

# Homologue™

**A Computer System for Identifying  
Similar Environments  
throughout the Tropical World**

**Version Beta a.0  
2005**

**P. G. Jones, W. Diaz, and J. H. Cock  
Edited by Annie L. Jones**

**CIAT**

The International Center for Tropical Agriculture (CIAT) is a not-for-profit organization that conducts socially and environmentally progressive research aimed at reducing hunger and poverty and preserving natural resources in developing countries. CIAT is one of the 15 centers funded mainly by the 58 countries, private foundations, and international organizations that make up the Consultative Group on International Agricultural Research (CGIAR).

CIAT also receives funds for research and development services provided under contract to a growing number of institutional clients.

Information and conclusions reported in this document do not necessarily reflect the position of any donor agency.

MLM  
QK  
46  
.5  
.E4  
J66  
C-2

ISBN 958-694-077-2

# Homologue<sup>TM</sup>

**A Computer System for Identifying  
Similar Environments  
throughout the Tropical World**

**Version Beta a.0  
2005**

**P. G. Jones, W. Díaz, and J. H. Cock**

**Edited by Annie L. Jones**

**CD-ROM Series**



Manual Copyright © 2005  
Software Copyright © 2005

Homologue is written using MapObjects LT and Delphi. MapObjects LT is a trademark of the Environmental Systems Research Institute, Inc. Copyright 1999. Environmental Systems Research Institute, Inc. All Rights Reserved. Delphi is a trademark of Borland International, Inc. Copyright 1983, 1999 Inprise Corporation.

Centro Internacional de Agricultura Tropical  
*International Center for Tropical Agriculture*  
Apartado Aéreo 6713  
Cali, Colombia

Fax: +57 (2) 4450000  
E-mail: p.jones@cgiar.org

CIAT Publication No. 342  
ISBN 958-694-077-2  
Press run: 500  
Printed in Colombia  
July 2005

Jones, Peter G.

Homologue: A computer system for identifying similar environments throughout the tropical world, version Beta a.0 / P.G. Jones, W. Diaz, and J.H. Cock; edited by Annie L. Jones. -- Cali, CO : Centro Internacional de Agricultura Tropical (CIAT), 2005.

1 CD + Manual: Homologue: A computer system for identifying similar environments throughout the tropical world, version Beta a.0 (100 p.) -- (CD-ROM series. CIAT publication no. 342)  
ISBN 958-694-077-2

AGROVOC descriptors in English:

1. Computer software.
2. Computer applications.
3. Simulation models.
4. Site factors.
5. Meteorological elements.
6. Climate.
7. Soil types.
8. Environmental factors.
9. Databases.
10. Forecasting.
11. Tropical zones.
12. Colombia.

AGROVOC descriptors in Spanish:

1. Programas de computador.
2. Aplicaciones del computador.
3. Modelos de simulación.
4. Características del sitio.
5. Condiciones atmosféricas.
6. Clima.
7. Tipos de suelos.
8. Factores ambientales.
9. Bases de datos.
10. Técnicas de predicción.
11. Zona tropical.
12. Colombia.

I. Diaz, William. II. Cock, James H. III. Jones, Annie L. IV. Centro Internacional de Agricultura Tropical. V. Tit. VI. Ser.

AGRIS subject category: U10 Mathematical and statistical methods /  
Métodos matemáticos y estadísticos

LC classification: QK 46 .5 .E4 J66

Copyright CIAT 2005. All rights reserved.

CIAT encourages wide dissemination of its printed and electronic publications for maximum public benefit. Thus, in most cases colleagues working in research and development should feel free to use CIAT materials for noncommercial purposes. However, the Center prohibits modification of these materials, and we expect to receive due credit. Though CIAT prepares its publications with considerable care, the Center does not guarantee their accuracy and completeness.

# Contents

	Page
Introduction	1
1. Getting Started	3
2. Tutorial	5
Arming the Map	5
Finding Your Way Around	8
Five Colombian Coffee <i>Fincas</i>	9
The Probability Analysis	14
The Soil Specifications Window	17
3. User's Guide and Reference Section	28
The Map Window	28
The Main Menu Service Icons	30
4. Theory	45
The Climate Surfaces	45
Climate Date Standardization (Rotation)	46
The FloraMap Probability Algorithm	49
Generalizing from FloraMap to Homologue	54
The Soil Probability Model	57
Combination of Soil and Climate Probabilities	61
The Cloud, or Combining Probability Maps Drawn from Various Calibration Points	61
5. Program Software Structure	63
Overview	63
Data Structures	63
Software Structures and Operation	69
References	91
Index	93

# Introduction

In agriculture one of the foremost questions is “What can be grown in a specific site?” Answering this question provides the initial basis for analyzing and then defining the best possible options in agriculture. Farmers, agri-business, food processors, supermarkets, development agencies, local planning organizations, public agricultural organizations, and credit organizations ask this question on repeated occasions. Information is available for the major crops, but for the myriad crops of minor importance on a global scale, although of great importance to rural communities and local economies, reliable information on what could be successfully grown is not readily available. Homologue™ was developed to help people determine “What crops or varieties are likely to perform well where”. The basic, extremely simple, premise behind Homologue is that crops will perform in a similar fashion in similar environments wherever they happen to be in space or time.

In the early 1980s, the focal points of targeting agricultural technology were the mega environments and the recommendation domain, which emphasized similar environments on a broad scale. Technology was geared to being broadly adapted within any one mega environment: tools were not available to tailor particular technologies to specific environmental niches. David Wood, then at CIAT, asked where it would be best to plant a collection of *Gliricidia* germplasm to conserve it for future use. This led one of us, Peter Jones, to develop a series of algorithms for comparing climates of specific sites. Later, Nick Galwey and Peter Jones used the central algorithm to develop FloraMap®, which maps the distribution of plants and other organisms in the wild.

FloraMap assumes we know nothing about the organism other than the geographic location of a set of collection points. From these it fits a climate probability model. This is being used widely; however, it has drawbacks for many applications. It requires a set of at least 20 points for each species, only works on climate, and has not been successfully applied to cultivated crops where the farmer alters the environment.

So, what do we do for those who ask the simple question, “Where else in the world is like my plot of land?” If we answer this question, we can infer, from the agricultural practices there, what may be applicable to my plot.

We specified from the beginning that Homologue would have to work for any species without having a specific calibration set. Although the question may be simple, the answer is not. Homologue has been developed to cope with the complexities of this simple question. For climate, the basic algorithm of FloraMap was generalized to fit a range of generic species designated by the user, whilst for soils we developed and incorporated statistical probability calculations to map characteristics. Homologue also can be used in the reverse sense. If we find a single instance of the cultivation of an interesting species or variety, we may look at where else it might be used. If we have a number of these instances, we can identify an environmental envelope, or “Cloud”, where the species fit. Homologue therefore circumvents the problem that FloraMap has with small sample sizes.

This version of Homologue is the Beta demonstration release. It has been crafted to give an idea of the operational capacity of the full release version that will work with the 30 arc-second climate grids. It loads as many data as possible into RAM in order to speed operation. It is designed as a “What if?” tool that scientists can use in an exploratory way. Sit with it, and try out ideas to see if they fit with the world and your conception of it. It should be fast enough to play with. Please do not expect Homologue software to be easy to manage intuitively. When the Wright brothers invented the airplane, they found that there were various rules that you had to learn in order to fly. Pilots nowadays respect those rules and many more. You may use this software in any way you like, but to use it effectively read the manual and follow the tutorial. Learn to fly it, play by the rules: it should not crash and should produce the required results.

Based on user feed back from FloraMap, we suggest that you have a quick look at the User’s Guide – Reference Section to get an idea of the look of the software, and then plow straight into the Tutorial. Learn the software in a hands-on exercise and use the reference section to look up things you are not sure of later. We hope that you like Homologue, and give us positive feedback so that we can improve it and find the funds for the next version, which will be even more miraculous.

# 1. Getting Started

Homologue is a Windows® application that will be installed from the CD-ROM and registered automatically. The program files normally will be installed in the directory C:\Program Files\CIAT\ and, unless you have a good reason for installing in another directory, we strongly recommend that you let the install package go ahead and do so.

- √ Insert the CD-ROM in your CD drive.
- √ Go to the run prompt and type X\setup where X is the drive letter of your CD drive.

The Homologue system comes with large data files. The first window of the install procedure shows an analysis of the disk space available on your system, and subsequent windows will allow you to tailor the installation and to make the best use of this space.

- √ Note where best to install Homologue.
- √ Hit Yes to proceed.
- √ Read the notes on the following screens.
- √ Then choose the relevant installation type.

The largest set of data files is the map coverages. These are Environmental Systems Research Institute (ESRI) shapefiles that are used to create the backgrounds for the maps you will use with Homologue. The directory is called \coverages\ and is 582 Mb. You can elect to leave it on the CD-ROM if you are short of disk space. In this case, choose the option "Typical" when the Install shield requests it. Leaving it on the CD-ROM will not slow Homologue operations to any great extent, but it does mean that you have to have the CD-ROM in the drive whenever you work. If you choose to install it on the hard disk, the install shield will attempt to put it in directory \HomologueFiles\ on a disk with sufficient space. You may override this and choose another site for it if you wish.

The climate grid files and all the model parameters are stored in the directory \application\_data\; this is currently 56 Mb. It must be

installed on a disk and will be flagged read only. We suggest that you install it, if possible, away from the program files on your C disk. The install shield will attempt to put it in the directory \HomologueFiles\ as above.

The last choice you have is where to put the working directories \dat\ and \output\. These will contain your input and output files. It is also best to keep these away from the program files directory. Homologue output can be voluminous, so make sure that wherever you decide to put them there is sufficient disk space.

√ Hit Finish to start the installation.

## 2. Tutorial

The purpose of this chapter is to lead you by the hand through a typical Homologue example. Most people do not read computer software manuals, so we have put the users' guide in the following chapter. There should be enough information in this chapter to get you through the analysis without having to look anything up in Chapter 2. We know you will only do that in desperation.

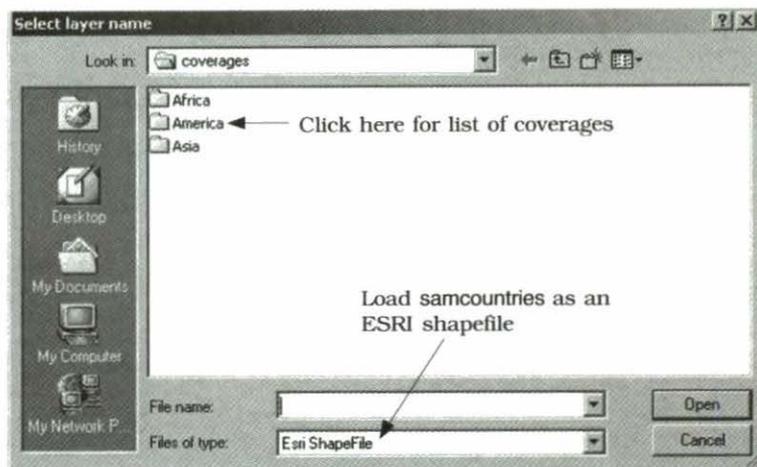
Here goes then. We are going to look at some coffee farms in Colombia. We have five to look at, so let's get Homologue fired up and the map of Colombia ready to find the fields of coffee.

### Arming the Map

Get the layer control tool with this icon.



If the map of Latin America is not showing we need to load it. The load layer icon on the layer control tool will do that. Look for America in the coverages directory.



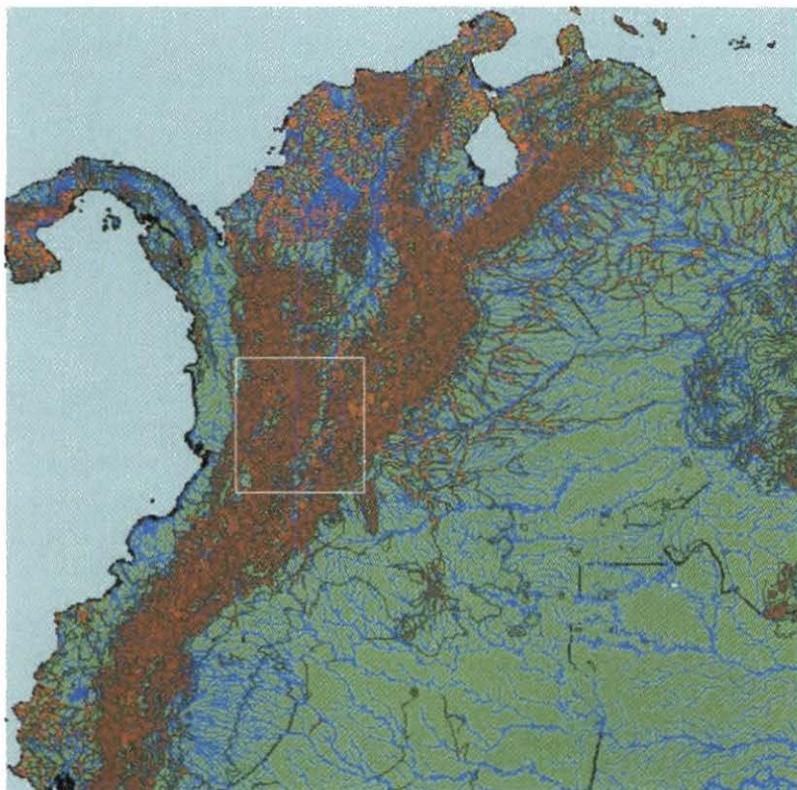
Check that the coverage loads with the right color. A light green is usually acceptable. If you need to change it, use the coverage color icon, click onto one you like, and click OK.



Use the zoom tool to clip the continent to the country boundaries of Colombia.



Now load *samrivers* using the load layer icon. Change the color to darkish blue as the color chart comes up as it loads. Do the same with *samroads*, changing the color to red, and *samcontours*, changing the color to brown. The map should now look like this. It is not very pretty at this scale, but we are just about to home in on the coffee regions.

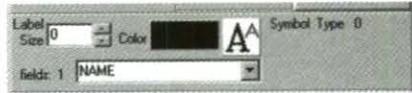


Look at the lower right corner of the map window. Select the

lat:5.3716 lon:-76.825

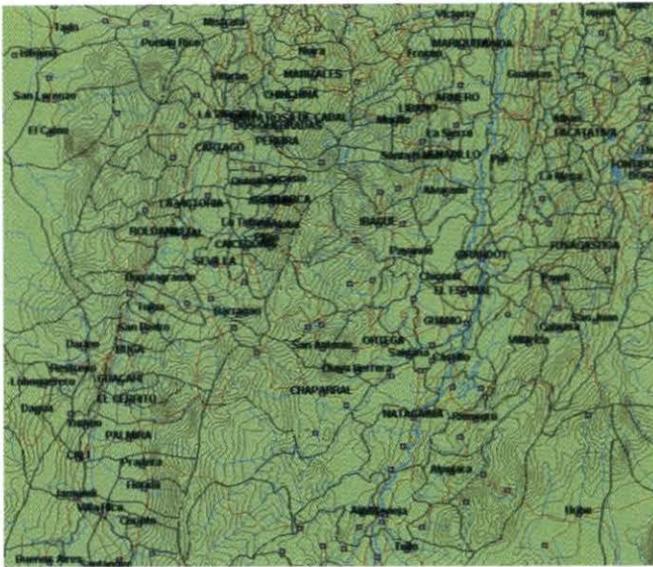
zoom tool. Set the cursor on these coordinates and draw down the zoom boundary to form a square roughly like the one drawn above. This will cover quite a number of significant Colombian coffee areas and we will be at a scale where we can go to the next step.

Load the coverage samtowns, color the symbols how you wish and set the names on. To do this, use the set layer properties icon on the layer control window. In the center panel, select the field name and set the label size to 0. You can change the color if you wish, but we prefer black.



The last thing to do is load sammunicip. Color it black. Now use the set layer properties to change to transparent fill.

The map will now look like this. You have it ready for work.



You can move up and down the Andes with the pan icon, and manipulate the scale with zoom out a bit and zoom in a bit.



**NOTE:** If you accidentally hit the zoom out to full extent key the map will become completely unintelligible because these coverages do not thin as they scale, and will completely cover each other. In this case, use the zoom tool to home back in on the window you had before.

## Finding Your Way Around

Ideally, everyone should be born with a Global Positioning System (GPS) receiver between the ears. As it is, only old sailors even come close to this. You and I have to rely on a GPS in the pocket, or on reading a map. If you are lucky, and whoever took the coordinates of the field you are trying to find had a GPS, then you will have good coordinates. If not, then you are relying on somebody reading a map, guessing, or merely describing where the field is. In many cases, all three options are roughly equivalent, so even if you have got coordinates someone has read off a map, it is a good idea to try to get a description as well.

*You have reliable coordinates:*

Use the location window at the lower left-hand corner of the map screen. The cursor coordinates are shown to an accuracy of about 10 m. That is a lot better than most hand-held GPS receivers. If you pan the cursor to the middle of the screen as you zoom in a bit, you can very quickly locate a known point.

lat:5.3716 lon:-76.825



*You have coordinates, but you would like a check on them against a description:*

Locate the point as above. Note the distance from a recognizable landmark, road, crossroads, river, confluence, or town. Select the zoom tool and click once on the point that you have found. Move the cursor to your recognized points, and note the distance at the lower left of the map window.

Dist:1.3554 Km



***Do not hold down the mouse key or you will draw out a zoom box.***

*You only have a description of the place:*

This actually can be almost as accurate as a GPS measure if done correctly. "1.4 km south east of the road junction south of Palmira" will get you to within about 100 m. Not bad for guesswork! Select the zoom tool, place the cursor on the road junction, click once, move the cursor southeast until the distance measures 1.4 km; you are there!



***The coverages in this Beta release are not as good as they could be. We are working on this for subsequent releases. If you have access to better shapefiles, use them.***

## Five Colombian Coffee Fincas

We are going to look at five different coffee-producing farms. These are real places producing first class Colombian Arabica coffee. Actually, all the fields were identified by high accuracy GPS, but for the purposes of this exercise we have changed some of the types of location descriptor.

Farm: LAS ACACIAS	Proprietor: NELSON MELO
Municipality: POPAYAN	Department: CAUCA
Latitude NORTH: 2° 27' 30.68"	2.458523538
Longitude WEST: 76° 32' 49.32"	-76.547032746
Elevation: 1913 m	Soils: TIMBIO series

### Maximum and minimum values of selected characteristics, TIMBIO soil series.

Characteristic	Minimum	Maximum
Cation exchange capacity (meq/100 g)	26	34
pH	5.0	5.6
Organic matter (%)	15.6	17.1

We have a really good reference here. It was taken with a Wilde GPS, and is highly accurate. All we have to do is move the cursor to the relevant point.

- √ Choose the pan tool and push the map upward and to the right until you reach the vicinity of Popayán. 
- √ Zoom in a bit until you can clearly see the black outline of the Popayán municipality. Use the pan tool to keep centered as you zoom in. 
- √ Check the coordinates in the right-hand lower corner of the map, and move the cursor to the correct coordinates. You may zoom in a bit to increase the accuracy of your placement.

- v Hit the site specification tool and check that you have the right place. If your coordinates are a bit off you can trim then numerically with the arrows or by entering the correct number. If you move the coordinates on the site specification screen,

use the site specification icon next to the elevation window to pick up the new elevation. If it does not change, then you are still on the same climate database pixel.

Site Specifications	
Latitude	
Deg/Min/Sec	14 40 4
Decimal	14.667970
Longitude	
Deg/Min/Sec	06 32 49
Decimal	86.547065
Elevation	1158

Let's leave it there and go on to the next problem.

Farm: SAN RAFAEL

Proprietor: GERMAN ESTRADA

Municipality: CALDONO

Department: CAUCA

Longitude NORTH: 2° 44' 53"

Latitude WEST: 76° 34' 25"

Elevation: 1644 m

Soils: PIENDAMÓ series

About 4.5 km east of Morales, 4 km from the Piendamó-Morales road.

**Maximum and minimum values of selected characteristics, PIENDAMÓ soil series.**

Characteristic	Minimum	Maximum
Cation exchange capacity (meq/100 g)	22	49
pH	4.6	5.2
Organic matter (%)	12.6	27.9

The picture on the next page shows a typical vista of the area.



Coffee area around Caldono, Cauca.

Now, the point for this field was read off a map. The coordinates are in degrees, minutes, and seconds. That is fortunate because many maps are projected to an alternative grid, often measured in meters from a baseline. Homologue cannot cope with these types of coordinates. A course in map projections can be found at: <http://classweb.gmu.edu/shirsch/MapProjections/>. If you need transform software many different packages are offered, but a good cheap one is available at: <http://franson.biz/coordtrans/>.

First, calculate the decimal degree coordinates. For latitude, this is  $2+44/60+53/3600 = 2.748$  positive, because it is North. For Longitude it is  $-76-34/60-26/3600 = -76.574$  negative, because it is West.

Caldono is just a bit north from the last farm we found so pan the image down a bit, checking against the coordinates window. Find the coordinate point with the cursor. Zoom in more if necessary.

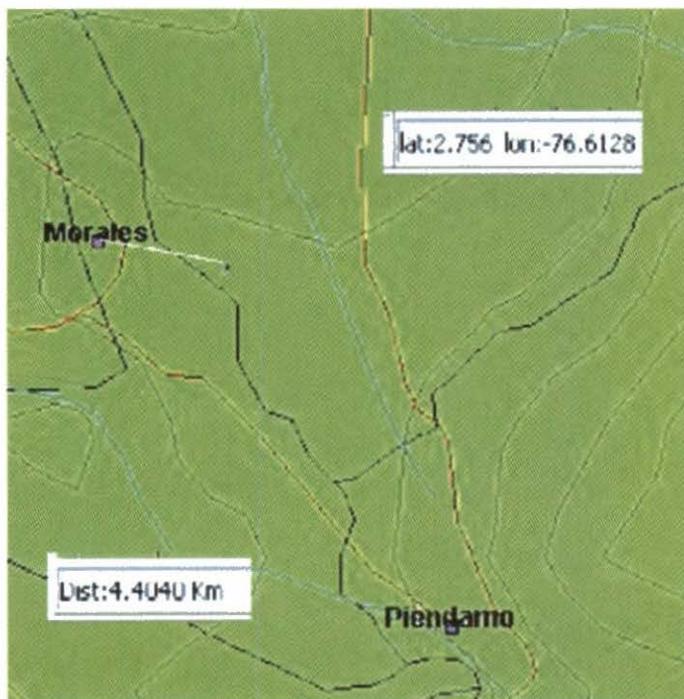
Select the layer control tool and highlight sammunicip by clicking on it.

