

Species 2000 technical manual:
Installation and maintenance of the
Species 2000 services

Peter W. Brewer

Last updated - 26 March 2007

Abstract

This document describes the installation and day-to-day running of the services provided by Species 2000 and attempts to draw together information (including cut and paste inclusions) from Ed Donovan, Richard White, Jorrit van Hertum and Peter Hollis. The Species 2000 services as concerns this document are: Dynamic Checklist; Annual Checklist (including archived versions); Websites; Metadatabase; mail lists and addresses. This document assumes the reader is a systems administrator with a good understanding of both Windows and Linux servers. An associated software repository DVD is available which includes all the software discussed in this document.

Contents

1	Installation	3
1.1	Introduction	3
1.1.1	Operating Systems and Servers	3
1.1.2	Server specifications	4
1.1.3	Dependencies	4
1.2	Installing the Dynamic Checklist	6
1.2.1	SPICE CAS	6
1.2.2	Installing a second instance of the CAS	7
1.2.3	MySQL cache database	7
1.2.4	PHP search interface	8
1.2.5	Installing Wrappers	9
1.2.6	Hierarchy Database	9
1.2.7	Installing SPICE Tester	9
1.3	Installing the Annual Checklist	9
1.3.1	MySQL database	10
1.3.2	PHP search interface	10
1.4	Installing the Annual Checklist Archives	10
1.5	Installing the Metadatabase	11
1.5.1	PostgreSQL database	11
1.5.2	PHP interface	12
1.6	Installing the Websites	13
1.7	Installing Web Statistics Software	13
1.8	Mail server	14
1.9	Security	14
1.10	Domains, DNS and Apache setup	15

2	Running the Species 2000 servers	17
2.1	Security updates	17
2.2	Dynamic Checklist	17
2.2.1	Starting SPICE	17
2.2.2	Caching data	17
2.2.3	Adding a new database	18
2.2.4	Troubleshooting	18
2.3	Annual Checklist	19
2.4	Websites	19
2.5	Backing up Databases	19

Chapter 1

Installation

1.1 Introduction

This chapter explains how to install the services provided by Species 2000 which are comprised of the following components:

1. Dynamic Checklist
 - SPICE Common Access System (CAS) software (W)
 - Dynamic Checklist database (cache database) (W)
 - Web search interface (L)
2. Annual Checklist
 - Annual Checklist database (L)
 - Web search interface (L)
3. Species 2000 and Species 2000 europa websites (L)
4. Metadatabase
 - Metadatabase database (L)
 - Web interface (L)
5. Archive of previous Annual Checklists
 - Annual databases (W)
 - Web search interfaces (W)
6. Mail server

All this software is available on the Species 2000 Software Repository DVD. The current version which relates to this document is version 1 - 15 May 2006.

1.1.1 Operating Systems and Servers

The Species 2000 services are platform independent with the exception of the SPICE CAS and the legacy Annual Checklists (2000-2004) which must be run on a Windows XP, NT or 2003 server.

All remaining services can be run on your platform of choice and most of them have been tested on Windows XP, NT and 2003 as well as SuSE and (K)Ubuntu distributions of Linux.

My personal preference is to use Linux wherever possible and so I have restricted the use of Windows to the essential elements. The University of Reading installation therefore uses two servers, one Windows and one Linux. The Windows server runs SPICE and the legacy Annual Checklists whilst the Linux server runs the remaining services (see table 1.1 for specification of the University of Reading installation). The separation of the services is shown in the list in section 1.1 where (W) means the element is run on the Windows server and (L) means it's run on the Linux (Kubuntu Dapper) server. For ease of documentation I will only explain how to set up each element on the platform indicated. In the case of the Linux server instructions are specifically for a Kubuntu Dapper server with 'Universe' apt repositories enabled. Feel free to experiment with a different platform but keep in mind you may run into problems not covered in these notes.

1.1.2 Server specifications

The hardware requirements are standard for web servers, however the SPICE CAS is quite memory hungry (in fact it is currently experiencing a memory leak requiring a periodic restart of the server) so it is best to install as much RAM as possible. Stress testing done as part of the Species 2000 europa project by Jorrit illustrated that bandwidth was the primary bottleneck of the system.

