SourceForge 3.4 Installation and System Administration Guide

Installation and System Administration Guide
SourceForge 3.4 Enterprise Edition
Part Number: 98-0020
Last Revision: September 2003

Restricted Rights

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION CONTAINED HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. VA SOFTWARE CORPORATION MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR PROGRAM(S) AND/OR SERVICE(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.

SourceForge is a trademark or registered trademark of VA Software Corporation in the United States and other countries. Linux is a registered trademark of Linus Torvalds. All other trademarks and product names are property of their respective holders.

Feedback

Please send comments and questions to:

Technical Publications
VA Software Corporation
47071 Bayside Parkway
Fremont, CA 94538 U.S.A.

Technical Support

You can contact the Technical Support department at VA Software for assistance with using SourceForge, filing bug reports, and feature requests.

E-mail: *support-sf@vasoftware.co*m
Knowledge Base: *https://kb.vasoftware.com*
Phone: *1-877-825-4689*, option 2

Contents

Preface

This guide provides instructions for installing and configuring SourceForge Enterprise Edition 3.4.

Overview of Contents

This guide includes the following chapters:

- Chapter 1: Introduction
- Chapter 2: Pre-Installation Tasks
- Chapter 3: Installing SourceForge
- Chapter 4: System Maintenance

Related Documentation

- *SourceForge Enterprise Edition 3.4 User Guide*
- *SourceForge Enterprise Edition 3.4 Release Notes*

Conventions Used

The following typefaces show specific typeface elements used in this guide:

Convention	Usage
Italic	File names, file paths, and directory names (*C:\Applications*) URLs and protocol names (*http://www.vasoftware.com*), and references to other manuals
`Fixed Width`	Literals, Code examples (<`code`>), and displayed text.
`Fixed Width Bold`	Text you enter.

CHAPTER 1

Introduction

This chapter includes:

- Overview
- Software Requirements
- Hardware Profile

Overview

The SourceForge installation environment is divided into “roles”: SourceForge (SFEE) server, Backend server, SCM server, LDAP server, and List server each with distinct functionality described as follows:

Roles Defined

1. SourceForge (SFEE) server
    - Contains the SourceForge codebase.
    - Creates users and projects.
    - Processes the main user web pages of the SourceForge system and is considered the main component of all SourceForge activity.
2. Backend (Log) server
    - Defines the database.
    - Calculates usage statistics and stores it in the database.
    - Collects usage information via the Log server.
    - Includes mass mailing initiation information.
3. SCM server
    - Contains the Source Control Management system.
    - Stores and processes SCM data.

      The SCM application has an independent repository for code storage and management. The standard SourceForge installation uses CVS.
4. LDAP server
    - This role contains the LDAP server - slapd. The LDAP server is used for authenticating users for the SCM role.
5. List server
    - Contains the mailing list management system.

      The standard SourceForge installation uses “mailman.”
    - Processes all mailing list management features in SourceForge.
    - Creates mailing lists.

Database (DB)

The SourceForge database processes data storage for the Mailing List, Forums, Tracker, Document Manager, and File Publisher systems. The database also stores user authentication information for the SourceForge front-end server.

Installation Process Defined

The installation process consists of the following steps:

1. Installing **one** of the following operating systems*:
   - RedHat Linux 7.2 or Advanced Server 2.1
   - SUN Solaris 9
2. Installing a database
   - Populate the database with baseline data
3. Installing SourceForge using the SourceForge Installation Wizard
4. Verifying the installation

***Note:** This manual provides detailed installation instructions for operating systems, databases, and SCM systems on which SourceForge has been fully tested. SourceForge 3.4 runs on a number of other configurations that have been confidence tested, but for which detailed installation instructions are not provided in this manual.

Please reference "Software Requirements" on page 4 for a complete list of full and confidence tested configurations. If you need assistance installing or configuring a confidence tested operating system, database, or SCM system, please contact VA Software Technical Support.

Software Requirements

This section lists the versions of the operating systems and databases on which SourceForge 3.4 has been tested.

Fully Tested Configurations

SourceForge 3.4 has been fully tested on all configurations in the following table. Detailed installation instructions are provided in this manual for all fully tested configurations.

Database	Database Operating System	SourceForge Operating System
IBM DB2 8.1	RedHat 7.2 with kernel version 2.4.20	RedHat Advanced Server 2.1*
PostgreSQL 7.3.3	RedHat 7.2 with kernel version 2.4.20	RedHat Advanced Server 2.1*
Oracle 9iR2 Enterprise Edition (9.2.0.3.0)	RedHat Advanced Server 2.1*	RedHat Advanced Server 2.1*
Oracle 9iR2 Enterprise Edition (9.2.0.3.0)	Solaris 9 (patch set 052803)	Solaris 9 (patch set 052803)

Confidence Tested Configurations

SourceForge 3.4 has been confidence tested on all configurations in the following table. If you need assistance installing or configuring a confidence tested operating system or database, please contact VA Software Technical Support...

Database	Database Operating System	SourceForge Operating System
PostgreSQL 7.3.3	Solaris 9 (patch set 052803)	Solaris 9 (patch set 052803)
PostgreSQL 7.3.3	RedHat 7.2 with kernel version 2.4.20	RedHat 7.2 with kernel version 2.4.20
Oracle 9iR2 Enterprise Edition (9.2.0.3.0)	RedHat 7.1**	RedHat 7.2 with kernel version 2.4.20
Oracle 9iR2 Enterprise Edition (9.2.0.3.0)	Windows 2000	RedHat Advanced Server 2.1*
Oracle 9iR1 Standard Edition (9.2.0.3.0)	RedHat 7.1**	RedHat 7.2 with kernel version 2.4.20
PostgreSQL 7.3.3	RedHat 7.3**	RedHat 7.3**
Oracle 9iR2 Enterprise Edition (9.2.0.3.0)	Solaris 9 (patch set 052803)	Solaris 8 (patch set 052803)

*Patched with kernel 2.4.9-e.31, glibc-2.2.4-31.7, fileutils-4.1-10, pam-0.75-46

**Please see the *SourceForge Enterprise Edition 3.4 Release Notes* for the RedHat 7.1 and 7.3 kernels.

Additional Required Software

In addition to one of the Database / Operating System configurations specified on the previous page, you will also need the following additional software.

Secure Socket Layer (SSL)

- A secure socket layer (SSL) certificate

Software Configuration Management (SCM) Application

- CVS 1.11.5(installed with SourceForge)
- Rational ClearCase 4.2 or 5.0 on Solaris
- Merant PVCS Version Manager 6.8.10 on Solaris

Web Browser

- Mozilla 1.3
- Internet Explorer 5.5 or 6.0
- Netscape 7.0
- Netscape 4.79 (confidence tested only)

Additional Optional Software

Optionally, you may also add the following additional software. Installation and configuration instructions for this optional software is included in the *SourceForge Enterprise Edition 3.4 User Guide*.

Integrated Development Environment (IDE)

IBM WebSphere Studio Application Developer 4.02 or 5.0

Lightweight Directory Access Protocol (LDAP) Systems

- OpenLDAP 2.0.27 (Linux)
- Netscape Directory Server 6.11 (Unix)
- Microsoft Active Directory (Windows 2000 Advanced Server)

Hardware Profiles

All hardware on which you plan to run SourceForge must be RedHat Linux or SUN Solaris compatible. This will reduce the risk of operating system installation issues during the installation of your SourceForge deployment.

For a complete list of RedHat compatible hardware, visit:

http://hardware.redhat.com/hcl/

For SUN server products, visit:

http://www.sun.com/servers/index.html

This section describes three possible system hardware configurations for SourceForge 3.4. These systems are used in various sample SourceForge configurations shown below.

Sample Configurations

The following configurations represent possible SourceForge 3.4 configurations across one or two machines.

Note: You may configure up to five machines with any of the SourceForge roles (SFEE, List, Backend, SCM, and LDAP), depending on your usage patterns and available hardware. Additional configurations using an external SCM (ClearCase or PVCS) require one SCM server and one database.

➪ For detailed descriptions of each SourceForge role, see “Roles Defined” on page 2.

One-Machine Configuration

Table 1. One Machine Configuration

System	System Roles
IBM xSeries 345	SFEE
Xeon 2 GHz Processor	List
2 x 512 MB DDR PC2100 ECC DIMMS	Backend
1 x IBM ServeRAID-5i Controller	SCM
4 x IBM 18.2 GB 10K-rpm Ultra160 SCSI Hot-Swap SL HDD	LDAP
1 x IBM xSeries 350 W Hot-Swap Power Supply Upgrade	

Two-Machine Configuration

Table 2. Two-Machine Configuration

System	System Roles
Machine One	
IBM xSeries 345	SFEE Backend
Xeon 2 GHz Processor	
2 x 512 MB DDR PC2100 ECC DIMMS	
1 x IBM ServeRAID-5i Controller	
4 x IBM 18.2 GB 10K-rpm Ultra160 SCSI Hot-Swap SL HDD	
1 x IBM xSeries 350 W Hot-Swap Power Supply Upgrade	
Machine Two	
IBM xSeries 345	Lists SCM LDAP
Xeon 2 GHz Processor	
2 x 512 MB DDR PC2100 ECC DIMMS	
1 x IBM ServeRAID-5i Controller	
4 x IBM 18.2 GB 10K-rpm Ultra160 SCSI Hot-Swap SL HDD	
1 x IBM xSeries 350 W Hot-Swap Power Supply Upgrade	

Note: For five-box installation each machine should have one system role. The system setup should be the same for all the machines.

CHAPTER 2
Preinstallation Tasks

It is necessary to load the necessary operating system (OS) and software, and test the hardware on the network prior to installing SourceForge.

This chapter describes the preinstallation tasks you must perform prior to installing SourceForge 3.4. It includes information on:

- Installing the Operating System
    - Red Hat Linux 7.2 or Advanced Server 2.1
    - SUN Solaris 9

  You need to install only one of these operating systems.
- Preparing the Database
    - Oracle 9iR2 Enterprise Edition (9.2.0.3.0)
    - PostgreSQL 7.3.3
    - IBM DB2 8.1

  You need to install only one of these databases.
- Installing the Software Configuration Management (SCM) client
    - Rational ClearCase 4.2 or 5.0
    - Merant PVCS 6.8.10
    - CVS 1.11.5

Installing the Operating System

You can install **one** of the following operating systems to work with SourceForge:

- SUN Solaris 9
- RedHat Linux 7.2 for SFEE system roles
- RedHat Advanced Server 2.1 for Oracle database host

Note: Do not add or create any additional user accounts either during the operating system installation or at the command line. Creating additional users has been shown to cause problems with SourceForge.

RedHat Linux 7.2 or Advanced Server 2.1

This section describes how to install RedHat Linux 7.2. Installing Advanced Server is a similar process.

RedHat Linux 7.2 is installed on the same server as SourceForge 3.4. Refer to the *RedHat Linux 7.2 x86 Installation Guide* for complete installation instructions.

For most installations, you will accept the defaults at each screen except where specific instructions are given here.

To install RedHat Linux 7.2

1. Follow the installation instructions shown on the screen and described in the *RedHat Linux 7.2 x86 Installation Guide.*
2. At the Install Options screen, choose Custom as the Install Type and click Next.
3. At the Firewall Configuration screen, click the Customize check box and select the following:
   a. Click the check boxes next to the following services to allow them to come through the firewall:
      ```
      DHCP
      SSH
      WWW
      Mail
      ```
   b. Enter ports 389, 443, 873, 5000, 17023, and 17024 in the Other ports field. Multiple ports are separated by commas.
      ```
      81, 389, 443, 873, 5000, 17023, 17024
      ```
      Note: Do not block ports 22, 25, or 80.
4. At the Package Group Selection screen, select Networked Workstation and deselect everything else. For DB2, Gnome and XFree86 need to be selected.

   Note: Gnome is required only for the GUI version of DB2.

5. Continue through the rest of the installation.

After installation is complete, you must set a hostname for the server in */etc/hostname*.

After you have completed the installation steps and set the hostname, you will have a system on which SourceForge 3.4 can be correctly installed.

Updating RedHat Security Packages

RedHat periodically releases security update packages. Some of these packages (RPMs) should not be installed because of potential conflicts with your SourceForge installation. These RPMs are listed below. All other RPMs can be installed safely.

up2date

RedHat includes a utility called up2date, which allows administrators to automatically download and upgrade the latest versions of installed software.

Installing SourceForge on any version of RedHat breaks this functionality, as SourceForge symlinks */usr/bin/python* to the SourceForge python2 binary. (For up2date to function correctly, the RedHat-installed python1.5 binary must be installed at */usr/bin/python*.)

Packages when Updating SourceForge Servers

The following RPMs are installed with SourceForge 3.4 and must not be updated.

- XFree86-libs
- cvs
- patch
- rcs
- rsync
- zlib
- sendmail
- sendmail-cf
- openssh
- openssh-clients
- openssh-server
- openldap
- openldap-clients
- nss_ldap
- apache
- postgresql
- python

The following files are modified by SourceForge 3.4 in /etc/ and must not be updated.

- crontab
- syslog.conf

The following RPMs are removed by SourceForge 3.4 upon installation and must not be reinstalled.

- sendmail
- sendmail-cf
- openssh
- openssh-clients
- openssh-server
- openldap
- openldap-clients
- nss_ldap

SourceForge 3.4 symlinks the following system binaries with the SourceForge tree upon installation. They must not be updated.

- python

SUN Solaris 9

To install SUN Solaris 9 from a standard Solaris distribution, follow the steps below. Otherwise you may use Flash Archives.

For additional installation information, see the Solaris installation documentation.

SUN Solaris 9 Installation Guide: *http://docs.sun.com/db/doc/806-5205*

The Sun Solaris 9 installation procedure involves four main activities.

1. Boot and select the Language and Locale.
2. Run sysidtool.
3. Run suninstall.
4. Finish the installation.

Boot and Select the Language and Locale

1. Boot from the Solaris 9 Software 1 of 2 CD.
2. Select a Language.

   Enter choice (0-9)
3. Select a Locale.

   Enter choice (0-59)

Run sysidtool

After selecting a language and locale, the next step is the sysidtool.

1. Use the sysidtool to enter and confirm information about your system when prompted.
2. Select the "No" response at the "Enable IPv6" and "Configure Kerberos Security" prompts.

   SourceForge 3.4 does not require IPv6 or Kerberos.

Run suninstall

The next step is suninstall.

1. From the initial menu of suninstall, "Solaris Interactive Installation," use "F4_Initial" to overwrite the system disks.
2. Then use "F2_Standard" to install from a standard Solaris Distribution.
3. Select the geographic regions for which support should be installed (i.e. en_US.ISO8859-1) and use "F2_Continue" to go to the "Select Software" menu.
4. Select "Developer System Support 64 bit" and use "F4_Customize" to go to the"Customize Software" menu.
5. Deselect the following and then use "F2_OK" to continue.

   Live Upgrade Software (root and usr modules)

   Secure Shell (Client, Common, and Server, Root and Usr modules)

   Sendmail support (root and user modules)

   jpeg - The Independent JPEG Groups JPEG software

   libtiff - library for reading and writing TIFF
6. Use "F2_Continue" to go to the next menu and continue accepting the default configuration options or adjusting them for your system, as required.

Finish the Installation

To finish the installation:

1. Create a root password.
2. If you are installing from CD, wait until the "Specify Media" menu appears and then click Next> to continue.
3. When prompted, insert the Solaris 9 Software 2 of 2 CD and then direct the installer to continue.
4. When the summary of installation details is displayed, direct the installer to reboot.
5. Obtain the Solaris 9 Recommended Patch Cluster 052803 and copy it onto your system. See the Reference Environment section of the SourceForge 3.4 Release Notes for the list of patches in this cluster that were applied to SourceForge 3.4 for testing.

   Put the system into single user mode (init S). Execute the install_cluster script. Reboot.

Cron jobs

SourceForge relies on a number of cron jobs run from the root user's crontab. At installation time, SourceForge will replace the current root crontab and place a backup at:

/sourceforge/etc/cron/cron-root

If you require other cron jobs run on this system, please add them to the SourceForge-installed crontab after SourceForge installation.

Preparing the Database

SourceForge 3.4 requires a database (Oracle, DB2, or PostgreSQL) to be installed and accessible by all servers.

This section describes how to prepare your database for the SourceForge 3.4 installation.

Oracle

This section describes how to prepare your Oracle database for the SourceForge installation. SourceForge 3.4 is compatible with Oracle 9iR2 Enterprise Edition (9.2.0.3.0) loaded on RedHat Advanced Server 2.1 or Solaris 9.

⇨ Refer to the Oracle documentation for Oracle installation information.

⇨ Refer to Oracle 9i documentation for installing Oracle 9i on a server that has RedHat Advanced Server 2.1 installed.

Before running the SourceForge Oracle database creation scripts, your database must meet the following conditions:

- Oracle must be installed and running.
- The *ctxsys* user must be unlocked.
- The target tablespace must be created. This tablespace should be at least 200 MB initially, and set to auto-extend.