Now select the information tool. **I**

Click on the point you have just found, and verify that the municipality is Caldono.

Select the zoom tool. Click on the point again and move the cursor to the town of Morales. Note the distance—does it match the description? Move the cursor to the closest point on the Morales-Piendamó road. Note the distance again.

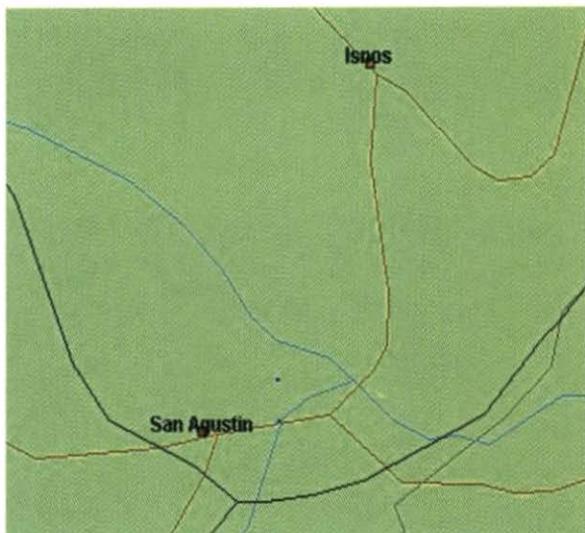
I think that if you have found the point on the right, you are there and we can go on to the next exercise. The white line will not show on your screen; it is indicating that that is the distance being measured, the distance and coordinates indicators have been moved onto the graphic to make it more compact.



Farm: Unnamed 1  
Municipality: ISNOS  
Elevation: 1696 m

Proprietor: ARGEMIRO PALADINES  
Department: HUILA

Two km along road to Isnos out of San Agustín, just before crossing the river bridge, north 1 km.



The department of Huila is south of your present position and San Agustín is about 100 km south of Morales and slightly to the east. Zoom out a bit to widen your view and pan down until you see San Agustín.

Zoom in to a window that just includes San Agustín and Isnos. Follow the road from San Agustín to Isnos until it crosses the first river. With the zoom tool selected, click on the river bridge. Now, with the mouse key free, measure 1 km due north from the bridge.

With the layer control tool highlight *sammunicip* layer and engage the information tool. Check the point you have found is in the municipality of Isnos.

You have followed the directions, now read off the latitude and longitude. You should be within 100 m of lat. 1.887 long. -76.234.

We will leave you to find the next two.

Farm: Unnamed 2

Proprietor: ORLANDO FRANCO

Municipality: YOTOCO

Department: VALLE DEL CAUCA

Latitude NORTH: 3° 56' 56"

Longitude WEST: 76° 25' 30"

Elevation: 1635 m

About 1.5 km from the road running NE from Darién.

Farm: LA AMERICA  
 Municipality: LA UNION  
 Latitude NORTH: 1.629  
 Longitude WEST: 77.119  
 Elevation: 1540 m

Proprietor: MARTIN ARLEY TULCAN  
 Department: NARIÑO

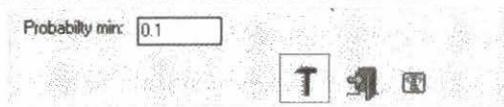


Coffee farm in La Unión, Nariño.

## The Probability Analysis

Now that you have had plenty of practice at finding the target fields we can go to the next stage of the analysis. You have already used the site specification tool so let's go back and find the farm of Sr. Nelson Melo. Use the site specification tool to define the target points and target climate. Coffee has a moderate adaptation range so we will set that on the slider. Note that the species type changes to 2. Now let's look at the probability distribution with only climate as the deciding factor. Choose an output filename; Melo01 would be appropriate to keep track of what we are analyzing.

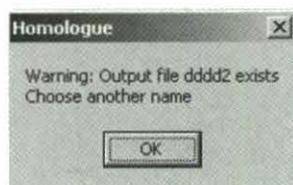
The last thing you need to do on this window is to set the minimum probability level that you want output to the probability coverage file.



Even though we are not going to set any soil specifications, we still need to go through the soil specifications window. You will find that with this button on the site specification window.

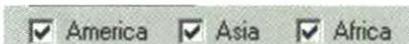


That is, as long as you have specified a unique output filename. If you are like us, you will not have done so, and you will see this come up. Choose a filename that does not exist and hit the soil specifications icon again.



Go through the window without setting any soil characteristics. Using the clear soil specifications button, at the left, makes sure that none are set.

Set for which continents you want it mapped. In this case, we have chosen all three continents. You could limit it to Latin America if you wished to save a little time; it would only be a few seconds but at least it would only map over the continent.

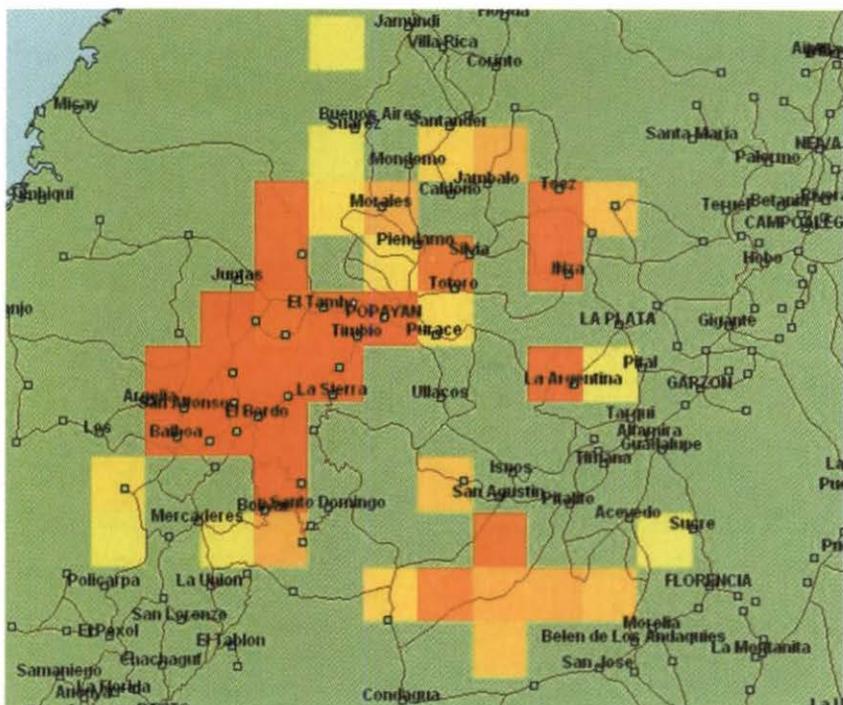


There is a man running away from the geological hammer that William Díaz chose to represent soil science. He is called construct the map. He will do the job for you. Follow the running man and you will get a probability map for Sr. Melo's fields.



That is, you will get a complete mess on the screen! Homologue always expands the image to the full extent when plotting the probability layer; this is to show the broad extent of the homologue you have just specified. Do not despair of the mess. We are eventually going to fix this, but at present all those layers that are so helpful at large scale when you are looking for a site are a real pain when you try to display the world. Do not erase them though. It will take longer to reload them should you need them. Highlight each one in turn in the layer control tool and make them invisible with the eye icon. You can check on the title bar of the window to see if a layer is visible. This may sound silly, but you may not be seeing it because it is hidden behind another layer, not that it is invisible. When you have cleaned the picture and zoomed back in to the area we are working on, this is what you get. The climate alone has marked out quite specifically some areas in the south of the río cauca valley.





Next, we need to define the soil characteristics we want. Get back to the analysis you have just done without having to find the point again using the recent analyses icon.



Homologue resident parameters specification

DATE: 20041112 TIME: 140120

DATA\_PATH: F:\Homolog\application\_data\

OUTPUT\_PATH: F:\Homolog\test\_output\

OUTPUT\_FILENAME: pop2

LATITUDE	LONGITUDE	ELEVATION	PHASE
2.45	-76.55	21.33	5.89

ROTATED	MINIMUM	VARIANCE	LAST ACTION	RETURN CODE
0	0.1	2	3	0

CLIMA DATA FILE: C:\20041112140120.txt

PROBABILITY DATA FILE: P:\20041112140120.txt

Preview

preview

SELECT

Use the arrow keys to find the analysis you want. Do not worry if you have erased the results from this analysis, Homologue remembers even if you have. If you do want to erase even the memory, you can do so with the X key to the right of the arrows. Look for the time and date you made the analysis as well as the output filename. This is because Homologue will remember ALL the runs you make even if you erase the outputs and create new files with the same names. Homologue's memory is in a special dbf file that references the climate data file and probability data file. You will not see these files in the user file space. If you want to know more about them, see the user reference section. Hit select to bring up the site specification window and soil specifications window from the previous analysis.

The soil specifications tool is the next to employ. It will open a complicated-looking window, which you can see on the next page, but before we look at how to use it, let's go back and think about what we know of coffee soils from these example farms. There were three soils characteristics specified with maxima and minima from two soil series. Note that this is just a little sample of the data you might have for a real analysis, but it gives us some idea of the sort of soils that we are looking for. We are going to look at both series to get an idea of the ranges of the soil characteristics we need.

The cation exchange capacity (CEC) ranges between 22 and 49 meq per 100 gram, pH goes from 4.6 to 5.6, and percent organic matter goes from 12.6 to 27.9.

## **The Soil Specifications Window**

Here we have the core feature of Homologue. We can specify soil characteristics to combine with the climate probability model. The window is not as horrifying as it looks because you can choose what you want from it. We will go through a typical exercise.

The slider is the main feature that you will be using. You could use the alternative limit control at the bottom if you preferred by entering the numeric values and using the arrows to throw them up into the soil definition area.

Sel	name	Min	Max	Sel Min	Sel Max	Type	Threshold
F	Depth	15.62	615.5			P	
F	Carbon (natural log)	-4.194	4.71			P	
F	Nitrogen(sqrt)	0.0653	1.172			P	
F	pH Water	3.594	9.134			P	
F	pH KCl	3.119	9.525			P	
F	pH CaCl2	1.623	8.932			P	
F	CEC(sqrt)	0.822	11.61			P	
F	Sand	0	100			P	
F	Silt	0	100			P	
F	Clay	0	100			P	
F	Rooting depth	10.66	263			P	

Let's start with CEC. Homologue uses CEC in a square root transform. The limits we found in the soil series from the two farms were 22 and 49 meq/100 per gram. These transform to 4.7 and 7 as square roots. Now, what actual limits do we want to set in the analysis? Click on soil characteristic select and turn it from false to true (F to T). We need coffee soils with at least the minimum CEC that we found in the data, but are we worried if it is better? Probably not, so set the slider to 4.7 at the low end and take it to the limit at the top end.

Sel	name	Min	Max	Sel Min	Sel Max
T	CEC(sqrt)	0.822	11.61	4.644	11.61

It will not take exactly 4.7, but we would be happy with 4.644. If you wanted to be really pedantic, you could go to the alternative limit control and force it to 4.7, but who really cares? We like the slider—it is an unusual one to have two limit controls. William Díaz had to program it as a special feature.

The next is pH 4.6 to 5.6. Well, we could infer that Colombian coffee is very choosy about the pH range. In fact, Dr. Thomas Oberthür is researching this point right now. From experience with building Homologue, we should say that these limits are very close. It may well be that this type of coffee is that specific, but it is best to try a little wider to start with and come down as you trim the map. So let's set the slider at more conservative limits.

Sel	name	Min	Max	Sel. Min	Sel. Max
T	pH Water	3.594	9.134	4.01896	6.75916

The whole concept of Homologue is that a scientist can sit at the console and play with the figures. Seeing the results will form hypotheses. Homologue cannot prove them. *You* can when *you* go out into the field with the new insights that Homologue has provided.

Let's now look at organic matter percentage. These figures are VERY high for tropical soils. This could be the defining feature of good coffee soils, or it could mean that they have been well fertilized and mulched in their top horizon. In fact, the values loaded into the Homologue database are for the full plow layer. This is a problem; we will have to be circumspect in the way we use the soil characteristics. Again, we will need to set the limits wider. Remember that you can always trim the map down, but if it is too narrow to start with there will be nothing there from which to infer.

Homologue analyzes soil carbon as a natural logarithm transform. This is because most soils have little carbon, whereas some few have a lot. The transform normalizes the distribution. Our bet is that the carbon content in the plow layer in these soils is actually no more than about 2%, but we could be wrong. Let's say 2.2 and we could put an upper limit of 20% because above that in the plow layer we are talking about histosols, and these are

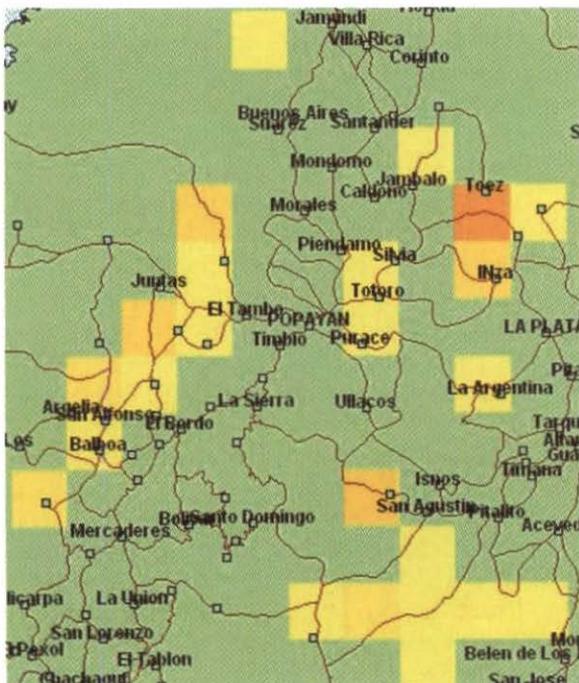
definitely not coffee soils. Take the natural logs of these percentages, 2.2 goes to 0.8, 20 goes to 3.0, 2.87 is near enough, that is almost half pure peat.

Sel	name	Min	Max	Sel. Min	Sel. Max
T	Carbon (natural log)	-4.194	4.71	0.8007	2.8731

Hit produce the map again, and clear it of rubbish by making it invisible. We have picked up some areas in and around Colombia, southern Brazil, and some in Africa. That sounds reasonable for coffee areas. Zoom in to the original Colombian area. This map now shows the local Colombian coffee areas similar to the farm, Las Acacias. Unfortunately, it does not show the farm of Sr. Melo in the probable areas. This is important! It showed in the climate analysis, but not in this one. The fault is therefore in the soil specifications. This can mean three possible questions.

1. Have we got the original soils specifications wrong?
2. Is the FAO soils map of the world wrong?
3. Is there something wrong with the way we are calculating the probabilities?

The answer to the first question is Yes! When we look at a few locations we always underestimate the full range of a phenomenon.



To the second question, the answer is also Yes! As Wim Sombroek (one of the authors of the map) put it so succinctly in the

Rome 1986 meeting (see Bunting, 1987), it is wrong in many places. There is a lot to be desired of it and to be corrected. However, it is the best global soils map that we have. Future versions of Homologue will also have more accurate local soils evaluations available.

The answer to the third possibility is more complex. The way we have applied the soil constraints is multiplicative, but takes into account the correlations between the soil characteristics. Thus, if you had specified pH in water AND pH in KCl, you would have got a reasonably similar answer because the two are very highly correlated. However, when you compound soil characteristics there is always a dilution of the probabilities because you are all the time asking for something more specific. There is another way of looking at it. But first, go back and play with the soils characteristic boundaries you have specified. Get a feel of how they change the look of the map, not just in Colombia but worldwide.

Now that you have investigated the effects of the soil characteristics individually and multiplied, let's look at another way of putting them together. The standard method combines the probability of finding those characteristics with any given soil, and then the probability of finding that soil in the pixel in question. This is a pretty tough test, but it is the one you would want to use if you wanted to calculate how many hectares are likely to fulfill the conditions you have specified (always assuming that the FAO map is right; Wim! bear with me). A pixel at the equator is 32,000 ha (multiply this by the cosine of latitude as you go away from the equator), multiply this by the probability you are getting from the analysis and you have got an upper bound for the hectares of possible terrain. The probabilities you will map in this mode are likely to be fairly low because the probabilities of finding soils with the characteristics you want are going to be fairly low. Finding 1500 ha of coffee soils in a pixel is pretty good, but the probability will only be 0.047.

The threshold method tests the probability of each characteristic against a threshold value. If this is exceeded, then the characteristic is deemed present in the area regardless of the extent of its area. You will not be able to calculate the probable hectares this way, but it might produce a map that is more in keeping with illustration purposes.

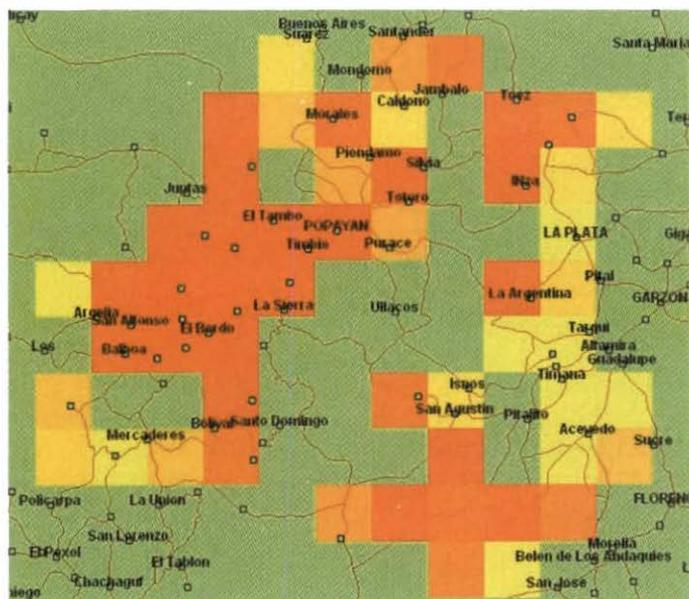
To change to this type of probability, click on threshold type, and change it to H. When you click on the threshold value window the threshold slider will appear, use it to set the threshold value. Change each of the set characteristics to be accepted over a threshold of 0.05. This is reasonable; 5% of an area being acceptable in broken Andean type territory is quite enough for coffee farming. Municipalities in this area average about 50,000 ha; this would give 2500 ha of coffee land, quite enough to flood the market.

		Probability type				Threshold value	
Sel	name	Min	Max	Sel. Min	Sel. Max	Type	Threshold
F	Depth	15.62	615.5			P	
F	Carbon (natural log)	-4.194	4.71			P	
F	Nitrogen(sqrt)	0.0653	1.172			P	
T	pH Water	3.594	9.134	3.6536	5.4804	H	0.05
F	pH KCl						
F	pH CaC						
T	CEC(sta)						
F	Sand						
F	Silt						
F	Clay						
F	Rooting depth	10.65	263			P	

Threshold slider

This time, the map shows clearly where coffee land is present, but gives less of an idea of the actual extent. This version could be better for illustrations where you do not want washed out colors.



Now go back and experiment with the other four farms. We will need the probability files from the five individual farms for the rest of the exercise. They could be just climate probabilities, climate probabilities with multiplicative soils, or with threshold soils. It is up to you to choose the set of criteria with which you are happiest, but make sure that they are all the same criteria so that we can compare the maps as we go on.

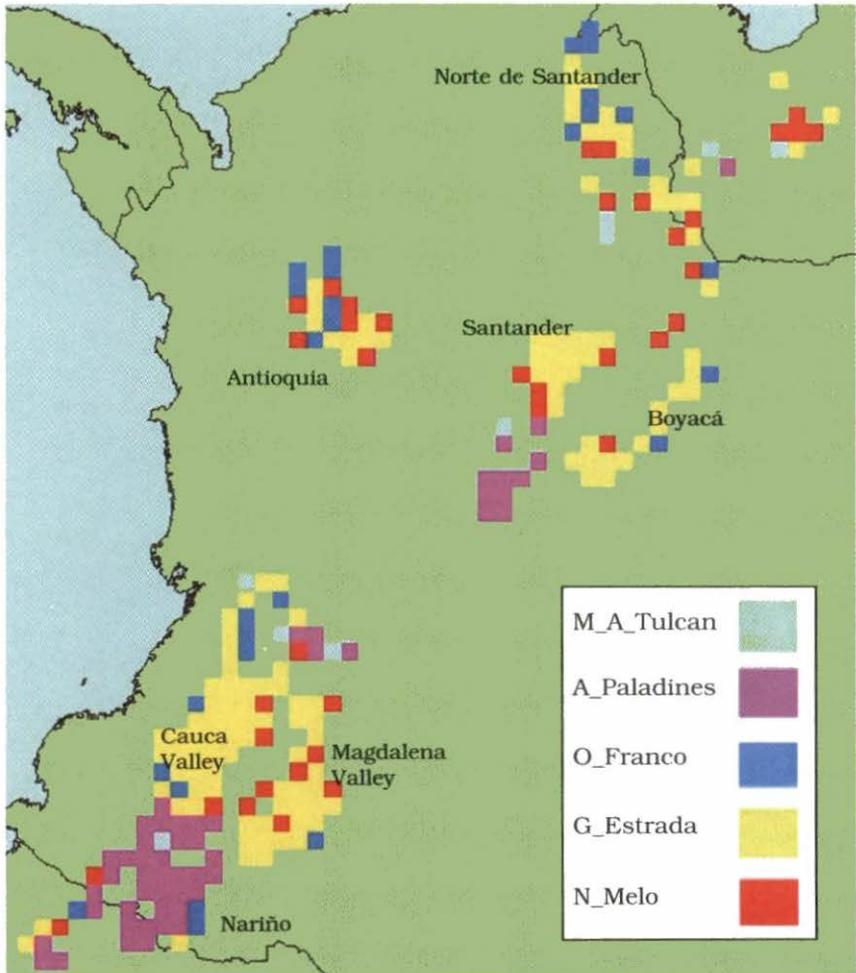
We have created five probability files with the threshold probabilities and called them `N_Melo`, `G_Estrada`, `A_Paladines`, `O_Franco`, and `M_A_Tulcan`. This time, we turned on the settings for Africa and Asia to produce complete tropics-wide maps. We also trimmed the lower limit for plotting the map down a little to 0.08. If you can do the same or similar we can carry on with the analysis and see if we can draw any conclusions about the environment of Colombian coffee.

