties are qualitatively estimated. A value of 0 indicates that accuracy is not defined, as in the case of cartographic lines. Domain: rational numbers >numerical precision of data and $<10^{8}$.
- **TrackingID**: Integer, width 16. Uniquely identifies the origin tracking for each object. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **TrackingDS**: Integer, width 16. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **ConceptID**: Integer, width 16. Specifies the concept used to identify the kind of spatial feature represented by this record. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 6.

Table 6. Example classification concept code values used in the Geo.aat table.

ConceptID	Name
7	Contact, not classified, timing not specified
58	Fault, High-angle, normal separation
596	Fault, Generic high-angle, separation unknown

ConceptID	Name
642	Mapping boundary surface
2423	Contact, intraformational, timing not specified

- **ConceptDS**: Integer, width 16. Specifies the data set that contains the data object identified by ConceptID for each record. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **CConf**: Character, width 16. Assigns a qualitative confidence level to the classification of an object. Domain: 'low', 'standard', or 'high'.
- **CartoObjID**: Integer, width 16. Identifies the cartographic symbolization for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: See Table 7.

Table 7. Example cartographic object codes used in Geo.aat table.

CartoObjID	Name
53	Null line symbol
54	0.5pt dashed black line (24K) (Approximate contact)
55	0.5pt solid black line (24K) (Accurate contact)
59	1.5pt dashed black line (24K) (Approximate fault)
60	1.5pt solid black line with queries (24K) (Queried accurate fault)
61	1.5pt solid black line (24K) (Accurate fault)
65	2.5pt solid black line (24K) (Map neat line)
67	0.75pt dotted black line (24K) (Concealed contact)
68	0.5pt solid black line with queries (24K) (Queried accurate contact)
71	0.5pt black line with dash-dot pattern (24K) (Scratch contact)

- **CartoObjDS**: Integer, width 16. Specifies the data set that contains the data object identified by CartoObjID for each record. Domain: Single value (typically), the DataSetID for the *CartographicObject* table.
- **Label**: Character, width 50. When used, contains strings that identify line features that have a label associated with them, as in the case of named faults. Domain: Free text.
- **Name**: Character, width 255. Identifies the default classification of each type of line and is included for simplification purposes. Features not shown on the default map visualization are indicated here as being hidden. Domain: Free text.

Point Coverage

The **Pnt** coverage is a point coverage that represents geologic spatial features located at a distinct point (structural measurement stations, rock samples collection stations...).

Point Attributes

- **ObjectID**: Integer, width 16. First part of the compound primary key. Uniquely identifies each feature in the Pnt point coverage. Each feature has a different value. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Integer, width 16. Second part of the compound primary key. Uniquely identifies the Pnt point coverage. All features in the data set have the same value. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.

- **Accuracy**: Integer, width 8. Represents the spatial uncertainty in the location of a feature, in meters. For example, a value of 10 for a point feature indicates that location of the point recorded in the database is within 10 meters of the mapped feature's actual location on the ground. At present this uncertainty combines the geologic uncertainty in the accuracy of location (e.g. for a gradational or poorly exposed contact), and the numerical uncertainty in the computer representation of the line location resulting from accumulated calculation and digitizing errors. The uncertainty must be greater than the numerical precision of the X,Y coordinates that locate a point (i.e. the accuracy cannot exceed the precision). A value of 0 indicates that accuracy is not defined. Domain: >numerical precision of data and $<10^8$.
- **TrackingID**: Integer, width 16. Uniquely identifies the origin tracking for each object. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <1016, no duplicates.
- **TrackingDS**: Integer, width 16. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **ConceptID**: Integer, width 16. Specifies the concept used to identify the kind of spatial feature represented by this record. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: 3340 = Field Observation Station.
- **ConceptDS**: Integer, width 16. Specifies the data set that contains the data object identified by ConceptID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **CConf**: Character, width 16. Assigns a qualitative confidence level (accuracy) to the classification of the object. Domain: 'low', 'standard', or 'high'.
- **CartoObjID**: Integer, width 16. Identifies the cartographic symbolization for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: See Table 8.

Table 8. Example cartographic object codes used in the Pnt.pat table.

CartoObjID	Name
52	Null point symbol
2055	Inclined bedding symbol - color black (24K)
2056	Approximate inclined bedding symbol - color black (24K)
2057	Inclined crenulated or warped bedding symbol - color black (24K)
2058	Inclined bedding w/tops known symbol - color black (24K)
2059	Overturned bedding symbol - color black (24K)
2060	Overturned bedding w/tops known symbol - color black (24K)
2062	Vertical bedding symbol - color black (24K)
2064	Vertical bedding w/tops known symbol - color black (24K)
2076	Generic inclined foliation symbol - color black, open triangle (24K)
2093	Inclined eutaxitic foliation symbol - color black (24K)
2096	Inclined flow foliation symbol - color black (24K)
2110	Inclined close disjunct cleavage symbol - color black (24K)
2111	Vertical close disjunct cleavage symbol - color black (24K)
2115	Inclined bedding parallel to cleavage symbol - color black (24K)
2143	Minor anticline symbol - color red (24K)
2165	Fault attitude symbol - color black (24K)
2172	Circle with filled central circle (USGS 26.2.5) - color black (24K)

- **CartoObjDS**: Integer, width 16. Specifies the data set that contains the data object identified by CartoObjID. Domain: Single value (typically), the DataSetID for the *CartographicObject* table.
- **Label**: Character, width 50. When used, contains strings that are equivalent to any labels associated with point features on the default map visualization. In this database, this field contains sample identification numbers or is empty. Domain: Free text.
- **Rotation**: Integer, width 4. Specifies the rotation of the font symbol to correctly represent the azimuth of the of geologic feature displayed on the default map visualization at this point. This value is specific to the graphical environment of ArcView 3.2 using the AZGSgeo.ttf font (/support/fonts directory in the distribution package); the rotation angle is measured anti-clockwise, starting from a compass azimuth of 90º, and is also dependent on the orientation of the symbol in its font definition. Domain: 0 to ±360.
- **Name**: Character, width 255. Identifies the default classification of each type of point and is included for simplification purposes. Features not shown on the default map visualization are indicated here as being hidden. Domain: Free text.

Geologic Shapefiles

An ESRI shapefile containing geologic lines may be included in the geospatial database for each geologic data set if necessary. The geologic features information in this file are conceptually equivalent to those included in the OtherLines coverage, but may be represented instead using a shapefile. This shapefile, and the user-defined features included in its feature attribute table, is summarized in Table 9.

Table 9. Summary of geologic shapefiles showing fields, field definitions, and associated database tables. The ObjectID field, along with the DataSetID field, is the compound primary key for each shapefile. If a field joins to a lookup table, the table name is shown adjacent to that field in the last column.

Shapefile Name	Type	Field Name	Data Type	Width	Lookup Tables
GeoLines	Line	ObjectID	Integer	16	
		DataSetID	Integer	16	DataSetAZ
		Accuracy	Integer	8	
(field definitions start		TrackingID	Integer	16	TrackingRecord
on page 14)		TrackingDS	Integer	16	DataSetAZ
		ConceptID	Integer	16	ClassificationConcept
		ConceptDS	Integer	16	DataSetAZ
		CConf	Character	16	
		CartoObjID	Integer	16	CartographicObject
		CartoObjDS	Integer	16	DataSetAZ
		Label	Character	50	
		Name	Character	255	

Other Geologic Lines Shapefile

The **GeoLines** shapefile is a line shapefile that contains those geologic lines that do not define polygon topology and do not represent surfaces that are discontinuous within polygons (concealed faults, fold hinges, dikes, marker beds...).

Arc Attributes

- **ObjectID**: Integer, width 16. First part of the compound primary key. Uniquely identifies each feature in the GeoLines line shapefile. Each feature has a different value. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Integer, width 16. Second part of the compound primary key. Uniquely identifies the GeoLines line shapefile. All features in the data set have the same value. Domain: Single value (typically), the DataSetID for this table in the _DataSetAZ_ data set.
- **Accuracy**: Integer, width 8. Represents the spatial uncertainty in the location of a feature, in meters. For example, a value of 10 for a line feature indicates that the geologic entity represented by the line on the default map visualization is within 10 meters of the mapped feature's actual location on the ground. At present this uncertainty combines the geologic uncertainty in the accuracy of location (e.g. for a gradational or poorly exposed contact), and the numerical uncertainty in the computer representation of the line location resulting from accumulated calculation and digitizing errors. The uncertainty must be greater than the numerical precision of the X,Y coordinates that locate a point (i.e. the accuracy cannot exceed the precision). This value determines the line style that represents the line by using standard solid, dashed and dotted lines. For most existing maps, this length will be based on standard map accuracy, i.e. the geologic entity is located within the width of the line shown on a map for a solid line. A value of 0 indicates that accuracy is not defined, as in the case of cartographic lines or the map neatline. Domain: >numerical precision of data and $<10^8$.
- **TrackingID**: Integer, width 16. Uniquely identifies the origin tracking for each object. It is a foreign key that joins to the TrackingID field of the _TrackingRecord_ table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **TrackingDS**: Integer, width 16. Specifies the data set that contains the data object identified by TrackingDS. Domain: Single value (typically), the DataSetID for the _TrackingRecord_ table.
- **ConceptID**: Integer, width 16. Specifies the classification concept used to identify the kind of spatial feature represented by this record. It is a foreign key that joins to the ConceptID field of the _ClassificationConcept_ table. Domain: See Table 10.

Table 10. Example classification concept codes used in the GeoLines table.

ConceptID	Name
58	Fault, High-angle, normal separation
530	Vein
596	Fault, Generic high-angle, separation unknown
612	Marker bed
1988	Fold hinge surface, upright anticline
1994	Fold hinge surface, upright syncline
2379	Mafic dike
2380	Intermediate dike
2381	Felsic dike

- **ConceptDS**: Integer, width 16. Specifies the data set that contains the data object identified by ConceptID. Domain: 1 = the DataSetID for the _ClassificationConcept_ table.
- **CConf**: Character, width 16. Assigns a qualitative confidence level to the classification of the object. Domain: 'low', 'standard', or 'high'.

- **CartoObjID**: Integer, width 16. Identifies the cartographic symbolization for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: See Table 11.

Table 11. Example cartographic object codes used in the GeoLines table.

CartoObjID	Name
53	Null line symbol
56	0.35pt dashed red line (24K) (Approximate fold hinge line)
58	0.35pt solid red line (24k) (Accurate fold hinge line)
63	1.75pt dotted black line (24K) (Concealed fault)
66	0.5pt solid black line with perpendicular hashes (24K) (Dike symbol)
69	0.5pt solid black line with spaced X's (24K) (Dike symbol)
70	0.5pt solid black line with open circles (24K) (Vein symbol)
72	0.5pt black line with dash-dot-dot pattern (24K) (Marker bed)
73	0.35pt dotted red line (24K) (Concealed fold hinge line)
74	0.5pt solid black line with alternating slashes (24K) (Dike symbol)

- **CartoObjDS**: Integer, width 16. Specifies the data set that contains the data object identified by CartoObjID. Domain: Single value (typically), the DataSetID for the *CartographicObject* table.
- **Label**: Character, width 50. When used, contains strings that identify line features that have a label associated with them, as in the case of dikes that are associated with a particular geologic map unit. Domain: Free text.
- **Name**: Character, width 255. Identifies the default classification of each type of line and is included for simplification purposes. Features not shown on the default map visualization are indicated here as being hidden. Domain: Free text.

CARTOGRAPHIC SPATIAL OBJECTS

Cartographic elements for the default map visualization of a particular geologic data set are included in a line and a point shapefile. These shapefiles, and the user-defined features included in their feature attribute tables, are summarized in Table 12. Because the locations of points and lines in these shapefiles are chosen to provide cartographic clarity, the Accuracy and CConf fields are irrelevant and therefore not included.

Table 12. Summary of cartographic shapefiles showing fields, field definitions, and associated database tables. The ObjectID field, along with the DataSetID field, is the compound primary key for each shapefile. If a field joins to a lookup table, the table is shown adjacent to that field in the last column.