  See *Creating the target tablespace* below for detailed instructions.
- The *ctxsrv* process must not be running.

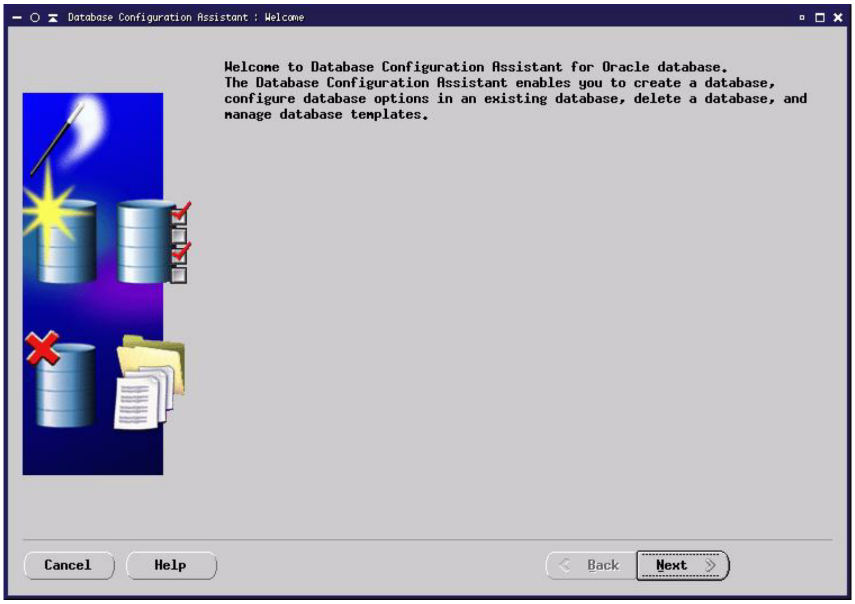
  The ctxsrv process is deprecated by Oracle. Its functionality has been replaced by an automated job (*index_sync.py*) on the SourceForge server.
- The *drsys* and *indx* tablespaces must be set up. The Oracle setup default values should be correct for most installations.

Oracle intermedia must be running (for SourceForge search functionality.)

Oracle Database Creation Best Practices

This section describes recommended best practices for creating your Oracle database.

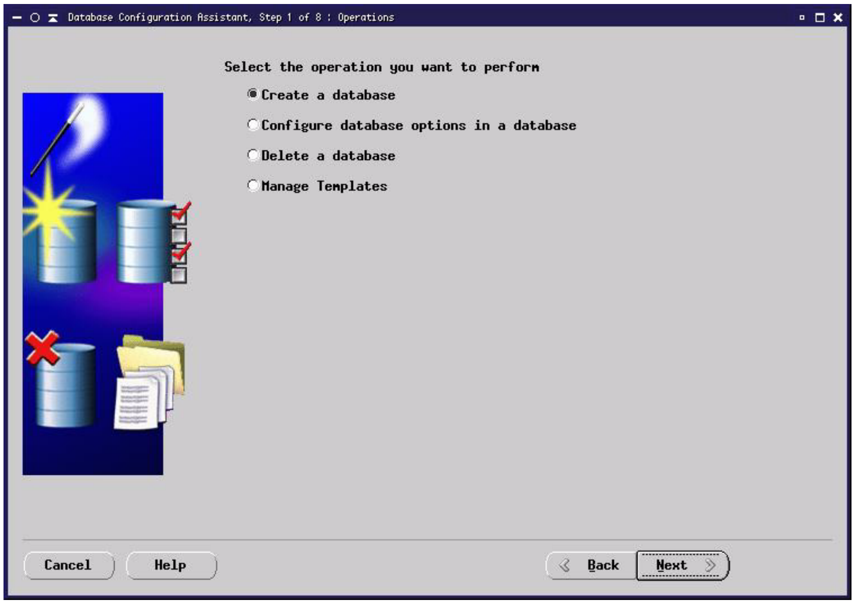
1. As the Oracle user, start the database configuration assistant (dbca), then click **Next**.



Database Configuration Assistant : Welcome

Welcome to Database Configuration Assistant for Oracle database.
The Database Configuration Assistant enables you to create a database, configure database options in an existing database, delete a database, and manage database templates.

2. Select **Create a database**, then click **Next**.



Database Configuration Assistant, Step 1 of 8 : Operations

Select the operation you want to perform

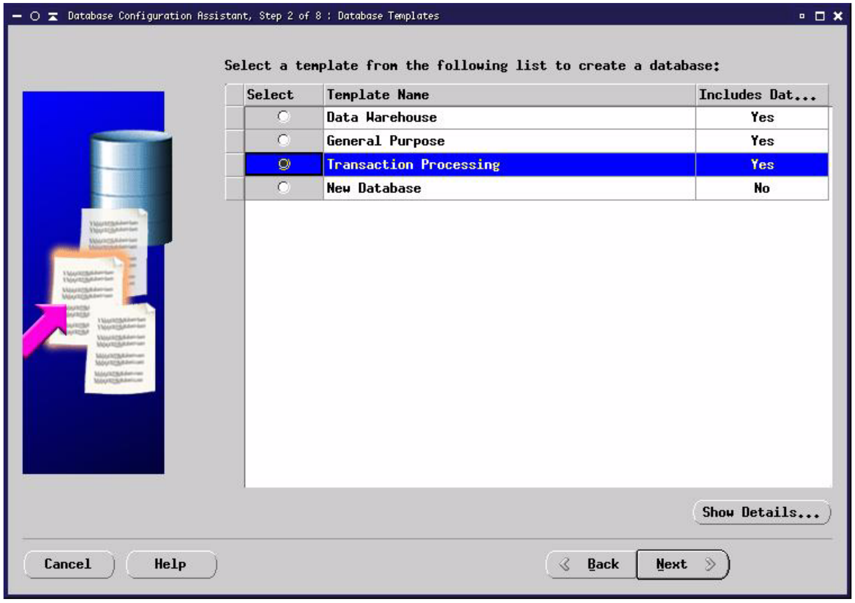
● Create a database

○ Configure database options in a database

○ Delete a database

○ Manage Templates

3. Select the **Transaction Processing** database template, then click **Next**.



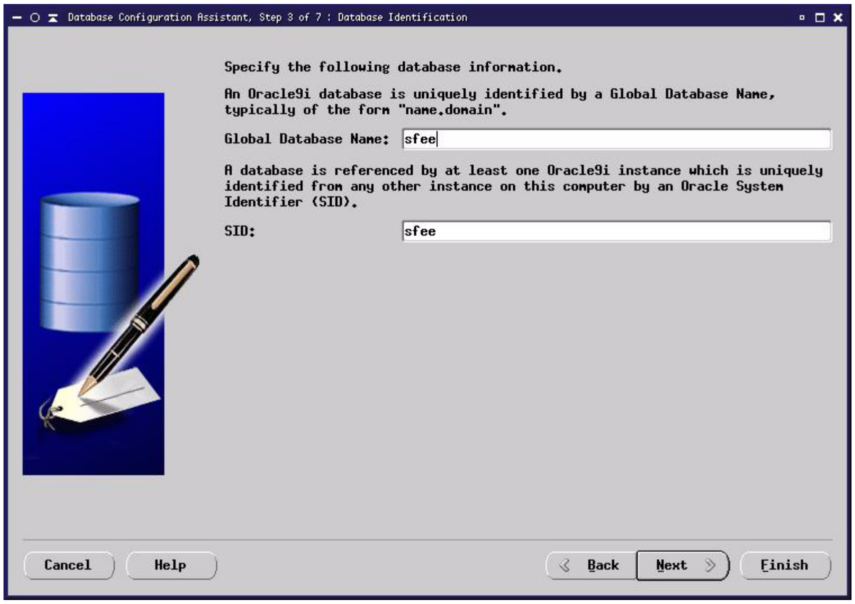
Database Configuration Assistant, Step 2 of 8 : Database Templates

Select a template from the following list to create a database:

Select	Template Name	Includes Dat...
○	Data Warehouse	Yes
○	General Purpose	Yes
●	Transaction Processing	Yes
○	New Database	No

4. The **Global Database Name** and **SID** are both purely arbitrary; however, for the sake of simplicity, you should make them the same value.

   For the remainder of these instructions, they will be set to **sfee**. Fill in both fields, then click **Next**.



Database Configuration Assistant, Step 3 of 7 : Database Identification

Specify the following database information.

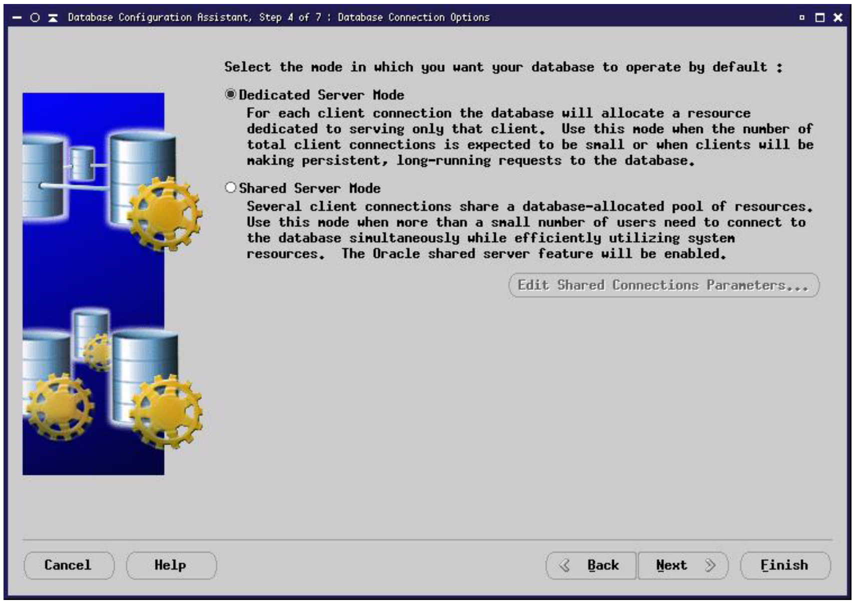
An Oracle9i database is uniquely identified by a Global Database Name, typically of the form "name.domain".

Global Database Name: sfee

A database is referenced by at least one Oracle9i instance which is uniquely identified from any other instance on this computer by an Oracle System Identifier (SID).

SID: sfee

5. **Dedicated Server Mod**e is sufficient for most SourceForge installations. Make this selection, then click **Next**.



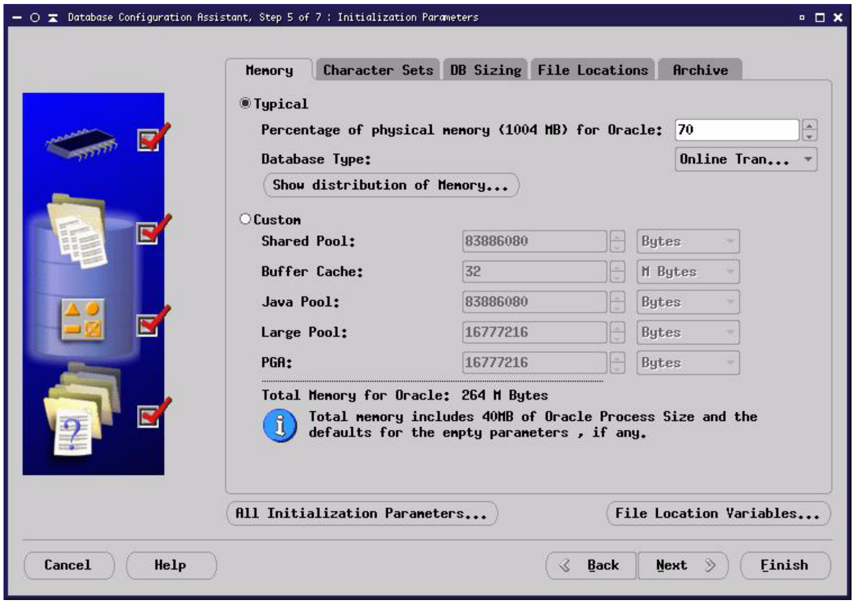
Database Configuration Assistant, Step 4 of 7 : Database Connection Options

Select the mode in which you want your database to operate by default :

● Dedicated Server Mode
For each client connection the database will allocate a resource dedicated to serving only that client. Use this mode when the number of total client connections is expected to be small or when clients will be making persistent, long-running requests to the database.

○ Shared Server Mode
Several client connections share a database-allocated pool of resources. Use this mode when more than a small number of users need to connect to the database simultaneously while efficiently utilizing system resources. The Oracle shared server feature will be enabled.

6. In Step 5, the default is usually a safe choice. Note that this is heavily dependent on the memory resources available on the Oracle server. Next, click **File Location Variables**.



Database Configuration Assistant, Step 5 of 7 : Initialization Parameters

● Typical

Percentage of physical memory (1004 MB) for Oracle: 70

Database Type: Online Tran...

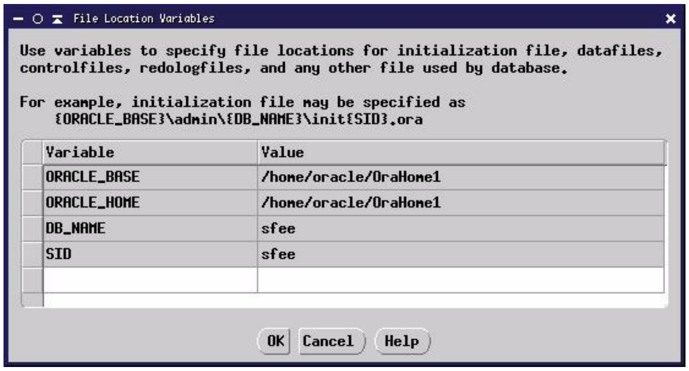
○ Custom

Shared Pool:	83886080	Bytes
Buffer Cache:	32	M Bytes
Java Pool:	83886080	Bytes
Large Pool:	16777216	Bytes
PGA:	16777216	Bytes

Total Memory for Oracle: 264 M Bytes

Total memory includes 40MB of Oracle Process Size and the defaults for the empty parameters , if any.

7. Take note of all four variables in the resulting popup window (**ORACLE_BASE, ORACLE_HOME, DB_NAME**, and **SID**), as they will be relevant when you are performing the SourceForge installation. Click **OK** when done, then click **Next** to continue.

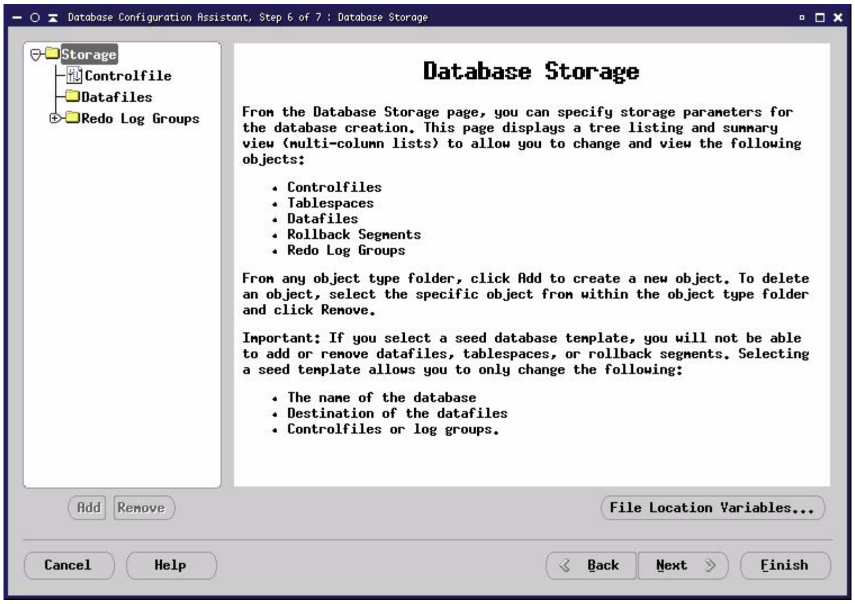


Use variables to specify file locations for initialization file, datafiles, controlfiles, redologfiles, and any other file used by database.

