

# Migrating from ArcInfo Workstation to ArcGIS

By Victor Dohar

Natural Resources Canada  
601 Booth Street  
Ottawa, Ontario, Canada K1A 0E8  
Telephone: (613) 943-2693  
Fax: (613) 952-7308  
e-mail: vdohar@NRCan.gc.ca

## INTRODUCTION

During the past ten years, geological maps at the Geological Survey of Canada (GSC) have been produced using ESRI's ArcInfo Workstation software along with a custom application called GEMS (GEological Mapping System – available at [http://www.nrcan.gc.ca/ess/carto/toolbox\\_e.html](http://www.nrcan.gc.ca/ess/carto/toolbox_e.html)). Supporting this application in delivering high-quality publications, Cartographic Digital Standards and Cartographic Design Specifications are used to manage the spatial data and surround elements that appear on the map, respectively.

With the recent introduction of ArcGIS and the benefits that this new technology offers, considerable re-tooling of the process will be required to achieve the product quality that we achieved with ArcInfo Workstation. This will cover all aspects of production, from data management to developing and implementing new applications. From a cartographic perspective, this document will explain the migration strategies for developing a geodatabase design to manage the data, as well as some ideas for producing maps with ArcMap. The content of this document is based on the ESRI product ArcGIS 8.3. General knowledge of ArcGIS, particularly geodatabase elements and design, will be beneficial.

## MIGRATING TO ARCGIS

### Goals

One of our main goals is to be able to produce geological maps with ArcMap as effectively and efficiently as with ArcInfo Workstation. The majority of this effort has involved designing a geodatabase schema to store and manage the geological features that constitute a map. The geodatabase also provides the means for feature-validation through the use of domains for standardizing feature names and user-defined topologies. Domains can be used to restrain the user in assigning values to an attribute field from an established list of coded values or a numeric range. Topologies will ensure users adhere to

the established spatial relationships between features.

Another goal is to keep the design of the geodatabase simple and easy to understand. This will be achieved by eliminating unnecessary data fields, using intuitive and obvious field names, implementing the use of domains, and using tables and defined relationships where necessary.

### Migration Strategy

The gradual migration from ArcInfo to ArcGIS can be characterized as occurring in three stages. These are outlined in ESRI's help documentation, and are summarized below:

1. The first stage is to use ArcGIS for data use and visualization, not data management. Workstation would still be used to edit and maintain coverage data, but ArcGIS's ArcMap would be used for producing the geological map.
2. The second stage is the most complicated. This involves moving the data management from coverage-based files used in ArcInfo Workstation to the geodatabase model of ArcGIS. Designing an effective geodatabase schema will require prototype implementations to determine the most appropriate design.
3. The final stage revolves around the geo-processing tools that will be introduced in ArcGIS 9.0. These tools will be used to create applications to upgrade data from Workstation to ArcGIS.

In our migration strategy, we decided to merge steps 1 and 2, delivering both at the same time. The primary reason for this approach is to eliminate the transition phase that could affect our service delivery of map products. Using two software suites to produce maps and manage data could potentially create bottlenecks, and data management and versioning issues that could affect the timely delivery of map products (both paper and digital). In addition, based on past experiences, it will be easier to train everyone at the same time to make the transition.

## DESIGNING A GEODATABASE

A geodatabase contains the geographic or spatial features and their attributes, which can be used to produce a geological map. Geographic features of a homogenous theme and spatial type (e.g., contacts, which are lines) are stored in feature classes. Feature classes that are physically related to one another (e.g., contacts, faults, and geological polygon units) are best to be grouped in a feature dataset. This grouping permits the application of user-defined topological rules that express the relationships between each of the feature classes. The feature name, definitions, attribute fields, and properties of these feature classes determine the schema or design used in the geodatabase. There are three basic methods in designing a geodatabase schema:

1. The first is using ArcCatalog's menus and wizards. This is probably the simplest method of creating a geodatabase schema. The geodatabase, feature datasets, feature classes, relationship classes, and tables can all be created using these wizards. Domains can also be entered as properties of the geodatabase. All the menus are simple to use and navigate, yet some general understanding is required when entering all of the geodatabase information. After becoming familiar with designing geodatabases, the procedures become very repetitive and somewhat rudimentary. Transferring or sharing the geodatabase schema and its properties with other geodatabases, however, is not possible. Even the smallest of changes will require extensive re-entry.
2. The second method uses ArcCatalog's CASE tool to import a schema that is created by another software, such as Visio. Although creating schemas using UML (Unified Modeling Language) is the preferred choice for developers, access to this type of software was unfortunately not possible for us, therefore this option was not investigated. One thing to point out, however (which is also the case for the first method), is that existing geodatabase schemas cannot be exported from ArcCatalog and used in CASE tools. Any changes must be performed using the CASE software and then the schema must be imported into ArcCatalog. The CASE tool is expensive and is not available for ArcView licenses.
3. The third method uses Geodatabase Designer to define the geodatabase schema. The main benefit of this tool is that it can take an existing geodatabase and export its schema to XML (extensible mark-up language). Changes to the XML (the geodatabase schema) can be performed in an XML editor and then imported either into the same geodatabase or to a new one. Geodatabase Designer is fully inte-

grated with ArcCatalog, and free to download from ESRI's Support Center website (<http://support.esri.com/index.cfm?fa=downloads.gateway>). At that site, search for "geodatabase designer"; however, it must be noted that it is an unsupported application.

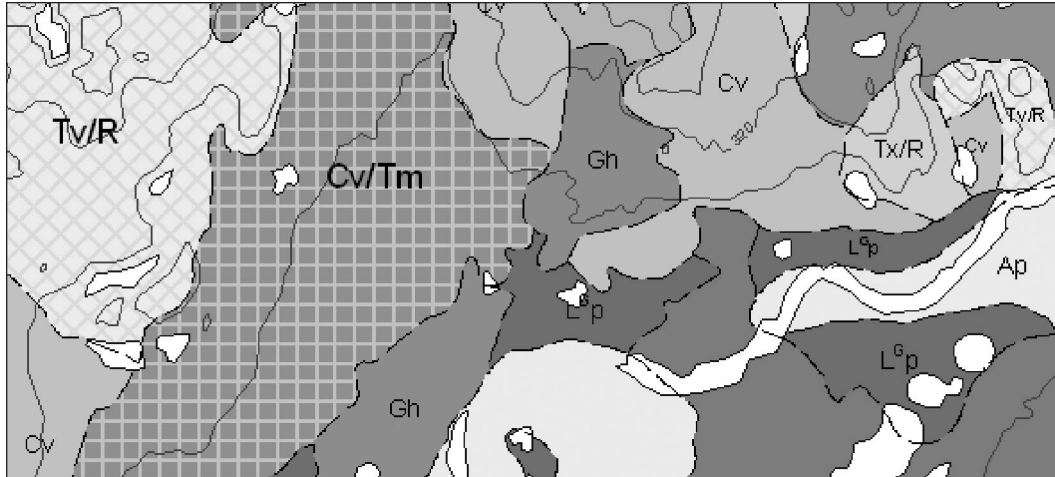
## Designing a Geodatabase for Map Production

Our map products are either bedrock or surficial maps, and so we decided to tailor the design of geodatabases to this type of information. Therefore, separate geodatabases will be created for bedrock and surficial geological datasets. This may seem redundant, but there are areas where both thematic features have been mapped, either on different maps or on one map. Storing the data in separate geodatabases will provide a better means for managing the data as well as referencing it when making maps with ArcMap. Some of the criteria we will use for creating a geodatabase schema are as follows:

- create separate geodatabase schemas for bedrock geology data, surficial geology data, and cartographic elements (non-geological features like borders, index maps)
- do not store data in a geodatabase if it already exists in another database and is accessible when making a map (e.g., geochronology database)
- avoid duplication and redundancy in feature attribute values by retaining only the fields that are necessary, and use tables where appropriate to expand attribute meaning (e.g., a single field can be used to define each geological polygon, which in turn relates to another table containing information of each unit)
- use domains and topology to validate features
- use sub-types along with domains to group/categorize simple features
- use relationship classes to define relates between tables.

