Legal Notices

There are various product or company names used herein that are the trademarks, service marks, or trade names of their respective owners, and March Hare Software Limited makes no claim of ownership to, nor intends to imply an endorsement of, such products or companies by their usage.

This document and all information contained herein are the property of March Hare Software Limited, and may not be reproduced, disclosed, revealed, or used in any way without prior written consent of March Hare Software Limited.

This document and the information contained herein are subject to confidentiality agreement, violation of which will subject the violator to all remedies and penalties provided by the law.

LIMITED WARRANTY.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, March Hare Software Limited AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, WITH REGARD TO THIS DOCUMENT, AND ANY ADVICE OR RECOMMENDATION CONTAINED IN THIS DOCUMENT.

NO OTHER WARRANTIES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL March Hare Software Limited OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE FOLLOWING DOCUMENTATION INCLUDING ANY RECOMMENDATION OR ADVICE THERIN, EVEN IF March Hare Software Limited HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY CASE, March Hare Software Limited’s ENTIRE LIABILITY UNDER ANY PROVISION OF THIS DOCUMENT INCLUDING ANY RECOMMENDATION OR ADVICE THERIN SHALL BE LIMITED TO THE GREATER OF THE AMOUNT ACTUALLY PAID BY YOU FOR THE DOCUMENT OR £5.00; PROVIDED.

© Copyright 2004 - 2021 March Hare Software Ltd

march-hare.com
sales@march-hare.com
Quick Install Guide

Getting the most out of this software requires that you are familiar with the theory as well as the practical. We the vendors strongly recommend that you first read the Fundamentals section of the eBook All About CVS or the Install Guide before proceeding.

Also see the Quick Configuration Guide and the Quick User Guide below.

If you are upgrading an existing CVS or CVSNT server to CVS Suite 2009 read Upgrading CVS Suite Server on Windows.

Ensure windows software is installed and updated using Windows Update. If the server will authenticate users to a domain, ensure the windows server is joined to that domain already.

Run the CVS Suite 2009 installer as a DOMAIN ADMINISTRATOR (or if not a part of a domain, any Administrator account).

After rebooting log into the same account and the CVS Suite 2009 Installer Wizard will automatically start to guide you to create a new repository.

Sales and Pre-Sales Support e-mail:
sales@march-hare.com
Quick Configuration Guide

Please also see above section Quick Install Guide.

Install using the installer - click on Server and Client for a single pc.

After reboot the wizard starts up & guides you to create a repository. If you skipped the wizard and need to run it again type:

```
wizard /server
```

After running the server wizard, start the CVS Suite Server control panel from the Programs->CVS Suite menu. The server control panel, Plugins tab allows you to configure integration with e-mail, build and defect tracking (Bugzilla).

After the server wizard, the client wizard runs and guides you to import files to the repository. If you skipped the wizard and need to run it again type:

```
wizard /server
```

After running the wizard, start the CVS Suite Studio from the Programs->CVS Suite menu. CVS Suite Studio is the main tool for working with CVS Suite Client.

Sales and Pre-Sales Support e-mail: sales@march-hare.com
Quick User Guide

Comprehensive tutorials for working with CVS Suite Studio, SQL Navigator, Eclipse, Visual Studio, TortoiseCVS and WinCVS are also available.

Use CVS Suite Studio:

• The help menu help and tutorial will show you a short video on how to use the software (no audio).

• On the left is the server & repository, on the right is your local computer directories.

• To add a directory to the server, drag the folder from the right and drop it on the left.

• To checkout a folder from the server and put it on the local computer, drag it from the left and drop it on the right.

Folders on the local PC that are connected to the CVS Suite Repository are automatically coloured with green ticks or orange crosses to indicate their state. Right clicking any controlled folder gives you plenty of options.

Sales and Pre-Sales Support e-mail:
sales@march-hare.com
Table of Contents

LEGAL NOTICES ..............................................................................................................................................C

QUICK INSTALL GUIDE ....................................................................................................................................E

QUICK CONFIGURATION GUIDE ......................................................................................................................F

QUICK USER GUIDE ..........................................................................................................................................G

TABLE OF CONTENTS .........................................................................................................................................1

PART I – THEORY..................................................................................................................................................3

FUNDAMENTALS ..................................................................................................................................................4

WHAT IS CVSNT AND CM SERVER ALL ABOUT ..................................................................................................4

CAN I INSTALL THE SOFTWARE AND READ THIS LATER ..............................................................................4

I DON’T LIKE VERSION CONTROL .......................................................................................................................4

I PREFER TO USE SOME OTHER TOOL ................................................................................................................5

RIGHT VERSUS WRONG ......................................................................................................................................5

SOME PLACE TO START ......................................................................................................................................5

CVS ARCHITECTURE ............................................................................................................................................7

CLIENT / SERVER ARCHITECTURE .......................................................................................................................7

WHEN DOES THE SERVER START .......................................................................................................................8

Advantages of a Non-Monolithic model ...............................................................................................................8

Advantages of a Monolithic model ......................................................................................................................9

MULTI SITE, REPOSITORY REPLICATION AND WAN PERFORMANCE ................................................................10

Overview ..............................................................................................................................................................10

WAN Optimised Workspaces and Procedures ...................................................................................................13

Multiple Repositories .........................................................................................................................................15

Large Dual Site Development - Repository Replication Cache .........................................................................15

Read Only Mirrors - Repository Replication Disaster Recovery or Ownership ..................................................17

Multi Site Parallel Development ........................................................................................................................17

Other Multi Site Techniques ................................................................................................................................18

SSH VERSUS CVS NETWORK PROTOCOLS (SSPI, SSERVER, ETC.) .....................................................................19

CVSNT Client connected directly to CVSNT Server ...........................................................................................20

CVSNT Client connected directly to SSH Server ...............................................................................................20

CVSNT Client connected to an external SSH client connected to SSH Server ..................................................21

CVSNT Client connected to the external EXTNT client connected to SSH Server .............................................21

DIRECTORY ACCESS ..........................................................................................................................................22

Directory access (windows) ................................................................................................................................22

Workspaces / Sandboxes on Network Shares or Netowrk Drives (SAN or NAS) ..................................................22

How CVS works with Network Shares and Drives ............................................................................................22

ANTIVIRUS CONTROLS ..................................................................................................................................24

Supported and Unsupported Anti Virus Software .............................................................................................24

An Introduction to Anti Virus Software .............................................................................................................25

Common Problems with CVSNT and Anti Virus Software ..............................................................................26

Configuring Anti Virus Software on the Server ...............................................................................................26

Configuring Anti Virus Software on the Client ..................................................................................................26

Conclusion ...........................................................................................................................................................27

DESIGNING A SOLUTION .................................................................................................................................28

SERVER ARCHITECTURE ...................................................................................................................................28

Security Requirements .........................................................................................................................................28

SECURITY ARCHITECTURE ...............................................................................................................................28

Security of the CVS repository ...........................................................................................................................29

Security of the CVS audit logs ............................................................................................................................29

Why are security protocols/authentication mechanisms important ....................................................................29

What are the drawbacks to using server authentication ...................................................................................29
Native usernames versus CVS PASSWD usernames ................................................................. 29
MUTIPLE REPOSITORIES........................................................................................................... 30

PART II – INSTALLATION ........................................................................................................... 32

SETTING UP CVS SUITE SERVER ............................................................................................. 33

UPGRADING CVS SUITE SERVER ON WINDOWS ..................................................................... 33
 Outstanding work ....................................................................................................................... 33
 Backup the installation directory ............................................................................................... 33
 Backup the repository directory ................................................................................................. 33
 Uninstall Bugzilla, Make and Audit plugins earlier than 2.5.02 ...................................................... 33
 Uninstall CVSNT versions earlier than 2.0.62 .............................................................................. 34
 Install CVSNT .............................................................................................................................. 34

QUICK GUIDE TO INSTALLING A NEW CVS SUITE SERVER ...................................................... 34
 CVSNT 2.0.62, CVSNT 2.0.58, CVSNT 2.0.51, CVSNT 2.0.41 etc......................................................... 34
 Windows Server Synchronisation to a Reliable Time Source ......................................................... 34
 Ensure Client and File Server is Synchronised to a Reliable Time Source ...................................... 35
 Choose a Protocol ....................................................................................................................... 35
 Install CVSNT on the server ...................................................................................................... 35
 Configure the server .................................................................................................................. 35
 Create a repository .................................................................................................................... 36
 Configure the Repository ........................................................................................................... 36
 Disable Virus Scanning on the Repository and CVSNT Temp Directories ................................. 38
 Disable or Remove Unused Protocols ......................................................................................... 38
 Configure Server for Auto Connection ...................................................................................... 38
 Importing your existing source code to the repository .............................................................. 39

QUICK GUIDE TO INSTALLING A NEW CVS SUITE CLIENT .................................................... 39
 Ensure Client Time is Synchronised to CVS Server Time ............................................................ 39
 Windows Client with Import, Checkin / Commit Rights ............................................................... 39
 Windows Client for “Release” or “Checkout Only” ................................................................... 39

SERVER ADMINISTRATION ....................................................................................................... 40

CONFIGURING NETWORK ACCESS ............................................................................................ 40
 Native usernames versus CVS PASSWD usernames ................................................................. 40
 Protocols ....................................................................................................................................... 41
 The ext protocol and EXTNT on windows .................................................................................... 41
 Protocols Ports, Firewalls and Simple File Sharing ......................................................................... 41
 Protocols Authentication Syntax ................................................................................................ 45
 Authentication ............................................................................................................................ 45
 Config ........................................................................................................................................... 48
 Admin ........................................................................................................................................... 48
 SSH .............................................................................................................................................. 48
 SSPI ............................................................................................................................................. 49
 Requiring Encrypted and/or Authenticated Network Traffic ......................................................... 50
 Requiring Compression .............................................................................................................. 52
 Disable or Remove Unused Protocols ........................................................................................ 53
 Configure Server for Auto Configuration .................................................................................... 53
 Requiring Secure SSPI connections .......................................................................................... 54
 Requiring Secure SSH connections ........................................................................................... 55

INTEGRATION DEPENDENCIES AND REQUIREMENTS ............................................................. 55
 INTEGRATION WITH MAKE (BUILD POST COMMIT TRIGGER) .................................................. 56
 AUDIT (TO DATABASE) .................................................................................................................. 56
 INTEGRATION WITH EMAIL .......................................................................................................... 56
 INTEGRATION WITH BUGZILLA, MANTIS OR JIRA DEFECT TRACKING ...................................... 56
 CVSWEBNT (REQUIRES IIS 5 AND ACTIVEPERL 5.6) ................................................................. 56

TROUBLESHOOTING .................................................................................................................. 57
 PERFORMANCE ........................................................................................................................... 57
 CONNECTIVITY .......................................................................................................................... 57
 Pserver and SSPI protocols .......................................................................................................... 57
 Using Telnet to Debug ................................................................................................................ 58
## Table of Contents

- Simultaneously Attempting to Run CVS ................................................................. 58
- Checkouts are atomic ......................................................................................... 58
- Eclipse ................................................................................................................. 58
- Error Messages ................................................................................................... 59
  - `cvs [checkout aborted]: end of file from server (consult above messages if any)` ........................................................................................................... 59
  - `cvs server: Can’t open info library: @D:/VerCON/CS_FW/CVSROOT/trigger.dll` ........................................................................................................... 60
  - `Fatal error: Can’t open privilege tables: Can’t find file: ‘host.MY’ (errno: 2)` ........................................................................................................... 60
  - `E [cvs aborted] couldn't connect to bugzilla database` ................................... 61
- Configuring MySQL for CVSNT ............................................................................. 63
  - Pre-Requisites .................................................................................................. 63
    - Pre-Requisites for Defect Tracking Integration – Bugzilla or Mantis ............. 63
    - Pre-Requisites for Defect Tracking Integration – Jira .................................... 63
    - Pre-Requisites for Audit Integration .............................................................. 63
  - Installing for MySQL ....................................................................................... 63
    - Installation of MySQL ................................................................................... 63
    - Configuration of System PATH ....................................................................... 65
    - Create the database ....................................................................................... 66
  - Installing Bugzilla on Windows ...................................................................... 67
    - Install, Download or Checkout Bugzilla ........................................................ 67
    - Configuration of IIS ....................................................................................... 68
- Part III – Client .................................................................................................... 75
  - Client Connection and Configuration ............................................................... 77
  - Client Time Synchronisation to CVS Server ................................................... 77
  - General Guide to Authentication ................................................................... 77
  - Storing passwords securely (CVSAGENT & Putty PAGEANT) ......................... 77
  - PServer ........................................................................................................... 78
  - SSH ................................................................................................................ 78
  - SSPI ................................................................................................................ 79
- Setting up CVS Suite Studio Client on Windows .............................................. 79
  - Import ............................................................................................................. 80
  - View Server Tags and Branches ...................................................................... 80
  - Checkout and Creating a Workspace ............................................................... 81
  - Synchronise a Workspace with the Server ..................................................... 81
  - Set and Manage ACL’s .................................................................................. 82
  - Browse files, versions and structure of the repository .................................... 82
  - Create modules / directories on the server ...................................................... 84
- Setting up SCCI / Visual Studio .NET Integration on Windows ....................... 84
  - Performance of VS.NET integration ............................................................... 84
  - Adding a project to source control .................................................................. 85
  - Check out for edit ............................................................................................ 86
  - Comparing local file to original (or any other version) .................................... 87
  - Open from Source Control .......................................................................... 88
  - Switching a project from PushOK to CVS Suite (Bind to CVS Suite) .......... 90
  - Switching between several installed SCCI providers ................................... 90
  - VS.NET & SCCI Integration Properties ......................................................... 91
- Setting up Oracle JDeveloper on Windows ....................................................... 92
  - JDeveloper 10g (10.1.3) ............................................................................... 92
  - Configure ...................................................................................................... 92
  - Import ........................................................................................................... 93
  - Adding a CVS Connection .......................................................................... 94
  - Check Out and CVS Navigator ...................................................................... 97
  - Commit/Checkin and Application Navigator ............................................... 97
  - Edit/Watch mode, Making Changes and Message Log ................................ 97
- Setting up Eclipse Client on Windows ............................................................. 97
- Setting up IntelliJ IDEA Client on Windows ................................................... 98
- CVSNT Password Agent .................................................................................... 98
  - Starting the Agent ......................................................................................... 98
  - Clear the passwords ....................................................................................... 98
## Table of Contents

- **Entering a password** ......................................................... 99
- **Setting Up TortoiseCVS Client on Windows** .......................... 99
- **Configuring TortoiseCVS with a Windows CVSNT Server** .......... 99
- **Configuring TortoiseCVS with a Unix CVSNT Server** ............... 99
- **Setting Up Mac CVS X Client on Mac OS X** .......................... 100
  - Introduction ........................................................................ 100
  - Preferences... General (CVSROOT) ...................................... 100
  - Side by Side Diffs preferences ........................................... 102
  - Login .................................................................................. 102
- **Setting Up XCode Client on Mac OS X** ............................... 103
  - Your code must already exist in CVSNT ............................... 103
  - Supported protocols ......................................................... 103
  - Import your source code and make a workspace .................. 103
- **Setting Up WinCVS Client on Windows** ............................... 104
  - Configuration ...................................................................... 104
  - WinCVS CVSROOT and Creating a Workspace ...................... 105
- **Using CVS Suite** ................................................................ 108
  - CVS Suite Studio ............................................................. 108
  - **Browse** ........................................................................ 108
  - Create a workspace .......................................................... 108
  - Import files and directories to the repository ...................... 108
  - Add a repository that is not already listed ......................... 109
  - Browse anonymously or specify a username and password ..... 109
  - Operations on your workspace: Add, Commit etc. ................. 109
- **Visual Studio Plugin (SCCI)** .............................................. 110
  - Modes of operation ........................................................... 110
  - Ensure that CVS Suite is the selected version control provider .. 110
  - Connect an existing project or solution that was checked out from a repository to the Visual Studio SCCI plugin 110
  - Add an existing project or solution to a repository .............. 110
  - Create a new project or solution and add it to the repository ... 112
  - Switch a project or solution from another CVS SCCI client to CVS Suite ................................. 113
- **TortoiseCVS** .................................................................... 113
  - Import (create project/modules in repository and add files to repository) .............................................. 114
  - Using Defect Tracking Integration (Change Sets) with Tortoise .. 116
  - Checkout a single file ......................................................... 117
  - Edit .................................................................................... 120
  - Diff .................................................................................... 120
  - Update ............................................................................... 120
  - Add .................................................................................... 121
  - Commit .............................................................................. 122
- **WinCVS** ........................................................................... 123
  - Edit .................................................................................... 123
  - Diff .................................................................................... 123
  - Update ............................................................................... 124
  - Add .................................................................................... 124
  - Commit .............................................................................. 125
  - Checkout ............................................................................. 125
- **Eclipse and IBM WebSphere Application Developer** ............... 125
- **Mac CVS X Client on Mac OS X** ........................................ 125
  - Introduction ........................................................................ 125
  - Login .................................................................................. 126
  - Import ............................................................................... 126
  - Checkout and Creating a Workspace .................................... 128
  - Annotate ............................................................................ 130
  - Log - View History of a file ............................................... 131
  - Graph - View History of a file graphically ............................ 131
- **XCode Client on Mac OS X** ................................................. 132
  - Using the project from XCode ............................................. 132
FIND OUT ABOUT OTHER PEOPLES CHANGES ................................................................. 133
Notify................................................................................................................................. 133
GOOD COMMENTS ........................................................................................................ 133
RELEASE MANAGER ....................................................................................................... 134
Release Manager Security .............................................................................................. 135
Registering a workspace .................................................................................................. 135
Select (or add) a CVS, CVSNT or FTP Server ................................................................. 136
Select (or add) a FTP Server .......................................................................................... 138
Selecting a Schedule Username ...................................................................................... 141
What happens when errors occur .................................................................................. 141
Allowing Release Manager Client to run on Windows Vista: ......................................... 141
Allowing Release Manager through Windows XP or Windows Vista Firewall: ................ 143
Modifying Schedule Master Process .............................................................................. 145
Modifying Schedule Username ...................................................................................... 145
Disabling all Schedules ................................................................................................. 146

PART IV – APPENDIX ..................................................................................................... 147
WHAT’S NEW IN CVS SUITE 2009 ............................................................................... 148
Part I – Theory
Fundamentals

What is CVSNT and CM Server all about

This software helps computer users keep track of changes to files.