Shapefile Name	Type	Field Name	Data Type	Width	Lookup Tables
CartoLines	Line	ObjectID	Integer	16	
		DataSetID	Integer	16	DataSetAZ
		TrackingID	Integer	16	TrackingRecord
(field definitions start on page 16)		TrackingDS	Integer	16	DataSetAZ
		ConceptID	Integer	16	ClassificationConcept
		ConceptDS	Integer	16	DataSetAZ
		CartoObjID	Integer	16	CartographicObject
		CartoObjDS	Integer	16	DataSetAZ
		Name	Character	255	

Shapefile Name	Type	Field Name	Data Type	Width	Lookup Tables
CartoPnts	Point	ObjectID	Integer	16	
		DataSetID	Integer	16	DataSetAZ
		TrackingID	Integer	16	TrackingRecord
(field definitions start		TrackingDS	Integer	16	DataSetAZ
on page 17)		ConceptID	Integer	16	ClassificationConcept
		ConceptDS	Integer	16	DataSetAZ
		CartoObjID	Integer	16	CartographicObject
		CartoObjDS	Integer	16	DataSetAZ
		Label	Character	50	
		Rotate	Integer	4	
		Name	Character	255	

Cartographic Lines Shapefile

The **CartoLines** shapefile contains the cartographic lines (text lead-in lines...) used in the default map visualization. The locations of these lines have no geologic significance.

Arc Attributes

- **ObjectID**: Integer, width 16. First part of the compound primary key. Uniquely identifies each feature in the CartoLines line shapefile. Each feature has a different value. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Integer, width 16. Second part of the compound primary key. Uniquely identifies the CartoLines line shapefile. All features in the data set have the same value. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **TrackingID**: Integer, width 16. Uniquely identifies the origin tracking for each object. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **TrackingDS**: Integer, width 16. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **ConceptID**: Integer, width 16. Specifies the classification concept used to identify the kind of spatial feature represented by this record. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: Integers >0 and $<10^{16}$, for example 2396 = Text lead-in line; 2419 = Cross Section Surface trace.
- **ConceptDS**: Integer, width 16. Identifies the data set that contains the data object identified by ConceptID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **CartoObjID**: Integer, width 16. Identifies the cartographic symbolization for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: Integers >0 and $<10^{16}$, for example 53 = Null line symbol; 57 = 0.35pt solid black line (24K).
- **CartoObjDS**: Integer, width 16. Specifies the data set that contains the data object identified by CartoObjID. Domain: Single value (typically), the DataSetID for the *CartographicObject* table.
- **Label**: Character, width 50. When used, contains strings that identify line features that have a label associated with them. Domain: Free text.
- **Name**: Character, width 255. Identifies the default classification of each type of line and is included for simplification purposes. Domain: Free text.

Cartographic Points Shapefile

The **CartoPnts** shapefile contains the cartographic points (text labels, fault symbols, fold geometry symbols...) used in the default map visualization. The locations of these points have no direct geologic significance.

Point Attributes

- **ObjectID**: Integer, width 16. First part of the compound primary key. Uniquely identifies each feature in the CartoPnts point coverage. Each feature has a different value. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Integer, width 16. Second part of the compound primary key. Uniquely identifies the CartoPnts point coverage. All features in the data set have the same value. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **TrackingID**: Integer, width 16. Uniquely identifies the origin tracking for each object. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}.
- **TrackingDS**: Integer, width 16. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **ConceptID**: Integer, width 16. Specifies the classification concept used to identify the kind of spatial feature represented by this record. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 13.

Table 13. Example classification concept codes used in the CartoPnts table.

ConceptID	Name
3057	Discrete feature point symbols
3317	Annotation, unit label
3318	Annotation, structural measurement label
3321	Annotation, generic text

- **ConceptDS**: Integer, width 16. Specifies the data set that contains the data object identified by ConceptID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **CartoObjID**: Integer, width 16. Identifies the cartographic symbolization for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: See Table 14.

Table 14. Example cartographic object codes used in the CartoPnts table.

CartoObjID	Name
2134	anticline symbol
2137	syncline symbol
2169	normal fault symbol
2177	plunge arrowhead
2270	structural measurement label
2270	unit label
2271	dike label
2271	generic text label
2271	structural measurement label

CartoObjID	Name
2271	unit label
2272	dike label
2272	generic text label
2272	unit label
2273	generic text label
2275	unit label
2276	unit label
2277	unit label
2279	generic text label

- **CartoObjDS**: Integer, width 16. Specifies the data set that contains the data object identified by CartoObjID. Domain: Single value (typically), the DataSetID for the *CartographicObject* table.
- **Rotation**: Integer, width 4. Specifies the rotation required for the font symbol that represents the azimuth of the geologic feature to display correctly on the default map visualization. This value is specific to the graphical environment of ArcView 3.2 using the AZGSgeo.ttf font (/support/fonts directory in the distribution package); the rotation angle is measured anti-clockwise, starting from a compass azimuth of 90º, and is also dependent on the orientation of the symbol in its font definition. Domain: 0 to ±360.
- **Label**: Character, width 50. When used, contains strings that identify point features that have a label associated with them. For text label points, the field contains strings that are equivalent to the text labels that appear on the default map visualization. Domain: Free text.
- **Name**: Character, width 255. Identifies the default classification of each type of point and is included for simplification purposes. Domain: Free text.

THEMATIC GEOLOGY DATABASE TABLES

Additional tables may be included that contain classification concepts and descriptions specific to a particular geologic data set. These tables, summarized in Table 15, are included as part of a Microsoft Access database. By default, each data set field below references a table that is included in the Arizona Geological Survey namespace.

Table 15. Summary of project-specific Microsoft Access database tables showing fields, field definitions, and associated database tables. If a field joins to a lookup table, the table name is shown adjacent to that field in the last column.

Table Name	Field Name	Data Type	Width	Lookup Tables
MapUnits	MapUnitID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	TrackingID	Number	Long Integer	TrackingRecord
(field definitions start on page 19)	TrackingDS	Number	Long Integer	DataSetAZ
	MapLabel	Text	50	
	Name	Text	255	
	Description	Memo		

Table Name	Field Name	Data Type	Width	Lookup Tables
Samples	ObjectID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	ActivityID	Number	Long Integer	Activities
(field definitions start on	ActivityDS	Number	Long Integer	DataSetAZ
page 20)	FieldID	Text	30	
	DateCollected	Date/Time		
	UTME	Number	Single	
	UTMN	Number	Single	
	UTMzone	Number	Long Integer	
	SpObjID	Number	Long Integer	Pnt
	SpObjDS	Number	Long Integer	DataSetAZ
	Area	Text	255	
	Quadrangle	Text	255	
	RockUnitID	Number	Long Integer	MapUnits
	RockUnitDS	Number	Long Integer	DataSetAZ
	MapUnit	Text	50	
	Notes	Memo		
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
StructureData	StructMeasureID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	SpObjID	Number	Long Integer	Pnt
(field definitions start on	SpObjDS	Number	Long Integer	DataSetAZ
page 20)	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
	Name	Text	255	
	StructMeasureTypeID	Number	Long Integer	ClassificationConcept
	StructMeasureTypeDS	Number	Long Integer	DataSetAZ
	CConf	Text	16	
	UTME	Number	Single	
	UTMN	Number	Single	
	LocErr	Number	Single	
	Azimuth	Number	Single	
	AzimuthErr	Number	Single	
	Dip	Number	Single	
	DipErr	Number	Single	
	DipDirection	Text	16	

Map Unit Table

The **MapUnits** table defines the map units used to classify polygons in the *Geo* coverage.

Database Table Fields

- **MapUnitID**: Number, long integer. First part of the compound primary key. Uniquely identifies each geologic map unit in the MapUnits data set. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the MapUnits data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.

- **OriginDate**: Date/Time. Records when the record was created. This information provides more detailed information on the time that records were originally entered, supplementing the information in the associated *TrackingRecord* table. Format: '*mm/dd/yy*'.
- **MapLabel**: Text, width 25. Contains labels equivalent to those used on the default geologic map visualization. These map labels do not necessarily correspond to the map labels in the Label field of the *Geo.pat* table for the polygon that contains the point. This is because some generalization of the geology has been made for the default visualization, and because labels may be located outside of the polygon they identify. Map labels with Tertiary, Pennsylvanian, Precambrian, Cambrian, Paleozoic, or Mesozoic geologic age prefixes are shown using their corresponding special font symbols included in the AZGSArial True Type font (included in database distribution package). When shown in the default ArcView font, as in ArcView tables, these prefixes map to the following characters: T̶R = ² (Alt-0178) ; P̶P = ³ (Alt-0179); pC̶ = ¹ (Alt-0185); C̶ = º (Alt-0186); P̶z = ¼ (Alt-0188); and M̶z = ¾ (Alt-0190). Domain: Limited to the map labels on the original source maps.
- **Name**: Text, width 255. Identifies the map unit name or rock type. Domain: Free text.
- **Description**: Memo. Full description of the rock unit. Domain: Free text.

Samples Table

The **Samples** table contains location and description information for rock samples collected within the extent of the geologic data set. The inclusion of both the UTM coordinates for the sample location and a link to a spatial object representing the sample location is redundant, but both forms of location are included for reliability. If the link with the spatial object data set is corrupted, the **Samples** table still contains sufficient information to locate the sample. Likewise, the sample table can be exported for data exchange without including a data set with location spatial objects.

Database Table Fields

- **ObjectID**: Number, long integer. First part of the compound primary key. Uniquely identifies each record in the Samples table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the Samples table. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **ActivityID**: Number, long integer. Identifies the Activity for collection of the sample. Activities for sample collection should indicate the person who collected the sample. Domain: Integers >0 and $<10^{16}$.
- **ActivityDS**: Number, long integer. Uniquely identifies the *Activities* data set that contains the Activity definition. Domain: 2 = the DataSetID for the *Activities* table.
- **FieldID**: Text, width 30. Records the sample identifier assigned to the rock collected in the field by the original collector.
- **DateCollected**: Date/Time. Records when a sample was collected. Format: '*mm/dd/yy*'.
- **UTME**: Number, real, single precision. UTM easting coordinate for sample location. Domain: $122000 < UTME < 700000$.
- **UTMN**: Number, real, single precision. UTM northing coordinate for sample location. Domain: $3420000 < UTMN < 4110000$.
- **UTMzone**: Number, long integer. Zone number for UTM coordinates. Domain: 11 or 12 for the State of Arizona.

- **SpObjID**: Number, long integer. Uniquely identifies the field station record in an associated point coverage where the sample was collected. It is a foreign key that joins with the ObjectID field of the *Pnt* point coverage. Domain: Integers >0 and <10^{16}, but limited to the values in the ObjectID field of the associated point data sets.
- **SpObjDS**: Number, long integer. Specifies the data set that contains the data object identified by SpObjID. Domain: Single value (typically), the DataSetID for the *Pnt* point coverage.
- **Area**: Text, width 64. Geographic area name from Arizona Geological Survey Place Names list. Domain: Place names included in Trapp and Reynolds [1998].
- **Quadrangle**: Text, width 64. Name of USGS 7½ minute quadrangle that contains the sample location. Domain: USGS 7½ Quadrangle names.
- **RockUnitID**: Number, long integer. Uniquely identifies the geologic map unit in the *MapUnits* data set from which the sample was collected. Domain: Integers >0 and <10^{16}.
- **RockUnitDS**: Number, long integer. Uniquely identifies the *MapUnits* data set. Domain: Single value (typically), the DataSetID for the MapUnits data set in the DataSetAZ data set.
- **Notes**: Memo. Free text notes on sample.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.

Structural Measurement Data Table

The **StructureData** table contains values that define the orientation of structural features. The inclusion of both the UTM coordinates for the station location and a link to a spatial object representing the station location is redundant, but both forms of location are included for reliability. If the link with the spatial object data set is corrupted, the **StructureData** table still contains sufficient information to locate the station. Likewise, the **StructureData** table can be exported for data exchange without including a data set with location spatial objects. A separate correlation table to link stations with locations is unnecessary because each station has a unique location.