## Managing Geological Polygons and Legend Information

At this time, much of the effort at GSC has involved implementing a surficial geological geodatabase; however, there will be many aspects applicable to a bedrock geological geodatabase. In the design of a surficial geodatabase schema, the first feature we examined was the geological polygons. The surficial map sample from ArcMap (Figure 1) displays one of the methods authors from the GSC use to classify geological polygons. Many of the surficial geological polygons are labeled on the map with one or two units, where each unit is listed and described separately in the legend. These compound units (e.g. "Cv/Tm") can consist of veneers (in this case, "Cv")



**Figure 1.** Sample black & white image of a colour surficial map from ArcMap.

overlying a dominant unit (“Tm”) or as a proportional mix between two geological units (Tx/R). In most cases, veneers include the lowercase character “v”, and for compound units, a symbol is used between the two geological units to express the type of relationship and ratio of each unit. These methods are intended as a visual aid for the map-reader, but a different approach is required to better manage this type of information and structure in a geodatabase.

The values assigned to the geological polygons, including the relationship information for these compound units, is stored in the attribute field MAPUNIT, in the polygon feature class table (Figure 2). The values do not necessarily have to match the polygon label on the map.

OBJECTID*	SHAPE*	MAPUNIT*
1	Polygon	LGp
2	Polygon	Ap
3	Polygon	Tv
4	Polygon	Tv/R
5	Polygon	Tb
6	Polygon	Gh
7	Polygon	Gx
8	Polygon	Gh
9	Polygon	Tm
10	Polygon	Tv/R
11	Polygon	R/Tx
12	Polygon	Gx
13	Polygon	Tv/R
14	Polygon	R/Tx
15	Polygon	Tv
16	Polygon	Gx
17	Polygon	Tx
18	Polygon	Tv

**Figure 2.** Snap shot from ArcCatalog. The left-hand side displays the catalogue tree, with the arrow pointing to the currently selected polygon feature class “OF4281\_SurficialUnits”. The attributes of these features are displayed on the right-hand side. The bold outline contains the attribute values of each feature in the MAPUNIT field.

In fact, they can be any value, string or numeric, as the value relates to another table that describes the composition of each unique polygon value as well as the label to be placed on the map.

The table UnitComp (Figure 3) is used to describe the composition of each unique polygon value. In this table, a field also named MAPUNIT contains all the unique polygon values from the above polygon feature class. Three additional fields are used to define the composition. The VENEER field contains geological units that are classified as veneers. The PRIMARY\_UNIT and SECONDARY\_UNIT fields contain geological units that are classified as the primary and secondary units respectively. The values in these three fields relate directly to the geological units that appear in the legend.

A fourth field named COMPLEX\_ID is used to identify the relationship or proportional mix between each of the geological units in a composition as defined by the author. The integer values in this field relate to the ComplexIDs table (Figure 4), where each type of relationship or complex ID is explained. On the map, symbols such as · (middle dot), - (dash), / (slash), and // (double slash) are used as visual aids to express these relationships (e.g., the middle dot between two units is used to express a 50/50 proportional mix between a primary and secondary unit).

Another table, called LegendUnits (Figure 5) contains all the geological units that appear on the legend and provide the definitions for all surficial geologic units listed in the UnitComp table. In the LegendUnits table, each geological unit contains a description, as well as heading information (not shown in figure) that would appear on the legend (e.g., Glacial Till).

The final table associated with geological polygons is used for dynamically labeling the polygons in ArcMap. This table, called UnitLabels (Figure 6), also contains all unique polygon values for the field MAPUNIT found in the above polygon feature class. The field MAPLABEL

MAPUNIT*	VENEER*	PRIMARY_UNIT*	SECONDARY_UNIT	COMPLEX_ID*
Ap		Ap		0
At		At		0
Ca		Ca		0
Cb		Cb		0
Cb/LGp		Cb	LGp	7
Cv	Cv	Cv		0
Cv-LGd	Cv	LGd		2
Cv-Tm	Cv	Tm		2
Cv/Gx	Cv	Gx		1
Cv/R	Cv	R		1
Cv/Tb	Cv	Tb		1
Cv/Tm	Cv	Tm		1
Gh		Gh		0
Gh/Cx		Gh	Cx	7
Gp		Gp		0
Gt		Gt		0
Gv	Gv	Gv		0
Gv-R	Gv	R		2