For each probability map, go to set layer properties and choose a different color for the coverage. You can leave them graduated or put the same color on each side, which will create a uniform color for the shown probabilities. Now start checking their distributions within Colombia. `G_Estrada` and `N_Melo` are quite similar. They are both present in each of the areas delimited, but do cover slightly different ranges. `M_A_Tulcan`, despite coming from quite a distinct area, seems to be adjacent to most of the previous areas, but with little overlap.



*A good way to check this is to put the file to be compared at the top of the stack and switch it from invisible to visible.*

O\_Franco comes from quite far south, but overlays G\_Estrada well, apart from a few details. These four then seem to define six quite geographically distinct but environmentally similar coffee areas. They account for the well-known coffee areas in the south of the Cauca and Magdalena river valleys in Antioquia, Boyacá, Santander, and Norte de Santander.



Adding in A\_Paladines we find a completely different pattern. None of the areas match the previous ones. A large area in the south of the country corresponds to the coffee area of Nariño, with a smaller region just to the southwest of Santander. The Nariño region is known for its unique style of coffee, prized by some connoisseurs. It would appear that it could be environmentally distinct from other Colombian coffee-growing regions.

There is one remaining thing to do in this analysis. We have looked at the Colombian coffee areas within Colombia, but where is the opposition, and with which areas of Colombia does it compete?

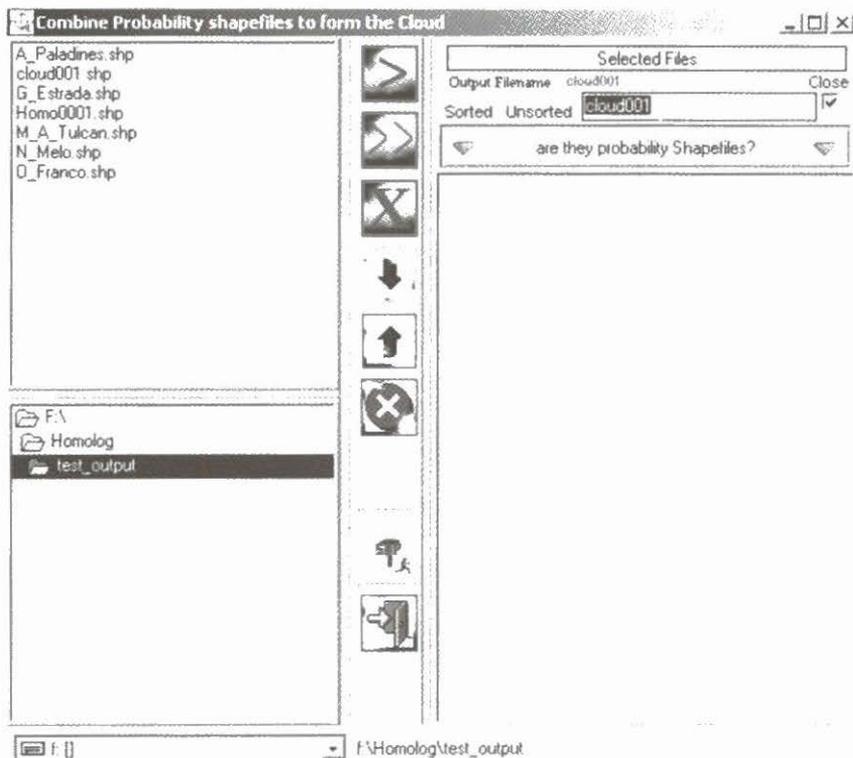
We have decided that there seem to be at least two main coffee environments included in the sample of five from which we started. Of these, the Nariño example of A\_Paladines stands out alone. The simple thing to do then is look for that in the rest of the world. Go to the layer control tool and look for the coverages directory from Africa, load `afrcountries` and from Asia, load `seacountries` as ESRI shapefiles. Turn the other four probability files to invisible and look for homologues of A\_Paladines throughout the rest of the world. There is a small area in southern Brazil, and three tiny areas in Africa, one in the Democratic Republic of Congo, and two in Kenya. These are actually all famous coffee-growing areas, but they should not be too much competition for a branded connoisseur line of Colombian coffee. The other four sample points seem to indicate a broader range of environment. There is one more Homologue function that we can use to look at this, the Cloud.



The Cloud function is designed to cover the case where we have multiple sites defining a highly productive environment. They do not have to be the same in any way. The only reason we have chosen to rule out the Nariño case from this exercise is to illustrate the power of this function of the software.

James Cock, who envisaged the function, drew it from the fact that raspberries were grown in Scotland and also near Bogotá in Colombia. These are very different environments, but, from the point of view of the crop, when managed can be very much the same. Therefore it is useful to be able to put together all instances of where a crop does well in order to view the envelope of its adaptation—the Cloud as James put it. Putting together homologues of many different successful crop situations gives us a cloud of geographic possibilities.

Hit the Cloud key and get the following window. You can choose shapefiles from anywhere on your local net. Use the disc accessor at the bottom left and the access tree just above it to search for shapefiles.



The double arrow facing right will transfer all the files on the left to the operation list on the right-hand side of the window. You may not want all of them, so highlight those that you want and transfer them with the single right-pointed arrow. If you are satisfied with the whole list, then use the double arrow. You can go away to other directories in search of more probability shapefiles and come back and add to the list if you like.

The X key will delete the highlighted file. The X in a circle will delete the lot (so do not do that unless you are sure).

When you have transferred all that you would like in the Cloud, hit the running man and he will do everything else for you.

What you get should look like this! You cannot see very much, can you? But you can, if you zoom in. Colombian coffee from the four sites we have analyzed could have a major competitor in Brazil, some in East Africa, and a little in New Guinea. Well that is not world-shattering news, but it is nice to know that Homologue works! Java, Kenya, and Jamaica Blue Mountain (zoom in and see) are obviously something else.



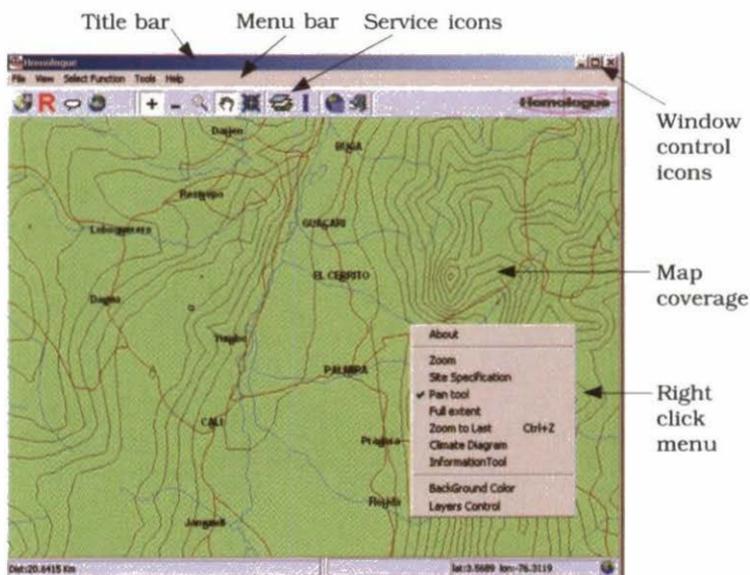
And we know that the Nariño option is almost exclusive. If we had put tighter options on the others, would they be as well? A connoisseur's dream!

### 3. User's Guide and Reference Section

This section describes the operation of the users' interface. It will only refer to the underlying algorithms where necessary to explain what is going on behind the scenes. The theory of the method can be found in Chapter 4, the software definitions and workings of the algorithms in Chapter 5. The basic idea of Homologue is to match a place or places to a larger area to determine the similarity of environments. So the first thing we will want to do is to enter the exact coordinates of a place to be matched. Now you could just type in the geographic coordinates and, as you will see, you can do this with Homologue. However, looking up latitude and longitude or grid reference points on a map is a tiresome business, so Homologue provides you with an on-line mapping facility to find the place you are looking for.

If you have not tried to use Homologue yet, do not spend too much time on this reference section. Get stuck into the Tutorial in Chapter 2 and learn how with a hands-on exercise.

#### The Map Window



The basic window of Homologue is termed the map window. This is the first that appears when you fire up the software. What you get there depends on the configuration settings that you have given in a previous encounter—see the configuration tool (we are not joking).

The window contains the menu bar and the service icons that you will use to do the job.

A right click on the title bar will give you the standard Windows® control—move, size, minimize, maximize, close. You can also control the window with the standard window control icons.

The menu bar consists of pull-down menus that will activate the various services and tools. All of these except help are available directly from the service icons. The about box gives you information about the authors and various copyright considerations for the software used in Homologue development.

The right click menu gives you an alternative route to some of the tools that are found in the main menu service icons.

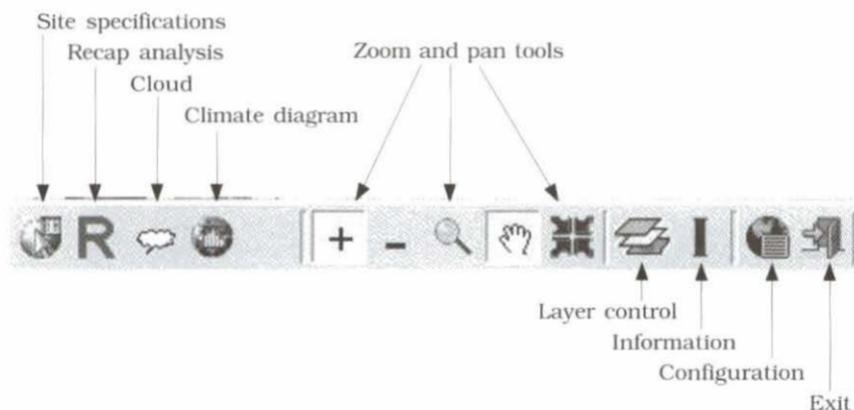
Place the cursor anywhere on the map window and right click. The right click menu will appear.

You will find a range of shapefiles that you can overlay on the map using the layer control tool. You can find these in the directory `\coverages\`. They will supply map features such as roads, rivers, and towns to help you navigate about the map. Beware! They are for use with the map zoomed well in, to present sufficient detail. If you apply them to the map at full extent, they will be so dense that they will practically color the map. The map in the illustration is composed of `samcountry.shp`, `samtowns.shp`, `samrivers.shp`, and `samroads.shp`, and shows the area around CIAT.

The layer control tool allows you to color the map to your liking.

## The Main Menu Service Icons

The figure below shows the main service icons. We will explain them each in turn, moving from left to right.



### Site specifications tool



This tool allows you to select a target site. Point at the map, left click with the mouse and the screen at right will appear.

The selected latitude and longitude appear both as degree/minute/second, and as decimal degrees. If you know the point you want with more precision than you have pointed at the map, you can trim these values to the ones you want. The elevation will be taken from the database for the point that you have indicated.

**Site Specifications**

Latitude  
 Deg/Min/Sec: -3 55 4  
 Decimal: 2.918

Longitude  
 Deg/Min/Sec: -67 8 39  
 Decimal: -67.1442

Elevation: 60

Directory paths and Filenames  
 Data path: I:\Homolog\application\_data\  
 Output Directory: I:\Homolog\test\_output\  
 Output Filename: Homo0001

Species type:  
 1 2 3 4 5  
 3 medium adaptation

Probability min: 0.1



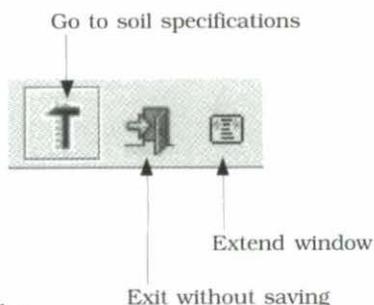
**NOTE:** There is no point adjusting the elevation in this version because we have not built in the capacity to accept the change from the database values. We will be fixing this in full release versions.

The directory paths and filenames are there for your information. You cannot change them on this screen. If you need to change them, go to the configuration tool.

The output filename you can change. It must be unique, and if it is not, Homologue will point this out to you. If you actually want to overwrite a previous output file, you must delete it.

The species type is very important to the analysis and here is where you must choose it. Since the climate model is generic you have the option of choosing the adaptation range of the species type that you want to model. Use the slider or enter a number from 1 to 5. This will be translated into the variance for the climate analysis.

Probability min is the minimum probability that you want shown on the output map. Note that this means that records for pixels with probabilities lower than this are not created for the coverage. They cannot be recovered once the shapefile is created.



The group of control buttons at the bottom right control the exit, either to the soil specification window, or to exit without saving the information. The extend window button expands the window to the right to accommodate long filenames if necessary.

 *The soil specification window has no main service icon of its own because it has no meaning without a site being specified.*

The soil specification window is reached from the site selection window or from the recent analyses tool (see next page). The window is divided horizontally into three main sections. At the top is the slider, at center the soil definition area, and below this is the lower control panel that serves a number of purposes.

The slider is a double-sided slider that can be used to set upper and lower limits to any of the soil characteristics. The scale changes automatically as a characteristic is selected. If you wish to set only upper or lower bounds, then move the other slide pointer to the limit of the scale.

You can choose as many (or none) of the soil characteristics as you like by double clicking on the soil characteristic selector that will turn from F to T. Once selected and highlighted, the slider scale will change to that for the characteristic, and moving the slider pointers will enter the maximum and minimum values for the chosen range.

Soil characteristic select
Soil definition area
Slider

Soil	Min	Max	User Min	User Max	Type	Threshold
Depth	15.62	615.5			P	
Carbon (natural log)	4.194	4.71			P	
Nitrogen (wt)	0.0653	1.172			P	
pH Water	3.534	9.134			P	
pH U	3.115	8.525			P	
pH CaCl2	1.623	8.932			P	
CEC (cm)	0.822	13.63			P	
Sand	0	100			P	
Silt	0	100			P	
Clay	0	100			P	
Flooding depth	10.66	263			P	

Max/Min Value:

Output tolerance:  America  Asia  Africa

Buttons: Run, Erase all, Print the specifications, Exit

Status: Regaining Process

Define continental coverage

If you wish to be more precise than the scale will allow, you can enter the maximum or minimum values into the alternative limit control, and use the red and orange up arrows to transfer these values into the relevant selection positions. If the probability type select value is P, then the probability used in the calculations will be the actual value of the probability integral between the limits that you have set. If, on the other hand, you double click on the probability type select field and change it from P to H, then the probability used in the analysis is 1 if the integral exceeds a threshold value and 0 if it fails. After you have made the change from P to H, click on the set threshold field, and a slider for the threshold value will appear. (See next page).

Sel	name	Min	Max	Sel. Min	Sel. Max	Type	Threshold
F	Depth	15.62	615.5			P	
F	Carbon (natural log)	-4.194	4.71			P	
F	Nitrogen(sqrt)	0.0653	1.172			P	
T	pH Water	3.594	9.134	3.6536	5.4804	H	0.05
F	pH KCl						
F	pH CaCl						
T	CEC(sq)						
F	Sand						
F	Silt						
F	Clay						
F	Rooting depth	10.66	263			P	

Threshold slider toggle

Set the threshold slider toggle to the probability desired, and hit exit. The threshold value will appear in the threshold value field.

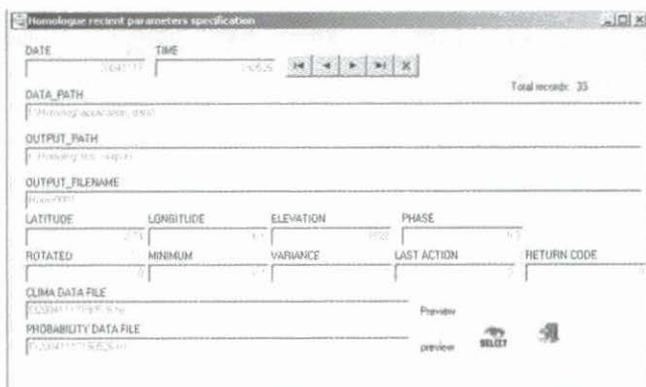
When all is set, use the run button to create the probability map. If you have made mistakes, or want to make sure that no soil characteristics are set for the analysis, use the erase all button to clear the window. You can print a resume of the settings for the analysis if you wish before using the run button.

**R** This main icon is very important to Homologue users although it does very little. It is the recent analyses tool. Homologue remembers the history of all the analyses you have done, whether you have logged off or not. How about that for clever? So you have spent ages making a specific analysis. Now you want to change it—just a little bit. Hit this key and you can scan what you have been doing in the past. You will find the analysis unequivocally by time and date. Do not trust the output filenames because you may have duplicated them.



*It does not matter if you have deleted all the files relevant to the previous analysis, Homologue remembers.*

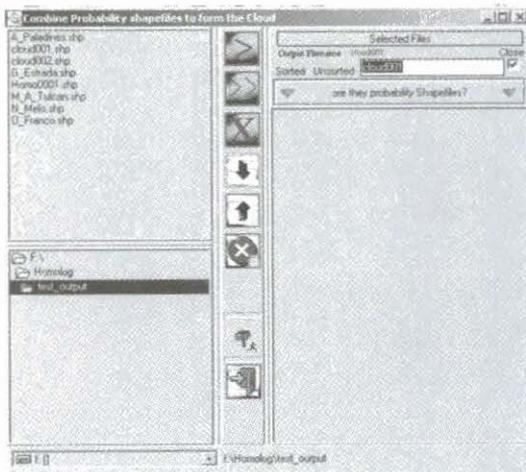
In fact, you have to tell Homologue to delete references to what you have done before. This is quite easy. All you have to do is check the Recall screen (see below).



### Select shapefiles Cloud tool



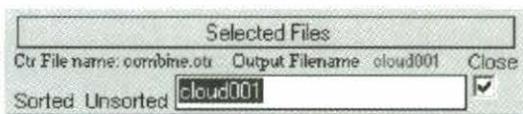
This icon selects the Cloud function. It is a way of combining various probability shapefiles. See the theory chapter to find out why you would want to do that. The window for this function (see below) is divided into two halves by a central column of buttons. The left side is for collecting files. You can use the lower directory tree and disk selector to find the relevant directories holding probability shapefiles. The shapefiles (all types in the directory) are shown in the in the top left part of the window. Select from these using the central buttons.



The following buttons control the building list on the right. These are the files to be incorporated into the Cloud.

-  Transfers the highlighted files
-  Transfers the whole list
-  Deletes the current (highlighted) filename
-  Moves the current filename down the list
-  Moves the current filename up the list
-  Deletes all building filenames

The header at the right-hand side gives you various options. You can construct the list either sorted or unsorted by alphabetic order of filename. You must supply a unique output filename for the probability shapefile of the Cloud. Homologue offers you the default name of cloud001, but if this file exists it will not proceed and will ask you for another name. The checkbox controls the window. If checked, the window will close as the Cloud is displayed on the map.



**NOTE:** *The sequencing functions have no effect on the outcome in this version of Homologue. Eventually, there will be a clustering function in the Cloud operations and the order of the input files may be significant.*

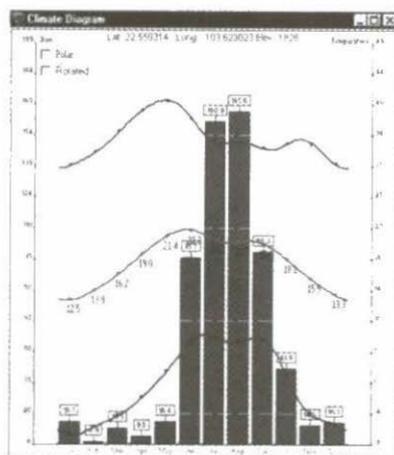
The following buttons complete the Cloud operations:

-  The running man signifies that Homologue is to get on with the job
-  Will exit this window without completing the Cloud

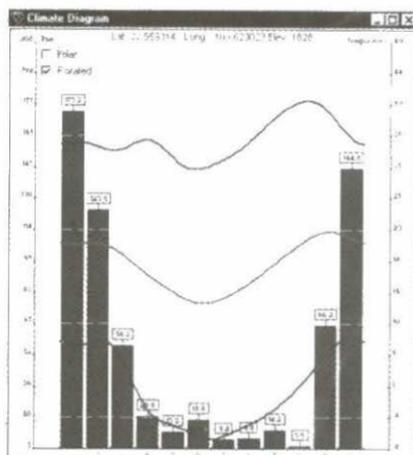
## The climate diagram tool



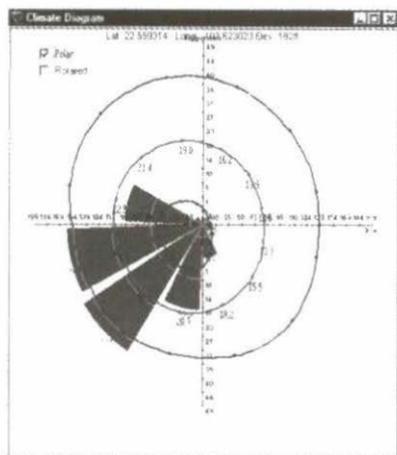
Click anywhere on the map and the following will be displayed.



