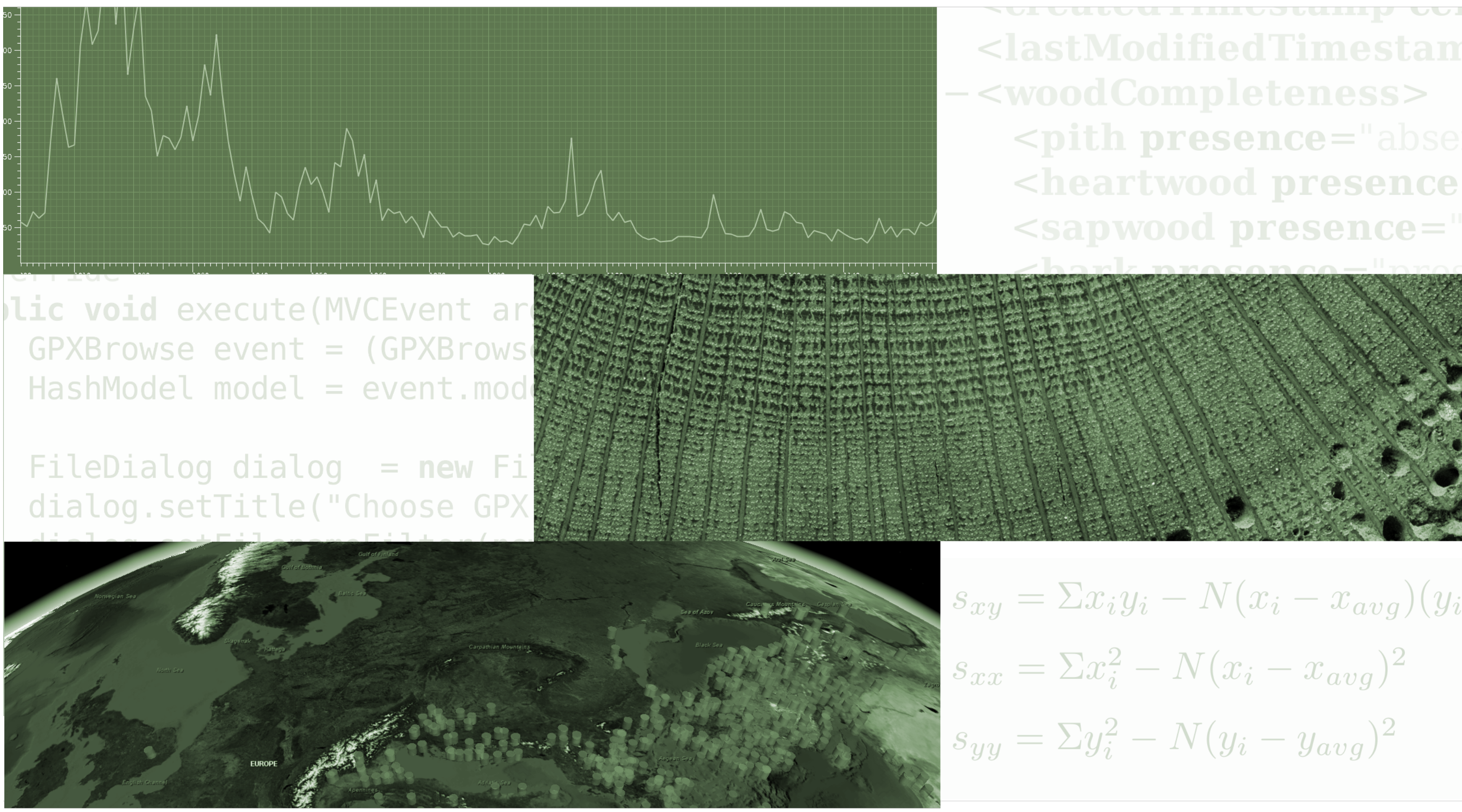
Corina

A guide for users and developers



By Peter W. Brewer and Ken Harris

For Corina server and desktop
version 2.12

GPL v3 Free Software — Free as in Freedom

Malcolm and Carolyn Wiener Laboratory for Aegean
and Near Eastern Dendrochronology
B48 Goldwin Smith Hall
Cornell University,
Ithaca, New York 14853. USA.

☎ +1 607 255 8650
✉ p.brewer@cornell.edu

CORINA

A guide for users and developers

By Peter W. Brewer and Ken Harris

Contents

Preface

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 Hensen measuring platforms.

Corina has been developed since 2000 as a desktop Java application, following an earlier DOS-based version which itself was derived from a collection of FORTRAN and C utilities. Earlier iterations of Corina (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 an object-relational database management system (ORDBMS) and server/client webservice infrastructure. This series of releases (versions 2.x) are what are described in this manual.

This manual is divided into two main sections, the first for users, the second for developers. Corina is open source software (see the details of the license on pages 55–60) so you are welcome to inspect and edit the code. The second part of this manual will help you do that.

Over the years Corina has been developed by a number of people: Peter Brewer, Chris Dunham, Aaron Hamid, Ken Harris, Drew Kalina, Lucas Madar, Daniel Murphy, Robert 'Mecki' Pohl and Kit Sturgeon. We hope that you find Corina useful and look forward to hearing your feedback.

