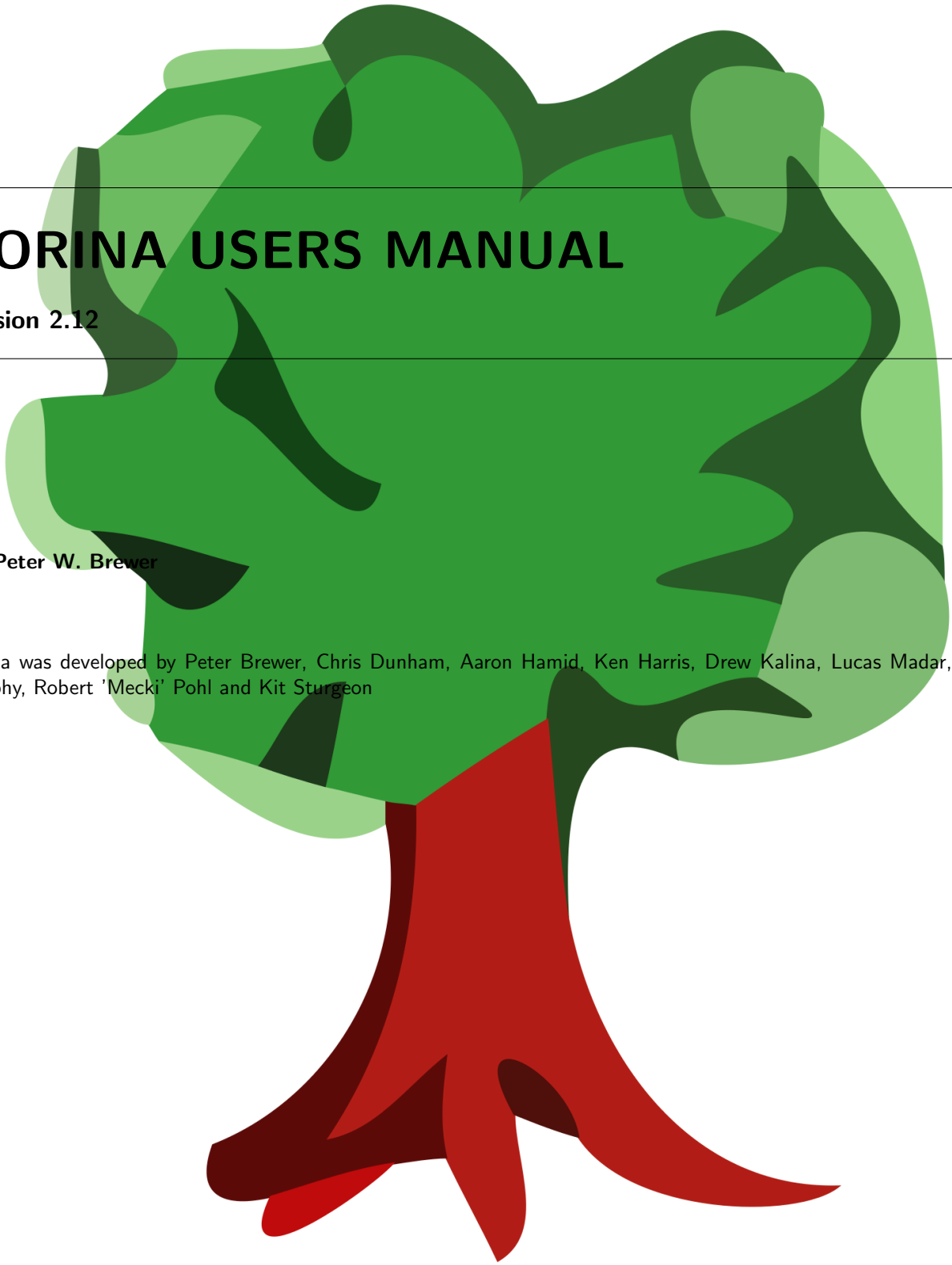
_

CORINA USERS MANUAL

Version 2.12



BY Peter W. Brewer

Corina was developed by Peter Brewer, Chris Dunham, Aaron Hamid, Ken Harris, Drew Kalina, Lucas Madar, Daniel Murphy, Robert 'Mecki' Pohl and Kit Sturgeon