Database Table Fields
- **StructMeasureID**: Number, long integer. First part of the compound primary key. Uniquely identifies each structural measurement in the StructureData table. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the StructureData table. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **SpObjID**: Number, long integer. Uniquely identifies a field station record in an associated point coverage. It is a foreign key that joins with the ObjectID field of the *Pnt* point coverage. Domain: Integers >0 and <10^{16}, but limited to the values in the ObjectID field of the associated point data sets.
- **SpObjDS**: Number, long integer. Specifies the data set that contains the data object identified by SpObjID. Domain: Single value (typically), the DataSetID for the *Pnt* point coverage.

- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **Name**: Text, width 255. Provides a descriptive name for each type of structural measurement. Domain: Free text.
- **StructMeasureTypeID**: Number, long integer. Classifies the geologic concept for each structural measurement. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 16.

Table 16. Example structural measurement type codes used in the StructureData table.

StructMeasureTypeID	Name
42	Close disjunct cleavage
543	Flow foliation
544	Eutaxitic foliation
546	Foliation, generic
555	Joints
559	Well developed s-tectonite
563	Cleavage parallel to bedding
567	Minor fault surface
572	Lineation, generic tectonic
581	Fold hinge, anticline
588	Orientation, fault surface
762	Bedding, crude or indistinct
768	Bedding, planar parallel
3324	Bedding, planar parallel, w/tops
3326	Bedding, contorted or variable
3329	Fold hinge, syncline

- **StructMeasureTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by StructMeasureTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **CConf**: Text, width 16. Assigns a qualitative confidence level to the classification of this object. Domain: 'low', 'standard', or 'high'.
- **UTME**: Number, single-precision real. The UTM easting coordinate (X-value), in meters, for the location of the structural measurement. Domain: Real numbers, >0 and <10^{8}.
- **UTMN**: Number, single-precision real. The UTM northing coordinate (Y-value), in meters, for the location of the structural measurement. Domain: Real numbers, >0 and <10^{8}.
- **LocErr**: Number, single-precision real. Records the location error, in meters, for the UTM coordinates. Domain: Real numbers, >0 and <10^{8}.
- **Azimuth**: Number, single-precision real. Records the trend or strike of a structural feature in degrees. For planar surfaces, the measurement is recorded using the right-hand rule (i.e. the measurement is made such that the down-dip direction is to the right when facing in the azimuth direction). The magnitude of the angle is measured clockwise starting from a compass azimuth of 0º. Domain: Real numbers, from 0 to 360.

- **AzimuthErr**: Number, single-precision real. Records the uncertainty, in degrees, associated with an azimuth measurement. For example, an AzimuthErr of 5 for an Azimuth of 127 would indicate that the azimuth actually falls within the range from 122 to 132 degrees. Domain: Real numbers, >0 and $<10^8$.
- **Dip**: Number, single-precision real. Records the angle between a planar or linear feature and horizontal (degrees). The angle is measured in the vertical plane perpendicular to strike for planar features and parallel to trend for linear features. The dip angle here measures total rotation rather than the conventional inclination measurement. For overturned beds this results in dips $>90°$. This allows conceptually consistent representation of the dip of upright, overturned, or doubly overturned structures. Overturned beds have $90 < dip <= 180$. Doubly overturned beds have dip >180. Domain: Real numbers, ≥ 0 and $<10^8$.
- **DipErr**: Number, single-precision real. Records the uncertainty, in degrees, associated with a dip measurement. For example, a DipErr of 3 for a Dip of 29 would indicate that the dip actually falls within the range from 26 to 32 degrees. Domain: Real numbers, >0 and $<10^8$.
- **DipDirection**: Text, width 2. Contains a dip direction modifier, based on compass directions, that may be used as a redundant check for structural measurements recorded using the right-hand rule. Domain: N, NE, E, SE, S, SW, W, NW.

ARIZONA GEOLOGIC DATA SYSTEM TABLES

The lookup tables defined below contain supporting data maintained by the Arizona Geological Survey to support all databases within the organization. These tables, summarized in Table 17, are included as a Microsoft Access database. By default, each data set below references a table that is included in the Arizona Geological Survey namespace.

Table 17. Summary of general Microsoft Access database tables showing fields, field definitions, and associated database tables. If a field joins to a lookup table, the table name is shown adjacent to that field in the last column.

Table Name	Field Name	Data Type	Field Size	Lookup Tables
Activities	ActivityID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Name	Text	255	
(field definitions start on	PersonOrgID	Number	Long Integer	
page 32)	PersonOrgDS	Number	Long Integer	DataSetAZ
	ProjectID	Number	Long Integer	
	ProjectDS	Number	Long Integer	DataSetAZ
	Comment	Memo		

Table Name	Field Name	Data Type	Field Size	Lookup Tables
AttributedRelationship (field definitions start on page 29)	RelationshipID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Sequence	Number	Long Integer	
	RelTypeID	Number	Long Integer	ClassificationConcept
	RelTypeDS	Number	Long Integer	DataSetAZ
	FirstRoleID	Number	Long Integer	
	FirstRoleDS	Number	Long Integer	DataSetAZ
	SecondRoleID	Number	Long Integer	
	SecondRoleDS	Number	Long Integer	DataSetAZ
	CConf	Text	50	
	CBasis	Text	50	
	StringValue	Text	255	
	NumberValue	Number	Single	
	AttributeObjID	Number	Long Integer	
	AttributeObjDS	Number	Long Integer	DataSetAZ
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
	Comment	Memo		
	OriginDate	Date/Time		
AZgeoBibLinkTable (field definitions start on page 34)	AzGeoBibID	Number	Long Integer	AzGeoBib [Trapp et al., 1996]
	DataSetID	Number	Long Integer	
	Authorship	Text	255	
	Title	Text	255	
	Citation	Text	255	
	PublicationDate	Date/Time		
CartographicObject (field definitions start on page 39)	CartoObjID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Sequence	Number	Integer	
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
	Name	Text	255	
	CartoObjTypeID	Number	Long Integer	ClassificationConcept
	CartoObjTypeDS	Number	Long Integer	DataSetAZ
	GraObjID	Number	Long Integer	GraphicLine GraphicPattern GraphicLineOrnamented GraphicTextFormat
	GraObjDS	Number	Long Integer	DataSetAZ
	ColorID	Number	Long Integer	Color
	ColorDS	Number	Long Integer	DataSetAZ
	OriginDate	Date/Time		
	Comment	Memo		
ClassificationConcept (field definitions start on page 27)	ConceptID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	TrackingID	Number	Long Integer	ClassificationConcept
	TrackingDS	Number	Long Integer	DataSetAZ
	Name	Text	255	
	ParentID	Number	Long Integer	TrackingRecord
	ParentDS	Number	Long Integer	DataSetAZ
	OriginDate	Date/Time		
	Definition	Memo		
Color (field definitions start on page 41)	ColorID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Name	Text	255	
	CMYK	Text	25	
	RGB	Text	25	
	R	Number	Integer	
	G	Number	Integer	
	B	Number	Integer	

Table Name	Field Name	Data Type	Field Size	Lookup Tables
DataSetAZ (field definitions start on page 34)	DataSetID	Number	Long Integer	
	NameSpace	Text	50	
	NameSpaceID	Number	Long Integer	ClassificationConcept
	NameSpaceDS	Number	Long Integer	DataSetAZ
	DataSetName	Text	255	
	DataSetTypeID	Number	Long Integer	ClassificationConcept
	DataSetTypeDS	Number	Long Integer	DataSetAZ
	DataSetSubjectID	Number	Long Integer	ClassificationConcept
	DataSetSubjectDS	Number	Long Integer	DataSetAZ
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
	SourceFileTypeID	Number	Long Integer	ClassificationConcept
	SourceFileTypeDS	Number	Long Integer	DataSetAZ
	PhysicalAddressTypeID	Number	Long Integer	ClassificationConcept
	PhysicalAddressTypeDS	Number	Long Integer	DataSetAZ
	PhysicalAddress	Text	255	
	IdentifierFieldName	Text	50	
	DataSetFieldName	Text	50	
	Comment	Memo		
GraphicLine (field definitions start on page 42)	GraObjID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Name	Text	255	
	Width	Number	Single	
	Scale	Number	Long Integer	
	Pattern	Text	50	
GraphicLineOrnamented (field definitions start on page 42)	GraObjID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Name	Text	255	
	SymbolID	Number	Long Integer	
	SymbolDS	Number	Long Integer	DataSetAZ
	Spacing	Number	Single	
	Scale	Number	Long Integer	
	Pattern	Text	50	
	Offset	Number	Single	
GraphicPattern (field definitions start on page 43)	GraObjID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Name	Text	255	
	PatternID	Number	Long Integer	PatternDefinition
	PatternDS	Number	Long Integer	DataSetAZ
	Scale	Number	Long Integer	
	Rotation	Number	Integer	
GraphicTextFormat (field definitions start on page 44)	GraObjID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	FontName	Text	25	
	Style	Text	16	
	Spacing	Number	Single	
	Alignment	Text	25	
	Size	Number	Single	
	Scale	Number	Long Integer	
	ColorID	Number	Long Integer	Color
	ColorDS	Number	Long Integer	DataSetAZ

Table Name	Field Name	Data Type	Field Size	Lookup Tables
HierarchyRelationship (field definitions start on page 30)	RelationshipID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	HierarchyTypeID	Number	Long Integer	ClassificationConcept
	HierarchyTypeDS	Number	Long Integer	DataSetAZ
	ParentID	Number	Long Integer	ClassificationConcept
	ParentDS	Number	Long Integer	DataSetAZ
	ChildID	Number	Long Integer	ClassificationConcept
	ChildDS	Number	Long Integer	DataSetAZ
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
MapLegend (field definitions start on page 45)	MapLegendID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	ConceptID	Number	Long Integer	ClassificationConcept
	ConceptDS	Number	Long Integer	DataSetAZ
	CartoObjID	Number	Long Integer	CartographicObject
	CartoObjDS	Number	Long Integer	DataSetAZ
	Sequence	Number	Long Integer	
	DispPriority	Number	Long Integer	
	DispVisibility	Boolean	Long Integer	
	ClassName	Text	64	
	ClassLabel	Text	255	
	ClassDesc	Memo		
	OriginDate	Date/Time		
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
MapViewDefinition (field definitions start on page 47)	MapViewID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Title	Text	255	
	Author	Text	255	
	PublicationDate	Date/Time		
	Description	Memo		
	DesignScale	Number	Long Integer	
	CatalogLinksDatsetID	Number	Long Integer	DataSetAZ
	ExtentID	Number	Long Integer	Extent table (Metadata)
	ExtentDS	Number	Long Integer	DataSetAZ
	ProjectionID	Number	Long Integer	ClassificationConcept
	ProjectionDS	Number	Long Integer	DataSetAZ
	MapHorizonDesc	Memo		
	MapHorizonID	Number	Long Integer	MapHorizon (Metadata)
	MapHorizonDS	Number	Long Integer	DataSetAZ
	MapLegendID	Number	Long Integer	MapLegend
	MapLegendDS	Number	Long Integer	DataSetAZ
	ViewSchemeTypeID	Number	Long Integer	ClassificationConcept
	ViewSchemeTypeDS	Number	Long Integer	DataSetAZ
	ClassSchemeID	Number	Long Integer	ClassificationConcept
	ClassSchemeDS	Number	Long Integer	DataSetAZ
	OriginDate	Date/Time		
MetadataRelationship (field definitions start on page 36)	MetadataRelationshipID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	RelTypeID	Number	Long Integer	ClassificationConcept
	RelTypeDS	Number	Long Integer	DataSetAZ
	FirstRoleID	Number	Long Integer	
	FirstRoleDS	Number	Long Integer	DataSetAZ
	SecondRoleID	Number	Long Integer	
	SecondRoleDS	Number	Long Integer	DataSetAZ
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
	Comment	Memo		

Table Name	Field Name	Data Type	Field Size	Lookup Tables
PatternDefinition (field definitions start on page 49)	GraObjID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	Name	Text	50	
	Description	Memo		
	Dimension	Number	Single	
	Density	Number	Long Integer	
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
SimpleRelationship (field definitions start on page 31)	RelationshipID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	RelTypeID	Number	Long Integer	ClassificationConcept
	RelTypeDS	Number	Long Integer	DataSetAZ
	FirstRoleID	Number	Long Integer	
	FirstRoleDS	Number	Long Integer	DataSetAZ
	SecondRoleID	Number	Long Integer	
	SecondRoleDS	Number	Long Integer	DataSetAZ
	Comment	Memo		
	TrackingID	Number	Long Integer	TrackingRecord
	TrackingDS	Number	Long Integer	DataSetAZ
TrackingRecord (field definitions start on page 37)	TrackingID	Number	Long Integer	
	DataSetID	Number	Long Integer	DataSetAZ
	TrackingRecordTypeID	Number	Long Integer	ClassificationConcept
	TrackingRecordTypeDS	Number	Long Integer	DataSetAZ
	Name	Text	255	
	LogDate	Date/Time		
	ActivityID	Number	Long Integer	Activities
	ActivityDS	Number	Long Integer	DataSetAZ
	DataProcMethodID	Number	Long Integer	ClassificationConcept
	DataProcMethodDS	Number	Long Integer	DataSetAZ
	Description	Memo		

Infrastructure Tables

Classification Concept Table

The **ClassificationConcept** table is a collection of terminology definitions – a term with a definition. These terms are used to classify other objects in all parts of the database. A unique identifier (ConceptID - DatasetID pair) identifies each concept. Thus the name of the concept may be changed without updating other links. The Arizona Geological Survey geologic information system has separate classification concept tables that are specific to different components of the system (e.g. Rock Unit Lexicon, Standard lithologic terms, etc.). Each of these classification concept tables has its own data set identifier defined in the DataSetAz table. (Return to Table 17.)