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.deb
  ```

  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. But to do this you will require a good 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 10 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 1.2.1 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

Measuring samples

Although it is possible to manually enter the ring widths of your samples into Corina, it is normal to automate this process using a measuring platform. Corina supports the most common measuring platforms including Velmex and Lintab.

2.1 Configuring measuring platforms

Measuring platforms typically use serial ports to communicate to computers. In recent years computer manufacturers have been phasing out serial ports so you may need to purchase a serial-USB converter. Modern MacOSX, Linux as well as Windows 7 should support most serial-USB adapters out of the box, otherwise you must install the relevant drivers before continuing. Recent Lintab USB platforms use internal serial-USB converters so are treated in exactly the same way by Corina.

To begin, shut down your computer, attach your platform, then reboot and launch Corina. Next, go to the preferences window and open the hardware tab and you should see an interface that looks like figure 2.1.

In the type pull down menu, select the type of measuring equipment you are using. Note that this refers to the equipment that the computer is attached to, and not necessarily the measuring platform itself. For instance, Velmex platforms are typically connected through a Metronics digital readout device. Included in this list is the EveIO device which is an open-source device designed for the Cornell Tree-Ring Laboratory. Circuit drawings for this device can be obtained from the Cornell lab to enable Hensen measuring platforms to be used with Corina (and other software).

> Standard Lintab platforms use a proprietary communications protocol. Rinntech–the manufacturers of Lintab platforms–claim intellectual property rights over this protocol. During discussions between the Corina development team and Rinntech an agreement was reached whereby the Corina developers agreed not to release details of the protocol. In turn Rinntech has agreed to produce an adapter that can be attached to Lintab platforms so that they communicate with an open ASCII protocol. Users wishing to use Lintab platforms with Corina (or any software not developed by Rinntech) must therefore contact Rinntech and purchase an adapter.

Next you must choose the port that your platform is connected to from the pull down menu. In Windows this will be a COM port, in Linux and Mac this will be a /dev/xxx port. Depending on the type of platform you choose, you may also need to set various communication parameters. If these boxes are enabled, please check the documentation that came with your measuring platform to ensure these values are set correctly.

To check whether your platform is working, click the 'Start measuring test' button and attempt to measure a few rings. Once you are satisfied, click OK on the preferences window and you will be returned to the Corina home screen.

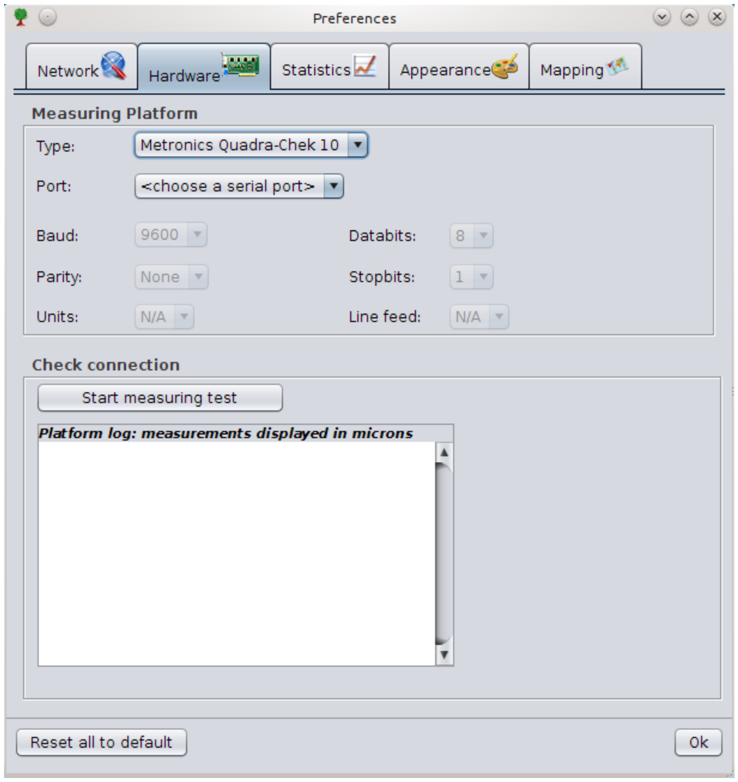


Figure 2.1: The hardware preferences dialog.

2.2 Differences in measuring hardware

Different measuring platforms have different capabilities. For instance, some include a physical switch for firing measurement events, others also include switches for resetting measurements to zero. Some platforms (e.g. Lintab) also continuously report the measurement values to the computer. So depending on the hardware you use, Corina will present the you with slightly different options.

Please note that the architecture of Corina is such that providing support for additional measuring platforms is a relatively simple task. If you have equipment that is not currently supported, please contact the developers to see if it can be included.

2.3 Measuring a sample

Once your measuring platform has been configured, measuring your first sample is simple. To start a new measurement go to *File* ➔ *New* or click the 'new' icon on the home screen. A dialog will appear where you can scan your sample's barcode, or press the button to enter metadata for your sample later. Barcodes minimize data entry errors and also speed up the process of measuring your samples. See section **??** for more information. Once you have scanned your barcode or pressed the button, you will then be presented with an empty Corina data screen (figure 2.2).

The next step is to fill out the metadata tab. If you have used a barcode, nearly all of this metadata will be filled in for you, otherwise you will need to fill this out yourself. Details about metadata can be found in chapter 3, page 13.

To begin measuring your sample you can now go to *Edit* ➔ *Start measuring* or you can press F5. While measuring you should be provided with audible feedback for each ring measured with a more pronounced sound made every 10th ring. If there is a problem communicating with your measuring hardware, check your settings

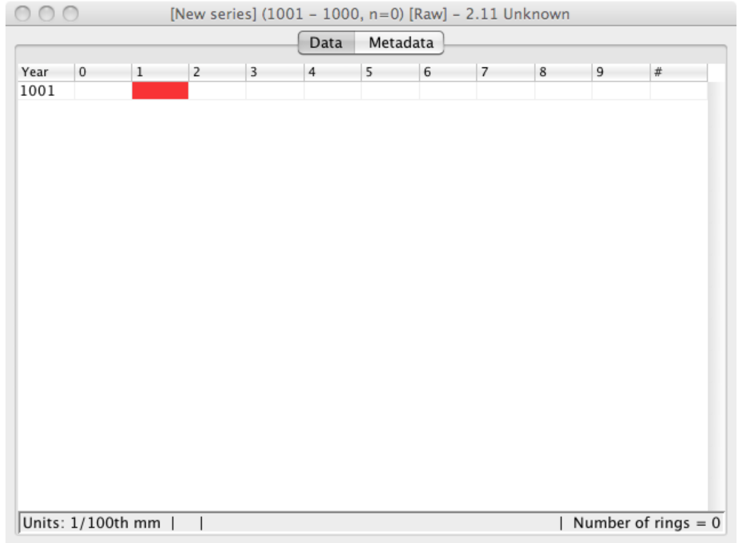


Figure 2.2: An empty data window ready to receive measurements. Note the status bar at the bottom includes buttons for changing the display units and cumulative statistics.

in the preferences dialog. If you still have problems contact the Corina developers by going to *Help* → *Report bug on last transaction*, making sure you include your email address and any further information.

While you measure your sample you can flag features in a ring by right clicking on any cell in the table and selecting one or more of the standard notes (see figure 2.3).

Corina supports all standard TRiDaS remarks including: fire damage; frost damage; crack; false ring(s); compression wood; tension wood; traumatic ducts; single pinned; double pinned; triple pinned and many others. Rings that include remarks are indicated by the relevant icon in the data screen. Depending on your method of work, this can be useful for keeping track of sample pin holes. For instance, if a missing or false ring is discovered after a sample has been pinholed, the offset in pinholes can be easily seen without resurfacing the sample. In the future Corina will also include support for user defined ring remarks.

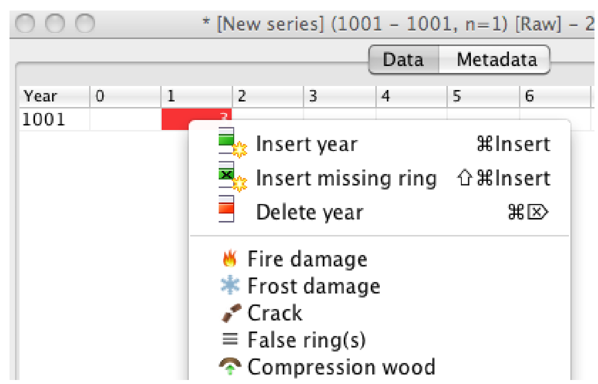


Figure 2.3: Right click context menu showing some of the options for adding remarks to rings.

The data screen also contains a status bar at the bottom. By click on the units section, you can switch between micron and 1/100th mm units. Corina understands the units being supplied by the measuring platform, therefore changes here are purely for display purposes only. If you have a platform that measures in microns, but prefer to see the values in 1/100th mm then you can use this feature. At the bottom ring of the status bar you can choose one of a variety of summary information about your series.

Once you have finished measuring your sample, you should then go to *File* → *Save* to save your series to the database.

2.4 Reconciling data

Corina has been developed not only for experience dendrochronologists, but as a tool for teaching students. It therefore includes a comprehensive 'reconciling' tool for supervisors to check the quality of measurements made by students. The reconcile dialog does a comparison of a measurement series made by a student with a references series of the same radius measured by the supervisor. The same dialog can also prove useful for comparing measurements from two experienced dendrochronologists when handling particularly difficult samples.

Chapter 3

Metadata

Metadata is 'data about data'. In Corina this means all the information associated with your physical samples and measurement series e.g. species, location, who measured it, dimensions, slope, soil type etc.

The metadata in Corina, and in fact the entire Corina data model, is based on the Tree Ring Data Standard (TRiDaS). Before you use Corina you may find it useful to read Jansma et al. (2010) so that you get a better understanding of the principles of TRiDaS, but a summary is also provided here.

3.1 Tree Ring Data Standard - TRiDaS