Table 1.1: Server specification for Reading installation

Hostname	APPC133	APLX5
OS	Windows XP	Kubuntu Dapper
Processor	AMD Athlon 64 - 3200+	AMD Athlon 64 X2 Dual Core 4200+
RAM	2Gb	4Gb
HD	120Gb SATA	2x 70Gb SATA RAID
Ethernet	NVIDIA Gigabit	NVIDIA Gigabit

1.1.3 Dependencies

The Species 2000 component services have a variety of software dependencies not all of which are compatible with each other (see tables 1.2 and 1.3). It is important that the correct versions of each dependency are used as problems have been found when using both older and newer versions. Where specific problems with different versions have been noted they are indicated in tables 1.2 and 1.3, however I suggest that you don't stray from the version indicated in the tables unless you have a very good reason. All the required software is open source with the exception of Borland Visibroker for which Species 2000 has a licence for use at the University of Reading.

Please note that the sequence in which these dependencies are installed is also important. For instance Visibroker must be installed after both Java and Apache Tomcat as some quirks in the installer cause problems. Continue through this manual step-by-step rather than installing all the dependencies first as there are a number of specific issues to consider.

Table 1.2: The dependencies of the Species 2000 services run on the Windows server

Dependency	Version	SPICE	AC	DC	Required for		
					MetaDB	Websites	Old AC
Java 2 Platform	J2SDK SE v1.4.2.10	✓					
Apache Tomcat	5.0 (not v5.5)	✓					
Apache Ant	1.6.5	✓					
Borland Visibroker	5.2 (not 5.1 or 6.2)	✓					
MySQL	4.1.21 (not newer)	✓					
Apache HTTP Server	1.3.35 (not 2.x)						✓
PHP	4.4.4 (not 5.x)						✓

Table 1.3: The dependencies of the Species 2000 services run on the Linux server

Dependency	Version	SPICE	AC	DC	Required for		
					MetaDB	Websites	Old AC
Apache HTTP Server	2.0.55		✓	✓	✓	✓	
MySQL	4.1.15		✓			✓	
PostgreSQL	8.1 (not 7.x)				✓		
PHP ¹	4.4.2 (not 5.x)		✓	✓		✓	
PHP	5.1.2 (not 4.x)				✓		
PHP - HTTP_Request	1.3.0		✓	✓			
PHP - Net Socket	1.0.6		✓	✓			
PHP - Net URL	1.0.14		✓	✓			
PHP - SOAP	0.9.4beta		✓	✓			
PHP - MySQL	4.4.2		✓				
PHP - PGSQL	??				✓		
Joomla	latest					✓	

1.2 Installing the Dynamic Checklist

The Species 2000 Dynamic Checklist service provides users with 'up-to-date' search results of the array of Species 2000 member databases. The core of the Dynamic Checklist is the SPICE CAS (Common Access System). For reasons of speed and reliability the SPICE CAS does not query the member databases on every request, but instead periodically downloads data to a MySQL cache database maintained on the Species 2000 server. A PHP search interface provides users with the ability to query the Dynamic Checklist in a number of ways.

1.2.1 SPICE CAS

As indicated in section 1.1.3, the SPICE CAS should be installed on a Windows server and requires: Java 2; Apache Tomcat; Apache Ant, Borland Visibroker; MySQL. Install the correct versions of these dependencies via the standard methods and ensure they are installed in this order. Once the dependencies have been installed the SPICE CAS itself can be installed by unzipping into the directory `c:\SPICE`. The following steps are then required to configure the CAS and its dependencies:

1. Set the `JAVA_HOME` environment variable to the path of your installation
 - `c:\j2sdk1.4.2_10`
2. Add the following to your `PATH` environment variable
 - Ant binary (probably `c:\apache-ant-1.6.5\bin`)
 - Visibroker binary (may have been done by installer)
3. Test that Visibroker has been successfully installed by running:
 - `osagent`
 - `nameserv`
4. Add the following files to your classpath:
 - `c:\BES\lib\vbjorb.jar`
 - `c:\BES\lib\vbsec.jar`
 - `c:\BES\lib\lm.jar`
 - `c:\SPICE\classes`
 - A separate entry for each of the jar files in `c:\SPICE\lib`
5. Build SPICE by navigating to `c:\SPICE\conf` and running:
 - `ant compileSPICE`
 - `ant buildSPICE`
6. Edit your `..\Tomcat 5.0\webapps\SPICE\index.htm` to ensure the webpage points to your server
7. In Control Panel > Administrative Tools > Services it is a good idea to ensure Apache Tomcat and MySQL are started automatically when the server is turned on.