For example, initialization file may be specified as
    {ORACLE_BASE}\admin\{DB_NAME}\init{SID}.ora

Variable	Value
ORACLE_BASE	/home/oracle/OraHome1
ORACLE_HOME	/home/oracle/OraHome1
DB_NAME	sfee
SID	sfee

8. No changes are necessary in Step 6. Click **Next** to continue.



Database Configuration Assistant, Step 6 of 7 : Database Storage
Storage
Controlfile
Datafiles
Redo Log Groups
Add
Remove
File Location Variables...
Cancel
Help
Back
Next
Finish

Database Storage

From the Database Storage page, you can specify storage parameters for the database creation. This page displays a tree listing and summary view (multi-column lists) to allow you to change and view the following objects:

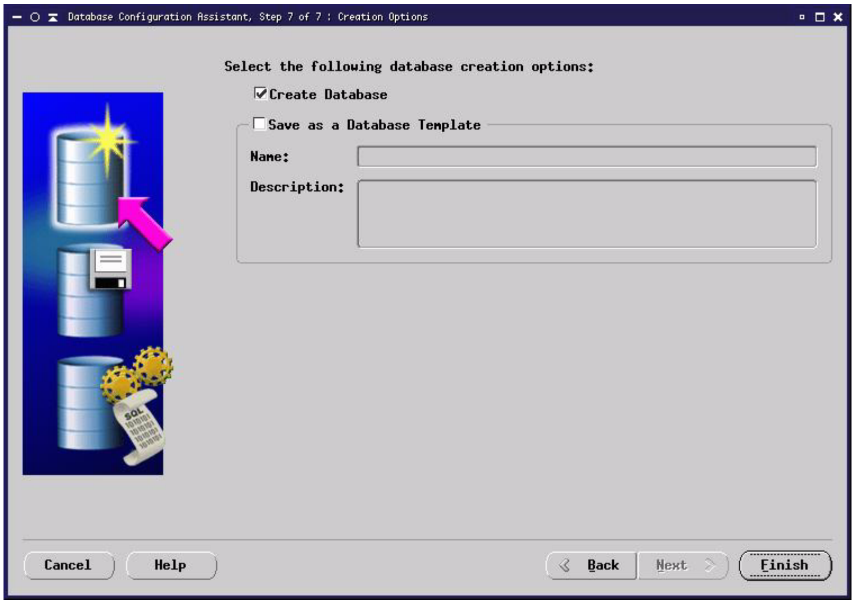
- Controlfiles
- Tablespaces
- Datafiles
- Rollback Segments
- Redo Log Groups

From any object type folder, click Add to create a new object. To delete an object, select the specific object from within the object type folder and click Remove.

Important: If you select a seed database template, you will not be able to add or remove datafiles, tablespaces, or rollback segments. Selecting a seed template allows you to only change the following:

- The name of the database
- Destination of the datafiles
- Controlfiles or log groups.

9. Check **Create Database**, then click **Finish** to continue.



Database Configuration Assistant, Step 7 of 7 : Creation Options

Select the following database creation options:

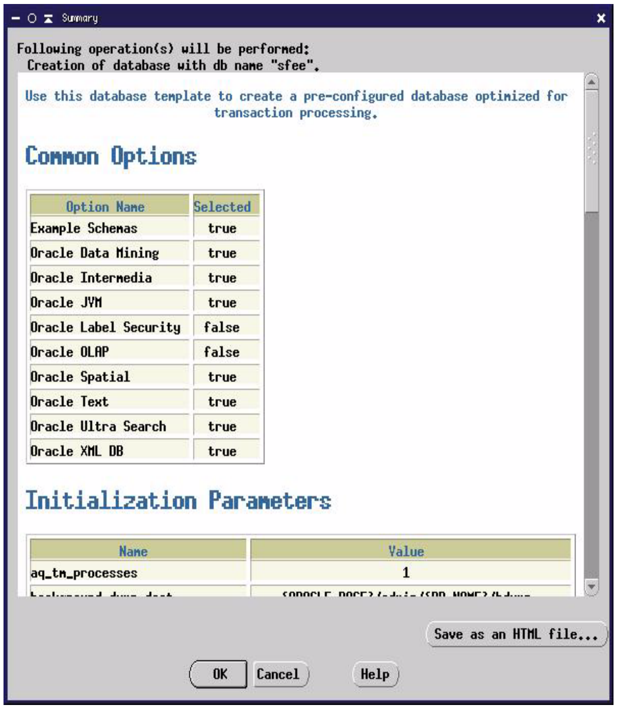
☑ Create Database

□ Save as a Database Template

Name:

Description:

10. Review the options and values and make certain that **Oracle Text** is set to **true**. Click **OK** to continue.



Summary

Following operation(s) will be performed:
Creation of database with db name "sfee".

Use this database template to create a pre-configured database optimized for transaction processing.

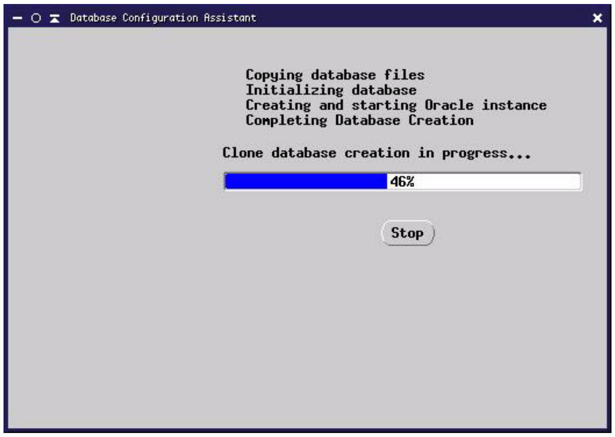
Common Options

Option Name	Selected
Example Schemas	true
Oracle Data Mining	true
Oracle Intermedia	true
Oracle JVM	true
Oracle Label Security	false
Oracle OLAP	false
Oracle Spatial	true
Oracle Text	true
Oracle Ultra Search	true
Oracle XML DB	true

Initialization Parameters

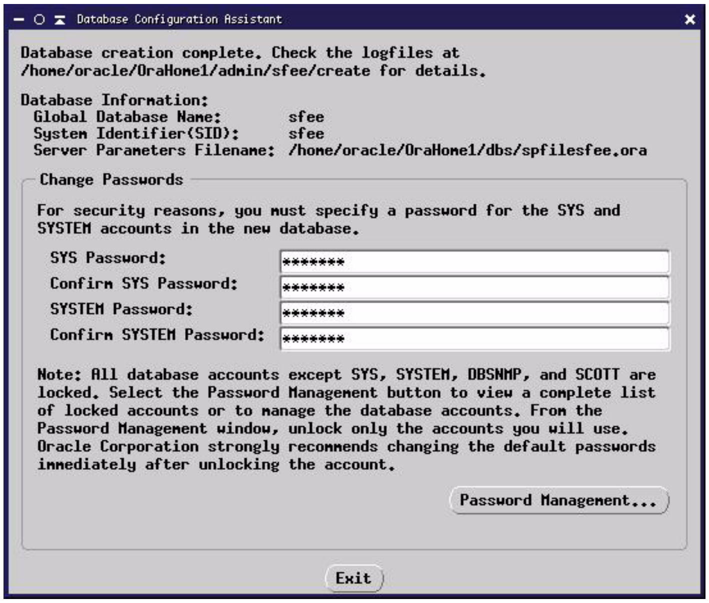
Name	Value
aq_tm_processes	1

11. At this time, the actual database creation begins and a progress bar is displayed. This commonly takes about five to ten minutes to complete on a server with low to moderate load. It is normal for the progress bar to take several minutes to reach 46%, then significantly less time for the second half of the database creation process.



```
Copying database files
Initializing database
Creating and starting Oracle instance
Completing Database Creation

Clone database creation in progress...
```

12. At this point, the database is setup and all that remains is to set a few passwords and unlock the ctxsys user's account. If there are already pre-existing Oracle databases on this server, then you should make certain to enter the same **SYS** and **SYSTEM** passwords that you've used in the past, otherwise new passwords will be set for those accounts. Once you've entered passwords for both the **SYS** and **SYSTEM** accounts, click **Password Management...** to continue.



```
Database creation complete. Check the logfiles at
/home/oracle/OraHome1/admin/sfee/create for details.

Database Information:
  Global Database Name:        sfee
  System Identifier(SID):      sfee
  Server Parameters Filename:  /home/oracle/OraHome1/dbs/spfilesfee.ora
```

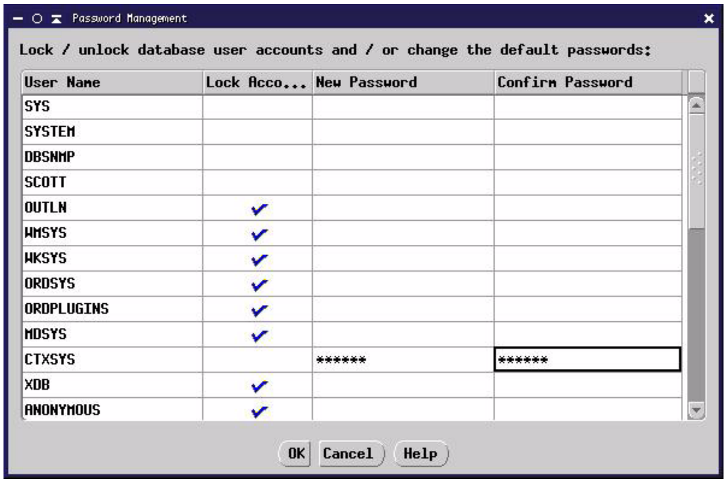
Change Passwords

For security reasons, you must specify a password for the SYS and SYSTEM accounts in the new database.

SYS Password: *******
Confirm SYS Password: *******
SYSTEM Password: *******
Confirm SYSTEM Password: *******

Note: All database accounts except SYS, SYSTEM, DBSNMP, and SCOTT are locked. Select the Password Management button to view a complete list of locked accounts or to manage the database accounts. From the Password Management window, unlock only the accounts you will use. Oracle Corporation strongly recommends changing the default passwords immediately after unlocking the account.

13. For the **CTXSYS** user, uncheck the checkmark in the **Lock Account** column. Then select a password for the **CTXSYS** user. If you are already using ctx in a pre-existing database, make certain to use the same password as was selected in the past. When done, click **OK**, then **Exit**.



Lock / unlock database user accounts and / or change the default passwords:

User Name	Lock Acco...	New Password	Confirm Password
SYS			
SYSTEM			
DBSNMP			
SCOTT			
OUTLN	✓		
WMSYS	✓		
WKSYS	✓		
ORDSYS	✓		
ORDPLUGINS	✓		
MDSYS	✓		
CTXSYS		******	******
XDB	✓		
ANONYMOUS	✓		

14. Edit **/etc/oratab** so that the N at the end of the last line is changed to a Y:

```
sfee:/home/oracle/OraHome1:Y
```

15. Create an init file for the new sfee database in **$ORACLE_HOME/dbs** like so:

```
cp init.ora initsfee.ora
```

Creating the Tablespaces

After you have created your Oracle database, you must create the tablespaces.

Creating the target tablespace

Using *sqlplus* as *username sys,* create the target tablespace by entering the following:

```
$ sqlplus sys/password@tnsname [in Oracle 8i]
$ sqlplus /nolog
$ connect sys/<PASSWORD>@<SID> as sysdba [in Oracle 9i]
SQL> CREATE TABLESPACE sfee
SQL> DATAFILE '/path/to/oracle/oradata/SID/sfee1.dbf'
SQL> SIZE 200M
SQL> AUTOEXTEND ON NEXT 10M;
SQL> quit
$
```

This will create 200 MB tablespace "`sfee`" in file "`sfee1.dbf`" that is allowed to expand in 10 MB increments up to 2 GB.

Creating the tablespace for index files

You must create separate tablespace for index files. The procedure is similar to the one for creating the target tablespace.

Using *sqlplus* as *username sys,* create the target tablespace by entering the following:

```
$ sqlplus sys/password@tnsname [in Oracle 8i]
$ sqlplus /nolog
$ connect sys/<PASSWORD>@<SID> as sysdba [in Oracle 9i]
SQL> CREATE TABLESPACE sfeeindex
SQL> DATAFILE '/path/to/oracle/oradata/SID/sfee2.dbf'
SQL> SIZE 200M
SQL> AUTOEXTEND ON NEXT 10M;
SQL> Alter user ctxsys account unlock
SQL> quit
$
```

This will create 200 MB tablespace "`sfee`" in file "`sfee2.dbf`" that is allowed to expand in 10 MB increments.

To prepare Oracle for SourceForge

1. Insert and mount the SourceForge 3.4 installation CD on the database server.
2. Change to the DB-Setup/core directory:

   ```
   cd /mnt/cdrom/DB-Setup/core
   ```

   Note: *sqlplus* must be started in this directory for relative path names to work properly.
3. Start *sqlplus* as follows:

   For Oracle 8i

   ```
   sqlplus sys/<your sys password>@<your_tnsname>
   ```

   For Oracle 9i

   ```
   $ sqlplus /nolog
   connect sys/<PASSWORD>@<SID> as sysdba
   ```
4. Create the SourceForge schema.

   ```
   @create_schema.se.sql or @create_schema.ee.sql, depending on the
   Oracle installation type, Enterprise Edition or Standard Edition.
   ```

   Note: The Global database name and the SID must be the same.

   The script will prompt for the following values:

Value	Description
Database tnsname	The tnsname where your database will be created.
Password for CTXSYS user	The password for the CTXSYS user.
User/Schema to create	The name of the schema that you want to create. This is the user that you will use to connect to Oracle.
Password for new user	The password for the user/schema.
Default tablespace	The tablespace where you want to create your database. (`sfee`, if you used the default `SQL> CREATE TABLESPACE sfee`, as described previously in the section *Creating the target tablespace*).
Tablespace for index files	In case of large installations you must create two tablespaces. (`sfeeindex`, if you used the default `SQL> CREATE TABLESPACE sfee`, as described previously in the section *Creating the tablespace for index files*.)

5. Exit *sqlplus*.

   ```
   quit
   ```

Installing Oracle Client Libraries

This section describes the steps for installing Oracle client libraries on the SourceForge server.

Use 8i Libraries for 9i Installations

SourceForge 3.4 has been qualified on Oracle 9i. However, the Apache binary distributed with SourceForge 3.0.x has only been compiled against libclntsh.so.8.0, which ships only with Oracle 8i. Libclntsh.so.9.0, which ships with Oracle 9i, is not backwards compatible.

This incompatibility prevents Apache from starting or running. As a workaround, users should download the 8i client libraries from Oracle to install with Oracle 9i.

To install Oracle client libraries on Linux

1. Download the Oracle 8i client libraries from the Oracle website:

   http://otn.oracle.com/software/products/oracle8i/content.html
2. Place the Oracle client libraries (*libclntsh.so.** and *libwtc*.so*) in a newly created directory, */lib/oracle*.
3. Add an entry in */etc/ld.so.conf* with the following:
   /lib/oracle
4. Run */sbin/ldconfig*.

To install Oracle client libraries on Solaris

1. Download the Oracle 8i client libraries from the Oracle website:

   http://otn.oracle.com/software/products/oracle8i/content.html
2. Place the Oracle client libraries (*libclntsh.so.** and *libwtc*.so*) in one of the following directories:
   /usr/lib

   /usr/local/lib
3. Run */usr/bin/crle*

   This displays the current configuration and confirms that your changes have been made.

Oracle Database Security

The following security measures are recommended to protect data stored in Oracle.

- Keep your Oracle installation current on security and bug patches.
- Use a firewall mechanism for the open ports on both your Oracle and SourceForge servers.
- Open the Oracle ports 1521 and 8080 only to the SourceForge servers (and any other servers that use Oracle.)

PostgreSQL

This section describes how to prepare your PostgreSQL database for SourceForge installation. SourceForge 3.4 is compatible with PostgreSQL 7.3.3 and later versions.

For PostgreSQL installation instructions, visit:

http://www.postgresql.org/idocs/index.php?admin.html

To prepare PostgreSQL for SourceForge

1. Insert and mount the SourceForge 3.4 installation CD on the database server.
2. Create the database.

   ```
   createdb -U<dbadmin_user> <your_database_name>
   ```

3. Change to the *DB-Setup/core* directory.

   `cd /mnt/cdrom/DB-Setup/core` *(for RedHat Linux installations)*

   `cd /DB-Setup/core` *(for all other installations)*

4. Create a new schema.

   ```
   bash create_schema.sh
   ```

   The script will prompt for Database name and Database user name.

PostgreSQL Maintenance Requirements

All database platforms require regular client maintenance to maximize performance and maintain stability. In the case of PostgreSQL, this maintenance is relatively simple and straightforward. There are three basic maintenance categories: backups, restoration, and optimization.

Backups

Any software application can fail unexpectedly, and databases are no exception. Intelligent contingency planning necessitates periodic full database backups. Backups should be run on a daily basis. This process involves invoking a straightforward, built-in backup command, which can be easily scripted to run automatically, late at night - a time of minimal server load.

To further mitigate risk, the resultant backup archive file should be stored on a separate server or medium, and/or added to an existing corporate-wide backup system, if applicable.

Restoration

PostgreSQL is extremely stable. However, prudent planning requires not only implementing continuous backup procedures, but also periodically testing and validating database restoration. Restoration should be performed using a secondary "scratch" or "development" server, which need not be of similar configuration to the production system. This testing should be undertaken on a quarterly basis.

Optimization

Normal use of database software often creates "data overhead" that needs to be periodically cleaned in order to ensure optimal speed and stability. This overhead is usually the result of temporary files and indexes that the database creates (analogous to a fragmented hard disk.)