TRiDaS is an XML-based data standard for recording dendrochronological data and metadata. More than 80 dendrochronologists, computer scientists and specialists from research disciplines that rely on dendrochronology have so far contributed to its development, including dendroarchaeologists, art and architecture historians, ecologists, geologists and climatologists. The standard is therefore capable of recording the wide variety of metadata required by these different fields. TRiDaS builds upon other established standards, such as GML for the recording of locality information. The extensible nature of XML also means that TRiDaS can evolve to accommodate the changing needs of dendrochronologists over time.

TRiDaS includes a total of eight data entities: project; object; element; sample; radius; measurementSeries; derivedSeries; and value. Detailed descriptions of each of these entities are given below and their relationships are illustrated in figure 3.1.

A project is defined by a laboratory and encompasses dendrochronological research of a particular object or group of objects. Examples include: the dating of a building; the research of forest dynamics in a stand of living trees; the dating of all Rembrandt paintings in a museum. What is considered a "project" is up to the laboratory performing the research. It could be the dating of a group of objects, but the laboratory can also decide to define a separate project for each object. Therefore, a project can have one or more objects associated with it. Due to the way research is conducted at the Cornell Tree-Ring Lab, TRiDaS projects are not currently supported within Corina, although future plans include adding project support.

An object is the item to be investigated. Examples include: violin; excavation site; painting on a wooden panel; water well; church; carving; ship; forest. An object could also be more specific, for example: mast of a ship; roof of a church. Depending on the object type various descriptions are made possible. An object can have one or more elements and can also refer to another (sub) object. For instance a single file may contain three objects: an archaeological site object, within which there is a building object, within which there is a beam object. The list of possible object types is extensible and is thus flexible enough to incorporate the diversity of data required by the dendro community. Only information that is essential for dendrochronological research is recorded here. Other related data may be provided in the form of a link to an external database such as a museum catalogue.

An element is a piece of wood originating from a single tree. Examples include: one plank of a water well; a single wooden panel in a painting; the left-hand back plate of a violin; one beam in a roof; a tree trunk preserved in the soil; a living tree. The element is a specific part of exactly one object or sub object. An

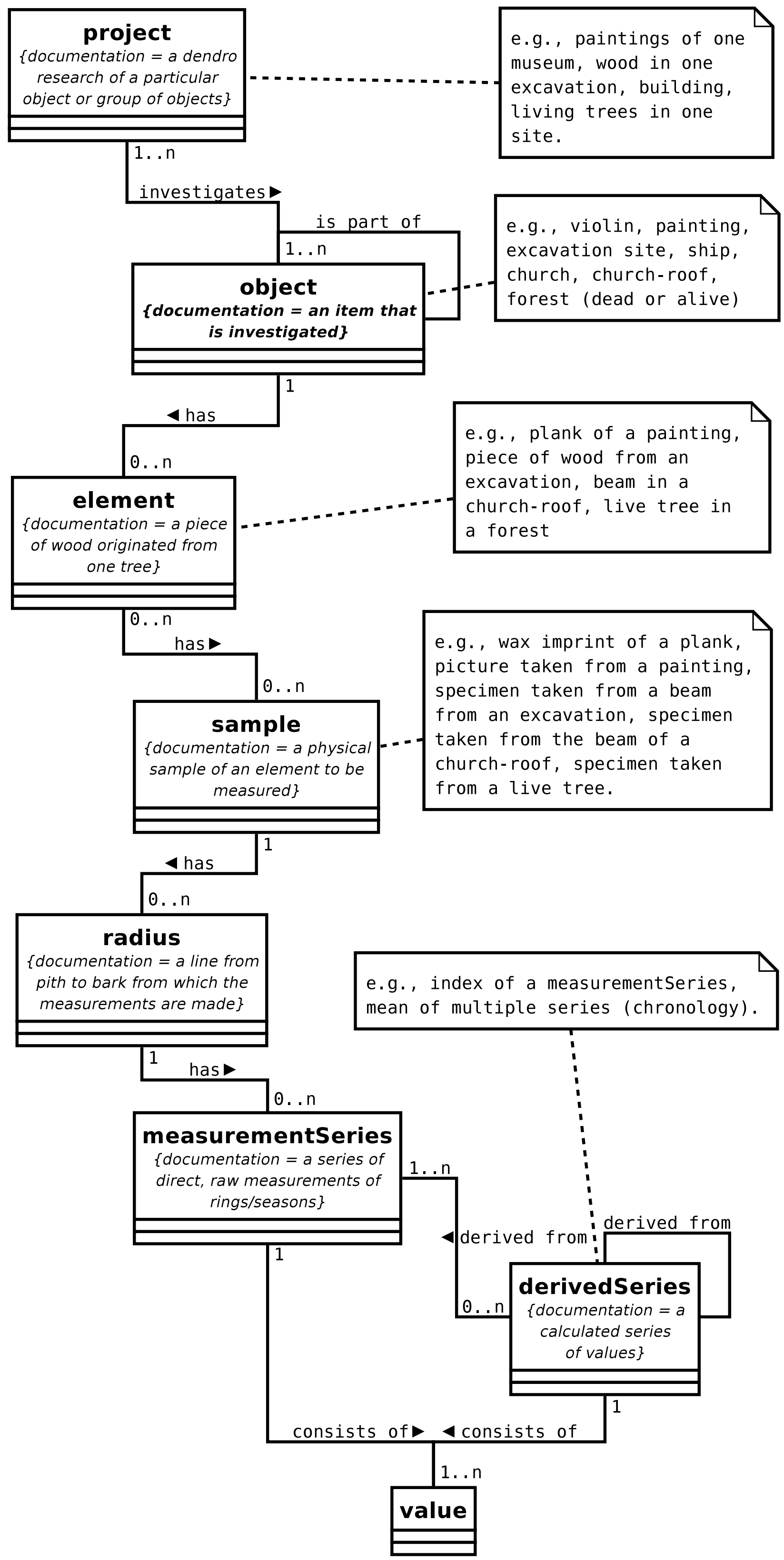


Figure 3.1: TRiDaS data model showing the relationships between data entities.

object will often consist of more than one element, e.g., when dealing with the staves (elements) of a barrel (object). One or more samples can be taken from an element and an element may be dated using one or more derivedSeries.

A sample is a physical specimen or non-physical representation of an element. Examples include: core from a living tree; core from a rafter in a church roof; piece of charcoal from an archaeological trench; slice from a pile used in a pile foundation; wax imprint of the outer end of a plank; photo of a back plate of a string instrument. Note that a sample always exists and that it can either be physical (e.g. a core) or representative (e.g. a picture). A sample is taken from exactly one element and can be represented by one or more radii.

A radius is a line from pith to bark along which the measurements are taken. A radius is derived from exactly one sample. It can be measured more than once resulting in multiple measurementSeries.

A measurementSeries is a series of direct, raw measurements along a radius. A single measurementSeries can be standardised or a collection of measurementSeries can be combined into a derivedSeries. The measurements themselves are stored separately as values.

A derivedSeries is a calculated series of values and is a minor modification of the "v-series" concept proposed by Brewer et al. (2010). Examples include: index; average of a collection of measurementSeries such as a chronology. A derivedSeries is derived from one or more measurementSeries and has multiple values associated with it.

A value is the result of a single ring measurement. Examples include: total ring width; earlywood width; latewood width. The values are related to a measurementSeries or a derivedSeries. In case of a measurementSeries the variable and its measurement unit (e.g. microns, 1/100th mm etc) are recorded as well.

Working top to bottom, the TRiDaS entities are nested within each other. For instance a project contains one or more objects, which in turn contains one or more elements, and so on. The benefit of this is that you record data once and once only. In standard file-based dendrochronological software, when creating measurement series you are typically required to type the name of the site, the species of tree etc over and over again. This is not only time consuming, but very error prone.

Keeping data consistent is also difficult. For instance, if it was determined that a tree species was identified incorrectly, in existing file-based software, the user would need to locate all data series from this tree and manally update the metdata. This is not the case in Corina. A tree is represented just once in Corina and samples of this tree, and the subsequent measurement series reference this one entry. If metadata for this tree needs to be changed, the tree record is updated in just this one place. Because the measurement series obtain this information by reference, then all associated series are automatically kept up to date.

3.2 Entering sample metadata

The metadata for a series is viewed and edited on the 'Metadata' tab of the main window such as that shown in figure 3.2. You can see the interface is organized according to the TRiDaS data model with separate screens for object, through to series.

When creating a new series, the metadata screens must be populated in order. This is necessary because of the nesting of entities described above. For instance, an element is associated with an object, so an object must be chosen because and element can be defined. Likewise, an element must be chosen before any samples of this element can be defined.

Much of the time the entities that you need will already be stored within the database. Instead of re-entering data, you simply need to select the existing entry from the database, saving a great deal of time. Depending on the situation buttons will appear at the top of the dialog to let you 'choose' an entry from the database, 'revert' to the previously chosen entry, 'change' the existing entry to a different one from the database, or create a 'new' record.

Please note that the content of these metadata screens is kept read-only by default. To edit the values, you must first click the padlock icon to unlock the fields. When you have finished making changes you need to press the save button to write the changes to the database before moving to another metadata screen.

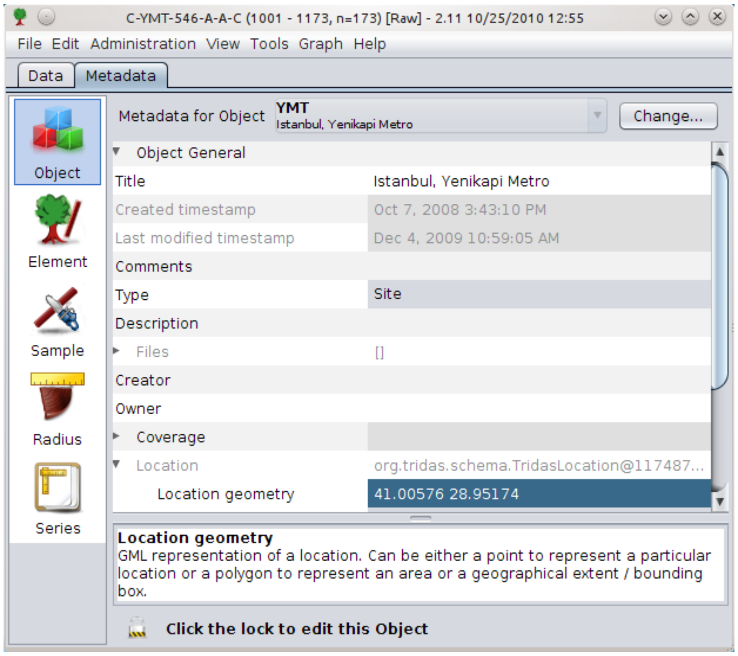


Figure 3.2: Example of the metadata dialog. The screen is showing the details of a TRiDaS object. Note that the location geometry field is highlighted and so a description of what is expected in this field is given below.

Very few of the metadata fields in the TRiDaS data model are mandatory, but a few are. In this case, these fields are highlighted with a red background. Note that whether a field is mandatory or not can depend on the other fields that have been filled in. For instance, the dimensions of an element are not required, but if dimensions are given then the units for these measurements must also be provided.

A number of the metadata fields are restricted with regards the values that you can enter. These are known as 'controlled vocabularies' in TRiDaS terms. Controlled vocabulary fields are represented by drop down menus. Similarly fields that expect numerical values (such as element dimensions) will only allow numbers. The final method data entry method is through custom dialogs. The only custom dialog currently implemented is for locations. This accepts coordinates in either decimal degrees or degrees minutes and seconds. Alternatively you can use data from a GPS handset by providing a GPS Exchange (GPX) format file containing the waypoints. The GPX format is the most common interchange format for GPS data. You can pick the relevant waypoint from the drop down menu. You can also preview the defined coordinates on a map using the 'view on map' button.

