WANdisco CVS High Availability Administration Guide

Revision History

REVISION	DATE
1.0	October 2008
2.0	October 17, 2008

Contents

Contents, cont'd

Contents, cont'd

Contents, cont'd

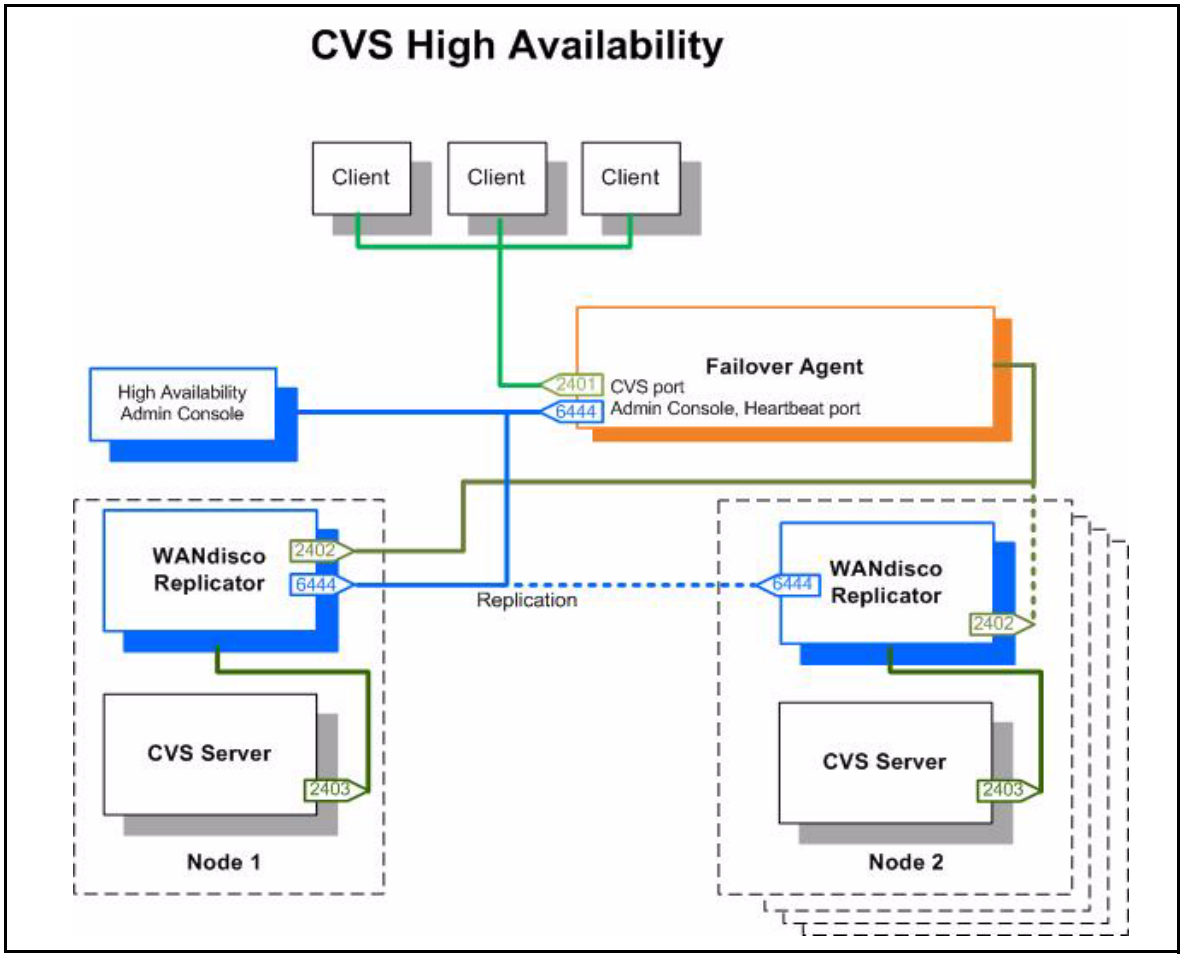
Contents, cont'd

1 Introduction

Welcome to the WANdisco world of replication. WANdisco CVS High Availability provides high availability and disaster recovery for CVS. A WANdisco CVS High Availability deployment consists of a Failover Agent and multiple CVS servers, each with an associated WANdisco proxy, functioning as exact replicas of each other. It offers CVS users business continuity. If one CVS server fails, the Failover Agent transparently directs users to the next available replica.

WANdisco CVS High Availability guarantees RPO (Recovery Point Objective) equal to 0, i.e., zero data loss, even if the failure occurs in the middle of a transaction. Your High Availability group of CVS nodes are synchronized at all times: each CVS database is a functional equivalent of the other CVS databases. High Availability employs WANdisco replication technology to ensure that all replicas remain synchronized, so that if one of the CVS servers goes down for some reason, there is zero data loss.

WANdisco provides an Admin Console, a web-based user interface, to administer and monitor the HA group.



Users access WANdisco High Availability with a standard CVS port, by default 2401. WANdisco HA configures an additional port (by default 6444) for replication communications (known as DConE) using the DConENet protocol. The Admin Console also communicates through this port, using HTTP protocol.

The CVS user connects to the WANdisco Failover Agent on the standard CVS port (configurable), such as 2401. For the replicators, the CVS client port is by default 2402, which helps ensure that no CVS user can bypass the replicator and connect directly to CVS. The actual CVS server port is changed again; by default it is 2403, further isolating the CVS server from any attempt at direct access.

1.1 How Replication Works Within an HA Group

The nodes in the HA group continuously communicate any CVS transactions that users are making. The HA group has to agree to an order for the transactions.

The group establishes transaction ordering through the agreement of the quorum. By default, High Availability has a majority response quorum, which means a majority of the nodes must agree on the transaction order. So if the group has five nodes, three of them must agree on the position of a transaction in the transaction order.

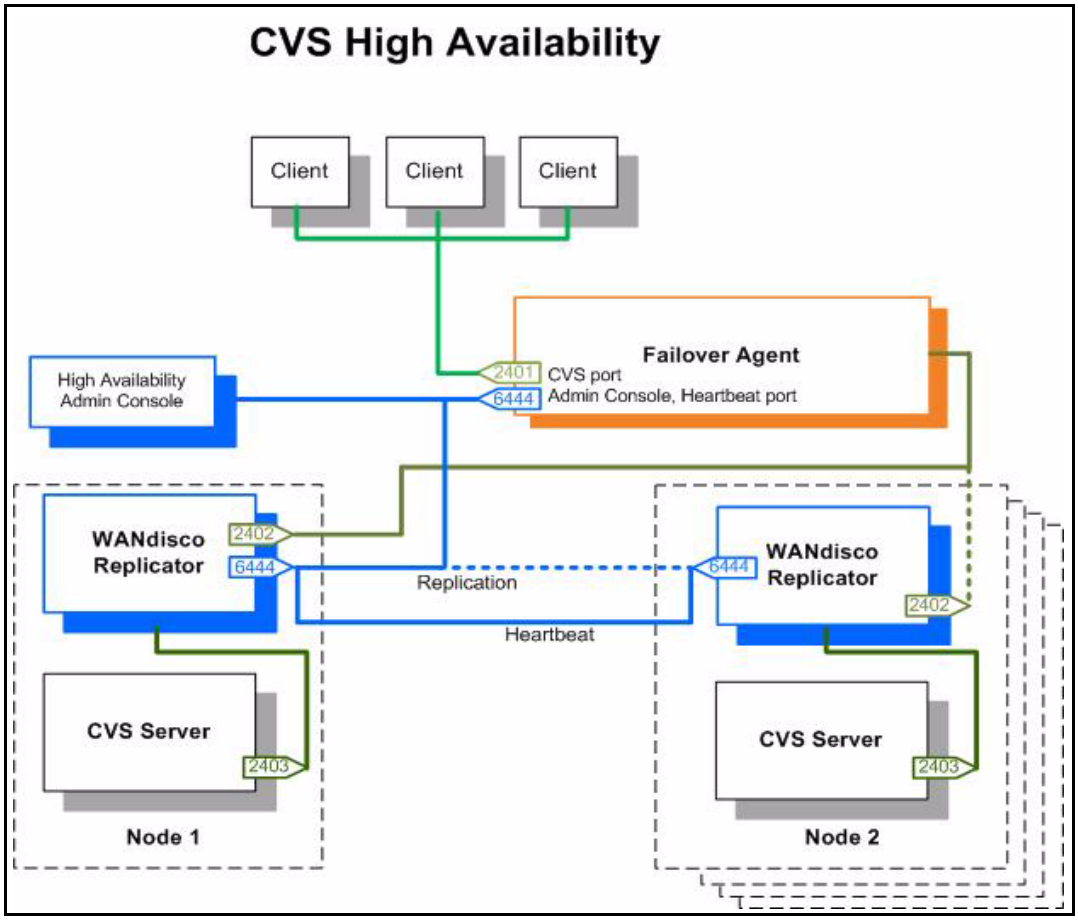
Having a majority quorum also ensures that if one node goes down in an HA group, the other nodes can continue uninterrupted. The HA group catches up the errored node when that node rejoins the group.