All of the things you create on your computer: Documents, Program Source Code, Web Pages, Pictures, Spreadsheets can be managed using CVS Suite or CM Server.

In addition to tracking the changes, the software also can provide assistance with publishing, reviewing, securing and managing those files and the ability for different computer users to make changes at different stages of the documents life.

CVS Suite was originally designed for tracking changes to files and documents written by computer programmers: computer source code. This is still the primary use of CVS Suite and the focus of this book, though the same procedures can be used for managing any type of file.

CM Server is a more advanced edition of the software and addresses the requirements of larger teams and organisations.

Can I Install the Software and Read this Later

Effectively managing your computer files and the changes to them largely depends on how you work and what your priorities for management are.

If you attempt to use the software without understanding the theory first then you will almost certainly find it is not optimal for your purposes.

Therefore you are encouraged to read the theory before attempting the practical.

I Don’t Like Version Control

In this book we use the term *Effective Configuration Management* over and over again. As an organisation it took us at March Hare Software Ltd a long time to discover that there is a difference between Configuration Management and Effective Configuration Management.

Most people who do not like version control feel that way because they have been exposed to it in an ineffective environment. Spending time doing things that are ineffective will lead to an enormous level of frustration.

If you don’t like version control, please read the theory section in this book and look for a process that you would be happy to use to effectively manage your work.
I prefer to use some other tool

CM Suite is the ideal server software for people who prefer to use other tools as it is client agnostic. Client agnostic CM Server is ideal for a heterogeneous CM environment, allowing each person to choose the tool most effective for them whilst not limiting the choices of other people and retaining the ability to comply with audit and management objectives. CM Suite supports most popular Version Control and SCCM client tools.

The benefits of Software Change and Configuration Management usually comes from the process, so we recommend that customers focus on identifying the process that directly contributes to resolving the business objectives first, then secondly look for what tools can support that process.

This ensures that you implement a system that provides the benefits all stakeholders are expecting and also ensures that the benefits can be measured to determine the overall cost benefit during post implementation review.

Right versus Wrong

There is no universally right way to implement version control and/or configuration management.

The purpose of this book is not to convince you that one methodology is the correct one to the exclusion of all others.

Each methodology has strengths and weaknesses. Each decision you make while implementing your solution has consequences – most notably in the procedures and management structures that interface with this system.

It is very likely a single methodology is correct for you. The purpose of this book is to help you find that and assist you with implementing either CVS Suite or CM Suite to work that way. If you believe some teams may require a different methodology then they should work through this book and make their own choices based on the provided information and questions.

Some Place to Start

Since there is no correct way to implement version control or Software Change and Configuration Management some people reach an implementation paralysis due to the number of choices. To assist in resolving this we have provided some basic samples of how to use the Suite software tools in the client section of this book.

The primary user interfaces of CVS Suite are

Suite Studio....................... to create projects, import files, set access control, checkout files etc

CM Server Web Admin .... to create CM Suite repositories; set server parameters; enable/disable and configure plugins such as audit and protocols such as sserver and sspi
CVSNT Control Panel ...... to create CVS Suite repositories;
set server parameters;
enable/disable and configure
plugins such as audit and protocols
such as sserver and sspi.
CVS Architecture

CVS Suite Server and Client are designed to work together and in conjunction with Configuration Management processes can improve developer productivity. Setting up CVS Suite Server and Clients is NOT as simple as installing the software. This section CVS Architecture and the next sections Setting up CVS Suite Server and Server Administration provide the necessary information to install and configure the CVS Suite system. If you do not follow these guides then severe performance problems can occur.

Client / Server Architecture

When using CVS to version control your documents and other objects you always will use the client program (with or without a GUI) with files in a sandbox. The CVS Server will always operate on the repository files.

The CVS Server and the CVS Client both use the same executable program cvsnt.exe. The server also requires additional components to configure it and start the servers when clients connect. See the section When does the server start for more information. The cvs.exe is a proxy that will silently start cvsnt.exe if required or proxy with ssh on a unix server.

There are various CVS Front Ends otherwise known as GUIs. These all call the CVS executable to perform their functions and to communicate with the server.

The CVS Server may also optionally run a web service (IIS on windows) with CVSWEB or ViewCVS to allow Clients to browse the CVS history using a web browser.

Clients connect to the server via one of several available authentication protocols. March Hare Software recommends that organizations choose their authentication protocols based in part on the security recommendations (see Security Architecture section below).
Communication between the clients and server can be encrypted using ssh, gserver or sserver protocols (and sspi on windows only), and some protocols such as sspi and pserver support compression. Compression is performed before encryption so both may be enabled at the same time.

The point at which the encryption starts depends on the client and server software in use. On a CVS client/server (including 3rd party CVS clients like Eclipse) the encryption starts as part of the initial setup, after the root request. CVSNT client and server supports 'rootless encryption' which means that it encrypts immediately after the initial handshake. This is required for working in a secure environment where the root is to be concealed from any network snooping.

When does the server start

CVSNT server does not have a monolithic architecture. What that means is that all clients do not connect to a single process running on the server.

The CVSNT server consists of three processes:

- CVSNT Deadlock Server (EVS combines this with EVSMANAGER)
- a Server Service, either:
  ⇒ CVSMANAGER High Performance Server Service, or
  ⇒ EVSMANAGER Server Service (EVS only)
- a Server, either:
  ⇒ CVSNT Server, or
  ⇒ EVS Server

The deadlock server runs constantly and serves all locking requests (read locks and write locks), and also publishes server information using Rendevous / Bonjour. The EVSMANAGER service provides similar functions for CM Server.

The CVSMANAGER High Performance Server Service starts and maintains a pool of several CVSNT Server processes. Each waiting process has already performed all initialisation including trigger initialisation and is ready to authenticate and serve a request from a client. If the client is using ssh then the CVSPROXY (cvs.exe or /usr/bin/cvs) will connect to a waiting CVSNT server.

The EVS Server or older 2.5.x CVSNT Servers are started only when a CVS or CVSNT client performs an operation that requires the server to respond. The client connects to the server using port 2401 (or port 22 in the case of SSH) and the operating system (or the EVSMANAGER or CVSNT Server Service in the case or Windows) starts a unique CVSNT Server process for that client.

Web folder or web browser clients and SVN clients do not create additional server processes (the EVSMANAGER service handles all these requests).

Advantages of a Non-Monolithic model

Since each client begins a new process on the server, it is easy to establish information about how many users are currently accessing the server, and if necessary terminate or monitor them.
Advantages of a Monolithic model
The primary advantage of using a monolithic model is when the host operating system has a large overhead when creating a new process. CVSNT high performance server overcomes this problem without the added complexity of a monolithic (multithreaded) server.

If your environment will have many hundreds of developers performing edit commands on single files at the same time then the overhead of process creation could become high, and the CVSNT Server should be upgraded to 2.8.01 or higher and a suitable pool size set in the server control panel.
Multi Site, Repository Replication and WAN Performance

CVSNT is designed to work well when users are distributed worldwide:

The purpose of this section is to explain in theory how to use CVS well in a distributed environment, and what should be avoided.

Overview
Before starting a technical discussion about Multi Site version control with CVSNT there are some introductory messages that we must clearly advise you of:

– If you have multiple sites relying on CVSNT then you should be using the latest version of the software with Silver level support or higher, and definitely not using a 10 year old community edition
– Only about 4% of CVS Suite is non-GPL code and the repository replication solutions we offer are all based on open source technologies – no large monolithic proprietary solutions are required
– Define the problems you are trying to solve with metrics

Support of CVSNT Server and your Technical Account Manager
Ensuring the reliability, resilience and integrity of your repository are important when implementing a SCM versioning server. If you are unhappy with the performance or reliability of an existing CVS or CVSNT server then the first step is to ensure that you have support from the people who wrote it, and speak to your Technical Account Manager regarding the specific problems you are experiencing – March Hare Software write CVSNT and can assist with resolving problems through CVS Professional Support Services.

Software Solutions
There are unscrupulous individuals and even some organisations that attempt to convince CVS and CVSNT users that they are experts on Multi Site Version Control and based on that expertise they recommend purchase of a proprietary software solution.
No author of CVS or CVSNT has ever written or said that additional proprietary software is required to resolve any multi-site, performance or reliability issue with the CVS or CVSNT server.

Some of the proprietary software that is advertised for enabling Multi Site CVS or CVSNT also claims to provide added security benefits, reports and controls. CVSNT contains sophisticated and reliable security controls that are better integrated and more reliable than any proprietary add-on package, including:

- Access Control Lists
- Authentication
- Deny/Allow clients

No proprietary software is required to resolve multi-site, performance security or reliability issues.

Define the Problem

It is our experience that people ask about Multi Site CVS for the following reasons:

- They are not currently using CVSNT and they expect problems due to previous experience with other SCM tools like ClearCase, Visual SourceSafe or Continuus/CM Synergy.
- They are not currently using CVSNT and they have been approached by a vendor of a proprietary Multi-Site product or have read a review of such a product.
- They are currently using CVS 1.x not CVSNT, CVS Suite or CM Suite.
- They are using CVSNT and do not have support and have not received professional advice by the authors on how to configure and use the system, eg: setting up ‘stand by’ servers. The eBook All About CVS (a part of CVS Professional or CVS Suite) describes how to set up a standby server and also a suite of all the necessary software that has been tested to work together as well as e-mail based support).
- They are using CVSNT and have a specific performance problem (eg: a ‘build’ process takes too long or a ‘tag’ process takes too long).

If the last point applies to you then you should document the specific problem you are experiencing and contact your March Hare Software Technical Account Manager.

For all other problems described above you should purchase CVS Suite or CM Suite with Software Maintenance and Professional Support from March Hare Software, and optionally the on-site training and installation services. See the section Learning more about CVSNT above and specifically the headings Professional Support and Contacting Us.
As with any IT project – you should define the expected outcomes before you begin the project so that after the project is completed then you can establish the relative success of it. If you require the tag to complete within 10 minutes then write that down, if you require the build to complete in 1 hour then write that down – and then give that information to your Technical Account Manager or pre-sales technical support. Ensure that all metrics are not arbitrary, but are based on a business requirement that can be explained easily to our team.

*If there is no proprietary software needed why do I need to pay for support?*

By encouraging those who require more assistance to pay for support we can ensure that the software continues to be developed and provide the best levels of service to those users who rely on CVSNT for their productivity at work.

*Why do March Hare require Silver or higher support for Multi Site?*

March Hare Software base our charges on the effort required to resolve problems. It is our experience that if a company has users at multiple sites that their problems are more difficult to resolve. Additional (open source) software may also be provided for customers with multiple sites designed to resolve particular technical problems that only occur in multi-site installations.

Silver level support is also a pre-requisite for on site services – we strongly recommend that one of our team come to your site(s) when installing and configuring Multi Site solutions. Customers with silver support also have telephone access to our team which is very likely to be needed in the first year after installation of a multi-site product.

*Multi Site authentication*

When working with multiple sites it is important that the authentication scheme is carefully designed. If users are being authenticated by a remote authentication server (eg: a domain controller) then CVSNT client operations may be delayed while that authentication takes place for each command. These types of authentication issues are often not noticeable with operations such as logging in to a PC or connecting network drives, however since CVSNT client operations can be performed many times a minute (especially if using SCCI/Visual Studio Integration) the lag for authentication becomes critical. Also see the sections *Unix and Linux Server Synchronisation to a Reliable Time Source* and *Windows Server Synchronisation to a Reliable Time Source* for other reasons why client authentication may be slow in a multi site installation.

A second multi site issue is that users who are not registered on the central authentication system (such as a domain controller) may require access to your CVSNT server. See the section *CVS only users (no system user with that username)* for details of how to overcome this problem.
If performance should be good without multi-site software, why is my performance poor?

The most common reasons why we see poor performance are:

– The existing tools are not being used in conjunction with CM and CVSNT best practice.
– There are existing procedures that are not required and are inefficient. They may have been introduced to resolve a problem that there were insufficient skills to resolve any other way or a problem that has been resolved in later releases of the software.
– The expectations are unrealistic (defined as performance expectations that could not be matched with a local workgroup repository). Eg: if people using the software are trying to highlight the advantages of an alternative version control system where the repository is not stored or managed centrally and stored only on their own computer.
– The performance has not been measured and the performance problem overstated. This is occasionally seen with ‘outsource’ suppliers who are looking for a scapegoat to blame for missed deadlines.
– The performance problems are a direct result of insufficient training of the people who are using the software (the people experiencing the performance problem). This is not common, however can be occasionally seen with ‘outsource’ suppliers who are using insufficiently trained staff and do not want to bear the cost of training.

WAN Optimised Workspaces and Procedures

Most people who need to use CVSNT at multiple sites or geographically distributed locations will use this solution. This solution provides high performance for remote sites as well as insulation from network connection failures.
When implementing WAN Optimised Workspaces it is important that the software and the procedures are correctly configured to reduce the network transmission overheads.

For example: it is typical to ensure that binary objects (eg: compiled objects such as .jar, .obj, .exe files) are not regularly checked out and updated using CVSNT, but instead that a ‘build server’ is located at each site that can create these for the users at the remote site. Implementing a more complex strategy for binary compiled objects is counter-productive since they still need to be regularly synchronised between the servers and results in more administration of conflicts.

Another example is to ensure that developers do not delete workspaces. If a developer was working on Trunk but now needs to work on branch-rel-1 then the trunk workspace should be kept and the developer switches to working on the branch-rel-1 workspace.

At the remote site it is typical to maintain several sandboxes (at least one per branch). These can be set to automatically synchronise to the server whenever the server is available, and most importantly set to synchronise before the work day for the remote site commences. This sandbox can be copied to another persons PC at the same remote location if they need a ‘clean’ copy of the latest sandbox.
Often remote developers are fearful of making changes when there is no connection to the server because they fear that their workspace may be out of date. Configuring the CVSNT server with automatic e-mail notification will advise users at all sites (via a group e-mail address) of what is changing on the server. A more complex repository replication strategy would not provide a more reliable solution since if the link is down the cache will be out of date.

**Multiple Repositories**

Multiple servers (repository replication) may be considered for the following reasons:

- Large Dual Site Development (Performance / Repository Replication Caching)
  
  *Typically there are two sites in different time zones so that work can proceed 24 hours a day on a project. It is common for each site to house more than 500 users.*

- Repository Mirrors for Disaster Recovery or Ownership
  
  *The majority of work is performed at one location (e.g., India) however the work is owned by an organisation elsewhere (e.g., North America)*

- Multi Site parallel Development - Offshoring
  
  *Where distinct work is carried out at different sites, e.g., In India version 2 is developed but England maintains version 1 and tests version 2.*

**Large Dual Site Development - Repository Replication Cache**

CVSNT client server communication was designed for the kind of networks that were available over twenty years ago. The networks typically in use today exceed the performance and bandwidth of those many times over. Therefore it is not necessary to use Repository Replication to obtain good performance from your CVSNT based CM system.

Implementing replication to improve performance generally does not work: if there is a performance problem it is most likely in the underlying infrastructure or CM methodology. If you have less than 50 users at each location then the best solution for you is described in the section *WAN Optimised Workspaces and Procedures.*

In a few rare cases it may be beneficial to have a local (or regional) cache if there are hundreds of users at each site or if policy prevents the introduction of more efficient workflows and procedures.
This technology is available to CVS Suite and CVS Suite x64 customers with Silver (or higher) level support who purchase additional on site consulting for the installation and certification of the replication software. Configuring repository replication requires:

- CVS Suite and Silver (or higher) support for each person using the software at each site
- CM Design and CVS Administration Training Course
- Installation Consulting for each site
- Professional Services (usually 1-3 days per site)

Prices and descriptions for each of these are available on the web store, including a sample package for two sites with up to a total of 100 users:

http://store.march-hare.com/s.nl/sc.2/category.12/f

- 100 x CVS Suite (32 bit edition) with Silver Support [may vary please contact sales]
- 1 x CVS Design and Administration Training Course
- 2 x CVS Installation Consulting for each site
- 3 x days on site CVS Professional Services [may vary please contact sales]
Read Only Mirrors - Repository Replication Disaster Recovery or Ownership

Your business may require copies of the repository to be stored at multiple physical locations in case of a critical failure in infrastructure at the primary location, or to ensure that the ownership of the property is centralised.

Included in CVS Suite is a repository replicator named Unison designed for this purpose.

Replication may be triggered at each commit to the primary server, or at a time based interval. See the section on **Unison Repository Replication**.

If the primary server becomes unavailable, a typical disaster recovery procedure would be to use the DR DNS server to assign the **Read Only Server 2** the same name as the original **Primary Server** therefore allowing users to recommence working with the least disruption.

Multi Site Parallel Development

This is required when some of the development of a project has been outsourced to a remote site. In this example each stream of development is clearly performed at a different physical location – though users at all locations need to see the work in progress or initiate merges.

This functionality is only available in CM Suite Multi Site – please discuss your requirements with your Technical Account Manager.
Other Multi Site Techniques

Active-Active Repository Replication (or Quorum based Multi Site)

This solution is not recommended – do not use the services of any consultant advertising these services. March Hare Software regularly receives support requests from companies that have implemented this software and experience problems that the vendor cannot resolve. Our recommendation is to uninstall this software and purchase professional support from the people who wrote CVS and CVSNT.

A quorum based multi site solution uses a large number of servers (usually a minimum of 5 or 8) which all regularly synchronise with each other. Based on the number of servers that can be reached via the network at any time (the quorum) the ‘authoratative’ server(s) are established and changes sent to servers that are not a part of the quorum are excluded.

This solution is typically offered based on a perception that it improves performance on unreliable networks (eg: India and China), however it does not achieve the desired result. As soon as the nodes in other continents cannot be reached they are excluded from the quorum and any changes there become non-authorative.

Technically this proprietary software is implemented as a proxy that attempts to intercept the protocol between the client and server – since the authors of the proxy are not members of the active open source CVS and CVSNT communities and the CVSNT clients and servers are written by March Hare Software and open source community members around the world it appears that the proxy intervention regularly causes simple operations to fail.

Due to the large number of servers required there are also very high procurement costs and administratative overhead to this solution.

CVS Server Clustering

This solution is not recommended – typically only organisations hosting projects for many geographically distributed companies require clustering, eg: SourceForge or gnu.org. Some enterprise customers with more than 500 users at a single location may consider this option once initial workflow and procedural issues have been resolved on a phased deployment of smaller stand alone servers.