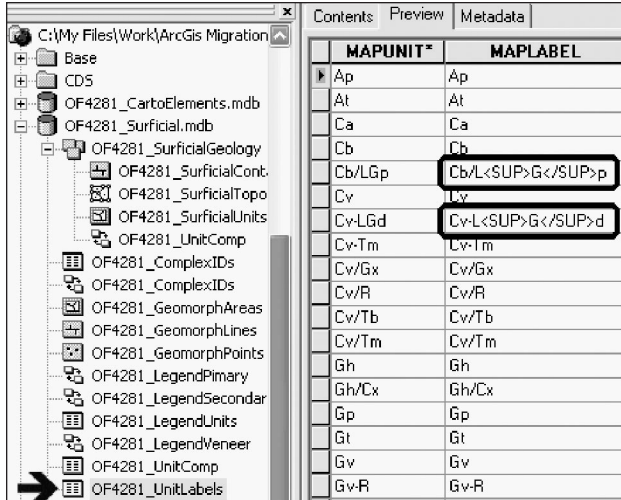
Figure 3. Snap shot from ArcCatalog. The arrow is pointing to the currently selected table “OF4281\_UnitComp”, which is displayed to the right. This table contains all unique polygon values in the field MAPUNIT. Subsequent fields further define the composition of each unique polygon value and the nature of the composition.

COMPLEX_ID*	DESCRIPTION
0	No Complex cover exists.
1	Veneers are observed as a minor component forming patches over another unit. (/ used on map)
2	Veneers are interpreted to stratigraphically overlie a dominant unit. (- used on map)
5	Complex cover, approximately 50:50 ratio between primary and secondary units. (- used on map)
7	Complex cover, approximately 70:30 ratio between primary and secondary units. (/ used on map)
9	Complex cover, approximately 90:10 ratio between primary and secondary units. (// used on map)

Figure 4. Snap shot from ArcCatalog. The arrow is pointing to the currently selected table “OF4281\_ComplexIDs”, which is displayed to the right. This table lists the different types of relationships for geological units that participate in a composition.

LEGUNIT*	DESCRIPTION
At	Alluvial terrace: Massive to stratified, moderately to well sorted sand and gravel with minor silt. Sediments are of floodplain o
Ap	Alluvial plain: Predominantly sands and gravels. May be locally overlain by or include lacustrine silt, clay and minor peat and
Cb	Colluvium Blanket: A mantle of colluviated material with a thickness greater than 1 m.
Cv	Colluvial veneer: Thin (<1 m) or discontinuous sheets of colluviated materials. Hatch-fill is used when colluvial veneer comp
Gh	Hummocky glaciofluvial (ice-contact): Complex arrangement of slopes extending from rounded depressions, to irregular con
Gp	Glaciofluvial plain (Dutwash): low-relief mantle of moderately to well sorted, cross-stratified sand and rounded gravels; 1-20
Gt	Glaciofluvial terrace (Dutwash): a scarp or face with a low-relief mantle of moderately to well sorted, cross-stratified sand an
Gv	Glaciofluvial veneer: Stratified to massive gravel, sand and silt. May occur in patches or gravel lag over rock; thicknesses a
Gx	Glaciofluvial - undifferentiated: Glaciofluvial complex - units are too small to be represented at the scale of mapping. Consist
LGd	Glaciolacustrine delta: a scarp or face with a low-relief mantle of cross-stratified sand and rounded gravels associated with
LGp	Glaciolacustrine plain: Well stratified clay, silt and sand. Local relief is less than 1 m (plain) and masks the underlying topogr
LGx	Glaciolacustrine - undifferentiated: Glaciolacustrine complex - units are too small to be represented at the scale of mapping.
R	Bedrock - undifferentiated: Cross-reference with Committee Bay Project, bedrock component.
Tb	Till blanket: Surface morphology conforms to underlying bedrock topography. May exhibit crag-and-tails, flutes, and/or roch
Th	Hummocky till: Stratified to massive diamicton and interstratified glaciofluvial gravel and sand. Stratification often exhibits sy
Tm	Rolling till plain: Surface morphology forms gently rolling plains with 1 to 3 m of relief; may exhibit flutes. Generally masks und
Tp	Till plain: Surface morphology forms a plain with < 2 m of relief. Generally masks underlying topography. Some areas have la
Tv	Till veneer: Till less than 1 m thick; occurs in patches over rock and is interspersed with rock outcrop; deposits are thin and
Tx	Till - undifferentiated: Till complex - units are too small to be presented at the scale of mapping. May contain relatively small