A popular open source GPS communication tool is GPS Babel. It is an easy to use application which can download data from the majority of GPS handsets. See *http://www.gpsbabel.org* for more information.

3.3 Entering bulk metadata

Entering metadata on a sample-by-sample basis works perfectly well, but does not necessarily fit best with the typical workflow of a laboratory. Samples do not typically arrive in a lab in ones and twos, rather in large quantities following a field excursion. In this case it is most efficient to enter all the metadata for the samples as they arrive. This is often best in terms of data accuracy as the metadata can be entered while the field notes are still fresh in the mind.

To enable the efficient entry of lots of metadata Corina includes the bulk data entry interface. This can be access from the file menu and is illustrated in figure 3.3. There are three pages, one each for objects, elements and samples.

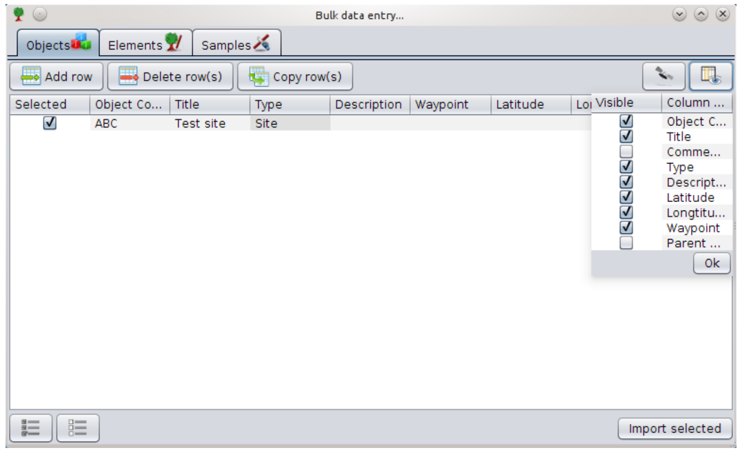


Figure 3.3: The bulk metadata entry screen. The 'show/hide columns' button has been pressed showing how the user can turn on and off particular columns.

The interface is designed like a spreadsheet so as to be as familiar to users as possible. Each row of the table represents a new entry in the Corina database. Which columns are shown to the user is determined by the 'show/hide columns' button on the top right of the screen.

The bulk entry interface also includes support for reading GPS units. By pressing the satellite button on the toolbar, the user can provide a GPS Exchange (GPX) format file containing the waypoint locations recorded in the field. Corina will add a waypoint column to the spreadsheet with a drop down menu which will automatically populate the latitude and longitude fields for the record.

It is common for many of the metadata fields to be same in a single field collection. For instance, when coring trees in a forest, they are often of the same species. Rather than requiring the user to repeatedly type the same data over and over, the 'copy row' button can be used to duplicate a record, and then the user can change the few fields that are different.

When you have entered all the data you want, you can press the 'Import selected' button to write the records to the database.

3.4 Metadata browser

The metadata browser interface provides a convenient way to view all the metadata within your Corina database. It can be accessed through the 'Administration' menu.

The metadata browser contains two parts: a hierarchical representation of all TRiDaS entities in your database on the left; and a metadata viewer for the selected entry on the right. This interface is also the best method for fixing mistakes in your database.

Although Corina's database architecture maintains integrity within your data, it does come at the price of being a little more complicated to fix mislabelled series. For instance, what if you were to measure a series 'B' and assign it to sample ABC-138-A only later to realize you misread the label and it was in fact ABC-188-A. In a traditional file-based system, you would probably just need to rename the file you'd just created. In Corina however, you need to redefine the relationship of the series within the database and reassign it to the create sample. This is best understood when looking at the hierarchical tree in the metadata browser. Hopefully you will see that you what you need to do is to move the series from its current position in the database to the correct one.

The reorganization of data in this way is achieved by right clicking on items in the hierarchical tree and choosing with 'merge' or 'reassign'.

Chapter 4

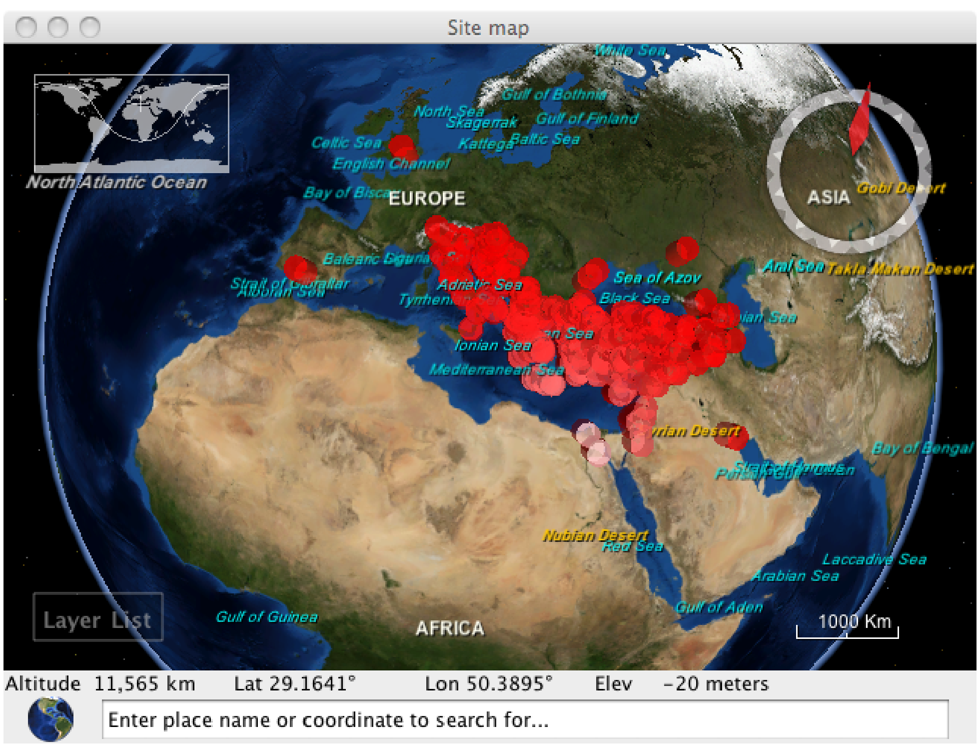
Mapping

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 13) 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 4.2.

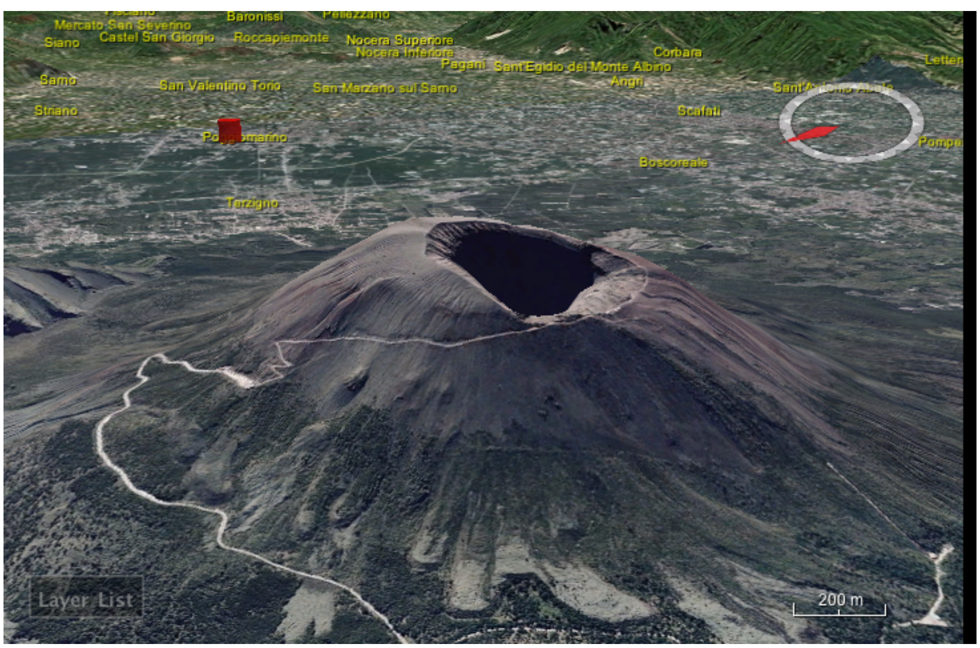


Figure 4.2: Example of 3D mapping in Corina.

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 **??**).

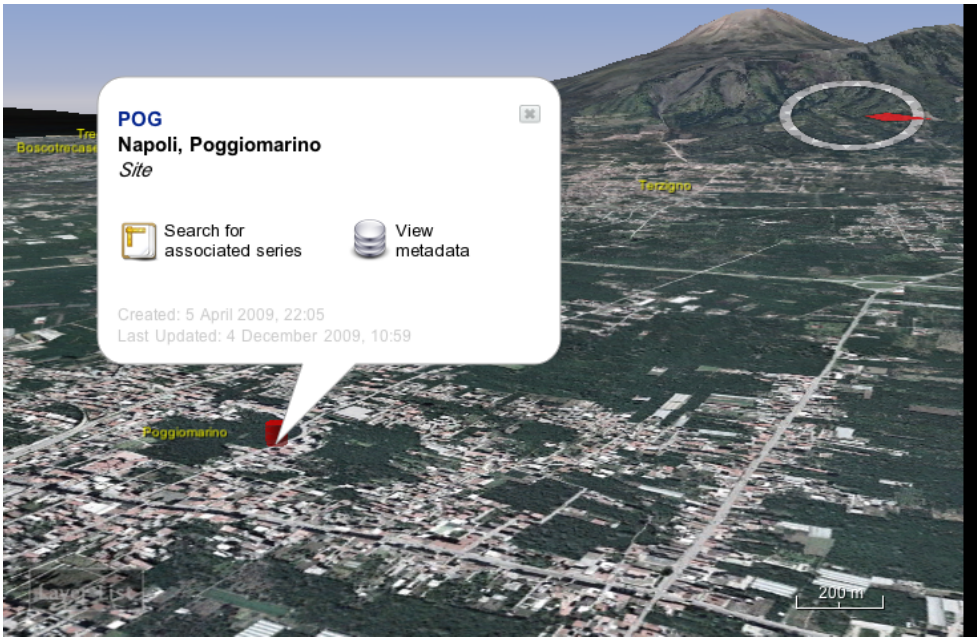


Figure 4.3: Screenshot of a map with information pin expanded

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.

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. By ticking layers in this list you can add data layers to your map.

You can add map data from other WMS servers by clicking the '+' tab and entering the URL of the server you would like to use. This will give an additional tab with all the available map layers. This server will only be available for the duration of your current session so will need to be added each time you start Corina. If you would like a particular WMS server to be made permanently available, your Corina administrator can do this (see 'Managing map services', on page 41 for further details). Additional WMS servers added in this way will be available to all users the next time they connect to your Corina server.

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, or environmental data for your study region. Figure **??** shows an example overlay of sea surfaces temperatures loaded dynamically from the NASA EO server.

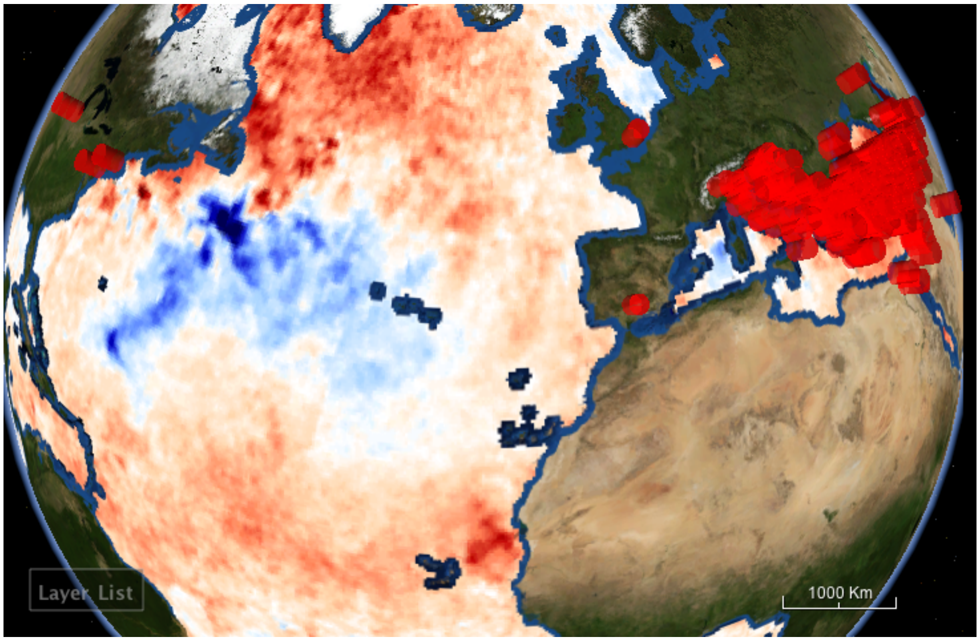


Figure 4.4: Map screenshot with a NASA sea surface temperature overlay dynamically loaded from the NASA WMS server.

4.4 Data map layers

Data map layers (i.e. site and sample locations) 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.

Chapter 5

Graphing

The graphing component is reused in many places throughout the Corina desktop application. The following description although based on the main graphing screen in Corina is largely applicable to all dialogs that include graphs (e.g. crossdating, indexing and reconciliation).

The main method for graphing your tree-ring data is by choosing an option from the Graph menu. Depending on the type of series you have open, the options available to you will be different. For raw measurement series, you will just have the option to 'Graph active series'. This will give you a simple graph of the current series that you have open. If you have a derived series open, then you may also choose 'Graph component series' which will plot all the series that go to create this series, or 'Graph all series' which graphs all the component series as well as the current series.

5.1 Controlling graphs

When newly created graphs are plotted according to the scale on the axes. A feature of Corina graphs though is that they can be manipulated directly on the screen. Both dendrochronology was computerized, dendrochronologists would plot rings manually on to graph paper. These paper graphs were then placed on lightboxes and moved around to enable comparisons. The graph function in Corina emulates this behaviour allowing users to click and drag graphs around to test for visual matches.

Figure 5.1 shows an example graph dialog. The mouse is hovering of the blue measurement series at relative year 1040 illustrating Corina's highlighting and guide line capabilities. A feature not shown in this screenshot is the illustration of sapwood rings. When sapwood rings are present the corresponding years on the chart are denoted via a heavier line.

The layout of graphs can be changed using both the toolbar buttons and menu options. The type of graph can be changed between a standard line graph, a semi-log graph and a toothed graph using the radio buttons. The remaining buttons are as follows:

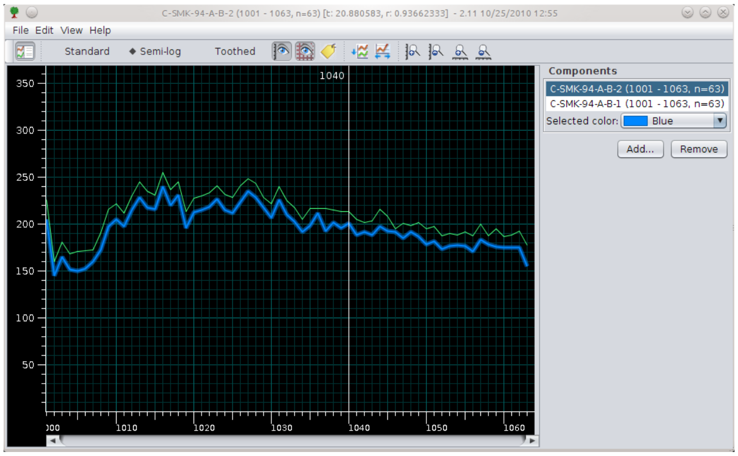


Figure 5.1: An example graph window containing two undated series of the same sample on a semi-log graph. Note the legend is visible with the options for adding or removing series.

- Zoom in on the horizontal axis
- Zoom out on the horizontal axis
- Zoom in on the vertical axis
- Zoom out on the vertical axis
- Toggle show/hide the grid lines
- Toggle show/hide the series labels
- Toggle show/hide the vertical axis
- Spread the series evening up the vertical axis
- Set the baselines of all the series to zero
- Resize graph to fit horizontally
- Toggle show/hide the legend

There are also a number of keyboard shortcuts that you might find useful:

Tab : Cycles through each graph component
Ctrl+W : Increase vertical scale
Ctrl+S : Decrease vertical scale
Ctrl+A : Increase horizontal scale
Ctrl+D : Decrease horizontal scale
Up arrow : Moves selected graph up by 10 units
Down arrow : Moves selected graph down by 10 units
+ : Moves selected graph up by 1 unit
- : Moves selected graph down by 1 unit
