

Main concepts of ILWIS for Windows

In chapter one, Introduction to ILWIS for Windows, you started with ILWIS, and learned the basics of the user interface. This chapter presents key concepts of ILWIS for Windows.

In the first exercise, the two basic types of maps that are used in ILWIS are treated: vector and raster maps. The various types of vector maps are shown: point, segment and polygon maps. Vector and raster maps can be displayed simultaneously in a map window using the **Layer Management** option. You will also have a closer look at one of the key concepts of ILWIS: the use of domains. A domain defines the possible contents of a map, table or column. It may contain class names, unique identifiers or values. In the second exercise of this chapter you will look in detail how vector maps are related to other objects, such as a coordinate system (containing information on the kind of coordinates that are used in the maps), a domain, and a representation (defining the colors in which the items of the domain should be displayed). In the third exercise of this chapter you will pay special attention to raster maps, and how they are related to other objects, such as a georeference (containing information about the number of lines and columns, the coordinates and the pixel size). In exercise 2.4 you will see how spatial and attribute data are linked and how you can retrieve information from both simultaneously using the pixel information window. The last exercise deals with another important concept of ILWIS: dependency links. In ILWIS the maps, tables and columns that are made through the various operations, know how they were made. This has some major advantages, such as the possibility to store only the definition of a map, without storing the data of the map itself, and the ease with which maps can be updated.

Before you can start with the exercises, you should start up ILWIS and change to the subdirectory `c:\ilwis21\data\usrguide\chap02`, where the data files for this chapter are stored.



- Double-click the ILWIS program icon in the ILWIS program group.
- Change the working drive and the working directory until you are in the directory `c:\ilwis21\data\usrguide\chap02`.

2.1 Structure of spatial data in ILWIS

In the previous chapter we have looked at how you can display maps and tables in ILWIS. Now we will go more in depth about the way the data are structured. We will first pay attention to spatial data.

Spatial features are represented in ILWIS in the following ways:

- **Points.** Many items can be represented as single points on a map. Points may refer to rainfall stations, houses, sampling points for field observations, etc.
- **Lines.** Linear features such as roads, drainage lines or contour lines.
- **Areas.** Features which occupy a certain area, such as a land use unit (forest), a geological unit etc.

The spatial entities described above can be represented in digital form in two data models (figure 2.1): *vector*, or *raster models*. Both models store details on the location of entities and their value, class name or identifier. The main difference between the two data models is the way they store and represent the locations.

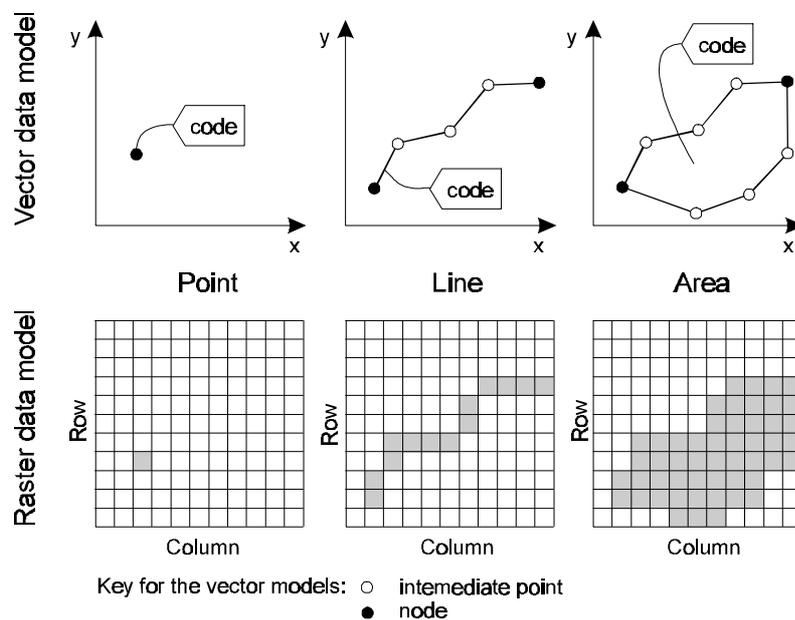


Figure 2.1: Vector and raster representation of points, lines and areas

Vector model

In a *vector model* the position of each spatial feature is defined by (a series of) X and Y coordinates. Besides the location, the meaning of the feature is given by a 'code'.

Points

The location of a point is described by a single *x-y coordinate pair*. Points are stored in **point maps** by registering their location in X and Y coordinates and by coding them (e.g. a rainfall station, elevation).

Segments

A line is stored as a series of X and Y coordinates (points) that best reflect its characteristics. The points are connected by straight lines. In ILWIS the line features are called *segments*. A segment is described by the coordinates of its starting and ending points (its *nodes*) and the *intermediate points*. The code of a segment describes what the segment represents (e.g. road, state boundary, contour value).

Polygons

Areas are represented by their borders or boundary lines, which are in fact segments, with the difference that all segments are enclosing areas. Such an area defined by its boundary line and by its code is called a *polygon*. The code represents the contents of the polygon (e.g. forest, city, sandstone, etc.).

Vector maps in ILWIS are either *point*, *segment*, or *polygon maps*. Most of these are obtained by digitizing. Vector maps require less disk storage space and are suitable for creating high quality outputs. They are less suitable for a number of GIS operations, especially those dealing with map overlaying. You can rasterize the vector maps (this will be explained in chapter 4) into raster maps. In ILWIS, most of the analysis operations are raster-based.

Raster model



In a raster model, spatial data are organized in *grid cells* or *pixels*, a term derived from *picture element*. Pixels are the basic units for which information is explicitly recorded. Each pixel is only assigned one value. The maps stored in ILWIS according to the raster model are called *raster maps*. Pixels in a raster map all have the same dimensions. Thus, it is unnecessary to store all their coordinates as the pixels are arranged in a regular pattern. It is enough to determine the pixel size and the parameters to transform between X and Y coordinates of a map and the pixel locations in the raster map (rows/lines and columns). The process to establish this relationship is called *georeferencing*. Through a georeference you calculate the parameters used in the equations to transform between a coordinate system and pixel locations in the image.

A *point* is described in the raster model by the position of a single pixel. The position of each cell is defined by a row and column number. A pixel is assigned one code (label). A *line* and an *area* are described by a set of connected pixels having the same code (label). In the raster model there is no basic difference in how points, lines and areas are stored.

Through *rasterization* vector data (points, segments or polygons) can be converted to raster formats. Apart from this source of raster maps, there are scanned aerial photographs, satellite images or scanned pictures. Most ILWIS analyses are based on the raster data model. The results of most of the operations are also raster maps.

Displaying vector and raster maps

Lets us now have a look at the different types of maps in ILWIS. In the previous chapter you have seen the Landuse map. Remember that it was a polygon map. Let us now compare the polygon map with a raster map, which is obtained by rasterizing the polygon map.



- Double-click the polygon map Landuse in the Catalog.
- Accept the default values in the Display Options - Polygon Map dialog box by clicking OK. The polygon map is displayed.
- Click the title bar of the map window, keep the left mouse button down, and move the map window to the side of the screen. Then release the left mouse button. This is called *dragging*.
- Double-click the raster map Landuse in the Catalog. After a while the Display Options - Raster Map dialog box is shown in which you can specify how you want to see the raster map. Note that there are less options than in the Display Options dialog box for a polygon map. Accept the default values in the dialog box by clicking OK. The raster map is also displayed in another map window.
- Drag both windows so that they are next to each other.

The first impression is that both maps are the same. They are similar with respect to the information they contain. You can check this by clicking the same unit in both maps. The units of both maps are also displayed in the same colors. The difference, however, is in the way the information is stored in both maps; one in vector format and the other in raster format.

You can best evaluate this by zooming in on both maps.



- Click the Zoom In button in the button bar of the raster map. Position the mouse pointer in a section of the map near the boundary of several units. Press the left mouse button, hold it down, and drag it to define a small area to zoom in on.
- Repeat this procedure for the polygon map. Zoom in on the same area.

Now you can see quite some difference between the two maps. The polygon map displays smooth boundaries between the units, whereas the raster map has a blocky appearance. If this is not clear then zoom in further. The blocky appearance in the raster map is related to the fact that each pixel in the map belongs to either one of the land use units. You are able to see individual pixels.