Database Table Fields

- **ConceptID**: Number, long integer. First part of the compound primary key. Uniquely identifies each classification object in the ClassificationConcept table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DatasetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the ClassificationConcept data set. Domain: Integers >0 and $<10^{16}$; Many concepts are from the system ClassificationConcept table, with DataSetID = 1, but for specific geologic datasets, classification concepts may be included from other classification concept datasets (e.g. standard lithology, standard minerals, formal stratigraphic units, etc.) with other Data-

SetID's. Some concepts may be specific to a particular geologic dataset and will have Data-SetID = the DataSetID for this particular ClassificationConcept table in the DataSetAZ table.

- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **Name**: Text, width 255. Provides a descriptive name for each classification concept. Domain: Free text.
- **ParentID**: Number, long integer. Represents the concept type. Semantically this is equivalent to the parent of the concept and the links between classification concepts and parent concepts define the classification concept hierarchy. This hierarchy is represented by the *HierarchyRelationship* table for use in general database queries. Inclusion of this attribute with each classification concept facilitates management of a single, simple tree hierarchy for classification concepts, but future development may allow a more complex concept hierarchy with multiple parent links. The ParentID is a foreign key that joins to the ConceptID field in this same table. Domain: Integers >0 and <10^{16}.
- **ParentDS**: Number, long integer. Specifies the data set that contains the data object identified by ConceptTypeID. Domain: 1 = the DataSetID for the ClassificationConcept table.
- **OriginDate**: Date/Time. Records when the record was created. This information provides more detailed information on the time that records were originally entered, supplementing the information in the associated *TrackingRecord* table.
- **Definition**: Text, width 255. Defines each classification concept. Domain: Free text.

Relationship Tables

Three sorts of relationship tables are used for representing semantic links between objects in the database (see Relationship Table Discussion, page 4). In the Arizona Geological Survey geologic information system, each component of the system (cartography, rock unit lexicon, standard lithology, geochronology…) has relationship tables specific to that sub domain. A particular geologic data set may include several different relationship tables of each of the types described below, each with its own DataSetID defined in the DataSetAz table..

Attributed Relationship Table

The **AttributedRelationship** table is used for representing relationships between objects in the database, i.e. for linking instances of two entities in which each relationship instance is assigned one or more attributes. This table is constructed to allow up to 5 attributes: CConf (Concept Confidence), CBasis (Concept Basis), StringValue (any string), Number Value (any number), or Attribute (a link to another object in the database). The RelTypeID link defines the semantics of the relationship links. Relationship constraints on RelType specify which attributes may have values and the domains of those values. Examples of attributed relationships include geologic classification of spatial objects, and various kinds of fractional analyses (e.g. chemical analysis, modal mineral analysis, grain size distribution). (Return to Table 17.)

Database Table Fields

- **RelationshipID**: Number, long integer. First part of the compound primary key. Uniquely identifies each record in the AttributedRelationship table. Although the compound key {RelTypeID, RelTypeDS, FirstRoleID, FirstRoleDS, SecondRoleID, SecondRoleDS} provides a unique key, the table has a standard {ObjectID, DatasetID} key to allow a relationship to play a role in another relationship using the standard relationship tables. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of compound primary key. Specifies the data set that contains the data object identified by RelationshipID. Domain: Single value, typically 28, the DataSetID for the AttributedRelationship table.
- **Sequence**: Number, long integer. Third part of compound primary key. Orders multiple instances of a single relationship link. Domain: Integers >0 and <10^{16}.
- **RelTypeID**: Number, long integer. Uniquely identifies the kind of relationship. This allows the AttributedRelationship table to represent any relationship that is defined by a ClassificationConcept. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: Integers >0 and <10^{16}.
- **RelTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by RelTypeID. Domain: Integers >0 and <10^{16}, typically 1, the DataSetID for the *ClassificationConcept* table.
- **FirstRoleID**: Number, long integer. First part of compound foreign key that identifies the object that fills the first role in the AttributedRelationship. Domain: Integers >0 and <10^{16}.
- **FirstRoleDS**: Number, long integer. Second part of compound foreign key that identifies the object that fills the first role in the AttributedRelationship. Specifies the data set that contains the data object identified by FirstRoleID. Domain: Integers >0 and <10^{16}, but must be a DataSetID that exists in the *DataSetAz* table.
- **SecondRoleID**: Number, long integer. First part of compound foreign key that identifies the object that fills the second role in the AttributedRelationship. Domain: Integers >0 and <10^{16}.
- **SecondRoleDS**: Number, long integer. Second part of compound foreign key that identifies the object that fills the second role in the AttributedRelationship. Specifies the data set that contains the data object identified by SecondRoleID. Domain: Integers >0 and <10^{16}, but must be a DataSetID that exists in the *DataSetAz* table.
- **CConf**: Character, width 16. Assigns a qualitative confidence level to the relationship. Domain: 'low', 'standard', or 'high'.
- **CBasis**: Text, length 255. Indicates the basis for assigning the relationship. Ideally this and CConf should be ClassificationConcept terms, but a text field is implemented here as an interim measure to get a better feeling for what sort of terms are required to assign values for confidence and basis.
- **StringValue**: Text, length 255. Allows assignment of a text attribute value for the relationship. An example is a text string that is displayed at a point location as cartographic annotation, in which case the relationship links the point with a TextFormat cartographic object. Domain: Free text.
- **NumberValue**: Number, single-precision real. Allows assignment of a numeric attribute value for the relationship. Examples include assignment of a fractional abundance for a component in a fractional analysis, or a symbol rotation value for a point location-structure symbol link. Domain: Real numbers, >0 and <10^8.

- **AttributeObjID**: Number, long integer. First part of the compound foreign key that identifies an attribute object associated with this AttributedRelationship instance. Domain: Integers >0 and <10^{16}.
- **AttributeObjDS**: Number, long integer. Second part of the compound foreign key that identifies an attribute object associated with this AttributedRelationship instance. Specifies the data set that contains the data object identified by AttributeObjID. Domain: Integers >0 and <10^{16}, but must be a DataSetID that exists in the *DataSetAz* table.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value, typically 18, the DataSetID for the *TrackingRecord* table.
- **OriginDate**: Date/Time. Records when the record was created. This information provides more detailed information on the time that records were originally entered, supplementing the information in the associated *TrackingRecord* table. Format: '*mm/dd/yy*'.
- **Comment**: Memo. Additional information about a relationship instance. Domain: Free text.

Hierarchy Relationship Table

The **HierarchyRelationship** table represents parent-child relationships. Multiple tree hierarchies may be represented, each identified by a HierarchyType – a classification concept that defines the nature of the hierarchy. For implementation simplicity, a hierarchy is represented in this table as a set of links between each parent and all the child objects beneath it in the hierarchy tree (its transitive closure). The depth of any child object in the tree is determined by the number of parent objects linked to it. This representation makes response to queries that require all kinds (sub types) of a thing (e.g. 'all spatial objects', 'all map units') simple to execute. Currently, each child has only one parent. (Return to Table 17.)

Database Table Fields

- **RelationshipID**: Number, long integer. First part of the compound primary key. Uniquely identifies each record in the HierarchyRelationship table. Although the compound key {HierarchyTypeID, HierarchyTypeDS, ParentID, ParentDS, ChildID, ChildDS} provides a unique key, the table has a standard {ObjectID, DatasetID} key to allow a relationship to play a role in another relationship using the standard relationship tables. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Specifies the data set that contains the data object identified by RelationshipID. Domain: Single value (typically), the DataSetID for the HierarchyRelationship table.
- **HierarchyTypeID**: Number, long integer. Uniquely identifies the kind of hierarchy. This allows the HierarchyRelationship table to represent multiple concept hierarchies as well as other unrelated hierarchies. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: 2478 = ClassificationConceptHierarchy; only one hierarchy is currently represented.
- **HierarchyTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by HierarchyTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.

- **ParentID**: Number, long integer. First part of the compound foreign key that identifies the parent object in the parent-child (IsA) relationship. Because one classification concept hierarchy is the only hierarchy in this database, this field is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: Integers >0 and <10^{16}.
- **ParentDS**: Number, long integer. Second part of the compound foreign key that identifies the parent object in the parent-child (IsA) relationship. Specifies the data set that contains the data object identified by ParentID. Because one classification concept hierarchy is the only hierarchy in this database, this field has only one value. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **ChildID**: Number, long integer. First part of the compound foreign key that identifies the child object in the parent-child (IsA) relationship. Because one classification concept hierarchy is the only hierarchy in this database, this field is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: Integers >0 and <10^{16}.
- **ChildDS**: Number, long integer. Second part of the compound foreign key that identifies the child object in the parent-child (IsA) relationship. Specifies the data set that contains the data object identified by ChildID. Because one classification concept hierarchy is the only hierarchy in this database, this field has only one value. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}, no duplicates.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.

Simple Relationship Table

This table is used to represent relationships that link instances of any two objects in which no uncertainty is involved and the relationship has no attributes. Examples include aggregations of parts, and linking SpatialObjects to CartographicObjects for symbolization. (Return to Table 17.)

Database Table Fields

- **RelationshipID**: Number, long integer. First part of the compound primary key. Uniquely identifies each record in the SimpleRelationship table. Although the compound key {RelTypeID, RelTypeDS, FirstRoleID, FirstRoleDS, SecondRoleID, SecondRoleDS} provides a unique key, the table has a standard {ObjectID, DatasetID} key to allow a relationship to play a role in another relationship using the standard relationship tables. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of compound primary key. Specifies the data set that contains the data object identified by RelationshipID. Domain: Single value, typically 33, the DataSetID for the SimpleRelationship table.
- **Sequence**: Number, long integer. Third part of compound primary key. Orders multiple instances of a single relationship link. Domain: Integers >0 and <10^{16}.
- **RelTypeID**: Number, long integer. Uniquely identifies the kind of relationship. This allows the SimpleRelationship table to represent any relationship that is defined by a Classification-Concept. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: Integers >0 and <10^{16}.