HOME : Scroll to first year of series
END : Scroll to last year of series
PAGE UP : Scroll left by one page width
PAGE DOWN : Scroll right by one page width
SPACE : Sets horizontal origin of all graphs to the same value

5.2 Exporting graphs

To export your graphs for use in reports you can go to File →Export plot as PDF file, or Export plot as PNG file. This presents you with a dialog for setting the colors, labels and size of the exported image. This functionality is due for an overhaul in the future to provide more flexible support for publication quality graphics.

Chapter 6

Importing and exporting

Importing and exporting of dendro data in Corina is provided through the TRiCYCLE libraries. TRiCYCLE is a universal dendro data conversion application for converting back and forth between over 20 supported data formats (Brewer et al., 2011). The open source libraries that provide the functionality to TRiCYCLE are incorporated directly into Corina providing support for all these formats.

Belfast Apple	Oxford
Belfast Archive	PAST4
Besancon (including SYLPHE variants)	Sheffield D-Format (Dendro for Windows)
CATRAS	Topham
Comma delimited text files (CSV)	TRiDaS
Corina Legacy	TRIMS
DendroDB	Tucson (RWL and CRN)
Heidelberg (TSAP-Win)	Tucson Compact
Microsoft Excel 97/2000/XP	VFormat
Microsoft Excel 2007	WinDENDRO
Nottingham	

Table 6.1: List of the twenty-one formats supported by Corina. See appendices **??** – **??** (pages **??** – **??**) for full descriptions.

6.1 Exporting data

Exporting data is initiated by the File ➔Export data menu. If this is called from the main Corina data window, it will export the current series. If it is called from the Corina home screen, then it will present you with the database browser and allow you to pick one or more series to export.

6.2 Exporting graphs

6.3 Exporting maps

6.4 Importing data

Chapter 7

Curation

7.1 Laboratory workflow

Corina includes a number of functions to assist you with the curation of your physical sample collection. To understand how these are designed to assist users, we must first consider the workflow within a laboratory.

In research laboratories, samples generally come to the lab in large batches following field collection. In this case the typical workflow may be as follows:

1. Collect samples and record field notes as accurately as possible
2. On returning to the lab enter field notes as soon as possible into the 'bulk data entry' interface
3. Print sample barcode labels
4. Prepare physical samples and label with barcodes
5. Assign samples to storage boxes
6. Measure samples, using barcodes to recall metadata from database
7. Crossdate samples / build chronologies
8. When all samples from a box are completed register box as archived and then store

For commercial labs offering dendrochronological dating as a service, samples more likely to arrive in smaller batches. In this case, the bulk data entry interface may not be the most efficient method for entering metadata. In this case the user may simply prefer to use the *File* ➔ *New* method for each sample.

Either way, the concept behind the curation of a collection in Corina revolves around the accurately recording as much metadata about a sample as possible, then labelling the physical sample with a label containing a barcode for Corina and sample code for the user. By entering a sample into the database as soon as it enters the lab, it can be traced throughout the workflow. When a chronology is built, it is easily to quickly and efficient locate all samples that have been used. By assigning samples to boxes, groups of similar samples (e.g. from the same site) can also be easily stored together and located quickly and efficiently.

7.2 Barcodes

Barcodes allow you to keep track of what samples you have and where they are stored. Although it is not essential to use the barcode functions, we strongly suggest you do because they save time and money, but most importantly they greatly reduce the scope for erroneous data entry. For instance, when measuring a sample a user simply scans its barcode and all the relevant metadata is retrieved from the database, rather than relying on them to enter data manually. Barcodes have been routinely used in the retail industry since the 1980s. They can be equally as useful in dendrochronology laboratories.

Corina creates and reads barcodes for samples, measurement series and boxes. Each barcode encodes the unique identification code stored in the Corina database for each of these entities. Due to Corina use of universally unique identifiers (UUIDs), these codes a garanteed to be unique opening the opportunity of labs to loan samples, much like libraries do with books. There are many styles (or 'symbologies') of barcodes in use today, but Corina uses one of the most common (Code 128) which is supported by the vast majority of barcode readers. For a detailed discussion on the specifications of the Corina barcode see section 13.4.

Basic barcode readers are now cheap and widely available, with basic devices retailing for a few tens of dollars. Most are characterised as 'keyboard interface devices' and work like an automated keyboard, typing in a string of characters when a label is scanned.

Within the Corina application, whenever the user is required to specify a box, sample or series, they have the option of typing the human readable lab code or scanning the barcode. By using the barcode, the user can be sure they are not entering typographic errors so we recommend using barcodes whenever possible.

The most important barcode is the label for the physical wood sample. These are easily generated through the *Administration* ➔ *Labels* ➔ *Sample labels* menu entry. Currently the layout of these labels is fixed, but in the future we aim to provide different styles.

7.2.1 Sample labels

Before labels can be generated, metadata entries the sample level must have been made in the database. This is typically done using the 'bulk data entry' interface (see page 16). If samples are already in the database, the user needs to select the object of interest in the label creation dialog to see all the available samples. It is then just a matter of selecting the samples of interest and moving them into the 'selected' column. Once the list is populated (samples from multiple objects can be included), then you can either click 'Preview' to see a PDF of the labels, or 'Print' to print directly.

The current label style is designed to fit on standard core mounts and most samples. There are no widely available die-cut labels that fulfill this need, so the labels are intended to be printed on archival grade full page sheet labels (e.g. Avery® layout 6575), and then manually guillotined.

7.2.2 Box labels

The procedure for printing box labels is the same as for samples. Samples must have already been assigned to boxes before the label is printed (see section 7.3 for details). To print (or preview) box labels go to *Administration* ➔ *Labels* ➔ *Box labels*. The label style is designed to be printed on $5'' \times 8\frac{1}{8}''$ labels, two per sheet such as the Avery® 6579 layout.

> Until dynamic label styles have been implemented, box labels will print one per page. To make use of the second label on the page, the same sheet should be fed through the printer a second time.

7.2.3 Series barcodes

Series barcodes are printed at the top of a standard series report. These are produced through the *File* ➔ *Print*, or *File* ➔ *Print preview*, menus.

7.3 Storage boxes

Corina uses the term 'box' to refer to the collection of samples you archive. Many labs (including Cornell) use cardboard bankers boxes to store samples once they are completed, but the same box concept could refer to draws or shelves in your collection.

7.3.1 Creating and editing boxes

Records for boxes in the system are created and edited through the *Administration* ➔ *Curation* ➔ *Box details* menu. To editing an existing box, you can scan the barcode label on the box, or select from the list. To create a new box, click the 'Create new box' button and enter its details. There is no restriction on what boxes should be called, but it is probably easiest if you use some sort of numerical sequence to assist with organizing the boxes in your store. At Cornell, we use a two part name for each, the first being the year of collection, the second being a sequential number (e.g. 2009-11).

The contents tab lists all the samples that have been assigned to this box. To add new samples, simply click the 'Add sample to box' button and scan the sample's barcode.