- Close the map window that displays the polygon map Landuse.

We can see the individual pixels even more clearly when we display another type of raster map: a satellite image (Tmb1: band 1 of a Landsat Thematic Mapper image).



- Move the map window with the raster map Landuse to the side of the screen, by dragging its title bar.
- Make the ILWIS main window active. If the main window is visible on the screen, then you can just click any part of the main window. If the main window is not visible anymore, click the **Control-menu** box and select **Switch to**.
- From the Catalog, drag the raster map Tmb1 to the map window (click the map with the left mouse button and move it while keeping the left mouse button pressed, and release the mouse button when you are in the map window). The **Display Options - Raster Map** dialog box is opened for the raster map Tmb1. As you can see this map has a domain image.
- Click **OK** in the dialog box. The rest of the display options for satellite images is explained in chapter 6.

Now the raster map Landuse in the map window is substituted by the raster map Tmb1. It is only possible to display one raster map in a map window at a time. You cannot see two colors of two pixels on top of each other at the same time, but any raster map in a map window can be displayed together with many vector maps.



- Zoom in on a part of the raster map Tmb1 until you clearly see the different pixels of which this image is composed.

In a satellite image nearly each of the individual pixels has a different value than its neighbors, due to the different reflectance values of the earth surface as recorded by the sensor of the satellite.



- Read some of the pixel values in the map Tmb1 by clicking individual pixels with the left mouse button.
- Click the **Entire Map** button  in the button bar to display the entire map in the map window.

Image

The values that you can read from the map are whole numbers between 0 and 255. This means that whenever we work with satellite images (such as SPOT or Landsat) the domain of these images is called: image. *Domain Image* is used for satellite images, and scanned black and white aerial photographs. Pixel values in a band of a satellite image represent reflection values in a certain wave-length interval.

Layer management

Until now you saw a polygon map (Landuse) and two raster maps:

- Landuse (which was obtained by rasterizing the polygon map).
- Tmb1 (a satellite image).

Now we will have a look at the other types of vector data: *point maps* and *segment maps*. These maps will be displayed in the same map window. A map window can contain only one raster image, but it can be combined with several segment, polygon or point maps. Each of these maps displayed in a map window is called a *data layer*.



- Open the **Layers** menu and select the **Layer Management** command.
- Click **Help** button in the **Layer Management** dialog box. The **Help** is opened and displays **Help** on this **Layer Management** dialog box (this is called a *Context-sensitive Help*).
- Read the text, click in the picture in the **Help** to learn the meaning of the various check boxes and buttons in this dialog box.
- Close the **Help** window.
- Select the data layer Tmb1, and clear the check box **Info**. This means that you will not see the pixel values anymore when clicking the map.

- Click the Add Map button, and select segment map Drainage from the list box. Click OK.
The Display Options -Segment Map dialog box is opened. Select the check box Info. Click OK. The segment map Drainage is now added to the list.
- Click the OK button in the Layer Management dialog box.

Segment map Drainage is now displayed on top of raster map Tmb1.



- Click the drainage lines to learn the contents of the map Drainage.

Class domain



Class

As you can see the segment map Drainage contains three units: Drainage, Lake and Main river. These are the names of three classes. The contents of the segment map Drainage therefore, consists of classes. In ILWIS terms it is said that the segment map Drainage has a Class domain.

A class domain consists of a list of class names and optionally class codes.

Elements in a map (points, segments, polygons or pixels), using a certain class domain, are coded by the class names and codes presented in that class domain; elements without a class name are considered undefined. The same class name can be used for several points, segments, polygons or mapping units in one map. As you will remember from the previous exercise, the polygon map Landuse also had a Class domain, consisting of the names of the different land use types. Let us now add another data layer to the map window: a segment map containing digitized contour lines.



- Open the Layers menu, select Add Layer and select Segment map.
- Select the map Contour in the Add Segment Map dialog box. Click OK. The Display Options - Segment Map dialog box is opened.
- Select the check box Info. Click OK. The segment map Contour is now added to the map window.
- Click the contour lines displayed, and read their meaning. Note that you can also still read the meaning of the drainage lines, when you click them.

Value domain

Value

The contents of segment map `Contour` is different from that of the segment map `Drainage`. The contour lines are all coded according to their elevation. In this case the contents cannot be represented as classes, but as values. The map `Contour` therefore has the domain `Value`.

Value maps and value columns use a value domain. Elements in value maps (raster, polygon, segment, or points) or fields in value columns represent measured, calculated, or interpolated values. Values in a value domain are defined by a minimum and maximum value (the value range) and a precision; values smaller than the minimum or larger than the maximum are considered undefined.

Let us now look at a polygon map. We have already seen the polygon map `Landuse`, which has a class domain. We will now look at the polygon map `Citybl`, which contains the city blocks of a part of the city of Cochabamba.



- Open the Layers menu and select the **Layer Management** command. Make sure to clear the check box `Info` for segment maps `Contour` and `Drainage`.
- Click the **Add Map** button, and select polygon map `Citybl`. Click **OK**. The dialog box **Display Options - Polygon Map** is opened.
- Select the check box `Info`. Click **OK**.

Polygon map `Citybl` is now added to the list.



- Click the **OK** button in the **Layer Management** dialog box. Polygon map `Citybl` is now added to the map window. The polygon map `Citybl` only covers the lower right part of the map window.
- Zoom in on the lower right part where the polygon map `Citybl` is displayed.
- Click the different city blocks to know their content.
- Click the **Entire Map** button.

Identifier (ID) domain

ID

You will see that each city block is identified by a number. These numbers, however, do not represent measured, calculated, or interpolated values. Each city block has its own code and information from each individual block can be obtained from a table connected to the map. The polygon map `Citybl` has an **Identifier (ID)** domain. Domain Identifier can be used in maps when each element in a map is identified by a unique name, i.e. an identifier (ID); elements without an identifier

are considered undefined. Each item occurs only in principle in one place in the map. Attributes can be linked to the unique names, that is to the identifiers. The attribute table uses the same domain as the map. The attribute table contains additional information on the uniquely identified elements.

Domain *Identifier* is valid in point, segment, and polygon maps, and in rasterized point, segment, and polygon maps. Domain Identifier is advised for detailed maps on large scales such as 1: 500 for cadastral and infrastructural purposes.

Finally, let us have a look at a point map. The point map that will be displayed is showing the location of rainfall stations in the area (it is a fictive map, only meant for this *User's Guide* and not representing the real situation in Cochabamba).



- Open the **Layers** menu and select the **Layer Management** command. Clear the **Info** check boxes for all data layers.
- Click the **Add Map** button and select point map **Rainfall**.
- Click **OK**. The dialog box **Display Options - Point Map** appears.

In the dialog box there is an option to also show the names of the points. Please note that point map **Rainfall** also uses an **Identifier** domain. The domain of a map is always indicated on the top of a **Display Options** dialog box.



- Select the check box **Text**. Further, click the button **Font**: select the **Font type: Arial; Font Style: Bold** and **Size: 10**. Click the **OK** button in the **Font** dialog box. Select: **Text color: Red**.
- Click the button **Symbol**. In the next dialog box, select **Symbol Type: Simple; Symbol: Square; Size: 10; Fill color: Green; Line width: 1, and color: Red**. Click **OK** in the **Symbol** dialog box.
- Click **OK** in the **Display Options- Points** dialog box.
- Click **OK** in the **Layer Management** dialog box. The point map is now added to the map window.

Notice that the names of some of the rainfall stations are only partially shown, because those stations are situated just outside the area covered by the satellite image.



- Zoom in on the upper left corner of the map. Make sure that the area you define for zooming extends even outside the left part of the map window. Now an area is displayed in which you see contour lines,

drainage lines and a rainfall station (Cerro MachaMach), but no information from the satellite image or the polygon map Citybl.

- Practice some more with the **Layer Management** options. Add some other polygon and segment maps, and see which domain those maps have. Practice also to change the order of the layers by dragging a layer up or down in the list. You can also choose not to display a data layer by clearing its check box **Show**.
- When you have finished, close the map window.