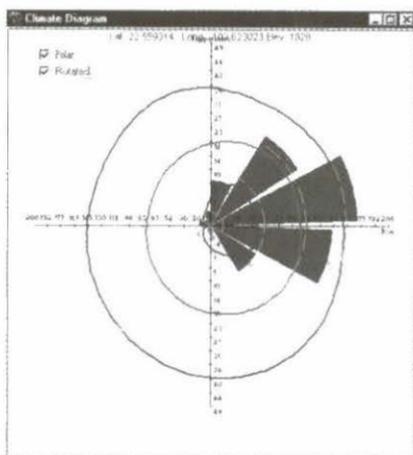
Cartesian normal



Cartesian rotated



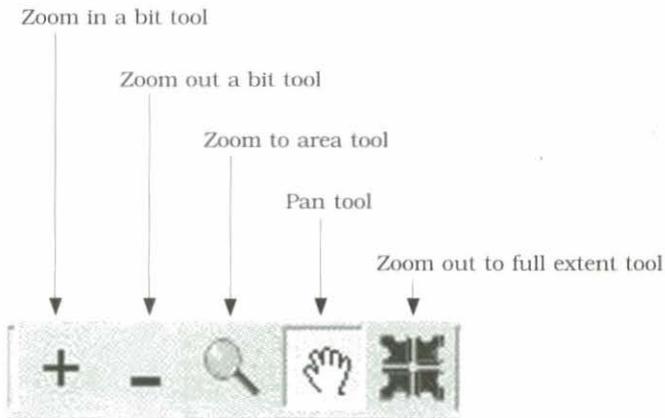
Polar normal



Polar rotated

The diagram shows maximum, minimum, and mean temperatures, and monthly mean total rainfalls. There are options for Cartesian or polar coordinates and for standard or rotated displays. (See theory section for explanation of the rotation.) Under rotation, the month names are meaningless so the months are merely numbered.

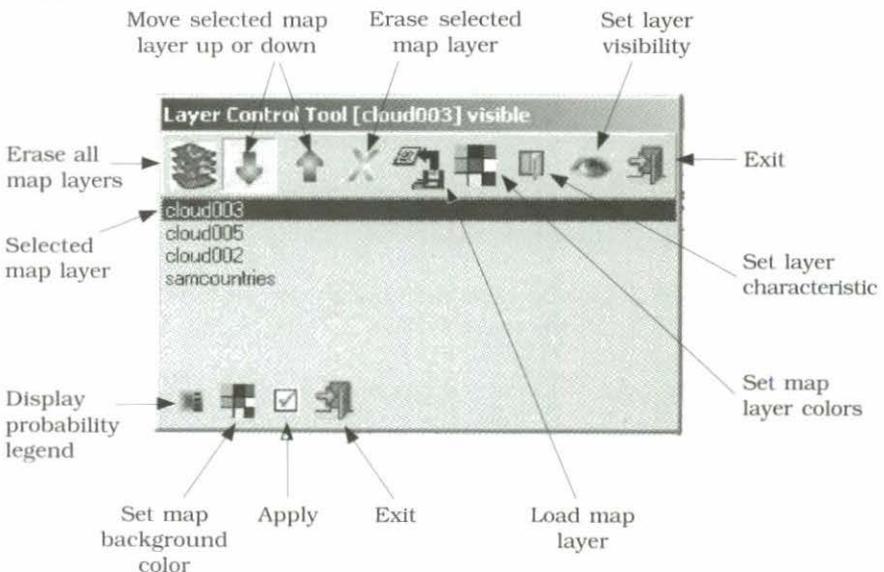
## The zoom tools



The zoom and pan operations are reasonably standard. Zoom in by drawing the desired window on the display map. Pan by pushing the map with the hand cursor. The zoom out feature is a map reset; it will zoom out to the full extent of the map. For a gradual zoom in or out, the zoom in a bit and zoom out a bit tools are available.



## The layer control tool



Each map layer is an ESRI shapefile representing geographic features that will help you identify the points you want to choose on the map. Shapefiles are described in ESRI (1998). A number of files are included on the CD-ROM. Homologue, however, will accept shapefiles from any source provided that they are in geographic coordinates (latitude, longitude) and are at a scale appropriate for the window in which you wish to work.



**NOTE:** *The files of roads and rivers included with Homologue are not suitable for display at the scales of the full map extent. However, they are useful when displaying at the department or district level, that is to say, zoomed in to large scales.*

The icon erase all map layers clears the map completely for you to start anew. It only removes them from the display map and does not affect the files themselves.

The move map layer up and move map layer down icons shift the selected layer up and down in the layer stack. When the map layers cover different regions, this has no effect on the map. However, when the layers are displayed over the same area, the stack order matters. The upper layers will obscure the lower layers. This has a variety of effects depending on the type of layer you are displaying. Closed polygon layers, such as samcountries (the country shapes for Latin America), will obscure everything beneath them. Obviously line and point files sit happily on top of closed polygon files, but would be completely obscured if they were underneath.

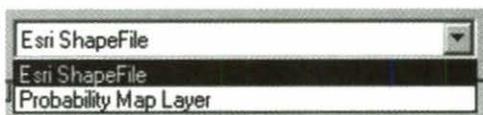


**NOTE:** *You can also drag and drop layers up or down with the cursor and mouse. However, this does not result in redrawing the image so you may not see a layer appear or disappear by this method until the map is redrawn.*

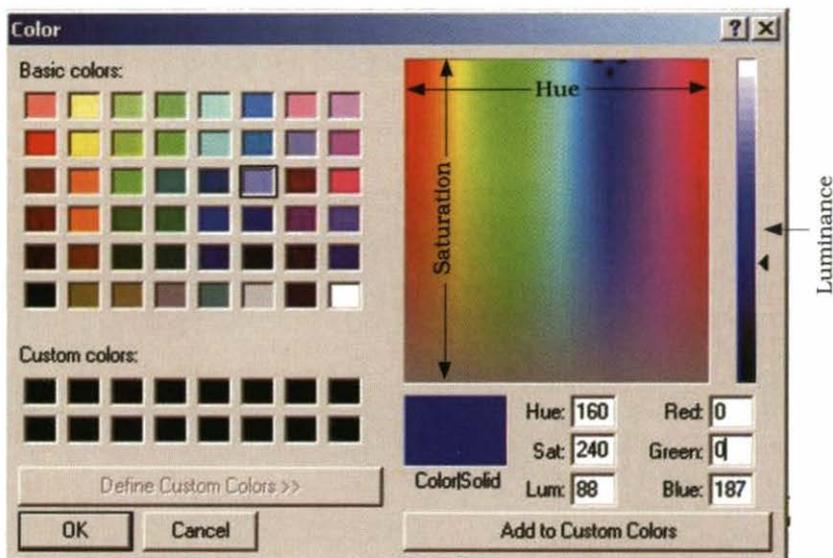
The erase selected map layer icon does precisely that; it removes the selected layer from the map. It does not delete the shapefile file.

Load a map layer will load a shapefile. You will be cued to browse for the file to load. It can be any shapefile that is appropriate to the map and anywhere that is accessible to the application. Be careful that it is compatible with the layers you are displaying. For example, you cannot see two closed polygon layers at the same time. If you wish to display a closed polygon layer (i.e., topography) below the country limits, you should use samboundaries and not

samcountries. Note that you have an option of file type to load. ESRI shapefile includes any type of shapefile that does not include a value field and can be displayed in simply defined colors (all those in the directory coverages). Probability Map Layer is a special form of rendered shapefile. It has the field PROBABILIT in the dbf file associated with it. This will produce the graded display. If you load a probability shapefile under the ESRI shapefile option you will not see the graded probabilities.

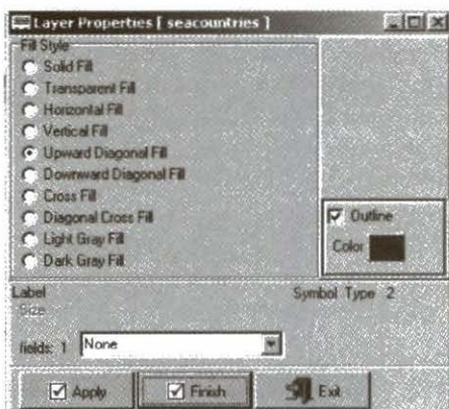


Set map layer color will take you to a color selection menu.

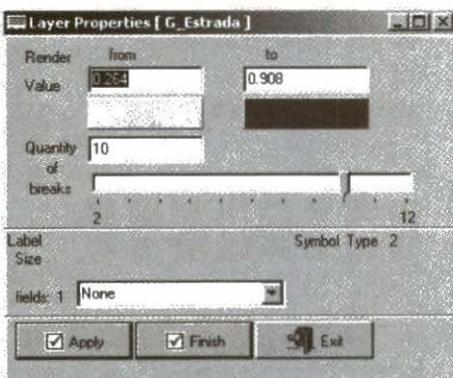


This window will first appear as only the left-hand side. You can expand it to full by hitting define custom colors. The right-hand side will appear, allowing you to select a color. You can do this by moving the cursor over the Hue/Saturation panel and moving the Luminance slider, or by keying in the red/green/blue values desired. Once you have selected your custom color, hit add to custom colors and it will appear on the left-hand side for use whenever you request the color window again in this session.

Set layer characteristic gives you more control over the layer. You can choose type of shading, and whether the layer is transparent or not. There are two formats depending on the type of coverage you are dealing with. This type (at right) is for a typical ESRI background shapefile. It allows you to choose the fill type if it is a polygon, the symbol type if it is a point file, etc. In the lower field, you can choose a data field to use as a label on the map. This is particularly useful in the case of the shapefile of towns, where you can choose the data field containing the town name.



A different format is used for a probability shapefile. In this case, the **PROBABILIT** data field contains the data on the probability to be mapped as colored shading. You can choose the minimum and maximum values for the rendering of the colors. You can set the start and end colors for the rendering. (Unfortunately, MapObjects takes only the shortest way from one color to the next. This means that elegant renders such as brown to red to yellow to white are not possible). Lastly, you can set the number of breaks in the rendering. Lower in the panel there is an option to show a label field. It is NOT recommended that you use this on a probability shapefile.



*The only case where this would be acceptable is if you wish to see the actual probability value of a pixel. Then zoom in until the pixels show as large squares on the screen and set the label to **PROBABILIT**. You will see the probability values as text centered in the pixels.*

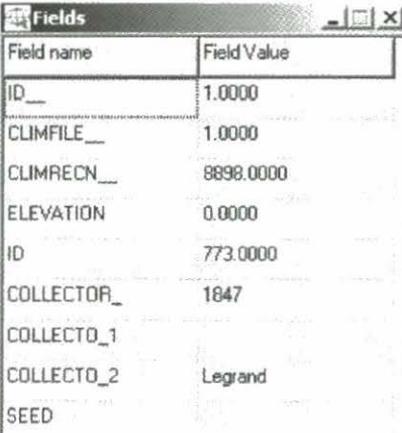
Set layer visibility leaves the layer incorporated in the map, but you can make it invisible if you wish. This is useful when you are composing an image and wish merely to test whether you should have a layer in or not. Note that the layer is cued as visible or not visible on the title bar of the window.

Display probability legend applies only to the probability shapefiles; the icon is not visible for other types of layers. It will display a legend of the probability grades used in the display in the colors you have allocated to the layer.

Set map background color allows you to change the background color. This is usually the ocean, and hence blue, but it is in fact any area not covered by a loaded layer, so this is not always the case.

### The information tool

**I** This tool will retrieve information on any feature within the current layer (that layer highlighted in the layer control tool window). A feature is a single instance of a geographic information systems (GIS) data entity: a point, line, or polygon. It is not a composite geographical entity such as Chile with all the islands, or the Amazon River with all its line segments. To select a feature, place the cursor over it and left click with the mouse. The selected feature will flash to ensure that you have the correct feature selected, and an information table will appear with the names of the fields in the dbf file and the values for that feature.



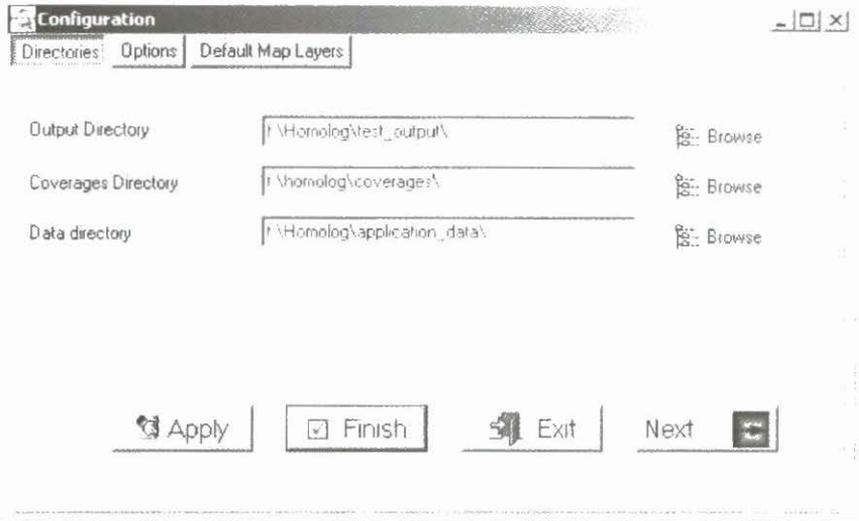
Field name	Field Value
ID__	1.0000
CLIMFILE__	1.0000
CLIMRECN__	8898.0000
ELEVATION	0.0000
ID	773.0000
COLLECTOR_	1847
COLLECTO_1	
COLLECTO_2	Legrand
SEED	

The image at left shows an example taken from a FloraMap accessions coverage loaded into Homologue. This shows the fields associated with a collected accession of *Arachis burkartii* from one of the files used to calibrate Homologue. It also shows the flexibility of Homologue. Whatever information you have in geographically projected shapefiles, you can display it in Homologue and use it to guide your analysis.

## The configuration tool



The first window available under the configuration tool holds the definitions of the paths to the directories important to Homologue. These are all defined at installation and only need to be changed here if you move the directories, or decide to use a different one.

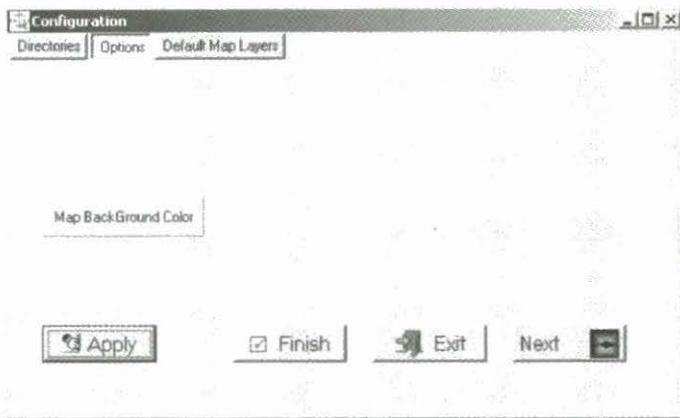


The output directory is where the output probability shapefiles will be stored. This is the one you are most likely to change, because you may choose a different output directory for each job.

The coverages directory is provided for you at installation, and holds copies of shapefiles for you to construct detailed local maps. Since Homologue can handle almost any form of ESRI shapefiles, you can add to these or store your own shapefiles in another directory. However, unless you have it defined as the coverages directory in configuration, you will have to browse for it when looking for a shapefile to load.

The data directory holds data files specific to Homologue. Although you may move the directory to any place convenient to you, the files must not be altered in any way. It is not recommended to store other files in this directory, and it is good practice to maintain it as read only.

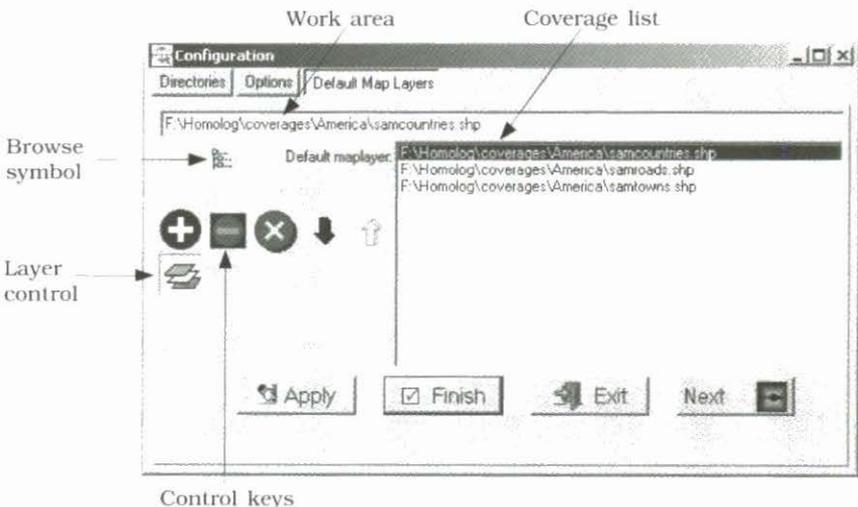
The next button defines the options available.

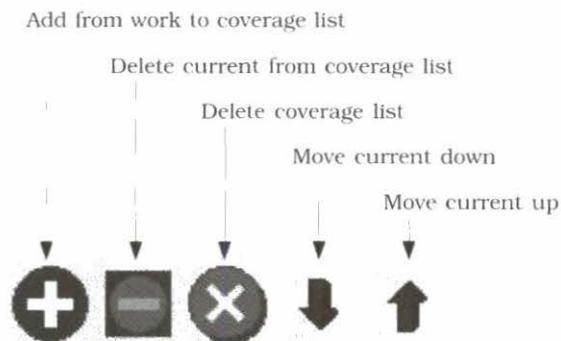


The background color is that used for the sea or for continents not displayed. It is normally set to a light turquoise, but you may want to change it for some illustration purposes.

The default map layers window is your memory for building a working map.

To enter a map layer into the coverage list, type it into the work area or use the browse symbol to look for it. Once you have it in the work area, you can add it to the coverage list with the plus key. Use the other control keys to manage the coverage list.





The layer control works exactly the same as it does on the working map, except that the specifications that you put on the layer are remembered and loaded with the default map.

## 4. Theory

Homologue is an extension of the FloraMap algorithm (Jones et al., 1997; Jones and Gladkov, 1999). The extension is in two parts: the first is to generalize the FloraMap algorithm so that it can generate a climate probability distribution from a single point; the second is to incorporate the probability of finding a soil with characteristics defined by the user. Note that FloraMap did not include soil characteristics because it was impossible to draw conclusions about the soil on which accessions were found; in Homologue we are relying on the investigator to have enough knowledge of the point s/he is trying to match to be able to provide these data. This chapter deals with the basic FloraMap method and the climate surfaces from which it works, the extension to a generic method and the inclusion of soil probabilities.

### The Climate Surfaces

Spatially interpolated climate surfaces are now available for many areas. These usually handle long-term climate normals interpolated over a digital elevation model (DEM) by various methods (Jones, 1991; Hutchinson, 1997). Pixel size depends on the underlying elevation model. It may be as little as 90 m (Jones, 1996), which results in a massive data set, or 10 minutes of arc (about 18 km), which is as large as is practicable in many instances. In the latter case, the normal elevation model is the National Oceanographic and Atmospheric Administration (NOAA) TGPO006 (NOAA, 1984). We have produced interpolated data sets at CIAT for Latin America and Africa using data from about 10,000 stations for Latin America, 7000 for Africa, and 4500 for Asia. Each set of surfaces consists of the monthly rainfall totals, monthly average temperatures, and monthly average diurnal temperature range. This makes 36 climate variates in three groups of 12.

We use a simple interpolation algorithm based on the inverse square of the distance between the station and the interpolated point. For each interpolated pixel we find the five nearest stations. Then the inverse distance weights are calculated and applied to

each monthly value of the data type being interpolated. Thus, for five stations with data values  $x$  and distances from the pixel distance  $d$ :

$$x_{pixel} = \frac{1}{\sum_{i=1}^5 d_i^{-2}} \times \sum_{i=1}^5 \frac{x_i}{d_i^2} \quad (1)$$

Temperature data are standardized to the elevation of the pixel in the DEM using a lapse rate model (Jones, 1991). Using this simple interpolation has various advantages. First, it is the fastest of all the common methods. Second, it puts the interpolated surface exactly through each station point, because the weight  $1/(d_i^{**2})$  becomes infinite as  $d$  approaches zero. Third, the interpolation is highly stable in areas of sparse data. It approaches the mean of the nearest stations while they all become equally distant. Fourth, it is relatively stable against errors in station elevation; only the local region of that station is affected. On the other hand, laplacian spline techniques and co-Kriging both propagate these errors more extensively. This is one advantage of using a proven lapse rate model instead of fitting a local one, as do both of these latter techniques.

The method has two small disadvantages. First, the derivative of the surface becomes zero as it passes through the station point. In other words, each station is on a small plateau or step in the interpolated surface. This is usually much smaller than the pixel size and hence is not noticeable. Second, a (usually small) step occurs in the fitted surface as stations come into or drop out of the fitting window. Where the station density is high with respect to the pixel size, this is almost impossible to see. Where the stations are not so dense, it can produce unsightly straight lines or smooth arcs in the fitted rainfall data, which are not tied to elevation. Inspection of the surface's profile usually shows that these are negligible artifacts, but they are unsightly and can undermine confidence in the surface maps.

## **Climate Date Standardization (Rotation)**

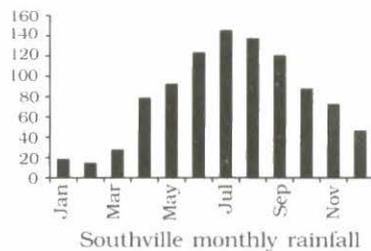
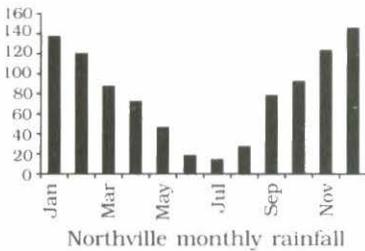
The climatic events that occur through the year, such as summer/winter and start/finish of the rainy season, are of prime importance when comparing one climate with another. Unfortunately, they

occur at different dates in many climate types. The most obvious case is where climates are compared between points in the Northern and Southern Hemispheres, but more subtle differences can be seen in climate event timing throughout the tropics. What we need is a method of eliminating these differences to allow us to make comparisons free of these annual timing effects.