There is currently a problem of stability some where between the SPICE CAS, Corba and Tomcat. The system works fine for approximately two days on the current Reading installation but the Tomcat crashes. As a temporary workaround to this it makes sense to set up scheduled jobs on the server to shutdown and then restart Tomcat every 24 hours. This can be done by adding entries for `shutdown.bat` and `startup.bat` from the Tomcat bin folder to the task scheduler in Windows. This is accessible from Programs & Accessories & System Tools & Scheduled Tasks. The startup should be scheduled for 1 minute after the shutdown, otherwise they interfere with each other and the server doesn't get restarted.

1.2.2 Installing a second instance of the CAS

It is likely that you will want to run two instances of the SPICE CAS - a live and test service. Perhaps the simplest solution for this is to use a second server to run the test service. In this case you would simply follow the instructions in section 1.2.1 again. The tidier way of doing this, however, is to install a second version to another directory such as `c:\SPICE_TEST`. When SPICE is installed to a folder other than `c:\SPICE` you will need to make the following changes:

1. In the `..\conf\build.xml` change:

- Line 1 - set the correct basedir
- Lines 11 to 18 - set the correct paths

2. In the `..\conf\spice.xml` change:

- Line 430 - `<spicecontext home="c:/SPICE/conf/">` to match the location of your conf directory
- Lines 440 to 443 - change the various URL's to point to your server
- Lines 495 to 500 - change NameingService, CASServer, CASServerMonitor and CachingService values
- Lines 512 to 514 - change `<logger name>` entries to match the location of your logs directory

3. In the `..\conf\web.xml` change:

- Line 12 - `<display-name>`
- Line 51 - `<param-value>`
- Line 59 - `<param-value>`

Once you've made these edits remember to compile and build SPICE using the Ant commands outlined in section 1.2.1.

1.2.3 MySQL cache database

As mentioned before the SPICE CAS stores its results in a local MySQL cache database. In the previous section you installed MySQL itself but before the SPICE CAS can be used, a database user and the database itself must be created. For security it is also important to set a password on the root user account. Database users can be created by opening a MySQL command shell as root (`mysql --user=root mysql`) and running the following commands²:

```
GRANT ALL PRIVILEGES ON *.* TO enteruserhere@localhost
IDENTIFIED BY 'enterpasswordhere' WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON *.* TO enteruserhere @"%"
IDENTIFIED BY 'enterpasswordhere' WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON *.* TO root@localhost
IDENTIFIED BY 'enterpasswordhere' WITH GRANT OPTION;
```

It is also a good idea to set up a user specifically for SPICE which can only connect from the localhost for security reasons. This can be done using the following commands:

²Please note that these are very minimal instructions of setting permissions for MySQL users. See the MySQL documentation for the latest advice on maintaining a secure installation.

```
GRANT ALL PRIVILEGES ON *.* TO spice@localhost
IDENTIFIED BY 'spicey' WITH GRANT OPTION;
```

The SPICE cache database can then be created by running the SQL commands in the createCacheDatabase.sql file. For instance in Windows run the command:

```
mysql --u=yourusername -p < c:\SPICE\conf\createCacheDatabase.sql
```

You now need to point your SPICE CAS installation at this newly created database:

1. Edit your `c:\SPICE\conf\spice.xml` file and alter the database URL, user name and password attributes to match
2. Restart the SPICE CAS

1.2.4 PHP search interface

As indicated in section 1.1.3, the Dynamic Checklist PHP search interface requires Apache HTTP Server and PHP to be installed along with the following PHP packages: HTTP_Request; Net_Socket; SOAP. I prefer to install the PHP package manager PEAR as well as it is useful for installing the PHP packages. As there is no requirement for the PHP search interface to be on the same server as the SPICE CAS, I recommend installing the interfaces on a Linux server like so:

1. `sudo apt-get install apache2 php4 php4-pear`
2. If you are behind an authenticated proxy (as most machines are at Reading) you will need to set the proxy settings in PEAR before using it to download packages. This can be done by typing:
 - `pear config-set http_proxy http://usr:pwd@wwwcache.rdg.ac.uk:8080`
3. Then use PEAR to install the required packages
 - `sudo pear install Net_Socket`
 - `sudo pear install Net_URL`
 - `sudo pear install HTTP_Request`
 - `sudo pear install SOAP-beta`
4. Check the packages have been installed correctly by running `pear list`