-

_

CORINA USERS MANUAL

Version 2.12

Contents

Part I

User Guide

Chapter 1

Installation

Corina is made up of two packages; the Corina desktop application and the Corina database server. Corina was designed primarily for laboratories with multiple users, each running the Corina desktop application on their own computer connecting to a single central server containing the lab's data. In this situation the Corina server would be run on a separate computer to those running the desktop client, but this need not necessarily be the case. It is perfectly possible to run both the server and the client on the same computer. This is likely to be the situation if you simply want to try out Corina, if you don't have a separate server, or if you do not work in a multi-user laboratory.

1.1 Desktop application

Installation packages for the Corina desktop application are available for Windows, MacOSX and Ubuntu Linux. Corina can also be run on other operating systems as long as they support Java 6.

To install Corina, download the installation file for your operating system from http://dendro.cornell.edu/corina/download.php. The website should provide you with a link to the installer for your current operating system:

- **Windows** – Run the setup.exe and follow the instructions. If you do not have Java installed the installer will direct you to the Java website where you can get the latest version. Once installed, Corina can be launched via the Start menu.
- **Mac OS X** – As mentioned above, Corina requires Java 6. Although MacOSX ships with Java installed, unfortunately Apple have been very slow to provide Java 6. Although it was released in 2006, it was not until August 2009 that Apple made Java 6 available as part of v10.6 (snow leopard). Corina can therefore only be run on Snow Leopard or later. To do so, download the dmg disk image file and mount it by double clicking on it. Drag the Corina.app into your applications folder and copy the manual and license files to somewhere convenient in your documents folder. For the more adventurous there is the possibility that you could run Corina using SoyLatte instead of the standard Java installation that comes with the operating system. This could be a possible method for running Corina even on earlier versions of MacOSX but is unsupported and largely untested.
- **Ubuntu Linux** – A deb file is available which was designed for use on Ubuntu distributions but should work on any Debian based system. Install using your favorite package management system e.g. `sudo dpkg --install corina\_2.xx-1\_all`. On Ubuntu and similar distributions, the package should add a Corina shortcut to your applications menu. Alternatively you can start Corina from the command line by typing corina.
- **Other operating systems** – Make sure you have Java 6 installed, then download the Corina jar file to your hard disk. You can run Corina from the command line by typing: `java -jar corina.jar`.

1.1.1 Mapping support

Corina includes 3D mapping for visualization of sampling locations. Although this is not necessary for most tasks, to make use of the mapping functions you will require a OpenGL 3D capable graphics card. To check whether your computer

already supports 3D mapping, open Corina, go to Admin, then Site map. Corina will warn you if your graphics card is not supported.

All MacOSX computers should automatically support OpenGL. Most Windows and Linux computers made since 2006 should also support OpenGL, however, this does require proper drivers to be installed. In some cases Windows computers may include a compatible graphics card, but may only have the default Windows video drivers installed. If you are having trouble with the mapping in Corina make sure you have installed the most recent drivers for your graphics card. Linux users may be required to install proprietary graphics drivers.

The mapping component of Corina makes use of NASA's open source World Wind Java. NASA's website http://worldwind.arc.nasa.gov/ contains further information and instructions that you may find helpful if you are having problems getting the mapping to work.

1.2 Server installation

For the Corina desktop application to be useful you will also require access to a Corina server. If you are running Corina in a lab where the Corina server has already been set up by your systems administrator, you can skip this section.

The Corina server is made up of a number of components, which unlike the desktop client, can not be easily combined together into cross-platform packages. Although all the constituent components are open-source and available for all major platforms, building and maintaining separate packages for each platform is too large a task for a small development team. To conserve resources, we therefore made the decision to utilize Virtual Machine technology to ensure that the Corina server could still be run on all major operating systems. This means that we can package the Corina server for a single operating system (Ubuntu Linux) and then distribute it as a Virtual Appliance that can be run as a program on your normal operating system.