Let us look at two hypothetical climate stations. They are in a typical Mediterranean climate—warm wet winters, hot dry summers. Northville could be somewhere in California, and Southville might be in Chile. The August rainfall in Southville is received in January in Northville. If we plot these rainfalls in polar coordinates, we can readily see that to compare them we need to rotate them to a standard time.

### Monthly rainfalls for Northville and Southville.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Northville	137	120	87	72	46	18	14	27	78	92	123	145
Southville	18	14	27	78	92	123	145	137	120	87	72	46



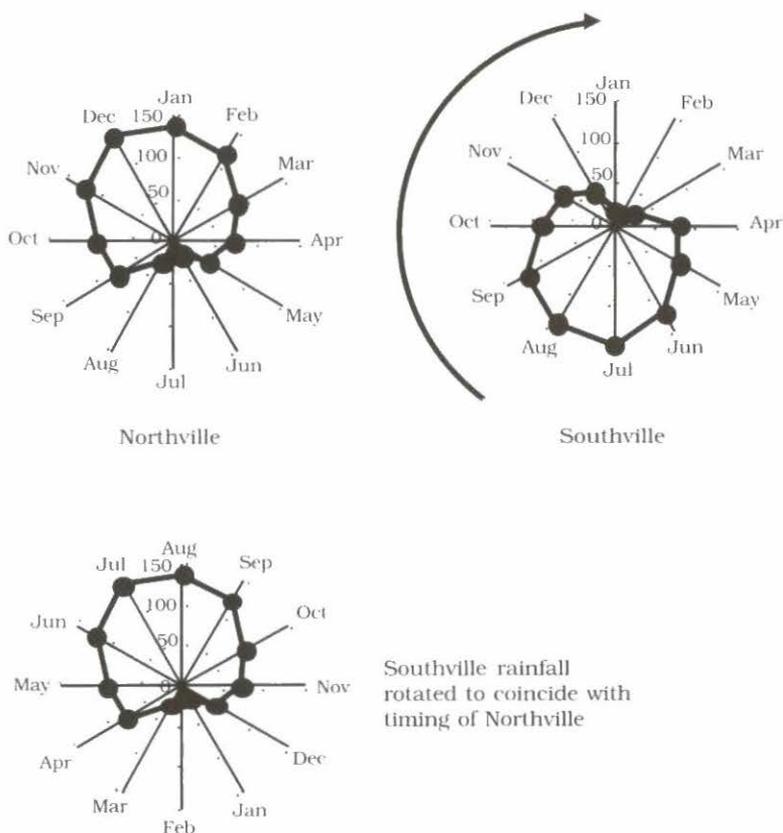
How do we do this automatically? The answer is the 12-point Fourier transform. This is fortunately the simplest of all the possible Fourier transform algorithms. It is highly computationally efficient and fast. In fact, it is the basis of nearly all Fast Fourier transform algorithms that break the problem down sequentially into the simple 12-point case. It takes the 12 monthly values and converts them to a series of sine and cosine functions. The one used in FloraMap has a modification to make it conserve the monthly total values (Jones, 1987). The equation produced is:

$$r = a_0 + \sum_{i=1}^6 a_i \sin(ix) + b_i \cos(ix) \quad (2)$$

This can be rewritten as a series of frequency vectors, each with an amplitude  $a_i$  and a phase angle,  $\theta_i$ :

$$\alpha_i = \sqrt{a_i^2 + b_i^2} \quad \theta_i = \sin\left(\frac{b_i}{\alpha_i}\right) = \cos\left(\frac{a_i}{\alpha_i}\right) \quad (3)$$

If we subtract the first phase angle from all the other vectors in the set, then we have produced a rigid rotation of the vectors. This is the rotation that we are seeking. It puts the maximum of the first frequency at a phase angle of zero and places the rest in positions equivalent to their angular separation in the original data. We then use the first phase angle for rainfall to rotate the data for temperature and diurnal temperature range, and these variates are rigidly rotated along with the rainfall.

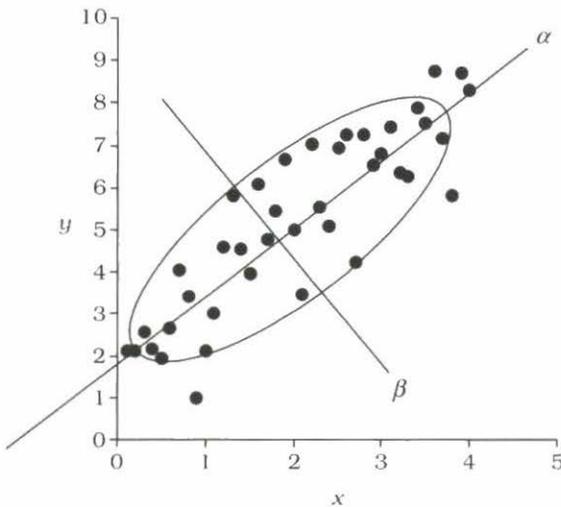


## The FloraMap Probability Algorithm

The FloraMap probability algorithm is based on a principal components analysis (PCA) of a large set of geographic coordinates from a germplasm, museum, or other collection where the original collection points have been noted. It works for just about any organism in the wild where the distribution is influenced mainly by climate (a very common occurrence). The algorithm can be seen as two separate parts: a principal components analysis that breaks down the climate data into orthogonal components; and a probability calculation, from these components, that compares any given pixel on a map to the fitted probability model.

### *The principal components analysis*

The operation can be illustrated in two dimensions as follows. Figure 4.1 shows a scatterplot of two variates,  $x$  and  $y$ , quite highly correlated and therefore not at all independent. For any change in  $x$ , we would expect a change in  $y$ . However, we can find two new axes,  $\alpha$  and  $\beta$ , such that they are not correlated, and that the variance accounted for in the first of the new axes is maximized.



**Figure 4.1.** Illustrating the rotation of correlated variables  $x$  and  $y$  to the orthogonal variables  $\alpha$  and  $\beta$ .

In this case,  $\alpha = 0.454x + 0.891y$ , and  $\beta = 0.891x - 0.454y$ . These new axes are orthogonal and uncorrelated. Movement along the  $\alpha$  axis does not imply any movement at all along the  $\beta$  axis. The component  $\alpha$  accounts for 95.6% of the original variance,  $\beta$  merely 4.4%. The trick to this linear transform is to calculate the eigenvalues and eigenvectors of the variance-covariance matrix of the system of variates. In FloraMap's case, this is a 36 x 36 matrix of climate variates.

In matrix notation, we need to find a matrix  $\mathbf{Q}$  and a diagonal matrix  $\mathbf{\Lambda}$  such that:

$$\mathbf{Q}^{-1}\mathbf{A}\mathbf{Q} = \text{diag } \hat{\lambda} = \mathbf{\Lambda} \quad (4)$$

Where  $\mathbf{A}$  is our variance-covariance matrix. The matrix  $\mathbf{\Lambda}$ , composed of the elements  $\hat{\lambda}$ , is the diagonal matrix of the eigenvalues, which in our case hold the variance of the eigenvectors. The matrix  $\mathbf{Q}$  is a symmetric matrix, which holds the eigenvectors as both rows and columns. The eigenvectors have two highly useful properties, one of which has been mentioned above—they are linearly independent of each other. The second useful property is that an eigenvector multiplied by any scalar is still an eigenvector.

A principal components analysis can be performed on the sums of squares and cross products (SSCP) matrix, the variance-covariance matrix, or on the correlation matrix of a group of variates. In FloraMap, we use the variance-covariance matrix by standardizing the variates before we calculate the SSCP. But, we differ from many standard analyses in that our data have a structure that we want to preserve rather than standardize completely. The data are actually three groups of 12 values for different climate variables—rainfall, temperature, and diurnal temperature range. We want to conserve this difference to allow the user to apply weight across the board for the climate variables, for example, increasing the importance of rainfall over that of temperature. In addition, the information across the 12 monthly values is of critical interest and we do not wish to standardize it away. We therefore standardize all rainfall values by the common variance for rainfall, and so forth.

Once we have found the  $\mathbf{\Lambda}$  and  $\mathbf{Q}$ , we can describe the system of climate variates in terms of the principal components and their variances (eigenvectors and eigenvalues). We can choose a subset of the components (because the eigenvectors are independent), and we

can scale them individually (because multiplying or dividing by a constant does not change the eigenvector's properties). This last point is important because this is exactly what we want to do to calculate the probabilities.

### Probability calculations

The normal probability density function for a single variate is given by:

$$z = \frac{1}{\sqrt{2\pi}} e^{-t^2/2} \quad (5)$$

From the integral of this function we can estimate the probability of observing a point drawn from this population.

Traditionally, we look at the probability that a point might lie further from the origin than the point in question. Also, we usually estimate the distribution parameters from the sample that we are investigating. Because of this, we use other statistics such as **Student's t** to estimate the probability. In FloraMap, we make a simplifying assumption that the accessions calibration set will contain sufficient points so that estimating from the sample will be equivalent to knowing the population parameters. This would not be true for small calibration sets, and even less for the Homologue case of a single point. We therefore assume a large calibration set and use the variance supplied by the user in the form of the expected adaptation range.

For multiple dimensions with  $n$  independent (orthogonal) variates, the probability density function becomes:

$$z = \frac{1}{\sqrt{2\pi}} e^{-(t_1^2 + t_2^2 + \dots + t_n^2)/2} \quad (6)$$

The integral of this can be obtained by repeated integration, but specifying the integration bounds for each subsequent integration in terms of the previous functions is untidy and tedious. Here is an easier way to look at it. We want the probability that any point in a distribution falls within a radius of:

$$r = \sqrt{(t_1^2 + t_2^2 + \dots + t_n^2)} \quad (7)$$

The volume of a sphere of dimension  $n$  is:

$$\frac{r^n \sqrt{\pi^n}}{\Gamma\left(\frac{n+2}{2}\right)} \quad (8)$$

Note that as  $n$  increases, the volume of the sphere tends to zero. Thus the probability integral constructed in space with large  $n$  will be counter-intuitively small.

The volume of an infinitely thin shell of this sphere at radius  $r$  is:

$$\frac{nr^{n-1} \sqrt{\pi^n}}{\Gamma\left(\frac{n+2}{2}\right)} \quad (9)$$

The derivative of the probability integral at this shell is:

$$z = \frac{nr^{n-1} \sqrt{\pi^n}}{\Gamma\left(\frac{n+2}{2}\right)} e^{-r^2/2} \quad (10)$$

Therefore the integral from 0 to  $r$  is:

$$\frac{n\sqrt{\pi^n}}{\Gamma\left(\frac{n+2}{2}\right)} \cdot \int_0^r r^{n-1} e^{-r^2/2} dr \quad (11)$$

Taking only the portion to the right of the integral sign, and dividing by the limit as  $r$  passes to infinity from the left, we have, for even dimensions:

$$\lim_{r \rightarrow \infty} \int_0^r r^{n-1} e^{-r^2/2} = \Gamma\left(\frac{n}{2}\right) 2^{(n-2)/2} \quad (12)$$

$$p = 1 - \frac{e^{-r^2/2} \left( r^{n-2} + (n-2)r^{n-4} + (n-2)(n-4)r^{n-6} \dots \Gamma\left(\frac{n}{2}\right) 2^{(n-2)/2} \right)}{\Gamma\left(\frac{n}{2}\right) 2^{(n-2)/2}} \quad (13)$$

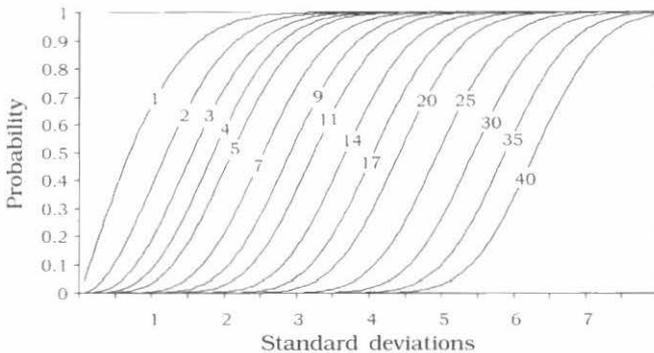
Factorizing, this becomes:

$$p = 1 - e^{-r^2/2} \left( \frac{1}{(n-2)} \frac{r^2+1}{r^2+1} \frac{(n-4)}{(n-4)} \frac{r^2+1}{r^2+1} \frac{(n-6)}{(n-6)} \dots \frac{\vdots}{(n-(n-2))} \frac{r^2+1}{r^2+1} \right) \quad (14)$$

And for odd dimensions, factorizing as we go:

$$\lim_{r \rightarrow \infty} = \frac{\sqrt{2} \cdot \sqrt{\pi} \cdot (3)(5)(7) \dots (n-2)}{2} \quad (15)$$

$$p = \operatorname{erf}\left(\frac{\sqrt{2}r}{2}\right) - \frac{\sqrt{2}r e^{-r^2/2}}{\sqrt{\pi}} \cdot \left( \frac{1}{(n-2)} \frac{r^2+1}{r^2+1} \frac{(n-4)}{(n-4)} \frac{r^2+1}{r^2+1} \frac{(n-6)}{(n-6)} \dots \frac{\vdots}{(n-(n-3))} \frac{r^2+1}{r^2+1} \right) \quad (16)$$



**Probability integral in multiple dimensions: The probability of finding a point between the origin and radius  $r$  for  $N(0,1)$  populations in selected dimensions from 1 to 40.**

This is an important result. If we did not have this, we could not maintain the correct level of probability as we passed from one set of dimensions to another. This is effectively what we do when we choose different sets of principal components.

## **Generalizing from FloraMap to Homologue**

Since we do not have the calibration set to work from in Homologue, the eigenvectors and eigenvalues have to be derived from a generalized form. To do this, we looked at the form of the models for over 300 species fitted with FloraMap. To standardize across species, we used the simple square root transformation for the rainfall and considered only the first five eigenvalues and eigenvectors. These usually account for about 95% of the overall variance unless the calibration set is very large.

The eigenvectors in many species show similar patterns, and tend to group within similar climate types. However, it was not possible to find a generic generator to simulate them. We therefore formed a simple climate classification to group the species types. This uses the mean and variance of the 12 monthly mean temperature record. By dividing these variables into five classes each, we construct a 5x5 table of climates. Mean temperature class breakpoints are: 15.5, 19.0, 22.0, and 24.0 °C. The variance class breakpoints are: 1, 2, 5, and 10.

The eigenvalues generally form a regular pattern, the first being by far the largest and the rest falling away in size sequentially. They sum to the variance. In Homologue we want the user to have control over the variance as a measure of adaptation, so we fitted a model to the log of the eigenvalues taken as a proportion of the variance. In all climate classes, regressions of the log of the percentage value of the eigenvalues against the eigenvalues number were highly significant and provide good predictors of the eigenvalues. Table 4.1 shows the estimated regression coefficients for the 25 climate classes. Note that they vary consistently with the mean and variance of temperature, giving us confidence that there is some logic behind the model. The values in bold are missing values calculated iteratively from the row and column effects.

**Table 4.1.** Regression coefficients for the eigenvalues for the 25 climate classes.

<b>a</b>							
	mean						
variance	1	2	3	4	5	mean	effect
1	4.060	<b>3.968</b>	<b>3.982</b>	<b>4.213</b>	<b>4.373</b>	4.119	0.038
2	4.275	4.007	3.988	4.108	4.165	4.109	0.027
3	3.765	3.791	3.706	3.598	4.278	3.828	-0.254
4	4.114	3.846	3.795	3.976	<b>4.250</b>	3.996	-0.085
5	3.897	4.039	4.251	4.982	<b>4.609</b>	4.356	0.274
mean	4.022	3.930	3.944	4.175	4.335	4.081	
effect	-0.059	-0.151	-0.137	0.094	0.253		
<b>b</b>							
	mean						
variance	1	2	3	4	5	mean	effect
1	-0.635	<b>-0.602</b>	<b>-0.610</b>	<b>-0.748</b>	<b>-0.738</b>	-0.666	-0.023
2	-0.769	-0.628	-0.680	-0.750	-0.643	-0.694	-0.051
3	-0.473	-0.503	-0.467	-0.493	-0.726	-0.532	0.111
4	-0.627	-0.535	-0.499	-0.608	<b>-0.656</b>	-0.585	0.058
5	-0.554	-0.624	-0.676	-1.023	<b>-0.808</b>	-0.737	-0.094
mean	-0.612	-0.578	-0.586	-0.724	-0.714	-0.643	
effect	0.031	0.065	0.057	-0.081	-0.071		

Since there was no way to generate the complex patterns of the eigenvectors, we decided to select a representative species from each climate class to provide the eigenvector template (Table 4.2). Note that *Bemisia tabaci* is not a plant, but an insect; however, it was the best representative species for that climate class. These species have various adaptation ranges, but since that is practically completely set by the variance they still provide adequate eigenvector templates.

**Table 4.2.** Representative species providing eigenvector templates for the 25 climate classes.

<b>species</b>					
	mean				
variance	1	2	3	4	5
1	<i>Oxalis tuberosa</i>	<b><i>Phaseolus leptostachyus</i></b>	<b><i>Phaseolus vulgaris</i></b>	<b><i>Vigna davyi</i></b>	<b><i>Vigna nervosa</i></b>
2	<i>Solanum nigrescens</i>	<i>Phaseolus leptostachyus</i>	<i>Phaseolus vulgaris</i>	<i>Vigna davyi</i>	<i>Vigna nervosa</i>
3	<i>Vigna parkeri</i>	<i>Vigna luteola</i>	<i>Vigna vexillata</i>	<i>Arachis burchartii</i>	<i>Arachis duranensis</i>
4	<i>Stylosanthes guianensis</i>	<i>Stylosanthes viscosa</i>	<i>Bemisia tabaci</i>	<i>Arachis glabrata</i>	<b><i>Arachis duranensis</i></b>
5	<i>Vigna ambacensis</i>	<i>Vigna venulosa</i>	<i>Manihot aesculifolia</i>	<i>Arachis cardenas</i>	<b><i>Arachis duranensis</i></b>

The templates provided by the selected species are completely variance free so we still need to estimate individual variates variances for each climate class.

Table 4.3 shows the results of estimating the variances for diurnal temperature range and rainfall proportional to that of temperature for the 25 climate classes. The variance estimates were erratic for sample sizes less than 20 species and so there are more missing values in this table because those samples were eliminated. Again, the missing values are calculated iteratively from the row and column effects.

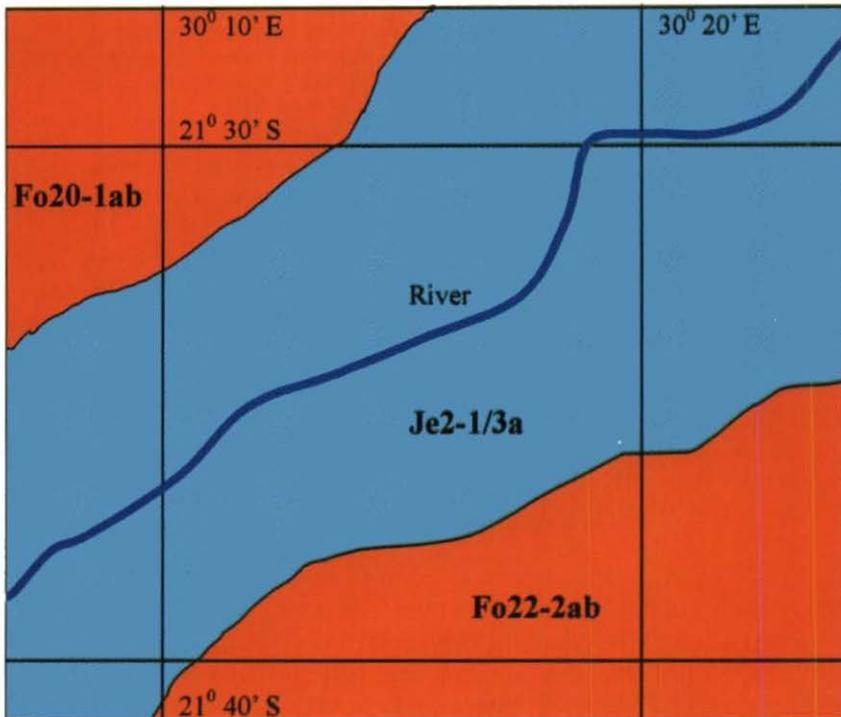
**Table 4.3.** Variances calculated as proportional to the variance for temperature for the 25 climate classes.

Diurnal temperature range variances iterated							
	1	2	3	4	5	mean	effect
1	0.428	<b>0.611</b>	<b>0.463</b>	<b>0.563</b>	<b>0.707</b>	0.554	-0.100
2	0.501	0.385	0.214	0.531	0.688	0.464	-0.190
3	0.363	0.440	0.440	<b>0.453</b>	0.525	0.444	-0.210
4	0.515	0.995	0.934	0.810	<b>1.004</b>	0.852	0.197
5	0.834	1.124	0.761	<b>0.957</b>	<b>1.110</b>	0.957	0.303
mean	0.528	0.711	0.562	0.663	0.807	0.654	
effect	-0.126	0.057	-0.092	0.008	0.153		
Rainfall variances iterated							
	1	2	3	4	5	mean	effect
1	0.454	<b>0.655</b>	<b>0.362</b>	<b>0.387</b>	<b>0.274</b>	0.426	-0.446
2	1.238	0.389	0.431	0.473	0.415	0.589	-0.284
3	0.651	0.876	0.612	<b>0.617</b>	0.525	0.656	-0.217
4	0.843	0.874	1.073	0.930	<b>0.739</b>	0.892	0.019
5	1.316	2.712	1.564	<b>1.761</b>	<b>1.648</b>	1.800	0.927
mean	0.901	1.101	0.809	0.834	0.720	0.873	
effect	0.028	0.228	-0.064	-0.039	-0.153		

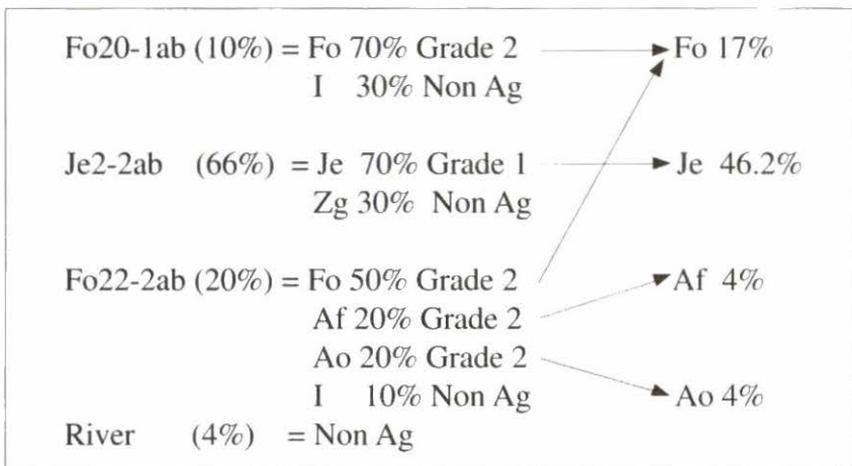
Thus we have a generic model for any climate, the breadth of adaptation being the only free parameter set by the user as the overall variance. The climate class of the calibration point is determined from the temperature record; the climate model parameters then are calculated from lookup tables, and the model is ready to run through the climate data tables to calculate the probability coverage. First, however, the soil probability model is initialized.