7.3.2 Inventory

An important feature of any collection management system is the ability to perform an inventory on the collection. Even with the most robust system, samples will always go astray so its important to be able to periodically check that the boxes contain what you expect.

The 'Contents' tab of the Box details dialog contains a feature to assist with this. Next to the list of samples that are recorded as present, there is a temporary checklist column. By checking the boxes for each sample actually stored in the box it is easy to see which samples have been mislaid. If the 'Mark unchecked as missing from box' button is then pressed, the date and time the discrepancy was noted is then recorded in the comments field for the box.

7.3.3 Checking boxes in and out

Corina includes function for checking boxes in and out of a store, much like when a book is borrowed from a library. The *Administration* ➔ *Curation* ➔ *Check out box from store* and *Administration* ➔ *Curation* ➔ *Return box to store* menus do just this. You can either scan the box barcode or select the box from the drop down menu. These options record when a box is checked out/in and by whom. These details can be seen by users in the box details dialog.

7.3.4 Locating samples

As you might expect, Corina also includes a function for locating your physical samples. This is available in the *Administration* ➔ *Curation* ➔ *Find a sample* menu. There are three methods for locating a sample: via barcode; via lab code; and manually by object/element/sample.

If you have the sample in your hand and you simply want to know which box it should be returned to you can scan the barcode. If you are looking for a sample and you know its lab code then you can enter this instead. Alternatively, you can use the drop down menus to search for one or more samples at once. For instance, you can locate all the samples for a particular object and element.

Chapter 8

Indexing

Trees tend to put on big rings when they're young, and smaller rings when they get older. Some trees put on very large rings, while others put on very small rings. These variations in growth can make it difficult to crossdate samples. Some dendrochronologists therefore prefer to index or normalize their ring width data before combining into chronologies.

Indexing is a manipulation you can perform on your data to make it easier to crossdate.

The procedure for indexing is as follows:

1. You open a series (raw data)
2. You ask Corina to index it
3. Corina shows you some possible curves
4. You pick a curve (based on its graph, statistical scores, and your expectation of how the tree is growing)
5. Corina converts each year's ring width to a ratio of actual growth to expected growth for that year
6. You save the series (indexed data)

Indexing changes the units of a dataset. A raw sample has units of hundredths of a millimeter (0.01 mm) or microns. An indexed sample has units of parts per thousand (0.1%, or ‰).

This doesn't cause a problem with crossdating. The t-score normalizes all samples as part of its test, and the trend only cares if the values are increasing or decreasing. For more information on crossdating and chronology building, see chapter 9. It does, however, cause a problem with 'summing' since summing needs to take the average (what's the average of 1mm and 75%?). Therefore, the samples in a sum must be either all raw, or all indexed.

8.1 Types of index

There are a total of six different indexing methods available in Corina:

8.1.1 Exponential Index

This is the most commonly used index as it matches the way trees typically grow. Quickly when young and then gradually slower. An exponential index is therefore by far the most common index you'll use as 9 times out of 10 this will be the best choice.

This index tries to fit an equation of the following form to your data, searching for the best values of a, b and p.

- $y = a + be - px$

This is sometimes called a negative exponential index, because the exponent is negative. Corina doesn't require that the exponent is negative, but if it's not, using this index probably isn't such a good idea; it means the tree is generally getting bigger, not smaller.

The least-squares algorithm used comes from Cormen et al. (2001); the matrix solving function comes from Van Loan (1999).

Sometimes the exponential index does a lousy job. If a tree is living in a crowded area and the trees around it get cut down, suddenly it has much better growing conditions, so it might grow faster as it gets older, instead of slower. If you tried to use an exponential curve on a tree like this, it would exaggerate this growth, and useful data would get flattened out.

The result is you're looking at the growing conditions of this one tree, so it's not going to crossdate as well.

Alternatively, imagine you are working on a tree with a fire scar that has a few very large rings. An exponential index wouldn't take much notice of this, because most of the sample is still shaped like an exponential curve, but when you applied it they would be grossly out of proportion. For these types of samples, there are other indexing algorithms available.

8.1.2 Polynomial Index

When you ask Corina to perform a Polynomical Index it tries to fit a polynomial curve to your data using the following equation:

- $y = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_2 x^2 + a_{1x} + a_0$

You decide what degree polynomial, n, to use and Corina automatically finds the best values of a_0, $a_1 \ldots a_n$, to fit your data.

8.1.3 Horizontal Line Index

This only changes the magnitude not shape of the curve and is used when you would link to combine raw and indexed data together. It is a special case of polynomial where the horizontal line is equal to the average value.

- $y = x_{avg}$

This index is not used for crossdataing because dividing each value by the same value doesn't change the shape of the curve, only its magnitude. A horizontal line index is, however, useful because every element in a sum must use the same units, either raw or indexed. Therefore if you want to include a raw sample with an indexed sample then a horizontal line index can be used to convert the raw sample without otherwise altering the shape of the curve.

8.1.4 Floating Index

This is a running average of the 11 surrounding years. The adaptive index is generally used as a "last resort" when both exponential and a high-degree polynomial have failed. It is simply the average of the eleven surrounding years:

- $ind_i = 1/11(data - i - 5 + data_{i-4} + \ldots + data_{i+4} + data_{i+5}$

This index was originally called floating average, probably in reference to the fact that the index curve "floats" around, not following any explicit $y = f(x)$-type formula. But people tended to call it floating, and then floating-point, which means something very different. You might still hear people calling this index by these other names.

8.1.5 High-Pass Filter Index

The high-pass index is a more general case of the adaptive index. Instead of simply taking the average of 11 values, it takes a weighted average. It's an example of a "high-pass" filter because high-frequency signals can pass through, but low-frequency signals are filtered out.

The default is "1-2-4-2-1", meaning:

- $ind_i = 1/10(data_{i-2} + 2{\cdot}data_{i-1} + 4{\cdot}data_i + 2{\cdot}data_{i+1} + data_{i+2})$

This comes from Cook and Peters (1981) who used it as a discrete filter before moving to a cubic spline. Note that almost half $(4/10)$ of the computed value is simply its old value. The high-pass index is nearly the same as the input, so the χ^2 values are usually the lowest, therefore do not choose this index solely on a low χ^2 value.

8.1.6 Cubic Spline Index

Cubic splines are a very specific type of high-pass filter. A cubic spline curve is created by combining a collection of cubic (3rd degree polynomial) functions.

There are many methods for constructing cubic splines through a dataset. The algorithm used by Corina has a parameter, s, which controls how tightly the spline fits the data. A lower value fits the data more tightly, a higher value fits the data more loosely. Therefore, s=0 fits the data exactly while s=1 is a simple line. A good starting point for dendro data seems to be around $s = 1x1016$.

Cubic splines were first used for dendro by Cook and Peters (1981) using an algorithm from Reinsch (1967).