An HA group needs at least three nodes to avoid having a single point of failure. In a two node group, one node has the potential to be a single point of failure, which is why WANdisco does not recommend having just two nodes in a group. If you are contemplating a two node group, please read section 1.8, Failover Group of Two Replicators.

1.2 Failover and the Heartbeat

The Failover Agent continuously sends out a heartbeat, or an “are you alive?” message, to each replica, on the WANdisco port using the DConENet protocol. By default the heartbeat messages are sent at intervals of one second. The replicas respond to this with an “I am alive” message. The Failover Agent expects a response from each replica. If the Failover Agent does not get a response within a configurable number of heartbeats, the Failover Agent marks that node as unresponsive. HA administrators can change the heartbeat interval and missing heartbeat count in the Admin Console.

If the Failover Agent has marked node 1 (the current node) as unresponsive, actual failover to the next node occurs lazily, that is, only when a CVS client request comes through. This reduces the number of false failover alarms, as a replicator may not respond within the configurable number of heartbeats during a restart, or a node may be restarted before the next client request, eliminating the need for failover.



1.3 Failover Sequence

The Failover Agent passes CVS transaction data to only one node. That node then replicates the transactions to the entire group, so that every node in the High Availability group is a functional replica of each other.

1.3.1 The Current Node and the Designated Node

The node that communicates with the Failover Agent is called the “current” node. In normal operation, the current node is also the “designated” node, the node the Failover Agent expects to talk to. If the designated node should fail, and failover occurs, the current node would become node two, however node one is still the designated node. When node one comes back online, the Failover Agent resumes talking to node one (which again becomes the current node as well as the designated node).

1.3.2 Priority Order

During HA installation, you determine the failover order for the group. For example, node 1 is first in the failover order (priority one), node 2 is the second in the failover order (priority two), and so on.

The first HA node, as the current node, receives all CVS transactions through the Failover Agent. If that node fails, the Failover Agent immediately begins communicating directly with node 2. If node 2 fails, then the Failover Agent immediately begins communicating with node 3, and so on. The order is referred to as the priority order, meaning the order the Failover Agent takes during a failover.

If node 1 comes back up, the Failover Agent immediately returns to communicating with that node.

You can see the failover priority order by looking at the value for the `PriorityOrder` element in the prefs.xml file (shown in **bold** in the following example). In the example, you see that this node is the current node, the first node in the priority order.

```
...
<ServerProxy>
   <ListenerIP>192.168.1.184</ListenerIP>
   <ListenerPort>2401</ListenerPort>
   <PriorityOrder>1</PriorityOrder>
</ServerProxy>
...
```

1.4 The Distinguished Node

Each HA group has a distinguished node (user-assigned at installation). If there are an even number of nodes, the distinguished node acts as a tie breaker. For example, in a four node group, say two nodes want transaction X to be committed first, while the other two nodes want transaction Z to be committed first. The distinguished node's vote carries a heavier weight, and its transactions always take precedence.

1.5 Replication Example

Here is an overview of what occurs when a write transaction is received by the current node in the HA group.

Step 1 The originating client sends the transaction to CVS, passing through the Failover Agent.

Step 2 The Failover Agent relays the information to node 1 (the current node).

Step 3 Node 1 replicates the transaction throughout the HA group.

Step 4 Transaction data is successfully received by the quorum (a majority of the nodes). The quorum agrees and assigns it a Global Sequence Number (GSN).

Step 5 High Availability passes the transaction data to CVS.

Step 6 CVS processes the transaction.

Step 7 High Availability waits for CVS to complete the transaction. High Availability only marks the transaction as complete when CVS returns a completion status.

If for some reason node 1 goes down during this process, the Failover Agent communicates with the next node in the failover order. Through replication, all nodes have a record of all transactions, and so are immediately available to act as the current node, should the need arise.

1.6 Replication and Site or Network Failures

High Availability defaults to majority quorum, which ensures continuos operation of the replication group, as long as a majority of the nodes are up. That is, if there are five nodes in an HA group, and three go down, then replication is suspended until at least one more node rejoins the group (three being a majority of the nodes).

1.6.1 Site Failures

Say you have a five node High Availability group on your LAN. One of the sites goes down. If that site was the current node, then the Failover Agent begins communicating (fails over) to the next node in the priority order. Replication continues uninterrupted.

If that site was not the current node, then replication continues uninterrupted and the Failover Agent does not need to take any action. Replication continues unless a majority of the sites go down.

As soon as a site comes back up, the replication group catches up the site on its missing transactions, so that all sites are again synchronized.

The recovery infrastructure and details of WANdisco fault-tolerance can be found at http://www.wandisco.com/pdf/dcone-whitepaper.pdf.

1.6.2 Failover Agent Failures

A watchdog runs on the Failover Agent, so if the Failover Agent crashes, the watchdog immediately restarts it. If the machine should crash, service is unavailable until you reboot and restart the Failover Agent.

If you do not want to wait for the Failover Agent machine to be restored, you could run the failover agent on a hardware cluster. The Veritas Cluster Server is an example of a commercial solution. See http://www.symantec.com/business/products/overview.jsp?pcid=pcat_business_cont&pvid=20_1.

Linux-HA is an example of an open-source solution. See http://www.linux-ha.org/.

1.7 Establishing a Replication Baseline

When you deploy High Availability, you must ensure that all the replicas start out in sync, meaning that all of them are identical. Once High Availability is deployed, WANdisco's replication technology ensures they remain in sync.