The next stage is to install the Dynamic Checklist interface by extracting it into the Apache HTTP Server htdocs folder. Within the `session.php` file a number of default values may be altered:

site_version - "standard" or "basic". The latter is a version for users with motor or vision disabilities, as well as users who do not use JavaScript. Default is normally set to "standard"

use_cache - true or false. This determines whether cached data is requested from the web service (the faster option) or whether data is directly requested from the source databases (the slower option). Default is normally set to true

use_test_hub - true or false. This determines whether data is requested from the test hub of the web service or the production hub of the web service. The default value is normally false.

number_of_records_shown_per_page : This determines the number of records shown per page in the search results. The default value is normally 10.

Finally you need to set the URL of your new CAS installation on lines 4 and 7 of `CASWebservice.php` so that the PHP interface is querying the correct version of the CAS.

1.2.5 Installing Wrappers

The SPICE CAS is the central hub of a distributed databases system. It is designed to talk to many remote databases which are wrapped to provide a uniform interface to the SPICE CAS. In an ideal world all of these wrappers would be located on the same remote server as the databases but for various reasons a number of these wrappers are still located with the Secretariat. Whilst it is hoped that the wrappers will gradually be moved to the control and maintenance of member databases, there is one database and wrapper that is run centrally by Species 2000 and that is the hierarchy database.

All the wrappers maintained at the Secretariat are written in Java and can be installed as follows:

1.2.6 Hierarchy Database

The hierarchy database contains the top levels of the hierarchy on to which the GSD's are added. The complete Catalogue of Life hierarchy is therefore derived from the hierarchy database at the top 'branches' and then from the GSD's towards the 'twigs'. The level at which the transition is made depends of the level of detail provided by the GSD. For example in some cases this may be at the level of phylum and in others at family level. In most respects the hierarchy database works just like another GSD in that it is a separate wrapped database which is queried using the same requests as other GSD's. It is however specified slightly differently in the SPICE CAS configuration file.

The hierarchy database itself is a MySQL database which can be rebuilt using the SQL commands file found in the Species 2000 repository. The wrapper itself can be installed as follows:

- Extract the wrapper into a suitable folder (e.g. c:wrappers).
- Locate catalina-ant.jar in your Tomcat installation and copy it to your ant lib folder.
- Add this new jar file location to your classpath.
- In the hierarchy wrapper folder run `ant`.

1.2.7 Installing SPICE Tester

Within the SPICE CAS package is an automated health status checker. This runs at a specified time each day and sends emails to administrators when it finds problems with the SPICE CAS or any of the wrappers. My current installation of this tool does not appear to work - investigations are ongoing.

1.3 Installing the Annual Checklist

The Species 2000 Annual Checklist service provides users with static annual snapshots of the Species 2000 member databases. Currently, the Annual Checklist is a distinct service from the Dynamic Checklist, compiled in a different way and from a different array of member databases. Each year the Annual Checklist is compiled manually from database dumps provided by contributing databases. It is hoped that this process can be phased out in the future and replaced by a snapshot of the Dynamic Checklist. This however can only happen once the majority of the databases contributing to the Annual Checklist are accessible through the Dynamic Checklist.

Like the Dynamic Checklist, the Annual Checklist is maintained in a MySQL database and queried by a PHP search interface. The Annual Checklist interface offers a wider number of search tools than the Dynamic Checklist.

1.3.1 MySQL database

Install MySQL using the standard installation procedure e.g. in Kubuntu:

```
sudo apt-get install mysql-server-4.1
```

Once installed copy the Annual Checklist database folder into the relevant location on your hard disk (typically /var/lib/mysql/ in Linux).

As with the Dynamic Checklist MySQL installation you will need to set up MySQL users with privileges to read the database and add a password to the root user account. This can be done using the following commands:

```
GRANT ALL PRIVILEGES ON *.* TO enteruserhere@localhost
IDENTIFIED BY 'enterpasswordhere' WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON *.* TO enteruserhere @%"
IDENTIFIED BY 'enterpasswordhere' WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON *.* TO root@localhost
IDENTIFIED BY 'enterpasswordhere' WITH GRANT OPTION;
```

1.3.2 PHP search interface

The PHP search interface is again dependent on Apache Web server and PHP as well as the MySQL PHP module. If you haven't already set them up for the Dynamic Checklist interface they can be installed in Kubuntu as follows:

```
sudo apt-get install apache2 php4 php4-mysql
```

The interface folders should be placed in the relevant web accessible folder (typically /var/www/). Please note that whilst the 2005 and 2006 interfaces look identical, changes were made between editions so there are specific interfaces for each version.