- **RelTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by RelTypeID. Domain: Integers >0 and $<10^{16}$, typically is 1, the DataSetID for the *ClassificationConcept* table.
- **FirstRoleID**: Number, long integer. First part of compound foreign key that identifies the object that fills the first role in the SimpleRelationship. Domain: Integers >0 and $<10^{16}$.
- **FirstRoleDS**: Number, long integer. Second part of compound foreign key that identifies the object that fills the first role in the SimpleRelationship. Specifies the data set that contains the data object identified by FirstRoleID. Domain: Integers >0 and $<10^{16}$, but must be a DataSetID that exists in the *DataSetAz* table.
- **SecondRoleID**: Number, long integer. First part of compound foreign key that identifies the object that fills the second role in the SimpleRelationship. Domain: Integers >0 and $<10^{16}$.
- **SecondRoleDS**: Number, long integer. Second part of compound foreign key that identifies the object that fills the second role in the SimpleRelationship. Specifies the data set that contains the data object identified by SecondRoleID. Domain: Integers >0 and $<10^{16}$, but must be a DataSetID that exists in the *DataSetAz* table.
- **Comment**: Memo. Contains any additional information about a relationship instance. Domain: Free text.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value, typically 18, the DataSetID for the *TrackingRecord* table.

MetaData Tables

Activities Table

The **Activities** table is a link to an activity responsible for update of, or addition to, the database. An activity is a particular person, working for a particular organization, under the auspices of a particular project. (Return to Table 17.)

<u>Database Table Fields</u>
- **ActivityID**: Number, long integer. First part of the compound primary key. Uniquely identifies each activity in the Activities table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DatasetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the Activities data set. Domain: 2 = the DataSetID for the Activities table.
- **Name**: Text, width 255. Provides a unique name identifier for each activity. This is the string the is displayed in combo boxes on data entry forms. Domain: Free text.
- **PersonOrgID**: Number, long integer. Uniquely identifies the person and the organization that are associated with each activity. It is a foreign key that joins to the PersonOrgID field of the PersonOrg table (the PersonOrg table is described in the report on metadata data structure implementation [in preparation]). Domain: See Table 18.

Table 18. Example PersonOrg codes used in the Activities Table.

PersonOrgID	DataSetID	PersonName	Organization
1	15	Dr. Stephen M. Richard	Arizona Geological Survey

PersonOrgID	DataSetID	PersonName	Organization
2	15	Mr. Tim R. Orr	Arizona Geological Survey
4	15	Mr. Null N Null	None
5	15	Mr. Jason . Brander	Bureau of Land Management
11	15	Dr. Philip A. Pearthree	Arizona Geological Survey
12	15	Ms. Ann . Youberg	Arizona Geological Survey
13	15	Mr. Ray C. Harris	Arizona Geological Survey

- **PersonOrgDS**: Number, long integer. Specifies the data set that contains the data object identified by PersonOrgID. Domain: Single value (typically), the DataSetID for the PersonOrg link table.
- **ProjectID**: Number, long integer. Identifies the project associated with each activity. It is a foreign key that joins to the ProjectID field of the Projects table (the Projects table is described in the report on metadata data structure implementation [in preparation]). Domain: See Table 19.

Table 19. Example ProjectID codes used in the Activities Table.

ProjectID	ProjectDS	Project_title	Prj_comment
1	17	Arizona NADM implementation development	Develop NADM 5.2 implementation and use for new geologic map of Arizona database
2	17	DI-8 Version 3 database development	Construct NADM-compliant database with geologic data compiled for Map 35.
3	17	Phoenix N, East Half Database development	Activities related to development of databases for east half of Phoenix North 30 by 60 minute quadrangle
4	17	Null	No project assigned
5	17	Digitize Geologic Map of Arizona, using MOSS	get 1:1,000,000 scale geologic map in digital form to assist management decisions
8	17	Edit Map 26 Data to release as DI8 V.1	Get original MOSS version converted to ARC and into a form that could be released
10	17	Statemap 1999, Waterman Peak 1:24000 quad	Generate Statemap deliverable, geologic map of Waterman Peak quad
11	17	TheodoreRoosevelt100KGISV2	Generate final, complete 100K geology GIS for Theodore Roosevelt Lake 100K quad.
13	17	Statemap 1999 Surficial	Surficial Geologic maps of Avra Valley and Green Valley areas
16	17	AZ Geologic Map Index database conversion	conversion of DI-9 (AZ Map Index) to new AZ_NADM datastructure
17	17	Statemap2000PhxDatabases	Project to complete 1:24k GIS databases for quads in phoenix area; includes preliminary development work on Waterman-Roskruge database also funded by this project
20	17	Statemap 1999, Roskruge/Waterman Digital Data	Digital geologic information for the Roskruge and Waterman Mountains

- **ProjectDS**: Number, long integer. Specifies the data set that contains the data object identified by ProjectID. Domain: Single value, typically 17 the DataSetID for the Projects table.
- **Comment**: Memo. Contains descriptive text about each activity, including the name of the person who conducted the activity, their employing organization, and the project they were working on. Domain: Free text.

Bibliographic Citations Table (AZgeoBibLinkTable)

The **AZgeoBibLinkTable** table is derived from the Arizona Geological Survey bibliographic data base (AzGeoBib, Trapp et al. [1996], DataSetID = 4 in the *DataSetAZ* table), and provides a

mechanism for citing published literature. In this database citations are related to tracking records through the *MetadataRelationship* table. This derivative table is included to replace links to the full AzGeoBib database. (Return to Table 17.)

<u>Database Table Fields</u>

- **AzGeoBibID**: Number, long integer. First part of the compound primary key. Uniquely identifies each citation in the AZgeoBibLinkTable table. Domain: Integers >0 and <10^{16}, no duplicates. The identifiers used here are the same as identifiers for the citation in AzGeoBib.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the AzGeoBib data set in the Arizona Geological Survey namespace. Domain: 4 = the DataSetID for the AzGeoBibLink table.
- **Authorship**: Text, length 255. Author of cited publication. Format: '*Last Name, First Initial.Middle Initial.*'; Author names separated by comma, with '*, and* ' before last author.
- **Title**: Text, length 255. Title of cited publication.
- **Citation**: Text, length 255. A text citation for the location of publication.
- **Year**: Date/Time. Year of publication for citation. Format: '*yyyy*'.

DataSetAZ Table

The **DataSetAZ** table is a catalog of the data sets within the Arizona Geological Survey namespace. A data set is any collection of data that is held in an individual file or table. Examples include individual ArcInfo coverages, ESRI shapefiles, tables in Microsoft Access databases, dBase tables in individual .dbf files, and files containing images (e.g. tiff, jpeg). The contents of the **DataSetAZ** table define the 'Arizona Geological Survey' namespace. This table is analogous to an Open GIS Consortium 'Catalog'. (Return to Table 17.)

<u>Database Table Fields</u>

- **DataSetID**: Number, long integer. First part of the compound primary key. Uniquely identifies each data set in the DataSetAZ table. Domain: Integers >0 and <10^{16}, no duplicates.
- **NameSpace**: Text, width 50. Second part of the compound primary key. Identifies the agency or organization that owns or maintains the data set. Domain: 'Arizona Geological Survey'.
- **NameSpaceID**: Number, long integer. Classifies the NameSpace for each data set record in the DataSetAZ data set. There is a 1:1 correspondence between values in this field and values in the NameSpace field, i.e. they are redundant. Both a string value and an numeric value are included to facilitate implementation using the convention adopted for this database system that a data object within a particular namespace is identified by a compound primary key consisting of 2 long integers. NameSpaceID is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: 2541 = the ConceptID for the "Arizona Geological Survey" namespace.
- **NameSpaceDS**: Number, long integer. Specifies the data set that contains the data object identified by NameSpaceID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **DataSetName**: Text, width 255. Uniquely identifies each data set. Domain: Free text.
- **DataSetTypeID**: Number, long integer. Classifies each data set according to a data set type from the *ClassificationConcept* table. The data set type identifies the physical data structure of the data set (e.g. ArcInfo coverage, Microsoft Access table….). It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 20.

Table 20. Example data set type codes used in the DataSetAZ table.

DataSetTypeID	Name
2744	Classification/Description/Definition Dataset
2761	Generic Attributed Relationship Dataset
2762	Description Container Dataset
2794	Geographic Dataset

- **DataSetTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by DataSetTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **DataSetSubjectID**: Number, long integer. Classifies each data set according to a subject classification term. The subject classification term identifies the domain of interest for the data in the data set. In future implementations, the data set subject will be used for error and consistency checking. A more complete key word index for data sets would need to be implemented through a correlation table allowing a many-to-many join between data sets and subjects. The DataSetSubjectID is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 21.

Table 21. Example data set subject codes used in the DataSetAZ table.

DataSetSubjectID	Name
2759	NADM Implementation Infrastructure
2767	AZ Cordlink base table
3306	Graphic Definition Tables
3336	Roskruge and Waterman Mountains and western Avra Valley

- **DataSetSubjectDS**: Number, long integer. Specifies the data set that contains the data object identified by DataSetSubjectID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **SourceFileTypeID**: Number, long integer. Classifies each data set by its physical file type or format. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 22.

Table 22. Example source file type codes used in the DataSetAZ table.

SourceFileTypeID	Name
2542	Microsoft Access Database Table
2543	dBase Table
2544	ESRI coverage, point
2545	ESRI coverage, arc
2547	ESRI coverage, polygon
2548	AV shapefile, point
2549	AV shapefile, line

- **SourceFileTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by SourceFileTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **PhysicalAddressTypeID**: Number, long integer. Classifies the type of physical address that records where each data set is stored. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: 2726 = DOS-style path name; 2727 = Microsoft Network file path name.
- **PhysicalAddressTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by PhysicalAddressTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **PhysicalAddress**: Text, width 255. Identifies the actual physical location of the data set. Domain: Free text restricted to formats defined by PhysicalAddressTypeID.
- **IdentifierFieldName**: Text, width 50. Records the name of the field in the DataSet that contains the identifier component of the compound unique identifier for each record. Domain: Restricted to the indexed, primary key field names. This is typically the first field in each data set, and the field name is typically the table name or an object type name with "ID" appended. Identifier field names always end with the string "ID".
- **DataSetFieldName**: Text, width 50. Records the name of the field in the DataSet that contains the data set component of the compound unique identifier for each record. Domain: 'DataSetID', 'Namespace'; typically the second field in each data set.
- **Comment**: Memo. Provides additional descriptive information about each data set. Domain: Free text.

Metadata Relationship Table

The **MetadataRelationship** table is a relationship table that provides a general mechanism for semantic links between metadata instances. A RelType (relationship type) identifier links to a ClassificationConcept that defines the semantics of the relationship. Constraints on kinds of objects that may play the first and second role, and the number of fillers allowed for each role, will eventually be specified by a ValidRelationshipConstraint data structure, but this part of the database is currently being revised and is not implemented here. In this database, this table is used to implement a many-to-many join between tracking records and citations. Other applications in a more developed database would include relationships like project hierarchy (large project with subprojects), organization successor (when an organization changes name), organization aggregation (to represent individual departments as part of a larger organization), StartDate and EndDate links between Person-Organization affiliations and a metadata dates entity, PersonOrg-ContactInformation links to allow multiple contact addresses and types (phone, internet, surface mail…), and Object-LogEntries to allow multiple tracking records to be related to any object, to track revisions, comments, etc. (Return to <u>Table 17</u>.)

Database Table Fields

- **MetadataRelationshipID**: Number, long integer. First part of the compound primary key. Uniquely identifies each record in the MetadataRelationship table. Although the compound key {RelTypeID, RelTypeDS, FirstRoleID, FirstRoleDS, SecondRoleID, SecondRoleDS} provides a unique key, the table has a standard {ObjectID, DatasetID} key to allow a rela-

tionship to play a role in another relationship using the standard relationship tables. Domain: Integers >0 and $<10^{16}$, no duplicates.