You can change the s-value used for the subic spline in the preferences. You might use a cubic spline in the same cases you would use a high-pass filter e.g. when the sample doesn't generally follow an exponential or polynomial curve very well, perhaps due to a fire scar.

8.2 Indexing data

To index your data, first you need to open the series you would like to index. Next choose *Tools* ➔ *Index* to display the indexing dialog (figure 8.1).

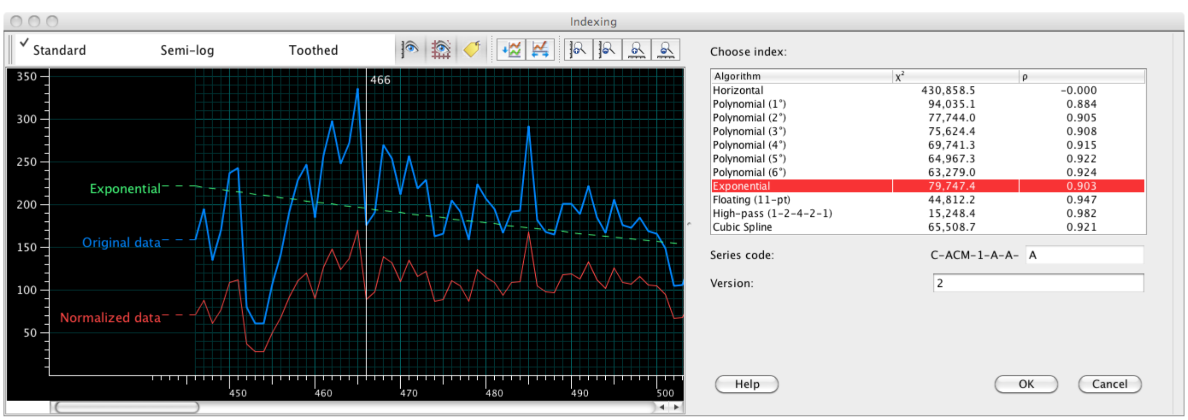


Figure 8.1: Indexing dialog showing the original data in blue, the exponential index of this data in green, and the normalized data in red.

From the indexing dialog you can then choose which type of index to apply to your data. The table on the right show the available options along with the χ^2 and p values to help you choose the correct index to use. The graph shows your original data, the index line and the result of applying the index to the data and changes dynamically as you pick between different indexing methods. Once you have decided which index you want to use, select it, and click OK ensuring that you have given your data series a new version number.

Chapter 9

Crossdating and chronology building

All algorithms work in pretty much the same way. There's a "fixed" sample, and there's a "moving" sample. Imagine you have printouts of their graphs on translucent paper. The fixed graph is taped to a table, and you can slide the moving sample left and right. This is actually how it was originally done, on graph paper, with one inch per decade. Start with the moving sample to the left of the fixed sample, overlapping it by 10 years. Look at how well the graphs match: this is the first score that's computed. Slide the moving sample to the right one year and so on until you reach the end.

You could do it all simply by moving graphs and eyeballing the crossdates like this but there are hundreds of sites and millennia of chronologies you'll want to crossdate your samples against, so that would take a while. Corina has a few algorithms to find likely crossdates almost instantaneously. They aren't perfect, though, and all crossdates should be inspected visually to ensure they are a good fit.

9.1 Algorithms

Corina includes a total of five different algorithms for crossdating:

9.1.1 T-Score

The t-score is the classic crossdate. Everybody quotes t-scores: if you want to brag about how good a cross is, you tell them your t-score. Unfortunately, every dendro program seems to have a slightly different implementation of tscore, so the numbers you get from Corina might not be exactly comparable to the numbers from other programs.

The version Corina uses is based on the algorithms given in Baillie and Pilcher (1973), though with some apparent bugs corrected. In the following equations, $x_0, x_1, x_2, \ldots$ are the data of the fixed sample in the overlap, $y_0, y_1, y_2, \ldots$ are the data of the moving sample in the overlap, and N is the length of the overlap.

The first step is to make each dataset bivariate normal by replacing each value with the mean of the values around it, and then taking its natural logarithm. The preparation for the T-Score is therefore done as follows and is done to both the fixed and moving series:

- $x_i \leftarrow \frac{x_{i-2}+x_i+x_{i+1}+x_{i+2}}{5}$
- $x_i \leftarrow ln(_{xi})$

The student's T computation is then done as follows:

- $s_{xy} = \Sigma x_i y_i - N(x_i - x_{avg})(y_i - y_{avg})$
- $s_{xx} = \Sigma x_i^2 - N(x_i - x_{avg})^2$
- $s_{yy} = \Sigma y_i^2 - N(y_i - y_{avg})^2$

- $r = \frac{s_{xy}}{\sqrt{(s_{xx}s_{yy})}}$
- $t = r\sqrt{\frac{N-2}{1-r^2}}$

The t-score is called a "parametric" algorithm, because it takes into account the magnitudes of the samples.

A t-score is considered statistically significant if it's greater than a certain value. Just what this value is varies with the length of the overlap between the samples: a 500 year overlap can have a t-score of 2.6 and pass, but an overlap of only 15 years would have to be higher, like 3.0. In reality most dendrochronologists require t-scores to be much higher than this, and would also insist on overlaps of many decades.

9.1.2 Trend

Trend is another popular crossdate statistic. It computes the percentage of years with the same trend (going-up- or going-down-ness). Scores greater than 60%-70% are good. Trend is also referred to as ufigkeitsko-Gleichläeffizient, Gleichläufigkeit and Eckstein's W.

The trend is the simplest crossdate. For each sample, it computes the trend of each 2-year interval (1001-1002, 1002-1003, and so on). The trend of a 2-year interval is simply whether the next ring is larger, smaller, or the same. The trend score is the percentage of intervals in the overlap which are the same. For example, a 75% trend (a very good score, by the way) means that for 75% of the intervals in the overlap, both samples went up in the same years and down in the same years.

If one sample stays the same, and the other increases or decreases, Corina considers that to be halfway between a same-trend and different-trend, and gives it half a point. Trend is a "non-parametric" algorithm, because it only takes into account if a given ring is bigger or smaller than the previous one, not by how much. To the trend, a drop of "100 1" looks exactly the same as a drop of "100 99". Two completely random samples will have a trend of 50%, on average. So you'd expect a trend must be greater than 50% to be significant.

According to Huber and Fletcher (1970), a trend is significant if:

1. $tr > 50\% + \frac{50}{\sqrt{N}}$ - For example a pair of samples with a 50-year overlap needs a $50 + 50\sqrt{50} = 57.1\%$ trend to be significant, but at a 400-year overlap need only a $50 + 50\sqrt{400} = 52.5\%$ trend. In practice, however, this doesn't tend to work terribly well. Using this scheme, there are typically about three times as many "significant" trend scores as t-scores, and users want this narrowed down a bit more. So take $\sigma = 3$ and use:
2. $tr > 50\% + \frac{50\sigma}{\sqrt{N}}$ - This gives about the same number of significant trend scores as t-scores.

Trends are also used in reconciliation. After they've been reconciled, both readings of a sample should have 100% trend.

9.1.3 Weiserjahre

The Weiserjahre algorithm is used for crossdating summed samples (chronologies) against single samples. All of the algorithms that have been mentioned so far only compare the ring widths. This works fine for raw samples, but when crossdating summed samples, there's a lot more information available, namely, the Weiserjahre data. Wouldn't it make sense to count a [20] 19×1 ring more heavily than a [1] $1 \div 0$ ring? 19 out of 20 samples think it's an increasing year, not just 1.

This is what the Weiserjahre cross does: for each possible overlap, it starts by counting the number of significant intervals of the master for that overlap. A significant interval is one with at least 3 samples, where at least 75% of them have the same trend. Then it computes the percent agreement (like the trend) between the master and the raw sample for only those significant years of the overlap. Of course, for the trend of the master, it doesn't use the trend of the master; it uses the trend of the majority of its elements. They're usually the same, but not necessarily.

Another way to think about the Weiserjahre crossdate is: it's like a trend, but ignoring years where the sum has only 1 or 2 samples, or where there isn't an overwhelming trend in the sum. Also like the trend, the results are given as a percentage.

9.1.4 R-Value

The R-value, or correlation coefficient, is a crossdate which you'll almost never use. It's not terribly useful to dendrochronologists, but statisticians might want to know its value, so Corina makes it available.

The R-value is used in the T-Score, the T-score being defined in terms of the r-value and the overlap, N. If you look at the equations for calculating a T-Score you will see on the penultimate line:

- $r = \frac{s_{xy}}{\sqrt{(s_{xx} s_{yy})}}$

An r-value can range from 0.0 (no correlation) to 1.0 (perfect correlation).

9.1.5 D-Score

The D-score (AXE), is a combination of the T-Score and Trend.

- $D = (tr - 50\%) \times t$

Corina considers 40 to be the threshold for significant AXE.

9.2 Crossdating series

9.3 Managing chronologies

Chapter 10

The Corina 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.

10.1 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 of these you need to type one commands listed below. The first line installs Gnome and the second KDE. Windows users that are new to Linux may find KDE more familiar than Gnome.