## The Soil Probability Model

There are two main sources of soils information that are uniform, compatible, and worldwide. These are the World Inventory of Soils Emission Potentials Database, WISE (Batjes and Bridges, 1994; Batjes, 1995), and the FAO Soils Map of the World at 1:5,000,000 (FAO, 1995). The FAO soils map gives mapping units that include a number of different soil types. Although these are not mapped in the sense that we know where they are, there are basic rules as to what percentage of the soil unit each soil type covers (see FAO, 1978). Within a climate pixel there may be varying proportions of a number of mapping units. Figures 4.2 and 4.3 illustrate how the probabilities of encountering a given soil within a climate pixel are calculated.



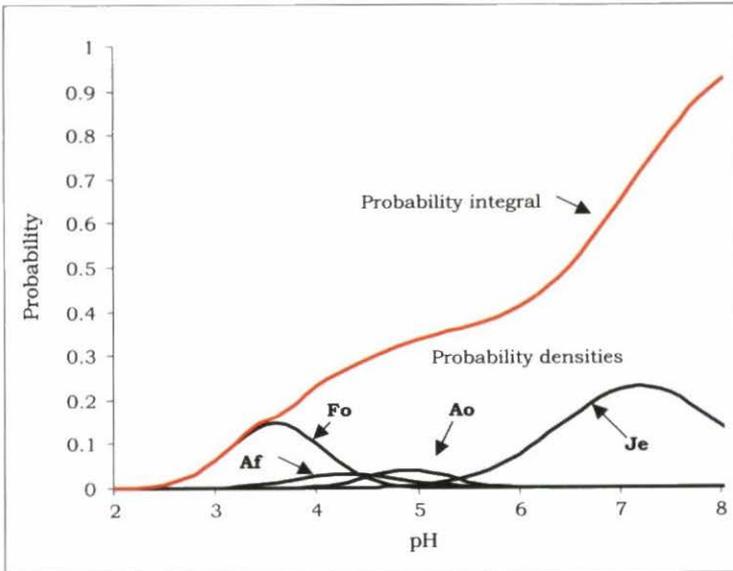
**Figure 4.2.** Schematic diagram to show a climate pixel with different proportions of three mapping units (from Jones and Thornton, 2004).



**Figure 4.3.** Schema for calculating probability areas of individual soils within the illustrated climate pixel. The left-hand column shows the pixel percentages for each mapping unit. The center column gives the member soils and their nominal percentages in each mapping unit. The result is shown on the right (after Jones and Thornton, 2004).

The map legend (FAO, 1974) gives descriptive and some quantitative characteristics of the soil types, but for quantitative data with some idea of the variance of each soil characteristic within a soil type we must turn to the WISE database. We extracted data for 11 measurable soil characteristics from over 3000 profiles in the database. These were transformed to normality where necessary, and we calculated means and variances for those with sufficient profiles within a soil type. This left considerable gaps in the table, and these were filled where possible by regression on known characteristics. Where even this failed, using analogies with known soil types filled in the table. Fortunately, this was the case for only a few of the less common soils.

Once we have the variances, we can calculate the probability that a given characteristic falls above, below, or between two values. However, what we need is the probability of encountering this condition within the climate pixel, and this involves combining the probabilities of finding the characteristic within a soil type, and the probability of finding the soil in the pixel. Thus we need to construct, for each pixel, the probability integral over all soils, for each soil characteristic. Figure 4.4 shows how this is done.



**Figure 4.4.** Illustration of how the probability densities of individual soils in Figures 4.2 and 4.3 are scaled and added to produce the overall pixel probability integral.

Each probability integral is then scaled and compressed into eight bytes, and stored in a composite structure including all the soil characteristics for each pixel.

### **Calculating the soil probability**

The soil characteristics taken from the WISE database files were:

Depth	Soil depth to the C-horizon
C	Total soil carbon content
N	Total soil nitrogen
pH	pH in water
pH KCl	pH in potassium chloride
pH CaCl <sub>2</sub>	pH in calcium chloride
CeC	Cation exchange capacity
Sand	Percentage sand
Silt	Percentage silt
Clay	Percentage clay
RD	Rooting depth

Although a number of important properties are missing from this list, notably phosphorus content, it was deemed expedient to get Homologue working as a demonstration model with at least a viable list of quantitative soil characteristics. Those more difficult to standardize and obtain sufficient data for can be added later as the data become available.

As Figure 4.5 shows, although some soil characteristics are distributed relatively independently between the 3000 profiles, many are not. This is hardly surprising between pH measures of different types. We wish, however, to create a generalized algorithm that can cope with ANY selection of soil parameters that can be entered in the future. Therefore we must cope with the vagaries of the cross correlations. The simple way to do this is to extract the eigenvalues and eigenvectors of this correlation matrix. However, the user can select any subset of these characteristics (or none at all). We therefore have to delay taking the eigenvectors and eigenvalues until that selection has been made and then take them from the subset of the correlation matrix at the time of the analysis. This yields a probability problem in a varying number of dimensions depending on the set chosen, and the problem can be solved in exactly the same way as in the climate section, using Equations 14 and 16.

Depth	1.00										
C	-0.10	1.00									
N	-0.13	0.89	1.00								
pH	-0.39	-0.34	-0.27	1.00							
pH KCl	-0.33	-0.19	-0.14	0.92	1.00						
pH CaCl <sub>2</sub>	-0.39	-0.30	-0.27	0.97	0.90	1.00					
CeC	-0.34	0.66	0.70	0.11	0.12	0.10	1.00				
Sand	-0.11	-0.41	-0.31	-0.03	-0.13	-0.02	-0.53	1.00			
Silt	-0.12	0.11	0.09	0.36	0.40	0.37	0.32	-0.64	1.00		
Clay	0.26	0.42	0.32	-0.32	-0.23	-0.33	0.37	-0.67	-0.14	1.00	
RD	0.35	-0.07	-0.09	-0.02	0.01	-0.02	-0.06	-0.13	0.12	0.05	1.00

**Figure 4.5.** The correlation matrix for 11 selected soil characteristics.

One further complication remains before we can come to combining the probabilities. Homologue allows the user to choose between using the actual probability of a characteristic falling between limits, and the mere fact that the probability exceeds a threshold value. In the latter case, the probability is evaluated and set to 1 if it exceeds the threshold, or to 0 if it falls below the threshold. In this manner, the orthogonalization and combination algorithm can operate transparently on probabilities regardless of the option selected.

Probability combination proceeds by forming the probit transforms of the probabilities (substituting a very small number for 0 and 0.99999 for 1). We then use the probits to calculate the orthogonal scores from the eigenvalues and eigenvectors; from these we calculate a radial distance in N space, where N is the number of characteristics selected and apply Equations 14 or 16 to determine the overall probability of finding a soil with the selected characteristics in the given ranges.

## **Combination of Soil and Climate Probabilities**

This is the least satisfactory part of the Homologue algorithm and really needs more work. This is because various soil characteristics can be correlated with climate (for example pH is usually, but not always, inversely correlated with rainfall). The correlation patterns generally become so complicated that it is hard to see how to allow for them. Therefore, in this version of Homologue, the algorithm for combination of orthogonal probabilities is used, assuming there is no overall pattern between climate and soils. A soil scientist would probably look askance at this, and more work needs to be done for future versions. It is quite possible that with further data, usable relationships could be resolved within various soil types. This would, however, require a substantial reworking of the algorithms.

## **The Cloud, or Combining Probability Maps Drawn from Various Calibration Points**

If we need to know the answer to the question "Where else in the world is like my plot of land?" then the above algorithm should provide a good approximation if used wisely. Once the answer to this question is known, we can look in these places for a possible crop to grow in my plot. Homologue is designed to work the other way round as well: "I've found a crop growing well at x, y, and z, where would I look for fields in which to grow it?" This is a many-to-many problem rather like that of FloraMap, except that FloraMap can only fit a climate model and it needs a large accession set to calibrate the model. It would be handy to form the envelope of a number of probability distributions to build up a picture at the full range of a species or variety from a limited number of places where it is known to do well.

The last function of Homologue Beta does just that. Merely giving the names of a sequence of probability grid files will cause it to form the grid file such that the probability for each pixel is the maximum probability from the set of pixels representing that point from the input set, or:

$$P_{o,i,j} = \max(p_{1,i,j} \dots p_{n,i,j}) \quad (17)$$

In future versions, it is planned to include clustering techniques to differentiate environmental subsets of this envelope or "Cloud"; indeed, they will be clusters of "Clouds", but for the present this is as far as the Homologue algorithms can take us.

# 5. Program Software Structure

## Overview

Homologue is written in two parts: the user interface, and the underlying algorithms. The user interface is written in Borland Delphi® and ESRI MapObjects Lite®; the author of this part is William Diaz. It is a standard modern programming system, and needs little description apart from the interfacing. The second part of Homologue is written in Visual Fortran® and operated as a Dynamic Link Library (dll) called from the user interface for a range of tasks. The author of this part is Peter Jones. We would suggest that since the dll is transparently callable from most modern languages (C++, Java, Delphi, Fortran, etc.) it should not be necessary to translate it to port to another application. However, it may be necessary to have a reasonably detailed description of its standards, data structures, and operation.

Note that some fields in ESRI shapefiles are Most Significant Byte Leading (MSBL). Because of problems of alignment in Fortran structures, the shapefile structures noted here have been redefined; in the reformatting process we have changed all MSBL fields to Least Significant Byte Leading (LSBL).

Source files are available; those specifically written for Homologue are in a directory in the Homologue development area along with a Visual Fortran workspace. There are a large number of files that are basic to CIAT climate applications and reside in the CIAT Fortran library area. Some of these have already been translated to C++, but unfortunately they are embedded in the source code of FloraMap, and we would not recommend trying to cut them out.

## Data Structures

The following data structures are used as passing parameters and basic storage units throughout the system; a brief description of the non-obvious parts is included for each where necessary.

```

type climate_structure
  sequence
  real*4      lat,long      ! decimal degree
  real*4      rain(12)     ! mm per month
  real*4      temp(12)    ! degrees centigrade
  real*4      diurn(12)   ! degrees centigrade
  integer*4   elev        ! elevation meters
  real*4      phase       ! radians
  logical*1   rotated
end type climate_structure

```

`climate_structure` is a generalized structure for a climate record used as a passing parameter. It is not the condensed storage record that follows below.

`diurn` is the monthly average diurnal temperature range, now defined by the World Meteorological Organization (WMO) as the difference between daily maximum and minimum.

`phase` is the angle through which the climate data are rotated to eliminate date effects when making comparisons of climate from different pixels.

`rotated` indicates that the data have been rotated when true, original monthly data when false. This frees us from having two separate defined structures for rotated or unrotated data.

```

type climate_record
  sequence
  real*4      corners(4)
  integer*2   nd(36)
  real        phase
end type climate_record

```

`climate_record` is a condensed version for internal storage in the Homologue fast lookup tables.

`corners` are the west, east, south, and north sides of the pixel in unprojected latitude, longitude in decimal degrees.

nd are the data values for rainfall, temperature, and diurnal temperature range converted to 2-byte integer. The temperature values (temp and diurn) are multiplied by 10 before conversion.

phase is as in `climate_structure`.

---

```

type soil_char_instruct
  sequence
  logical*1      select      ! This characteristic
                                selected when true
  real*4         low         ! Lower limit
                                requested
  real*4         high        ! Upper limit
                                requested
  integer*4      criterion   ! see below
  real*4         threshold
end type soil_char_instruct

type calling_structure
  sequence
  real*4         lat, long    ! decimal degrees
  real*4         threshold
  logical*1      selector(3) ! selects continent
                                when true
  real*4         variance
  type (soil_char_instruct) soil(11)
end type calling_structure

```

`calling_structure` is the main calling parameter to transfer soils information for the dll to create a probability coverage. In this version of Homologue, 11 possible soil characteristics are included; information on the request for each of these is transmitted in structure `soil_char_instruct`.

`calling_structure.threshold` is the lower limit of probability for displaying on the map.

`calling_structure.selector` instructs the dll to process for individual continents. A true value indicates that the continent should be mapped. The order for continents is Latin America, Africa, and Asia for selector values 1, 2, and 3.

calling `structure.variance` is a value of 1 to 5 denoting the adaptation level chosen by the user. It is used to calculate the variance for the climate probability model.

`soil_char_instruct.criterion` selects the type of probability to be calculated for each selected soil characteristic. A value of 1 indicates the probability should be used as is, a value of 2 indicates that the value 1 should be used if the calculated probability exceeds a threshold value; it should be 0 if it does not.

`soil_char_instruct.threshold` defines the threshold for the above operation if a value of 2 is selected for `soil_char_instruct.criterion`.

---

```
type probability_integral
    sequence
        integer*1 increment(8)
end type probability_integral

type soil_record
    sequence
        type(probability_integral) :: characteristic(11)
end type soil_record

type extent
    sequence
        integer*2    agricultural_area
        integer*2    total_area
end type
```

The three structures above are used to store the soils' probability and extent data in RAM for fast access when mapping. The probability integral is stored very efficiently as eight single-byte increments; these are summed and rescaled for reading out the integral value. The values in extent are used in calculation of the overall soil probabilities.

---

```
type dBaseHeader
    character*1    VersionNumber
    integer*1     LastUpdate(3)
    integer*4     RecordNumber
    integer*2     HeaderLength
    integer*2     RecordLength
    character*2   Reserved1
    character*1   IncompleteTransaction
    character*1   EncryptionFlag
    integer*4     FreeRecordThread
    character*8   reserved2
    character*1   MDXFlag
    character*1   LanguageDriver
    character*2   Reserved3
end type dBaseHeader
```

```
type dBaseField
    character*10  FieldName
    character*1   space
    character*1   FieldType
    integer*4     FieldOffset
    integer*1     FieldLength
    integer*1     DecimalCount
    character*2   Reserved4
    character*1   WorkAreaID
    character*2   Reserved5
    character*1   SetFieldsFlag
    character*7   Reserved6
    character*1   IndexFieldFlag
end type dBaseField
```

```

type dBaseFieldInfo
    character*10  Name
    integer*4    num
    integer*4    offset
    integer*1    Length
    character*7  form
end type dBaseFieldInfo

```

dBaseHeader and dBaseField are standard structures for a dBase 4-file structure; dBaseFieldInfo is an internal structure for passing information to the Homologue routine that constructs the dBaseHeader fields.

---

```

type shapefileheader ! Not ESRI standard because of box
alignment
    sequence
    integer          filelength,      ! All LSBL
                    shapetype
    real*8          box(8)
end type

type shapefilepoint
    real*8          x,y
end type

type polygon_header ! not ESRI standard because of box
alignment
    sequence
    integer record_number,content_length ! All LSBL
    real*8  box(4)
    integer numparts,numpoints ! shapetype is known 5
end type

type index_record
    integer          offset
    integer          content_length
end type index_record

```

---

ShapeFileHeader and polygon\_header are not ESRI standard structures because the ESRI formats put the real\*8 box fields on an odd word boundary and Visual Fortran cannot handle this. Routines are provided for reading and writing these structures to the correct ESRI structures in the files.

---

## **Software Structures and Operation**

The dll Homologue contains only one user-called subroutine, `homologue_driver`. The other subroutines are all called by `Homologue driver` (see dll module map, p. 89).

### ***Subroutine homologue\_driver***

The first action is to expose subroutine `homologue_driver` to users of the dll. This is done with a standard `dllexport` declaration. The parameters are similarly declared public. Note that they are all called by reference; this is important for cross language communication.

The subroutine has four almost independent actions called by the Delphi routines depending on the user's choice.

Action 1 loads the data. All data to be used in the session are loaded into RAM to speed operation and allow the scientist/user to play with ideas without having to wait for the answer. The Delphi script calls action 1 once only; if it fails, the session is closed with an error message. The most likely cause of failure is incorrect path information for the data files.

Action 2 returns a climate structure for the pixel requested. The Delphi script loads the required latitude and longitude into the climate structure and action 2 fills in the rest.

Action 3 is the main one for Homologue. It accepts an input structure and creates a probability grid as a shapefile. The Delphi script then displays it.

Action 4 is the Cloud formation. It takes a range of shapefiles and constructs the joint probability map. The Delphi script then displays the map.

**Calling parameters**

- |                    |  |
|--------------------|--|
| 1. action          | integer, values 1 to 4 as described above            |
| 2. data_path       | path to the data directory                           |
| 3. output_path     | path to the output directory                         |
| 4. output_filename | output filename, less path—must be a unique filename |
| 5. climate         | climate structure, required variates depend on call  |
| 6. prob            | action 3 calling structure                           |
| 7. return_code     | return code, used also as input—see below            |

All parameters must be present for every call, but they may be unset if not required. For action 1, they may all be unset apart from action and return\_code. For action 2, only the latitude and longitude need to be set in climate, in addition to action and return\_code. For action 3, all parameters must be set. For action 4, only action, output\_path, output\_filename, and return\_code need to be set.

return\_code has a special meaning on input. It is used as a debugging indicator. The last four bits, set to 1, indicate a request that the input parameters are dumped to the file *homologue.log* when that particular action is requested. Thus, if bit 0 is set, the value of return code is 1, and action 1 results in an input parameter dump. If return\_code has the value 5, then bits 0 and 2 are set, and actions 1 and 3 result in a dump. If return\_code is 15, then all four bits are set, and dumps are made for all four actions.

**Non-intrinsic routines called**

- |                            |   |
|----------------------------|---|
| 1. convert_Cstring(string) | Converts a C format string to a Fortran string—removes the null character from the end of the string. |
| 2. check_path_end(path)    | Checks that the string that defines a path concludes with a \ character; if not, it adds one.         |

- |    |                       |   |
|----|-----------------------|---|
| 3. | CreateFileMapping     | Sets up a file mapping area for interchanging process indicator information with the Delphi script. Made available by the Use DFWIN statement. See Microsoft documentation for details <sup>1</sup> . |
| 4. | MapViewOfFile         | As for 3, but finds the address of the newly created file mapping area.   |
| 5. | debug_write           | Homologue routine for dumping the parameters to the file homologue.log on entry.  |
| 6. | open_files            | Action 1 executive routine  |
| 7. | get_climate           | Action 2 executive routine  |
| 8. | probability_shapefile | Action 3 executive routine  |
| 9. | combine_probabilities | Action 4 executive routine  |

### ***Subroutine open\_files***

Action 1, the subroutine loads the data into common blocks to be held over between calls to the dll. No data reads are made after the call to `open_files`.

### **Calling parameters**

- |    |             |                                |
|----|-------------|--------------------------------|
| 1. | data_path   | Path to data files             |
| 2. | output_path | Unused                         |
| 3. | memory_loc  | Address of file mapping memory |
| 4. | return_code | Integer, output only           |

`open_files` creates two common blocks called `homologue_climate_data` and `homologue_soil_data`. Note that these common blocks are tailored exactly to the grid structure of the data files. This can be readily changed if the grid sizes change, but it will not be easy to keep this structure if the option of varying grid sizes is employed. This will be the main area for serious work when changing to 1-km grids.

1. [http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcesdkr/html/\\_wcesdkr\\_Win32\\_CreateFileMapping.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcesdkr/html/_wcesdkr_Win32_CreateFileMapping.asp)

## Non-intrinsic routines called

1. CopyMemory Routine to copy the process indicator value to the file mapping area.<sup>2</sup>

## Files read

The path is either the default data path, or as changed by the user in the configuration tool. This will depend on where the files were placed on installation. All files are binary. There are four groups of files. Record (or logical record) numbers are: Asia (42884), Africa (90696), and America (64740). The images are grids with the following bounds: Asia (626,294), Africa (420,432), and America (510,540).

1. 'asia.DA.new.rotated', 'africa.DA.new.rotated', 'america.DA.new.rotated'. These are direct access files with record length 92. The record structure is climate\_record (see data structure section).
2. 'asia.index', 'africa.index', 'america.index'. These three files are integer\*4 images containing the index from lat long (row, column) to the direct access file record number. The files are binary image structure.
3. 'elevasia.img', 'elevafri.img', 'elevamer.img'. These are IDRISI integer\*3 binary images of elevation that match the index files.
4. 'asia.soil', 'africa.soil', 'america.soil'. These are binary array images of length = logical record number equal to the direct access files. The data structure is soil\_record (see data structure section).
5. 'asia.soil.extent', 'africa.soil.extent', 'america.soil.extent'. These are binary array images of length = logical record number equal to the direct access files. The data structure is extent (see data structure section).

## Other data structures

Although not read by open\_files it is apposite here to mention the rest of the data held in the common block /homologue/ and set by the block data script Homologue\_block\_data.