Summary: combining vector and raster maps

In this exercise you learnt the following:

- ILWIS uses two types of maps: vector maps and raster maps.
- Vector maps in ILWIS are either point maps, segment maps or polygon maps.
- Raster maps can be obtained either by rasterizing vector maps, or by importing them from other sources.
- ILWIS allows the display of many data layers in the same map window. A map window can display only one raster map, but together with many different point, segment or polygon maps.
- The contents of a map is defined by the domain of the map.
- The same domain can be used by a vector map, a raster map and a table.
- The following domains are possible:

Class domain: a list of class names

Value domain: measured, calculated, or interpolated values

Image domain: reflectance values in a satellite image or scanned airphoto between 0 and 255. This is a special type of value domain.

Identifier (ID) domain: a unique code for each item in the map.

- Vector and raster maps can be displayed simultaneously in a map window, using the option **Layer Management**.

The way in which various data layers are displayed in a map window (point, segment, and/or polygon maps, and/or a single raster map) can be manipulated via the **Layer Management** dialog box: you can select the layers to be displayed; the layers from which you want to obtain information. You can change the display options of a layer, and you can add or remove layers.

2.2 Vector data in ILWIS

In the previous exercise you investigated the relation between vector and raster data, and different types of domains. Now we will look more in detail to vector data and find out which files are needed in order to show them.

At this point, it may be useful to explain that ILWIS uses different types of objects:

- *Data objects*. Raster maps, polygon maps, segment maps, point maps, tables and columns are called data objects. They contain the actual data.
- *Service objects*. Service objects are used by data objects; they contain accessories that data objects need besides the data itself. Domains, representations, georeferences and coordinate systems are called service objects.

In this section we will look to these service objects that are needed for vector data.

Properties

In section 1.3 we have looked at the polygon map `Landuse` and you saw that:

- the map has coordinates,
- the map has a domain: an explanation of the contents of the map.

These things are defined in service objects. You can see which service objects are used by the data object (in this case polygon map `Landuse`) by viewing the *properties* of the data object.



- In the Catalog, click polygon map `Landuse` with the right mouse button. A Context-sensitive menu appears. Select the **Properties** command.

A dialog box is opened titled: **Polygon map properties**. The dialog box contains the information which ILWIS needs to use the map `Landuse`. There are three important lines in this dialog box:

- **Coordinate system** `Cochabam`
- **Domain Class** `Landuse`
- **Attribute table** `Landuse`

In section 1.3 you already displayed the domain of map `Landuse`. You also saw that the attribute table `Landuse` uses the same domain (domain class: `Landuse`). Let us now have a closer look at the coordinate system.

More about coordinate system object

In section 1.3, you learnt that the status line in a map window displays the coordinates of a map.



- Close the Polygon Map Properties dialog box: you can click the Cancel button.
- Open polygon map landuse by double-clicking it in the Catalog.
- Click OK in the Display Options dialog box. Polygon map landuse is displayed. Maximize the map window.
- Move the mouse pointer through the map and verify the display of coordinates in X and Y values, as well as in degrees, minutes and seconds on the status line.
- Close the polygon map landuse.



The information on the X and Y coordinates of a map is stored in a service object: the *Coordinate system*. A coordinate system is a service object for point, segment and polygon maps, and for georeferences (they are used for raster maps and will be treated in the next section). A coordinate system contains information on the kind of coordinates you are using in your maps; you may for instance use user-defined coordinates, coordinates defined by a national standard or coordinates of a certain UTM zone.



- Activate the main window, and click the **Customize Catalog** button in the button bar in the main window (the button with the spectacles).
- In the **Customize Catalog** dialog box select only the **Coordinate system**. Select the check box **Include System Objects** and click OK. The Catalog now only shows coordinate systems. One coordinate system was made especially for the data set: Cochabam. The other two, Lat Lon and Unknown, are standard ones.
- Open coordinate system Cochabam. by double-clicking it in the Catalog.

A dialog box is opened, titled **Edit Coordinate System Cochabam**.

A coordinate system defines the possible minimum and maximum X's and Y's that can be used in a map, along with other information on map datum, projection etc.

Normally locations on the earth's surface are shown in geographic coordinates.

The *Geographic Coordinate* system is a spherical coordinate system composed of parallels of latitude and meridians of longitude (see figure 2.2).

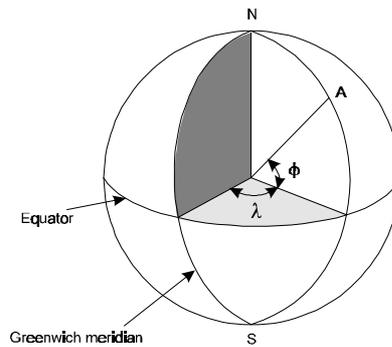


Figure 2.2: Geographic coordinate system: the location of A is determined by the longitude (λ) and the latitude (ϕ)

The surface of the earth is curved but this should be presented in maps as a plain surface. To correctly do this, you need a special *map projection*. With the help of a map projection, geographic coordinates are converted into a metric coordinate system, measuring the X and Y directions in meters. Practically every country in the world has its own coordinate system, based on a certain projection. One of the most used projections is the so-called UTM.

The *Universal Transverse Mercator* is an international metric coordinate system which extends around the world into 60 zones each covering six (6) degrees longitude. Because of the small area covered by each zone, a high degree of accuracy is possible. The UTM coordinates are metric coordinates.

In ILWIS you can digitize a map either in metric or in geographic coordinates. The conversion from geographic to metric coordinate is done via the map projection. For example, it may be that you obtain point data using GPS (Global Positioning Systems), which contain data in geographic coordinates.

ILWIS has two standard coordinate systems:

- Latlon: Geographical coordinates in degrees
- Unknown: this is the coordinate system you can use if you do not want to specify the projection used in your maps. In that case you define your own local coordinate system; only the minimum and maximum X and Y values are stored and there is no need for projection parameters.

In most situations, however, it is better to generate a coordinate system for the area where you are working, especially when the data set that you prepare is intended to be used for a longer period, and may have to be combined with maps from other sources, which may be in another projection.

In principle, all the maps that are used of a certain area should have the same coordinate system. ILWIS expects that all maps that you want to superimpose or display together in one map window have the same coordinate system. Displaying maps with different coordinate systems is possible if projection information is

available for both coordinate systems. Then the coordinates of a vector map are transformed on-the-fly according to the coordinate system that the map window is using at the moment. Of course, you can use the **Transform operations** to permanently transform coordinates (see chapter 4).



- Note that the projection in the coordinate system Cochabam is UTM, with the ellipsoid International 1924, and the datum Provisional South America 1956, area Bolivia.
- Click the OK button in the Edit Coordinate System dialog box.
- Click the **Customize Catalog** button in the button bar in the main window and select all objects. Clear the check box: **Include System Objects**. Click OK.

The use of projections is further discussed in chapter 4.

More about domain objects

Besides a coordinate system which contains information on the location of the items in a map, each vector data object in ILWIS has a *domain* which contains information on what the features in the map represent. In the previous section we have already seen the most important types of domains in ILWIS. A domain describes the possible contents of a map, a table or a column.

ILWIS uses domains extensively and for many operations you should decide in advance about the domains for the resulting maps: a *class map*, an *ID map*, or a *value map*. This enables you to have a direct control over your data types, e.g. having the domain as a separate object you can have a direct look at what kind of data are stored in your maps.



In several other raster based GIS systems, and also in the previous DOS versions of ILWIS, you always had to remember what each pixel represented. For example, you had a land-use map with 4 classes (1 = agriculture, 2 = forest, 3 = grassland, and 4 = shrubs) and you wanted to find the areas which are suitable for further development. Suppose that these would be the areas with shrubs. In order to show the suitable areas you had to use a calculation formula (this will be explained in depth in chapter 7 and 8) which had the following form:

```
Result:=iff(Landuse=4,1,0)
```

This formula meant: if the land use map has a pixel value of 4 (shrubs), then the resulting map will have a value of 1 (meaning that it may be used for further development), and otherwise a value of 0 (not suitable).

In ILWIS for Windows you use domains and you no longer need to know the relation between the pixel values and the classes, since ILWIS keeps track of that. In ILWIS for Windows, your formula reads:

```
Result:=iff(Landuse="Forest","Suitable","Not suitable")
```

The resulting map is also a class map, so you directly know what the units mean.