Figure 5. Snap shot from ArcCatalog. The arrow is pointing to the currently selected table “OF4281\_LegendUnits”, which is displayed to the right. This table contains each geological unit and its description as it appears in the legend on the map.



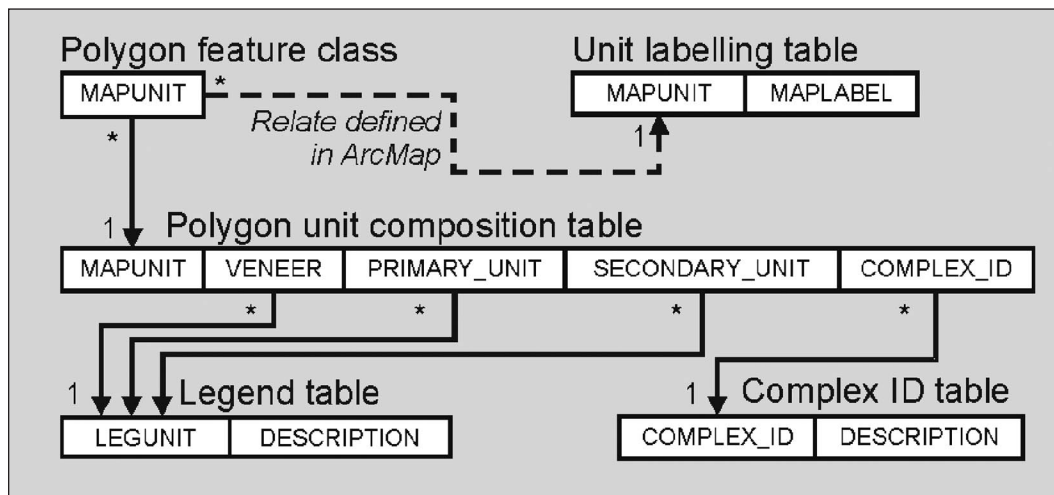
**Figure 6.** Snap shot from ArcCatalog. The arrow is pointing to the currently selected table “OF4281\_UnitLabels”, which is displayed to the right. This table is used in ArcMap for dynamically labeling polygons. The field MAPLABEL is used as the label text field for each unique polygon. The bold outline displays two examples of using text-formatting tags, in this case to achieve the superscript character.

contains the text string that will be used by ArcMap to label the polygons. Using this kind of table ensures consistency and allows the use of text formatting tags to show unit labels with superscript, subscript, bold, and coloured type (see text formatting tags in ArcMap Help for more information). In the example map in Figure 1, the geological unit L<sup>G</sup>P contains a superscript character (the superscript G for this unit was requested by the author for this particular map). The field MAPUNIT is defined as a text field and cannot contain any superscript or special

characters, therefore the value “LGP” is used for these types of polygons. The field MAPLABEL, also defined as a text field, contains the superscript formatting tags <SUP></SUP> to achieve the superscript character when labeling (circled in Figure 6).

In order to use this table in ArcMap for labeling, an ArcMap relate must be specified between the layer (polygons) and the table. A relationship class (discussed below) is not necessary, as this table is only used for labeling purposes, and not used for querying the polygon and legend information. Furthermore, when labeling polygons in ArcMap, classifications can be created based on the area of the polygons being labeled to achieve different point sizes for the labeled text strings. For example, small polygons (area ≤ 10,000 m<sup>2</sup>), medium polygons (area > 10,000 m<sup>2</sup> and ≤ 100,000 m<sup>2</sup>), and large polygons (area > 100,000 m<sup>2</sup>) can be labeled with 8, 10, and 12 pt. type respectively. As with all GIS software, the labels are arbitrarily placed, and therefore further cartographic refinement is required. The labels can be converted to annotation and their placement adjusted to provide a more legible map. At this time, we are unsure if this table and/or the use of relates in ArcMap can be used in advanced labeling engines like the MAPLEX extension.