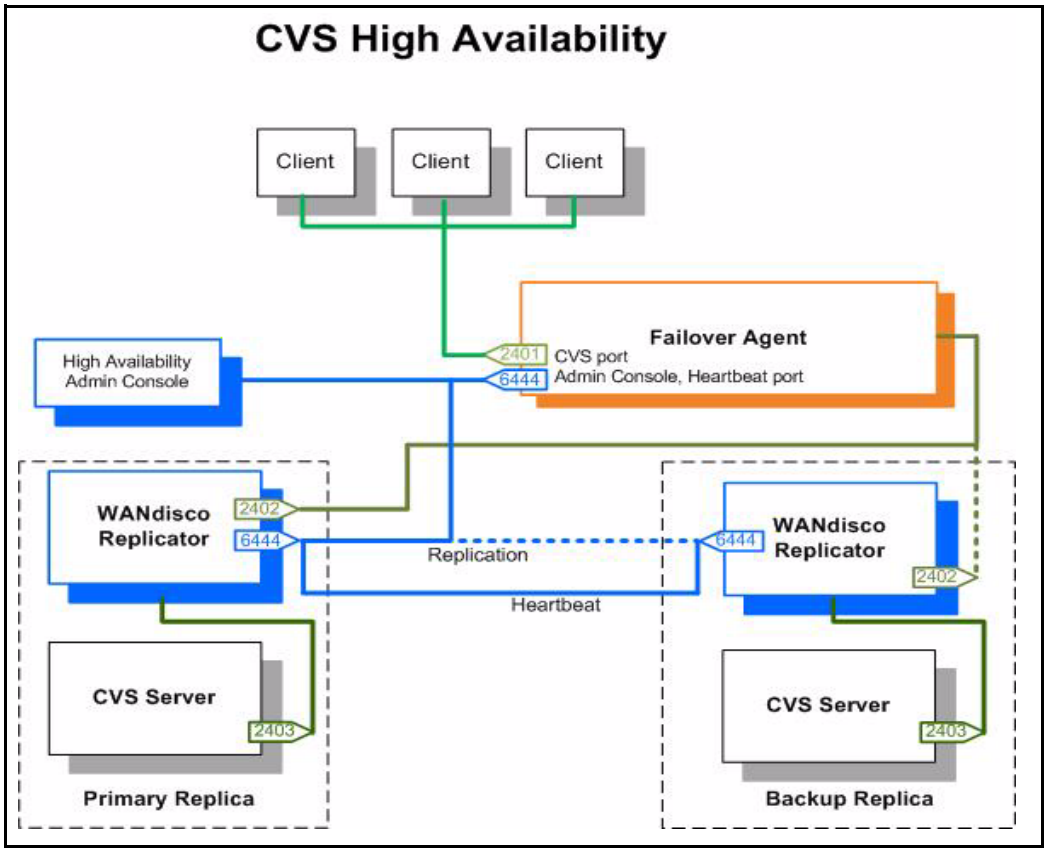
You start with one CVS repository, referred to as a replication baseline. To create this baseline, follow the procedure in 5.2, Establishing a Baseline for Replication.

1.8 Failover Group of Two Replicators

WANdisco strongly discourages using High Availability with just two nodes. Having only two replicators in the HA group forces changes to the failover mechanism. Whereas the objective of a larger failover group is continuous operation, the main objective of a two-replicator based deployment becomes dealing with a single replicator node failure.

In a two node deployment, High Availability defaults to singleton quorum, and the distinguished node is the second node.

Please see Chapter 6, Procedures for Two-Node HA Groups.



1.8.1 What Happens If the First (Designated) Node Fails

If the designated node fails, the Failover Agent engages the second node, setting the “failed to backup” flag. In order to have the designated node rejoin the HA group, you must perform some steps in the Admin Console, where a wizard guides you.

1.8.2 What Happens If the Second Node Fails

If the second node fails, users see no interruption, since the CVS transactions are engaging the first node anyway. The Admin Console displays the current status. Since the second node is the distinguished node, its failure forces the first node to go into unilateral mode, setting the unilateral flag. Returning the distinguished node to the replication group is described in 6.2, Recovering from Backup Node Failure.

1.9 Terms

You should familiarize yourself with these terms.

TERM	DEFINITION
replica	a CVS instance that is an exact equivalent or copy of another CVS instance. In WANdisco's High Availability product, a replica is also called a site.
replicator	It is the intermediary that acts as an application proxy/gateway between the Failover Agent and a given SCM server. Each *Replica* has an associated *Replicator*. It coordinates with other peer replicators to ensure that all replicas of the SCM repositories stay in sync with each other.
Failover Agent	It is the intermediary that acts as an application proxy/gateway between the CVS client and the replicators. The Failover Agent keeps track of which replicas are available, and proxies the CVS client's request to one of them.
replication group	a collection of replicators that work together to keep replicas of a CVS repository in sync.
current node	The node the Failover Agent is currently communicating with. If it is not the designated node, then a failover has occurred.
designated node	The node the Failover Agent expects to talk to. The node with the priority order 1 (as listed in the prefs.xml file).
one copy equivalence	all replicas are functionally equivalent copies of each other
GUID	Globally Unique Identifier. WANdisco CVS High Availability assigns each node a GUID on installation. The nodes identify each other by their GUIDs.
site or node	a server on which is installed a replicator and a replica. The sites comprise the replication group.
distinguished node	The distinguished node acts as a tie breaker for a majority quorum when there are an even number of nodes, making the final decision on replicator operations.
Quorum	A set of nodes that can reach agreement without participation from all nodes. In the case of an even number of nodes, the distinguished node settles a conflict. Quorum is defined in the prefs.xml file. HA by default has Majority Response quorum.
prefs.xml	The preferences files contain information on the replication group. Each site contains all preference files for the entire replication group. The files are specific to each site. The preference files are located in `cvs-ha\config`.
SCM Repository	Software Configuration Management repository like CVS
SCM Server	A network server that provides remote access to an SCM Repository

TERM	DEFINITION
installDir	this is the installation directory for WANdisco High Availability, for each node and the Failover Agent.
DConE	WANdisco's Distributed Coordinated Engine, the software engine underlying replication.
Install node	The node where you run the High Availability install program.
heartbeat	mechanism the Failover Agent uses to monitor availability of all HA nodes
heartbeat interval	the interval, in seconds, the Failover Agent waits to send an "are you alive?" message to all HA nodes
heartbeat connection timeout	The time the Failover Agent waits before assuming the non-responding node is unavailable. If the designated node is unavailable, this triggers failover, and the Failover Agent begins communicating with the next node in the priority order.

2 Recommended Deployment Practices

2.1 Administrative Pre-requisites

This guide is intended for a CVS administrator or a user who is reasonably comfortable with:

- Setting up a CVS based repository
- Configuring inetd/xinetd service on Unix/Cygwin or Windows service
- Installing Perl and required Perl modules
- Installing Java
- Unix or Windows system administration

If you don't meet the above pre-requisites, you may want to contact your CVS administrator or request that WANdisco perform a professional install for you.

2.2 Physical Environment

WANdisco strongly recommends that you follow these guidelines to ensure the successful installation and use of High Availability:

- a dedicated server for the Failover Agent
- running servers for each node in the HA group, pre-configured with
  - the same operating system
  - the same version of CVS server
  - a command line zip/unzip utility
  - Java (see Appendix A - Installing Java and Perl)
  - Perl (see Appendix A - Installing Java and Perl)
  - browser with network access to all servers
- e-mail from WANdisco containing the tar file link and the attached production licence key file

2.2.1 Firewalls and Virus Scanners

2.2.1.1 Firewalls

You must determine if your HA group sits inside a firewall or outside of one. If the HA group is inside a firewall, the HA group ports are untouched by the firewall and you need take no action.

However, if any part of your HA group sits outside a firewall, you must configure the firewall to not block or filter the port numbers you specify during installation.

2.2.1.2 Virus Scanners

If you have a virus scanner running on your network, you must configure it to not filter traffic on the ports you specify during installation.