More about class domains

Lets us look now at some other advantages of domains.



- Open the domain of polygon map Landuse by double-clicking the domain Landuse.

The Domain editor Class/ID is opened. The domain is actually a list of names. The domain editor allows you to add new items to the domain, and to change or delete existing ones. One should be careful, however, with deleting domain items, since this may cause problems in the maps and tables that have this domain. For example, if you delete item Forest, there will be polygons in the map Landuse, with a code that is not anymore in the domain. The forest unit in the map will be represented as *undefined (?)* and will also loose its color. You should always take care that the domain contains all the items that can occur in a map or a table.

A domain may contain more items, however, than are present in the map. This is very useful if you have different maps dealing with the same topic, for example multi-temporal land use maps of a certain area, or various geological maps with the same legend.

What is the advantage of the domain approach? Let us give an example:

- ! Suppose you have two land use maps of different regions in the same country. Suppose in one map you have land use types agriculture, grassland and shrubs, and in the other map agriculture, forest and shrubs. Under previous ILWIS versions, pixel values of your units might be agriculture = 1, grassland = 2 and shrubs = 3 for the first map, and agriculture = 1, forest = 2 and shrubs = 3 in the second map. So, pixel value 2 is once forest and once grassland, which is very confusing! However, if you use one domain containing all the possible land-use classes, you do not need to care about pixel values of separate maps since all the necessary information for all your land-use maps is stored in one table, the domain, containing the following class names: Agriculture, Forest, Grassland, Shrubs.
-



- Open the Edit menu in the Domain editor, and select the Properties command.

The Domain Properties dialog box appears, which describes among others what this domain is used for.



- Click the button: Used by.

You see a list of the objects that are using this domain Landuse: polygon map Landuse, raster map Landuse, table Landuse, and representation Landuse. A representation defines the colors with which the domain items can be displayed. We will look at that later.

In ILWIS, a raster map and a polygon map dealing with the same topic are usually linked to the same domain, as well as their attribute table.



- Click the OK button in the dialog box domain Class Landuse.
- Click the Cancel button in the dialog box Domain properties.
- Close the domain Landuse.
- Open another class domain: Geom.

This is the domain of the Geomorphologic map. The list of classes in this domain looks a bit different to the one of the land use classes. You see first a code, then a colon (:) and then a class name. Since some of the names for classes can be very long, ILWIS allows the use of abbreviations, or *codes* for the domain items. This is very handy when you want to do calculations with maps, or columns. Instead of having to write a formula like:

```
Result:=iff(Geom="Accumulation of debris on slope","Possible  
landslide","Not a possible landslide")
```

You can now write:

```
Result:=iff(Geom="Ad","Possible landslide","Not a possible landslide")
```

Codes can be entered, edited and deleted with the Add, Edit and Delete commands on the Edit menu.

More about Identifier domains

Let us now look in more detail at Identifier domains.



- Close domain Geom.
- Open identifier domain Citybl.

Domain Citybl is an identifier (ID) domain. It is linked to the polygon map with the city blocks, as well as to the table Citybl. Domain Citybl contains the cadastral codes of the city blocks of the central part of Cochabamba city. In this case the identifiers are numbers.



- Close domain Citybl, and open domain Rainfall.

Domain rainfall is also an identifier domain (you can check this for instance by opening the Properties of the domain), but in this domain the individual items are the names of the rainfall stations. It is not a class domain, since a name is only valid for one rainfall station alone. If it would be related to more rainfall stations the system would not know to which station the data in the corresponding table Rainfall are referring.



- Close the domain Rainfall.
- Open the domain Height.

More about value domains

Finally we will look at value domains. Domain Height is used for the segment map Contour, which contains contour lines. A value domain looks quite different from a class or ID domain. It is not a list of names or codes, but it is a description of the values: Height in meters.

For the Cochabamba area, height values range from 2500 to 4600 meters. These are the minimum and maximum values of the domain Height.

The precision of a value domain determines the increment between two possible successive values. The precision determines for instance the number of decimals that are used to store the values. When you use a precision of 0.1 the values are stored with 1 decimal. You can also store values with a certain step size, for example only the values ending on 5,10,15,20,25 etc. In that case you use a precision of 5. You can use any value for the precision, as long as it makes sense

with respect to the minimum and maximum value, and the meaning of the values. For instance, it would be quite useless to use a precision of 0.001 for the domain height, since the altitude of the contour lines are in meters on the topographic map from which they were digitized.

It is important to keep in mind, that if you use very small values for the precision (for example 0.0000001), the size on disk of the map can become very large (up to 8 bytes per pixel). This will also lead to slower performance of the system and to disk space problems. So you should use a precision which is useful for the data you are working with.

The domain Height is a so-called *user-defined value domain*. It was created for the demo data set. User-defined value domains are useful when you have a series of maps for which you need the same range of values and the same precision. You do not have to define a new domain for each map, and even more important, you can show these maps with the same representation. User-defined value domains can have their own representation value.

For example, if you are working with digitized contour maps of different areas and you want to display them in the same colors.



- Close the domain Height.
- Click the domain Height with the right mouse button, and select the command Properties from the Context-sensitive menu. The Domain Properties dialog box is opened. As you can see the domain Height also has a representation called Height.
- Close the Domain Properties dialog box.

There are also a number of system-defined value domains. These value domains are defined by the system, and are stored in the subdirectory c:\ilwis21\system. It is not possible to change these system-defined value domains. The system-defined value domains can also be displayed in the Catalog.



- Click the Customize Catalog button in the button bar of the main window.
- Select only the domains and click OK. Now you will see only the user-defined domains in the Catalog. These have been made especially for the data set.
- Click again on the Customize Catalog button. Do not select any object types, but select the check box Include System Objects at the bottom of the dialog box. Click OK.

Now the listed domains also include the system-defined domains. The most important value domain is the one called *value*. In fact you can make all other value domains with this domain. The others are given to prevent you from needing to define domains too often. Let us look at some examples.



- Open the domain `Value`. As you can see the minimum value that can be stored is `-9999999.9` and the maximum is the positive equivalent of that. So it is safe to conclude that you can store all possible values which may occur in maps and tables in the value domain. Close the domain `Value`.
- Open the domain `Perc`. This is a special value domain which is intended for storing percentage values with a default value range between 0 and 100, and a precision of 0.01 (two decimals). Close the domain `Perc` again.
- Open the domain `Bool`. This is the Boolean domain, which only allows two situations: 0 (false) or 1 (true). Apart from these two there is also the possibility for undefined values, indicated as a question mark.
- Have a look at some of the other system-defined value domains. When you have finished, select all object types again via the `Customize Catalog` button, and clear the check box `Include System Objects`.

There are some domain types that we have not yet treated so far, because they are specific for certain operations. They will be discussed later on in this book. A complete overview of domains can be obtained from the Help. Search in Help with the keyword: *domain*.

Representation



The last one of the *service objects*, that is needed for vector data, besides a coordinate system and a domain, is a *representation*. A representation contains the information that determines how your spatial data should be represented on screen or in a printout. Representations are service objects defined by the domain. The domain determines what is in a map (or table or column), and the representation defines which colors these domain items should have.

There are different types of representations, depending on the type of domain.

Representation Class

Class domains have the most straightforward representation. A map with a class domain should have a representation, otherwise it is impossible for ILWIS to know

how to display it. Representations for class domains are always user-defined. They should be created by the user.



- Open the representation Landuse.

Each item in the domain has a color assigned to it. The colors can be edited by clicking on an item, and then changing the red/green/blue amount by dragging the scroll bars or by double-clicking an item.



- Practice with the changing of colors, by selecting a unit and changing the red/green/blue scroll bars. Also change the colors of some units by double-clicking them and select a new color from the list box.
- Select from the **Options** menu the command **Segments**, and look at the representation. After that, select from the **Options** menu the command **Symbol**. Since we can use the same domain, and the same representation for raster, polygon, segment and point maps, these menu options are used to see what a line would look like, or a point, or an area.
- Practice also with the different menu items.
- After that close representation Landuse.

Identifier domains (ID) do not need a user-defined representation.



- Double-click the polygon map Citybl. In the Display options - Polygon map dialog box you see several options: Single color, or Multiple color (7, 15 or 31). Select 31 multiple colors. Click OK.