1.4 Installing the Annual Checklist Archives

The current Annual Checklist system was introduced in 2005 so subsequent versions are installed and used in the same way as described in section 1.3. Prior to this the Annual Checklist were stored in Microsoft Access databases and accessed through a PHP system designed by Eli. These checklists must therefore be run on a Windows server. This section describes the steps required to install the Annual Checklist versions 2000, 2002, 2003 and 2004.

First install Apache HTTP Server using the standard installer. I have had problems with the PHP installation program in Windows so prefer to use the manual installation method. It is better to set up PHP using SAPI modules rather than the CGI executable as the performance is better and SAPI has proven to be reliable:

1. Extract the contents of the PHP zip file to c:\PHP
2. Copy c:\PHP\php4ts.dll to c:\windows
3. Copy c:\PHP\php.ini-dist to c:\windows\php.ini
4. Edit the c:\windows\php.ini file and change the following entries:

- `max_execution_time = 300`
 - `include_path = ".;c:\php\pear"`
5. Configure Apache by editing `..apache\conf\httpd.conf` and adding:
 - `LoadModule php4_module c:/php/sapi/php4apache.dll`
 - `AddModule mod_php4.c`
 - `AddType application/x-httpd-php .php`
 6. Restart Apache using the commands `net stop apache` and `net start apache`

Once Apache and PHP are set up you'll then need to set up the Annual Checklist databases and interfaces:

1. Copy the Access `.mdb` files for each of the four checklists into a suitable directory on your Windows server.
2. In Control Panel > Administrative Tools > Data Sources (ODBC) select the System DSN tab and press add. Select the Microsoft Access driver, name the datasource (sp2000, sp2002, sp2003, sp2004) and press the select button to locate the Access database file.
3. Copy the PHP interface folders for each checklist to your Apache htdocs folder
4. Repeat for each version.
5. Edit your `php.ini` file and set `register_globals = On`
6. See section 1.9 about techniques for securing the server.

1.5 Installing the Metadatabase

The Species 2000 Metadatabase stores information regarding the current member databases as well as other global and regional databases approached with regards joining the Species 2000 project. The database is stored in PostgreSQL and is queried through a PHP web interface. The installation of the Metadatabase is described below.

1.5.1 PostgreSQL database

Install PostgreSQL and the PHP PostgreSQL library on your Linux server using apt commands below. The Metadatabase was developed using v8.1 of Postgres so it is advisable to use this version but certainly not to use v7.x. You may like to install `pgadmin3` as it can make maintaining the database simpler. The metadatabase interface does a lot of XML processing that requires PHP5. Making v4 and v5 of PHP run on the same server is non-trivial so it is best to run the Metadatabase on a separate server to the main checklist interfaces. Installation is done as follows:

```
sudo apt-get install postgresql-8.1 php5-pgsql pgadmin3
```

Once you have installed PostgreSQL you need to set up two users, an administrator and a webuser. You can do this using `pgadmin` or log in to a standard shell as the default 'postgres' user and issue the following commands:

```
createuser yourusername -EdaP
createuser webuser -EP
```

You now need to create your new Metadatabase and then populate it from the latest SQL dump file. This can be done from pgadmin or in a standard shell using the following commands:

```
createdb metadb
psql metadb < sp2kmetadbdump.sql
```

1.5.2 PHP interface

The PHP interface to the Metadatabase is in two parts: a PHP database library and the interface itself. The PHP interface itself should be extracted from its archive and placed in a relevant web accessible folder on your server. An entry must then be added to your Apache conf file to ensure users can access your new Metadatabase.

The database library should be extracted from its archive and placed in a folder (I suggest /usr/local/php-db-gui-lib) with the inc subfolder specified as a PHP include folder. This is done by editing /etc/php4/apache2/php.ini, uncommenting the include_path and adding the /usr/local/php-db-gui-lib/inc folder to the list.