2.3 CVS User Password Management

WANdisco recommends you allow High Availability to manage CVS user passwords across the HA group. The Admin Console offers an easy user interface for administrators.

If you do not allow WANdisco to manage CVS user passwords, you must ensure that each CVS user is entered also in High Availability.

During the installation process, you are asked if you want WANdisco to handle the CVS user passwords.

2.4 Deployment Checklist

You may be familiar with the Deployment Checklist from an evaluation copy of CVS High Availability. It is included here as reference. .

System Setup ❖ All sites must share the same operating system	
Supported Operating Systems	Fedora (32 or 64 bit): 6, 7, 8, 9 Red Hat Linux Enterprise Server (32 or 64 bit): 4, 5.2 Sun Solaris (32 or 64 bit): 9, 10 Linux: Linux kernel 2.6 or higher CentOS-4 Windows Server, (32 or 64 bit) 2003 Please read this for more information on NPTL. **Note:** VMware has a tendency to become unresponsive due to memory paging issues even without WANdisco present. Extra tuning may be needed to ensure optimal performance.
CVS Server Versioin	Must be the same on all sites. CVS version 1.11.17 - 1.11.23 CVSNT 2.0.x
CVS Client Version	Compatible with local CVS servers
System Memory	Ensure RAM and swapping containers are at least three or four times the largest CSV file you have. Recommended: 1 GB RAM; 2 GB swapping container
Disk Space	• CVS: depends on the number of projects and issues • High Availability Transaction Journal: Recommended - equivalent of seven days of changes • 100MB for Failover Agent
Failover Agent	The most reliable hardware (lowest failure rate), at least two CPUs greater than 1.8GHz each
File Descriptor limit	Ensure hard and soft limits are set to 64000 or higher. Check with the `ulimit` or `limit` command.
Journaling File System	Replicator logs should be on a journaling file system, for example, ext3 on Linux or VXFS from Veritas. Notes: NTFS is not a journaling file system: ext4 is a journaling file system, however WANdisco does not support its use because of its deferred writes.
Maximum User Process Limit	At least three times the number of CVS users.
Java	Install JDK 1.5.0. **Note:** There should not be any spaces or control characters in the path where Java is installed. For example, c:\Program Files\java does not work with WANdisco as a JAVA install directory. See Appendix A - Installing Java and Perl.
Perl	Install version 5.6.1 or later. Optional modules Date::Manip, File::Which, Term::ReadKey. See Appendix A - Installing Java and Perl.

Network Setup	
Reserved Port (i.e. 6444, another for synchronizing)	CVS High Availability needs a dedicated port for DConENet (replication protocol) and HTTP (for Admin Console). WANdisco also recomends having a port available in case you have to copy (rsync) the repository from one site to another. If your network has a firewall, notify the firewall of the port numbers.
Firewall virus scanner	Notify the firewall virus scanner of the CVS High Availability port numbers.
VPN	Set up IP sec tunnel. Ensure WAN connectivity.
Persistent Connection Keep Alive	Ensure VPN doesn't reset persistent connections for WANdisco, or else ensure there are no RST bugs
Bandwidth	
DNS Setup	Best to use IP address for WANdisco related hostnames, or else ensure there is DNS availability
CVS pserver Set Up	
CVS Server Version	All sites must have the same version
Repositories are synchronized	Before starting WANdisco, ensure that repositories at all sites are in sync (See 5.2, Establishing a Baseline for Replication).
User Name/Password database	Must be the same at each site. Best practise is to use CVSROOT/passwd.
Use port 2401 for replicator	Using the standard port 2401 avoids confusion. If you are not going to use 2401 for the replicator port, rename the pserver entry
file permissions in CVSROOT	Does the CVS user have write permissions? See article at http://www.linux.ie/articles/tutorials/managingaccesswith-cvs.php?style=printer.
The `--allow-root` option for cvsroots	Is the `[x]inetd conf` verified for cvspserver set up?
CVS triggers (`loginfo`, etc) for e-mail	Ensure e-mail triggers only fire from one location. See 5.18, Using CVS Triggers for Sending E-mails.
CVS `commitinfo` triggers	Ensure deterministic behavior at each site (either yes or no)
CVS tmpdir has adequate space	Defaults to /tmp : Ensure the size is three times the largest file
Xinetd/Inetd Set Up for cvsperver (Not applicable to Windows Platform)	
`/etc/services` cvspserver port	Modify cvspserver port to avoid conflict with replicator port. We recommend changing cvspserver 2401/tcp and 2401/udp to cvspserver 2402/tcp and 2402/udp. Please make sure that there is only one cvspserver entry. For example, you cannot have cvspserver on 2401 and 2402.
`/etc/xinetd.d/cvspserver`	Renamed service file or changed cvspserver port to avoid conflict with replicator port.

Increase Connection Per Second (CPS) limit	For example, increased to "10000 30"
per_source limit	Make it unlimited
instances limit	Make it unlimited
Restarted `[x]inetd` daemon	Ensure changes go into effect by restarting. This can be accomplished by executing the following command as **root**: `/sbin/service xinetd restart`
CVS High Availability Application Server	
Agreement Threads	Tune based on number of concurrent CVS writers
Reader/Writer Network IO Thread Pool	Tune based on CVS client connection rate, file transfer rate
ConnectionKeepAlive timeout	Tune inactivity timeout for persistent DConeNet/DFTP connections based on VPN/WAN router set up
Message Queue Max Thread Pool Size	Tune based on CVS write concurrency
Maximum connections per IO thread	Tune if active CVS user population is large (greater than 100)
Log Rotation	Modify `log.properties`
HeartBeatInterval	Tune based on WAN latency. Ensure the interval is twice as long as the maximum ping time between the Failover Agent and the replicator.
MissingHeartBeat Count	Default is 4. Increase this number if you see too many false alarms (spurious failover events).
Disk space for recovery journal	Provision large disk for `logs/tmp`, at least number of commits within a two to four hour window
Admin Email Address	To generate email notifications from the failover agent. Requires /usr/sbin/sendmail
SSH Set Up For CVS Client Side	
`/etc/prefs.conf`	Generated prefs.conf at setup or with install-cvsrelay utility
`CVSROOT/passwd`	Ensure same cvsrelay password for each user in CVSROOT/passwd file, generated using cvspasswd utility

3 Installation

Before you start the installation, make sure you have met the requirements listed in the first paragraph of Chapter 2, Recommended Deployment Practices.

Current CVS users must back up their current repository to use it as a baseline for replication. See 5.2, Establishing a Baseline for Replication.

3.1 First Time Installation

You can run the installation program from any machine. The output is zip files, which you then install on the High Availability nodes and the Failover Agent.

NOTE:

This procedure is for an HA group of three or more nodes. For a two node group, see 3.1.12, First Time Installation for Two-Node Group.

Step 1 Save the cvsha.tar.gz file.

Step 2 For Windows, create a directory, and unzip the file in that directory. For other platforms, untar the file. The uncompressed file produces a directory, `cvs-ha-installer`.

Step 3 Copy the license key file to the `config` folder in the `cvs-ha-installer` folder.

Step 4 At the command prompt (or editor), go to `cvs-ha-installer/bin`.