CVSNT supports clustering ‘out of the box’ – the deadlock service started on only one node of the cluster co-ordinates the activities of cvs server processes on all nodes of the cluster. The repository is stored on a suitable cluster filesystem\(^1\) on a SAN.

\(^1\) Eg: SAN Filesystem from DataPlow, or IBM Tivoli SANergy: http://www-306.ibm.com/software/tivoli/products/sanergy/
This cluster software is typically combined with a network load balancing switch such as the Barracuda Load Balancer\(^2\) or Cisco Catalyst 4500/4000 Series Switch\(^3\) or with Cluster load balancing software (built into the operating system, eg: Windows Server 2003 Advanced Edition or Red Hat Linux AS 4).

A cluster provides both redundancy and high performance.

As you can see from those references – clusters are older technology that whilst tried and tested and very reliable, are not much in fashion today. Clusters are still supported by Windows Server 2019 and 2016 and you can find about them via Microsoft\(^4\).

**SSH versus CVS Network Protocols (SSPI, Sserver, etc.)**

Network protocols allow the CVSN 1 nt client and server to connect. There are many to choose from. One protocol or another is typically chosen because of the features it supports.

We encourage customers to choose the protocol which is the most secure whilst providing the easiest use (lowest friction) for the person using CVSN T – eg: SSPI on Windows or SSH on Linux. The relationship between compression and encryption, and the point that encryption starts is explained above in the section *Client / Server Architecture*.

There are two network servers (or classes of network protocols) that can be used to connect the CVSN T client and the CVSN T Server.

*CVSN T 'class 1′ network protocols (SSPI, SServer, etc):*

The first and most common class is the ones that CVSN includes its own network server (network protocol) which will directly start the CVSN T Server. Because this protocol is native to CVSN T and directly starts the server, the protocol can only be used for accessing CVSN T.

*CVSN T 'class 2′ network protocols (SSH):*

The second class is an external protocol, most commonly on Unix/Linux if you already have SSH server (SSHD) running on the machine which will host the repository, you can use it as the connection between CVSN T client and CVSN T server.

---


Because this is a generic server, it can be used to login to the Unix/Linux command line or to start other services between windows and Unix/Linux. If you want to use SSH protocol but want to limit it to only CVSNT then you will need your Unix/Linux administrator to configure the user account in such a way as to prevent any other use. This is a Unix/Linux administration issue and will vary depending on your Unix/Linux operating system and version and your policies and is therefore not covered here.

It is not recommended to use an SSH server if your repository will be stored on a Windows server, use the inbuilt CVSNT SSPI protocol instead. See the section SSPI Protocol for more information.

**CVSNT Client connected directly to CVSNT Server**

The most straightforward connection is between a CVSNT client and the CVSNT Server using the included network protocol which operates over port 2401.

The CVSNT network protocol consists of several elements that can be chosen:

- Authentication Protocol (pserver, SSPI, sserver etc).
- Encryption
- Compression

Some authentication protocols are more secure than others, please read the authentication and SSPI Protocol sections.

**CVSNT Client connected directly to SSH Server**

This is the most common technique for connecting a CVSNT windows client to a CVSNT server on Linux, Unix or Mac OS X.

The CVSNT client includes a copy of the Putty SSH client. This avoids the need to set up, install and maintain a separate SSH client to CVSNT. The Eclipse CVS client also includes an SSH client called EXTSSH.
CVSNT Client connected to an external SSH client connected to SSH Server

This is the most common technique for connecting a CVSNT Unix, Linux or Mac OS X client to a CVSNT server on Linux, Unix or Mac OS X. It is no more secure than using the inbuilt ssh client in CVSNT, however Unix, Linux and Mac OS X administrators may feel this is easier to maintain. We do not recommend using this on Windows (eg: using TortoisePlink) – the included SSH client in CVSNT on Windows is newer and contains more bug fixes than the last version of TortoisePlink.

To use this technique requires:
- Use of the :ext: protocol.
- Setting of CVS_RSH environment variable with the ssh command
- Optional setting of CVS_SERVER environment variable

CVSNT Client connected to the external EXTNT client connected to SSH Server

This is a common technique for connecting an Eclipse Windows client to a CVSNT server on Windows. This allows the Eclipse client to use SSPI via the ext protocol of Eclipse.

To use this technique requires:
- Use of the :ext: protocol in Eclipse, TortoiseCVS and CVSNT client.
- Setting of CVS_RSH environment variable with the CVSNT client
- Setting of TortoiseCVS ext helper
Directory access

Directory access (windows)
The following CVSNT directories require all CVSNT users to access (unless you use RunAsUser or pserver aliases):

<table>
<thead>
<tr>
<th>Example Location</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write access</td>
<td></td>
</tr>
<tr>
<td>d:\cvsfiles\repository</td>
<td>Repository</td>
</tr>
<tr>
<td>d:\cvsfiles\reference</td>
<td>Checkout reference area</td>
</tr>
<tr>
<td>Read and Execute access</td>
<td></td>
</tr>
<tr>
<td>c:\program files\cvs suite</td>
<td>CVSNT Program and</td>
</tr>
<tr>
<td></td>
<td>CVSNT Shared Libraries</td>
</tr>
<tr>
<td>Read access</td>
<td></td>
</tr>
<tr>
<td>[HKLM]</td>
<td>Configuration registry settings</td>
</tr>
</tbody>
</table>

Workspaces / Sandboxes on Network Shares or Network Drives (SAN or NAS)
March Hare Software provide support for commercial support customers to use network shares (either a Windows share, Samba share or NFS share), or a network drive (SAN or NAS) as the destination for a `cvs checkout` command (this is called a workspace or a sandbox).

Under no circumstances should a compressed drive, network share or network attached storage (NAS) be used to store a repository, see the section Creating a Repository below for more information.

If a network share or network drive is used to store a sandbox / workspace then the following limitations apply:

- The workspace should only be used for a single architecture client, ie: do not use the same workspace from both Linux and Windows.
- The client computer and the network share or drive must use synchronised time within 500 milliseconds (half a second). Note: some Samba shares and NAS devices by default have a time inaccuracy of 3 seconds which makes them unsuitable for use as a workspace for CVS.

How CVS works with Network Shares and Drives
CVS has a complex interaction with Network Shares and Drives during the `checkout` and `update` tasks.

Network Shares and Drives during Checkout
During the `checkout` task the CVS client requests a version of a file from the CVS server and writes it to a workspace / sandbox. If the workspace / sandbox is located on a network share or network drive then the resulting interaction can be complex.
The CVS Server informs the client of the date/time that the file was last committed (checked in). The CVS Client will attempt to write the file to the workspace / sandbox filesystem using that date/time (expressed in UTC).

The network share host or the processor running the network drive will interpret the CVS client request to set the file date/time based on its own rules and limitations. Some file systems such as the OS/400 may not permit the date/time to be set and will always set it to the date and time the file is written (the current date/time), other file systems may have a low resolution time (eg: Some Samba and network drives) and will round up or round down this time to a three second interval.

The CVS Client will inspect the date / time of the file once it is written and closed and store that in the CVS client hidden data (CVS/Entries).

**Network Shares and Drives during Update**

During the *update* task the CVS client command compares the date / time of each file with the date and time recorded in the CVS client hidden data (CVS/Entries) and if they are different then the entire file is read and sent to the CVS Server for comparison with the original version.

If the file sent is different from the file stored in the CVS Server then the client is notified and the CVS Server sends back the differences.

The CVS Client handles a special case where the CVS server reports no differences. In this case the CVS Client updates the CVS client hidden data with the date on the file in the sandbox / workspace filesystem.

Therefore if the dates on the files in the sandbox / workspace filesystem become out of sync with the hidden data stored by the CVS Client a single *update* command will cause them to be resynchronised.
AntiVirus Controls

You can – and it is possible to use CVSNT and some virus scanners on both servers and clients. However March Hare Software does not provide support for problems directly arising from using Anti Virus Software in conjunction with CVSNT. We make this particular exception from our support because:

- Anti Virus software alters the behaviour of processes on the system, and so poorly implemented Anti Virus software can make a well designed and implemented application like CVSNT unstable
- It is important to design your virus scanning strategy together with your CVSNT server implementation
- In high performance environments – no amount of support or application software design can compensate for a poorly implemented Anti Virus system or implementation.

March Hare Software has never had a reported case of Anti Virus Software on a Unix, Linux or Mac OS X client or server affect the CVSNT supported applications. Sophos and some other Linux kernel based on access anti-virus scanning software is becoming more common, and for performance reasons we strongly recommend that you disable scanning of the CVSNT Temp and repository directories (though scanning the repository CVSROOT directory is OK).

Supported and Unsupported Anti Virus Software

March Hare Software do not recommend any particular Anti Virus Software. To obtain an environment where CVSNT and Anti Virus Software work well together is outlined in the next sections Configuring Anti Virus Software on Client and Configuring Anti Virus Software on Server.

The Windows Nod32 Anti Virus Software (and in particular the IMON service) has been reported as defective by several customers and March Hare Software do not support or recommend this package in any way.

The Windows BitDefender Anti Malware Software (particularly the ‘fast ring’ software which is recommended by BitDefender for non critical systems) has been reported as having occasional false positives against CVS Suite – these are usually quickly rectified by BitDefender themselves.

Most of the laptops that March Hare Software use for demonstrating servers and clients to prospective customers run the Microsoft Security Essentials, however these environments are not high performance. Our Windows and Unix/Linux servers run Sophos endpoint protection. We have previously used McAfee Anti-Virus software successfully on Windows.
If you are running Anti Virus software on a client PC or Server that you require support for from March Hare Software – always advise your support representative. This will NOT preclude you obtaining support. However if the problem is escalated to an engineer and the engineer decides that the Anti Virus may be significantly affecting the behaviour of our application software then March Hare Software reserve the right to ask you to reproduce the support issue on a system without Anti Virus installed. If after using diagnostic procedures on the affected system(s) and on systems without the Anti Virus software March Hare Software are unable to reproduce the problem without the Anti Virus software then your support representative will provide suggestions and support you in the design of an Anti Virus implementation that will not affect the behaviour of the supported application.

An Introduction to Anti Virus Software

Anti Virus software runs mostly on Windows Client and Server PC’s. Due to a history of unsecure software running on Windows PC’s a range of applications have been developed that exploit this poor software design to allow for additional applications to run without the users explicit consent – these are generally called viruses or spyware.

Anti Virus software typically links to the Windows operating system in such a way that regardless of how a file (or sometimes a communications port) is opened the Anti Virus software is informed (regardless of the users consent).

The majority of Anti Virus software achieves this by linking in to the Windows Kernel as an IFS driver. These drivers do not appear in the process table – but rather appear as an overhead in every application that uses the file system.

Often problems (such as high memory use, disk activity, performance) that are easily attributed to a program by using the Task Manager is in fact caused by an IFS driver or Anti Virus Package that does not appear as a separate task. This is because “tasks” are user-space programs that can be started and stopped by other user-space programs. By ensuring the Anti Virus software runs at the kernel level the Anti Virus author can be guaranteed that a virus cannot close the anti virus application.

Based on our definition of a virus above - for a virus to be a virus it must execute. Therefore viruses are always held in a file format that can be executed, such as Applications (.exe) Macros (.scr, .xls, .doc, .html) etc.

By default most Anti Virus packages scan memory, and every file on the disk as it is opened or read (a file rename includes an open and read), and some also scan the traffic on the networking ports. Much of this activity is counter productive because the files (or traffic) being monitored can never be executed. To improve the performance anti virus software can usually be configured by an administrator to exclude (not scan) some ports, files or directories.
However since the Anti Virus configuration program runs in the user space and the IFS driver in the Kernel then this configuration change does not do exactly what it is advertised to do. The Anti Virus software cannot ask the kernel to not call it for some files – it is called for all files or not called at all. So the Anti Virus software is still called but depending on its configuration, may perform more or less actions when it is called.

**Common Problems with CVSNT and Anti Virus Software**

The most common problems that occur with CVSNT and Anti Virus software are due to performance. When a user “updates” thousands of files in a CVSNT project the CVSNT client can download changes and apply them as well as copy (rename) them over the old version very quickly. Some Anti virus software is so poorly designed that when CVSNT attempts to rename a file that is still being scanned it causes CVSNT to be denied access or to crash. This same problem can be easily reproduced with test programs that read, write and rename files quickly.

**Configuring Anti Virus Software on the Server**

Anti Virus software should NOT scan the CVSNT repository since all these files have a “.v” suffix and are therefore not executable under any operating system including Windows and Red Hat Linux. Scanning these files is a waste of resources and the Anti Virus software can interfere with CVSNT Servers ability to update the repository files in a timely manner. If there is any .exe or .dll or other “executable” files stored in the repository CVSROOT directory it is acceptable to allow a virus scanner to scan those.

March Hare Software recommends that you specify a temporary directory that is used for ONLY the CVSNT server and set that in the CVSNT Server Control Panel (on windows). That temporary directory should not be scanned – or to be pessimistic – only allow “executable” files in that directory to be scanned.

If the vendor provides different versions of the software for mission critical systems (eg: the BitDefender ‘slow ring’) then we recommend you use this rather than software which is designed for non-critical systems (like the BitDefender ‘fast ring’).

**Configuring Anti Virus Software on the Client**

Since most of the files CVSNT is working with are plain ASCII (or Unicode) text files they do not warrant scanning – particularly the files in the hidden “CVS” directories of a workspace or sandbox.

If possible configure the client Anti Virus to ignore the hidden CVS directories.

Broadly speaking Anti Virus problems occur more frequently on servers than on clients. If an occasional Anti Virus related problem occurs on a client it is usually acceptable by the user to retry the operation and for it to be successful a second time.
Conclusion

If you install commercially supported Anti Virus software on your Windows or Unix/Linux/MacOS based client or server PC’s based on industry best practice and the manufacturers recommendation with CVSNT then there is little chance this will cause a functional problem with CVSNT. March Hare Software technical support will assist you in finding a solution in the rare cases when this occurs.

If you require more assistance in determining if CVSNT will operate and can be supported in your environment please do not hesitate to contact pre-sales technical support or technical support. Your Technical Account Manager can also discuss with you options for March Hare Software engineers to visit your site to install or validate the installation of the supported software.
Designing a solution

Server Architecture

CVSNT Server runs equally well on Unix, Linux, Mac OS X and Windows. Deciding which server architecture to use may depend on:
- security requirements
- performance and hardware requirements
- memory requirements
- disk requirements
- locally available administrative skills

Security Requirements

These requirements may be based solely on your authentication needs – but may also incorporate the relative security of the underlying system architecture.

Security requirements are also discussed in Part II.

*chroot jail*

CVSNT on Unix, Linux and Mac OS X may be set up in a *chroot jail* which prevents any hacker gaining access via the cvs server to the remainder of your system. Microsoft windows does not have a similar concept.

*Authentication*

CVSNT Server on Windows, Unix or Mac OS X can authenticate:
- to native operating system usernames/passwords
- to windows active directory usernames/passwords
  (on unix/mac uses the *winbind* daemon: *Samba* plus *winbind* must be installed and operational)
- to CVSNT's own internal database of usernames and passwords

To use CVSNT's own internal database of usernames and passwords you must use the pserver or sservr protocols

See the section *Security Architecture* below, particularly the headings *What are the drawbacks to using server authentication* and *Why are security protocols/authentication mechanisms important* for more information.

Security Architecture

By choosing to implement CVSNT you have chosen the most secure Open Source Version Control solution available today. Most version control solutions provide very little security choices in their protocols. When making decisions about your CVS Security you will need to decide about the security of the network traffic as well as the authentication.

Security requirements are also discussed in Part II under the heading *Configuring Network Access*. 
Security of the CVS repository
Once a person gains direct access to the CVS Repository or access to the CVS “admin” commands then it is possible for them to corrupt or remove audits, including changes made to your source code and documents.

The security of this information is therefore the lowest common denominator when setting up a secure CVS environment. Administrators should be chosen carefully and should be the only ones allowed direct access to the files that comprise the CVS repository.

Security of the CVS audit logs
If you are creating any user defined audit logs using the audit plugin or using notifyinfo or other trigger scripts then the location that you store the audit file and the users who have access to delete or amend it will provide the parameters for its integrity.

If a security log can be deleted or amended by any user then the value of that log is reduced. This is important for text log files as well as logs stored within a database.

Proper backup of these log files also need to be taken into consideration when planning your CVS security.

Why are security protocols/authentication mechanisms important
The authentication mechanism is separate to the other functions of the CVSNT Server. Until you successfully log on you cannot use any of the more sophisticated hacking techniques to damage files and folders.

Authentication can be further removed to the operating system of the server itself. For maximum security March Hare recommend allowing the server operating system to handle security: SSPI (on Windows), and ssh (using a Unix server and high strength RSA keys). See the SSPI authentication section for details about the relative security of Kerberos and NTLM when using SSPI.

What are the drawbacks to using server authentication
Some administrators can be concerned that allowing the server operating system to authenticate users can pose a security threat in itself. The expectation being that if a hacker can bypass the operating system security then they have "free reign".

This can be mitigated by setting expiration times on the security tickets (eg: Microsoft Active Directory has a kerberos ticket lifetime that can be locked down so it doesn't automatically renew), and the CVSNT passwd file can be used to restrict authenticated users to a subset of the possible users on the server.

Native usernames versus CVS PASSWD usernames
CVSNT Server on Windows, Unix or Mac OS X can authenticate:
– to native operating system usernames/passwords
– to Windows Active Directory usernames/passwords
  (on Unix/Mac uses the winbind daemon: Samba plus winbind must be installed and operational)
– to CVSNT’s own internal database of usernames and passwords.

To use CVSNT’s own internal database of usernames and passwords you must use the pservers or sservers protocols.

Multiple Repositories

A single server may define multiple repositories. On Windows every repository should be created on an NTFS disk.

There is no performance benefit or advantage to setting up multiple repositories however it may be required if:
– a lack of space on a single partition of a disk.
– setting up a server to host repositories for several companies
– The administrative files will be very different for each repository

The location of each repository is defined using the /etc/cvs/Pserver configuration file on Unix or using the Windows Control Panel:
Part II – Installation
Setting up CVS Suite Server

This section is divided up into what needs to be done on the server and what needs to be done on the client. They can be the same machine if you prefer, however the procedure is the same.

Note that setting up CVS Suite Server and Clients is NOT as simple as installing the software. At a minimum you should read CVS Architecture, and the this section Setting up CVS Suite Server and the next section Server Administration in this guide of the eBook All About CVS..