For data analysis and map production, these many tables must be related by creating relationship classes. A relationship class defines which fields are related between tables, as well as the nature of the relate. A total of five relates exist, each defined as a separate relationship class. Figure 7 shows the table and field names, and their relationships. You can see the relationship classes in the previous figures, preceded by the icon (two small boxes with arrows pointing to each other in the catalogue tree). These tables and relationship classes exist in the geodatabase, but they could also exist in the feature dataset containing the SurficialGeology polygon feature class



**Figure 7.** Diagram showing the relationship between the fields in the tables. Each line is defined as a relationship class in the geodatabase. Cardinality is expressed as 1 (one) and \* (many).

to which they are related. Storing them at either location is acceptable, but it would have been better in these examples, from a file management perspective, to store the tables and relationship classes in the SurficialGeology feature dataset.

Creating relationship classes will allow end-users to query information from ArcMap (at this point it is unsure if this can also be achieved with ArcIMS, ArcReader, or ArcExplorer). By selecting a polygon, the end-user can display the attributes from all of the tables by navigating through the established relates (Figure 8).

## Managing Geological Line, Point, and Other Area Features

Our current plan for managing all other surficial geological line, point, and area features on a geological map is to store them in separate, stand-alone feature classes. In the surficial geodatabase schema discussed above, these would be the feature classes GeomorphLines, GeomorphPoints, and GeomorphAreas. These feature classes would also incorporate the use of sub-types and domains to assist in categorizing and validating the features.

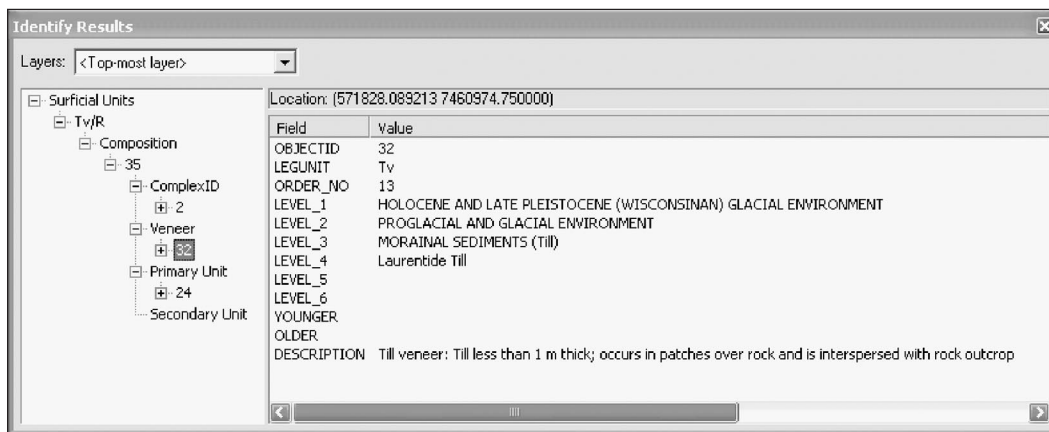
Domains are a property of the geodatabase, and as such can be applied to any feature class. In fact, the domains created for these features will be generic to all geodatabases. Domains are used to constrain values for a field, and can include acceptable coded values or a numeric range. In most cases, coded value domains are used to specify each type of geological feature, and numeric range domains are applied where numeric fields are used to store attribute information, such as the strike (rotation) and dip of a geological symbol. Domains are used in all custom fields in a feature class; thus validating a feature

class will ensure that the attribute values of all features are acceptable and valid. In the following example (Figure 9), the Glacial domain has been created as a coded value domain that lists valid values and their descriptions.