Step 5 Type

```
perl setup
```

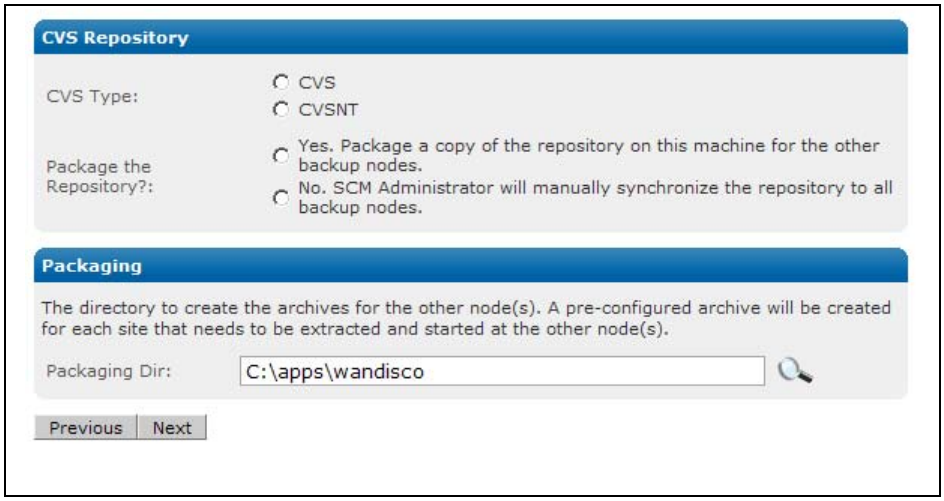
The last line of text returned contains a WANdisco URL.

Step 6 Copy the URL returned in the last step and paste it into a browser. The High Availability Welcome page appears.

Step 7 Click **Continue** at the Welcome text.

Step 8 Read the User Licence Agreement. You must agree to the terms to continue.

3.1.1 Configuring the High Availability Group



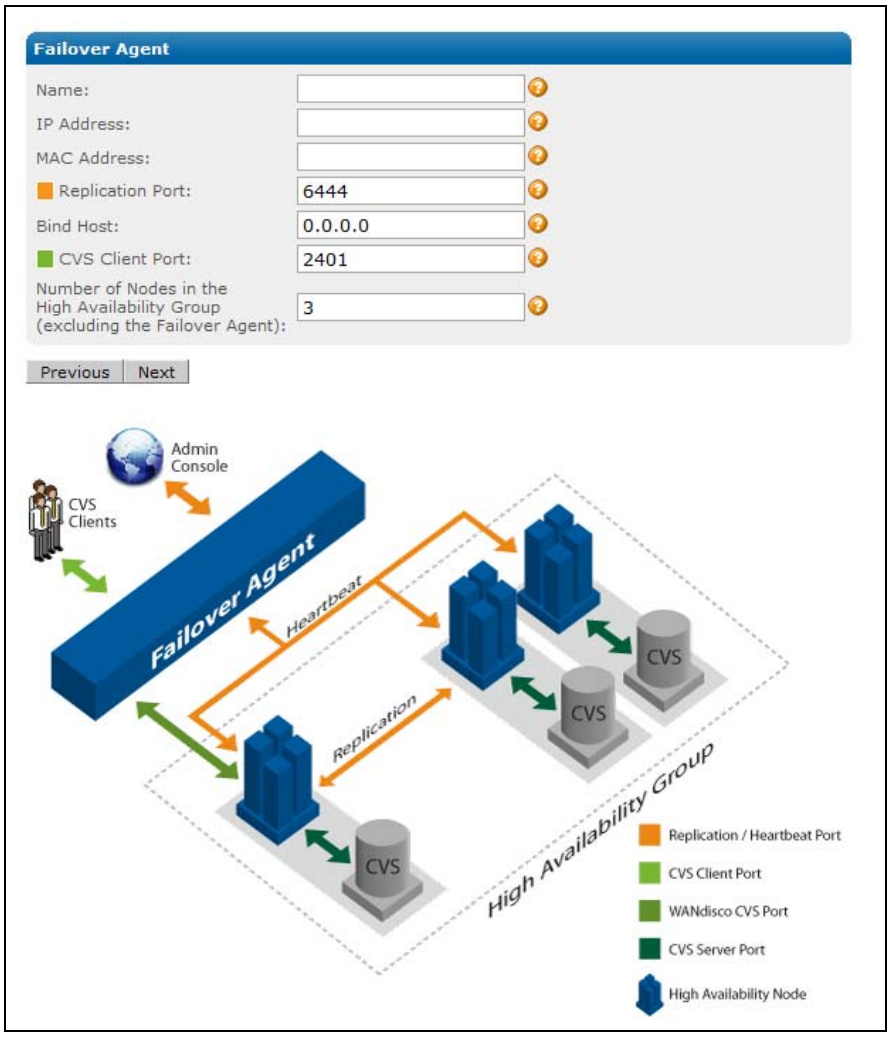
Step 9 Select the CVS type.

Step 10 The installer can copy your CVS repository and distribute it during the installation process, or an administrator can manually synchronize the repository to all the sites.

Step 11 Select a packaging directory.

Step 12 Click **Next**.

3.1.2 Configuring the Failover Agent



Step 13 In the Name filed, specify a name for the Failover Agent. The filed accepts alphanumeric characters and spaces are allowed.

Step 14 Enter the IP and MAC address. To find these addresses, go to that server's command prompt. For Unix, type

```
ifconfig
```

For Windows, type

```
ipconfig /all
```

Make sure the IP address is one you know is valid. The MAC address is the physical address.

For example, on a Windows machine, the output would look like:

```
Physical Address. . . . . . . . . : 00-1A-A0-36-53-3C
Dhcp Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IP Address. . . . . . . . . . . . : 192.168.1.124
Subnet Mask . . . . . . . . . . . : 255.224.255.0
Default Gateway . . . . . . . . . : 192.124.1.1
DHCP Server . . . . . . . . . . . : 192.122.1.1
DNS Servers . . . . . . . . . . . : 192.128.1.50
                                   64.405.172.26
```

Step 15 Replication Port is the port that WANdisco uses for its communications. (Protocols are replication, HTTP and heartbeat.)

Step 16 The Bind Host is the IP address/host name that WANdisco listens on for incoming connections. Unless there are multiple network cards on the Failover Agent, this is typically set to 0.0.0.0.

Step 17 The CVS Port is for CVS client communication.

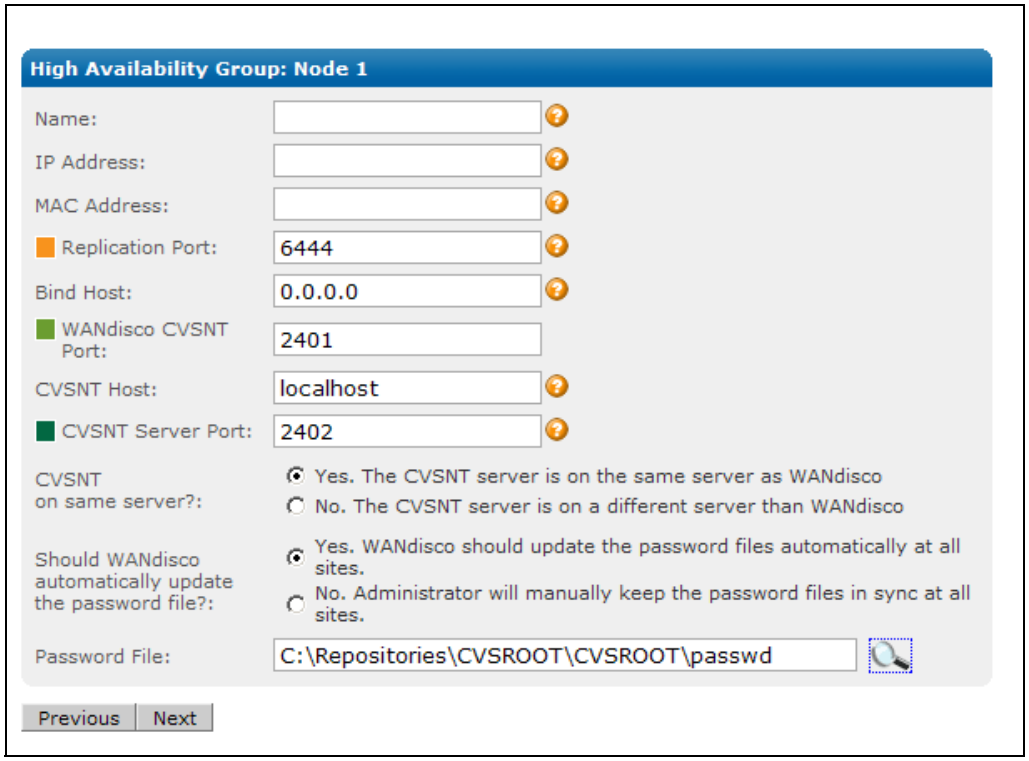
Step 18 Specify the number of nodes in the High Availability group, excluding the Failover Agent.