Upgrading CVS Suite Server on Windows

If you are upgrading an existing CVS or CVSNT server to CVS Suite 2009 Build 7799 follow these steps before continuing.

Outstanding work

Have all users commit their pending changes on all sandboxes served by the CVS or CVSNT Server you are about to upgrade.

This step is not strictly required - you can skip this step.

Backup the installation directory

Backup the installation directory before installing the new CVSNT. You must install CVSNT into the same directory that the earlier version was installed. The installer will allow you to choose a different directory however some protocols and triggers will no longer operate.

Backup the current installation directory by copying it or creating a ZIP archive. Ensure that any certificates are backed up and if the extnt.ini file has been customised ensure a copy is kept.

Backup the repository directory

Backup the repository by copying all the files to another disk or tape.

If you are upgrading from CVS to CVSNT then you should definitely backup the repository before continuing. CVSNT adds information to the repository that CVS does not understand, so this version will be your last copy of the repository compatible with the older CVS.

If you are upgrading from CVSNT to CVSNT then this step is optional.

Uninstall Bugzilla, Make and Audit plugins earlier than 2.5.02

Versions of CVSNT earlier than 2.5.02 used a different plugin mechaism which is incompatible with the new server. This may have involved installing trigger dll’s in the CVSROOT and modifying the checkoutlist administrative file.

If upgrading this type of installation then checkout the CVSROOT to a local disk and edit the checkoutlist and triggers administrative files to remove the references to the Bugzilla, Make and Audit plugins. Commit the new checkoutlist and triggers administrative files, then cvs remove the DLL’s from the CVSROOT and commit this change as well.
On the server check that the DLL’s have been removed from the CVSROOT. If they have then you are ready to proceed with the upgrade.

Uninstall CVSNT versions earlier than 2.0.62
Versions of CVSNT earlier than 2.0.62 used the Innosetup installer. These are not compatible with recent setups. Stop the CVSNT Server service and CVSNT Deadlock service and uninstall this version of CVSNT before continuing.

Install CVSNT
Install CVSNT into the same directory that it was originally installed. Do not specify a different directory name.

Once the installation is complete, copy back any modified files from the original version, including extnt.ini and certificates.

Quick guide to installing a new CVS Suite server
This section provides a quick guide, or a checklist of the actions needed to install a new CVSNT server. For more information read the detailed guide in Server Administration below.

CVSNT 2.0.62, CVSNT 2.0.58, CVSNT 2.0.51, CVSNT 2.0.41 etc
If you previously installed a CVSNT release prior to 2.0.62 then you will need to uninstall the previous version before installing the new CVSNT package.

If you previously installed a CVSNT release from an “innosetup” then you will need to uninstall it before installing the new CVSNT package.

Windows Server Synchronisation to a Reliable Time Source
For CVS Authentication to operate correctly, and to maximise performance of the CVS System CVS Servers must be synchronised to a reliable time source. This is usually achieved by synchronising the Windows Server to the Primary Domain Controller (PDC).

<table>
<thead>
<tr>
<th>Windows Time Synchronisation</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 2000 Server</td>
<td><code>w32tm -s</code></td>
</tr>
<tr>
<td>Windows XP/2003/2008</td>
<td><code>w32tm /resync /rediscover</code></td>
</tr>
<tr>
<td>2012/2016/2019</td>
<td></td>
</tr>
</tbody>
</table>

For further instructions see:
http://support.microsoft.com/?kbid=216734

Note: if the Windows Server that the CVSNT Server will be installed on is the Primary Domain Controller (PDC) then the Windows Server system should be configured to publish itself as a reliable time source to all CVSNT Clients and Network Attached Storage (NAS) and optionally configured to automatically synchronize the Windows Server system clock to an external reliable time source (eg: `pool.ntp.org`). For further instructions on configuring a Windows 2000 Primary Domain Controller as a reliable time source see the following Microsoft article:

http://support.microsoft.com/?kbid=216734
**Ensure Client and File Server is Synchronised to a Reliable Time Source**
CVS Clients must have their time synchronised to the CVS Server or the same time source as the CVS Server. If the CVS Client will use a network share or disk as a workspace / sandbox then the network share or disk should be synchronised to the same reliable time source as the CVS Server.

**Choose a Protocol**

Choose a communications protocol that suits your client connectivity requirements and your security requirements. For most users with a Windows client this will be SSPI. The SSPI protocol can be used on a Unix or Linux server.

Do not use the :local: protocol except when instructed to do so by a support engineer from March Hare Software Ltd. The :local: protocol is always installed.

**Install CVSNT on the server**

If using a windows server, during the installation you can click “custom” and remove any protocols that you do not want to use. On Unix or Windows you can delete the unused protocols after installation.

**Starting the Servers**

The CVSNT installer for Windows automatically installs and starts the server.

**Configure the server**

It is essential that you configure the following using the CVSNT Server control panel:

- If you have Eclipse clients:
  - Switch on all the compatibility options for non-CVSNT clients
  - Set the Temporary Directory

We recommend that you configure the server to manage file types and prevent clients from manually selecting file types (eg: ASCII/Binary/Unicode and Reserved/Unreserved):

- Set the compatibility option “Ignore client-side force –k options”:
  - for both CVSNT and non-CVSNT clients

We strongly recommend you should configure the following as described in the eBook *All About CVS*:

- Configure the Build Integration
- Configure the Bugzilla Integration
- Configure the Audit Integration
You administer CVSNT using the CVSNT control panel. Open the CVSNT Control Panel to configure the server.

Create a repository
Repositories on a Windows server should always be created on an uncompressed NTFS partition. They should never under any circumstance be created on compressed drive or a network share. Most NAS (Network Attached Storage) devices are merely a network share and unsuitable for a repository. A local disk or RAID array, NAS which is connected to the server using iSCSI or any SAN (Storage Area Network) are suitable for the repository.
There are two ways to create a repository, either using the command line or using the CVSNT Server control panel. To create a repository using the command line use the :local: protocol and the init command:

```command
cvs –d :local:c:\myrep init –a description
```

Or you can add the repository to the defined list using the CVSNT Server control panel:

When you define a new repository you can specify:

- **Location**: The absolute path to a local hard drive where the repository will be stored. For example, "c:\my repositories\repository1". Using reserved words like CVS or CVSROOT in repository names can create compatibility and reliability problems and is not recommended. If there is no repository in the location specified in the add dialog box then it is created.

- **Name**: A unix-style absolute pathname with a leading / and no spaces or special characters (- and _ are OK). For example, "/repository1". Using reserved words like CVS or CVSROOT in repository names can create compatibility and reliability problems and is not recommended. Using drive letters in repository names can create compatibility problems with Unix client and is not recommended.

- **Description**: A text description of the repository. For example, "Repository 1 for the Alpha group". This is the name displayed by repository browsers such as CVS Suite Studio.
Publish Repository: Enable auto-discovery of the repository when using smart browsers like CVS Suite Studio.

Default Repository: If short repository names are accepted then this is the default repository (usually used in conjunction with an anonymous login for read-only access).

Online: The repository is available for read and write connections.

Configure the Repository

The repository configuration is stored in text files in the CVSROOT. You edit these by using a text editor (such as Notepad on Windows) and CVS command line client.

The most common things a new CVSNT administrator will need to do are:

– Grant an administrator access to the repository using cvs rchacl
– Set per repository settings
  ⇒ File extensions that are binary that are not a part of the default list (eg: Microsoft Word files *.doc).
  ⇒ Enable Binary deltas for certain file extensions
  ⇒ Default keyword expansion for certain file extensions
  ⇒ If you wish to use mixed reserved/concurrent editing or exclusively use reserved editing then set the default option for which file types
  ⇒ Add, Modify or Remove Keywords
– Set per server settings
  ⇒ Configure default access control (default – allow, or default – deny)
  ⇒ Configure plugins

Disable Virus Scanning on the Repository and CVSNT Temp Directories

Virus scanning should be disabled on the repository directories and also on the temporary directory used by the CVSNT Server. Failing to disable Virus Scanning may result in poor performance and unrecoverable corruption of your version history.

Be sure to read the section Anti Virus Controls under Server Administration for detailed information on virus scanner support.

Disable or Remove Unused Protocols

If you expect users to connect to your new repository using only sspi, then disable the other protocols (with the exception of the Enum protocol).

For details of how to do this see the Server Administration section below under the heading Disable or Remove Unused Protocols.

Configure Server for Auto Connection

Smart CVSNT client such as the CVS Suite Studio can identify that the server exists on the same local network (subnet) as the client and automatically connect the user without the user needing to know the protocol, repository name etc.

For details of how to correctly set this up see the Server Administration section below under the heading Configure Server for Auto Configuration.
Importing your existing source code to the repository

To create a project (or modules) structure and to add files to the repository you must use the CVSNT client, or a graphical front end such as CVS Suite Studio, Tortoise or WinCVS. See the Client section Tortoise and the heading Import for an example.

To quickly import a folder to the server use CVS Suite Studio, and simply drag the folder from the right side to the appropriate server on the left.

Quick guide to installing a new CVS Suite client

See the Clients section of this manual for detailed information on configuring and using the CVS clients.

Ensure Client Time is Synchronised to CVS Server Time

For CVS Authentication to operate correctly, and to maximise performance of the CVS System CVS Clients must have their time synchronised to the CVS Server or the same time source as the CVS Server. If the CVS Client will use a network share or disk as a workspace / sandbox then the network share or disk should be synchronised to the same reliable time source as the CVS Server.

Windows Client with Import, Checkin / Commit Rights

Install the client using the CVSNST Combined Client Installer (CVS Suite Client).
This will install CVSNST, CVS Suite Studio, Visual Studio .NET Integration
WinCVS, TortoiseCVS, WinMerge, Release Manager etc.

Testing Client and Server communication

Once the CVS Suite Client is installed start the CVS Suite Studio (Start -> Programs -> CVSNST -> CVS Suite Studio) and your server should appear automatically in the panel on the left side. If your server does not automatically appear for browsing then there is some firewall or router between your PC and the Server preventing automatic discovery, you can add your server manually by selecting File -> New.

Importing your existing source code to the repository

To create a project (or modules) structure and to add files to the repository you must use the CVSNST client, or a graphical front end such as CVS Suite Studio, Tortoise or WinCVS (all of these are installed with the CVS Suite Client installer). To quickly import a folder to the server use CVS Suite Studio, and simply drag the folder from the right side to the appropriate server on the left. See the Client section Tortoise and the heading Import for an example using Tortoise instead.

Windows Client for “Release” or “Checkout Only”

If you have any client that should only “checkout” workspaces, such as a web server or a remote deployment, then install the release manager and run it to configure the location of each workspace.

Using the Release Manager it is also possible to mirror one remote workspace to other workspaces in the same location. For example a web server farm, a single web server would receive the latest web site from CVSNST and the other web servers would mirror it.
Server Administration

Note that setting up CVS Suite Server and Clients is NOT as simple as installing the software. The sections CVS Architecture, Setting up CVS Suite Server and this section Server Administration provide the necessary information to install and configure the CVS Suite system. If you do not follow these guides then severe performance problems can occur.

Configuring Network Access

CVSNT Server and CVSNT Clients communicate via the network using one or more protocols. These protocols have an impact on security and the performance of your network, the physical server and CVS itself.

In a typical installation people using CVSNT will be authenticated using an operating system token – this means that they do not login to CVSNT – logging into to PC is enough. It is also possible to configure CVSNT Client to require a separate login or require a certificate to authenticate with the server. See the heading Authenticate bypassing Operating System below for more information.

Native usernames versus CVS PASSWD usernames

CVSNT Server on Windows, Unix or Mac OS X can authenticate:

– to native operating system usernames/passwords
– to windows active directory usernames/passwords
  (on unix/mac uses the winbind daemon: Samba plus winbind must be installed and operational)
– to CVSNT’s own internal database of usernames and passwords.

To use CVSNT’s own internal database of usernames and passwords you must use the pserver or sserver protocols.

The default way to use CVSNT Server is for the server to authenticate using the operating system usernames and passwords. If this is how your team works then there is no need to perform any additional configuration and no need to use the cvs passwd command.
### Protocols

CVS client and server may communicate with each other using various supported protocols. When you install CVSNT you are given a choice of installing these:

<table>
<thead>
<tr>
<th>Protocol Components Installed</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>External command protocol. This includes the EXTNT facility on windows.</td>
<td>:ext:</td>
</tr>
<tr>
<td>Password server protocol.</td>
<td>:pserver:</td>
</tr>
<tr>
<td>Named pipe protocol.</td>
<td>:ntserver:</td>
</tr>
<tr>
<td>Require that clients use authenticated or encrypted packets.</td>
<td>:fork:</td>
</tr>
<tr>
<td>GSSAPI for Active Directory.</td>
<td>:gserver:</td>
</tr>
<tr>
<td>GSSAPI for MIT Kerberos.</td>
<td>:gserver:</td>
</tr>
<tr>
<td>SSPI protocol.</td>
<td>:sspi:</td>
</tr>
<tr>
<td>RSH client protocol.</td>
<td>:server:</td>
</tr>
<tr>
<td>SSH protocol.</td>
<td>:ssh:</td>
</tr>
<tr>
<td>SSL protocol.</td>
<td>:sserver:</td>
</tr>
</tbody>
</table>

March Hare Software recommends that you do not install any protocols that you do not intend to use, or you remove them manually.

#### The ext protocol and EXTNT on windows

The EXTNT facility on windows allows windows clients that do not have native support for protocols such as SSPI (eg: Eclipse) to use a native protocol via the EXT protocol. See the section Setting up Eclipse Client on Windows for more information.

#### Protocols Ports, Firewalls and Simple File Sharing

CVSNT authentication occurs on port 2401 using TCP (not UDP). In order to use the CVSNT server on one computer from a different computer – network connections to the server on port 2401 must be enabled.

If you are hosting the CVSNT server on a computer with a software firewall such as Windows Firewall or Red Hat firewalld or iptables then you must ensure that port 2401 is open to allow other computers to use the CVSNT Server.
Disable Simple File Sharing in Windows 2000 Pro / XP Pro to host a CVSNT Server

CVSNT server cannot be used on Windows 2000 Pro or XP Pro while simple file sharing is enabled. Most new PCs will have this option enabled preventing CVSNT from being used. To disable this open Windows Explorer and select the Tools pulldown menu Folder Options:

Navigate to the View tab and ensure that “Use simple file sharing” is not enabled:

Press OK to save the change.
**Opening Port 2401 for Windows XP Pro or Windows Vista Business to host a CVSNT Server**

Use the network connection control panel to open port 2401 in Windows XP/Vista.

XP: Right click on the network connection and select properties. Choose the “Advanced” tab and press the “Settings button”.

Vista: On Windows Vista use the Network Sharing Center and click Windows Firewall then click Change Settings.

All:
The firewall can be switched on however exceptions must be allowed. On XP *Don’t allow exceptions* and on Vista *Block all incoming connections* should be disabled as shown on the following screens:
Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

**On (recommended)**
- The setting blocks all outside sources from connecting to the computer, with the exception of those selected on the Exceptions tab.
- **Don't allow exceptions**
  - Select this when you connect to public networks in less secure locations, such as airports. You will be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

**Off (not recommended)**
- Avoid using this setting. Turning off Windows Firewall may make the computer more vulnerable to viruses and intruders.

Tell me more about these settings.
To enable other computers to access the server running on XP/Vista then you must open Port 2401:

![Windows Firewall](image)

![Add a Port](image)

**Protocols Authentication Syntax**

Each protocol has its own syntax for authentication. This table describes the key differences.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Usr</th>
<th>Pass</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gserver</td>
<td>No</td>
<td>No</td>
<td>:gserver[;keyword=value...]::host[:port]::/path</td>
</tr>
<tr>
<td>Sserver</td>
<td>Opt</td>
<td>Opt</td>
<td>:sserver[;keyword=value...]::[username[:password]@]host[:port]::/path</td>
</tr>
<tr>
<td>Ext</td>
<td>Opt</td>
<td>No</td>
<td>:ext[program][;keyword=value...][;user@]host::/path</td>
</tr>
<tr>
<td>Pserver</td>
<td>Opt</td>
<td>Opt</td>
<td>:pserver[;keyword=value...]::[username[:password]@]host[:port]::/path</td>
</tr>
<tr>
<td>Ssh</td>
<td>Opt</td>
<td>Opt</td>
<td>:ssh[;keyword=value...]::[username[:password]@]host[:port]::/path</td>
</tr>
<tr>
<td>Sspi</td>
<td>Opt</td>
<td>Opt</td>
<td>:sspi[;keyword=value...]::[username[:password]@]host::/path</td>
</tr>
<tr>
<td>Server</td>
<td>Opt</td>
<td>Opt</td>
<td>:server[;keyword=value...]::[username[:password]@]host::/path</td>
</tr>
</tbody>
</table>

**Authentication**

*Authenticate using Operating System*

A remote CVSNT Server repository is usually configured so that the authentication requirements are the same as for the operating system using a ‘normal’ interactive logon (including two factor authentication such as RSA SecurID). This is the usual and default authentication type.
Authenticate bypassing Operating System

Alternatively the administrator can choose to use the CVSNT security system to provide CVSNT accounts without having to set up accounts valid on the underlying operating system and to bypass operating system security. This is not the default and is uncommon, but is supported.

Authenticate using RSA SecurID

If you have configured your Windows Active Directory to require RSA SecurID two factor authentication during login then the Windows client receives a token from the active directory – this token is then used to authenticate the client actions with CVSNT Server.

Step 1: Authenticate during login using PIN & Key

![Login window with RSA SecurID](image)
Step 2: Provide Windows Active Directory password

Step 3: Use CVS Suite with no further tokens until the login expires
Requiring RSA SecurID token with cvs login command

The `cvs login` command is typically used when you want to avoid using the operating system for authentication, eg: if your PC is not a part of the active directory or you want to perform some work identified as a different person than who you are logged in as.

If you do this then CVSNT will not prompt for the RSA Securid two factor authentication – it will use the active directory authentication only.

Config

The `config` administration file has a setting `SystemAuth` which defines how logins are authenticated and it also effects how administrator users are defined. More information on how to edit this file is contained in the section Administrative Files.

If `SystemAuth = Yes` the user is authenticated by the host operating system and is considered to be an administrator if they are listed in the CVSROOT/admin file or if they are in the 'Administrators' group (NT) or 'cvsadmin' group (Unix).