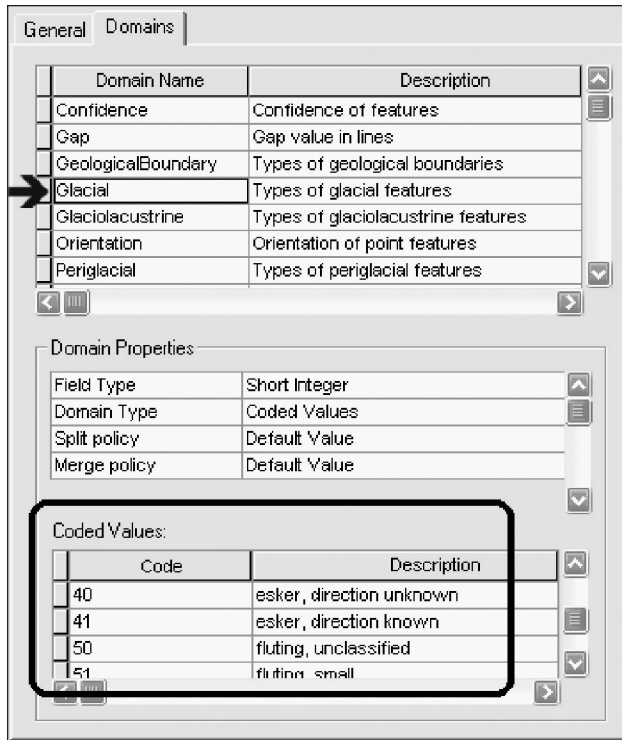
Sub-types provide a means for categorizing or grouping features within a feature class by assigning unique values to a long-integer defined field. In the example below (Figure 10), the polygon feature class has a sub-type field named CATEGORY, with two valid values: 1 (to store glacial features), and 2 (to store glaciolacustrine features). Sub-types allow for different domains to be applied to each sub-type classification. For example, for the glacial and glaciolacustrine feature sub-types, the field FEATURE will only accept coded values from the Glacial and Glaciolacustrine domains respectively. This is highly useful, because each sub-type classification can have its own unique attribute values, which are specified by the domain. Furthermore, these domains can be applied to more than one feature class. Depending on the compilation scale, features can be stored either as points, lines, and/or areas in the respective feature classes (GeomorphPoints, GeomorphLines, GeomorphAreas). For example, drumlins are usually stored as point or line features. In both feature classes they would exist in the sub-type Glacial, and the Glacial domain would apply to the field FEATURE. Therefore, these features must be coded as “drumlin” based on the coded value from the Glacial domain, regardless of the type of feature.

## User-defined Topologies

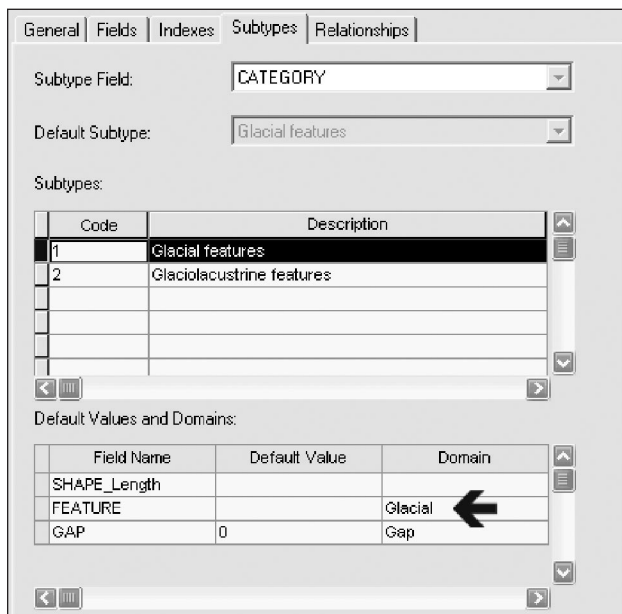
In addition to feature validation, another benefit of geodatabases is the ability to create custom topologies



**Figure 8.** ArcMap snap shot displaying results of an identify procedure on a selected feature. The left-hand side displays the tables in a tree format as one migrates from the selected feature, through the established relates to other tables (just as confusing as it appears). The right-hand side displays the attributes of the geological unit “Tv” from the legend table. For this selected feature, the geological unit “Tv” also participates in a composition as a veneer with the map unit “Tv/R”.