NOTE:

WANdisco recommends using more than two nodes in the HA group. A two-node group is vulnerable to a single point of failure if one node goes down.

If you did enter 2 in step 18 for a two node replication group, then proceed to section 3.1.12, First Time Installation for Two-Node Group.

3.1.3 Configuring the First Node



NOTE:

The priority order for failover is based on the order you specify here. For a full discussion on priority order, see section 1.3, Failover Sequence.

Step 19 Fill in the fields.

Step 20 For CVS Port, WANdisco recommends using a different value than the Failover Agent's CVS port number.

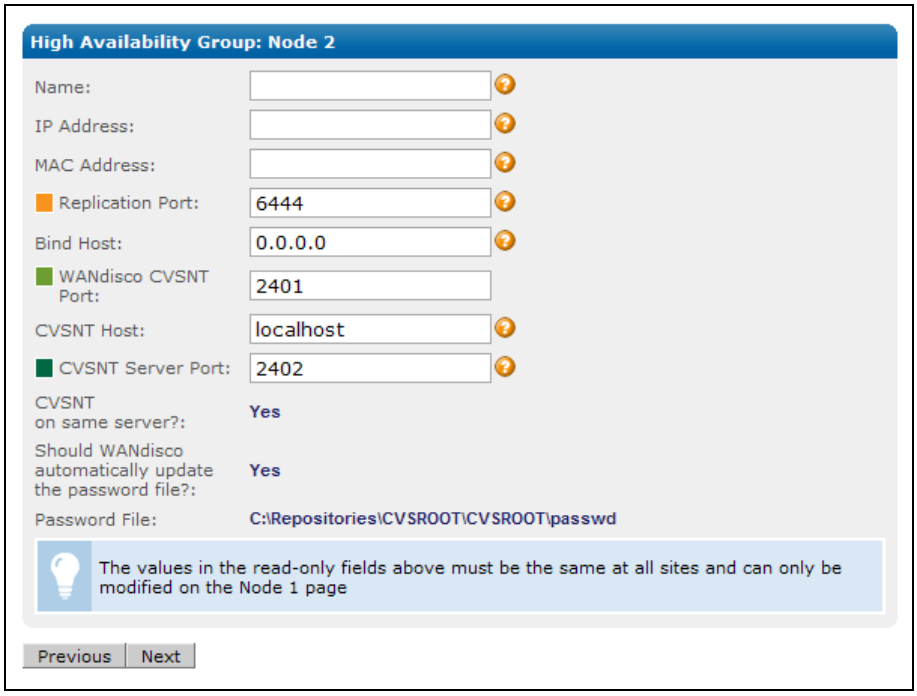
Step 21 For the CVS server port, WANdisco recommends you use a different port than the CVS port.

Step 22 WANdisco needs to know if it is sharing a server with CVS or not: answer yes or no to the *CVS on same server?* question.

Step 23 WANdisco needs to know if it is updating the CVS password file, or whether you (or a CVS administrator) is going to synchronize the password files: answer yes or no to the *Should WANdisco automatically update the password file?* question.

Step 24 If you answered yes to the previous question, enter the CVS password file path. Use the magnifying glass to browse.

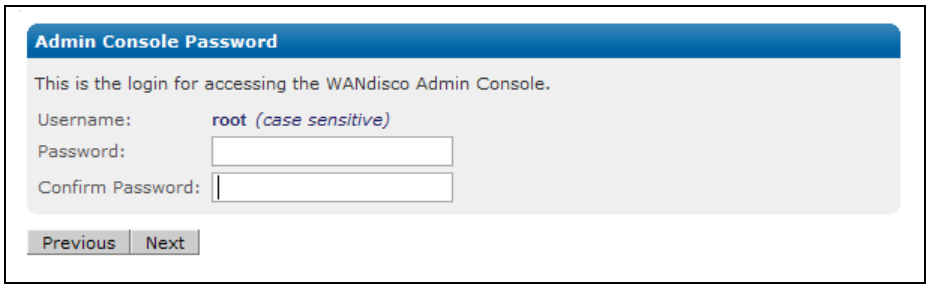
3.1.4 Configuring Subsequent Nodes



Step 25 Fill in the values. WANdisco recommends you use the same port numbers as you did for the previous node, as it can eliminate confusion.

Step 26 Since all the nodes must be configured identically, the install program uses the response from previous pages for the questions *CVS on same server?* and *Should WANdisco automatically update the password file?*

Step 27 Fill out a page for each node.