If `SystemAuth = No` the user is authenticated only by the password in CVSROOT/passwd (see the notes on the sspi protocol below in the section Limiting user access with sspi below). Administrative users are defined by the CVSROOT/admin file.

To deny access by default: set `aclmode = normal` (if you do not set this CVSNT will allow access by default). Before setting this you must ensure that you have already granted the administrator access using `cvs rchacl` command, eg:

```
$ cvs -d :pserver:localhost:/cvsrepos rchacl -u cvsadmin -a all
```

Admin

The `admin` administrative file contains a list of usernames who are designated repository administrators, one per line. This file should *not* be put under CVS control, as that would be a security risk.

Repository administrators are automatically made members of the group 'admin'.

SSH

SSH authentication allows Unix servers to authenticate the logged in user using the standard secure SSH protocol. See the above section SSH versus CVS Network Protocols (SSPI, Server, etc.) for information on how this is different to the CVSNT in-built protocols.

After your repository is created (cvs init) and published (cvs init -a or by editing `/etc/cvsnt/PServer`) test that you can connect to your server using ssh from the server:

```
ssh user@localhost
```

From the windows client connect to the server using:

```
cvs -d :ssh:user@host:/cvsrepo ls
```
For a Unix client you will need to set up an :ext: connection. We strongly advise against using :ext: connections with TortoisePlink on Windows – the inbuilt ssh client is newer and contains more security fixes than any version of TortoisePlink.

**SSPI**

SSPI authentication allows Windows to authenticate the logged in user. The authentication will be at least as secure as the authentication used to log you in the Windows workstation and connect to the network.

SSPI works over TCP/IP so it can more easily traverse firewalls. The SSPI protocol does not need a login, the login you did when you started your workstation is used with this protocol.

**NTLM vs Kerberos**

Depending on the Domain Controllers in your Windows network the authentication will be handled by NTLMv1, NTLMv2 or Kerberos. Windows domains based on Active Directory will generally use Kerberos (which is the most secure) unless specifically configured not to.

To force CVSNT to connect using a specific authentication protocol (kerberos or ntlm) when connected using SSPI use the force keyword. E.g:

```plaintext
:sspi;force=kerberos:hostname:/repository
```

**CVS only users (no system user with that username)**

You may have external users who need to access the CVSNT server who do not have a system login. You can define those users in the CVSROOT/passwd file (or using the `cvs passwd -a 'adduser'` command) and specifying a generic user (which is a valid system user) for the process to with the credentials of.

Example the CVSROOT/passwd file:

```plaintext
melissa:tGX1fS8sun6rY:pubcvs
qproj:XR4EZcEs0szik:pubcvs
```

The user `melissa` will be authenticated by CVSNT server only, and any processes initiated by that user will run on the server with the credentials of pubcvs. The CVSNT Access Control (ACL lists etc) however will all be restricted based on the username `melissa` not the user `pubcvs`.

**Limiting user access with sspi**

When used ‘normally’: SSPI will accept connections from all system users that authenticate against the system (local or domain). Often this is not really what is wanted; instead we want to use the same mechanism as is used with the PSERVER protocol. Here the passwd file limits the logins accepted by CVSNT to those mentioned in the file.

If you set the parameter `SystemAuth = No` in the config administration file then only users listed in passwd will be authorized to connect to the CVS Suite Server. You can add a user to the passwd file with the following command run as a CVS Suite Administrator:

```plaintext
cvs passwd -a -d DOMAIN newuser
```
If the newuser is a member of a different domain than the default domain for the CVS Suite Server, then use this syntax:

\texttt{cvs passwd -a -d \textit{DOMAIN} \textit{DOMAIN\newuser}}

For example, if your CVS Suite Server is located on a server in the ALPHA domain but your user account is in the BETA domain, you would use this command to add the user to the passwd database:

\texttt{cvs passwd -a -D BETA BETA\username}

Note: the username and the domain name are case sensitive. On Windows, Domain names are typically all UPPERCASE, but usernames on both Windows and Unix/Linux are however the administrator entered them. You can see the exact case information that CVS Suite is receiving if you type the following command:

\texttt{cvs -d :sspi:host:/repo login}

The username will be shown in the line above the password prompt, press CTRL-C to cancel the login request.

You can specify either an authenticating domain or a password in the passwd file (press enter twice to tell CVSN(T that no password is used). Failure to include one or the other will allow anyone to connect as that user without a password and is NOT recommended.

Note: Be sure to add an administration user to the passwd file before setting SystemAuth = No, or otherwise you will be locked out.

If you are not going to be using other protocols like pserver or sserver, we recommend that you disable them using the CVS Suite Server control panel.

**Requiring Encrypted and/or Authenticated Network Traffic**

CVS client and server may communicate with each other using various supported protocols, some of these are encrypted and/or authenticated, and with some of the protocols there is a choice.

*What is Authenticated Network Traffic*

Authenticated network traffic contains additional information to guarantee that each individual packet was sent by the specified client and has not been impersonated in some way.

*What is Encrypted Network Traffic*

Encrypted network traffic has been scrambled so that its contents are meaningless without the “keys” known only to the server. Encrypted network traffic is also authenticated (as defined above). The level and type of encryption used is dependent on the protocol. CVSN(T does not use one type of encryption for all protocols. Many of the protocols will have options for stronger or weaker encryption, eg: SSPI will allow NTLM (weaker) or Kerberos (Stronger).

NOTE: It is not possible to use an encrypted traffic with an insecure protocol such as pserver does and this would not result in secure transmission. Defining a secure environment is not as simple as ticking one checkbox.
How do you specify what level of encryption and authentication to use?

The authentication and encryption choices at the server are:

<table>
<thead>
<tr>
<th>Description</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encryption in the Server Settings Tab</td>
<td></td>
</tr>
<tr>
<td>Allow clients to connect without encryption.</td>
<td>Optional (Default)</td>
</tr>
<tr>
<td>Request that clients use authenticated packets if available.</td>
<td>Request Authentication</td>
</tr>
<tr>
<td>Request that clients use encrypted packets if available.</td>
<td>Request Encryption</td>
</tr>
<tr>
<td>Require that clients use authenticated or encrypted packets.</td>
<td>Require Authentication</td>
</tr>
<tr>
<td>Require that clients use encrypted packets.</td>
<td>Require Encryption</td>
</tr>
</tbody>
</table>

Windows

The CVSNT Control Panel is used to set CVSNT Server to require a particular level of encryption and authentication.
What protocols can be encrypted?
CVSNT does not use one encryption library for all protocols, instead each protocol defines its own encryption. SSPI will use NTLM or Kerberos encryption (the default is Negotiate), GSERVER will use Kerberos encryption.

If the server option 'require encryption' is enabled then SSPI is NOT allowed as a connection protocol. This is a deliberate decision since the 'default' negotiated protocol on standalone windows systems uses NTLM unless the user specifically alters the CVSNT SSPI protocol plugin to only use Kerberos, or joins the system to a Domain/Active Directory.

Requiring Compression
CVS client and server may optionally compress the network communication before it is encrypted. This may improve CVS performance on a congested network or a network where bandwidth is limited (such as a WAN or Public Internet). You may use the CVSNT Control Panel to set CVSNT Server to require a particular level of encryption. The encryption choices are:

<table>
<thead>
<tr>
<th>CVSNT Control Panel Description</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compression in the Advanced Tab</td>
<td></td>
</tr>
<tr>
<td>Allow clients to use compressed packets.</td>
<td>Optional (Default)</td>
</tr>
<tr>
<td>Request that clients use compression when available.</td>
<td>Request Compression</td>
</tr>
<tr>
<td>Require that clients use compression.</td>
<td>Require Compression</td>
</tr>
</tbody>
</table>
Disable or Remove Unused Protocols

If you expect users to connect to your new repository using only sspi, then disable the other protocols (with the exception of the Enum protocol).

Disable protocols on Windows

On Windows use the CVSNT Server control panel to configure the protocol plugins and disable the ones that are not required, eg:

![Protocol Configuration](image)

Configure Server for Auto Configuration

Smart CVSNT client such as the CVS Suite Studio can identify that the server exists on the same local network (subnet) as the client and automatically connect the user without the user needing to know the protocol, repository name etc.

This will work for 99% of windows CVSNT server “out of the box”, however unix, linux and Mac OS X servers will require additional steps. These steps are also good for Windows CVSNT server administrators to consider.
Steps to allow server to allow advanced clients to automatically connect:

The following should be configured to allow advanced clients such as CVS Suite Studio to connect automatically to the new CVSNT Server:

- Protocols that have not been configured have been disabled, eg: sserver. See above section Disable or Remove Unused Protocols for more information.
- Set an anonymous username and protocol that is most often required for Unix servers to allow non-unix users to browse the module list. The username specified can have the access restricted using an ACL. (See /etc/cvsnt/PServer)
- Set the repository to Publish and the choose one repository as the Default. (See /etc/cvsnt/PServer). Windows users can configure this in the CVSNT Server control panel by editing the repository in the Repository Configuration tab.
- Enable Zeroconf publication in the Advanced tab of the CVSNT Server control panel or in the /etc/cvsnt/PServer configuration file.

Requiring Secure SSPI connections

Requiring secure SSPI connections requires these settings on the server:

- Encryption: Optional or Request Clients use Encryption
- Protocol plugins: disable (or remove) all protocols except SSPI and ENUM
- Protocol plugin SSPI: enable protocol, disable NTLM & Negotiate, enable Kerberos

The client can use a connection string CVSROOT like this to force the encryption:

:sspi;force=Kerberos:host:/repo

Using Kerberos requires that the CVSNT server is joined to a domain, and when you authenticate with a domain user (not a local user) then the domain policy will enforce Kerberos authentication and encryption\(^5\). If the server is not on a domain, the client connection will fail when force=Kerberos.

It is not possible to have a securely encrypted SSPI connection to CVSNT Server when the server computer is not joined to a domain because then NTLM authentication is used – this is a limitation of windows. Also Linux clients are unable to use Kerberos encryption with SSPI connections – this is a limitation of Linux. CVSNT Linux clients can use GSERVER to Windows CVSNT Servers. GSERVER is a secure Kerberos protocol.

Ensuring SSPI connections are secure (single sign on):

The weakest point of any encrypted authentication is the password. The encryption is entirely dependent on the protocol, which for SSPI is dependant on the authentication – evidenced by the fact that the same CVSNT protocol ‘SSPI’ is ‘secure’ if it uses Kerberos authentication and encryption but insecure if it uses NTLM. So if the password is stored poorly in the client, the value of the encryption is negligible. If the user enters their Windows domain password into CVSNT, WinCVS, TortoiseCVS or CVS Suite Studio, then it will be stored with trivial encryption (or plain text) in the Windows Registry.

\(^5\) See Microsoft NTLM documentation:
https://docs.microsoft.com/en-us/windows/win32/secauthn/microsoft-ntlm
Therefore it is important that the users use ‘single sign on’ and not enter passwords manually. Single Sign on uses the authenticated token that Windows holds from their desktop login is passed to the server for authentication. To enable ‘single sign on’ do NOT specify a username or password with SSPI and do not use ‘cvs login’.

Here is an example of a SSPI secure client connection:

```cvs :sspi;force=Kerberos:host:/repo checkout mymodule```

Here is an example of an insecure SSPI secure client connection:

```cvs :sspi:user@host:/repo login
password: *****
cvs :sspi:user@host:/repo checkout mymodule```

### Requiring Secure SSH connections

Requiring secure SSH connections requires these settings on the server:

- Encryption: Require Clients use Encryption
- Enable your SSH server on the CVSNT server
- Protocol plugins: disable (or remove) all protocols except ENUM

### SSH Authentication:

There are three ways customers typically authenticate SSH connections via CVSNT:

- password
- keys
- GSSAPI

### Ensuring SSH connections are secure (private and public keys):

The weakest point of any encrypted authentication is the password, or frequently in the case of SSH logins: the private keys. If the password or keys are stored poorly in the client, the value of the encryption is negligible. If the user enters their SSH password into CVSNT, WinCVS, TortoiseCVS or CVS Suite Studio, then it will be stored with trivial encryption (or plain text) in the Windows Registry. If the private key is stored in a file and passed to the SSH client then it is also insecure.

Customers typically use either of these methods to secure their SSH authentication:

- Storing the keys in the Active Directory
- Using a secure key agent, eg: putty pageant and loading the keys from a secure place, eg: an encrypted key file, or secure key storage device

Here is an example of an insecure SSH secure client connection:

```cvs :ssh;privatekey=d:\me.ppk:user@host:/repo checkout mymodule```

### Integration dependencies and requirements

Whilst the CVSNT Server integrations do not have any direct dependencies, you may need to install dependent software for the integrated system, eg: the Bugzilla integration requires Bugzilla to be installed – and Bugzilla requires Perl and MySQL. Detailed instructions are available in the section *Configuring MySQL for CVSNT*. 
Windows XP/Vista and Windows 7 Professional can generally be used as a server provided that simple networking is switched off and the number of users will not exceed the pre-defined limits of that edition of Windows. March Hare Software strongly recommends running Windows Server 2003 or 2008 rather than Windows XP/Vista or Windows 7.

**Integration with Make (Build Post Commit Trigger)**

When a user commits a file to the CVSNT server it is possible to automatically begin a build process. The process begins after the command has completed on the client.

For more information please refer to the eBook *All About CVS*.

**Audit (to database)**

Using the Audit integration it is possible to generate detailed audits of CVSNT repository activity including activity based reporting for a time period such as:

- most frequently changed file
- user with most changes
- most modified file by number of lines
- user with largest changes by number of lines

For more information please refer to the eBook *All About CVS*.

**Integration with Email**

When a user commits, tags or begins work on a file (with cvs edit) it is possible to automatically send an e-mail to people who are “watching” that file.

More complex Email integration is possible using CVSMailer a separate (free) product.

For more information please refer to the eBook *All About CVS*.

**Integration with Bugzilla, Mantis or Jira Defect Tracking**

CVS Suite **2009** Build 7799 is the fourth release of CVS Suite which March Hare supports linking CVS with Defect Tracking systems like Bugzilla, Mantis and JIRA.

For more information including information about upgrading from earlier releases please refer to the eBook *All About CVS*.

**CVSWEBNT (requires IIS 5 and ActivePerl 5.6)**

CVSWEBNT provides a mechanism for users to browse the repository using a web browser when the repository is. For more information please refer to the eBook *All About CVS*. 
Troubleshooting

Some basic troubleshooting information is also listed in the appendix Frequently Asked Questions.

Performance

Note that setting up CVS Suite Server and Clients is NOT as simple as installing the software. The sections CVS Architecture, Setting up CVS Suite Server and Server Administration provide the necessary information to install and configure the CVS Suite system. If you do not follow these guides then severe performance problems can occur.

Performance problems are most often caused by connectivity issues. CVS Suite Client is very robust and will often appear to work but be very slow if the connectivity and authentication are not correctly configured.

For the reasons listed above, if you experience performance problems the best action is to contact March Hare Software Support and an engineer can work through traces and your settings to determine where the performance problem is.

The following sections give a very rough guide as to the main causes of performance issues:

- Anti Virus software
- RAID drivers, disks and Network devices
- Reverse DNS
- Authentication delays (particularly with Active Directory, Domain and LDAP authentication)
- Network Latency
- Memory and CPU constraints

If you have followed the installation procedure as described in this manual then performance problems usually do not occur, therefore the diagnostic process for performance problems can be complex – so please contact March Hare Software technical support and be prepared to answer many questions. Diagnosing the cause of a performance problem may take several days, however they are always solvable.

Connectivity

If you are using pserver or SSPI protocols the following section may help you debug what is going wrong.

Pserver and SSPI protocols

Errors along the lines of "connection refused" typically indicate that the cvsservice is not listening for connections on port 2401 whereas errors like connection reset by peer or recv() from server: EOF typically indicate that the cvsservice is listening for connections but is unable to start cvsnt.exe.

The error unrecognized auth response is caused by a bad command line or invalid option used to start the CVS Server on Unix or Linux systems (eg: forgetting to put the authserver command at the end of the line). Another less common problem is an invisible control character that your editor "helpfully" added without you noticing.
Using Telnet to Debug
One good debugging tool is to `telnet servername 2401`. After connecting, send any text (for example "foo" followed by return). If CVS Server is working correctly, it will respond with

```
cvs [authserver aborted]: bad auth protocol start: foo
```

If it fails to work at all, then make sure the CVS Server `cvsservice` is working right.

Simultaneously attempting to run CVS
If several developers try to run CVSNT at the same time, one may get the following message:

```
```

CVS will try again every second, and either continue with the operation or print the message again, if it still needs to wait.

Checkouts are atomic
If the user Fred commits some changes on one workstation, and simultaneously another user Mary performs an update at another workstation, Mary will either get all of Fred’s changes, or none of them.

By default atomic checkouts occur across an entire checkout, (provided the deadlock server is running), but commits are only atomic at the file level. For example, given the files

```
a/one.c
a/two.c
b/three.c
b/four.c
```

If someone runs:

```
cvs ci a/two.c b/three.c
```

And someone else runs CVS `update` at the same time, the person running `update` will either get all of the changes, or none of them. If however, the person running `cvs commit` suffers a power failure in the middle of the commit, it is possible that only one of the files will be committed / checked in.

Eclipse
For more information please refer to whitepaper-eclipse-cvs.
Error messages

cvs [checkout aborted]: end of file from server (consult above messages if any)
Check for messages in the Event Viewer:

If this error occurs immediately after the installation of the integration DLL then it is most likely caused by a failure to set the System PATH (note: this requires a reboot of the server).

To check, open the Control Panels, then the panel System. Go to the Advanced tab and press the button labelled “Environment Variables”. The PATH should include the complete directory name of MySQL (including BIN), and the complete directory name of the CVSROOT for the repository:
In our example the PATH is:
D:\VerCON\CS_FW\CVSROOT;C:\Perl\bin;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;c:\Program Files\cvsn;C:\Program Files\MySQL\MySQL Server 4.1\bin

If this does not resolve the problem then please send the indicated crash dump file to support@march-hare.com

Try running the same command on the server using the CVSROOT :
local:d:\VerCON\CS_FW\

Report the results to support@march-hare.com .

cvs server: Can't open info library:  
@D:/VerCON/CS_FW/CVSROOT/trigger.dll

Try running the same command on the server using the CVSROOT :
local:d:\VerCON\CS_FW\

When ran from a :local: repository the following dialog may be shown:

This error is most likely caused by a failure to set the System PATH (note: this requires a reboot of the server). See the previous troubleshooting point (above) for instructions on setting the SYSTEM path.

Fatal error: Can't open privilege tables: Can't find file: 'host.MYI' (errno: 2)

Check for messages in the Event Viewer:

The mysql directory does not contain a complete database. It was probably copies/moved from c:\Program Files\MySQL\MySQL Server 4.1\data\mysql while the mysql service was running.

Restore the original or uninstall MySQL and reinstall MySQL and repeat the move procedure without the service running.
Troubleshooting

E [cvs aborted] couldn't connect to bugzilla database
The username and password for the bugs (bugzilla) database must be correct (the
default is bugs/bugs).
Configuring MySQL for CVSNT

CVSNT Audit and Bugzilla integration both use MySQL databases. Auditing can also use other databases (e.g., Oracle) – to install Oracle or another database please follow the vendors guidelines.

Pre-Requisites

To install an integrated server which includes CVSNT and Defect Tracking and/or Audit integration you require some pre-requisites.

Pre-Requisites for Defect Tracking Integration – Bugzilla or Mantis

To install an integrated server which includes CVSNT and Defect Tracking integration you require:

- ActiveState Active Perl, e.g., ActiveState Active Perl 5.8.4.810 (or compatible)
- Bugzilla 2.18, 2.22, 3.0, 3.2 or Mantis 1.1.1 (or compatible)
- MySQL 4.1.7 (or compatible)

The following must already be installed:

- Windows 2003/2008 Server or Advanced Server
- Internet Information Services (IIS), a part of the Application Server component of Windows including the SMTP service (unless another SMTP server is available in the same subnet).

Pre-Requisites for Defect Tracking Integration – Jira

CVSNT Defect Tracking integration has been tested with Jira 3.12.2 and MySQL 4.1.7. Please see the vendors requirements for further information.

Pre-Requisites for Audit Integration

CVSNT Audit (database) integration has been tested with MySQL 3.23, MySQL 4.1.7 and MySQL 5. Audit integration is also available to other databases. Please see the vendors requirements for further information.

Installing for MySQL

Installation of MySQL

1. Install a supported version of MySQL using the “complete” option, for the version of Bugzilla you are using. Eg: installing MySQL 4.1.7 using “Complete” to c:\Program Files\MySQL\MySQL Server 4.1.
Do NOT run the configuration wizard.

If required, install MySQL Control Center (eg: mysqlcc version 0.94).

2. Delete the contents of c:\Program Files\MySQL\MySQL Server 4.1\data including all sub directories EXCEPT the directory mysql.

3. Run the MySQL Server Instance Config Wizard.

4. Optionally configure mysql to store the mysql data and log files in a different location by stopping the MySQL service and copying the contents of the c:\Program Files\MySQL\MySQL Server 4.1\data directory to the new location and alter the c:\Program Files\MySQL\MySQL Server 4.1\my.ini file datadir property (eg: datadir=D:/VerCON/MySQLData/).
5. You must add the MySQL “bin” directory, and the CVSROOT of the repository to the SYSTEM path.

Go to the Control Panels and open the panel System. Navigate to the Advanced tab and press on button labelled “Environment Variables”. The PATH should include the complete directory name of MySQL (including BIN), and the complete directory name of the CVSROOT:

In our example the PATH is:

D:\VerCON\CS_FW\CVSROOT;C:\Perl\bin;%SystemRoot%;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\Program Files\cvsnt;C:\Program Files\MySQL\MySQL Server 4.1\bin
You must reboot the server for this change to take effect. It is not necessary to do this immediately, but you MUST do it before configuring the integration plugin.

Create the database

6. Create the database, eg: bugs or audit. Ensure that mysql has been added to the PATH.

Command Prompt

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 4.1.7-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create database bugs;
Query OK, 1 row affected (0.00 sec)

mysql> grant all privileges on bugs.* to 'bugs'@'localhost'
identified by ''; 
Query OK, 0 rows affected (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye

C:\>
7. If you want to set a password for the bugs account then you will also need to use the “set password” command to ensure the password is encoded in the old format:

```
Command Prompt
Microsoft® Windows NT(TM)  
(C) Copyright 1985-1996 Microsoft Corp.
C:\> mysql --user=root --password=xxx mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2 to server version: 4.1.7-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> set password for bugs@'%' = old_password('password');
```

Note: if you are using MySQL 3.23 server the set password command will be:

```
set password for bugs@'%' = password('password');
```

If you do not use this setting then you will get the following error message:

```
Client does not support authentication protocol requested by server; consider upgrading MySQL client
```

**Installing Bugzilla on Windows**

Ensure you have first completed the above section for installing MySQL.

**Install, Download or Checkout Bugzilla**

8. You can install Bugzilla three ways:

   - Using the Bugzilla.zip supplied by March Hare consulting. If you have purchased this option with consulting then it will be available in the customer area of the march-hare.com web site
   - Using a Bugzilla tarball available from the Bugzilla.org download site
   - Using CVS with the following command:
**Command Prompt**

Microsoft® Windows NT(TM)  
(C) Copyright 1985-1996 Microsoft Corp.

D:\>cvs -d :pserver:anonymous:anonymous@cvs-mirror.mozilla.org:/cvsroot checkout -d Bugzilla -rBUGZILLA-2_18 Bugzilla  
Checking perl modules ...
.
.
U Bugzilla/template/en/default/search/search-create-series.html.tmpl  
U Bugzilla/template/en/default/search/search-help.html.tmpl  
U Bugzilla/template/en/default/search/report-graph.html.tmpl  
U Bugzilla/template/en/default/search/report-select.html.tmpl  
U Bugzilla/template/en/default/search/report-table.html.tmpl  
U Bugzilla/template/en/default/search/search-specific.html.tmpl  
U Bugzilla/template/en/default/search/tabs.html.tmpl  
cvs checkout: Updating Bugzilla/template/en/default/whine  
cvs checkout: Updating Bugzilla/template/en/extension  
cvs checkout: Updating Bugzilla/template/en/extension/hook  

D:\bugzilla>

**Configuration of IIS**

9. Create a new virtual directory named Bugzilla for the path d:\bugzilla in the default web site using IIS. Add the document index.html to the default content pages. The permissions should be set as follows:

<table>
<thead>
<tr>
<th>IIS Windows Version</th>
<th>Permission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bugzilla</td>
<td></td>
</tr>
<tr>
<td>IIS 4 – Windows NT 4</td>
<td>Read, Run Scripts and Execute.</td>
</tr>
<tr>
<td>IIS 6 – Windows 2003</td>
<td>Virtual Directory: Read</td>
</tr>
<tr>
<td>IIS 7 – Windows 2008</td>
<td>Execute Permissions: Scripts and Executables</td>
</tr>
<tr>
<td>IIS 7.5 – Windows 2008 R2</td>
<td></td>
</tr>
</tbody>
</table>
10. If using Windows 2003/2008 then set the Perl CGI and Perl ISAPI extensions to Allow.

11. Create or modify the application extension mapping for .CGI for the virtual directory created in step 8 above. Open the properties of the virtual directory, select the Configuration… button.
A window titled Application Configuration opens.

Select the CGI extension mapping and edit it so that it appears as shown below (if the CGI extension mapping does not already exist use the Add… button to create one). Note: on Windows 2003 it may be necessary to deselect the “Verify that file exists” option:

12. Create c:\temp and ensure SYSTEM has write access.
13. Add c:\perl\bin to the PATH.
14. Use the Perl Package Manager to install the following perl modules:

- AppConfig
- TimeDate
- DBI
- http://theoryx5.uwinnipeg.ca/ppms/DBD-mysql.ppd
- Template-Toolkit
- GD
- Chart
- GDGraph
- PatchReader
- MailTools
- http://theoryx5.uwinnipeg.ca/ppms/Template-GD.ppd

```
Command Prompt
Microsoft® Windows NT(TM) 
(C) Copyright 1985-1996 Microsoft Corp.
C:\> ppm
xxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxx
xxxxxxxxxxxx.
ppm> rep add Bugzilla http://landfill.bugzilla.org/ppm
  [1] ActiveSate PPM2 Repository
  [3] Bugzilla
ppm> install AppConfig
```
Alternatively download the ppd files from the web to f:\perl_modules and use the PPM command “rep add Bugzilla f:\perl_modules”.

15. Run “perl checksetup.pl”. This will create the initial configuration file “localconfig”.

16. Edit d:\Bugzilla\localconfig to set:

- MySQL database name (db_host)
- MySQL username (db_user)
- MySQL password (db_pass)
17. Re-run “perl checksetup.pl”.

```
Command Prompt
Microsoft® Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.
D:\bugzilla> perl checksetup.pl
Checking perl modules ...  
Checking for AppConfig (v1.52) ok: found v1.55
Checking for CGI (v2.93) ok: found v3.04
Checking for Data::Dumper (any) ok: found v2.121
Checking for Date::Format (v2.21) ok: found v2.22
Checking for DBI (v1.36) ok: found v1.48
Checking for DBD::mysql (v2.1010) ok: found v2.9005_3
Checking for File::Spec (v0.82) ok: found v0.87
Checking for File::Temp (any) ok: found v0.14
Checking for Template (v2.08) ok: found v2.13
Checking for Text::Wrap (v2001.0131) ok: found v2001.09291
The following Perl modules are optional:
Checking for GD (v1.20) ok: found v2.16
Checking for Chart::Base (v1.0) ok: found v2.3
Checking for XML::Parser (any) ok: found v2.34
Checking for GD::Graph (any) ok: found v1.43
Checking for GD::Text::Align (any) ok: found v1.18
Checking for PatchReader (v0.9.4) ok: found v0.9.5

All the required modules are available at:
    http://landfill.bugzilla.org/ppm/
You can add the repository with the following command:
    ppm rep add bugzilla http://landfill.bugzilla.org/ppm/
Checking user setup ...
Removing existing compiled templates ...
Precompiling templates ...
Checking for MySQL Server (v3.23.41) ok: found v4.1.7-nt
Creating table user_group_map ...
Creating table series_data ...
Creating table namedqueries ...
Creating initial dummy product 'TestProduct' ...
Populating duplicates table...
Creating duplicates directory...
Migrating old chart data into database ...
Adding group tweakparams ...
Adding group canconfirm ...
```

Looks like we don't have an administrator set up yet. Either this is your first time using Bugzilla, or your administrator's privileges might have accidentally been deleted.

Enter the e-mail address of the administrator: admin@mycompany.org

You entered 'support@march-hare.com'. Is this correct? [Y/n] Y
Enter the real name of the administrator: Joe Bloggs
Enter a password for the administrator account: ****

Please retype the password to verify: ****

'admin@mycompany.org' is now set up as an administrator account.
18. Set the security on the d:\Bugzilla\data directory to permit the Internet Guest Account write access.

19. Using IIS Admin, set the default page on the Bugzilla directory to index.cgi.

20. You should now be able to view Bugzilla using a web browser:
Part III – Client
Client Connection and Configuration

Each CVS client can connect to the server using any of the authentication techniques installed on the server. This section describes how to set up each of the most popular clients.

Note that setting up CVS Suite Server and Clients is NOT as simple as installing the software. The sections *CVS Architecture*, *Setting up CVS Suite Server* and *Server Administration* provide the necessary information to install and configure the CVS Suite system. If you do not follow these guides then severe performance problems can occur.

Client Time Synchronisation to CVS Server

For CVS Authentication to operate correctly, and to maximise performance of the CVS System CVS Clients must have their time synchronised to the CVS Server or the same time source as the CVS Server. If the CVS Client will use a network share or disk as a workspace / sandbox then the network share or disk should be synchronised to the same reliable time source as the CVS Server.

### Windows

- **Windows 2000**
  - `w32tm –s`
- **Windows XP/2003/2008**
  - `w32tm /resync /rediscover`

  For further instructions see:
  - [http://support.microsoft.com/?kbid=216734](http://support.microsoft.com/?kbid=216734)

### Unix

- **HPUX**
  - `/usr/sbin/ntpdate -u servername`
- **Solaris**
  - `/usr/sbin/ntpdate -u servername`
- **Red Hat ES Linux**
  - `/usr/sbin/ntpdate -u servername`
- **SuSE Enterprise Linux**
  - `/usr/sbin/ntpdate -u servername`

**Unix systems must run this command as root and the command can be put in a cron job to run at regular intervals. Alternatively some systems may have ntp client daemons that can achieve this task.**

General Guide to Authentication

**Storing passwords securely (CVSAGENT & Putty PAGEANT)**

By default CVSNT client will store your password insecurely. There are several ways this can be done:

- `cvs login`
- saved CVSROOT with password (WinCVS, TortoiseCVS, CVS Suite Studio)
- `ssh keys in unencrypted .ppk file`
We strongly recommend that you never let CVSNT use its default password storage. Do not create connection profiles that include passwords in CVS Suite Studio, TortoiseCVS, WinCVS etc.

**Safe passwords on Windows**

On Windows your connection string (CVSROOT) usually not contain a password or username, eg:

```
:sspi:host:/myrepo
```

When using SSPI protocol in this way Windows passes the authentication token generated at login to the server and you are authenticated against the active directory securely with no password ever being stored.

If you do need to use a username and password, eg: if you are using pserver protocol or SSPI without an active directory, then you should run CVSAGENT when you login to your desktop (put it in startup items). Do NOT use a password in your connection string, use only the username, eg:

```
:pserver:myuser@host:/myrepo
```

CVS Suite and all the client programs will automatically store your password in memory. Note: the password prompt will come from CVSAGENT and not from WinCVS or TortoiseCVS as previously. You should use this for any password based logins, including pserver, sspi (no active directory) and ssh.

**Safe keys on Windows with Putty SSH**

If you are using SSH to connect to the CVSNT Server from Windows, then often you will use a public/private key pair.

If you supply an unencrypted private key file in the connection string CVSROOT then this is just as insecure as storing your password in plain text, eg:

```
:ssh;privatekey=d:\me.ppk:user@host:/repo
```

Instead, run Putty PAGEANT on login (eg: in startup items) and load your private key, either from a password protected file on the local or network drive, or from a secure USB drive (eg: with a fingerprint reader or other 2FA method). CVSNT client will automatically ask PAGEANT to supply they key when required.

**PSERVER**

Many people set up their clients to use pserver authentication because that is the easiest, or the most commonly documented. This manual does not describe the pserver connection method because it is insecure and unreliable. It is possible to set up Eclipse, the Visual Studio .NET Integration, CVS Suite Studio, WinCVS, TortoiseCVS and Mac CVS X to connect using SSPI, SSERVER or SSH secure protocols.

**SSH**

SSH is a recommended secure protocol for connecting to CVS Servers hosted on Unix or Mac OS X. When using SSH with Tortoise or WinCVS then it is best to also use CVSNT Password Agent (see the section *CVSNT Password Agent* below for more information).
SSPI
SSPI is a recommended secure protocol for connecting to CVS Servers hosted on Windows. When using SSPI it is not necessary to use the “cvs login” command to log in to the CVS Server. Each time a command is ran the current windows credentials are passed to the server and each command is authenticated separately.

If you accidentally use the \textit{cvs login} command with the SSPI protocol then your password will be cached in the Windows registry and when you change your password then your client will stop authenticating successfully with the server.

Setting up CVS Suite Studio Client on Windows
CVS Suite Studio is the recommended tool graphical front end for CVSNT and provides a range of powerful commands for performing the most common CVSNT tasks in a clear and simple user interface. Functions include:

– Import
– Checkout
– Set and Manage ACL’s
– Browse files, versions and structure of the repository
– View the tip revision of each branch
– View History of each file in the repository
– Create modules / directories on the server

Also Tortoise may be used within the workspace window (right side) where you can use functions to manage the workspace such as:

– Update / Refresh
– Add new files
– Check in / Commit changes and new files
– Merges / Promotions
– Reserve (Local) versions

CVS Suite Studio has several levels of detail that can be displayed for the repository:

– Structure only (directories)
– Branches
– Files (and top version numbers)
Import
To import a directory or project and all the files contained in it to CVSNT drag the directory from the local view on the right to the server view on the left.

Hidden files are automatically ignored by the CVS Suite Studio and are not imported. If you want to always ignore other files see the cvsignore administration file for information on how to do this.

View Server Tags and Branches
Use the View pulldown menu to select what level of detail can be seen:
Checkout and Creating a Workspace

To copy a single version of a directory or project from the repository server to a personal work area (eg: C: drive or H: Drive) then drag the folder from the server view on the left to the local view on the right.

Synchronise a Workspace with the Server

Once you make changes to files in your workspace they will change colour to orange to indicate they need to be synchronised (committed) to the repository server. Right click the folder and choose the ‘CVS Commit’ option and then enter a comment in the dialog box:
Set and Manage ACL’s
To use ACL’s you first must configure the ACLMode, see the Administration section for more information. To deny access by default set `aclmode = normal` (if you do not set this CVSNT will allow access by default). Once the repository wide default is set, you can then use CVS Suite Studio to add or remove permissions to folder on the server:

Browse files, versions and structure of the repository
You can use the server side (left side) of the CVS Suite Studio to browse any CVS Suite Server on your network. You can also add a server by selecting the “Local Servers” then the File->New pulldown menu.
Folders are represented by the usual folder icon and branches and tags are represented by a small tree icon. Text files are shown in GREEN and binary files in RED. Selecting any file in the server view gives you a range of options for inspecting and reviewing:

**Open / Open With - View the tip revision of each branch**

View the file using the default viewer or a selected viewer. Note: Using Open on an executable file (e.g., fred.bat, hello.scr, or welcome.exe) will run that program on your computer.

**Annotate**

For text files – shows each line prefixed by the revision number that it was assigned, the name of the person who made the change and the date.
Log - View History of each file in the repository

This option returns a list of every change made to the file:

Create modules / directories on the server

Right click on any directory on the server side (left side) and select New Directory to create a new module / directory / project on the server

Setting up SCCI / Visual Studio .NET Integration on Windows

The Visual Studio .NET 2005 (SCCI) integration allows you to use CVS directly from Visual Studio and other SCCI compliant IDEs.

NOTE: the SCCI interface assumes that your workflow is similar to Visual SourceSafe, ie:

- Only source code is versioned (not compiled code)
- All work is reserved
- The CVS repository server is on a high speed network link to the local PC

If the above restrictions do not suit your CM process, then you probably do not want to use the SCCI plugin. It may be possible to use a combination of TortoiseCVS and the SCCI plugin – however that is not described here.

Performance of VS.NET integration

Note that setting up CVS Suite Server and Clients is NOT as simple as installing the software. The sections CVS Architecture, Setting up CVS Suite Server and Server Administration provide the necessary information to install and configure the CVS Suite system. If you do not follow these guides then severe performance problems can occur.

Performance problems are most often caused by connectivity issues. CVS Suite Client is very robust and will often appear to work but be very slow if the connectivity and authentication are not correctly configured.
Any performance problem between the CVS Suite client and CVS Suite Server will be exacerbated by the VS.NET integration. VS.NET often requests many small transactions with the CVS Suite Client—each of which the CVS Suite Server must authenticate (as opposed to a single large transaction that is authenticated once). Therefore performance problems that may not be a concern in other parts of the CVS Suite system can appear enhanced in VS.NET.

Once the system is correctly configured VS.NET and CVS Suite Server should perform equally with the other components of the system. If you are experiencing performance difficulties please contact support using the information at the front of this book.

Adding a project to source control
Use the pulldown menu option File->Source Control->Add solution to source control:
The TortoiseCVS Make New Module dialog appears with the new module name defaulted to the name of the solution. You may simply press OK to continue:

When the process is complete you may need to press OK on the Importing dialog. When the import is complete Visual Studio will show that all the files are imported/locked:

Check out for edit
To begin work on a file all you need to do is begin making a change. As soon as you begin to make a change CVS Suite will automatically put the file into edit mode. You can also right click on a file and select “Check out for edit”.

Comparing local file to original (or any other version)
You can compare the local file to the original file that you checked out, or any previous version or a version on another branch. Use the compare option on the right click menu:

The TortoiseCVS history dialog is then displayed for you to choose which version you wish to compare with. Select the version to compare with and right click and choose “compare with working file”: 
Open from Source Control
You can open a project directly from source control:

Select the method to find the project you want to open:
The first time you choose the browse method you may be promoted to unblock the browser:

Navigate to the module you wish to open from source control. It is very important that you open the “top” directory of the solution – not a subdirectory (or use the Define CVS Suite Server Manually option):

Press OK on the browse screen the OK on the main screen and your project will be opened from source control (you may be prompted for a password).
Switching a project from PushOK to CVS Suite (Bind to CVS Suite)
If you have previously used the PushOK connection between Visual Studio and CVS then when you open Visual Studio with the new CVS Suite SCCI integration you will be prompted to rebind the source control:

In the pending checkins window you will see the status of the bind:

After the bind has completed the Change Source Control window will show the new bindings:

Switching between several installed SCCI providers
Visual Studio .NET 2003 and many other IDEs do not provide a mechanism to allow the user to switch between several installed SCCI providers. Use the VS.NET & SCCI Integration Preferences icon to choose between several providers.
Some editions of Visual Studio .NET 2005 has its own settings to provide a mechanism to allow the user to switch between several installed SCCI providers. Use the pulldown menu Tools, then Options:

In the Options dialog box choose “Show All Settings” then choose the CVS Suite SCCI Provider:

VS.NET & SCCI Integration Properties
Use the *VS.NET & SCCI Integration Preferences* icon to change the default properties of the integration.

- Import options (defaults to exclusive –k+x)
- How to choose between ASCII/Unicode and Binary file types (by default the integration will let the server decide)
- Enable debug tracing (for support)
- Enable defect tracking integration (allows you to enter a “bug id” on checkout and/or checkin)
Setting up Oracle JDeveloper on Windows

Oracle JDeveloper is a popular free commercial IDE / Editor for use in Oracle Java programming environments.

**JDeveloper 10g (10.1.3)**

These examples are created using Oracle JDeveloper 10g and the JClient sample project.

**Configure**

After CVS Suite is installed on the PC configure JDeveloper Preferences with the Tools->Preferences pulldown menu. Set the CVS Client to use “External Executable”. Optionally you may enable “Run CVS in Edit/Watch Mode” to trigger notification e-mails and work with either an unreserved or reserved methodology.
Import

Perform Module Checkout

During step 5 of the import wizard the user may choose to *Perform Module Checkout*. This option is not supported by CVS Suite. Once the import is complete the project should be checked out using the CVS Browser.

Source and Object Code

By default JDeveloper only imports source code. If you want to import “compiled” java code you can modify the import wizard window or you can use CVS Suite Studio.
Adding a CVS Connection

Select Version System

Before you can work with CVS and JDeveloper you must select CVS as the chosen Version Control tool in the Versioning -> Select Version System pulldown menu and you must define the connection profile (CVSROOT) for the CVS server you want to use.

Add CVS Connection

JDeveloper cannot auto discover repository servers on your network and so you must enter the CVSROOT manually. Use the File->New… pulldown menu to bring up the “New CVS Connection” dialog:

Press OK to start the CVS Root connection wizard, and the OK again:
Enter the Access Method (usually SSPI) name of the host (use localhost if the CVSNT server is running on the same PC as JDeveloper) and the repository path then press OK.
This screen simply confirms that the Wizard has correctly constructed the CVSROOT. This value is identical to the value returned by the CVS Suite Studio repository properties window.

You can use this window to test that the connection is working correctly.
Client Connection and Configuration

Check Out and CVS Navigator
The JDeveloper CVS client includes a simple navigator that allows you to explore the CVS Repository. To check out a project browse the server for the module and then right click and select Checkout. Once the checkout is complete the module will be automatically added to your application list.

Commit/Checkin and Application Navigator
The JDeveloper CVS client includes a visual representation of modified files in the Application Navigator. Right clicking on any folder (or file) allows you to commit your changes to the repository.

Edit/Watch mode, Making Changes and Message Log
The JDeveloper CVS client includes a message log that shows the communication with the CVS Suite Server and any error messages. If you are using the Edit/Watch mode then you will see messages telling you who the other editors of the file are when you begin to make changes:

Setting up Eclipse Client on Windows
Eclipse is a popular IDE / Editor for use in Java programming environments; it is also the basis of the IBM WebSphere Application Developer. Please refer to the whitepaper-eclipse-cvs document for detailed setup and user guides.
Setting up IntelliJ IDEA Client on Windows

IntelliJ IDEA is a popular IDE / Editor for use in Java programming environments and it is also the basis for Android Studio. Please refer to the whitepaper-intellij-cvs document for detailed setup and user guides.

CVSNT Password Agent

The CVSNT Password Agent is a utility for Windows which caches passwords securely in memory. If you are using the pserver protocol then issuing a `cvs login` command will store your pserver password in the windows registry, however if you use the CVSNT Password Agent then it is stored in memory instead. If you are using the ssh protocol then you should not use `cvs login` and each command is authenticated separately – which requires that you re-type your password each time you do a new CVS command. Using the ssh protocol with CVSNT Password Agent securely caches the password in memory so that you only need to enter the password once per session.

Starting the Agent

Start the CVSNT Password Agent after loggin in to the workstation or add a shortcut to the agent in the “Startup Items” so it starts automatically when you log in. The CVSNT Password Agent will appear in the Windows System Tray:

Clear the passwords

Right click on the password agent and clear the password list before using it for the first time:
Entering a password

Now start TortoiseCVS to checkout using the SSH protocol and when you press “Fetch List” the CVSNT Password Agent will ask for your password.

The password will only need to be entered once rather than each time a command is executed.

Setting up TortoiseCVS Client on Windows

TortoiseCVS is a popular Graphical front end to CVSNT providing a user interface within Windows Explorer. TortoiseCVS may be used with either Unix or Windows based CVSNT Servers.

Configuring TortoiseCVS with a Windows CVSNT Server

Generally most users will configure Tortoise to connect to a windows based CVSNT server using the SSPI protocol. If you require an “ssh” style secure connection then we recommend using the SSERVER protocol which uses a SSH style connection without needing to set up and configure an SSH server on Windows. It is not recommended to set up a Windows based SSH server.

There is no need to set any preferences to work with TortoiseCVS and a Windows based CVSNT server. The protocol and connection details are set during Checkout.

Configuring TortoiseCVS with a Unix CVSNT Server

Generally most users will configure Tortoise to connect to a unix based CVSNT server using the SSH protocol. If you do not have an SSH server running on the Unix operating system and do not want to set one up then the ssver protocol is a good alternative. The SSERVER protocol which uses a SSH style connecting without needing to set up and configure an SSH server.

Tortoise may connect to the Unix server using its own SSH client (TortoisePlink) or the inbuilt CVS SSH client. It is recommended that you use the inbuilt SSH client.
You may need to set the CVS_SERVER preference to work with TortoiseCVS and a Unix SSH server. This setting specifies the command to run on the server, eg:
/Apps/cvs/bin/cvnsnt.

**CVSAgent with sserver and ssh authentication**

The CVSAgent is used to cache user passwords in memory without the need to ever store them in the Windows registry or on the Disk. This is a good security mechanism to use with the sserver and ssh authentication protocols.

Do not use CVSAgent with SSPI.

**Setting up Mac CVS X Client on Mac OS X**

Mac CVS X is the recommended tool graphical front end for CVSNT on Mac OS X and provides a range of powerful commands for performing the most common CVSNT tasks in a clear and simple user interface. Functions include:

- Import
- Checkout
- Add new files
- Check in / Commit changes and new files
- Merges / Promotions
- View differences between revisions (in a graph on side-by-side)
- Report on the history of a file (log)
- Reserve (Local) versions

**Introduction**

The Mac CVS X application has a console window which shows the results of most activities, including “log” (show history of a file) and “status”.

Most of your time using Mac CVS X is spent in the Browser Window. You can have one or more Browser Windows open, each of which represent a different workspace on your Mac. A workspace can be a directory with files checked out from CVS Suite Server, a directory with files that you wish to import into CVS Suite Server, or a mixture of both.

The first thing you will want to do is either import a directory into CVS Suite Server or checkout a directory from CVS Suite Server.

**Lock and Unlock**

The Mac CVS X client has a “lock” and an “unlock” function. Do not ever use these functions with CVS Suite Server. If you wish to use “reserved” locking with CVS Suite Server define the files as type -kx and disable client keyword expansion using the CVSNT Server control panel (or the /etc/cvsnt/PServer file) on the server. To “reserve” a file on the Mac CVS X client use the “Edit” command not the “Lock” command. If the file is defined as -kx then the “edit” command will “lock” it automatically.

**Preferences… General (CVSROOT)**

Before starting to use Mac CVS X you must have a CVS Suite Server, CVSNT Server or CVS Server installed and configured. The server can be the mac you are currently using, or another mac, windows, linux or unix computer. Download and install the correct server package and install and configure the server before continuing (see the sections CVS Architecture, Setting up CVS Suite Server and Server Administration).
If you have installed the CVS Suite Server on your local mac then you can create a repository by starting a terminal window and using this command (you will be prompted for an administrator password):