For an identifier map such as the city block map it would be useless to spend much time in defining the color of each individual item by hand, since different colors do not represent any thematic difference. The colors are only used to make a differentiation between the units. Therefore, a user-defined representation is not needed.

Representation Value

Maps with a Value domain have a representation Value.



- Close the polygon map Citybl.
- Open the representation Height.

A representation for a value domain is defined on the basis of a number of limiting values between which the color changes from one color to another in a number of user-defined steps. In this example the limits are: 2500 (the minimum of the domain), 3500, and 4600 (the maximum value of the domain). You can insert as many intermediate limits as you wish.



- Select the Limit 3500.
- Open the Edit menu and select Edit Limit. Change the color to Lawn green. Click OK.
- Open the Edit menu and select Stretch Steps. Change the value to 100. Click OK.
- Select the word Stretch in the editor. If you double-click it the word changes to Upper. This means that all values between the limits are displayed in the color of the upper limit. Double-click again, and it changes to lower limit.
- From the Edit menu choose Insert Limit. A dialog box appears. Insert a new limit: 4000, and select a color: Gold.
- Close the representation Height.

You can make two types of representations for value maps:

- a *representation Gradual*, in which all boundaries appear in percentages. If you use it on a value map, the percentages of the frequency distribution of pixel values, are taken from the histogram.
- a *representation Value*, in which all boundaries appear in values (such as in the example of the representation height).

As was the case for value domains, there are also a number of system-defined value representations available. These representations all use percentages as limits, so that they can be used on different types of value ranges.



- Click the Customize Catalog button in the button bar in the main window and select only the representations. Select the check box Include System Objects. Now you will see only the representations,

including the system-defined ones.

- Open some of these representations to see how the colors are assigned to different percentage values.
- Click again on the **Customize Catalog** button. Select all the objects from the list. Clear the check box **Include System Objects**.
- Double-click the segment map **Contour** (this one has a value domain). Select a system-defined representation, for example: **Red**. Click **OK**. Now the contour lines are displayed according to that representation.
- Click with the right mouse button in the map viewer. You will see a **Context-sensitive** menu. Select the last line: **1 seg Contour**. This is a shortcut to get the **Display Options -Segment Map** dialog box back.
- Select another system-defined representation, for example: **Pseudo**. Click **OK**. Repeat this for a number of other representations.
- After finishing, close the map **Contour**.

Summary: Vector maps

Vector data in ILWIS can be point, segment or polygon maps. They are called data objects (together with raster maps and tables). They need a number of service objects:

- **Coordinate system.** Vector maps always need a coordinate system. A coordinate system may have information on the map's projection, ellipsoid and datum.
- **Domain.** Vector maps always have a domain. Class and ID domains are always user-defined. Value domains can be user-defined, or system-defined. A value domain specifies the range of values (minimum and maximum) and the precision.
- **Representation.** Class domains always need a user-defined representation Class. Vector maps with ID domains cannot have a user-defined representation. They can only be displayed in 1, 7, 15 or 31 different colors. Vector maps with a value domain can have two types of representations:
 - representation Gradual, using percentage values as limits
 - representation Value, using actual values as limits.

See also chapter 13 if you want to know more about representations.

2.3 Raster data in ILWIS

In the previous section we have looked how vector data in ILWIS is stored, and which service objects are needed. We have looked at the coordinate system, the domain, and the representation. These service objects are also needed for raster data, together with another one: georeference. In this exercise we look only at that specific service object, used only for raster maps.

In ILWIS raster maps can be of different origin:

- They are rasterized vector maps (point, segment or polygon maps);
- They are imported images, either satellite images, scanned airphotos or imported raster maps from other sources;
- They are result maps from GIS or image processing operations.

All raster maps contain information in discrete cells or pixels, which are ordered in rows and columns. To relate these pixels to map coordinates a georeference is needed.

Georeference



A georeference stores the relation of the rows and columns of the raster map(s) with ground-coordinates. If you work with several raster maps of the same area, and you want to overlay them (e.g. with the cross operation), it is important that they have the same number of rows and columns, and the same pixel size. The pixels in both maps refer to the same position on the ground. Therefore all raster maps of the same area should have the same georeference. Let us look at a georeference.



- Open georeference Cochabam. If you don't see it in the Catalog, click the **Customize Catalog** button, and select all object types.

Georeference Cochabam is what we call a *georeference corners*. By defining the X and Y values of the corner of the raster map, ILWIS is able to find the correct coordinates of each pixel. This assumes that the raster map is oriented to the North. The X-coordinate increases from the left column to the right column, and the Y - coordinate increases from the lowest line to the upper line. The X and Y values relate to the coordinate system that we have seen in the previous exercise.

The georeference has information on the number of rows and columns in the map, the coordinate system, the minimum and maximum X- and Y-coordinates of the corners of the map, and the pixel size.

The georeference corner type is used for those maps which are converted from vector to raster.



- Use the Windows calculator (you can find that in the Accessories window of the Windows program manager), and subtract the minimum X value from the maximum. Then divide the resulting value by the number of columns. The resulting value is the pixel size.
- For such calculations you can also use the command line of the main window. The command line can also be used as a pocket calculator, when you put a question mark before the calculation. Position the mouse pointer on the command line and type an expression such as:
`?(maxX-minX)/pixel size.`
- Close the georeference corners dialog box.

For satellite images and other imported images, which may not be North-oriented, we need to have another georeference: a *georeference tiepoints*.



- Open the georeference Tmgeo. First a window is opened, with the title: **Georeference Editor**. Then a **Display Options - Raster map** dialog box is opened. Click **OK**.

You are now in the **tiepoint Editor**. In the satellite image you will see seven points displayed, which are related to the seven points in the table of the georeference tiepoint editor. For each point the X- and Y-coordinates are given, along with the row and column locations of these points in the image. On the basis of these coordinates using an affine transformation, all rows and columns will be assigned X-and Y-coordinates, with a certain error, expressed as sigma. In Chapter 6 (image processing) we will treat this in much more detail.

What is important to note here is that the image is not oriented perfectly to the North. The image is distorted with respect to a topographic map. When you want to overlay images in combination with North oriented maps, you need to remove the distortion of the image by resampling it (see chapter 6).



- Close the table of the georeference tiepoints editor. Do not close the map.
- Open the **Layers** menu and select **Add Data Layer, Segment map**. Select segment map **Contour**. Click **OK**, and again **OK** in the **Display Options - Segment Map** dialog box. The contour map is now displayed on top of the image.



- Press the **Zoom In** button and select an area which is larger than the map window (start the zoom rectangle outside the upper left corner of the map, and extend it beyond the lower right corner). Now you will see that the contour lines extend outside the map, and that they are displayed in a rotated way. From this you can conclude that the image is distorted and not oriented to the North. In order to make the image North oriented it should be resampled (see chapter 6).
- Drag and drop raster map `Landuse` in the map window. Raster map `Tmb1` will be replaced by raster map `Landuse`. You will see that the display of the contour lines now changes. They are now displayed in the normal, North-oriented, way.
- Close the map window.

Summary: Raster maps

- Raster maps have the same service objects as vector maps (with respect to domain, representation and coordinate system).
- There is one object specific for raster maps: the georeference. A georeference stores the relation of the rows and columns of the raster map(s) with X and Y ground-coordinates.
- There are two types of georeferences for raster maps that we have discussed here:
- **Georeference corners**, used for raster maps made from rasterizing vector maps, and the resulting maps from operations on these maps.
- **Georeference tiepoints**, used for uncorrected (satellite) images.

In chapter 10 (dealing with Digital Elevation Models) you will see that there is yet another georeference type, called **Georeference 3-D Display** which is used to define the parameters for constructing a 3-dimensional view.

2.4 Linking spatial and attribute data

In the sections 2.1, 2.2 and 2.3 we have looked at the way maps are stored in ILWIS, either as point, segment, polygon or raster maps. We have also looked at the service objects that contain information about these maps, such as the coordinate system, the georeference, the domain and the representation.

In section 1.3 we have also seen that spatial data (maps) can be linked to non-spatial, or attribute data (tables). In order to be able to link a map with a table, the contents of both should be the same. In other words, they should have the same domain. Let us briefly check this again.