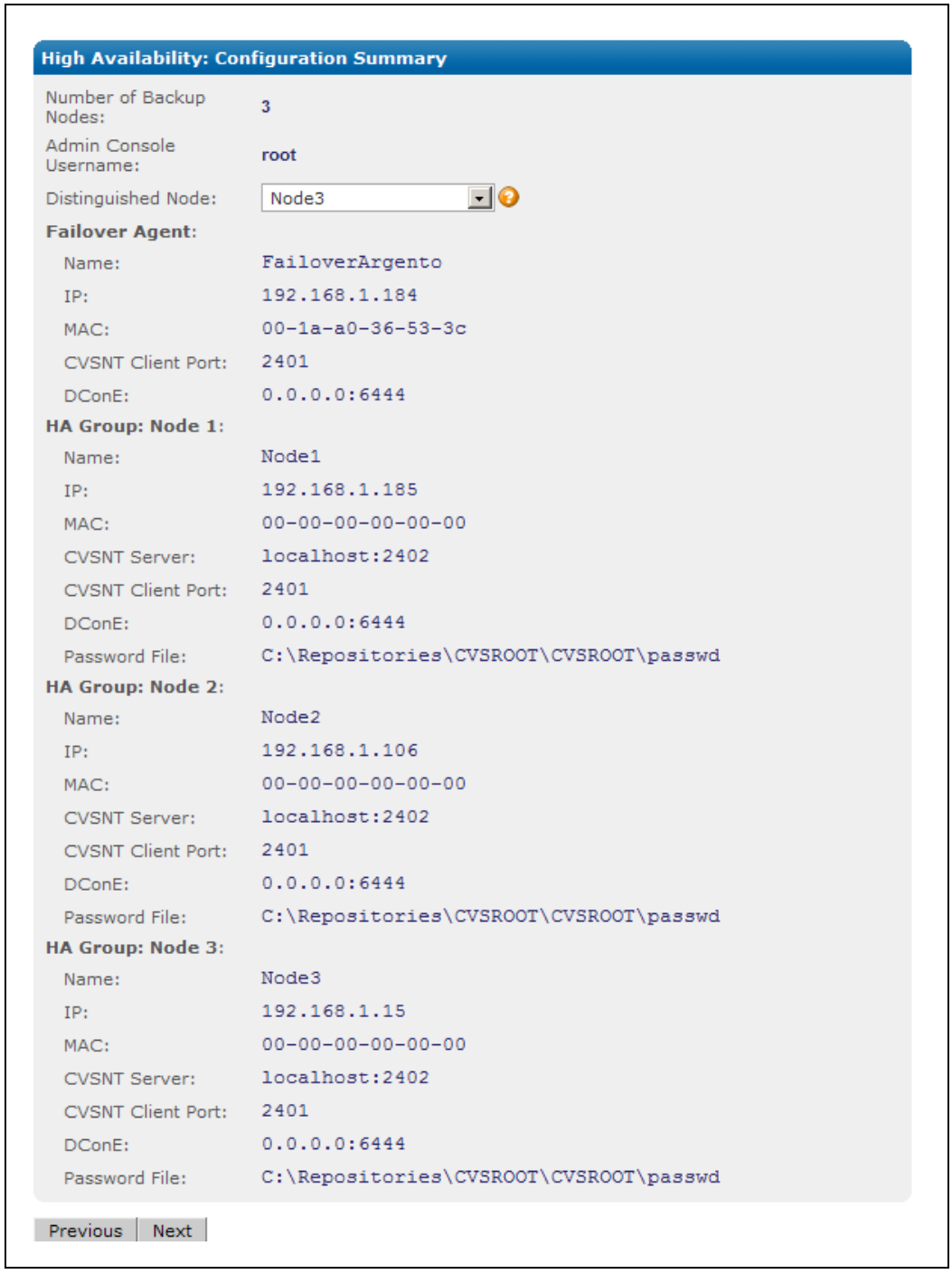


Step 28 Enter and confirm a password for WANdisco's username.

Step 29 Click **Next**.

3.1.5 Looking Over the Summary Page

After you have defined each node in your cluster, you are presented with the Summary page. You may want to print this page for reference.



3.1.5.1 Establishing a Distinguished Node

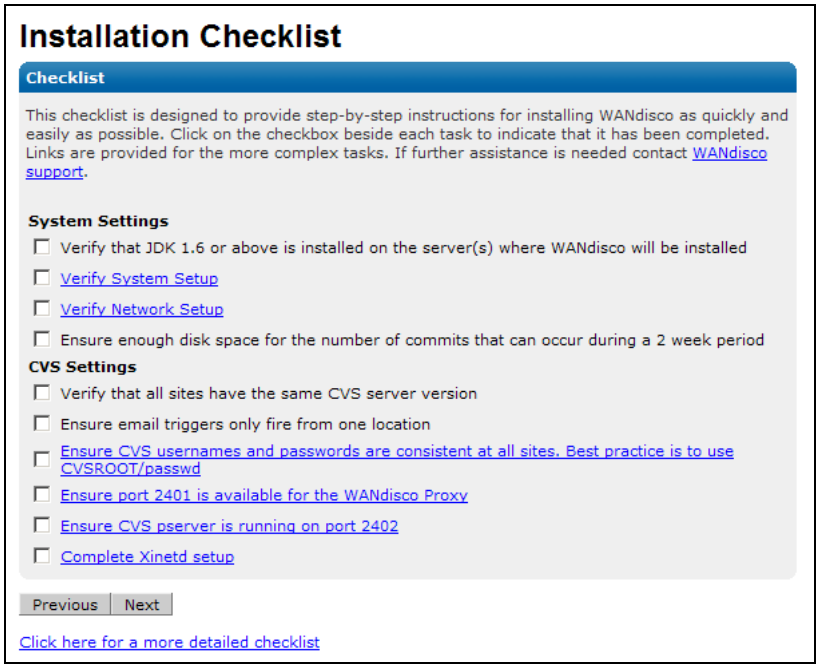
In the **Distinguished Node** field, select the node to be the HA group's distinguished node. The pull-down list includes all of the group's nodes. WANdisco recommends you choose the most reliable server in the group as the distinguished node.

NOTE:

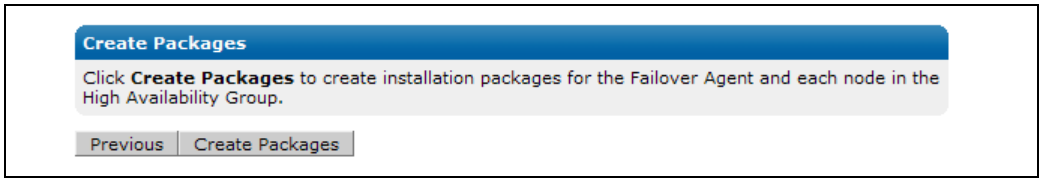
> If you have a two-node group, the backup node is always the distinguished node.

WANdisco recommends you print the Summary page for handy reference.

Step 30 After verifying all settings are appropriate, check each checkbox.



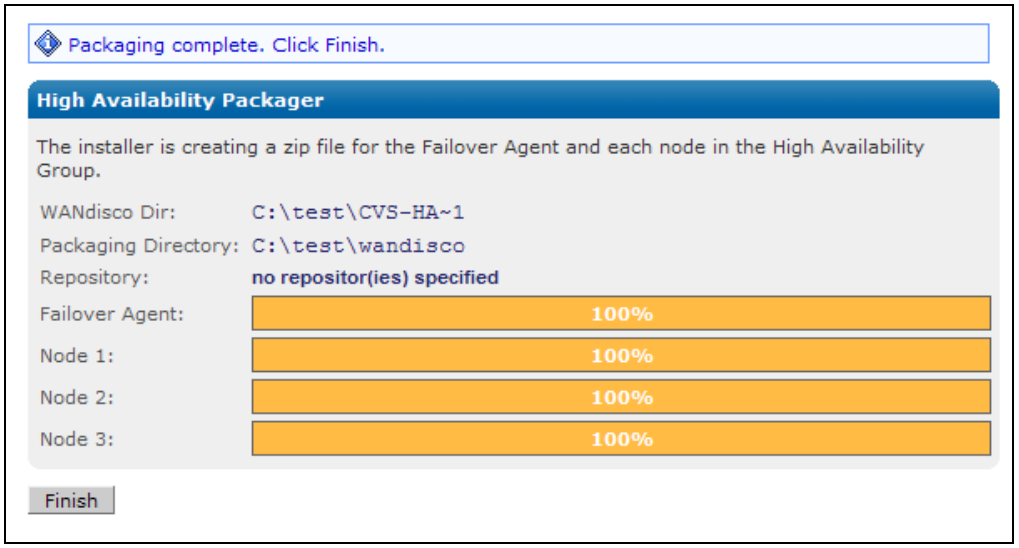
Step 31 Click **Next**.