Optimizing a PostgreSQL database means running a built-in utility called "Vacuum." This utility runs on a live database and, like the backup command, can be scripted to run nightly during minimal server load.

Best Practices for PostgreSQL Maintenance

As part of standard product documentation, VA provides detailed instructions and best practices guidelines for PostgreSQL backup, restoration and optimization procedures. This documentation includes command syntax and process recommendations.

1. Run Nightly Cronjobs

   As with all databases, PostgreSQL requires regular maintenence to ensure the integrity of its data, and to improve performance. It is recommended that the dump and vacuum portions of this document be integrated into a nightly cronjob to ensure optimal performance and reliability of your PostgreSQL database.

2. Make Regular Backups

   Making regular full backups of the database ensure that recent SourceForge data is always available should there be any kind of service failure. A full dump of all databases hosted by PostgreSQL can be accomplished by running the following command as the PostgreSQL user:

   ```
   pg_dump -Ft -b -o DB_NAME > /tmp/db.tar
   ```

   This command will dump the database contents to /tmp/db.tar and can be run on a live database with a moderate degradation in performance. As such, it is recommended that the dump be performed via a cronjob during a period of least server activity. Note that the file, db.tar, will still need to be backed up as part of a serverwide backup routine.

   You can restore[1] from the dumped db.tar file with the following commands run as the PostgreSQL user:

   ```
   createuser -U SFUSER
   createdb -U SFUSER -T template0 DB_NAME
   pg_restore -d DB_NAME /tmp/db.tar
   ```

   In these commands SFUSER is the SourceForge database user, which can be referenced on the SourceForge server in */sourceforge/etc/sfee/local.inc*.

3. Optimize Database Tables

   Optimizing the database tables improves overall performance, as well as recovers diskspace occupied by deleted or updated rows. PostgreSQL comes with a tool known as vacuum which can perform such optimizations on a live database. To vacuum all hosted databases, run the following command as the PostgreSQL user:

   ```
   vacuumdb -a -f -v -z 2>&1 | tee /tmp/vacuumdb.log
   ```

   While vacuumdb can be run via a cronjob, it is strongly recommended that it be run manually the first time to ensure that there are no unexpected problems. vacuumdb will result in moderate to severe performance degradation, so it is recommended that it be run during a period of low server activitiy. Note that this logs the results of the vacuum to /tmp/vacuumdb.log.

Note: The information provided in this document is not intended to be a substitute for the official PostgreSQL documentation. The PostgreSQL-7.3 Administrator's Guide can be obtained here:

http://www.postgresql.org/docs/view.php?version=7.3&idoc=0&file=admin.html

Questions regarding the contents of this document should be sent to Technical Support (*support-sf@vasoftware.com*).

Internet Resources

For organizations interested in greater information about PostgreSQL, complete documentation and extensive technical information as available at:
http://techdocs.postgresql.org

VA Software Technical Support

For information on PostgreSQL support services offered by VA Software, please see "VA Software PostgreSQL Support Offerings" on page 95.

DB2

The following is a recommended database set up procedure. The procedure may vary depending upon your requirements.

1. Login as db2inst1 (or, the name of the user you specified as the DB2 instance owner):

   ```
   $ su - db2inst1
   password: <password>
   ```

2. Create the database with the following command:

   ```
   $ db2 create database <db_name>
   ```

3. Create tablespaces with the following commands:

   ```
   $ db2 "CONNECT TO <db_name> USER <db_user> USING <db_user_password>"
   $ db2 update db cfg for <db_name> using dbheap 10000
   $ db2 update db cfg for <db_name> using applheapsz 5000
   $ db2 update dbm cfg using query_heap_sz 2400
   ```

4. Create buffer pools with the following commands, where <BUFFERSPOOLNAME#> is an arbitrary name of your choice:

   ```
   $ db2 "CREATE BUFFERPOOL <BUFFERSPOOLNAME1> SIZE 1000 PAGESIZE 8 K"
   $ db2 "CREATE BUFFERPOOL <BUFFERSPOOLNAME2> SIZE 1000 PAGESIZE 16 K"
   $ db2 "CREATE BUFFERPOOL <BUFFERSPOOLNAME3> SIZE 1000 PAGESIZE 32 K"
   ```

5. Restart DB2

   ```
   $ db2 "RESTART DB <db_name> USER <db_user> USING <db_user_password>"
   ```

6. Create Tablespaces with the following commands, where <table_space_name> is an arbitrary name of your choice:

   ```
   $ db2 "CREATE REGULAR TABLESPACE <table_space_name> PAGESIZE 8 K MANAGED
   BY SYSTEM USING('*</path of your database dir>') EXTENTSIZE 32 OVERHEAD
   24.1 PREFETCHSIZE 32 TRANSFERRATE 0.9 BUFFERPOOL SFBUFFER8"
   ```

   Note: <table_space_name> should be unique.

Example:

```
$ db2 "CREATE REGULAR TABLESPACE tbspace8 PAGESIZE 8 K MANAGED BY
SYSTEM USING('/home/db2inst1/') EXTENTSIZE 32 OVERHEAD 24.1
PREFETCHSIZE 32 TRANSFERRATE 0.9 BUFFERPOOL SFBUFFER8"
$ db2 "CREATE REGULAR TABLESPACE <table_space_name> PAGESIZE 16 K
MANAGED BY SYSTEM USING('/home/db2inst1/') EXTENTSIZE 32 OVERHEAD
24.1 PREFETCHSIZE 32 TRANSFERRATE 0.9 BUFFERPOOL SFBUFFER16"
$ db2 "CREATE REGULAR TABLESPACE <table_space_name> PAGESIZE 32 K
MANAGED BY SYSTEM USING('/home/db2inst1/') EXTENTSIZE 32 OVERHEAD
24.1 PREFETCHSIZE 32 TRANSFERRATE 0.9 BUFFERPOOL SFBUFFER32"
$ db2 "CREATE SYSTEM TEMPORARY TABLESPACE <table_space_name> PAGESIZE
32 K MANAGED BY SYSTEM USING ('/home/db2inst1/') EXTENTSIZE 32
OVERHEAD 24.1 PREFETCHSIZE 32 TRANSFERRATE 0.9 BUFFERPOOL SFBUFFER32"
$ db2 "CREATE USER TEMPORARY TABLESPACE <table_space_name> PAGESIZE
16 K MANAGED BY SYSTEM USING ('/home/db2inst1/') EXTENTSIZE 32
OVERHEAD 24.1 PREFETCHSIZE 32 TRANSFERRATE 0.9 BUFFERPOOL SFBUFFER16"
$ db2 "GRANT SELECT ON SYSIBM.SYSDUMMY1 TO PUBLIC"
```

7. For Clob/Blob to function properly, append the following two lines in the file db2cli.ini (located at */home/db2inst1/sqllib/cfg/db2cli.ini*)

    [common]

    PATCH2 = 6

Creating the SourceForge Schema

Login as Root and do the following

1. Change the path to the core directory with the following command:

```
$ cd <staging directory>/DB-Setup/core
```

2. Run create_schema.sh with the following command:

```
$ ./create_schema.sh
```

This command will prompt for the following entries:

Enter the DB name:

Enter the DB user:

Enter the DB password:

The create_schema.sh script will create a log file create_schema.log in the current directory. Ensure that write permissions are given to the current user; otherwise the log file will not be created.

Creating the Client for the DB2 Server

1. Get the *DB2_V81_ESE_LNX_32_NLV.tar* file

   The tar file is available on the DB2 installation CD.

2. **`tar -xvf`** ***`DB2_V81_ESE_LNX_32_NLV.tar`***
3. **`cd 009_ESE_LNX_32_NLV/`**

   Note: The tar file and directory name might be different on your media.

4. Run the db2setup wizard script.

   ```
   $ su - root
   ```

   db2setup works with Bash, Bourne & Korn shells only. db2setup is a graphical tool, so X window libraries are required. You must install *XFree86-libs-4.1.0-29.i386.rpm* from the CD, and then export your display to your local machine.

   Note: RedHat has released a new security update versioned as XFree86-4.1.0-44, which can be downloaded from the site.

5. Assuming you have your RedHat Advanced Server 2.1 CD mounted as /mnt/cdrom, execute the following steps:

   ```
   # rpm -ivh --nodeps /mnt/cdrom/Redhat/RPMS/XFree86-libs-4.1.0-29.i386.rpm
   # export DISPLAY = {IP of local machine}:0
   # ./db2setup
   ```

6. Follow the screens

   Screen 1: IBM Universal Database Setup Version 8.1 screen

   Select "Install Products."

   Screen 2: Select the product you would like to install screen

   Choose "DB2 Administration Client."

   Click Next.

   Screen 3: Welcome to the DB2 Setup Wizard (for Client) screen

   Click Next.

   Screen 4: Software License Agreement screen

   Select Accept.

   Click Next.

   Screen 5: Select the installation type screen

   Select Custom.

   Click Next.

Screen 6: Select the installation action screen

Click Next

Select "Install DB2 Administration Client on this computer."

Screen 7: Select the features to install screen

Select the following options:

Client Support

Administration Tools

Click Next.

Screen 8: Language screen

By default "English" is set.

Click Next.

Screen 9: Set up a DB2 instance screen.

Select "Create a DB2 instance."

Click Next.

Screen 10: Set user information for the DB2 instance owner screen.

User Name dbinst1, Group Name db2grp1 and Home Directory */home/db2inst1* are set by default.

Provide a password.

Click Next.

Screen 11: Start copying files screen

Click Finish.

7. Check the user permission on the newly created */home/db2inst1* directory. Make sure that this directory has the "755" permission. The permission change must be recursive.
8. Login as db2inst1.
9. Append the following two lines in the file db2cli.ini located at: */home/db2inst1/sqllib/cfg/* db2cli.ini:

```
[common]
PATCH2 = 6
```

Configuring the TCP/IP communication

1. Issue the following commands on the DB2 client:

   ```
   $db2 catalog tcpip node <node_name> remote <DB2 server IP> server
   <service name/port number>
   $ db2 catalog database <db_name> as <alias> at node <node_name>
   $ db2 connect to <alias> user <existing user at server side>
   ```

   Note: By default, <service name/port number> 50000 is selected.

   <node_name> can be any name but should not be more than 8 characters.

   <alias> must be the same as that of <db_name> for all servers.

   <existing user at server side> must be the same user that was used when creating the SourceForge schema.

   The above DB2 Client installation must be performed for all participating servers.

2. Before running setup.sh, issue the following command as root:

   ```
   #export DB2_HOME=/home/db2inst1
   ```

3. Run setup.sh to install SourceForge and follow the instructions described in *Chapter 3, Installing SourceForge* on page 69.

Setting up Rational ClearCase

SourceForge 3.4 supports Rational ClearCase versions 4.2 and 5.0. The installation procedures in this section are applicable to both 4.2 and 5.0 unless otherwise stated.

Setting up Rational ClearCase involves the following two steps:

1. Install a setup tarball.
2. Configure the ClearTool wrapper script.

Rational ClearCase 4.2 or 5.0 Installation for SourceForge

1. As a root user, untar the installation file.

   ClearCase does not require a user/group to be defined for installation.

   - For 4.2, the installation binaries will be in the directory *<folder>/v4.2/sun5/install*
   - For 5.0, the installation binaries will be in the directory *<folder>/CC_5.0_Sun/v5.0/sun5*

2. In the installation binaries folder, execute the file site_prep.

   Use the following settings:

   ```
   ./site_prep
   -<server name>
   -<server name>
   -<server name>
   -default
   -no to the rest of the prompts
   ```

   This step prepares for ClearCase installation.

3. In the installation binaries folder, execute the file install_release.

   Use the following settings.

   ```
   ./install_release
   -method 1
   -full copy
   -/usr/atria
   -options 6 and 8
   -default for the rest of the prompts
   -yes for > 2GB vobs
   ```

   Enter the ClearCase license information in the following file:

   ```
   /var/adm/atria/license.db
   ```

   The license file should look like:

   ```
   -license...
   -root
   ```

   To confirm that you have a valid license use the following command:

```
/user/atria/bin/clearlicense
```

To set the registry password for ClearCase use the following command:

```
/usr/atria/etc/rgy_passwd
```

4. For ClearCase integration to function, ccweb must be running. Ccweb is a CGI program so it must be on a webserver in Solaris 8. Install the Apache package, which is located in the Solaris software media. The packages to be installed are SUNWapchd, SUNWapchr and SUNWapchu

5. The following entries must be made in the *httpd.conf* file of the web server. The *httpd.conf* file will be located in the folder */etc/apache*. If the web server is not being used already, the file *httpd.conf-example* can be copied as *http.conf* and used. The additions mentioned here can come at the end of the *httpd.conf* file.

   ```
   ScriptAlias "/ccbin" "/usr/atria/web/bin"
   Alias "/applets" "/usr/atria/web/applets"
   Alias "/help" "/usr/atria/web/help"
   Alias "/pages" "/usr/atria/web/pages"
   Alias "/images" "/usr/atria/web/images"
   ```

   ### For ClearCase v 5.0

   For ClearCase v 5.0, the following must also be added:

   ```
   Alias "/stylesheets" "/usr/atria/web/stylesheets"
   Alias "/scripts" "/usr/atria/web/scripts"
   ```

6. Restart the apache server by using the following commands. If apache was not running already, it only needs to be started.

   ```
   /etc/init.d/apache stop
   /etc/init.d/apache start
   ```

Configuring the Setup Tarball

The setup tarball holds the minimal SourceForge directory structure needed to integrate with ClearCase. You must untar the contents of the setup tarball and either rename or symlink them as */sourceforge*.

Once the contents of the setup tarball are accessible from the */sourceforge directory,* you must configure the tarball to work with the main Sourceforge installation.

To configure the setup tarball:

1. Edit the *template.conf* file
2. Edit the *postinstall.sh* file
3. Run the *postinstall.sh* file

The *template.conf* file contains a set of name/value pairs that must be customized for the specific needs of your SCM server and your SourceForge installation.

Installing SCM Tar

1. Provide for the directories in which we have the ccvobs, vobs, and views. For example we have them in the directories */ccvobs, /vobs*, and */views*. They can be any directories based on the user requirements. Use the following commands

   ```
   mkdir /ccvobs /vobs /views
   chmod 777 /ccvobs /vobs /views
   ```

   Note: Step 1 is only applicable if you are installing ClearCase with SourceForge. If you have already installed ClearCase, you do not need to create directories for ccvobs, vobs, and views.

2. Untar the SCM tarball in the / directory.

   An */SCM_Setup* directory will be created. Rename this directory to */sourceforge* using the following command:

   ```
   mv /SCM_Setup / sourceforge
   ```

3. Modify the settings in */sourceforge/template.conf*. This file follows perl syntax, therefore care should be taken in making sure that each variable/value associations are properly quoted and end with a comma where appropriate.

4. The sample ClearCase parameters to be set in the template.conf file are:

Parameter	**Value**
'CLEARCASE_HOST_NAME'	“<host name of the SCM server>”
'CLEARCASE_CCWEB_URL_PATH'	“/ccbin/ccweb”
'CLEARCASE_VOBTAG_ROOT'	“/vobs”
'CLEARCASE_VOB_STORAGE_PATH'	“/export1/vobs”
'CLEARCASE_VOB_STORAGE_HPATH'	“export1/vobs”
'CLEARCASE_VOB_STORAGE_GPATH'	“/net/<hostname>1/ vobs”
'CLEARCASE_VIEW_STORAGE_PATH'	“/export1/views”
'CLEARCASE_VIEW_STORAGE_HPATH	“/export1/views”
'CLEARCASE_VIEW_STORAGE_GPATH'	“/net/<hostname>1/ views”
‘CLEARCASE_REGISTRY_PASSWORD’	“password”

The host name should have the host name of the server on which you are installing. The password should have the password given for the ClearCase registry.

The parameters CLEARCASE_VOBTAG_ROOT and CLEARCASE_CCWEB_URL_PATH must be provided when installing the SFEE server. Forms for entering these will come up in the installer of the SFEE server.

Other parameters that must be entered and their explanations are:

Parameter	Value
'SFEE_HOST_NAME'	“<hostname of the SFEE server>”
'BACKEND_HOST_NAME'	“<hostname of the BACKEND server>”
'DATABASE_HOST_NAME'	“<hostname of the DATABASE server>”
'DATABASE_TYPE'	“<the database type>”
'DATABASE_USER'	“<the database user name>”
'DATABASE_PASSWD'	“<the database password>”
'DATABASE_NAME'	“<the database name>”
'THIS_HOST_NAME'	“<hostname of the SCM server>”

All other parameters need not be filled in.