The Corina server is therefore available via two main methods. The first is as a VirtualBox Virtual Appliance which can be run on any major operating system, the second is as an Ubuntu package for running natively on a Linux server. The source code for the server is also available so it is perfectly possible for more experienced users to set up the Corina server to run natively on other platforms. For this you will require knowledge of Apache 2, PHP and PostgreSQL.

1.2.1 Virtual Appliance - all operating systems

To run the Corina server Virtual Appliance, you will first need to download and install VirtualBox from http://www.virtualbox.org. Installation packages are available for Windows, MacOSX, OpenSolaris and many Linux distributions.

Once you have VirtualBox installed, you will then need to download the Corina server from the Cornell website http://dendro.cornell.edu/corina. This package contains a bare-bones Ubuntu Linux server with everything required to run the Corina server installed and ready to use. As VirtualBox, the entire Ubuntu operating system and Corina server components are all open source there are no license fees to pay.

Open VirtualBox and go to File, Import Appliance, then follow the wizard selecting the Corina server appliance file when prompted. Once installed you can run your server by highlighting it in the list and pressing start. The server will boot up in a window alongside your normal operating system and eventually reach a login prompt. To save on download size and disk space only the essential packages to make the server run have been installed. This means there is no graphical interface just a command line. Hopefully this should not be a problem as once set up, the only interaction needed with the Virtual Appliance will be through the normal Corina desktop application. If you would prefer to use a graphical interface to the server this can be easily installed. See chapter 3 for further details.latex define environment

Before you can use your server you will need to know the IP address that the server has been assigned by your network. To do this login at the prompt with the default admin credentials: user – corina; password – w3l0v3tr33s. Once logged in, type `corina --test` and a basic configuration test will be performed. If all is well, all tests will be passed and it will tell you the URL of your new server. You will need to set your Corina client to point at this webservice to use your server.

1.2.2 Ubuntu native installation

If you are fortunate enough to be running Ubuntu then the native Ubuntu deb package is the best and easiest method for installing the Corina server, otherwise see section **??** to install the server as a Virtual Appliance.

To install the Corina server in Ubuntu simply download the deb package from the Cornell server `http://dendro.cornell.edu/corina` and install with your favourite package manager. For instance, to install from the command line simply type `sudo dpkg --install corina-server.deb`. The package will automatically run a configuration script to assist with creating a database user, building the Corina PostgreSQL database, setting database permissions and setting up the Apache webservice. The configuration ends with a test routine to check all services are set up correctly and if so, will provide you with the URL of the newly configured Corina webservice.

1.2.3 Advanced install on other operating systems

As mentioned previously, the limited resources available for Corina development means that we have been unable to produce native installers for platforms other that Ubuntu. If you are an experience systems administrator though, it should not be too difficult to set up the Corina server manually.

The Corina server is essentially a PostgreSQL database accessed via a PHP webservice running on Apache 2. The following dependencies are therefore required: postgresql-8.4; postgis; postgresql-contrib-8.4; postgresql-8.4-pljava; sun-java6-jre; apache2; php5; php5-pgsql; php5-curl; php5-mhash.

The basic procedure for installation is as follows:

- Install all dependencies
- Create PostgreSQL database from Corina template SQL file
- Set up a database user and provide access to the server in the pg_hba.conf file
- Give this user read and write permissions to the database
- Copy the webservice code into a web accessible folder
- Set up Apache to see this folder by creating an entry in the sites-enabled folder
- Restart PostgreSQL and Apache and check you can access the webservice from a web browser

Chapter 2

Getting started

Corina is the tree ring measuring and analysis program developed at the Cornell Tree-Ring Laboratory. It is focused primarily on the measurement of tree ring widths and the organization and curation of the data, metadata and physical samples. It is cross-platform (running on all Java 6 enabled operating systems including Windows, MacOSX and Linux) and open-source. It includes support for Velmex, Lintab and Bannister measuring platforms.