- **DataSetID**: Number, long integer. Second part of the compound primary key. Specifies the data set that contains the data object identified by MetadataRelationshipID. Domain: Single value (typically), the DataSetID for the MetadataRelationship table.
- **RelTypeID**: Number, long integer. Uniquely identifies the kind of relationship. This allows the MetadataRelationship table to represent any kind of relationship. The RelType defines the semantics of the relationship. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: 2535 = TrackingRecord-Citation link. The only relationship currently represented, is a link between records in the *TrackingRecord* table and citations in the *AZgeoBibLinkTable* table, allowing a many-to-many relationship between citations and tracking records.
- **RelTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by RelTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **FirstRoleID**: Number, long integer. First part of compound foreign key that identifies the object in the first role of the metadata relationship. Because only the TrackingRecord-Citation relationship is represented in this database, this field is a foreign key that joins to the TrackingRecordID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **FirstRoleDS**: Number, long integer. Second part of compound foreign key that identifies the object in the first role of the metadata relationship. Specifies the data set that contains the data object identified by FirstRoleID. Because only the TrackingRecord-Citation relationship is represented in this database, this field has only one value. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **SecondRoleID**: Number, long integer. First part of compound foreign key that identifies the object in the second role of the metadata relationship. Because only the TrackingRecord-Citation relationship is represented in this database, this field is a foreign key that joins to the RefNum field of the *AZgeoBibLinkTable* table. Domain: Integers >0 and $<10^{16}$.
- **SecondRoleDS**: Number, long integer. Second part of compound foreign key that identifies the object in the second role of the metadata relationship. Specifies the data set that contains the data object identified by SecondRoleID. Because only the TrackingRecord-Citation relationship is represented in this database, this field has only one value. Domain: 4 = the DataSetID for the *AZgeoBibLinkTable* table.

Tracking Record Table

The **TrackingRecord** table keeps a record of the intellectual and physical sources for objects and data by defining links to tables that describe the processes and activities through which data was created. (Return to <u>Table 17</u>.)

<u>Database Table Fields</u>

- **TrackingID**: Number, long integer. First part of the compound primary key. Uniquely identifies each record in the TrackingRecord data set. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the TrackingRecord data set. Domain: Single value (typically), the DataSetID for this table.

- **TrackingRecordTypeID**: Number, long integer. Uniquely identifies the type of origin track-ing record. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 23.

Table 23. Tracking Record Type codes used in the TrackingRecord table.

ConceptID	Name	Definition
2534	Origin Tracking Record	Tracking record that records the origin of a data object or data set
2742	Log Entry Tracking Record	Tracking record type for tracking records that add information about a data entity
2765	Termination Tracking Record	Tracking record that indicates a data object has been superceded by a newer object.
3210	Feature-level Origin Tracking Record	Use as supertype to group tracking records that document origin of individual fea-ture records in data sets.
3211	Feature-Level Tracking for DI8 V3	Supertype to group feature tracking records for Geologic map of Arizona Database, v3
3228	Feature-Level Tracking for In-frastructure Objects	Tracking record type for records that track data objects in the infrastructure tables
3231	Dataset Origin Tracking	Tracking records that record facts about the origin of a Dataset, and are inherited by contents of data set unless feature-level tracking is included for data set

- **TrackingRecordTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingRecordTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **Name**: Text, width 255. Uniquely identifies each origin tracking record and is included for simplification purposes. Domain: Free text.
- **LogDate**: Date/Time. Records when an entry was created. Format: '*mm/dd/yy*'.
- **ActivityID**: Number, long integer. A foreign key that links ActivityID field of the *Activities* table. Domain: See Table 24.

Table 24. Example Activity ID codes used in the TrackingRecord table.

ActivityID	Name	Comment
1	SMRDataModelDevelopment	Stephen M. Richard, Arizona Geological Survey, Arizona NADM implementation development
2	SMR-DI8V3DevelopmentActivity	Stephen M. Richard, Arizona Geological Survey, DI-8 Version 3 da-tabase development
4	Null	No Activity assigned; Null N Null, None, Null
5	BLMMOSSdigitizeMap26	BLM activity to produce MOSS version of Reynolds, 1988, AZGS Map26; Jason . Brander, Bureau of Land Management, Digitize Geologic Map of Arizona, using MOSS
8	SMRDI8V1	Convert MOSS data to ARC, adjust to match ALRIS state outline, minor editing to correct obvious linework problems, edit faults to match contacts better, reclassify some polygons; Stephen M. Rich-ard, Arizona Geological Survey, Edit Map 26 Data to release a
38	TRODataModelDevelopment	Tim R. Orr, Arizona Geological Survey, Arizona NADM implemen-tation development
39	AY-GreenValleyUnitAssignment	assignment of map unit names to geologic polygons; Ann Youberg, Arizona Geological Survey, Statemap 1999 Surficial
40	TRORoskrugeWatermanDatabaseDevelopment	project specific database construction; Tim R. Orr, Arizona Geo-logical Survey, Statemap2000PhxDatabases
45	RCHRoskrugeWatermanDI	Digitizing, editing, and attribution of geologic information by Ray Harris from data collected for Statemap 1999 contract; Ray C. Har-ris, Arizona Geological Survey, Statemap 1999, Rosk-ruge/Waterman Digital Data
82	PAPRoskrugeWatermanDI	DI database contributions by Phil Pearthree

- **ActivityDS**: Number, long integer. Specifies the data set that contains the data object identified by ActivityID. Domain: 2 = the DataSetID for the *Activities* table.
- **DataProcMethodID**: Number, long integer. It is a foreign key that links to a data processing name and definition in the *ClassificationConcept* table. A complete data processing object define the steps in developing a particular data item (digitized spatial feature, record in a data table). In this database the processing steps are not described at a feature level. Domain: 2748 = MS Access Database Construction; 2764 = no processing.
- **DataProcMethodDS**: Number, long integer. Specifies the data set that contains the data object identified by DataProcMethodID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **Description**: Memo. Contains a description of the people and processes that define each tracking record. Domain: Free text.

Cartographic Tables

Cartographic Object Table

The **CartographicObject** table is an implementation-independent representation of symbols used to display points, lines, polygons, and text on a map visualization. This is done by defining links to tables that provide implementation-dependent descriptions of graphical objects used for symbolization. Graphical object tables in this database are designed to describe symbology for ArcView 3.2 running in a Microsoft Windows environment. Individual cartographic objects may consist of several graphical objects stacked according to the sequence attribute in the table, with the lowest sequence symbol overlain by subsequent symbols in the sequence. (Return to Table 17.)

Database Table Fields
- **CartoObjID**: Number, long integer. First part of the compound primary key. Uniquely identifies each object in the CartographicObject table. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DatasetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the CartographicObject data set. Domain: Single value (typically), the DataSetID for the CartographicObject table.
- **Sequence**: Number, integer. Third part of the compound primary key. Corresponds to the layer order in which graphical elements are created. For example, an ornamented line, such as a line with queries, would be created using two layers. The first layer, the line itself, would have a sequence value of 1, while the second layer, the query symbol, would have a sequence value of 2. Domain: Integers >0 and $<10^{8}$.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.
- **Name**: Text, width 255. Uniquely identifies and describes each cartographic object and is included for intelligibility. Domain: Free text.

- **CartoObjTypeID**: Number, long integer. Classifies the graphical element type. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: See Table 25.

Table 25. Example cartographic object type codes used in the CartographicObject table.

CartoObjTypeID	Name
1957	Cartographic Object -- point
1958	Cartographic Object -- line
2392	Point symbol from font
2393	Annotation at point
2408	Fill, solid
2409	Fill, pattern
3019	Line, solid
3020	Line symbol, dash-dot pattern
3021	Line symbol, ornamented

- **CartoObjTypeDS**: Number, long integer. Specifies the data set that contains the data object identified by CartoObjTypeID. Domain: 1 = the DataSetID for the *ClassificationConcept* table.
- **GraObjID**: Number, long integer. Corresponds to a specific graphical element. It is a foreign key that joins to the GraObjID field of either the GraphicLine table, the *GraphicLineOrnamented* table, the *GraphicPattern* table, or the *GraphicTextFormat* table. Domain: See Table 26.

Table 26. Example graphic object codes used in the CartographicObject table.

GraObjID	GraObjDS	Name
11	26	thick line (1.5 pt), solid
3	26	medium thin line (0.35 pt), standard short dash
4	26	medium thin line (0.35 pt), solid
7	26	medium line (0.5 pt), dash-dot
10	26	thick line (1.5 pt), standard medium dash
20	26	medium line (0.75 pt), dotted
14	411	cross hatch, lines at 30° 90° and 150°, separation = 4 pt.
33	411	vertical hatch, separation = 1 pt.
13	411	cross hatch, lines at 0° and 90°, separation = 4 pt.
0	411	Null Pattern
1	411	Solid Color Fill; scale invariant
11	411	cross hatch, lines at 0° and 90°, separation = 3 pt.
27	411	hatch, 60°, separation = 1 pt.
32	411	horizontal hatch, separation = 3 pt.
28	411	hatch, 60°, separation = 1.75 pt.
16	411	cross hatch, lines at 45° and 135°, separation = 2.5 pt.
26	411	hatch, 45°, separation = 4 pt.
23	411	hatch, 135°, separation = 3.25 pt.
5	412	Open circle
1	412	Solid triangle
2	412	Query

GraObjID	GraObjDS	Name
3	412	Perpendicular hash
4	412	X pattern
6	412	Alternating slash
6	420	Arial, Normal, Spacing: 1, JUST_LEFT, 11 point, PMS-Black
3	420	Arial, Normal, Spacing: 1, JUST_LEFT, 8 point, PMS-Black
12	420	AzGSArial, Normal, Spacing: 1, JUST_LEFT, 7 point, PMS-Black
19	420	Arial, Bold, Spacing: 1, JUST_LEFT, 7 point, PMS-Black
32	420	Arial, Italic, Spacing: 1, JUST_LEFT, 12 point, PMS-Black

- **GraObjDS**: Number, long integer. Specifies the data set that contains the data object identified by GraObjID. Domain: 26 = the DataSetID for the *GraphicLine* table; 411 = the DataSetID for the *GraphicPattern* table; 412 = the DataSetID for the *GraphicLineOrnamentated* table; 420 = the DataSetID for the *GraphicTextFormat* table.
- **ColorID**: Number, long integer. Represents a specific color. It is a foreign key that joins to the GraObjID field of the *Color* table. Domain: Integers >0 and <10^{16}.
- **ColorDS**: Number, long integer. Specifies the data set that contains the data object identified by ColorID. Domain: Single value (typically), the DataSetID for the *Color* table.
- **OriginDate**: Date/Time. Records when the record was created. This information provides more detailed information on the time that records were originally entered, supplementing the information in the associated *TrackingRecord* table. Format: '*mm/dd/yy*'.
- **Comment**: Memo. Provides additional descriptive information about a record. Domain: Free text.

Color Table

The **Color** table defines RGB values for colors used in the default symbolization. The RGB values were derived from the Pantone® [Pantone®, Inc., 1991] color swatch library in Adobe Illustrator®, and represent red, green, and blue values that approximate Pantone® colors for onscreen viewing. (Return to Table 17.)

Database Table Fields
- **ColorID**: Number, long integer. First part of the compound primary key. Uniquely identifies each color in the Color table. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the Color data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **Name**: Text, width 255. Uniquely identifies each color by Pantone® name (PMS-466), or by a string concatenated from a color description and the RGB values for that color (Blue (R39,G146,B182)). Domain: Pantone color names from the Pantone® [Pantone®, Inc., 1991] color swatch library, or free text.
- **CMYK**: Text, width 25. This optional field contains a string concatenated from the CMYK color values for a particular color. Domain: A sequence of four numbers each consisting of three integers ranging from 0 to 255.
- **RGB**: Text, width 25. This optional field contains a string concatenated from the RGB color values for a particular color. Domain: A sequence of three numbers each consisting of three integers ranging from 0 to 255.

- **R**: Number, integer. Represents the red color intensity for the screen display of a particular color. Domain: An integer from 0 to 255.
- **G**: Number, integer. Represents the green color intensity for the screen display of a particular color. Domain: An integer from 0 to 255.
- **B**: Number, integer. Represents the blue color intensity for the screen display of a particular color. Domain: An integer from 0 to 255.

Graphic Line Table

The **GraphicLine** table contains descriptions of the graphical elements used to symbolize lines. This description is implementation dependent, and is based on attributes used to define line symbols in ArcView 3.2 and Adobe Illustrator. (Return to .)