5. Copy the xmlrpcd shared secret from the SFEE server to the corresponding entry in the *template.conf*. In the SFEE server the entry can be found in the file

   /sourceforge/etc/scripts/local.py

   It must be copied to the variable XMLRPC_SHARED_SECRET in the template.conf
6. The file */sourceforge/etc/xmlrpc/trusted_hosts* should have the IP address of the following servers:

   The DB server

   The SFEE server

   The SCM server
7. The SCM server should have connectivity to the DB server. If using a PostgreSQL database ensure that the *pg_hba.conf* file permits connection from the SCM server
8. Run *postinstall.sh*

For ClearCase v 5.0

For ClearCase v 5.0, the following two files must be copied using the following commands:

```
cp /sourceforge/sf-cc5-link/InfoPageFmt.js /usr/atria/web/scripts
cp /sourceforge/sf-cc5-link/history /usr/atria/web/pages/Info
```

These commands will overwrite the existing ClearCase files with patched files that provide the link from the ClearCase v 5.0 vobview to the SourceForge tracker.

9. The library files necessary for db connection should be available. To test, use either of the following commands depending on the db type being used.

   ```
   ldd /sourceforge/python/lib/python2.1/site-packages/cx_Oracle.so
   ```

   or

   ```
   ldd /sourceforge/python/lib/python2.1/site-packages/psycopgmodule.so
   ```

10. Start xmlrpcd using the following command

    ```
    /sourceforge/etc/init.d/xmlrpcd.init start
    ```

11. SourceForge does not create unix users on the SCM server, so you must create them manually.

    Note: Each username must have the same username and password as the corresponding user account in SourceForge. It is also essential to ensure that each user has a home directory. A user should also not be logged into a shell in ClearCase while admin functions are being performed on the account.

    Example:

    To create a user tuser9 with a home directory, use the following command:

    ```
    useradd -d /export/home/tuser9 -m tuser9
    ```

Changing the Project Creation Parameters

When you create a new ClearCase project, a second page appears showing all parameters that will be used for the project. The page is pre-populated with the values that were entered at the time of installing the SCM tar. You can edit the parameters from this page.

If you change any of the path parameters for vobs and ccvobs views, ensure that the directory exists in the SCM server with the 777 permission. For example, if you want the vobs to be in the directory */myvobs*, ensure that the directory */myvobs* is created with the permission 777. If the default parameters are used, the directory will be automatically created during SCM tar installation.

If youselect 'Allow all views', you can select from the list of views already created. Otherwise, only the default view will appear in the list.

Connecting to an Existing ClearCase Project

To connect to an existing ClearCase project:

1. Ensure that the ClearCase binaries are installed in the directory */usr/atria*.
2. Create a user in SourceForge who has the same Unix name as the user owning the ClearCase project.

   Have that user register the SourceForge project to be connected to the existing ClearCase project.
3. In the second page of the project registration, enter the parameters with which the ClearCase project was created.

When the project is approved, all of the SourceForge-related triggers are put in place and the SourceForge project is successfully connected to the ClearCase project.

Use the same Unix name for ClearCase project members when they are created in SourceForge. Add them to the SourceForge project.

If you need to create a new ClearCase project on the same machine, you must first manually create the users/group for a project on the SCM machine before the project will be approved.

Example

The following is an example of creating a project in ClearCase and connecting to it from SourceForge.

Use the directories /ccvobs /vobs and /views.

Create the project tclproj9 for the user tuser9.

1. Create the user and group using the following commands:

   ```
   groupadd tclproj9
   useradd -d /export/home/tuser9 -m tuser9
   passwd tuser9
   ```
2. Create the ClearCase project using the following commands:

   ```
   su - tuser9
   umask 002; /usr/atria/bin/cleartool mkvob -tag /ccvobs/tclproj9_sfvob \
       -public -nc -password password -host sfsol1.sf.net \
       -hpa /vobs/tclproj9.vob -gpa /vobs/tclproj9.vob \
       /vobs/tclproj9.vob
   umask 002; /usr/atria/bin/cleartool mount /ccvobs/tclproj9_sfvob
   umask 002; /usr/atria/bin/cleartool mkview -tag tclproj9_sfview \
       -host sfsol1.sf.net -hpa /views/tclproj9_sfview \
       -gpa /views/tclproj9_sfview /views/tclproj9_sfview
   ```

   The project can now be linked to from SourceForge.

3. Create the user **tuser9** in SourceForge.
4. The user **tuser9** registers a project **tclproj9** using ClearCase as the SCM.
5. In the project registration page, choose “Existing vob”.

   Fill in all project details.

When approved, all SourceForge triggers will be put into the view.

Rooting a Project in a Version Extended Directory

Another provision for project creation is to have the project rooted in the Version Extended directory of a existing Vob. At project registration when you choose the option to create in Version extended path (Vxd), there are two important parameters that you must provide:

- viewtag
- version extended path in the vob.

After providing these paramaters, click on the button to check if the given Vxd is valid. The viewtag is used for accessing the Vxd.

Assume that a project is created in clearcase with tcp1_sfvob as vob.

To create a version extended path:

1. Login to the SCM machine as the user
2. Execute the following commands:

   ```
   export PATH=$PATH:/usr/atria/bin
   cleartool setview tcp1_sfview
   cd /ccvobs/tcp1_sfvob
   cleartool checkout .
   cleartool mkdir module1
   cleartool checkin module1
   cleartool checkin .
   exit
   ```

   This will create a directory called module1 for the tcp1_sfvob using the tcp1_sfview view.
3. To root another ClearCase project in the version extended path, in the registration screen first select the view tcp1_sfview and give the path */ccvobs/tcp1_sfvob/module1@@/main/1* as the version extended path.

   Then click on the button to check the validity of the version extended path given.
4. Then provide the following additional parameters:
   - Vob Owner: The user who is the owner of the vob in which the Vxd exists
   - Vob Owners Group: The group of the vob
   - New Projects View: The view to be created after the sub dir is created in Vxd
   - View Storage Path: The path in which the new view is to be created
   - View Storage hpath: The hpath of the new view
   - View Storage gpath: The gpath of the new view

   These parameters are needed to root the project in a Vxd

5. When the project is approved all the existing SourceForge triggers for */ccvobs/tcp1_sfvob/module1* are removed. A new directory of the form *sf_<ProjectName>* is created under the Vxd. All the triggers for the new SourceForge project are put in the directory created.

ClearCase Commands

The following is a set of ClearCase commands useful for basic operations

It is assumed that a user who is a member of a ClearCase project executes them. Add the directory having cleartool and ctwrap to PATH.

```
export PATH=$PATH:/usr/atria/bin/cleartool:/sourceforge/scm_integration/
bin
```

Table 3. ClearCase commands

Command	Function	Example
cleartool setview <viewname>	Launches ClearCase and displays the users view	setview tclproj1_sfview
cd <pathname>	Changes directory to a subdirectory in the VOB	cd /vobs/tclproj1_sfvob
cleartool checkout -nc	Checks out the current directory from ClearCase	
cleartool mkelem -nc <filename>	Creates a new element or file	cleartool mkelem -nc afc.c
	Create the file using an editor or copy from some other location	vi afc.c
cleartool checkin -nc <filename>	Checks in the newly created file	cleartool checkin afc.c
cleartool checkin -nc	Checks the current directory into ClearCase	
ctwrap ci <filename>		ctwrap ci afc.c

For all commands -nc is used. Do not use -nc when entering a comment. End comments with "."

To enable SCM-Tracker integrated checkin, use the wrapper shell file ctwrap. Set the environmental variable SFPROJECT to the project name.

For example

```
export SFPROJECT=tcproj1
ctwrap ci <filename>
```

SCM-Tracker integration checkins can also be done without using ctwrap. In such cases, the template is not displayed and the user must type in comments manually.

ClearCase Triggers

The following triggers are created by SourceForge during project creation.

Trigger	Function
sf-cc-phase1-<project unix name>	Used in Tracker-SCM integration. Checks whether the user is allowed to make an Tracker-SCM integrated checkin. If not allowed, it returns 1 which prevents the commit.
sf-cc-phase2-<project unix name>	Used in Tracker-SCM integration. Logs the files that were committed. Does not prevent commits.
sf-cc-phasemid-<project unix name>	Used in Search. Logs the files commited for search purposes. makes an xmlrpc call, which can deadlock if initiated from an xmlrpc process. Does not prevent commits.
sf-cc-rbac-commit-<project unix name>	Used for implementing RBAC on checkins. Can prevent commits.
sf-cc-rbac-tab-<project unix name>	Used for implementing RBAC for tagging. Can prevent tag or branch operations.

Setting up PVCS

To set up PVCS

1. Create a user PVCS and a group pvcsgrp on the PVCS installation.

   To avoid conflict in IDs with the ID's created on the SourceForge machine, use 207 as the ID.

   The commands for the user and group creation are

   Login as user root.

   ```
   groupadd -g 207 pvcsgrp
   useradd -d /export/home/pvcs -u 207 -g pvcsgrp -m pvcs
   ```

2. Create a directory pvcs in /usr/ as user root.

   ```
   mkdir/usr/pvcs
   ```

3. Change the owner of /usr/pvcs to pvcs

   ```
   chown pvcs /usr/pvcs
   ```

4. Change the group of /usr/pvcs to pvcsgrp

   ```
   chgrp pvcsgrp /usr/pvcs
   ```

5. Change permission

   ```
   chmod 755 /usr/pvcs
   ```

6. Unzip/untar pvcs installation files.
7. Change the permission of installer file vminst to 755.

   ```
   cd <path of pvcs installation file>/vm/unix
   chmod 755 vminst
   ```

8. Login as user pvcs and run the installer.

   ```
   su - pvcs
   cd <path of pvcs installation files>/vm/unix
   ```

   (For all unix flavors, use /vm/unix/vminst)

   ```
   ./vminst
   ```

9. Apply Serial Number and IPE Key for installation.

Installing the PVCS Tarball

Complete the following before starting the xmlrpc daemon.

1. Untar the tarball in a destination directory.

   If the directory is not /, create a symlink that links the untarred contents of the xmlrpcd-dist tarball with */sourceforge*. The contents of the tarball are then available through the path */sourceforge*.

2. Modify the settings in */sourceforge/template.conf*.

   This file follows perl syntax, therefore ensure that all variable/value associations are properly quoted and end with a comma where appropriate.

3. Modify the configuration section of the postinstall.sh.

   Determine whether you want the postinstall script to set up nss_ldap, which will cause the host the tarball resides on to authenticate from the SourceForge server's LDAP.

4. If you wish postinstall to set up nss_ldap, modify */etc/nsswitch.conf* to add 'ldap' to 'passwd', 'group' entries. Ldap does not have to appear as the first entry in each nis map.

5. Set the PVCS parameters in the template.conf of the SF SCM installation as follows:

   ```
   'PVCS_HOST_NAME' =>"sfsol1.t.com",
   'PVCS_HOST_IP' =>"11.1.20.1",
   'PVCS_PROJECT_ROOT_PATH' =>"/pvcsroot",
   'PVCS_PCLI_PATH' =>"/usr/pvcs/vm/solaris/bin/pcli",
   ```

   Make sure that the values in */sourceforge/etc/scripts/local.py* with regards to pvcs reflect the values you use to run the pvcs (pcli) binaries.

   In the above sample entry, */pvcsroot* is the folder in which all PVCS projects will be created. That folder must be created with 777 permission.

   ```
   mkdir /pvcsroot
   chmod 777 /pvcsroot
   ```

6. Copy the xmlrpcd shared secret from the SourceForge server to the following entry of template.conf.

   Copy from the SourceForge server:

   The xmlrpcd secret from the variable xmlrpc_shared_secret in file

   /sourceforge/etc/scripts/local.py

   Copy to the PVCS server:

   The xmlrpcd secret to the following variable of template.conf

   'XMLRPC_SHARED_SECRET' =>"password",

   For example:

   ```
   XMLRPC_SHARED_SECRET' =>"kFhQmEhEbXnVgGjSfRnEjHjFzBbBvLhG,
   ```

7. Ensure that the trusted_hosts file in */sourceforge/etc/xmlrpc/trusted_hosts* has the IP addresses of:

   The database server

   The SourceForge server

   The PVCS server

8. Ensure that the SCM server has connectivity to the database server. If using a Postgres DB ensure that the pg_hba.conf permits connection from the PVCS server.

9. If the LDAP server ('LDAP_HOST_NAME' parameter in template.conf) is running Linux and the PVCS server is running Solaris, create a group on the PVCS server “users” with the group id 100.

   ```
   groupadd -g 100 users
   ```

10. Make a directory users in /home.

    ```
    $ /etc/init.d/autofs stop
    $ vi /etc/auto_master
    ```

    Comment the line:

    ```
    #/homeauto_home-nobrowse
    $ /etc/init.d/autofs start
    $ mkdir /home/users
    ```

11. Run postinstall.sh.

12. Restart the Solaris machine.

13. Start xmlrpcd with the command:

    ```
    /sourceforge/etc/init.d/xmlrpcd.init start
    ```

Potential Issues

Oracle

If dirCreator faults with an Oracle connectivity error or xmlrpcd has problems connecting and logging into the database, check the following:

1. Verify that the Oracle client libraries are installed.
2. Run ldd on */sourceforge/python/lib/python2.1/site-packages/cx_Oracle.so*.
3. Make sure that the directory these files exist is in the crle path.

   Any libraries not found by ldd will cause connectivity problems.

PostgreSQL

Verify that PostgreSQL client libraries are installed. Run ldd on */sourceforge/python/lib/python2.1/site-package/psycopgmodule.so*.

PVCS Commands

This section provides a number of common PVCS commands and explains their use within the SourceForge framework. The steps are explained with a set of sample commands.

The following assumptions are made:

- */pvcsroot* is the folder under which all SourceForge PVCS projects are created
- The users are tuser14 and tuser15
- The SourceForge PVCS project is tpproj22
- The project tpproj22 is created by tuser14
- tuser15 is a project member with the role “Developer”

Sample PVCS usage scenario

1. Login as ProjectAdmin/Developer/Member with SCM access on pvcs server.

   ```
   cvssh-2.03$
   ```

2. Create project root level folders for the project.

   ```
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli createproject
   -pr/pvcsroot/tpproj22 module1
   ```

   This command creates a folder */pvcsroot/tpproj22/archives/module1*.

3. Add files inside a project root level folder.

   (Two files afc.h and afc.c are created in the home directory first.)

   ```
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli addfiles -pr/pvcsroot/tpproj22
   \
   -pp/module1 -t"import comment" afc.*
   ```

4. Check out files for editing.

   ```
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli get -pr/pvcsroot/tpproj22 \
   -pp/module1 -a/home/users/t/tu/tuser14 -l afc.h
   ```

   Here it is assumed that we are checking out to the user’s home directory

   ```
   /home/users/t/tu/tuser14
   ```

5. Commit the file after modification.

   ```
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli put -pr/pvcsroot/tpproj22 \
   -pp/module1 -a/home/users/t/tu/tuser14 -m"Commit comment" \
   -ym afc.h
   ```

   Here it is assumed that we are checking in from the user’s home directory

   ```
   /home/users/t/tu/tuser14
   ```

6. Checkout a file and checkin by another user.

   Login as tuser15 on pvcsserver.

   ```
   cvssh-2.03$
   ```

   Checkout edit and checkin afc.h

   ```
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli get -pr/pvcsroot/tpproj22 \
   -pp/module1 -a/home/users/t/tu/tuser15 -l afc.h
   cvssh-2.03$ vi afc.h
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli put -pr/pvcsroot/tpproj22 \
   -pp/module1 -a/home/users/t/tu/tuser15 -m"Commit comment" \
   -ym afc.h
   ```

   We are assuming that the user is using the tuser15's home directory

   ```
   /home/users/t/tu/tuser15 for checkout edit and checkin
   ```

7. Checkout and checkin a file using SCM - Tracker integration

   Login as tuser14 on the PVCS server.

   ```
   cvssh-2.03$
   cvssh-2.03$ /usr/pvcs/vm/solaris/bin/pcli get -pr/pvcsroot/tpproj22 \
   -pp/module1 -a/home/users/t/tu/tuser14 -l afc.h
   cvssh-2.03$ vi afc.h
   cvssh-2.03$ export PCLIBIN=/usr/pvcs/vm/solaris/bin/pcli
   cvssh-2.03$ export PCLI_PR=/pvcsroot/tpproj22
   cvssh-2.03$ export SFPROJ=tpproj22
   cvssh-2.03$ /sourceforge/scm_integration/bin/pvcswrap put -pp/module1 \
   -a/home/users/t/tu/tuser14 afc.h
   ```

   We are assuming that the user is using tuser14's home directory.

   ```
   /home/users/t/tu/tuser14 for checkout edit and checkin
   ```

   Note: The script pvcswrap is a wrapper for displaying the SCM Tracker integration template for the user when a checkin is made. You must set the environment variables PCLIBIN, PCLI_PP and SFPROJ.

Installing Multiple SCM Servers

SourceForge 3.4 allows you to set up multiple SCM servers of the same or different types. For example, you can set up 1 CVS, 2 PVCS, and 3 ClearCase servers (or any desired combination.) This facilitates environments where different organizations use different SCM servers.