Corina has been developed since 2000 as a desktop Java application. Earlier iterations (version 0.x and 1.x) were built around a standard file-based data management system. In 2007, work began on a major rewrite of the software whereby this file-based data management was replaced with a true relational database and server/client webservice infrastructure. This series of releases (versions 2.x) are what are described in this manual.

Chapter 3

Running the server

For basic day-to-day running of the Corina server, you simply need to make sure that the server is running. All other interaction and managment (creating users, granting permissions, accessing data) is done through the Corina desktop application. This section, however, outlines a number of aspects of the server that advanced users may find useful.

3.1 Extending the Virtual Appliance

For those of you that are unfamiliar with Linux, the basic command line prompt is not likely to be very comfortable. If you are interesting in looking at the server in more detail you may therefore prefer to install a full graphical interface. Unlike Windows, there are a number of different graphical interfaces (or desktops) to choose from in Linux, the most popular being Gnome and KDE. To install one (or both) of these you need to type `sudo apt-get install ubuntu-desktop` (for Gnome) or `sudo apt-get install kubuntu-desktop` for KDE. Windows users that are new to Linux may find KDE more familiar than Gnome.

3.2 Security

The basic installation of the Corina server includes the standard configuration for Apache, PHP and PostgreSQL. Although these products are considered secure by default, there are a number of measures that can be taken to make them more so. If your server is only accessible within your local intranet (e.g. behind a robust firewall) then you may not feel it necessary to modify the standard setup. Precautions may be deemed more important if you server is accessible from the internet. In this case it would be wise to contact your local network administrator for further information.

3.2.1 Usernames and passwords

There are a number of default usernames and passwords setup on your server. If your server is accessible for the internet we strongly advise you to change these defaults and anyone with knowledge of the Corina server could access and compromise your machine.

System user - these are the credentials you use to log in to the command prompt in your Corina Virtual Appliance. By default the user is 'corina' and the password is 'w3l0v3tr33s'. To change this log in to the command prompt and type `passwd` and follow the instructions.

Database user - these are the credentials used by the webservice to read and write to the database.

Corina admin user - these are the admin credentials that you use to log in with in your Corina desktop application. Be default the user is '???' and the password is '???'. To change these open the Corina desktop application, then go to Admin then Change password.

3.2.2 Authentication and encryption

Corina uses a relatively sophisticated method to ensure that unauthorised users cannot access the Corina database through the webservice. It is loosely based around http digest authentication and uses a challenge and response scheme. This makes use of cryptographic hashes (a relatively short digital fingerprint of some data but which cannot be decompiled to retrieve the original data) and nonces (a pseudo-random string used just once). All hashes used in the Corina webservice use the MD5 algorithm. This decision will be periodically reviewed to ensure that MD5 is the most appropriate and secure algorithm to use. Whilst an MD5 hash of a short phrase can be compromised, the length and randomness of the original data means with current cracking techniques this is essentially impossible. For a complete description of Corina's authentication procedure see section 5.1.

The default Corina server setup, however, uses standard HTTP protocol to communicate between the server and the desktop application. This is the same protocol used for the majority of web pages on the internet and a determined hacker could eavesdrop on this communication. Depending on how important and private you perceive your data you may choose to use Secure Socket Layer (SSL) to encrypt this communication. This is the same technology used by websites such as online banking. To make full use of this upgrade in security you will however also require a SSL certificate from an official licensing authority. These certificates typically cost several hundred dollars per year.

3.3 Directly accessing the database

Although the Corina database is designed to only be accessed by the Corina desktop application via the Corina server's webservice, you may decide that you'd like to directly access the database yourself. For instance, you may like to write complicated SQL queries to probe your database in ways not currently supported by the Corina desktop client.