```
sudo cvs -d :local:/usr/local/cvsrepo init -a /cvsrep -d "test repository"
sudo chmod –R g+rwxs /usr/local/cvsrepo
```

Once you have created that repository or found a repository on your network you can use Preferences… screen in Mac CVS X to define the CVSROOT (the connection string that enables Mac CVS X to connect to CVS Suite Server):

![MacCvs Preferences](image)

Note: it is essential on the mac that you provide a username even if it is identical to your current username.
Side by Side Diffs preferences
For Mac CVS X to show “side by side” differences you must have a file comparison tool installed such as FileMerge (which comes with the free Apple developer CD), or BBEdit. In the Mac CVS X preferences set the name of your External Differences Viewer:

Login
Before you can work with the server (for import or checkin, diff etc.) you must log in using the Admin pulldown menu. When you select the Login option you may be asked for a password. Use the password for your user on that server (if you are using the same Mac for a server that you are using for Mac CVS X just use your mac login).
Setting up XCode Client on Mac OS X

XCode is the development environment for building applications in Mac OS X. XCode includes integrated Version Control clients.

Your code must already exist in CVSNT

The XCODE documentation states (version 2.5):

Choosing the SCM System for a Project

To use Xcode for source code control, your project’s files must already be part of a repository and checked out. To choose the source code management system used for a project, select the project in the Groups & Files list and open the inspector. SCM setup options are at the bottom of the General pane. Source code management integration is enabled when the Enable SCM option is selected. If a fatal error occurs—for example, if Xcode is unable to find the SCM system server—Xcode notifies you of the problem and gives you the option to disable SCM integration.

Supported protocols

Since the protocol is chosen when you perform the initial checkout (or import with create workspace -C switch), then there is no need for Xcode to understand the advanced protocol options of CVSNT. All supported protocols are available straight away. The XCODE documentation states (version 2.5):

CVS. Xcode reads the CVS root file in your project’s folder and sets itself up automatically. The CVS support works in both direct (mounted repository) mode and client/server (pserver) mode. If you can use a particular repository from Terminal, Xcode should also be able to use it. For more information on CVS, see CVS in Tools documentation.

Import your source code and make a workspace

It is necessary to use the command line client (or a GUI client such as Mac CVS X) to import your existing source code.
Command Line Login

$ cvs -d :sspi:username@server.host.com:/cvsrepo login
Logging in to :sspi:username@server.host.com:2401:/cvsrepo
CVS password:

Command Line Import

$ cd WindowFun

$ cvs -d :sspi:username@server.host.com:/cvsrepo import -Cnm "import sample" WindowFun
I WindowFun/.DS_Store
I WindowFun/build/.DS_Store
I WindowFun/English.lproj/.DS_Store
I WindowFun/WindowFunPB.pbproj/.DS_Store
N WindowFun/ReadMe.txt
N WindowFun/SampleUtils.c
N WindowFun/ToolbarPoof.png
N WindowFun/WindowFun.c
N WindowFun/WindowFun.h
.
.
.
N WindowFun/WindowFunPB.pbproj/project.pbxproj

No conflicts created by this import

Setting up WinCVS Client on Windows

WinCVS is a popular graphical front end for CVSNT and provides a range of powerful macros for performing bulk changes to your workspace.

Configuration

To use the “diff” functions in WinCVS you must configure WinCVS to use the WinMerge client included with CVS Suite. Open the preferences by choosing the “Preferences…” option from the “Admin” pulldown menu:
Set the “External diff” on the “WinCvs” tab of the “WinCvs Preferences window:

WinCVS CVSROOT and Creating a Workspace
You can use CVS Suite Studio, WinCVS or Tortoise to create a workspace.
You do not need to specify the CVSROOT in the WinCVS preferences – simply specify it at the time of checkout. Select “Checkout module…” form the “Remote” pulldown menu:
The information required to checkout a module from CVSNT is called the CVSROOT. You can enter the CVSROOT directly into the CVSROOT field on the checkout dialog or you can build it using the browse option:

Enter a repository path:

The complete CVSROOT will be displayed near the OK button, press the button when complete:
Either enter the name of the module in the “Checkout settings” dialog or press the browse button to take you to the “Select module” screen. From the “Select module” screen set the “List” option and press Refresh (F5) to find the module from the repository that you intend to checkout.

Select the module that you want to checkout and press OK to return to the Checkout screen. If all the details are correct press OK here as well:

As CVSNT checks out the module form the server the progress bar will move:
Using CVS Suite

This section is intended to give the practical workflow when working with one of the popular methodologies discussed in the theory section. This section assumes that your repository triggers integrations (e.g., keeping a read only “reference” copy of the source) and access controls are already set up.

CVS Suite Studio

CVS Suite Studio is the primary client graphical user interface of CVS Suite. CVS Suite Studio allows you to browse CVSNT repositories and to create workspaces containing individual files or entire modules and directories from either the Trunk or any branch or tag.

Browse

Open CVS Suite Studio and it will automatically discover any servers on your local network. It will discover servers located behind a router or on the other side of a VPN.

Browse the server by pressing the + icon to open modules and directories. If you wish to see branches and tags then select those options from the pulldown menu. The options in the pulldown menu only affect modules and directories not already opened with the + icon.

Create a workspace

Create a workspace on your local PC drive by dragging a file or folder from the CVS Suite Studio panel on the left to a directory on the right.

Import files and directories to the repository

Import files and directories from your local PC drive to the repository by dragging a folder from the CVS Suite Studio panel on the right to the server on the left. The directory on the PC is unaffected by this operation. To create a “versioned workspace” on the client PC follow the above section Create a Workspace.
Add a repository that is not already listed
If your repository is not listed automatically in the CVS Suite Studio then you can add it by selecting `File->new` from the pulldown menu. Do not specify logon details in the name – if a login is required CVS Suite Studio will prompt you.

Browse anonymously or specify a username and password
If a login is required to browse a repository then CVS Suite Studio will prompt you. If you want to check files into the repository or create a workspace that will allow `commit` then you would normally specify a user and password. If you are creating a work area for viewing only, or viewing and update then the anonymous user will usually be sufficient:

Operations on your workspace: Add, Commit etc
You can add files, commit (checkin) changes to your workspaces listed in the left hand window by right clicking on the folder. The actions available are identical to the Tortoise actions listed below.
Visual Studio Plugin (SCCI)

The Visual Studio VS.NET 2005/2008 (SCCI) plugin allows you to work with CVSNT directly from the Microsoft Visual Studio environment. Note: this integration is a generic SCCI and may also be able to be used from other IDEs installed on your PC. The CVSNT SCCI interface also provides a selector so that you can switch between different SCCI providers installed on your PC.

Modes of operation

There are four basic modes of operation with the VS.NET SCCI integration:

– Add an existing project or solution to a repository
– Connect an existing project or solution that was checked out from a repository to the Visual Studio SCCI plugin
– Create a new project or solution and add it to the repository
– Switch a project or solution from another CVS SCCI client to CVS Suite.

Ensure that CVS Suite is the selected version control provider

Use the Tools -> Options pulldown menu in Visual Studio to ensure that CVS Suite is the selected version control provider:

Connect an existing project or solution that was checked out from a repository to the Visual Studio SCCI plugin

This strategy is not recommended and may produce unexpected results. The Microsoft SCCI interface incorporates a workflow that is designed to make the developers interaction with the underlying CM tool very simple. If it is any different to decisions that you have made it may be more productive to use another CVS Suite tool such as CVS Suite Studio or TortoiseCVS instead. Also see the section below Switch a project or solution from another CVS SCCI client to CVS Suite.

Add an existing project or solution to a repository

To add an existing project or solution to CVS Suite Server use the ‘Add Solution to Source Control’ item in Visual Studio:
The next screen you are shown allows you to choose how to connect to the repository and the name of the cvs module that will contain the solution (usually the solution name):

The typical options are:

- **Protocol**: Windows Authentication (:sspi:)
- **Server**: hostname/dns name of server
- **Repository folder**: repository name chosen when you created the repository
- **Module**: same as solution name (you can press Fetch List... for a list of current modules)

When done press OK on the TortoiseCVS Make New Module dialog and the connection will be established:
Press OK to dismiss the TortoiseCVS Make New Module progress dialog and then CVS Suite VS.NET SCCI will add the solution to your repository. There is no visual feedback during this process and Visual Studio may seem unresponsive for a few minutes. When the solution has been added control is returned to VS.NET:

Create a new project or solution and add it to the repository
To create a new project or solution and add it to the repository simultaneously simply tick the ‘add to source control’ checkbox on the ‘new solution’ dialog and then follow the same procedure as for Add an existing project or solution to a repository described above.
Switch a project or solution from another CVS SCCI client to CVS Suite

To switch a project to CVS Suite use the Visual Studio Source Control -> Change Source Control:

Then select/highlight each project in the solution and press the bind button:

The procedure and dialogs after this are dependant on the source control provider you are moving from. See also *Add an existing project or solution to a repository* section above.

**TortoiseCVS**

Tortoise uses the command line CVSNT to communicate between your PC client and the CVSNT server.
Tortoise is a “plug in” for Windows Explorer, and it is automatically invoked in the “right click” context menu when you click on a file that is a part of a CVS sandpit.

After selecting a command the results are displayed in a pop up dialog. Using the Tortoise preferences this can be set to automatically close.

**Import (create project/modules in repository and add files to repository)**

To create a project structure (modules) in the repository and add files you use a client such as Tortoise.

**Create project / module from a directory**

Follow these steps to create a project structure in the repository from existing files:

The first step is to associate a directory accessible from your local PC with a module on the server (eg: projects/mysimpleapp)
The name and structure of the module on the server does not have to match the name of the directory on the client. In the Module field enter the project name / location that this folder will reside in the repository.

You also have to specify the server to create the new module in. Specify:

- Protocol (usually SSPI)
- Server (the name of the server computer)
- Repository folder (the repository prefix you defined when you created the repository)

**Add files to the newly created module on the server**

To mark the files in the directory for adding to the project / module on the server use the *add contents* option:
Commit and transfer files to the newly created module on the server

To transfer the first version of the files to the server use the commit action:

Using Defect Tracking Integration (Change Sets) with Tortoise

You can group your changes together using a change set number (bug number, job number etc) and if you are using defect tracking integration this will also record the activity against that bug / job. See the above section Testing of Integration (CVS Suite Tortoise) for an example.
Checkout a single file

You can checkout a single file using Tortoise or CVS Suite Studio. If you want to use Tortoise, right click on an empty directory and choose the Checkout option:

Enter the CVSROOT that designates the name of your server, authentication protocol and the path to the repository root.
Press the “Fetch List” button to obtain a list of all the modules on that repository:

Use the tree to open the module and directories that you are interested in and navigate through the directories.

When you find the file (or directory) that you want to checkout then select it and its name should appear in the Module box:

If you are checking out a single file you can also elect to specify a bug or job number that this work relates to:
Press the “OK” button to checkout the file:

Tortoise will automatically create an appropriately named directory for the file:

Open the directory and the file will be checked out and in edit mode ready for your changes:
Edit
If you are using CVS in a reserved mode then this step is required, otherwise it is optional. This step allows you to inform CVS that you are beginning work on an object.

Diff
Once you have completed your changes you may want to check that you have not left any debugging messages in the code, so perform a diff between the original repository version and your local copy.

Update
If your changes are correct you then need to synchronise your local copy with the latest copy on the server. This is most often required when not using reserved edit.
Add
If you have just created a new file that you need to add to the repository then you will use the Add command.

Tortoise will ask you to confirm if this is the correct file to add, and if so what the file format should be. If your administrator has set up this file type already then the default format will be correct.
Commit

When you are finished your changed files will appear in red, select then and choose commit. Alternatively you can commit the directory and CVS will find all the modified files for you.

Enter a comment about your changes and you can also optionally right click any of the selected files to view the diff's. When finished press the OK button to commit the changes to the server.
WinCVS uses a Windows Application to navigate the sandbox. The WinCVS GUI will be very familiar to people who have used tools like PVCS to manage source code.

**Edit**

If you are using CVS in a reserved mode then this step is required, otherwise it is optional. This step allows you to inform CVS that you are beginning work on an object.

The *edit* functions are available in the *trace* pull down menu.

**Diff**

Once you have completed your changes you may want to check that you have not left any debugging messages in the code, so perform a diff between the original repository version and your local copy.
The Diff Selection... option is available in the Query pulldown menu.

Update
If your changes are correct you then need to synchronise your local copy with the latest copy on the server. This is most often required when not using reserved edit.
The Update Selection... option is available in the Modify pulldown menu.

Add
If you have just created a new file that you need to add to the repository then you will use the Add command.
Commit
When you are finished your changes files will appear in red. To commit them to the server select the directory and WinCVS will find all the modified files for you.

The commit command is available in the modify pulldown menu.

Checkout
The workflow of WinCVS checkout is described above in the section Setting up WinCVS Client on Windows.

Eclipse and IBM WebSphere Application Developer
Eclipse is a popular IDE / Editor for use in Java programming environments; it is also the basis of the IBM WebSphere Application Developer. Please refer to the whitepaper-eclipse-cvs document for detailed setup and user guides.

Mac CVS X Client on Mac OS X
Mac CVS X is the recommended tool graphical front end for CVST on Mac OS X and provides a range of powerful commands for performing the most common CVST tasks in a clear and simple user interface. Functions include:
- Import
- Checkout
- Add new files
- Check in / Commit changes and new files
- Merges / Promotions
- View differences between revisions (in a graph on side-by-side)
- Report on the history of a file (log)
- Reserve (Local) versions

Introduction
The Mac CVS X application has a console window which shows the results of most activities, including “log” (show history of a file) and “status”.

Most of your time using Mac CVS X is spent in the Browser Window. You can have one or more Browser Windows open, each of which represent a different workspace on your Mac. A workspace can be a directory with files checked out from CVST Suite Server, a directory with files that you wish to import into CVST Suite Server, or a mixture of both.

The first thing you will want to do is either import a directory into CVST Suite Server or checkout a directory from CVST Suite Server.
Lock and Unlock

The Mac CVS X client has a “lock” and an “unlock” function. Do not ever use these functions with CVS Suite Server. If you wish to use “reserved” locking with CVS Suite Server define the files as type -kx and disable client keyword expansion using the CVSNT Server control panel (or the /etc/cvsnt/PServer file) on the server. To “reserve” a file on the Mac CVS X client use the “Edit” command not the “Lock” command. If the file is defined as -kx then the “edit” command will “lock” it automatically.

Login

Before you can work with the server (for import or checkin, diff etc.) you must log in using the Admin pulldown menu. When you select the Login option you may be asked for a password. Use the password for your user on that server (if you are using the same Mac for a server that you are using for Mac CVS X just use your mac login).

Import

Use the File menu “New Browser” function to choose a folder that you wish to import to CVS Suite Server. If there are several folders you are going to import, then select the folder one level above.
Now use the Remote pulldown menu and choose “Import”. A list of the file types will be displayed. Note: It is recommended that you enable “ignore client side keywords” on the CVS Suite server so that you can ensure that the server rules for file types always override the client rules. Press OK.

On the import settings screen, on the General tab select “Don’t create vendor chanch or release tags” (these are only used when working with vendor branches), and on the Options Tab select “Create CVS directories while importing”. Press OK to import
When the import has completed you can view the results in the console window:

![Console Window](image)

Checkout and Creating a Workspace

To create a workspace open a New Browser with the File menu and select the folder where you want the workspace to be created:

![Choose a Folder](image)

A new empty browser window is shown:
Note: it is not possible to browse all the server module names using Mac CVS X, if you do not know the name of the module type the “cvs -d :pserver:user@localhost:/cvsrep ls” command from the terminal window):

```
Terminal — bash — 66x8
Last login: Mon Apr 30 16:35:14 on ttyp7
Welcome to Darwin!
ti800:~ abarrett$ cvs -d :pserver:abarrett@localhost:/cvsrep ls
Listing modules on server

CVSROOT
newdir-local
ti800:~ abarrett$
```

Now use the “Checkout module…” command from the Remote pulldown menu. Enter the name of the module you are checking out from the server and press OK.
When the import has completed you can view the results in the console window, and you can expand the size of the browser window so you can see the checked out files:

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Ext</th>
<th>Rev.</th>
<th>Option</th>
<th>Encoding</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>new-checkout-dir</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>newdir-local</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>test.cpp</td>
<td>cpp</td>
<td>1.1</td>
<td>-kky</td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td>test.h</td>
<td>h</td>
<td>1.1</td>
<td>-kky</td>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>
```

Annotate

For text files – shows each line prefixed by the revision number that it was assigned, the name of the person who made the change and the date. Use the Query menu “Annotate” command:

```
1.1  (abarrett 30-Apr-07): // this is a test
1.1  (abarrett 30-Apr-07): // really
1.1  (abarrett 30-Apr-07): #include "test.h"
1.1  (abarrett 30-Apr-07): main ()
1.1  (abarrett 30-Apr-07): {
1.1  (abarrett 30-Apr-07):  printf("hello\n");
1.1  (abarrett 30-Apr-07):  exit;
1.1  (abarrett 30-Apr-07): }
```
Log - View History of a file
This option returns a list of every change made to the file, the results are shown in the console window.

Graph - View History of a file graphically
To view the history of a file graphically use the Query pulldown or the right click menu (if you are using a multi button mouse). You can select options on the settings screen or just press OK. The revision graph shows branches, revisions and merges (if you are using CVS Suite Server).
XCode Client on Mac OS X

XCode is the development environment for building applications in Mac OS X. XCode includes integrated Version Control clients.

Using the project from XCode

It is now possible to open this project in XCODE, choose "Get Info" from the Action menu with the project selected and enable version control "CVS" (CVSNT replaces CVS when you install it).

Open WindowFun.c and make some changes, save them and choose "Commit Changes" from the SCM menu. 

![XCode interface](image1.png)

![XCode interface](image2.png)
Find out about other peoples changes

You can reserve a file for edit using CVSNT so that you are the only one who will have access to it while you are using it. However there are times that you do not want to lock people out – but need to know when changes are made. For example if one program depends on another.

It is possible to request that CVS notify you when other users begin work on a file or finish work on a file using the notify command.

Notify

For Notify to work your repository administrator must have already enabled e-mail notification via the Notify configuration file using a tool such as CVSMailer.

To enable watches you should use the CVSNT command line.

Use the cvs watch on command to enable watches and then cvs watch add each file that you want to be informed about.

Good comments

When using configuration management the kind of comments you enter and where you enter them becomes important.

If you do not have a tool such as CVS to automatically track who has changed what then you often add comments in the source code to indicate your initials and the change date. You may want to continue this practice however CVS will now automatically track this same information for you.

What is important are the comments you make when returning the file through version control. If in six months time you are trying to determine why a change was made a comment such as “fixes for September” is of little value – the date of the revision tells you that.

Good comments include:

- Bug number(s)
- References to other people / sponsors
- References to other documentation
- A succinct explanation of the purpose of the change

Here is an example of a good comment:

*Bug 1234. Changes detailed in the functional spec h:\docs\fs23-1234.doc. Joe explained that the customer component should always display a warning before changes are written to the database. This is a first draft and probably needs some work.*

Here is a poor example of a comment:

*Changes by Arthur. I made some changes to the WRITE trigger. Used askmess. A part of the September work.*

If you have a defect tracking system it is possible for the CVS repository administrator to have your comments at checkin automatically added to the defect tracking database.
Release Manager

Release Manager client is typically ran on a web server or application server to keep a workspace up to date from an alias tag. For example, the alias tag: web-site-prod may always point to the latest version of the web site that should be on the server.

Release manager can also obtain files from a local release manager client. This is especially useful in web server farms. One server obtains the up to date site from the central server and the other web servers all FTP it from the first one. This saves time and bandwidth.
Release Manager Security
Release Manager must be allowed to contact the network to operate, when prompted allow the Release Manager to access the network:

![Windows Security Alert]

Registering a workspace
Start the Release Manager and press Add Server… to create a workspace schedule definition:
Select (or add) a CVS, CVSNT or FTP Server
Use the provided list of servers to select a CVS, CVSNT or FTP server. If the server you need to use is not displayed in the list then click button labelled New…

You must browse at least the 2nd level before you can select a folder to schedule the checkout. When you “open” the repository level folder (My Sunny Repository) then you may be prompted for a username and password. You can leave the username field blank and the current process username will be used:
Once the server, repository and module are selected then you can set up the schedule:

The minimum information to set are the checkout interval, starting time and checkout to directory. After each execution the starting time will be updated to the next execution time automatically.
You may also customise the CVS Connection string with the Edit Root button:

Select (or add) a FTP Server
You can configure Release Manager to automatically FTP and unzip a file from an FTP server. Another host in the same subnet can run the FTP server service and also run Release Manager configured to checkout and create a ZIP file. This way Release Manager can update an entire server farm whilst using as little bandwidth as possible on an external network connection.

Use the provided list of servers to select a FTP server. If the server you need to use is not displayed in the list then click button labelled New…
You must provide the correct information in the FTP definition screen since it is not validated here at all. The Directory must be the name of a ZIP file:

![FTP Definition Screen](image)

Once the FTP server and directory/file is selected then you can set up the schedule:

![Schedule Setup](image)
The minimum information to set are the checkout interval, starting time and checkout to directory. After each execution the starting time will be updated to the next execution time automatically. If you use the “add tag” option then you can use CVS Suite Server to identify which releases/versions of each file have been successfully deployed to each release manager client.

You may also customise the CVS Connection string with the Edit Root button:
Selecting a Schedule Username
The scheduled processes must run as a particular user. When you exit the release manager it will prompt you for the username and password:

What happens when errors occur
If the scheduled task fails three times consecutively then the task will be automatically disabled from running. This behaviour can be altered by setting the Do Not Stop setting on the Edit Schedule dialog.

Allowing Release Manager Client to run on Windows Vista:
To run the Release Manager client on Windows Vista you must follow the actions in the next section Allowing Release Manager through Windows XP or Windows Vista Firewall plus also enable Release Manager to run at a higher privilege level with the following steps:

- Set the releasemanager.exe properties to run as administrator
- Create the first Release Manager task (as described above)
- Alter the Release Manager task in the scheduler to run as administrator
- Perform the action sin the section Allowing Release Manager through Windows XP or Windows Vista Firewall
Set the `releasemanager.exe` properties to run as administrator

Open the folder where you installed CVS Suite and the sub-folder “CVSNT” contains the file `releasemanager.exe`. Right click the file to open the properties window. In the properties window on the Compatibility tab check the box “Run this program as an administrator”:

![Image of properties window]

Alter the Release Manager task in the scheduler to run as administrator

Open the Control Panel and go to the system maintenance page, click the link at the bottom labelled “Schedule tasks”:

![Image of Control Panel]

In the task scheduler browse to the Task Scheduler Library:
Right click on the task *CVSNT Release Manager Timer* and go to the properties window. On the general tab tick the box labelled “Run with highest privileges”, and then press OK (you will be prompted to enter your password):

Allowing Release Manager through Windows XP or Windows Vista Firewall:
To run the Release Manager client on Windows XP (SP2) or Windows Vista you must enable the firewall to allow the program to allow external network connections.

*Windows XP*

Use the network connection control panel to allow Release Manager external connections to the internet in Windows XP.

Right click on the network connection and select properties, then choose the “Advanced” tab and press the “Settings button”.

The firewall can be switched on however exceptions must be allowed as shown on the following screen:
To enable Release Manager to access the server CVSNT server, add it to the exceptions list:

**Windows Vista**

Open the Control Panel and go to Security. Click “Allow a program through Windows Firewall” to allow Release Manager external connections to the internet in Windows Vista. You will be prompted for permission to continue before the Windows Firewall Settings dialog appears.

To enable Release Manager to access the server CVSNT server, add it to the exceptions list, press the “Add Program” button and choose release manager from the list of programs, and then press OK:
Modifying Schedule Master Process

The scheduler master process runs once every minute to check for schedules that need to run. This can be viewed in the Schedules control panel in XP (left) or in Vista (right):

![Modifying Schedule Master Process](image)

Modifying Schedule Username

The scheduler master process controls which username will be used to run the processes. Select the master schedule using the above procedure then select properties to change the username and password:

![Modifying Schedule Username](image)
Disabling all Schedules
The scheduler master process can be disabled (see previous dialog in the section *Modifying Schedule Username*) – however if you run ReleaseManager and press OK to close then it will be automatically re-enabled.
Part IV – Appendix
What's new in CVS Suite 2009

CVS Suite 2009 is the fifth release of CVSNT by March Hare Software Ltd. CVS Suite 2009 is a major upgrade to CVS Suite 2.5.03 and CVS Suite 2008 (CVS Suite 2008 was an incremental upgrade to CVS Suite 2.5.03 providing an easy migration for existing customers and is still supported).

CVS Suite delivers features that are essential when implementing an enterprise wide versioning solution. March Hare Software has worked closely with the original open source developers to strengthen CVSNT for the requirements of a commercial environment.

- Multi lingual filenames
- Synchronise Servers
- Server side external diff
- Plugins, including
  - Comprehensive Audit
  - Defect Tracking (Bugzilla, Jira and Mantis)
  - Build Management
  - E-mail Notification
  - Shadow Update (Reference Copy Update)
- Support for OS X resource fork extensions
- Utf-8 Server
- Rendevous / Bonjour / Zero Config support
- New native file access system for Windows
- Multi threaded deadlock server
- Microsoft MSI Installer
- Update by bug number
- Definable Keywords
- General Improvements
- Versioned Rename
- Client Installer Package
- Visual Studio Integration (SCCI)
- CVS Suite Studio
- Release Manager

Why should you upgrade from CVS Suite 2.5.03 or 2008 to CVS Suite 2009:

- Improved performance, particularly for update and short operations
- Distributed servers (caching servers, master servers)
- Improved WAN performance with large files
- Support for Windows Server 2008 R2, Windows 7 Professional on both 32 and 64 bit platforms (as a 32 bit application)
- Client compatibility with CM Suite Server 2009
- Support for Bugzilla 3.2 and 3.4
- Support for Visual Studio 2008