To install multiple PVCS or ClearCase servers, follow the instructions listed in this manual for each server.

Since CVS is installed with SourceForge, follow the instructions below to install multiple CVS servers.

To install multiple CVS servers:

1. Untar the SourceForge installation tar on the machine where the additional CVS server is to be installed.
2. cd to the directory with setup.sh
3. Copy the file */sourceforge/bin/sfeeimd/localinst.xml* from the SourceForge server to the current directory.
4. Execute the following command:

```
./setup.sh cvs
```

5. When prompted by the installer screen, enter the additional CVS server's FQDN and IP address.

The SourceForge installer will allow you to connect your SourceForge installation to either zero or one of each type of SCM server. If you want to connect your SourceForge installation to additional SCM servers, you must connect them manually after installation. See the *SourceForge Enterprise Edition 3.4 User Guide* for detailed instructions.

SourceForge Installation Worksheet

Collect the following information before proceeding to the next chapter on installing SourceForge. The information required during the installation process is shown in Table 4.

Table 4. Required Information for the Installation Wizard

Required	Description	Comments
SourceForge server	The fully qualified domain name (FQDN) and IP address for the SourceForge server.	For server to server communication.
CVS server	The fully qualified domain name (FQDN) and IP address for the CVS server.	For server to server communication.
LDAP server	The fully qualified domain name (FQDN) and IP address for the LDAP server.	For server to server communication.
Backend server (Log server)	The fully qualified domain name (FQDN) and IP address for the Backend server.	For server to server communication.
List server	The fully qualified domain name (FQDN) and IP address for the List server.	For server to server communication.
List server alias (optional)		For server to server communication. Secondary MX server.
ClearCase Server	The fully qualified domain name (FQDN) and IP address for the ClearCase server.	For server to server communication.
PVCS Server	The fully qualified domain name (FQDN) and IP address for the PVCS server.	For server to server communication.
SFEE Admin Password	The administrator password for the SourceForge administrator.	Case sensitive. Must be between six to eight alphanumeric characters in length.
LDAP Admin Password	The administrator password for the LDAP administrator.	Case sensitive. Must be between six to eight alphanumeric characters in length.
Mailman Password	The password for Mailman.	Case sensitive.
Database server	The fully qualified domain name (FQDN) and IP address for the database server.	

Table 4. Required Information for the Installation Wizard (continued)

Required	Description	Comments
Database Instance	For Oracle, the `tnsname`. For PostgreSQL, the database instance. For DB2, the alias name	
Database User Name	For Oracle, the user name is the database schema name that you selected. Refer to Step 4 for preparing Oracle described on "To prepare Oracle for SourceForge" on page 29. For PostgreSQL, use the user name that your database administrator has set up to connect to the database. For DB2, use the user name that your database administrator has set up as the DB2 instance owner.	
Database Password	For PostgreSQL, use the password that your database administrator has set up to connect to the database. For DB2, use the password that your database administrator has set up for the DB2 instance owner.	Case sensitive.
Admin Email Address	The email address for the SourceForge administrator.	E-mail address of the administrator.

Installing and Setting up Secure Shell (SSH)

This installation procedure is for users who do not have SSH installed on their system.

Installing SSH

Almost any command-line version of SSH will work with WinCVS, but each works slightly differently. The recommended version is available at:
http://sourceforge.net/projects/sfsetup

To install SSH

1. Download the SSH zip file from
   http://www.sourceforge.net/projects/sfsetup
2. Extract the contents of the zip file into a directory

   For example: sf-ssh on a local drive such as C
3. Note the path (*C:\sf-ssh*) to the SSH installation directory.

Setting Up the Environment for SSH

You need to set up three environment variables: HOME, CVS_RSH, and CVS_EDITOR.

To set up the environment for SSH

1. Set a HOME environment variable where the .ssh directory can be created.

   For example: *C:\sf-ssh*
2. Set a CVS_RSH environment variable to the full path to ssh.exe.

   For example: *C:\sf-ssh\ssh.exe*.
3. Set a CVS_EDITOR environment variable to your text editor or *notepad.exe* (if you use the command-line version of CVS.)

   For example: *notepad.exe*.

Authenticated PServer

Overview

Authenticated PServer is a protocol for communicating with CVS, instead of using SSH.

Because the data transfer is unencrypted, there is a possible security risk using authenticated PServer. Therefore, it is recommended that you use this protocol only for internal transfer of data or within trusted networks.

The benefits of using authenticated PServer are:

- Easier to setup because there is no ssh client for CVS users
- More web-based clients available compared to the one available for SSH in CVS (WinCVS)

SourceForge does not provide support for unauthenticated or anonymous PServer access.

Security Note

If no projects will use it, disable the password-authenticating CVS server (pserver) for optimum security. The PServer is disabled by default on Solaris system. On Linux systems, modify the "disable" variable in the /sourceforge/etc/cvspserver configuration file as required. To disable it, assign the value "yes" to the "disable" variable in the configuration file.

Enabling PServer Support for SourceForge on Linux

Adding /etc/xinetd.d/cvspserver

This file is not required unless PServer is enabled. When PServer is enabled:

- This variable must be set to "no"
- The inetd must be restarted.

The file itself should reside in */sourceforge/etc/cvspserver*. At installation time, a link is created from */etc/xinetd.d/cvspserver*.

File Contents

```
service cvspserver

{

        disable = yes

        socket_type = stream

        wait = no

        user = root

        server = /sourceforge/cvs/bin/cvs

        server_args = -T/tmp -f pserver --allow-root=/sourceforge

        log_on_failure  += USERID

}
```

Enabling PServer Support for SourceForge on Solaris

On Solaris, PServer must registered in /etc/services and /etc/inetd.conf.

1. Add the following line to /etc/services:

   ```
   cvspserver 2401/tcp #CVS Pserver
   ```

2. Add the following to the end of /etc/inetd.conf::

   ```
   # CVS PSERVER SUPPORT
   cvspserver stream tcp nowait root /sourceforge/cvs/bin/cvs -T/tmp -f
   pserver --allow-root=/sourceforge
   ```

   Note: the above two lines should be entered without a line break, as shown.

3. Restart inetd.

   Run the following in a shell script:

   ```
   pkill inetd
   /usr/sbin/inetd -s
   ```

4. Check that PServer service has started

   Run the following in the shell script:

   ```
   netstat -na | grep 2401
   ```

   You should see something similar to:

   ```
   *.2401 *.* 0      0 24576 0 LISTEN
   ```

   If you do not see this line then inetd has not been started properly. Try Step 3 again.

CHAPTER 3
Installing SourceForge

This chapter describes how to install SourceForge 3.4 using the installation wizard.

It includes the following topics:

- Installing SourceForge
- Logging in for the first time

Installing SourceForge

You must complete the pre-installation tasks described in the previous chapter before proceeding to install SourceForge.

Initial Tasks

This section describes the initial tasks to perform before installing SourceForge.

Configuration

- Include the fully qualified domain name (FQDN) of your machine in *etc/hosts.*
- Include the local domain name in *etc/resolv.conf*.

Understanding SourceForge Roles and Role Dependencies

The SourceForge installation environment is divided into the following “roles”:

- SourceForge (SFEE) codebase server
- Backend server
- LDAP server
- List server
- SCM server

Each of these roles has a distinct functionality.

➪ For detailed descriptions of SourceForge roles, see “Roles Defined” on page 2.

Distribution of SourceForge Roles

All roles can be installed on one machine or distributed across a maximum of five (or six) machines in your environment.

Multiple roles can reside on one machine in your installation.

- SCM, List, and Backend depend on SFEE

  The SCM server is normally not installed on the same server as the five required roles. In fact, it is not possible to have them on the same server if SourceForge is installed on Linux and the SCM is ClearCase.
- SFEE depends on LDAP and DB

For each server, you will need to know the server’s fully qualified domain name (FQDN) and the IP address.

The following ports are used to be accessible for all users:

Table 5. SourceForge Roles and Ports Accessible for All Users

Role	Ports
SourceForge (SFEE) server	80 (optional) 443
SCM server	22 80 (optional) 443
List server	25 80 (optional) 443

Preparing the Servers

You can install SourceForge 3.4 using up to five (or six) different servers. However, with the installation wizard you can include only five servers in an installation. You need to run the setup program from the CD on each of the servers you will use, prior to running the installation wizard.

Note: Do not install *sshd* (Secure Shell Daemon) on the SourceForge servers. SourceForge comes with its own *sshd* binary.

To prepare each of the servers

1. Select a staging directory with a minimum of 400MB of available space.
2. Insert the SourceForge 3.4 installation CD in the CD-ROM drive.
3. Log in to the server as root.
4. Mount the SourceForge 3.4 installation CD.
5. Change the path to the staging directory.
6. Copy files from the CD into the staging directory with the following command:

   tar xzvf /mnt/cdrom/sfee32linux.tgz (for Linux)

   tar xvf /cdrom/cdrom/sfee32solaris.tar (for Solaris)
7. Run `./setup.sh` from the staging directory.

   The installer displays the welcome screen.

   First, the installer checks if the $DB2_HOME variable is defined (when installing SourceForge on DB2.)

   a. If the $DB2_HOME variable is not defined (or does not point to a directory), the installer displays the message:

      $DB2_HOME is not set or set to an invalid directory. Please correct. Aborting the installer.

      The installer aborts and returns the system prompt.

   b. If the $DB2_HOME variable is defined, the installer displays the message:

      DB2_HOME is set to <*your DB2_HOME directory*>. Is this correct? [yes|no]

      If you respond with **no**, the installer aborts with the message:

      Please set the $DB2_HOME variable accordingly. Aborting the installer.

      You will see the following message.

      Will you be configuring the SFEE install from this machine? [y/n]?

8. If you want to install SourceForge on a remote server, type `n` and press **Enter**.

   This process will copy the necessary installation files onto the server and prepare for the setup process.

   If this is the server where you want to run the installation wizard, type `y` and follow the instructions in“Running the Installation Wizard” on page 74.

9. Remove the SourceForge 3.4 installation CD and repeat this process on the next server.

Any server in the SourceForge environment can be used to run the installation wizard. Make sure that you only select one machine to be used as the SourceForge installer machine.

Running the Installation Wizard

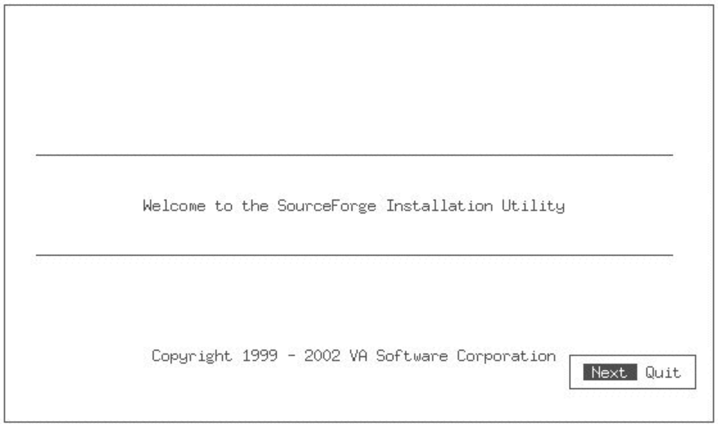
This section describes the installation wizard that will install the SourceForge 3.4 software. Regardless of how many servers are in your SourceForge environment, run this installation wizard from only one of the servers. See “Preparing the Servers” on page 72. For additional help at each point in the installation process, click F1.

To install SourceForge

1. Type **y** and press **Enter** at the following message on the server where you will install SourceForge 3.4:

```
Will you be configuring the SFEE install from this machine? [y/n]?
```

The Welcome screen displays.

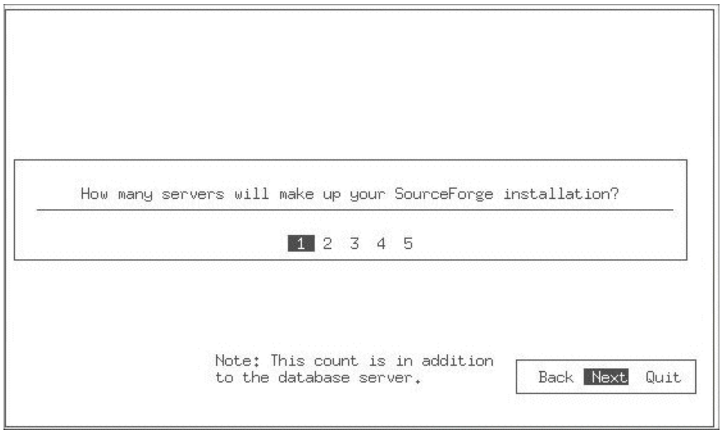


Welcome to the SourceForge Installation Utility

Copyright 1999 - 2002 VA Software Corporation

2. Select **Next** and press **Enter** to continue.

The next screen will prompt you for how many servers you wish to include in your SourceForge environment. You can include up to five servers.



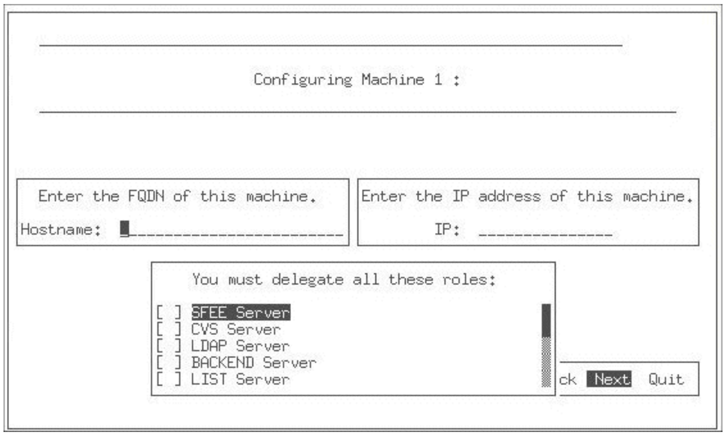
How many servers will make up your SourceForge installation?

1 2 3 4 5

Note: This count is in addition
to the database server.

3. Select the number of servers in your environment and select **Next**.

   The next step is to configure each of the servers in your SourceForge environment. The wizard will display the configuration screens for each of the servers.



```
                    Configuring Machine 1 :

 Enter the FQDN of this machine.    Enter the IP address of this machine.
Hostname: █_______________________          IP:  _______________

                You must delegate all these roles:
            [ ] SFEE Server
            [ ] CVS Server
            [ ] LDAP Server
            [ ] BACKEND Server
            [ ] LIST Server                          ck  Next  Quit
```

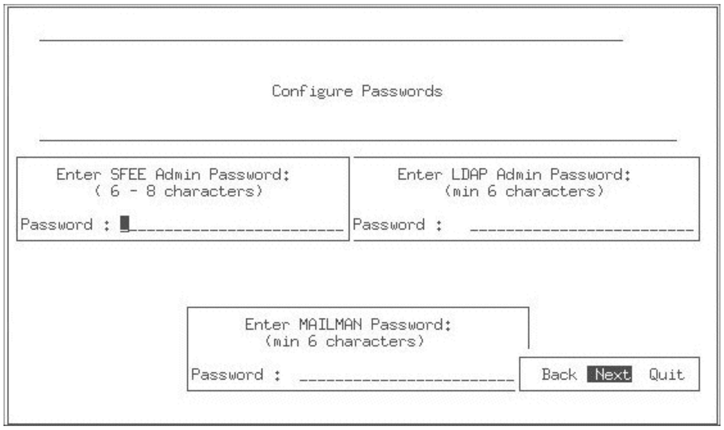
4. Enter the FQDN (for example, *sourceforge.yourcompany.com*).
5. Enter the IP address for this server.

   The Installer will check to make sure that the FQDN resolves to the given IP address.
6. Select the role(s) you wish to assign to this server using the UP or DOWN arrow keys.

   You can select more than one role. However, do not duplicate your roles on different servers.
7. Press **Enter** once you've selected the roles.

   The wizard displays the roles you have already assigned.
8. Select **Next** to continue.
9. Repeat step 4 through step 8 for all the servers in your environment.

The next screen prompts you to enter the passwords for the SourceForge administrator, the LDAP administrator and the Mailman.



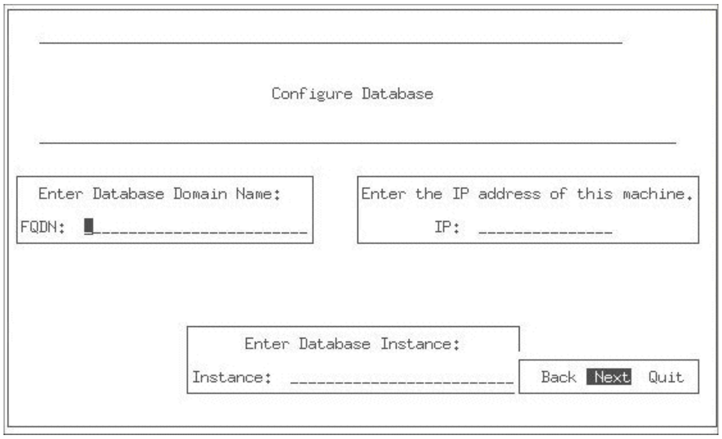
```
Configure Passwords

Enter SFEE Admin Password:          Enter LDAP Admin Password:
( 6 - 8 characters)                 (min 6 characters)
Password : _______________________  Password :  ________________________

                Enter MAILMAN Password:
                (min 6 characters)
                Password :  ________________________   Back  Next  Quit
```

10. Enter each of the requested passwords and select **Next**.

    You need to confirm each password by entering it a second time.