The database library also requires a number of aliases to be set up in the relevant sites-available file (/etc/apache2/sites-available)

```
<IfModule mod_alias.c>
    Alias /css/ /usr/local/php-db-gui-lib/css/

    <Directory /usr/local/php-db-gui-lib/css>
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    Alias /jscript/ /usr/local/php-db-gui-lib/jscript/

    <Directory /usr/local/php-db-gui-lib/jscript>
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    Alias /images/ /usr/local/php-db-gui-lib/images/

    <Directory /usr/local/php-db-gui-lib/images>
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>
</IfModule>
```

There is an undocumented feature/bug in PHP4 associated with sessions, therefore default installations of PHP5 print a warning message which needs to be switched off as this 'feature' is not used in the Metadatabase. In the php.ini file set session.bug.compat_warn=0.

The most up-to-date apache configuration files can be found on the Species 2000 repository DVD.

1.6 Installing the Websites

There are two main websites run by Species 2000 - www.sp2000.org and www.sp2000europa.org. Both are run using the Joomla Content Management System (CMS). Joomla is a PHP based system that uses a MySQL database to store the contents of the website. Installation on a Linux server is fairly straight forward assuming you already have Apache, MySQL and PHP installed:

1. Unzip latest version of Joomla (www.joomla.org) into a web accessible folder
2. Copy the database into the relevant MySQL folder (/var/lib/mysql/ in Kubuntu)
3. Set user permissions on the MySQL database for the user specified in the Joomla admin pages
4. Unzip the Species 2000 template file into the templates subdirectory
5. Unzip the configuration.php into the root of your installation

There is also a service checker webpage that is separate to the Joomla installation. This should be unzipped into the folder /var/www/servicechecker. The menu links to this page should already be set up in the www.sp2000.org Joomla site.

1.7 Installing Web Statistics Software

The Species 2000 websites and services are monitored using the awstats web stats software. Installation on the Linux server can be done using apt: `sudo apt-get install awstats`. File locations for awstats in Kubuntu varies from that noted in the documentation:

Apache log files - normally located in /var/log/apache2. The location is specified in the apache.conf file.

awstats configuration files - located in /etc/awstats. There is a separate file for each service that is to be monitored. The file names must take the form awstats.websitename.conf.

awstats statistics files - normally located in /var/lib/awstats but can be specified in the awstats configuration files.

Once awstats has been installed copy the Species 2000 awstats configuration files from the software repository into /etc/awstats. The statistics should be compiled on a regular basis by setting up cron jobs. Open a root shell by typing `sudo su -`, then edit the crontab by typing `crontab -e`. Add an entry for each website in the form:

```
0,15,30,45 * * * * /usr/lib/cgi-bin/awstats.pl -config=websitename -update
```

The numbers at the beginning refer to what times of the hour to run the job, in this case each quarter hour.

1.8 Mail server

In addition to providing sp2000.org email aliases for the directors and Secretariat, Species 2000 also hosts a total of twelve mailing lists:

- directors@sp2000.org
- isg@sp2000.org
- members@sp2000.org
- taxonomy@@sp2000.org
- team@@sp2000.org
- europasg@sp2000.org
- wp1@sp2000.org
- wp2@sp2000.org
- wp3@sp2000.org
- wp4@sp2000.org
- wp5@sp2000.org
- wp6@sp2000.org

The lists are currently run by University of Reading IT Services. Current members of each list can be obtained from the Species 2000 software repository. The lists are run through a default Mailman installation with the following options altered:

- Reply_goes_to_list - This list
- Require_explicit_destination - No
- Generic_nonmember_action - Accept (for ISG, Taxonomy, Team) Hold (for remainder)
- Archive_messages - Yes
- Max_message_size - 0 (unlimited)

1.9 Security

It is wise to install firewall software on both Linux and Windows servers before opening them up to the internet. My personal choices are Zone Alarm for Windows (<http://www.zonelabs.com>) and Firestarter for Linux. Firestarter is in apt so can be installed using the following command `sudo apt-get install firestarter`. Ensure you open port 80 on the Linux server and ports 80 and 8080 on the Windows server.