> ⚠ *Any changes made to the database may have drastic consequences. We strongly recommend that you never write changes directly to the database as this can cause loss of data and corrupt future upgrades to Corina.*

3.3.1 PGAdmin3

One of the easiest ways to access the PostgreSQL database is through the application PGAdmin3. This is a cross-platform open source application for communicating with PostgreSQL databases. You can install PGAdmin3 on your desktop computer and access the remotely running database using your database user credentials. By default, PostgreSQL runs on port 5432.

3.3.2 ODBC

It is also possible to connect to your Corina database via an ODBC connection. This allows limited access to the database from a variety of database applications including programs like Microsoft Access for which further details are given here. To use ODBC you will need to install the PostgreSQL ODBC driver (http://www.postgresql.org/ftp/odbc/) on your desktop computer.

Once you've installed the driver you can then open a blank database in Access and go to Files, Get external data then Link tables. In the file dialog box change the file type to ODBC Databases(). Next, select the PostgreSQL Unicode driver, then fill out the server details. You should then be able to open the tables and views from the Corina server database directly from within Access as if they were local tables. Be warned though that Access and ODBC have many limitations compared to PostgreSQL, especially with regards data types. For this reason we *strongly* recommend using this for read only purposes. Using the ODBC connection to write changes to your PostgreSQL database is quite likely to cause serious issues.

3.3.3 PSQL

The final, and most advanced method is to use the psql client on your server. This is a command line client which can be used to interrogate the database. If you're not already familiar with psql it is unlikely that this is a good method for you to use!

Chapter 4

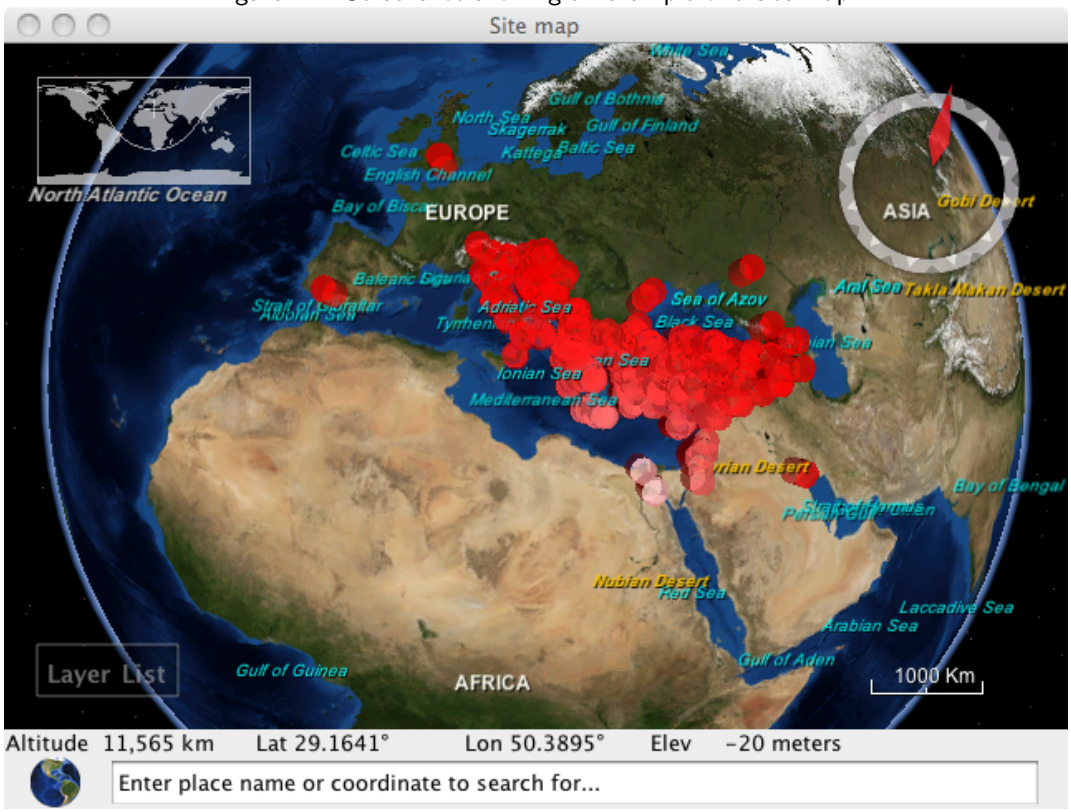
Mapping

Corina includes an integrated open source 3D mapping system (based on NASA's award winning World Wind Java SDK) similar to the program Google Earth which you're no doubt familiar with. As mentioned in the installation chapter, this mapping system requires an OpenGL 3D capable graphics card.