    Note: Passwords must be a combination of at least six characters or numbers.

    The next two screens prompt you for the database connection information.

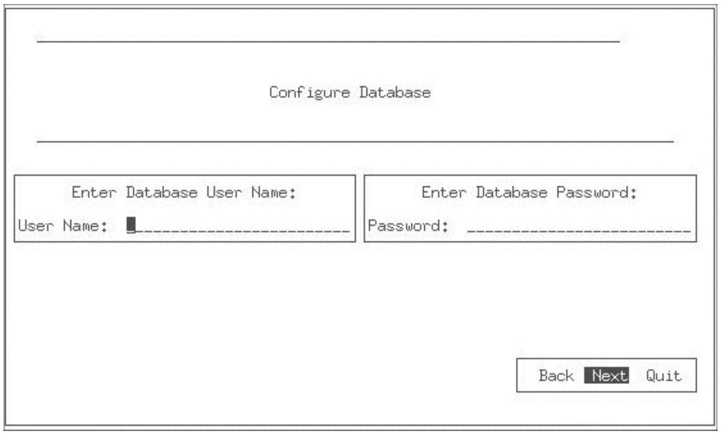


```
Configure Database

Enter Database Domain Name:         Enter the IP address of this machine.
FQDN: _______________________             IP:  ______________

                Enter Database Instance:
                Instance:  ________________________   Back  Next  Quit
```

11. Enter the database server’s FQDN (for example, sf-db.yourcompany.com), the IP address, the database instance, and select **Next**.

    Important: For Oracle, the database name is th TNSNAME. For PostgreSQL, the database name is the name that you created as part of the SourceForge database creation. see “PostgreSQL” on page 32. For DB2, the dbname is the alias name.

The next screen prompts you for the database user name and password.



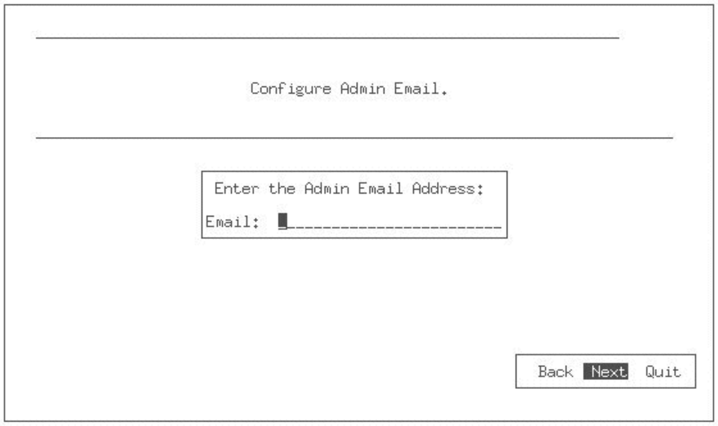
```
Configure Database

Enter Database User Name:
User Name: _________________________

Enter Database Password:
Password: _________________________
```

12. Enter the database user name and password and select **Next**.

Important: For Oracle, the user name is the database schema name and password that you selected. See “Creating the target tablespace” on page 28. For PostgreSQL, use the user/password combination that your database administrator has set up to connect to the database.

This information is checked in the setup step of the wizard to verify that the connection information is correct and the servers can connect to the database.

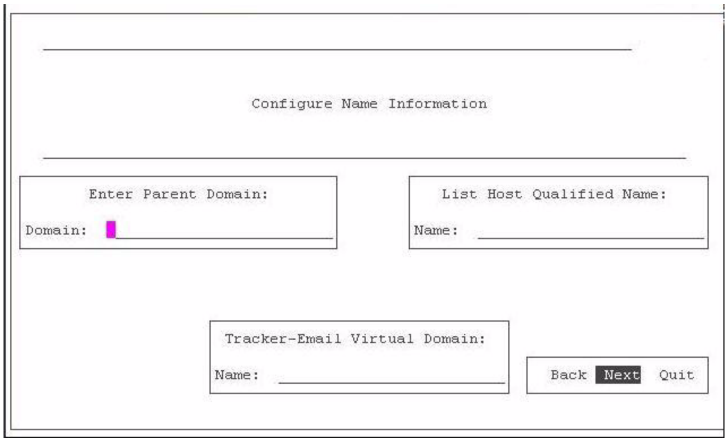
Next, you are asked for the email address of the SourceForge administrator.



```
Configure Admin Email.

Enter the Admin Email Address:
Email: _________________________
```

13. Enter the email address for the SourceForge administrator.

Next, you are asked for the domain name of the parent server and an alias list host qualified name for the list server.


Configure Name Information
Enter Parent Domain:
Domain:
List Host Qualified Name:
Name:
Tracker-Email Virtual Domain:
Name:
Back
Next
Quit

14. Enter the domain name for the parent server.

The parent domain name is the same as the FQDN without the actual machine name. For example, the parent server for the FQDN *sourceforge.dev.yourcompany.com* is *dev.yourcompany.com,* where *sourceforge* is the actual machine name. This can vary depending upon how your domain name server is set up.

15. Enter an alias for the list server and select **Next**.

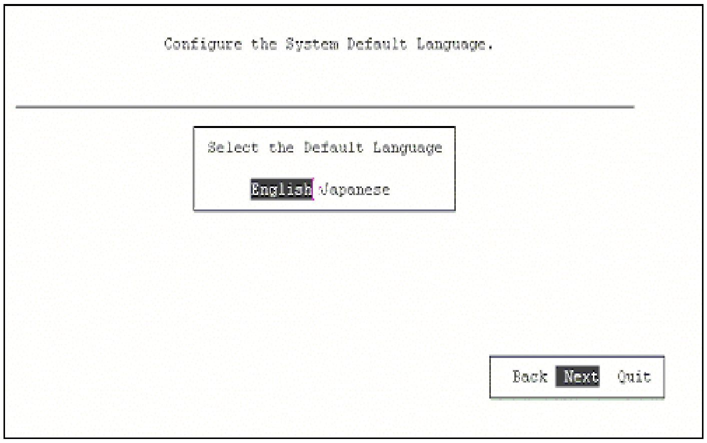
By default, all email sent from the SourceForge system will appear to be from the hostname assigned to the Mailing List system. If you would like mail to appear to be from a different domain, you need to configure your DNS system to have a secondary MX name pointing to the Mailing List host. This secondary MX name is the alias.

16. Enter a virtual domain for tracker email integration. This virtual domain must exist in DNS.

The recommended default is '*tracker*' prefixed to the FQDN.

For example, *tracker.sf.yourcompany.com*.

Next, you are asked to choose the system default language.



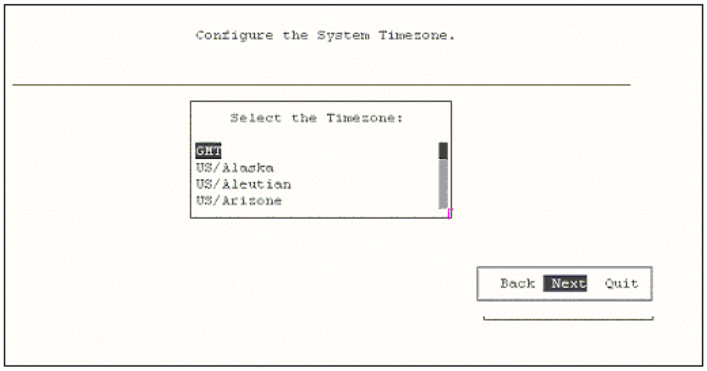
Configure the System Default Language.

Select the Default Language

English Japanese

17. Select the default language **English** and select **Next**.

The System Timezone screen is displayed.



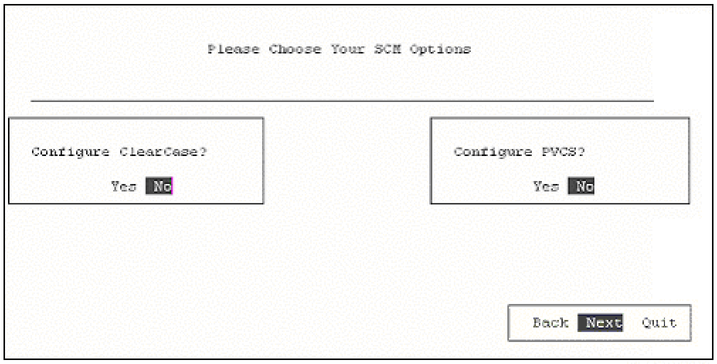
Configure the System Timezone.

Select the Timezone:

GMT
US/Alaska
US/Aleutian
US/Arizone

18. Select the Timezone from the list and select **Next**.

Next, you are asked to choose your SCM (Software Configuration Management) option.



Please Choose Your SCM Options

Configure ClearCase?

Yes [✓ No]

Configure PVCS?

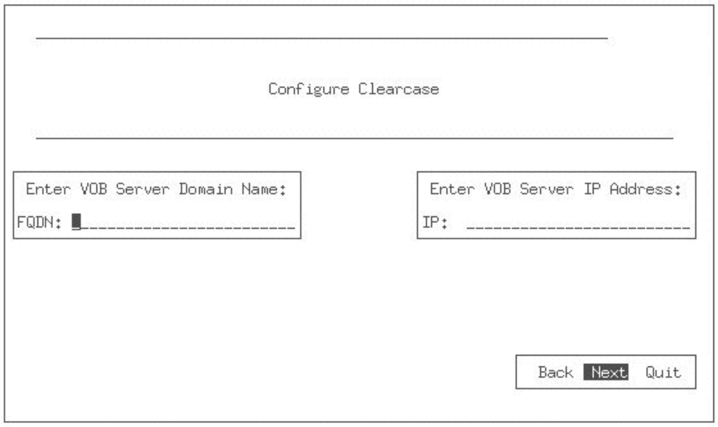
Yes [✓ No]

19. Select ClearCase or PVCS by choosing **Yes** then select **Next**.

If you are using the SourceForge default CVS option, select **Next**.

The Configure screen will display.

For ClearCase (VOB):



Configure Clearcase

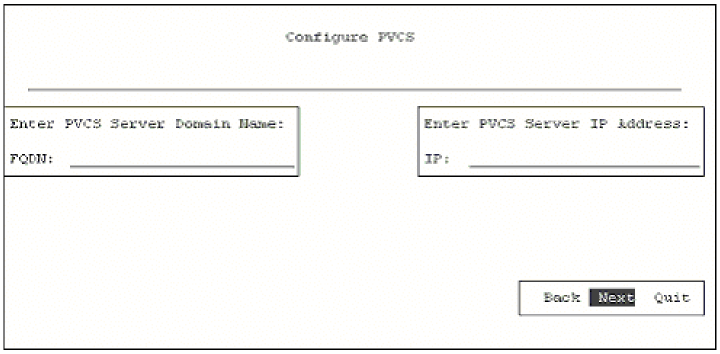
Enter VOB Server Domain Name:

FQDN: ________________________

Enter VOB Server IP Address:

IP: ________________________

For PVCS:



Configure PVCS

Enter PVCS Server Domain Name:

FQDN: ____________

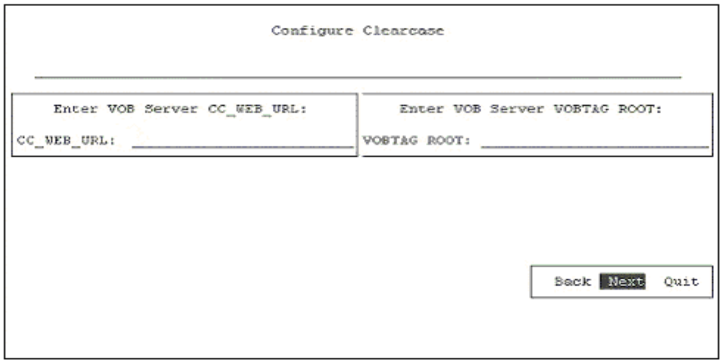
Enter PVCS Server IP Address:

IP: ____________

20. Enter the server domain name and IP address and select **Next**.

21. Enter the project root path for ClearCase/PVCS.

For ClearCase:



Configure Clearcase

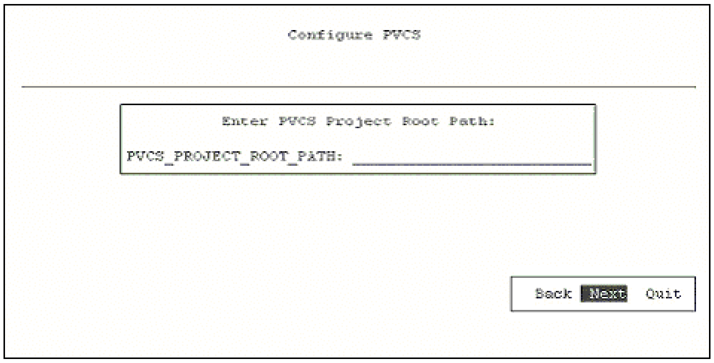
Enter VOB Server CC_WEB_URL:

CC_WEB_URL: ____________

Enter VOB Server VOBTAG ROOT:

VOBTAG ROOT: ____________

For PVCS:

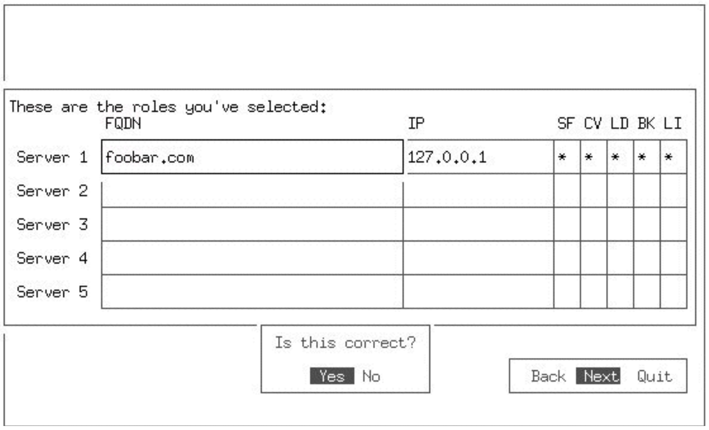


Configure PVCS

Enter PVCS Project Root Path:

PVCS_PROJECT_ROOT_PATH: ____________

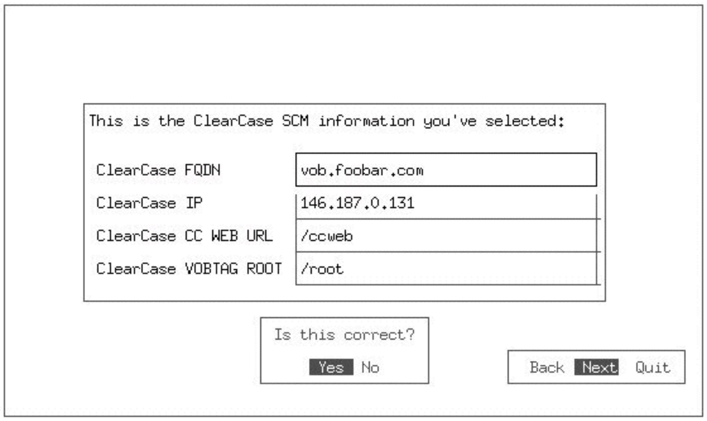
22. The next screen will confirm that you have assigned the correct roles for the servers.



These are the roles you've selected:

	FQDN	IP	SF	CV	LD	BK	LI
Server 1	foobar.com	127.0.0.1	*	*	*	*	*
Server 2							
Server 3							
Server 4							
Server 5							

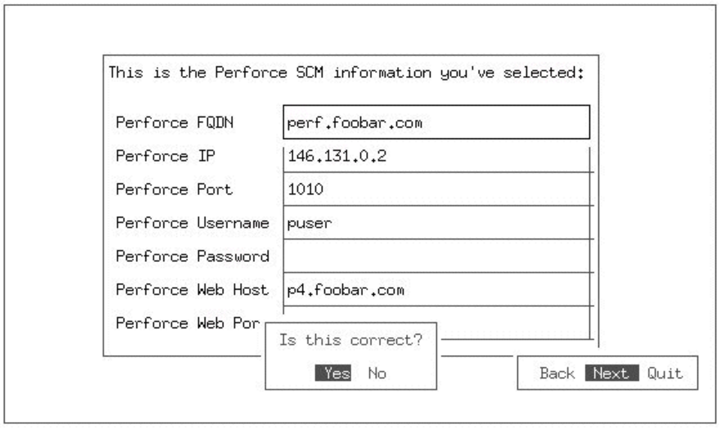
23. Click **Yes** to confirm your choices. Click **No** to return to the Machine Selection screen.