There are a number of additional measures you can take to tighten up security on Apache by editing the apache.conf file:

- Keep the version of Apache and PHP secret by adding:

```
ServersSignature Off
ServerTokens Prod
```

- Disable track and trace to stop cross site scripting attacks by adding:

```
<IfModule mod_rewrite.c>
RewriteEngine on
RewriteCond %{REQUEST_METHOD} ^ (TRACE\|TRACK) }
RewriteRule .* - [F]
</IfModule>
```

- Disable user homepages by adding

```
<IfModule mod_userdir.c>
UserDir disabled
UserDir public_html
</IfModule>
```

To ensure the above code all works you will need to make sure that the rewrite and userdir modules are loaded. This can be done on Debian based systems like Kubuntu by typing `a2enmod` and then select the relevant modules.

1.10 Domains, DNS and Apache setup

Currently there are a number of Species 2000 domains registered with various companies, namely:

- sp2000.org - with Network Solutions
- sp2000europa.org - with Network Solutions
- catalogueoflife.org - with Freeola.com
- species2000.org - with Network Solutions
- catalogueoflife.com - with Freeola.com

The usernames and passwords for accessing these accounts are on file at the Secretariat along with details of renewal dates. The registered owner of all these domains is currently Species 2000 at the University of Reading. The first three domains are considered as the 'correct' domains with the remaining two registered to catch URL variations. The DNS for the Network Solution registered domains is handled through the Network Solutions web interface. Unfortunately, Freeola does not provide such a service, so the Freeola registered domains are pointed at a free DNS run by XNAME (<http://www.xname.org>). It would probably be sensible to consolidate all the domains at Network Solutions at some point in the future.

As Species 2000 provides a number of services to its users which are identified using subdomains. The current list of subdomains is as follows:

metadb.sp2000.org - Metadatabase v2

www.sp2000.org - original Species 2000 website

www.sp2000europa.org - original Species 2000 europa website

www.catalogueoflife.org - Google style webpage which searches the product we currently regard as 'best'

documents.sp2000.org - Document repository site

webservice.sp2000.org:8080 - SPICE webservice

In addition there are a number of legacy subdomains and webpages that are retained for backward compatibility. These are:

spice.sp2000.org - redirects to `dynamic.sp2000.org`

www.sp2000.org/AnnualChecklist.htm - redirects to `annual.sp2000.org`

spice.sp2000europa.org - SPICE configuration interface

metadb.sp2000europa.org - Metadatabase v1

www.usa.sp2000.org - Defunct?

admin.sp2000.org - ?

logs.sp2000.org - ?

spice-test.sp2000.org - ?

test.sp2000.org - ?

test.sp2000europa.org - ?

There are also domains that are not controlled by the Species 2000 Secretariat:

www.sp2000.nies.go.jp

www.sp2000.cn

With the exception of the webservice, `www.usa.sp2000.org` and externally controlled domains, all the remaining domains should be pointed at the Linux server. The most recent apache configuration files which handles these subdomains can be found on the Species 2000 repository DVD. Because the Annual Checklists are run on two different servers (2005-6 on Linux and 2000-2004 on Windows) we need to set up `mod_proxy` so that requests to the `catalogueoflife.org` domain can be passed on to the correct server. All the required alias and proxy entries are in the apache configuration files stored in the repository but the `mod_proxy` module must be installed as follows before these will work:

- On the Linux server install `mod_proxy` using `apt`:
`sudo apt-get install libapache2-mod-proxy-html`
- Enable the modules using `sudo a2enmod`
- Enable the modules `proxy proxy_html proxy_http`
- Copy the `proxy.conf` from the repository to the `mods-available` folder
- Restart apache `sudo /etc/init.d/apache2 restart`

Chapter 2

Running the Species 2000 servers

2.1 Security updates

It goes without saying that the most important maintenance issue for the servers is to ensure they remain secure. The Windows server should of course have its virus definitions regularly updated as well as the standard Windows updates. On the Linux server updates can be done manually by running the Adept Manager or they can be automated by adding apt update commands to the crontab. This can be done by opening a root shell by typing `sudo su -`, then editing the crontab by typing `crontab -e`. Finally you should then add an entry for each command like so:

```
5 * * * * command_to_run
```

The first five columns of the line control when to execute the command and represent minute, hour, day of month, month and day of week respectively. In the example above the command would be run at 5 minutes past every hour of every day.

2.2 Dynamic Checklist

2.2.1 Starting SPICE

Assuming installation has gone smoothly the SPICE CAS can be started by running the command `c:\spice\bat\startSpiceApp.bat`. You may find it helpful to set up shortcuts to this command (both test and live versions) on your desktop for easy access.