- Open domain Landuse. Open table Landuse. Open polygon map Landuse.
- Move the three windows in such a way that you can see all three of them next to each other.
- Click several units in the polygon map to know their meaning. Note that the same class names appear in the map, in the table and in the domain.
- When finished, close all three windows.
- Click with the right mouse button on the table Landuse in the Catalog, and select **Properties**. The **Table Properties** dialog box is opened. Note that the domain of the table is **Domain Class: Landuse**. Click the **Cancel** button.
- Click with the right mouse button on polygon map Landuse, and select **Properties**. The **Polygon Map Properties** dialog box is opened. Note that this map also uses the same domain. Cancel the **Properties** dialog box.

So we can conclude here, that *maps and tables can be linked if they have the same domain*. However, there is a very important restriction that should be mentioned here. It is only possible to have a table with a **Class** domain, or with an **ID** domain. Tables with a value domain are not possible. The reason for that is:

- Suppose you have defined a value domain with minimum 0 and maximum 10000, and a precision of 0.01. In that case, a table connected to the map should have a very large number of records ($10000 \times 100 = 1,000,000$). So the number of records and the extreme size of a table becomes a problem.
- A more important reason, however, is that value maps themselves do not contain information to which attributes can be linked. They are in fact attribute maps, and not thematic maps. A landuse map (with a class domain) can have an attribute table, in which for each land use type information is given, such as the biomes production per hectare, the value of the land per hectare etc.

- The city block map (with domain ID) can have an attribute table in which information is stored for each city block, such as the number of people living in the block, the number of houses, public buildings, shops etc.
- A value map, on the other hand, such as the segment map contour, contains values, representing a certain phenomena (in this case terrain elevation). It is difficult to imagine which attributes could be linked with such information, other than purely statistic information (length of each line).

The linking of maps with attribute tables is useful if you want to substitute the units of a map with an attribute value from the connected table (a process called renumbering). In that case you generate an attribute map.

Although value maps can not be linked to attribute tables, you can generate tables from value maps. The most important ones are cross tables and histograms. These are not used, however, to renumber the value maps. It would make no sense to replace the DN values (reflectance values) of a satellite image with the area attribute from its histogram.

The Pixel Information Window

ILWIS has a special tool with which you can examine both maps and tables linked to these maps simultaneously: the *pixel information window*. The pixel information window is used to interactively inspect coordinates, class names, IDs or pixel values, in one or more maps and attribute values of map-related tables. The pixel information window shows information at the mouse pointer position in a map window.



- Click the **Show Pixel Information** button in the button bar of the main window.

The pixel information window is opened (figure 2.3). It contains many of the items that we have seen before in the map- and table windows, such as a control-menu box, title bar, status line, menu bar etc. Some special items in this window are:

Top left button: the upper left gray button in the pixel information window can be used to open the **Customize PixellInfo** dialog box.

Information viewer: where the information of the selected data objects for the actual position of the mouse pointer in a map window is displayed.

Menu bar: the menus of the ILWIS pixel information window are: **File**, **Edit**, **Options**, and **Help**.

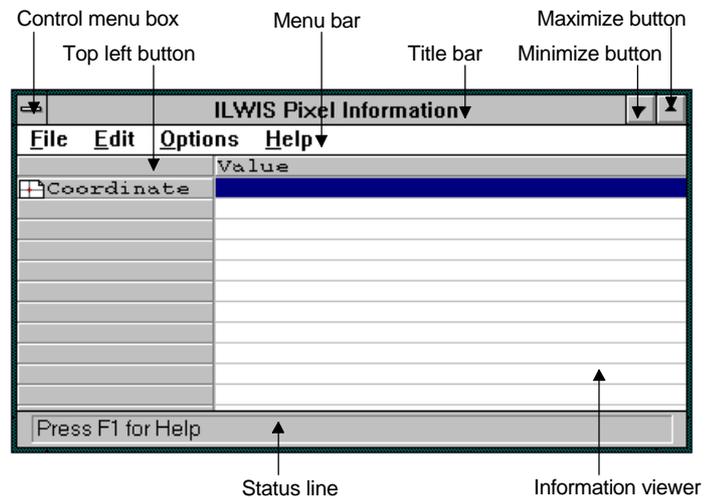


Figure 2.3: The ILWIS Pixel Information window



- To get more information on the pixel information window, select the menu item **Help**, and **Getting help** on this window, from the drop down menu.

If you want to display information on maps and tables in the pixel information window, you must make sure that the map is displayed in a map window, and the map and the table are actually linked.



- Click with the right mouse button on the polygon map Landuse.
- Select **Properties**. The Polygon Map Properties dialog box is opened.
- Make sure that the check box **Attribute Table** is selected, and the table Landuse should be in the list box.
- Click **Cancel** to close the dialog box.

We will now display the map Geom (geomorphologic map) and use the pixel information window to read the information from this map, along with that of the Landuse map.



- Open polygon map Landuse.
- Drag the polygon map Landuse from the Catalog and drop it to the

pixel information window.

- Add polygon map Geom to the pixel information window in the same way.
- Position the mouse pointer on the map Landuse in the map window.
- You can read the corresponding information for the mapping unit of the map Geom, along with that of the mapping unit in the map Landuse at the same location, in the pixel information window.
- Move the mouse pointer on the map to display the information continuously.

It is possible to hide or to show maps and attributes in the Pixel Info window:



- Click the Top left button in the Pixel Info window.

The **Customize Pixel Information** dialog box appears which shows a list of maps, tables and columns that are presently visible in the pixel info window. With the check boxes you can define if the information is displayed when working with a mouse pointer or with a digitizer cursor, and whether the information should be displayed continuously (while moving over the map), or only when you click a unit. The List box **Show value of** can be used to select which of the maps, tables or columns you would like to see. You can select or deselect them by clicking on them. Selected items appear highlighted.



- Deselect all the columns of the table Landuse, and click OK.
- When you now move with the mouse pointer over the map you will see that no information is given on the table Landuse.
- Close the pixel information window by double-clicking the Control menu box.
- Close the map window.

Summary: Linking spatial and attribute data

Maps and tables can be linked only if they have the same domain. Only maps with domain Class or ID can be linked to attribute tables. Value maps cannot be linked to tables. Information from maps and their attribute tables can be displayed using the pixel information window.

2.5 How ILWIS defines links: properties and dependencies

As shown in preceding sections, different data objects (point, segment, polygon and raster maps) and service objects (coordinate system, georeference, domain, and representation) are required to form a map in ILWIS. How does ILWIS know which objects are linked ?

The *properties* show which service objects are used for a data object or another service object. In section 2.1 you saw that for the display of vector maps, you need to have a coordinate system, a domain, and a representation (depending on the domain). In section 2.2 you saw that a raster maps requires the same objects, plus the georeference. In section 2.3 it was shown that class and ID maps can also be linked to attribute tables.

In figures 2.4 and 2.5 a schematic representation is shown of the *properties* of vector and raster maps. The properties of vector maps with a class domain, identifier and value domain are shown in 2.4.

The objects indicated in the shaded boxes are the *data objects* (point, segment, polygon maps and tables) and the objects outside of these boxes are the *service objects*. The arrows indicate the relations between the objects. For example, in figure 2.4, a domain is used by a vector map (either a point, segment, or polygon map), by a table, and by a representation.

All the properties of an object can be viewed from their **Properties** dialog box.



- Click with the right mouse button on the Class domain Geom, and select **Properties** from the Context-sensitive menu. The **Domain Properties** dialog box is opened.
- Click the button: **Used by**.

A list of the objects is shown that use this domain Geom: polygon map Geom, table Geom, and representation Geom.



- Click **OK**.
- Close the **Domain Properties** dialog box.