Database Table Fields
- **GraObjID**: Number, long integer. First part of the compound primary key. Uniquely identifies each line in the GraphicLine data set. Domain: Integers >0 and <10^{16}, no duplicates.
- **DatasetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the GraphicLine data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **Name**: Text, width 255. Uniquely identifies each type of line and is included for simplification purposes. Domain: Free text.
- **Width**: Number, single-precision real. Specifies line thickness in millimeters. Domain: Real numbers, >0 and <10^{8}.
- **Scale**: Number, long integer. Represents the denominator of the map scale at which the graphical specifications of a line are valid. For example, a map scale of 1:12,000 would be recorded as '12000'. Domain: Integers >0 and <10^{16}.
- **Pattern**: Text, width 50. A sequence of numbers, as a space- or comma-delimited string, that specifies the alternating solid (on) and empty (off) length of line segments, in millimeters, starting with the 'on' value, that are repeated to create each line pattern. For example, a string shown as '0.36 0.71 1.07 0.71' defines a repeating line pattern created by a 0.36 mm line segment, followed by a 0.71 mm space, followed by a 1.07 mm line segment, and followed by another 0.71 mm space. This approximates a line with a dot-dash pattern. Domain: A string composed of a series of real numbers separated by spaces.

Graphic Line with Ornamentation Table

The **GraphicLineOrnamented** table contains descriptions of symbols used to create ornamented lines (e.g. queries dashed lines, thrust faults). This description is implementation dependent, and is based on attributes used to define line symbols in ArcView 3.2 running in a Microsoft Windows environment. (Return to .)

Database Table Fields
- **GraObjID**: Number, long integer. First part of the compound primary key. Uniquely identifies each style of ornamented line in the GraphicLineOrnamented data set. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the GraphicLineOrnamented data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.

- **Name**: Text, width 255. Uniquely identifies each type of line ornamentation symbol and is included for simplification purposes. Domain: Free text.
- **SymbolID**: Number, long integer. Identifies the graphical object used as an ornament along a decorated line (the SymbolID field is not implemented here but is included for future compatibility). In the implementation environment for this table, these symbols are characters from a font, and SymbolID would identify the index of a symbol in the font data set specified by SymbolDS. Domain: 0
- **SymbolDS**: Number, long integer. Specifies the data set that contains the data object identified by SymbolID (the SymbolDS field is not implemented here but is included for future compatibility). In the implementation environment for this table, SymbolDS would identify a font data set. Domain: 0.
- **Spacing**: Number, single-precision real. Specifies spacing, in millimeters, between the centers of adjacent line ornamentation symbols. Domain: Real numbers, >0 and <10^8.
- **Scale**: Number, long integer. Represents the denominator of the map scale at which the graphical specifications of line ornamentation is valid. For example, a map scale of 1:12,000 would be recorded as '12000'. Domain: Integers >0 and <10^{16}.
- **Pattern**: Text, width 50. Records a stream of integer draw or skip intervals. For example, '12' means draw one and skip two. A leading zero, such as in '021', means skip two and draw one. The interval size is dependent on ornament size (the Pattern field is not implemented here but is included for future compatibility). Domain: Free text composed of a single string of integers.
- **Offset**: Number, single-precision real. Specifies the offset of an ornamentation symbol perpendicular to the line with which it is associated (the Offset field is not implemented here but is included for future compatibility). Domain: Real numbers, >0 and <10^8.

Graphic Pattern Table

The **GraphicPattern** table defines the graphical specifications for polygon fills in the default symbolization. (Return to Table 17.)

Database Table Fields
- **GraObjID**: Number, long integer. First part of the compound primary key. Uniquely identifies each polygon fill in the GraphicPattern data set. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the GraphicPattern data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **Name**: Text, width 255. Uniquely identifies each type of polygon fill pattern and is included for simplification purposes. Domain: Free text.
- **PatternID**: Number, long integer. Corresponds to the pattern used to create each type of polygon fill pattern. It is a foreign key that joins to the GraObjID field of the *PatternDefinition* table. (The *PatternDefinition* table is not included with this database, but the PatternID field is included for future compatibility.) Domain: 0 (solid fill); -1 (not defined).
- **PatternDS**: Number, long integer. Identifies the data set that contains the data object identified by PatternID. Domain: Single value (typically), the DataSetID for the *PatternDefinition* table.

- **Scale**: Number, long integer. Represents the denominator of the map scale at which the graphical specifications of each polygon fill pattern is valid. For example, a map scale of 1:12,000 would be recorded as '12000'. Domain: Integers >0 and <10^{16}.
- **Rotation**: Number, integer. Represents the rotation angle, in degrees, of the graphical pattern for a particular polygon fill. The magnitude of the angle is measured clockwise starting from a compass azimuth of 0°. Domain: 0 to ±360.

Graphic Text Format Table

The **GraphicTextFormat** table defines the font specifications (type, style, size...) for cartographic text used in the default symbolization. This description is implementation dependent, and is based on attributes used to define text formatting in ArcView 3.2 running in a Microsoft Windows environment. (Return to Table 17.)

Database Table Fields

- **GraObjID**: Number, long integer. First part of the compound primary key. Uniquely identifies each type of text in the GraphicTextFormat data set. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the GraphicTextFormat data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **FontName**: Text, width 25. Represents the name of the font used to symbolize a text object. Domain: Any valid font (requires that custom, or non-standard fonts, be distributed with the data sets they accompany). Typically is restricted to 'AZGSArial' and 'Arial'.
- **Style**: Text, width 16. Identifies the style of the font used to symbolize a text object. Domain: 'Normal'; 'Bold'; 'Italic'; 'Bold Italic'.
- **Spacing**: Number, single-precision real. Specifies the vertical spacing between lines of text. Domain: Real numbers, >0 and <10^8.
- **Alignment**: Text, width 25. An ArcView-generated text string that specifies line justification. Domain: 'TEXTCOMPOSER_JUST_RIGHT', 'TEXTCOMPOSER_JUST_LEFT', 'TEXTCOMPOSER_JUST_CENTER'.
- **Size**: Number, single-precision real. Defines the font size, in points, of a text object. Domain: Real numbers, >0 and <10^8.
- **Scale**: Number, long integer. Represents the denominator of the map scale at which the graphical specifications of a text format is valid. For example, a map scale of 1:12,000 would be recorded as '12000'. Domain: Integers >0 and <10^{16}.
- **ColorID**: Number, long integer. Represents a specific color. It is a foreign key that joins to the ColorID field of the *Color* table. Domain: Integers >0 and <10^{16},.
- **ColorDS**: Number, long integer. Specifies the data set that contains the data object identified by ColorID. Domain: Single value (typically), the DataSetID for the *Color* table.

Map Legend Table

The **MapLegend** table contains relationship links between a ClassificationConcept and an implementation-independent CartographicObject used to symbolize objects belonging to the class. A particular map legend may contain only one instance of each symbol included, but different symbols may correspond to the same classification concept (e.g. symbols for horizontal, inclined, vertical, and overturned planar bedding). The **MapLegend** table assigns a Name, Label, and Description

44

for objects of that class which are used to generate the explanation to display on the map. The Sequence field orders items in the legend. Legend items may be present that have no corresponding classification concept; these typically act as headings. The compound key for the **MapLegend** table is the tuple {MapLegendID, DataSetID, Sequence}. Hierarchy in the legend is represented by a *HierarchyRelationship* with RelTypeID = MapLegendID. (Return to Table 17.)

Database Table Fields

- **MapLegendID**: Number, long integer. First part of the compound primary key. Uniquely identifies each type of text in the GraphicTextFormat data set. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the MapViewDefinition data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set. Typically will be 410, the DataSetID for the standard MapViewDefinition table in the Arizona Geological Survey namespace.
- **Sequence**: Number, long integer. Third part of compound primary key. Orders records with the same MapLegendID/DataSetID values within a legend display. Domain: Integers >0 and $<10^{16}$.
- **ConceptID**: Number, long integer. Specifies the classification concept symbolized by the associated cartographic object in this table. It is a foreign key that joins to the ConceptID field of the *ClassificationConcept* table. Domain: Integers >0 and $<10^{16}$.
- **ConceptDS**: Number, long integer. Specifies the data set that contains the data object identified by ConceptID. Domain: Integers >0 and $<10^{16}$; typical value is 1, the DataSetID for the *ClassificationConcept* table.
- **CartoObjID**: Number, long integer. Identifies the cartographic symbolization for each spatial object on the default map visualization. It is a foreign key that joins to the CartoObjID field of the *CartographicObject* table. Domain: Integers >0 and $<10^{16}$.
- **CartoObjDS**: Number, long integer. Specifies the data set that contains the data object identified by CartoObjID for each record. Domain: Integers >0 and $<10^{16}$; typical value is 21, the DataSetID for the *CartographicObject* table.
- **DispPriority**: Long Integer. A priority number that allows the user to specify the order in which objects are drawn when the map is displayed. Objects with larger numbers are drawn on top of , and may hide, objects with smaller numbers. Domain: Integers >0 and $<10^{16}$.
- **DispVisibility**: Text, width 1. Determines whether or not a symbol is displayed in the legend. Domain 'Y' (the symbol is displayed in the legend); 'N' (the symbol remains hidden from view when the legend is displayed).
- **ClassName**: Text, width 255. The name for the geologic feature represented by the cartographic object (CartoObjID) in this MapView. Domain: Free text.
- **ClassLabel**: Text, width 16. The label to use in the map display to identify the geologic feature represented by the cartographic object (CartoObjID) in this MapView. Domain: Free text.
- **ClassDesc**: Memo. A text block for use in the map legend display that describes the geologic feature represented by the cartographic object (CartoObjID) in this MapView. Generally this description will correspond to the ConceptID that the symbol represents, modified by location and identification accuracy values from the classification scheme. Domain: Free text.
- **OriginDate**: Date/Time. Extra information field automatically filled with the date and time this record was added to the database.

- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value, typically 18, the DataSetID for the *TrackingRecord* table.

Map View Definition Table

The **MapViewDefinition** table defines a Title, Description, Extent, Projection, DesignScale, MapHorizon, ClassificationScheme and MapLegend to use for a particular MapView. A MapView is a collection of SpatialObjects within a bounded area (the Extent), classified using a particular ClassificationScheme, and symbolized using a particular MapLegend. The MapView does not necessarily use all the items in the MapLegend, or all the SpatialObjects classified under the ClassificationScheme. Every ClassificationObject in the ClassificationScheme that is related to a SpatialObject included in the MapView must have a CartographicObject assigned by the MapLegend associated with the MapView.

SimpleMapView – All SpatialObjects symbolized in the MapView are entirely within the MapExtent, and the set of CartographicObjects in the MapLegend is the same as the set of CartographicObjects used to symbolize spatial objects in the view.

GeneralMapView – SpatialObjects may come from different DataSets that may have extents different from the MapView extent, and the MapLegend may include Cartographic-Objects not used in the MapView. SpatialObjects symbolized in the MapView must be clipped to the MapExtent, and the MapLegend must be filtered to select only the items that appear in the MapView.

The ViewSchemeType in the **MapViewDefinition** table determines how the MapView is constructed. In addition to specifying if the view is a GeneralMapView or SimpleMapView, the ViewSchemeType also varies along a second dimension based on how the link between CartographicObjects and SpatialObjects is defined, as follows:

DefaultMapView – Represents a default visualization of a geologic data set. Default ClassificationObjects, CartographicObjects, necessary CartographicObject attributes (e.g. rotation for strike-and-dip symbols) and feature-linked annotation (polygon labels, dip values) are assigned using fields embedded in the SpatialObject tables. SimpleRelationship aggregates the DataSets containing the SpatialObjects through a simple relationship of type MapViewID; sequence attribute establishes display order for DataSets. All DataSets contain data within the same MapExtent. The MapLegend can be produced through a query that returns the union of unique ClassificationObject/CartographicObject pairs included in the records for all Spatial-Objects represented in the view. MapLegendID and ClassSchemeID are not required, but a predefined MapLegend is necessary to structure the MapLegend display, display order, and explanatory name, label and text (ClassName, ClassLabel, ClassDesc) for features; otherwise the default legend layout for the particular GIS implementation will be used.

DirectMapView – MapViewID is a RelationshipType for a SymbolizationScheme Relationship linking SpatialObject with CartographicObject, and MapLegendID identifies the appropriate MapLegend objects. All CartographicObjects used must be included in the MapLegend. The ClassificationSchemeID link in the MapViewDefinition identifies the classification scheme used as the basis for assigning symbols to spatial objects. The direct scheme is necessary for individually varying symbolization (e.g. structure symbols), and also allows for map generalization in which an object classified in the same way may be symbolized differently.