1. a, Regression constant for eigenvalues array 5x5

2. <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/memory/base/copymemory.asp>

- |    |         |  |
|----|---------|--|
| 2. | b,      | Regression slope for eigenvalues array 5x5 |
| 3. | rvar,   | Variance adjustment for rainfall array 5x5 |
| 4. | dvar,   | Variance adjustment for rainfall array 5x5 |
| 5. | eig,    | Eigenvectors array 36x5x5x5                |
| 6. | limits, | Soil characteristics limiting values       |

Common block /correlations/ holds the 11x11 correlation matrix for the soil characteristics.

### ***Subroutine get\_climate***

Action 2 takes a climate structure with latitude and longitude defined, and fills in the rest of the data for that pixel. The data are returned in non-rotated form, although the Delphi script then rotates it so that the climate diagram first shows rotated data for quick comparison of different places.

### **Calling parameters**

- |    |             |                                 |
|----|-------------|---------------------------------|
| 1. | climate     | Climate structure, input/output |
| 2. | return_code | Integer, output only            |

### **Non-intrinsic routines called**

- |    |               |   |
|----|---------------|---|
| 1. | rotate_record | Rotates the data in the climate structure and sets the rotation indicator accordingly |
|----|---------------|---|

### ***Subroutine probability\_shapefile***

Action 3 constructs a shapefile of a probability grid from user input requirements. This is the main routine of Homologue. It first calculates the eigenvectors and eigenvalues of the correlation matrix of the selected soil characteristics. It then retrieves the climate data from a user-defined site. It fits a generalized FloraMap-type model to the adaptation level the user requires. It calculates the probability of encountering a soil with the user-defined characteristics, using the eigenvectors to produce orthogonal probabilities. It then combines

these probabilities and writes out a rendered shapefile for display by the Delphi script.

### **Calling parameters**

- |                    |   |
|--------------------|---|
| 1. output_filename | Output filename with complete path  |
| 2. prob            | Calling structure with complete information on latitude, longitude, adaptation, and soil characteristics required |
| 3. memory_loc      | Memory location for process indicator   |
| 4. return_code     | Return code, output only  |

### **Non-intrinsic routines called**

- |                             |   |
|-----------------------------|---|
| 1. orthogonal_probabilities | Combines an arbitrary number of orthogonal probabilities from an n dimensional normal distribution      |
| 2. compare_climates         | Uses a generic FloraMap model to compare the climate of a map pixel with that of the user's chosen site |
| 3. soil_prob_calc           | Calculates the probability of encountering soils with the characteristics specified by the user         |
| 4. eigen_real_symmetric     | Eigenvalues and eigenvectors for a real symmetric matrix  |
| 5. CopyMemory               | Copies the process indicator value  |
| 6. dbf_field                | Constructs the dbf field information  |
| 7. write_dbf_header         | Writes a dBase4 file header   |
| 8. bytinv4                  | Inverts the bytes of a 4-byte word  |

### ***Subroutine combine\_probabilities***

Action 4, constructs the Cloud by combining the probabilities from a set of probability shapefiles. The subroutine reads the full paths and filenames for a set of probability shapefiles from a control file produced by the Delphi script. It then reads through each file,

writing in the probabilities to an internal grid, replacing the grid value if a subsequent file has a higher value. A direct access scratch file receives the shapefile polygon information whenever a new pixel is identified. Once complete, shapefile headers are calculated, the new box is calculated from the box limits of the input files, and the headers are written to the output shapefile. Then the internal grid is scanned, and, for each pixel with a non-zero probability, the relevant polygon is read from the direct access scratch file. Finally, the dbf record, shapefile index record, and shapefile polygon record are written.

### Calling parameters

- |    |             |                                       |
|----|-------------|---------------------------------------|
| 1. | fn          | Full output filename                  |
| 2. | memory_loc  | Memory location for process indicator |
| 3. | return_code | Return code, output only              |

### Non-intrinsic routines called

- |    |                       |                                      |
|----|-----------------------|--------------------------------------|
| 1. | bytinv4               | Inverts the bytes of a 4-byte word   |
| 2. | read_shapefileheader  | Reads an ESRI shapefile header       |
| 3. | read_polygonheader    | Reads an ESRI polygon header         |
| 4. | CopyMemory            | Copies the process indicator value   |
| 5. | dbf_field             | Constructs the dbf field information |
| 6. | write_dbf_header      | Writes a dBase4 file header          |
| 7. | write_shapefileheader | Writes an ESRI shapefile header      |
| 8. | write_polygonheader   | Writes an ESRI polygon header        |

### *real\*4 function soil\_prob\_calc*

This routine calculates the probability of finding, in each pixel, a soil with the characteristics specified by the user. The probability of each selected soil characteristic is calculated. If the threshold option is selected, and the probability is greater than the threshold, then it is set to 1; if it is lower than the threshold, it is set to zero. The eigenvalues routine already has returned the eigenvalues in ascending order, and the parameters start and n denote the range of eigenvalues accounting for 95% of the variance. These eigenvectors are used to calculate the scores in n-start+1 dimensioned space. The

probabilities are transformed to orthogonal standard normal deviates, and we use the  $n$  dimensional probability routine developed for FloraMap to estimate the overall probability.

### Calling parameters

- |                |   |
|----------------|---|
| 1. prob        | Calling structure with soil characteristic limits and options |
| 2. s           | Structure containing the soil probability integrals           |
| 3. eigvec      | Eigenvectors  |
| 4. eigval      | Eigenvalues in ascending order                                |
| 5. start       | Index of the first significant eigenvalues                    |
| 6. n           | Number of soil characteristics selected                       |
| 7. return_code | Integer, output only  |

### Non-intrinsic routines called

- |              |  |
|--------------|--|
| 1. probndim  | Probability of exceeding a standard normal deviate in $n$ dimensions   |
| 2. prob_soil | Homologue-specific routine for calculating the probability of a soil characteristic falling between two limits from an 8-byte increment coded probability integral |
| 3. probit    | CIAT climate applications library function returning the value of a standard normal deviate from a probability   |

### **Function *bytin4***

This function reverses the byte order of a 4-byte word. It thus can be used to change between LSBL and MSBL, and the reverse. It can be used for both integer and real variables with care in declaration.

**Calling parameter**

- |    |      |  |
|----|------|--|
| 1. | word | Word for byte re-sequence, integer or real |
|----|------|--|

**Return value**

- |    |        |   |
|----|--------|---|
| 1. | bytin4 | Re-sequenced word, integer or real depending on declaration |
|----|--------|---|

***real\*4 function compare\_climates***

Homologue function for comparing climates according to the generic FloraMap model derived for the climate type and the user-defined variance. The function uses the eigenvectors and eigenvalues defined in the data blocks associated with the 5x5 tables of the generic model. It first classifies the target climate, and then calculates the standard normal variates' distance from the scores provided by the eigenvectors and variance applied to the scaled differences between the target climate and the pixels for comparison. On obtaining this distance, it uses the FloraMap routine probndim to calculate the overall probability.

**Calling parameters**

- |    |             |  |
|----|-------------|--|
| 1. | x           | Vector of 36 climate data for model centroid |
| 2. | y           | Vector of 36 climate data for compared pixel |
| 3. | var         | Variance                                     |
| 4. | return_code | Integer, output only                         |

**Non-intrinsic routines called**

- |    |                  |  |
|----|------------------|--|
| 1. | classify_climate | Uses the 12 monthly temperature values to classify the climate of the target site for the model centroid into a 5x5 matrix |
| 2. | probndim         | Probability of exceeding a standard normal deviate in n dimensions   |

***real\*4 function orthogonal\_probabilities***

This is merely a convenience function to call `probndim` without the need to perform the conversion to standard normal deviates and the possibilities of error due to overflow or underflow.

**Calling parameters**

1. `p` Array of orthogonal probabilities
2. `n` Size of array `p`
3. `error` Logical error flag, true on error

**Return function**

1. `orthogonal_probabilities` Real return single probability

***Function convert\_Cstring***

Converts a C-type string to a Fortran-type string. It does this by searching a C-type string for the terminal null character, and padding out 255 blank characters in the return string.

**Calling parameter**

1. `aC` C-type input string, must be less than 255 characters

**Return value**

1. `convert_Cstring` Fortran-type string, length 255 characters

***real\*4 function probndim*****Probability integral in  $n$  dimensions**

This returns the probability  $q$  that a point lies between the origin and radius  $r$ . In one dimension, this is related to the normal probability integral for a two-tailed test by  $q = 1-p$  (two tailed).

Values of  $r$  less than 0 have no meaning, and are treated as an error condition.  $n$  is assumed to be integer, and has no meaning when negative. Very large values of  $n$  will be evaluated correctly in most cases, but will result in long computation times. When  $r$  becomes so great that  $e^{-(r^2/2)}$  might create an underflow, it is assumed that the function is in the limit, and the value 1.0 is returned. The test  $r < 13$  effectively does this as it is approximately the limit of  $r$  where  $p > 1$  as  $n \rightarrow \text{inf}$ .

### Accuracy

To 6th decimal place up to  $n = 121$ .

### Calling parameters

- |          |   |
|----------|---|
| 1. $r$   | Radial distance from a multidimensional normal centroid |
| 2. $n$   | Dimensionality of the space                             |
| 3. error | Logical error condition, true on error                  |

### Return value

- |             |                   |
|-------------|-------------------|
| 1. probndim | Real, probability |
|-------------|-------------------|

### *real\*4 function prob\_soil*

This function calculates the probability that the value of a particular soil characteristic will exceed or fall short of a target value. It uses a special condensed version of the probability integral compressed into 8 bytes of increments. The first operation is to convert the integer\*1 array to integer\*2, eliminating negatives. We then calculate the 10-valued integral by summing increments between known endpoints and applying a multiplier that is characteristic dependent.

The probability that the characteristic will exceed or fall short of the target is then calculated by interpolation, and the direction set by the parameter relation.



**Calling parameters**

- |              |  |
|--------------|--|
| 1. prob      | Index number of the soil property (1 to 11)                            |
| 2. relation  | Relation desired, 'GT' or 'LT'   |
| 3. limit     | Target value for the probability interpolation                         |
| 4. increment | 8-valued single byte array holding the probability integral increments |

**Return value**

- |              |                   |
|--------------|-------------------|
| 1. prob_soil | Real, probability |
|--------------|-------------------|

***real\*4 function probit***

CIAT climate applications library routine returns the standard normal deviate associated with a probability.

***Subroutine rotate\_record***

This routine takes a `climate_structure` and rotates the climate data rigidly through an angle depending on the first phase angle of rainfall. A 12-point Fourier transform determines this. For more information, see the relevant subroutines. The routine checks the variable `climate_structure` rotated to determine in which direction to rotate the data, and resets the logical switch accordingly.

This version of *Homologue* uses the older version of the `rotate` algorithm. This will be updated to the new CIAT climate data rotation in the next version. This was necessary because the *FloraMap* models used to produce the generic model were all fitted using the old routine. The new one is better for the temperate regions, and will be necessary when *Homologue* is extended to these regions.

**Calling parameters**

- |            |  |
|------------|--|
| 1. climate | <code>climate_structure</code> to be rotated |
|------------|--|

### Non-intrinsic routines called

- |    |        |  |
|----|--------|--|
| 1. | rotate | Original FloraMap and MarkSim rotation routine |
|----|--------|--|

#### **Subroutine rotate**

This is the original FloraMap and MarkSim rotation routine. It encodes the data using a 12-point Fourier transform (Jones, 1987). If the parameter phase is zero, it extracts the first phase angle and rotates the data by this angle, sets the phase parameter to the angle, then decodes the data using the reverse of the Fourier algorithm. If the phase parameter is not zero it takes this as the angle through which the data should be rotated. It operates on a 12-valued array of monthly climate data and not a climate\_structure as the above routine.



**NOTE:** This routine uses *encode* and *decode*. It was written before these were available as intrinsic Fortran functions.

#### **Calling parameters**

- |    |       |                             |
|----|-------|-----------------------------|
| 1. | a     | Climate data, Real Array 12 |
| 2. | phase | Rotation angle              |

### Non-intrinsic routines called

- |    |        |  |
|----|--------|--|
| 1. | encode | CIAT climate applications Fourier encode routine |
| 2. | frrota | Rotates an encoded data array                    |
| 3. | decode | CIAT climate applications Fourier decode routine |

#### **Subroutine encode**

This routine encodes a 12-point array a into its 12-point Fourier coefficient storage form in array f. The curve fitted is not the curve that joins the data points, but rather that curve from which the 12 monthly means would have been calculated by integration over

monthly periods. *f* contains only 12 valid coefficients. *f*(13) is always zero on exit from this routine. It is present for compatibility with other routines that produce rotation in the phase space.

### **Calling parameters**

1. *a* 12-valued array for transformation
2. *f* 13-valued array of Fourier coefficients

### ***Subroutine decode***

Converts Fourier coefficient back into the exact monthly means from which they were calculated. Note that this includes the adjustment for fitting to the mean value and not the central points.

### **Calling parameters**

1. *f* 13-valued array of Fourier coefficients
2. *a* 12-valued array for transformation

### ***Subroutine frrota***

Rotates the curve described by the Fourier coefficients in *coef*:

1. If parameter *thru* is equal to zero, then rotates until the first phase angle is zero. Then returns the angle rotated through in phase.
2. If the parameter *thru* is non-zero, rotates through the angle specified.

### **Calling parameters**

1. *coef* 13-valued array of Fourier coefficients
2. *phase* Rotation angle



**Non-intrinsic routines called**

- |    |         |  |
|----|---------|--|
| 1. | freq    | Converts Fourier coefficients to phase angles and amplitudes |
| 2. | freqinv | Reverse of freq  |

**Subroutine freq**

Returns amplitudes and phase angles for the six frequencies of a data record. The phase angles are corrected for frequency, and are in radians.

**Calling parameters**

- |    |         |  |
|----|---------|--|
| 1. | amp     | Amplitudes, real array 6, output           |
| 2. | phase   | Phase angles, real array 6, output         |
| 3. | average | Mean, real, output                         |
| 4. | coef    | Fourier coefficients, real array 13, input |

**Non-intrinsic routines called**

- |    |       |   |
|----|-------|---|
| 1. | angle | Returns full circle angle in radians from sine and cosine |
|----|-------|---|

**Subroutine freqinv**

Takes the amplitudes and phase angles of the 12-point Fourier transform, and reconstitutes them as a 13-point coefficient array.

- |    |         |   |
|----|---------|---|
| 1. | amp     | Amplitudes, real array 6, input             |
| 2. | phase   | Phase angles, real array 6, input           |
| 3. | average | Mean, real, input                           |
| 4. | coef    | Fourier coefficients, real array 13, output |

**Subroutine check\_path\_end**

This Homologue routine checks a path string to see if it has a '\ ' character at the end; if not, it places one there.

**Subroutine *debug\_write***

This is a highly specific Homologue routine that is only used in development. It allows the Delphi developer to check the values of parameters and structures passed to the dll. It is turned on through a non-standard use of the return code. If the last four bits of the return code are set on entry to the dll, the input parameters are written to the file *homologue.log*. Bit 0 controls action 1, bit 1 controls action 2, bit 2 controls action 3, and bit 3 controls action 4.

Thus, if the value of return code is set at 1, the input parameters are dumped when action 1 is invoked. If it is set to 5, they are dumped if action 1 or action 3 is invoked. If it is set to 15, the input parameters are dumped for every action. Under normal conditions, the return code is always set to 0, so the user never sees this facility.

**Subroutine *eigen\_real\_symmetric***

This routine uses subroutines obtained from the package EISPAC. The original is due to Burton S. Garbow, mathematics and computer science division, Argonne National Laboratory in a version dated August 1983. It has been simplified for inclusion into Homologue.

**Calling parameters**

- |         |  |
|---------|--|
| 1. n    | Order of the real symmetric matrix                                 |
| 2. a    | The real symmetric matrix  |
| 3. w    | Eigenvalues in ascending order                                     |
| 4. z    | Eigenvectors   |
| 5. ierr | Error integer, 0 no error, number of nonzero eigenvalues otherwise |

**Non-intrinsic called routines**

- |          |   |
|----------|---|
| 1. tred2 | Reduces a real symmetric matrix to a tri-diagonal matrix  |
| 2. rql2  | Produces eigenvalues and eigenvectors from tri-diagonal matrix by the method of Bowdler et al. (1968) |

3. `pythag`                      Calculates double square root of  $(a^{**2} + b^{**2})$  without producing overflow or underflow



**NOTE:** All of these routines are documented in the source and, because they have not been modified since the version of Garbow (1983, see above), they are not described in detail here.

### **Subroutine `dbf_field`**

Constructs the *n*th dbf field descriptor in the common block `/field_descriptor_array_table/nf,field(1000)`, where *nf* is the number of actual fields and `field` has the structure `dBaseField`. The parameters set only the necessary fields for a valid dBase IV file. This routine does not write out the header information.

### **Calling parameters**

- |                             |   |
|-----------------------------|---|
| 1. <code>number</code>      | Number of the field to be constructed, integer, input |
| 2. <code>name</code>        | Field name, character*10, input                       |
| 3. <code>field_type</code>  | Field type, allowed (CDFGLMNP), character*1, input    |
| 4. <code>length</code>      | Field length, integer, input                          |
| 5. <code>decimals</code>    | Number of decimals, integer, input, numeric only      |
| 6. <code>return_code</code> | Return code, integer, output                          |

### **Subroutine `write_dbf_header`**

Writes out a dBase IV file header using information from structures and the common block `/field_descriptor_array_table/`. The file must be declared `form='binary'`. Note that this is defined as completely undelimited binary and may not be available in some older languages.

**Calling parameters**

- |    |                |  |
|----|----------------|--|
| 1. | number_records | Number of records to be written in the file. This is mandatory, you cannot write a dBase file on the fly. It is best to have all information either in RAM or written to a scratch file before writing this header. Integer, input |
| 2. | unit_number    | Logical file number, integer, input  |
| 3. | return_code    | Integer, output  |

***Subroutine read\_shapefileheader***

Reads an ESRI shapefile header into the CIAT-defined ShapeFileHeader structure. As mentioned elsewhere, the ESRI shapefile header is an awkward structure with MSBL and LSBL words and non-standard word boundaries. This routine translates all this into a more easily interpreted structure.

**Calling parameters**

- |             |   |
|-------------|---|
| unit        | Logical file number, integer, input                   |
| head        | Header information, ShapeFileHeader structure, output |
| return_code | Integer, output                                       |

**Non-intrinsic routines called**

- |    |         |   |
|----|---------|---|
| 1. | bytinv4 | Inverts the byte order in a 4-byte word |
|----|---------|---|

***Subroutine read\_polygonheader***

Reads an ESRI shapefile header into the CIAT-defined Polygon\_Header structure. Translates ESRI shapefile header as explained above.

**Calling parameters**

unit	Logical file number, integer, input
head	Header information, Polygon_Header structure, output
return_code	Integer, output

**Non-intrinsic routines called**

1. bytin4                      Inverts the byte order in a 4-byte word

***Subroutine write\_shapefileheader***

Writes an ESRI shapefile header from the CIAT-defined ShapeFileHeader structure. Translates ESRI shapefile header as explained above.

**Calling parameters**

unit	Logical file number, integer, input
head	Header information, Polygon_Header structure, output
return_code	Integer, output

**Non-intrinsic routines called**

1. bytin4                      Inverts the byte order in a 4-byte word

***Subroutine write\_polygonheader***

Writes an ESRI shapefile header from the CIAT-defined Polygon\_Header structure. Translates ESRI shapefile header as explained above.