Before you can use the mapping in Corina, it must have something to map! See the chapter on Metadata (page **??**) for information about adding coordinates to your system.

There are two ways to map data from your database. First of all, you can see a map of all the sites (i.e. TRiDaS objects) by going to Administration, then Site map. This will give you a screen like this:

Figure 4.1: Screenshot showing an example of a site map.



You can also see a map of your current series if you have latitude/longitude metadata by clicking on the map tab on the main data screen.

4.1 Navigation

You can navigate around your maps using both your mouse and keyboard.

4.1.1 Mouse with scroll wheel

Pan Left mouse button click and drag – all directions

Zoom Use the scroll wheel on the mouse or Left and Right mouse (both buttons) click and drag up and down

Tilt Right mouse button click and drag – up and down or use 'Page Up' and 'Page Down' on the keyboard.

Rotate Right mouse button click and drag – left and right Note: Crossing the top and bottom half of the screen while rotating will change direction.

Stop Spacebar

Reset Heading N

Reset all R

4.1.2 Single button mouse

Pan Left mouse button click and drag - all directions. L left mouse button click once to center view.

Zoom Hold 'Ctrl' on the keyboard and Left mouse button click and drag - up and down

Tilt Hold 'Shift' on the keyboard and Left mouse button click and drag - up and down or use "Page Up" and "Page Down" on the keyboard.

Rotate Hold 'Shift' on the keyboard and Left mouse button click and drag - left and right

Stop Spacebar

Reset Heading N

Reset all R

These controls enable you to explore your location information in 3D such as the example of Mount Vesuvius in figure 4.3.

Another method of navigating around the map is by using the built in gazetteer. You can enter and place name or coordinate information into the box at the bottom of the screen and you will fly to the requested location.

4.2 Interacting with data

Each marker on the map represents either a TRiDaS object or element in your Corina database. By clicking on these pins you can get more information from the database (see figure **??**).

The example above shows the ring marker is of a site in Napoli called Poggiomarino (code name POG). You can see the option for searching for all series in the database associated with this site, and also the option for viewing all the metadata.

4.3 Background map layers

Corina comes ready configured with basic map layers, including high resolution satellite imagery and basic political features. You can turn background layers on and off by going to View ¿ Layers.

Figure 4.2: Example of 3D mapping in Corina.