NADM43MapView – SpatialObjects are linked with ClassificationObjects through a ClassificationScheme specified by the MapViewDefinition, and ClassificationObjects are linked with CartographicObjects through the MapLegend. Assignment of CartographicObjects to SpatialObjects requires two joins, and ClassificationObjects are conceptually equivalent to CartographicObjects because, in order to symbolized an object differently, it must be classified differently. Thus, in order to rotate structure symbols to the correct display azimuth, ClassificationObjects for each azimuth must be generated, or the azimuth attribute of the data to symbolize must be propagated from the structural measurement table, through the SpatialObject, ClassificationScheme link (SpatialObject-Classification), and MapLegend link (Classification-CartographicObject).

Database Table Fields

- **MapViewID**: Number, long integer. First part of the compound primary key. Uniquely identifies each Map View defined in this MapView table. Domain: Integers >0 and <10^{16}, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the MapViewDefinition data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set. Typically will be 410, the DataSetID for the standard MapViewDefinition table in the Arizona Geological Survey namespace.
- **Title**: Text, length 255. Records the title displayed on the map view. Domain: Free text.
- **Author**: Text, length 255. Records the authorship displayed on the map view. For views that attempt to duplicate a published map, this would be the original authorship of the published map. Domain: Free text.
- **PublicationDate**: Date/Time. Records the date of creation of the map view. For views that attempt to duplicate a published map, this would be the original date of map publication.
- **Description**: Memo. Text description of the map. Could be used to store text blocks for display on the map layout. Should describe purpose of map. Domain: Free text.
- **DesignScale**: Number, Long Integer. Records display scale for which map view has been designed. The number is the denominator of the scale fraction. For example, if the map is designed for display at 1:24000, this field would contain the value '24000'. Domain: Integers >0 and <10^{16}.
- **CatalogLinksDatsetID**: Number, Long Integer. Identifier for a simple relationship data set (in DataSetAZ catalog table) that contains set of links of type 'MapView components', linking the MapViewID with the DataSetID's for all data sets required to construct the MapView. This aggregation must identify at least the data sets containing spatial data used by the map view, and the relationship tables that contain classification and symbolization links. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}).
- **ExtentID**: Number, Long Integer. Identifier for an Extent object (in an Extents metadata table) that defines the boundary of the geology displayed in this map view. Domain: Integers >0 and <10^{16}.
- **ExtentDS**: Number, Long Integer. Identifier for data set that contains ExtentID. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}). Typically will be 9, the DataSetID for the standard Extents table in the Arizona Geological Survey namespace.
- **ProjectionID**: Number, Long Integer. Identifier for a projection in a Projection metadata table. The projection describes the mapping between a non-planar map horizon and the planar map view surface. Domain: The set of ProjectionID values in the Projection data set (>0 and <10^{16}).

- **ProjectionDS**: Number, Long Integer. Identifier for the data set that contains ProjectionID. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}). Typically will be 16, the DataSetID for the standard Projection table in the Arizona Geological Survey namespace.
- **MapHorizonDesc**: Memo. Description of the map horizon, which is the physical surface that contains the geologic features displayed on this map view. Domain: Free text.
- **MapHorizonID**: Number, Long Integer. Identifier for the MapHorizon record in a MapHorizon metadata table. This link defines the base map and representation of the 3-D geometry of the physical surface represented by the map view. Domain: The collection of MapHorizonID values in the MapHorizon data set (>0 and <10^{16}).
- **MapHorizonDS**: Number, Long Integer. Identifier for the data set that contains the MapHorizonID. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}). Typically will be 434, the DataSetID for the standard MapHorizon table in the Arizona Geological Survey namespace.
- **MapLegendID**: Number, Long Integer. Identifier for the MapLegend aggregation in the MapLegend table. The MapLegendID serves as the filter for selecting ClassificationConcept-CartographicObject links, and identifies the symbols used in the map view. Domain: The collection of MapLegendID values in the *MapLegend* data set (>0 and <10^{16}).
- **MapLegendDS**: Number, Long Integer. Identifier for the data set that contains the MapLegend. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}). Typically will be 29, the DataSetID for the standard *MapLegend* table in the Arizona Geological Survey namespace.
- **ViewSchemeTypeID**: Number, Long Integer. Identifier for the classification concept that defines how symbols are assigned to spatial objects for this map view. Domain: 2785 = NADM4.3 type (spatial object-classification, classification-symbol); 2786 = MapLegend and Direct (spatial object-symbol through relationship table); and 3364 = Default (spatial object-symbol through attribute in native spatial object table).
- **ViewSchemeTypeDS**: Number, Long Integer. Identifier for the classification concept data set that contains the ViewSchemeType definitions. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}). Typically will be 1, the DataSetID for the standard *ClassificationConcepts* table in the Arizona Geological Survey namespace.
- **ClassSchemeID**: Number, Long Integer. Identifier for the ClassificationConcept that represents a collection of AttributedRelationship links between spatial objects and classification concepts that assign geologic significance to spatial objects. This value is used as the RelationshipType to select the relevant classification links. Domain: The set of ClassSchemeID values (children of ConceptID = 3360) in the *ClassificationConcepts* data set (>0 and <10^{16}).
- **ClassSchemeDS**: Number, Long Integer. Identifier for the classification concept data set that contains the definition of the ClassificationScheme. Domain: The collection of DataSetID values in the *DataSetAZ* data set (>0 and <10^{16}). Typically will be 1, the DataSetID for the standard *ClassificationConcepts* table in the Arizona Geological Survey namespace.
- **OriginDate**: Date/Time. Extra information field automatically filled with the date and time this record was added to the database.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and <10^{16}, no duplicates.

- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.

Pattern Definition Table

The **PatternDefinition** table defines graphical building blocks for constructing pattern fill symbols. (Return to <u>Table 17</u>.)

<u>Database Table Fields</u>

- **GraObjID**: Number, long integer. First part of the compound primary key. Uniquely identifies each polygon fill in the PatternDefinition data set. Domain: Integers >0 and $<10^{16}$, no duplicates.
- **DataSetID**: Number, long integer. Second part of the compound primary key. Uniquely identifies the GraphicPattern data set. Domain: Single value (typically), the DataSetID for this table in the *DataSetAZ* data set.
- **Name**: Text, width 255. Uniquely identifies the pattern element, for use in pick lists and to quickly indicate the nature of the pattern. Domain: Free text, unique values for each record in data set.
- **Description**: Text, width 255. Description of the pattern as a graphical element. Domain: Free text.
- **Dimension**: Number, single-precision real. Typical distance in millimeters between graphical elements in the pattern. Domain: Real numbers, >0 and $<10^8$.
- **Density**: Number, integer. Represents the equivalent gray-scale density of the pattern when printed at design size. Domain. Integers >0 and <100.
- **TrackingID**: Number, long integer. Uniquely identifies the origin tracking for each record. It is a foreign key that joins to the TrackingID field of the *TrackingRecord* table. Domain: Integers >0 and $<10^{16}$.
- **TrackingDS**: Number, long integer. Specifies the data set that contains the data object identified by TrackingID for each record. Domain: Single value (typically), the DataSetID for the *TrackingRecord* table.

Table 27. Example Pattern Definitions.

GraObj ID	DataSet ID	Name	Description	Dimen- sion	Den- sity
-1	31	to be defined	Pattern not defined	0	0
0	31	Solid	Solid color, opaque fill	0	0
1	31	Transparent	Transparent polygon	0	0
101	31	Stipple & blobs, low density	Variable size, tiny unfilled blobs and dots, random pattern. Conglomerate		15
105	31	Stipple & blobs, medium density	Variable size, tiny unfilled blobs and dots, random pattern. Conglomerate		25
114	31	Stipple, fine, grid, medium density	fine stipple, with points regularly space on a grid with horizontal rows and vertical columns, points same size	0.6	25

GraObj ID	DataSet ID	Name	Description	Dimen- sion	Den- sity
120	31	Stipple, variable dots, random	fine to medium dot size, random stipple; spacing between dots variable	0.75	20
121	31	Stipple, coarse, medium density, vague pattern	Dots same size, vague overlapping circles pattern, like cross-bedding. Dots about 0.5 mm dia	1.25	17
314	31	Double hatch pattern	porphyritic gran		
317	31	V, Random	randomly oriented and positioned v symbols, symbol size constant. Granite		
327	31	Cross, random	diorite		
401	31	Triangles, open	breccia		
431	31	Lines, wavy, diagonal	Wavy lines, 45 deg. CW from vertical. gneiss or schist		

REFERENCES

Bernknopf, R. L., Brookshire, D. S., Soller, D. R., McKee, M. J., Sutter, J. F., Matti, J. C., and Campbell, R. H., 1993, Societal Value of Geologic Maps: Washington, D.C., U. S. Geological Survey Circular 1111, 53 pages.

Brodaric, B., 2000, Digital Geological Knowledge: From the Field to the Map to the Internet, *in* Soller, David R., Ed., Digital Mapping Techniques '00 -- Workshop Proceedings, U. S. Geological Survey Open-File Report 00-325, p. 3-12.

Brodaric, B., and Gahegan, M., 2000, Geoscience Map Data Models, Open Systems GIS and Semantics, Proceedings, GeoCanada2000-The Millenium Geoscience Summit , Calgary, Alberta, p. 7. *available at* http://www.gisworld.org/gac-gis/2000/1147.pdf, Accessed June 14, 2001.

Brodaric, B., Gahegan, M., Takatuska, M., and Harrap, R., 2000, Geocomputing with geological field data: Is there "a ghost in the machine", Proceedings, Geocomputation 2000, Chatham, UK, *available at* http://www.geog.psu.edu/~mark/geocomp2000b/gc028.htm, *accessed* June 14, 2001.

Brodaric, B. and Hastings, J., 2000, Evolution of an Object-Oriented, NADM-Based Data Model Prototype for the USGS National Geologic Map Database Project [web page, abstract]: Annual Conference of the International Association for Mathematical Geology, IAMG2001, Cancun, Mexico, *Available at* http://www.kgs.ku.edu/Conferences/IAMG/Sessions/I/brodaric.html, *Accessed* June 14, 2001.

Brodaric, B., Journeay, M., Talwar, S., and others, June 18, 1999, CordLink Digital Library Geologic Map Data Model Version 5.2 (Web Page), *Available at* http://cordlink.gsc.nrcan.gc.ca/cordlink1/info_pages/English/dm52.pdf., *Accessed* June 13, 2001.

Johnson, B. R., Brodaric, B., and Raines, G. L., 1998, Digital Geologic Maps Data Model, V. 4.3 (Web Page): Available at http://ncgmp.usgs.gov/ngmdbproject, U. S. Geological Survey.

Richard, S. M. 1999, Geologic concept modeling, with examples for lithology and some other basic geoscience features, in Soller, D. R., ed., Digital Mapping Techniques 1999, Workshop Proceedings: U. S. Geological Survey Open-File Report 99-386, p. 59-75.

Richard, S. M., 1998, Digital Geologic Database Model (Web Page): *Available at* http://www.azgs.state.az.us/GeoData_model.pdf, *Accessed* June 13, 2001.

Richard, S. M., and Orr T. R., in prep., Metadata data structure for Arizona Geological Survey Geologic Information System: Tucson, Arizona Geological Survey Open-File Report.

Richard, S. M., and Thieme, J. P., 1997, Data structure for Arizona Geological Survey digital geologic maps: Tucson, Arizona Geological Survey Open-File Report 97-5, 15 pages.

Trapp, R. A., and Reynolds, S. J., 1998, Physiographic areas in Arizona used by the Arizona Geological Survey: Tucson, Arizona Geol. Survey, Digital Information Series DI-10, 4 pages, 1 floppy disc.

Trapp, R. A., Schmidt, N., and Reynolds, S. J., 1996, AZGEOBIB, Version 2.1: A List of References on the Geology of Arizona: Arizona Geological Survey Open-File Report OFR-96-01, p. 308.