This is the ClearCase SCM information you've selected:

ClearCase FQDN	vob.foobar.com
ClearCase IP	146.187.0.131
ClearCase CC WEB URL	/ccweb
ClearCase VOBTAG ROOT	/root

The next screen prompts you to confirm the database information that you have entered.



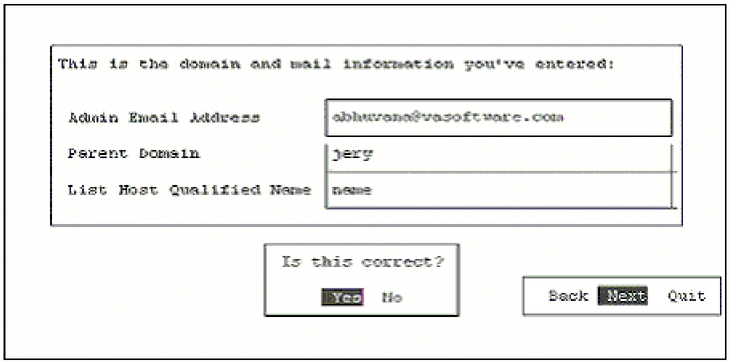
This is the Perforce SCM information you've selected:

Perforce FQDN	perf.foobar.com
Perforce IP	146.131.0.2
Perforce Port	1010
Perforce Username	puser
Perforce Password	
Perforce Web Host	p4.foobar.com
Perforce Web Por	

Is this correct?

24. Click **Yes** to confirm your choices. Click **No** to return to the Database Information screen.

The next screen prompts you to confirm the domain and mail information that you have entered.



This is the domain and mail information you've entered:

Admin Email Address	abhuvana@vasoftware.com
Parent Domain	jery
List Host Qualified Name	name

Is this correct?

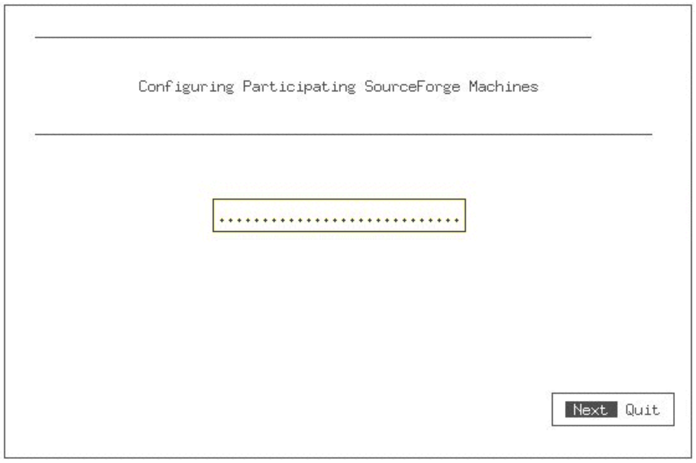
25. Click **Yes** to confirm your choices. Click **No** to return to the Domain and Mail Information screen.

The wizard now connects to all the servers and starts the setup process.

26. The next screen verifies Language and Timezone entered earlier.

27. The next screen verifies ClearCase/PVCS SCM information entered earlier.

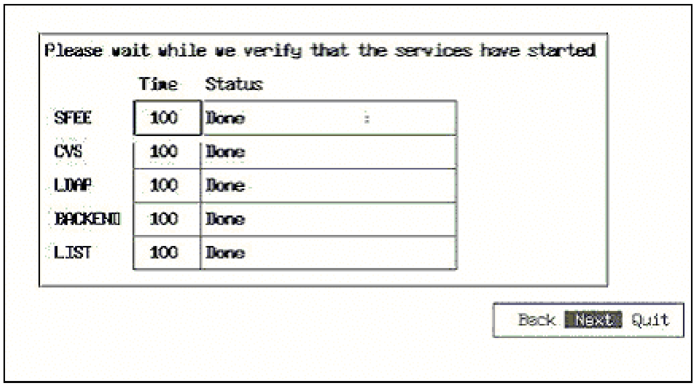
28. The next screen verifies the progress of your installation.



Configuring Participating SourceForge Machines

....

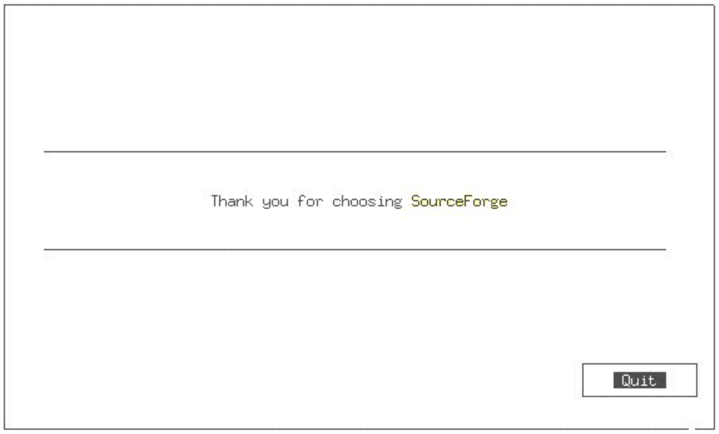
29.The next screen verifies the services started.



Please wait while we verify that the services have started

	Time	Status
SFEE	100	Done
CVS	100	Done
LDAP	100	Done
BACKEND	100	Done
LIST	100	Done

30. The installation is complete when you see the following screen.



Thank you for choosing SourceForge

If you encounter any connection errors during the installation, quit the wizard and rerun.

Verifying the Installation

The SourceForge installation wizard logs its action and output to the following directory:

/sourceforge/log/install

Review these files to make sure that they do not contain any errors. In case of errors, contact VA Software Technical Support at:

E-mail: *support-sf@vasoftware.co*m
Knowledge Base: *https://kb.vasoftware.com*
Phone: *1-877-825-4689*, option 2

Logging in for the First Time

Once the installation has completed successfully, you can log in to SourceForge. The installation process creates one SourceForge administrator user.

To log in for the first time

1. Point your browser to the server that you selected as the SFEE server.
2. Enter the login information.

   `Login:` **`admin`**

   `Password:` *<The password you selected for the SourceForge administrator during the installation>.*

You can now manage the SourceForge application with administrator privileges.

⇨ For details on using SourceForge as an application administrator, refer to the *SourceForge Enterprise Edition 3.4 User Guide*.

CHAPTER 4
System Maintenance

This chapter provides information on the following topics:

- System Administration
- SourceForge Directories and Files
- Statistics-Related Logs

System Administration

System administration responsibilities consist of:

- Backing up the log and data application files on a regular (at least, daily) basis.
- Locating system and error logs in the SourceForge log directories and the SourceForge-wide Administration facilities log. (There is a link to this log from the SourceForge Administration page.)

System Backup and Restoring

Initial—After the installation has been completed, configure your back-ups to incremental backups of the systems in their entirety in case it becomes necessary to backup any one of the machines due to a catastrophic failure. The backup should be a snapshot of the machines storing everything from the "/"directory and should take place once a day or once every two days.

Daily—It is desirable to stop SourceForge processes before running backups, in order to ensure a coherent backup set. For data updates, backup all the files and directories from the SourceForge log files. Also, you must back up the SourceForge application files shown in the following table.

Table 6. Application Data Files

Server	File	Description
SourceForge	*/sourceforge/log/*	All log files
CVS role server	*/sourceforge/cvsroot*	All CVS files
SFEE role server	*/sourceforge/var/frs_files*	All uploaded files
LDAP role server	*/sourceforge/openldap/var*	LDAP user database files
MAILMAN role server	*/sourceforge/mailman/*	All mailing list data
SCM server	*/sourceforge/scm_integration/template_root*	All template files for SCM check-in activity
SCM server	*/home/users* */home/groups*	SSH keys

SourceForge Directories and Files

The following table provides a reference to all the files and directories stored in */Sourceforge/*.

Table 7. SourceForge Directories and Files

Directory or File	Type
/apache	3rd party
/backend	Cron jobs
/BerkeleyDB	3rd party
/bin	SourceForge binaries
/cvs	3rd party
/cvs-active	3rd party
/cvsroot	3rd party
/docbook-xsl	3rd party
/enscript	3rd party
/etc	SourceForge
/exim	3rd party
/Fop	3rd party
/gettext	3rd party
/home	SourceForge
/users	SourceForge
/j2re	3rd party
/lib	SourceForge
/libmcrypt	3rd party
/log	SourceForge and 3rd party
/mailman	3rd party
/man	3rd party
/openldap	3rd party
/openssh	3rd party
/openssl	3rd party
/perl	3rd party
/php	3rd party
postgresql	3rd party (Only for the PostgreSQL version of SourceForge.)
/postinstall	SourceForge

Table 7. SourceForge Directories and Files (continued)

Directory or File	Type
/python	3rd party
/scm_integration	SourceForge
/scripts	SourceForge
/sfcache	SourceForge
/sfee	SourceForge
/sys	SourceForge
/var	SourceForge
/vcs	3rd party
/xmlrpcd	SourceForge

Statistics-Related Source Logs

This section lists all backend logs (and other files) that serve as the input data for statistic-processing cron jobs in the Cronjob Suite. The backend log sources, processing paths, and the cron jobs they are fed into are listed in the table below.

Definitions

Backend—Server that runs a specific facility under the control of the main site, such as CVS, downloads, etc.

Middleend—Server that runs cron jobs which import logs to the database. For SFEE, this is the main server.

Frontend—Server that runs statistics processing cron jobs. For SFEE, this is the main server.

Table 8. Statistics-Related Source Logs

Source Server/ App.	Source Location	Form of Log/ Growth	Receiving Server	Receiving Location	Means of Transfer	To Be Processed By
CVS backend, CVS, SCM server	*/cvsroot/*/ CVSROOT/ history* *SCM history commands*	Continuous log, new entry added for each SCM add, check-out, or commit performed	CVS backend	*/sourceforge/log/cvs/ %Y/%m/%d/ cvs_history.log*	*xmlrpc*	*cronjobs/ middleend/cvs/ db_stats_cvs_ history.py*
Download HTTP server, Apache	*/sourceforge/ log/sfee/ %Y/%m/%d/ httpd_access .log*	Daily log, per special Apache module. New entry on each project file download attempt - successful, partial, or unsuccessful	middleend	*/sourceforge/log/sfee/ %Y/%m/%d/ httpd_access .log*	rsync	*cronjobs/ middleend/ download/ stats_http_ logparse.py*
%Y, %m, %d—4-digit year, 2-digit month, and 2-digit day numbers, respectively.						
DB server, target DBMS	*activity_log table*	Daily log as a DBtable, rotated by cronjob. New row on each access to a page of mainsite (inserted by web frontend)	frontend	*activity_log_ old table*	*cronjobs/ frontend/ cleanup/ rotate_ activity.php*	*cronjobs/ frontend/stats/ site_stats.php*
Dump logs	*/sourceforge/ cronjobs/ scripts/ frontend/ cleanup*	Monthly cron for cleaning up db tables and dumping the data in .csv format	frontend	*/sourceforge/log/ facilities/*	*/sourceforge/ cronjobs/ scripts/ frontend/ cleanup*	*/sourceforge/ cronjobs/ scripts/ frontend/ cleanup/ dump_logs.php*

VA Software PostgreSQL Support Offerings

SourceForge Enterprise Edition uses an external database as a repository for application data and development artifacts. SourceForge supports PostgreSQL along with Oracle 9i and IBM DB2.

This section outlines support options for the PostgreSQL database. PostgreSQL makes an ideal choice for organizations that do not possess Oracle or DB2-specific database administration resources in-house.

VA Software provides extensive support for the SourceForge application for customers with valid maintenance agreements. However, standard SourceForge support does not include database support, maintenance or management services. These activities remain a customer responsibility.

- VA Software support will answer general usability and maintenance questions.
- For organizations looking to manage their own database installations, VA also makes a best practices and technical instructions document available.
- For organizations looking to outsource database management, VA Software provides a reasonably-priced, flat-fee administration package.

In all cases, a VA Software sales manager can help an organization understand its options and plan the best database maintenance strategy, based on its specific requirements, skills, and resources.

Maintenance Service Options

Organizations looking to outsource database management services have two options: contract VA Software or obtain the services of a different external organization.

VA Software Services

VA Software provides a reasonably-priced, flat-fee administration package. The PostgreSQL Database Support package includes backup, restoration testing and optimization services to:

- Ensure backups are configured correctly and running properly
- Ensure that backups can be restored (test restoration)
- Ensure that the database is optimized and running efficiently

In the event of database malfunction, this packaged service offering includes a full restore from the most recent database backup. In the event that a customer seeks to repair corrupt data in a production database (without reverting to a backup), given the complexity and uncertainly about such operations, such services are provided on a time and materials basis, at additional cost.

External Services

These or similar services may also be obtained through external support organizations. A listing of vendors providing third-party PostgreSQL support is available at: *http://techdocs.postgresql.org/companies.php*

For More Information

For additional information, or to review your specific requirements and obtain database support pricing information, please contact your SourceForge sales representative.

APPENDIX A
Digital Certificates

This appendix provides a synopsis of digital certificates under the following topics:

- Introduction
- Sample Shell Scripts
- Installing the Certificates

Introduction

Digital certificates have the following purposes:

- Validate the identity of the user and the server.
- Confirm the public key

Digital certificates are signed by a trusted agency, so that the user and the server are assured that they are communicating with whom they think they are. The trusted agency is called a Certificate Authority, and certificates are used for authentication.

Digital certificates associate a public key with the real identity of an individual, server, or other entity, known as the subject. Information about the subject contains the distinguished name and the public key. It also includes the identification and signature of the Certificate Authority that issued the certificate, and the period of time for which the certificate is valid. It may contain administrative information for the Certificate Authority's use, such as a serial number.

For additional information on digital certificates, visit:

http://www.modssl.org/docs.

Sample Shell Scripts

This section provides two sample shell scripts and configuration files for creating the files needed for ApacheSSL.

```
=========================================================================
Simple shell script and configuration file for creating the csr, crt, and
key files needed for ApacheSSL.

#
# sfee.cnf - OpenSSL configuration file
#
RANDFILE                = $ENV::HOME/.rnd
prompt                  = no

[ req ]
distinguished_name      = req_distinguished_name

[ req_distinguished_name ]
C               = US
ST              = California
L               = Fremont
O               = Temporary Certificate
OU              = SourceForge Enterprise Edition
CN              = $ENV::LONGNAME
```

```
======================================================================
```

```
Simple shell script and configuration file for creating the csr, crt, and
key files needed for ApacheSSL.

#!/bin/sh

# For SFEE 3.0
# OPENSSL=/sourceforge/openssl/bin/openssl
# - or -
OPENSSL=`which openssl`

if [ "$1" = "" -o "$2" = "" ]; then
        echo
        echo "Usage:    $0 [short name] [long name]"
        echo "Example:  $0 mybox mybox.mydomain.com"
        echo
        exit 0
fi

SHORT=$1
LONGNAME=$2

export LONGNAME
$OPENSSL req -newkey rsa:1024 -keyout $SHORT.key -out $SHORT.csr \
        -nodes -config sfee.cnf

$OPENSSL req -new -key $SHORT.key -out $SHORT.crt -x509 -days 365 \
        -config sfee.cnf

echo
"**********************************************************************"
echo
echo "A private key, certificate request and self signed certificate"
echo "have been created for $LONGNAME."
echo
echo "Please copy these files into /sourceforge/etc/apache/
ssl.[crt,csr,key]"
echo "and edit /sourceforge/etc/apache/httpd.conf accordingly."
echo
echo "The $SHORT.csr file should be given to the customer so they can use"
echo "it to get a signed certificate from Verisign or a similar CA."
echo
echo "Remember to STOP apache and then START it again to reload the new
cert,"
echo "simply RESTARTING it will not work and apache will die."
echo
echo
"**********************************************************************"
```

===

Installing the Certificates

Installing the certificates involves copying the correct files into */sourceforge/conf/apache/ssl.[key|crt|csr]* and updating the *httpd.conf* file with the locations of new files.

The procedure is identical for both RedHat Advanced Server 2.1 and SUN Solaris.

3. The script generates three files:

   a key (*.key*), self-signed certificate (*.crt*), and a certificate signing request (*.csr*)
4. The key (*.key*) and the self signed certificate (*.crt*) are installed into the configuration directory, and the certificate signing request (*.csr*) is sent to the Certifying Authority (CA).

   Certifying Authority (CA) sends you a new CA-signed certificate (*.crt*) which can overwrite the previously generated certificate.

Note: Changing keys/certificates requires an Apache start/stop task, not a restart.

Index