**Figure 9.** Snap shot from ArcCatalog displaying the domains as a property of the geodatabase. The arrow is pointing to the “Glacial” domain with valid coded values and descriptions shown below in the bold outline.



**Figure 10.** Snap shot from ArcCatalog displaying the subtypes as a property of a feature class. Highlighted is the “Glacial features” subtype, in which case values for the field FEATURE must adhere to the “Glacial” coded value domain.

based on feature classes that exist in a feature dataset. Custom topology involves specifying which feature classes will be participating, the relationship rules between the features, and the cluster tolerance (similar to the fuzzy tolerance from ArcInfo Workstation). For our application, user-defined topologies apply only to the geological polygons datasets, where the SurficialContacts and SurficialUnits feature classes participate. The two rules that are implemented are:

1. Polygons from the feature class SurficialUnits must not overlap,
2. The lines (contacts) from the feature class SurficialContacts must overlies the boundary of the polygons from the feature class SurficialUnits.

The second rule simply states that the contact lines must coincide with the outline of the polygons. The rule could have been expressed the other way around: the outlines of polygons must overlies the contact lines, but this would have caused excessive topological dirty areas (errors) where contact lines do not exist (i.e., at the edges of bodies of water which close some polygons).

## MAKING MAPS WITH ARCMAP

### Style Files and Symbols

Unfortunately, customized symbolsets from ArcInfo Workstation cannot be used directly or imported into ArcMap. This has required a significant investment of time, to rebuild the libraries for approximately 400 line and 700 point symbols. The majority of the time has been spent creating TrueType fonts containing the point symbols, using third-party software (Font Lab and Fontographer). Our current set of symbols/style files for ArcMap are available free of charge from our “Migration to ArcMap” website ([http://www2.nrcan-rncan.gc.ca/ess/carto/arcgis\\_e.asp](http://www2.nrcan-rncan.gc.ca/ess/carto/arcgis_e.asp)).

### Using XML for Map Surrounds and Legends

During the past ten years of producing geological maps using ArcInfo Workstation, several customized commands have been developed to assist cartographers in placing surround information (title blocks, legends, logos, credits, descriptive notes, recommended citation, figures, etc...). Unfortunately, these commands are all written in AML (Arc Macro Language), and as such cannot be used to position these elements, or to aid in the layout of a map using ArcMap. In order to provide the same level of functionality, scripts will have to be written using VBA (Visual Basic for Applications) and ArcObjects. In addition, the information associated to each of the surround elements will be stored in XML. The source of this information will either be obtained from existing metadata or inputted by

the cartographer. Storing this information in XML will not only assist in the placement of these surround elements, but the XML file can be used by third parties to extract the information about the map they require. This effort is currently being developed, and is not yet incorporated in the production of maps with ArcMap.

Another aspect of the map surround information is the geological legend. These legends, created using ArcInfo Workstation, also rely on AML programming. A similar approach will also be taken in producing legends using ArcMap, utilizing XML to store content derived from the legend tables. Much of this work is just beginning.

## ACKNOWLEDGMENTS

I wish to acknowledge the efforts of Roger MacLeod and Parm Dhesi, in migrating the ArcInfo symbolsets to

style files; Norah Brown for her expertise in ArcObjects programming and XML; and Sheila Hynes, Barb Szlavko, and Terry Houlahan for their support and expertise in geodatabase, SDE, and Oracle administration.

## REFERENCES

- ESRI Inc., 1999-2002, ArcGIS Desktop Help 8.3: ESRI Inc.
- ESRI Inc., 2003, Geodatabase Designer Help: ESRI Inc. (developed by the Applications Prototype Lab).
- ESRI Inc., 2003, Geodatabase Case Studies: ESRI Inc., Geodatabase Case Studies, Chapter 9 Topography and the Base Map (draft publication).
- Fontlab software is available from Fontlab Ltd., Inc., accessed at <http://www.fontlab.com>.
- Fontographer software is available from Macromedia, Inc., accessed at <http://www.macromedia.com/software/fontographer>.