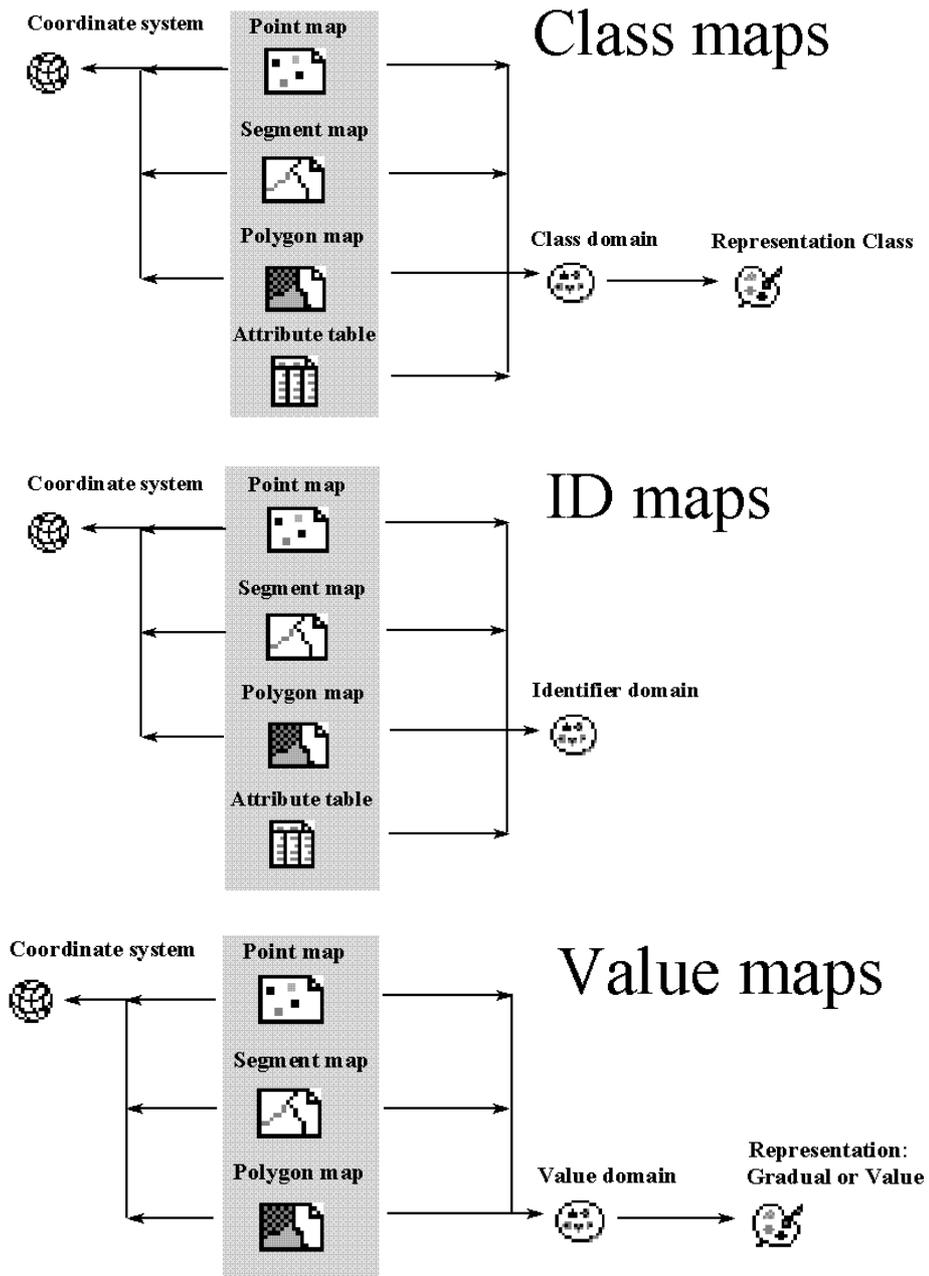
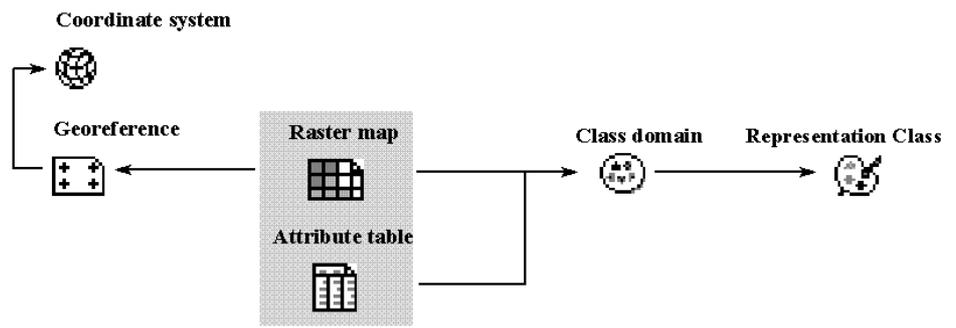
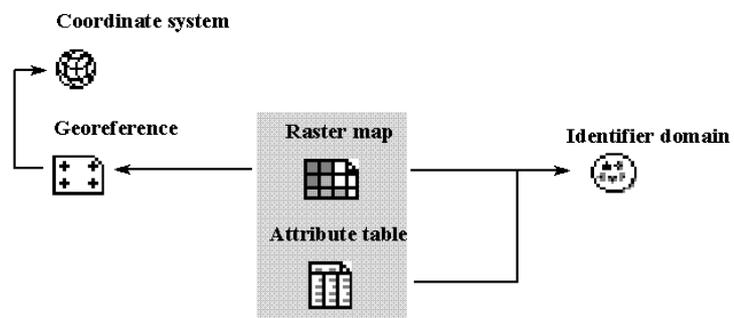


Figure 2.4: The properties of vector maps with a Class, Identifier and Value domain in ILWIS. The data objects are shown in a shaded box

Class maps



ID maps



Value maps

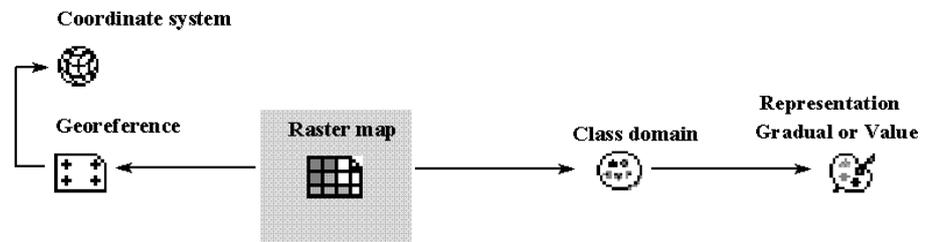


Figure 2.5: The properties of raster maps with a Class, Identifier, and Value domain in ILWIS. The data objects are shown in a shaded box

To understand these figures it is good to look at the properties of several maps.



- Click with the right mouse button on domain `Cityb1` and select **Properties** from the context-sensitive menu. `Cityb1` is an ID domain, used for polygon map `Cityb1`, and table `Cityb1`. Click the **Used by** button. Compare the result with figure 2.4. Close the **Properties** dialog box.
- Repeat this for value domain `Height`. This domain is used by the segment map `Contour`, and by the representation `Height`. Compare with figure 2.4.

The properties of tables and columns are schematically represented in figure 2.6. A table should have a class or ID domain. A table contains a number of columns. Each column also has a domain.

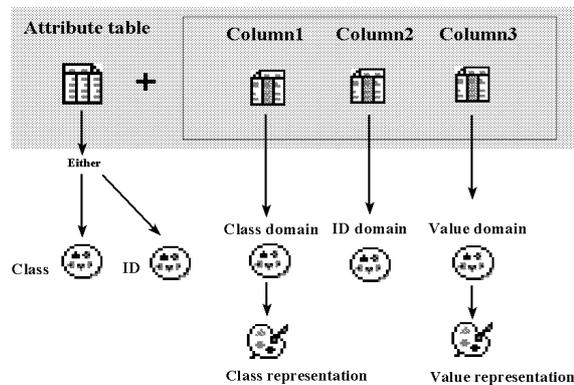


Figure 2.6: The properties of tables and columns



- Open table `Landuse` (**Class** domain).
- To view the **Properties** of the table: open the **Edit** menu and select **Properties**.
- To view the properties of a column: double-click a column button in the table window.
- Close the table `Landuse`.

- Repeat this for the table `Citytbl` (ID domain).
- Close the table `Citytbl`.

Dependencies

The relationship between a data or service object and several service objects, is called the *properties* of the object.

When maps are used to create other maps, for instance by performing an operation or executing an expression, then this operation or expression and the input map name(s) are stored inside the new map. This is what we call *dependency*.

Output maps thus know how they are created and on which input maps, tables or columns they depend. Such output maps are called *dependent data objects*. The same applies for tables and columns. The operation or expression stored inside the dependent data object is called the definition.