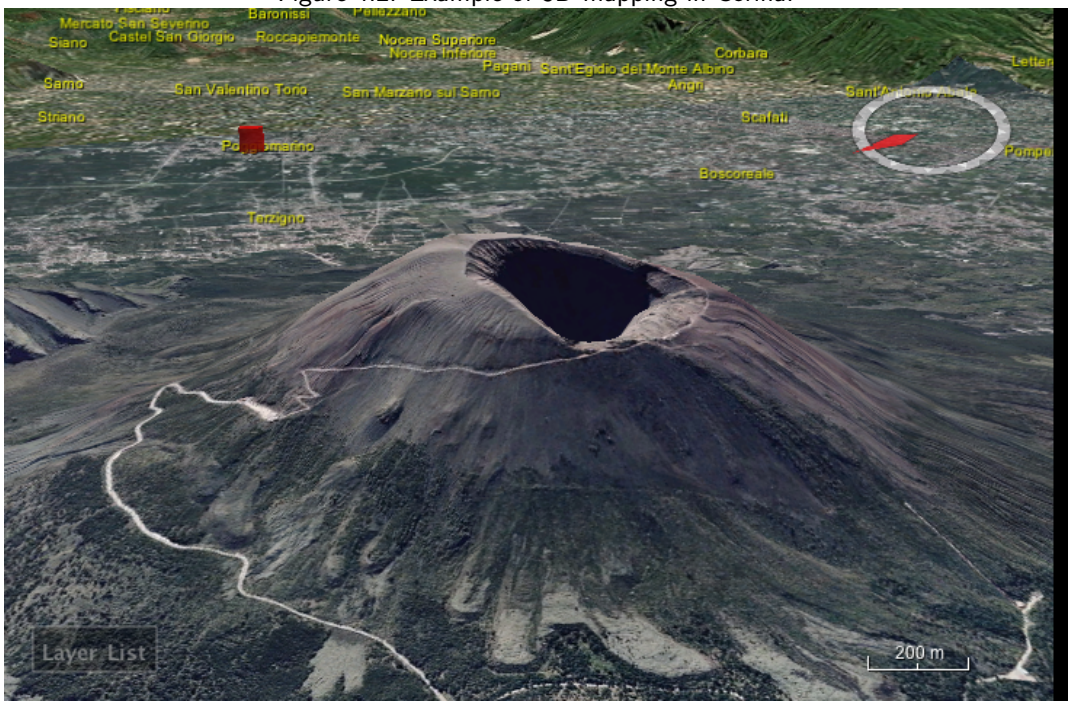
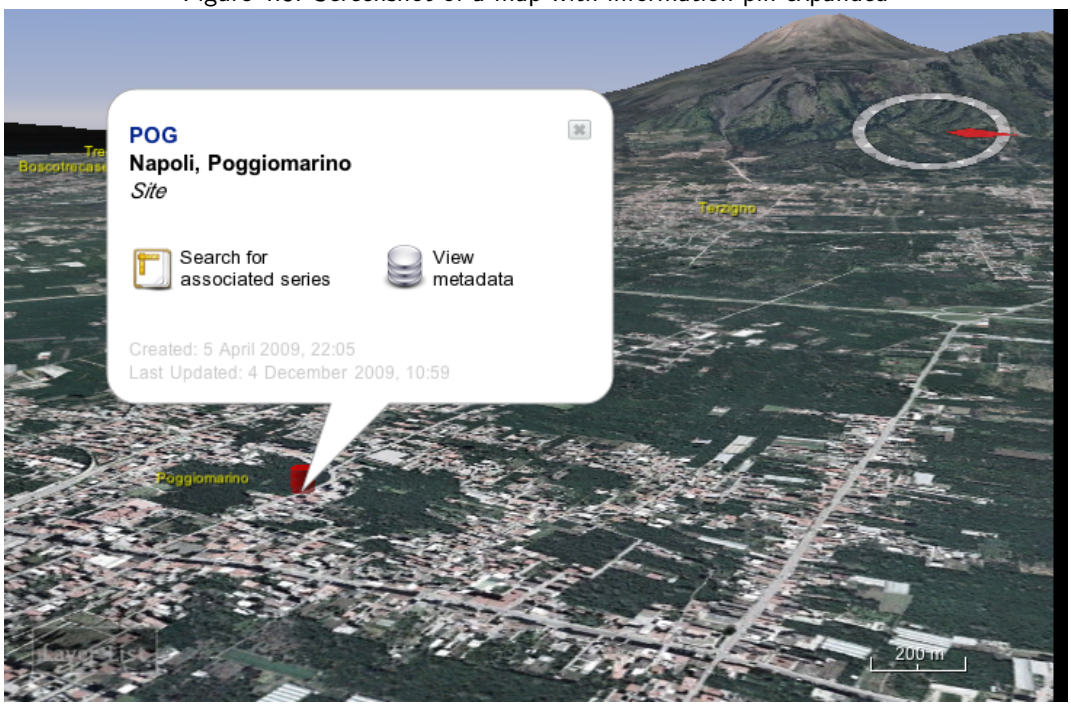