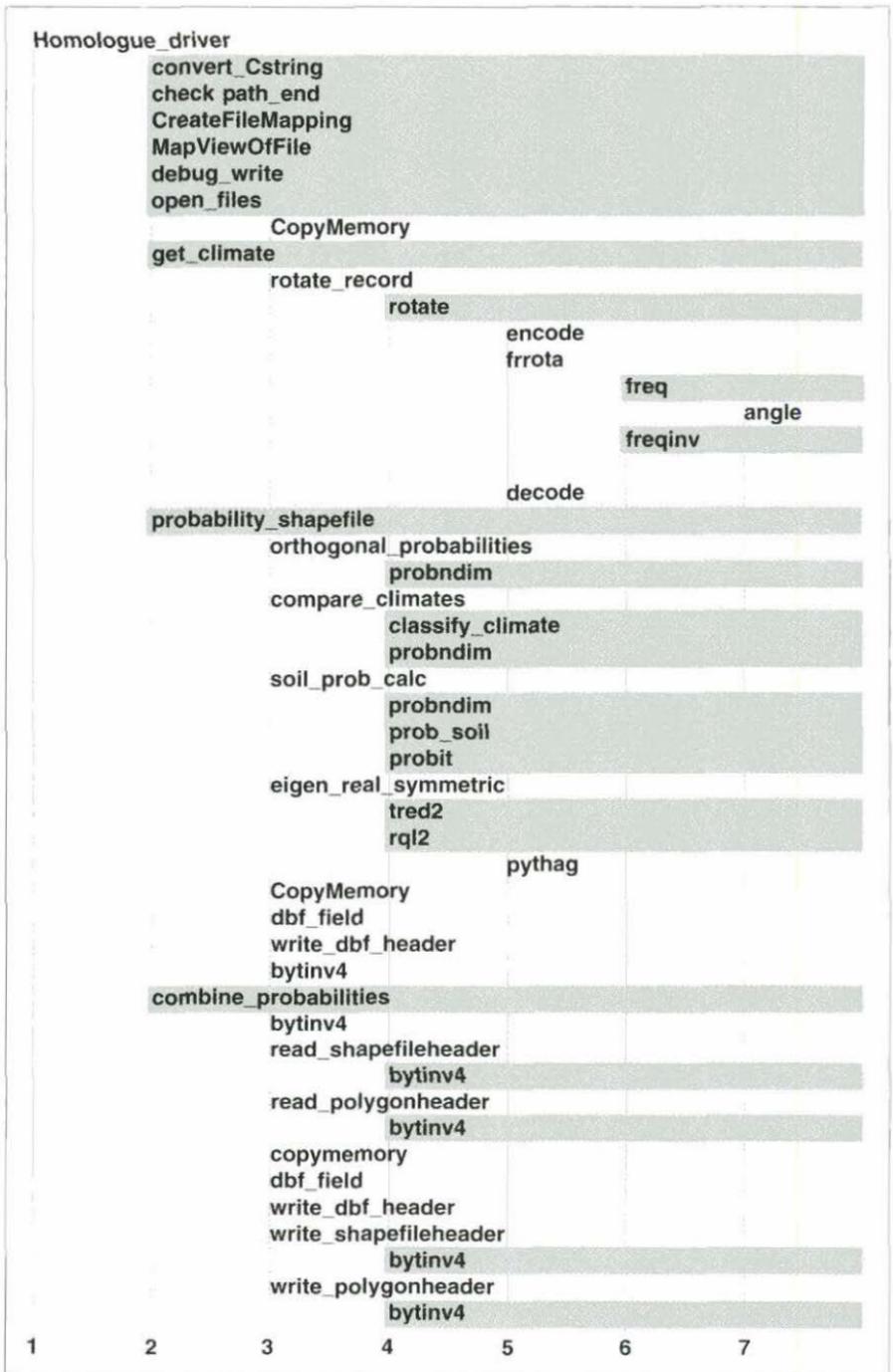
**Calling parameters**

unit	Logical file number, integer, input
head	Header information, Polygon_Header structure, output
return_code	Integer, output

**Non-intrinsic routines called**

1. `bytin4` Inverts the byte order in a 4-byte word

**dll module map with calling depth.**



---

**Error codes returned from the dll**

<b>Code</b>	<b>Explanation for display</b>	<b>Action</b>
1	Error opening a climate data file. See log for details. Check data path	T
2	Error reading a climate data file. See log for details.	T
3	Error opening shapefile. Check the path.	N
4	Error Latitude out of bounds. Move to a valid pixel	N
5	Error Longitude out of bounds. Move to a valid pixel	N
6	Climate not available for pixel. Sea, River or Lake. Move to a valid pixel	N
7	Error creating dBase field descriptor	T
8	Error writing dBase file header	T
9	Homologue internal error. Action unacceptable	T
10	Homologue internal error. Unable to open log file.	T
11	Homologue internal error. probndim	T
12	Attempting to use action 2. Data not loaded.	T
13	Attempting to use action 3. Data not loaded.	T
14	Error opening timing file. Check output path.	N
15	Error calculating soil probability.	T
16	Error cannot open combine.ctr	T
17	Error reading combine.ctr	T
18	Error too many probability shapefiles, limit 20. Eliminate some files from list	N
19	Error opening probability shapefile.	T
20	Error reading shapefile header	T
21	Error shapefile header incorrect	T
22	Error creating shared memory	T
23	Error reading polygon header	T
24	Error writing polygon header	T
25	Error writing shapefile header	T
26	Error opening shapefile (map_probability)	T
27	Error Output file exists, choose another name	N

The code is the numeric value of the parameter `return_code`. The explanation for display is the message to be shown in the message box initiated from the Delphi script. Action denotes the action taken by the Delphi script. N indicates "give the user another try", T "terminate the run".

# References

- Batjes, N.H. 1995. World Inventory of Soil Emission Potentials: WISE 2.1 - Database user's manual and coding protocols. Tech. Pap. 26, International Soils Reference and Information Center (ISRIC), Wageningen, NL.
- Batjes, N.H.; Bridges, E.M. 1994. Potential emissions of radiatively active trace gases from soil to atmosphere with special reference to methane: Development of a global database (WISE). *J. Geophys. Res.* 99(D8):16,479-16,489.
- Bowdler, H.J.; Martin, R.S.; Reinsch, C.; Wilkinson, J.H. 1968. The QR and LR algorithms for symmetric matrices. *Numer. Math.* 11:293-306.
- Bunting, A.H. 1987. *Agricultural environments*. CAB International, Oxford, GB.
- ESRI (Environmental Systems Research Institute). 1998. Shapefile technical description. An ESRI White Paper, Redlands, CA, USA. Available in: <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>. Last accessed Sept 2004.
- FAO (Food and Agriculture Organization). 1974. FAO-UNESCO soil map of the world 1:5,000,000. Vol 1, Legend. UNESCO, Paris, FR, and FAO, Rome, IT.
- FAO (Food and Agriculture Organization). 1978. Report on the AgroEcological Zones Project, volume 1. Methodology and results for Africa. World Soil Resources Report 48. UNESCO, Paris, FR, and FAO, Rome, IT.
- FAO (Food and Agriculture Organization). 1995. Digital soil map of the world and derived soil properties (ver. 3.5). Land and Water Digital Media Series 1, Rome, IT.
- Hutchinson, M.F. 1997. ANUSPLIN Version 3.2 Users guide. The Australian National University, Centre for Resource and Environmental Studies, Canberra, AU. 39 p.

- Jones, P.G. 1987. Current availability and deficiencies data relevant to agroecological studies in the geographical area covered in IARCS. *In*: Bunting, A.H. (ed.). *Agricultural environments: Characterization, classification and mapping*. CAB International, GB. p. 69-82.
- Jones, P.G. 1991. The CIAT climate database version 3.41. Machine readable dataset. Centro Internacional de Agricultura Tropical (CIAT), Cali, CO.
- Jones, P.G. 1996. Climate database for Haiti. Machine readable dataset, Centro Internacional de Agricultura Tropical (CIAT), Cali, CO.
- Jones, P.G.; Gladkov, A. 1999. FloraMap: A computer tool for predicting the distribution of plants and other organisms in the wild; version 1, 1999. CD-ROM with 87 p indexed manual. CIAT CD-ROM Series. Centro Internacional de Agricultura Tropical (CIAT), Cali, CO.
- Jones, P.G.; Thornton, P.K. 2005. How crop production will be affected by climate change in Africa and its repercussions on the continent. *In*: International Workshop on Adaptation to Climate Change, Sustainable Livelihoods and Biological Diversity. Centro Agronómico Tropical de Investigación y Enseñanza (CATIE) Turrialba, CR. (In press.)
- Jones, P.G.; Galwey, N.; Beebe, S.E.; Tohme, J. 1997. The use of *geographical information systems* in biodiversity exploration and conservation. *Biodivers. Conserv.* 6:947-958.
- NOAA (National Oceanographic and Atmospheric Administration). 1984. TGPO006 D. Computer compatible tape. Boulder, CO.

# Index

- A**  
about box, 29  
adaptation range, 31, 51  
    *Bemisia tabaci*, 55  
    coffee, 14  
Africa  
    coffee areas, 20  
    coverage directory, 25  
    data set, 45  
algorithm, 28, 60, 61, 63  
    FloraMap, 1, 45  
    Fourier transform, 47, 81  
    interpolation, 45  
    probability, **48–54**  
    rotate, 80  
alternative grid, 11  
alternative limit control, 19, 32  
analysis  
    climate, 31  
    principal components, **49–51**  
    probability, **14–17**  
    soil carbon, 19  
Asia data set, 45
- B**  
background color, 43  
    change, 41  
    set, 41  
*Bemisia tabaci*  
    adaptation range, 55  
building list  
    control, 35
- C**  
calculate  
    climate model parameter, 56  
    coordinates, 11  
    orthogonal score, 61  
    probability, **51–54**, 58, 77  
    probability coverage, 56  
    radial distance, 61  
    soil probability, **59–61**  
    SSCP, 50  
    variance, 56, 66  
calibration set, 2, 51, 54  
Cartesian coordinates, 36  
cation exchange capacity. *See*  
    CEC  
CEC, 10, 59  
    range, 17  
characteristics  
    soil, 15, 45, 59, 61  
        correlation, 21  
        correlation matrix, 60  
        define, 16  
        range, 17  
        select, 18  
check  
    coordinates, 8  
    distribution, 23  
classification  
    climate, 54, 55, 56  
clear  
    map, 20, 38  
    soil specifications, 15  
    window, 33  
climate  
    analysis, 31  
    classification, 54, 55, 56  
    comparison, 46  
    data, 49  
        file, 17  
    date standardization  
        rotation, **46–48**  
    diagram tool, **35–36**  
    grid, 2  
    model, 31, 61  
    model parameter  
        calculate, 56  
    pixel, 57  
    probability, 23  
        combine, **61–62**

- distribution, 45
- model, 17
- standardize across species, 54
- surface, **45–46**
- type
  - similar, 54
  - variables, 50
  - variate, 45
    - matrix, 50
- closed polygon layer, 38
- display, 38
- Cloud, 2, **61–62**
  - display, 35
  - form, 26
  - operation, 35
  - select, **34**
- coffee
  - adaptation range, 14
  - Africa, 20
  - Colombia, 6, **8–27**
  - environment, 25
  - soil, 18
- Colombia
  - coffee, 6, **8–27**
- color, 23, 39
  - background, 43
    - change, 41
  - change, 5, 7
  - chart, 6
  - choose, 23
  - coverage icon, 5
  - custom
    - define, 39
  - map, 29
  - map background
    - set, 41
  - map layer
    - set, 39
  - render, 40
  - select, 39
  - selection menu, 39
- combine
  - probability shapefile, 34
  - soil and climate probabilities, **61–62**
- communication
  - cross language, 69
  - compare climates, 46
- configuration
  - settings, 29
  - tool, **41–44**, 72
- construct
  - list, 35
  - map, 15, 42
  - probability integral, 58
- control
  - alternative limit, 19
  - building list, 35
  - file, 74
  - layer, 40, 44
    - tool, 5, 29, **37–41**
    - window, 7
  - map window, 29
  - standard Windows®, 29
  - variance, 54
  - window
    - icon, 29
- coordinates, 8, 11
  - calculate, 11
  - Cartesian, 36
  - check, 8
  - geographic, 28, 38, 49
  - match, 28
  - move, 10
  - polar, 36, 47
  - window, 11
- correlation
  - cross, 60
  - matrix, 50, 60
    - soil characteristics, 60
  - soil characteristics, 21
- coverage
  - color icon, 5
  - directory, 3, 5, 29, 39, 42
    - Africa, 25
  - list, 43
  - map, 3
  - probability
    - calculate, 56
    - file, 14
- criteria
  - choose, 23
- cross correlation, 60
- cross language communication, 69

custom color  
define, 39

## D

data  
climate, 49  
directory, 42  
field, 40  
file, 3  
climate, 17  
load, 69  
path  
default, 72  
quantitative, 58  
rainfall, 46  
sample, 17  
set  
Africa, 45  
Asia, 45  
Latin America, 45  
structure, **63-69**  
temperature, 46  
database, 19  
WISE, 58, 59  
date standardization, **46-48**  
dbf file, 17, 39, 41  
default  
data path, 72  
map, 44  
map layers window, 43  
name, 35  
define  
custom color, 39  
soil characteristics, 16  
directory  
choose, 42  
coverage, 3, 5, 29, 39, 42  
Africa, 25  
data, 42  
move, 42  
output, 42  
path, 42  
change, 31  
display  
closed polygon layer, 38  
Cloud, 35  
layer, 38

map, 37, 69  
probability legend, 41  
rotated, 36  
standard, 36  
distribution  
check, 23  
dll, 63, 69  
module map, 89

## E

eigenvalue, 50, 54, 60, 75  
orthogonal score, 61  
regression coefficient, 55  
eigenvector, 50, 54, 60, 75  
orthogonal score, 61  
template, 56  
elevation, 30  
model, 45  
station, 46  
window, 10  
enter map layer, 43  
erase  
all map layers, 38  
memory, 17  
selected map layer  
icon, 38  
error code, **89-90**  
ESRI  
format, 69  
shapefile  
format, 40  
extend window, 31  
eye icon, 15

## F

FAO soil map, 20, 57  
field  
data, 40  
label, 40  
LSBL, 63  
MSBL, 63  
name, 7  
value, 39  
file  
control, 74  
data, 3  
climate, 17

- dbf, 17, 39, 41
- grid, 62
  - climate, 3
  - probability, 23
  - probability coverage, 14
- filename
  - change, 31
  - output, 33
    - choose, 14
- FloraMap, 1
  - algorithm, 1, 45
  - method, 45
  - probability algorithm, **48–54**
  - source code, 63
- format
  - ESRI, 69
  - ESRI shapefile, 40
  - probability shapefile, 40
- Fourier transform
  - 12-point, 47, 80
  - algorithm, 47, 81
- G**
- geographic coordinates, 28, 38, 49
- graded probability, 39
- grid
  - alternative, 11
  - change size, 71
  - climate, 2
  - file, 62
  - probability, 69, 73
  - reference point, 28
  - structure, 71
- I**
- icon
  - coverage color, 5
  - erase
    - all map layers, 38
    - selected map layer, 38
  - eye, 15
  - layer control tool, 5
  - load layer, 5, 6
  - move map layer, 38
  - pan, 7
  - R, 33
  - recent analyses, 16
  - service
    - main menu, **29–44**
    - map window, 29
    - set layer properties, 7
    - site specification, 10
    - window control, 29
    - zoom, 7
- information
  - soil, 57
  - tool, 11, **41**
- installation, 3, 42
- integral
  - probability, 52, 53, 59
    - construct, 58
- integration, 51, 81
- interpolation, 45
- L**
- label field, 40
- Latin America
  - data set, 45
  - map, 5
- layer
  - characteristic
    - set, 40
  - closed polygon, 38
  - display, 38
  - control, 40, 44
    - tool, 5, 29, **37–41**
    - window, 7
  - display, 38
  - load
    - icon, 6
  - map, 38
    - enter, 43
    - erase, 38
    - move icon, 38
    - probability, 39
    - set color, 39
  - set properties
    - icon, 7
  - stack, 38
  - visibility
    - set, 41
- list
  - construct, 35
  - coverage, 43

- load
  - data, 69
  - layer icon, 5, 6
  - map layer, 38
  - shapefile, 38
- LSBL, 76
  - field, 63
- M**
- map
  - arm, **5**
  - clear, 20, 38
  - color, 29
  - construct, 15, 42
  - coverage, 3
  - default, 44
  - default layers
    - window, 43
  - display, 37, 69
  - features, 29
  - Latin America, 5
  - layer, 38
    - enter, 43
    - erase, 38
    - move icon, 38
    - probability, 39
    - set color, 39
  - load layer, 38
  - module
    - dll, 89
  - probability
    - combine, **61-62**
    - produce, 20
    - projection, 11
    - set background color, 41
  - soil
    - FAO, 20, 57
  - window, **28-29**
    - control, 29
    - menu bar, 29
    - service icon, 29
- MapObjects, 40
- match coordinates, 28
- matrix
  - climate variate, 50
  - correlation, 50, 60
    - soil characteristics, 60
  - notation, 50
  - SSCP, 50
    - variance-covariance, 50
- memory
  - erase, 17
- menu
  - bar
    - map window, 29
  - color selection, 39
  - main
    - service icon, **29-44**
  - right click, 29
- method
  - FloraMap, 45
  - threshold, 21
- minimum probability, **31**
- model
  - climate, 31, 61
  - generic, 56
  - normal elevation, 45
  - parameter
    - calculate, 56
  - probability
    - climate, 17
    - soil, **56-61**
- move
  - coordinates, 10
  - directory, 42
  - map layer
    - icon, 38
- MSBL, 76
  - field, 63
- N**
- name
  - default, 35
  - field, 7
- normal elevation model, 45
- O**
- operation
  - Cloud, 35
  - pan, 37
  - software, **69-90**
  - zoom, 37
- organic matter percentage, 19
- orthogonal score

- calculate, 61
- eigenvalue, 61
- eigenvector, 61
- output
  - directory, 42
  - filename, 33
    - choose, 14
  - probability shapefile
    - store, 42
- P**
- pan
  - icon, 7
  - operation, 37
- PCA, **49–51**
- pH, 60, 61
  - range, 19
- pixel, 62
  - climate, 57
  - size, 45
- polar coordinates, 36, 47
- polygon layer
  - closed, 38
- principal components analysis.
  - See PCA
- probability
  - algorithm
    - FloraMap, **48–54**
  - analysis, **14–17**
  - calculate, 51, **54**, 58, 77
  - climate, 23
  - coverage
    - calculate, 56
    - file, 14
  - density function, 51
  - distribution
    - climate, 45
  - file, 23
  - graded, 39
  - grid, 69, 73
  - integral, 52, 53, 59
    - construct, 58
  - legend
    - display, 41
  - map layer, 39
  - minimum, **31**
  - model

- climate, 17
  - soil, **56–61**
- orthogonal, 61
- select type, 32
- shapefile, 26
  - combine, 34
  - format, 40
- soil
  - calculate, **59–61**
  - soil and climate
    - combine, **61–62**
- produce map, 20
- program file, 3
- Q**
- quantitative data, 58
- R**
- R icon, 33
- radial distance
  - calculate, 61
- rainfall, 36, 45
  - data, 46
- range
  - CEC, 17
  - of adaptation, 31
  - pH, 19
  - soil characteristics, 17
- recall screen, 34
- recent analyses
  - icon, 16
  - tool, 31
- reference point grid, 28
- regression coefficient
  - eigenvalue, 55
  - estimated, 54
- render color, 40
- right click menu, 29
- rotate algorithm, 80
- rotated display, 36
- rotation
  - climate
    - date standardization, **46–48**
- S**
- sample data, 17
- scatterplot, 49

- screen
    - recall, 34
  - select
    - Cloud, **34**
    - color, 39
    - probability type, 32
    - shapefile, 34
    - soil characteristics, 18, 32
  - set
    - layer
      - characteristic, 40
      - properties icon, 7
      - visibility, 41
    - map
      - background color, 41
      - layer color, 39
    - threshold, 32
  - settings configuration, 29
  - shapefile, 39
    - choose, 26
    - ESRI, 38
      - format, 40
    - load, 38
    - probability, 26
      - combine, 34
      - format, 40
    - range, 29
    - select, 34
  - site specification
    - icon, 10
    - tool, 10, 14, **30**
    - window, 15
  - size
    - pixel, 45
  - software
    - map projection, 11
    - operation, **69–90**
    - structure, **69–90**
  - soil
    - carbon, 19
    - characteristics, 15, 17, 45, 59, 61
      - choose, 32
      - correlation, 21
      - correlation matrix, 60
      - define, 16
      - range, 17
      - select, 18
    - characteristics selector, 32
    - coffee, 18
    - constraints, 21
    - definition area, 18
    - information, 57
    - map
      - FAO, 20, 57
    - probability
      - calculate, **59–61**
      - combine, **61–62**
    - probability model, **56–61**
    - specifications
      - clear, 15
      - tool, 17
      - window, **27, 31**
  - species type
    - choose, 31
  - SSCP
    - calculate, 50
    - matrix, 50
  - stack layer, 38
  - standard display, 36
  - standardize climate
    - across species, 54
  - station elevation, 46
  - store output probability shapefile, 42
  - structure
    - data, **63–69**
    - grid, 71
    - software, **69–90**
  - sums of squares and cross products. *See* SSCP
  - surface
    - climate, **45–46**
- T**
- temperature, 36, 45, 56
    - data, 46
  - template
    - eigenvector, 56
  - threshold
    - set, 32
    - value, 32
  - tool
    - climate diagram, **35–36**

- configuration, **41–44**, 72
- information, 11, **41**
- layer control, 29, **37–41**
  - icon, 5
- recent analyses, 31
- site specification, 14, **30**
- soil specifications, 17
- zoom, **37**

## **U**

- user interface, 63

## **V**

- value field, 39
- variables
  - climate, 50
- variance
  - calculate, 56, 66
  - control, 54
  - estimates, 56
- variance-covariance matrix, 50
- variate
  - climate, 45
  - matrix, 50

## **W**

- window
  - clear, 33
  - control icon, 29
  - coordinates, 11
  - default map layers, 43
  - elevation, 10
  - extend, 31
  - layer control, 7
  - map, **28–29**
    - menu bar, 29
  - site specification, 15
  - soil specifications, **27**, 31
  - threshold value, 22
- Windows® standard control, 29
- WISE
  - database, 59
- WISE database, 58

## **Z**

- zoom, 6, 29
  - boundary, 6
  - icon, 7
  - operation, 37
  - tool, **37**

### **LICENSE AGREEMENT**

READ THE TERMS AND CONDITIONS OF THIS LICENSE AGREEMENT CAREFULLY BEFORE OPENING THE PACKAGE CONTAINING THE PROGRAM DISKETTES. BY OPENING THE PACKAGE CONTAINING THE PROGRAM, YOU ARE ACCEPTING AND AGREEING TO THE TERMS OF THIS LICENSE AGREEMENT. IF YOU ARE NOT WILLING TO BE BOUND BY THE TERMS OF THIS LICENSE AGREEMENT, YOU SHOULD PROMPTLY RETURN THE PACKAGE IN UNOPENED FORM, AND YOU WILL RECEIVE A REFUND OF YOUR MONEY.

### **LICENSE GRANT**

The enclosed software is a proprietary product of CIAT, and protected under US copyright law. The software may be used only on computers owned, leased, or otherwise controlled by you. Neither concurrent use on two or more computers nor use in a local area network or other network is permitted without separate authorization and the payment of other license fees. You agree that you will not assign, sublicense, transfer, pledge, lease, rent, or share your rights under this License Agreement. You agree that you may not reverse assemble, reverse compile, or otherwise translate the software.

Upon loading the software into your computer, you may retain the program CD-ROM for backup purposes. In addition, you may make one copy of the software for the purpose of backup in the event the program CD-ROM is damaged or destroyed. You may not copy the user manual or parts of the user manual. Except as authorized under this paragraph, no copies of the software or the user manual may be made by you or any person under your authority or control.

### **LIMITED WARRANTY**

CIAT MAKES NO WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. ANY REMEDY UNDER THIS WARRANTY IS LIMITED TO RETURN OF THE SOFTWARE AND USER MANUAL TO THE DEALER OR TO CIAT FOR REPLACEMENT OR REFUND. THIS WARRANTY IS VALID FOR 30 DAYS FROM YOUR DATE OF PURCHASE. CIAT EXCLUDES ANY WARRANTY COVERAGE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES. SOME STATES DO NOT ALLOW LIMITATIONS ON IMPLIED WARRANTIES, SO SOME OF THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.