```
sudo apt-get install ubuntu-desktop
```

```
sudo apt-get install kubuntu-desktop
```

10.2 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.

10.2.1 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.

10.2.2 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 13.1.

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.

10.3 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.*

10.3.1 PGAdminIII

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.

10.3.2 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.

10.3.3 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!

10.4 Managing map services

Chapter 11

Help

Part II

Developers guide

Chapter 12

Development environment

Corina is open source software and we actively encourage collaboration and assistance from others in the community. There is always lots to do, even for people with little or no programming experience. Please get in touch with the development team as we'd love to hear from you.

12.1 Developing Corina Desktop

The IDE of choice of the main Corina developers is Eclipse (*http://www.eclipse.org*. There are many other IDEs around and there is no reason you can't use them instead. Either way, the following instructions will hopefully be of use.

We have successfully developed Corina on Mac, Windows and Linux computers over the years. The methods for setting up are almost identical.

The first step is to install eclipse, sun-java6-jdk and subversion. These are all readily available from their respective websites. On Ubuntu they can be install from the command line easily as follows:

```
sudo apt-get install eclipse subversion sun-java6-jdk
```

Once installed, you can then launch Eclipse. To access the Corina source code you will need to install the Subversive plugin to Eclipse. As of Eclipse v3.5 this can be done by going to *Help* ➔ *Install new software*. Select the main Update site in the 'Work with' box, then locate the 'Subversive SVN Team Provider' plugin under 'Collaboration'. If you are using an earlier version of Eclipse you may need to add a specific Subversive update site. See the Subversive website (*http://www.eclipse.org/subversive/*) for more details. Once installed you will need to restart Eclipse.

Next you need to get the Corina source code. Go to File *New* ➔ *Project*, then in the dialog select *SVN* ➔ *Project from SVN*. There are two methods of accessing the Corina repository: anonymously, in which case you will have read only access; or with a username provided by the Corina development team. Anonymous users will need to add a repository in the form: *http://dendro.cornell.edu/svn/corina/* and full users will need to use *svn+ssh://dendro.cornell.edu/home/svn/corina/*.

Once the project has downloaded to your workspace, you may need to set the compliance level. This can be done by going to *Project* ➔ *Properties* ➔ *Java compiler* and choosing compliance level of 6.0. Corina uses a handful of Java 6 specific functions, particularly with regards JAXB, so will not run successfully with Java 5.

To launch Corina, you will need to *Run* ➔ *Run Java application*. Create a new run configuration with the main class 'set to edu.cornell.dendro.corina.gui.Startup'.

12.2 Developing the webservice

The Corina webservice is written entirely in PHP. Eclipse is used for this development too so most of the setup steps are identical to setting up Eclipse for developing the Corina desktop client. You will, however, probably want to install the PHP development plugin so that you get syntax highlighting etc. See the Eclipse PDT website (*http://www.eclipse.org/pdt/*) for further information.

12.3 Dependencies

Apache commons lang Apache 2.0 TridasJLib Apache 2.0 Batik Apache 2.0 RXTXcomm LGPL JDOM Apache 2.0 Swing layout LGPL Log4J Apache 2.0 JNA LGPL Apache mime 4J Apache 2.0 Commons codec Apache 2.0 http Client LGPL Http core Apache 2.0 Http mime Apache 2.0 Jsyntaxpane Apache 2.0 L2prod-common-shared Apache 2.0 L2prod-common-sheet Apache 2.0 l2fprod common buttonbar Apache 2.0 itext GAPL PDFRenderer LGPL DendroFileio Apache 2.0 java simple mvc MIT JgoogleAnalyticsTracker MIT gluegen BSD jogl BSD+ nuclear clause worldwind NASA slf4j MIT jfontchooser LGPL miglayout BSD pljava BSD postgresql PostgreSQL License (BSD/MIT) forms BSD simplelog GPLv3 jxl LGPL wizard GPLv2 netbeans swing outline GPLv2

Chapter 13

Systems architecture

The centralised nature of the Cornell Tree-Ring Lab data required a server-client architecture of some type. In the original Corina v1 this was achieved simply by having users save their data in a network folder saved on a central server. Whilst this method was adequate, it has many data storage issues that can be largely solved by moving the data storage infrastructure to a relational database management system.

Although it would be possible (and arguably simpler) to refactor Corina to talk directly to one central database server it was decided to go a step further and implement a Web Services orientated server-client architecture for Corina v2.

A web services approach decouples the desktop client from the server so that the server can work on its

13.1 Authentication design

The authentication mechanism 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.

The first time a client attempts to retrieve data from the webservice (or when the client's credentials are incorrect or have expired) the following events occur:

- Server returns an message requesting authentication. This message includes a nonce (a hash of the current date and time to the nearest minute) which we will call 'server nonce'.
- The client creates a second nonce (client nonce) which is a random hash of it's choosing, and a response which is a hash of "username:hashofpassword:servernonce:clientnonc". It sends this response, along with the username and client nonce back to the server but does not send the original server nonce.
- The server computes the same "username:hashofpassword:servernonce:clientnonce" hash using the information it has stored in the database. As the server nonce is constant for a minute the two response should match. If not the server recomputes the server nonce for one minute ago and tries again. This ensures that the server nonce sent to the client is valid for between 1 and 2 minutes.
- Once the server authenticates the user a session cookie is sent to the client. On subsequent requests the server recognises the session id and doesn't request authentication again.

As the user's password is hashed at all points, even if the communication is hijacked the attacker will not be able to derive the users password. The user's password is also stored in hash form within the database. This also means that system administrators do not have access to the passwords either.

The use of the server nonce within the response means that it will only be valid for a maximum of two minutes. This removes the possibility of a replay attack.

13.2 Database permissions design

The database has a user and group based security scheme at three object levels: site, tree and measurement. A user can be a member of one or more groups, and groups can be members of zero or more other groups. The current implementation allows for one nested level of groups within groups however this could be extended if required. Security is set on a group-by-group basis rather than on a single user to ensure ease of management.

There are five types of permissions granted: create, read, update, delete and no permission. Each permission is independent of each other with the exception of 'no permission' which overrides all other permissions.

A group can be assigned one or more of the permissions types to any of the sites, trees or measurements in the database. Intermediate objects such as subsites, specimens and radii inherit permissions from their parent object. For instance if a group has permission to read a site then it will have permission to read all subsites from that site.

It is envisaged that most of the time, permissions will be set on a site-by-site basis. It will not be necessary to explicitly assign permissions to trees and measurements as all permissions will be inherited. So assuming that no permissions are set on a tree for a particular group, the permissions for the tree will be derived from the site from which the tree was found. If, however, permissions are assigned to the tree, then these will override those of the site. In this way it will be possible to allow a group to read the data from one particular tree from a site in which there otherwise do not have permission to access.

Privileges are cumulative. This means that if a user is a member of multiple groups then they will gain all the privileges assigned to those groups. If one of the groups that the user is a member of has 'no privileges' set on an object it will however override all other privileges. Therefore if a user is a member of groups A and B, and group A has read privilege and group B has 'no privilege' then the user will not be able to access the record.

A special 'admin group' has been created into which only the most trusted users are placed. Members of the admin group automatically gain full privileges on all data within the database. They also have permission to perform a number of administrative tasks that standard users are insulated from.

13.3 Universally Unique Identifiers

All entities in the Corina database have a primary key based on the Universally Unique Identifier (UUID) concept. This is a randomly created 128-bit number which due to the astronomically large number of possibilities (3×1038) means that it is guaranteed to be unique across all installations of Corina. This code is typically represented by 32 hexadecimal digits and 4 hyphens like this: 550e8400-e29b-41d4-a716-446655440000.

13.4 Barcode specifications

Barcodes in Corina are based on the UUID primary keys of database entities. Because they are used for different entities in Corina (boxes, samples and series) it was also necessary to incorporate a method for determining what type of entity a barcode represents. This is done by appending a single character and a colon to the beginning of the UUID: 'B:' for box; 'S:' for sample; 'Z:' for series.

The barcodes in Corina use the Code 128 scheme. This symbology was chosen as it allows the encoding of alphanumeric characters in a high-density label and can be read by all popular barcode scanners. While it would have been possible to create a barcode of plain UUIDs, the 36 (or even 32) characters would result in a barcode wider than many scanners could read. Most scanners on the market have a maximum scan width of at least 80mm, so this was used as the baseline to work to.

To make the barcodes less than 80mm, the UUID (with prepended entity type character code) are Base64 encoded. For example the series with UUID 3a8f4336-d17d-11df-abde-c75e325aebae would be encoded from Z:3a8f4336-d17d-11df-abde-c75e325aebae to become: Wjo6j0M20X0R36vex14yWuuu

Part III

Appendices

Appendix A

GNU General Public License

The Corina server and desktop client are released under the GNU General Public License (GPL) version 3.

Copyright © 2011 Peter Brewer

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A.1 Preamble

Copyright © 2007 Free Software Foundation, Inc. *http://fsf.org/*

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

A.2 Terms and Conditions

0. Definitions.

   "This License" refers to version 3 of the GNU General Public License.

   "Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

   "The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

   To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

   A "covered work" means either the unmodified Program or a work based on the Program.

   To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

   To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

   An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

   The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

   A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

   The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

   The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

   The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

   The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

   All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

   You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

   Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

   No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

   When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

   You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

   You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

   You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

   (a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
   (b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
   (c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
   (d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

   A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

   You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

   (a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
   (b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
   (c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
   (d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

(e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

(a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
(b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
(c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
(d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
(e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
(f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

   You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

   Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

   An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

   You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

   A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

   A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

   Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

   In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

   If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

   If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

   A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

   Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

   If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

    Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

    The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

    Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

    If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

    Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

    THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

    IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

    If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

Appendix B

GNU Free Documentation License

B.1 Preamble

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. *http://fsf.org/*

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

B.2 Terms and conditions

1. Applicability and definitions

   This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "**Document**", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "**you**". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

   A "**Modified Version**" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "**Secondary Section**" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "**Invariant Sections**" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "**Cover Texts**" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "**Transparent**" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "**Opaque**".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "**Title Page**" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "**publisher**" means any person or entity that distributes copies of the Document to the public.

A section "**Entitled XYZ**" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "**Acknowledgements**", "**Dedications**", "**Endorsements**", or "**History**".) To "**Preserve the Title**" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. Verbatim copying

   You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

   You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. Copying in quantity

   If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

   If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

   If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

   It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. Combining documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. Collections of documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. Aggregation with independent works

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. Translation

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. Termination

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. Future revisions of this license

    The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

    Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. Relicensing

    "Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

    "CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

    "Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

    An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

    The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

References

Baillie, M., Pilcher, J., 1973. A simple crossdating program for tree-ring research. Tree-ring Bulletin 33, 7–14.

Brewer, P., Murphy, D., Jansma, E., 2011. TRiCYCLE: a universal conversion tool for digital tree-ring data. Tree-Ring Research 67.

Brewer, P., Sturgeon, K., Madar, L., Manning, S., 2010. A new approach to dendrochronological data management. Dendrochronologia 28, 131–134.

Cook, E., Peters, K., 1981. The smoothing spline: A new approach to standardizing forest interior tree-ring width series for dendroclimatic studies. Tree-Ring Bulletin 41, 45–53.

Cormen, T., Leiserson, C., RL, R., C, S., 2001. Introduction to Algorithms. The MIT Press.

Huber, B., Fletcher, J., 1970. Dendrochronology in Europe.

Jansma, E., Brewer, P., Zandhuis, I., 2010. TRiDaS 1.1: The tree-ring data standard. Dendrochronologia 28, 99–130.

Reinsch, C., 1967. Smoothing by spline functions. Numerische Mathematik 10, 177–183.

Van Loan, C., 1999. Introduction to Scientific Computing: A Matrix-Vector Approach Using MATLAB. Prentice Hall.

Index