Let us take an example: *polygon map* Landuse was used in combination with the *georeference* Cochabam to generate a *raster map* Landuse. This is shown schematically in figure 2.7.

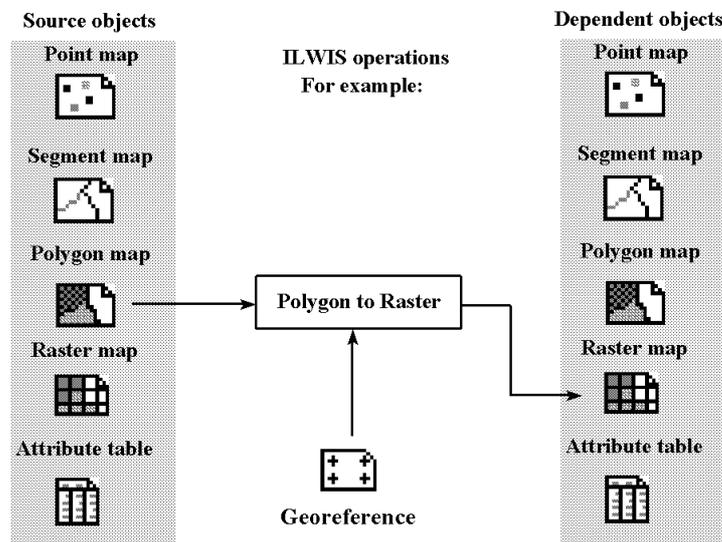


Figure 2.7: The dependency link between objects. The raster map is made from the polygon map and the georeference, using the *Polygon to Raster* operation

The dependent raster map, which is the result of the ILWIS operation Polygon to Raster, contains information on the source objects from which it was created. In

this case a polygon map and a georeference. Let us look at an example from the data set.



- Click with the right mouse button on raster map Landuse., and select **Properties** from the context-sensitive menu. The **Properties** dialog box is opened. The title of the map is: **Dependent Raster Map Landuse.**

The dialog box gives you, among the service objects for this map, the information on how the map was made:

- the expression by which this map was made:

```
MapRasterizePolygon(Landuse.mpa,Cochabam.grf)
```

The operation MapRasterizePolygon is shown in front of the brackets.

- The objects which are used to generate this map are shown between the brackets:

```
Landuse.mpa: polygon map Landuse
```

```
Cochabam.grf: georeference Cochabam
```

Why does ILWIS keep track of the objects that are used to generate another one? When you update input source maps by editing, the dependent output maps which use these source objects can be easily recalculated.

Note that the last line above the button **Release Disk Space** says: **Object is up to date**. This means that none of the source objects have been updated since the dependent raster map was generated.

It is important to keep in mind that the maps are not automatically updated. ILWIS does keep track of the dates and times at which the source objects were modified. If one of the source objects was modified, a message will be displayed. Only when you click the button **Check up to date**, ILWIS will ask you if you want to recalculate the map.

There are some other important advantages of the use of dependencies:

- For maps that are created via the **Map calculation** operation (discussed in chapters 7 to 9) you can change the definition of dependent data objects; so you can do the same operation again with different parameters without having to create a new map.
- You can release disk space: it is possible to delete the files that contain the actual data values (e.g. **.MP#** files), as long as the object definition files (e.g. **.MPR** files), which contain the definition to recalculate the dependent data objects remain.



- Click the button **Release disk space**. Click **Yes** in the next dialog box. The **Raster map properties** dialog box closes.

Now the actual data file of the raster map is deleted. Only the definition of the map remains, so the map can always be recalculated. In this example the raster map had 932 lines and 637 columns. Each Pixel was stored as 1 byte. So you save $932 * 637 = 593684$ bytes of disk space, without losing any actual information.



- Open the **Properties** dialog box of the raster map **Landuse** again.

Now you will see that the contents of the dialog box have changed. Note the size of the object: 784 bytes. This amount of storage space is needed for the information of the definition file. The button **Release disk space** is now replaced by the button **Calculate**. By pressing this button you can recalculate the map, with the expression given above in the dialog box.



- Click the button **Calculate**.

A so-called *tranquilizer* is displayed, showing the progress of the calculation of the map. When it stops, the data file of the raster map **Landuse** is available again.



- Open the **Properties** dialog box of the map **Landuse** again.

Now the dialog box looks like it was in the beginning. Note the object size: 594578 bytes (which is 593684 bytes of data and 784 bytes of the definition).

It is also possible to convert a map (or table, or column) from being a dependent object to a source object. In that case you will break the dependency link. After that the map can not be updated anymore from the property dialog box. You actually need to do the operation again.



- Click the button **Break Dependency Link**. Confirm with **Yes** in the next dialog box. The **Properties** dialog box is closed.
- Open the **Properties** dialog box again and note the difference.

Now the Raster Map Properties dialog box does not contain anymore information on the dependency links. The buttons that were there before are now replaced by the buttons Edit properties and Used By.

More examples of the use of the dependency links between maps and tables will be shown in chapter 4 (spatial data management) and chapter 9 (map calculation).

Summary: Properties and Dependencies

- ILWIS has two basic groups of objects: *data objects* (maps, tables and columns), and *service objects* (coordinate systems, georeferences, domains, and representations).
- The relation between a data object and service objects, or between service objects among themselves, is specified by the *properties* of the object.
- The properties are different for vector maps, raster maps or tables. The properties depend largely on the type of domain which is used.
- The link between data objects, derived from other data objects by ILWIS operations, is defined by *dependency links*.
- Maps, tables and columns, which are original are called *source objects*, and those that are made from them, are called *dependent objects*.
- The concept of dependency has a number of important advantages:
 - Easy update*: when you have new information and you edited input maps, tables or columns accordingly, the dependent output maps, tables or columns can be recalculated as the system can execute their definition again.
 - Easy change of calculation formulas*: MapCalculate and ColumnCalculate formulas, that are stored as the definition of dependent raster maps and columns, can be modified in the Properties dialog box of such a map or column.
 - Minimum use of disk space*: when you perform a calculation or an operation and the outcome is a dependent map or table, in principle only the object definition file of the dependent map (.MPR, .MPA, .MPS, .MPP) or table (.TBT), is stored on disk.

Object definition files store, amongst others, the expression by which the output object is created and contain a reference to the data file. Data files can be very large and are automatically calculated and stored on disk when the map or table is displayed. You can continue to perform calculations and/or operations on the dependent objects that do not have calculated data files yet.

2.6 How to use geographic data in ILWIS

After we have looked quite extensively at the way geographical data is stored in a Geographic Information System like ILWIS, it is important to ask ourselves now what we will do with it. In general there are four different phases that we have to go through when we work with geographic data:

Data input

Using ILWIS, you can input the spatial and non-spatial data in your computer. How you do this depends largely on the data type. For instance, you can digitize maps and can type tabular data. ILWIS can convert data files derived from scanning, or you can import other digital formats as inputs. Through image processing, ILWIS enables data input from remotely sensed images. The aspects of spatial data input will be dealt with in the next chapter (chapter 3).

Data management

Data can be stored in and retrieved from ILWIS. The data management in ILWIS emphasizes efficiency of storage and retrieval. File formats used are inter-convertible so different data types can be combined. Many aspects of data management in ILWIS will be treated in chapter 4 (for spatial data) and in chapter 5 (for attribute data).

Data analysis

Data analysis is the most important feature of ILWIS, qualifying it as an analytical GIS package. Data analysis is the process in which you generate information that is useful for solving a particular problem. This data can be in the form of satellite images. The data analysis on satellite images is also called digital image processing. The objective is to generate images that can be interpreted by experts, or to generate classification results, containing thematic information on features such as land use, lithology etc. The various ILWIS operations on image processing will be covered in chapter 6.

The data analysis techniques dealing with maps, combined with tables, will be treated in the chapters 7 to 12, in which there is an increasing degree of complexity.

Data output

The final objective of working with a GIS such as ILWIS is to produce useful output results. Using ILWIS you can show your results on the screen, as hard copy output (printed data) or as soft copy (files to be used further in ILWIS or in other software).

All aspects dealing with the generation of output with ILWIS will be treated in the final chapter 13.