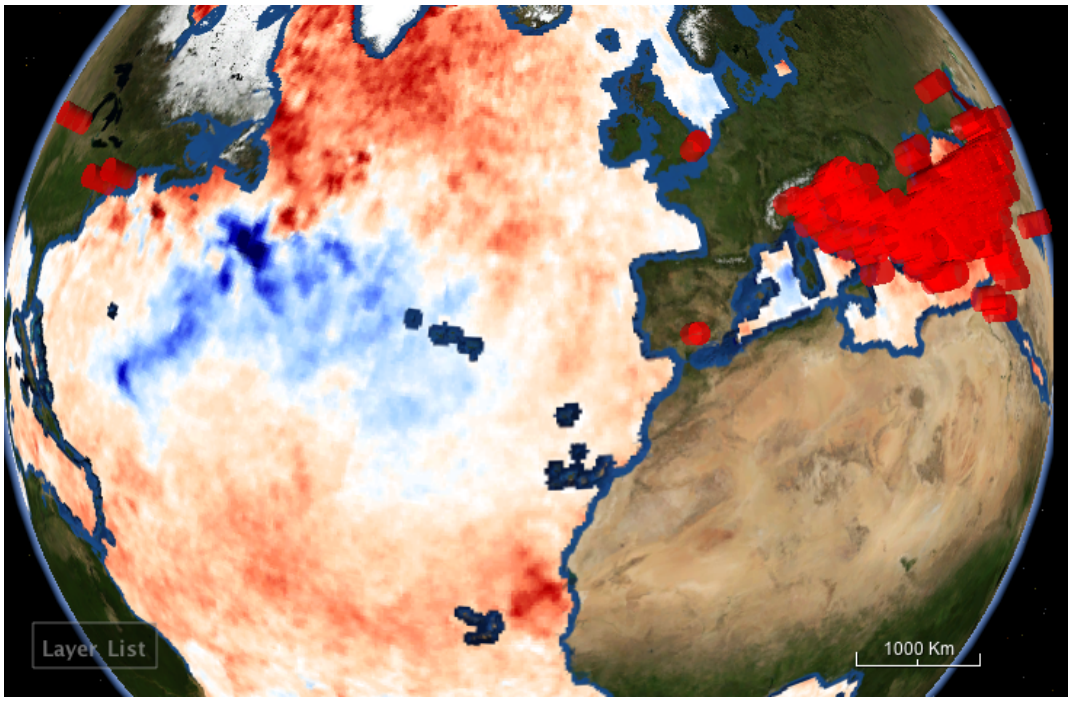
Figure 4.3: Screenshot of a map with information pin expanded



Map layers are downloaded on-the-fly so there is likely to be a delay when you initially visit to a new region. However, up to 2Gb of map data can be cache locally to your hard disk, so on future visits, maps should load quickly.

The mapping system in Corina includes support for remote map servers that use the OGC Web Mapping Service (WMS) standard. If you go to View ¿ Layers ¿ Add remote layers you will get a dialog with a tab for each WMS server configured for your system. By default this includes the NASA Earth Observation and Jet Propulsion Lab servers, but your Corina administrator can add others. By ticking layers in this list you can add data layers to your map. Your system administrator may host a map server specifically for your lab, for instance, containing high resolution plans of an archaeological site that you are working on. Figure **??** shows an example overlay of sea surfaces temperatures loaded dynamically from the NASA EO server.

Figure 4.4: Map screenshot with a NASA sea surface temperature overlay dynamically loaded from the NASA WMS server.



4.4 Data map layers

Data map layers are controlled with the layer list in the bottom left of the screen. When viewing series, you will have the option of adding layers containing points for all the other series at the current site, and showing all the sites in the database.

4.5 Exporting maps

You can export maps by going to File ¿ Export map as image. For best results, maximize your map window first. You may also like to turn off various map widgets by going to the View menu. The exported image will include everything you can see on your map screen.

Part II

Developers guide

Chapter 5

Systems architecture

5.1 Authentication design