2.2.2 Caching data

The primary purpose of the SPICE CAS is to cache data from the member databases. The SPICE CAS differentiates between caching species data and hierarchy data, each of which are instigated by separate entries in the SPICE GUI caching menu. Either specific GSD's or all GSD's can be updated using the relevant menu items. Support for offline and sluggish servers is currently fairly minimal so it is normally best to update GSD's one at a time to minimise the risk of crashing the server. The progress of a species cache can be monitored by viewing the data in the cachestatus table of the spicecache database.

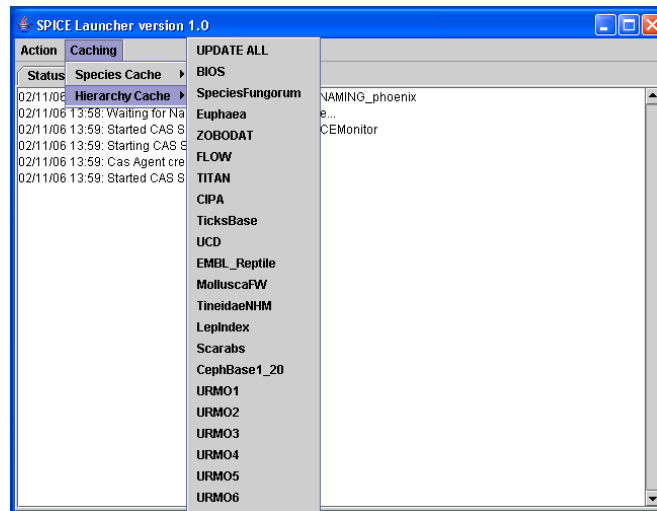


Figure 2.1: Screen shot of the SPICE CAS interface

2.2.3 Adding a new database

There are a number of distinct stages in adding a new database to the Dynamic Checklist:

1. Liaise with the wrapper writer to ensure wrapper complies to the SPICE XML Schema
2. Add an entry for the database to the Metadatabase and include details about the wrapper on the wrapper page.
3. Run technical tests on the wrapper from the wrapper page of the metadatabase
4. Once all request tests are passed, add an entry to the wrapperconfig.xml file on the test hub
5. Cache the species data
6. The content manager then tests the quality and validity of the data returned
7. Once biological tests have been passed the wrapper can then be added to the live hub and the species and hierarchy data cached

2.2.4 Troubleshooting

There are a large number of unhandled exceptions in the SPICE CAS software. As such, it is a regular occurrence to see large numbers of error messages in the logs. As a systems manager it is probably simpler to familiarise yourself with the correct functioning of the CAS so you can recognise the difference between an error, a bug and a missing feature.

At the time of writing the most significant bug in the SPICE CAS is a memory leak that causes the software to periodically crash. On the Thompson installation this happens approximately every fortnight, but on the Reading installation it is more frequent (every few days). The symptom of a CAS crash is all queries to the Dynamic Checklist report 'species not found', even when searching for a*. The service checker on the Species 2000 website will report the Dynamic Checklist as being online but returning unexpected data. In such a case, the CAS instances should be shut down, Tomcat should be restarted and then CAS should be restarted.

When examining further problems with the CAS, troubleshooting starts by viewing the array of log files created by the server. These are found at `c:\spice\logs`. It is often helpful to run the

MySQL Administrator at the same time as the CAS when debugging problems so that you can manual search for the same data as the CAS.

2.3 Annual Checklist

The Annual Checklist software appears to be completely stable and as such requires no maintenance.

2.4 Websites

Whilst the websites are not the most insecure of services run by Species 2000, they are probably the most prone to attack due to the use of a content management system. Content management systems make running websites quicker and easier but if they are not updated regularly they are easy targets for hackers, as was shown when the Species 2000 europa site was compromised in September 2006. As Joomla is not installed via the apt repository you must manually check for updates using the option in the administrative interface.

2.5 Backing up Databases

All databases, with the exclusion of the Annual Checklists, should be backed up regularly in case of server failure. Whilst database backups are normally automated, in my opinion this is not necessary for the SPICE cache as you will be aware when the database has changed. Regular backups will result solely in large quantities of identical backups. The MySQL Administrator GUI is the simplest way of making backups on the Windows server.

On the Linux server, regular backups should be made of the Joomla MySQL website databases and also the PostgreSQL metadatabase. As these are likely to change more frequently it makes sense to automate these as a cron job.