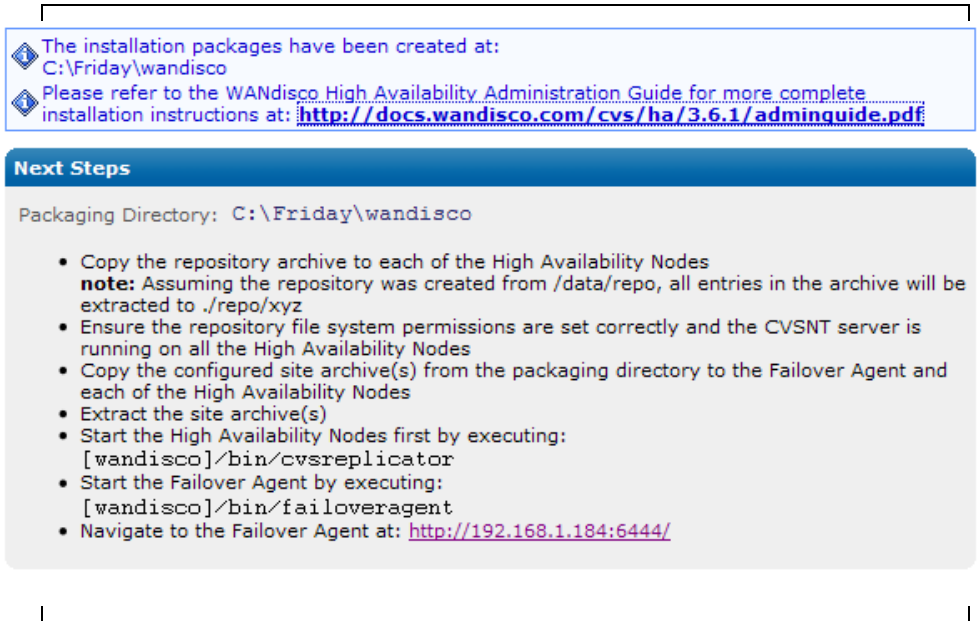
Step 32 Click **Create Packages**. The server restarts.

3.1.6 Package Creation

The installer creates a package for the Failover Agent and each node in the group.



Step 33 Click **Finish**.



The install program places zip files for the Failover Agent and the nodes in the `wandisco` directory.

Step 34 You can follow the instructions on the installation page, or continue with this procedure. However, make sure you start the Failover Agent last.

3.1.7 Installing the Nodes and the Failover Agent

Step 35 Copy the *<node name>.zip* files to a new directory their respective sites.

Step 36 Unzip the files for each node and for the Failover Agent. This creates a directory, `cvs-ha`, where all the uncompressed files are.

Step 37 For Unix, go to the *<node name>/bin* directory, and type

```
chmod +x *
```

Each node and the Failover Agent contains these High Availability directories:

DIRECTORY	CONTENTS
bin	Contains scripts like `failoveragent`, `shutdown`
config	Contains the [replicator]/config/prefs.xml file used to configure HA.
lib	Contains the `jar` files and DLLs that are required to run the product.
docs	Contains the administration guide in PDF format.
logs	Contains the pid file, log files and other temporary files. WANdisco HA's log file is named `FailoverAgent?prefs.log.0` and `ProxyServer-prefs.log`.
systemdb	Contains the system database with its transaction journal. **Warning**: Deleting or modifying files from systemdb will likely corrupt your installation.

3.1.8 Handling CVS Data

Step 38 If you had WANdisco include a copy of the repository in the installation package, it is extracted to a folder with the same name as the original repository.

If you are manually synchronizing the CVS repositories, see 5.2, Establishing a Baseline for Replication.

3.1.9 Validating the Installation

It is a good idea to take the time to validate the installation. That includes ensuring:

- all the databases are identical
- all database user privileges are identical at all nodes

3.1.10 Starting the Nodes and the Failover Agent

Step 39 Start HA at each node. At each node, go to `/cvs-ha/bin` and type

```
cvsreplicator
```

Step 40 Start the Failover Agent. At the Failover Agent, go to `cvs-failover/bin`, type

```
failoveragent
```

High Availability is now running and ready for use.

3.1.11 Post Installation Configuration

There are a few issues you may want to address pretty quickly after installation. They are described here, but any action you take is through the Admin Console. It is a simple interface, and is described in detail in 4, Using the Admin Console.

3.1.11.1 Tuning the Heartbeat Frequency

The WANdisco Failover Agent by default uses a heartbeat frequency of one heartbeat every second. This is appropriate for a LAN deployment.

If you are deploying High Availability over a WAN, then you may want to change the heartbeat frequency based on WAN latencies. The heartbeat interval should be two to three times the expected WAN latency between the Failover Agent and the replicator site. This ensures that the Failover Agent can distinguish between missing heartbeats and a slow network link.

You can also adjust the number of missing heartbeats that dictates when the WANdisco Failover Agent marks a replicator node as unresponsive, triggering failover. The default is four.

For a complete discussion of failover and the heartbeat, see section 1.2, Failover and the Heartbeat. To change the default values for the heartbeat, see the heartbeat definition in Chapter 4, Using the Admin Console.

3.1.11.2 Configuring Node Start Up Commands

The WANdisco Failover Agent is capable of starting the WANdisco replicator nodes from the Admin Console directly, provided you enter the startup commands in the Admin Console. The WANdisco Failover Agent also can use `ssh` to launch WANdisco replicators on remote machines.

To configure start up commands, see the Start Command definition in Chapter 4, Using the Admin Console.

3.1.11.3 Email Alerts for Failover Events

The WANdisco Failover Agent can generate email alerts whenever it detects an event related to failover. Examples of such events are:

- unable to transition to unilateral mode
- when transitioned to unilateral mode
- when unilateral mode starts
- when failed to backup
- when the primary is not available when the failover agent starts

To set up email alerts, see the Failover Notification definition in Chapter 4, Using the Admin Console.

NOTE:

Email notifications are not supported on the Windows platform.

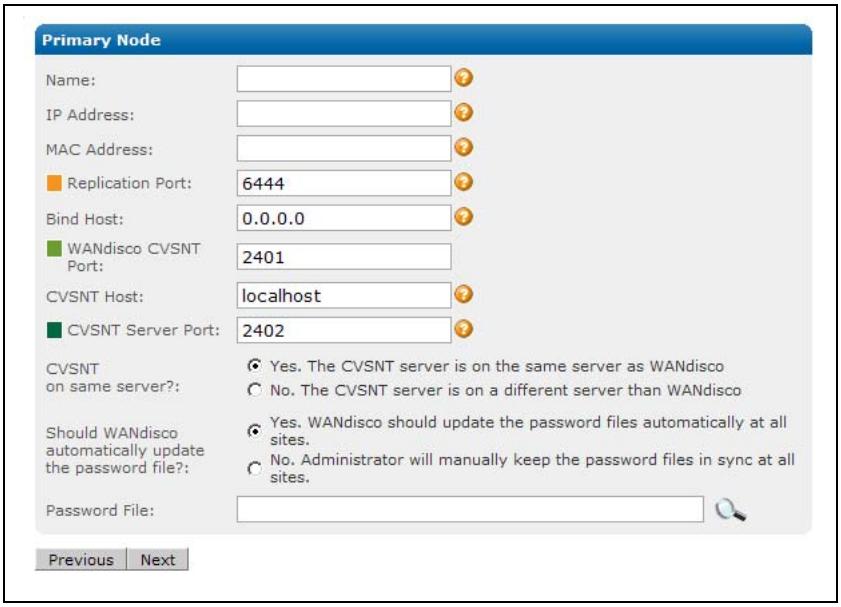
3.1.12 First Time Installation for Two-Node Group

This section is for users who have selected a two-node group on the Failover Agent page.

Although WANdisco does not recommend it, you can choose a two node group. A two node High Availability group forces changes to the failover logic as well as the quorum. For a two node group, the default quorum is singleton response, and the backup node is the distinguished node.

There is a separate section for procedures for two-node groups. See Chapter 6, Procedures for Two-Node HA Groups.

3.1.12.1 Configuring the Primary Node



NOTE:

The priority order for failover for a two-node group is always the same: the primary node is priority 1, the backup node is priority 2.

Step 41 Fill in the Name, IP and MAC address values.

Step 42 For CVS Port, use a different value than the Failover Agent's CVS port number.

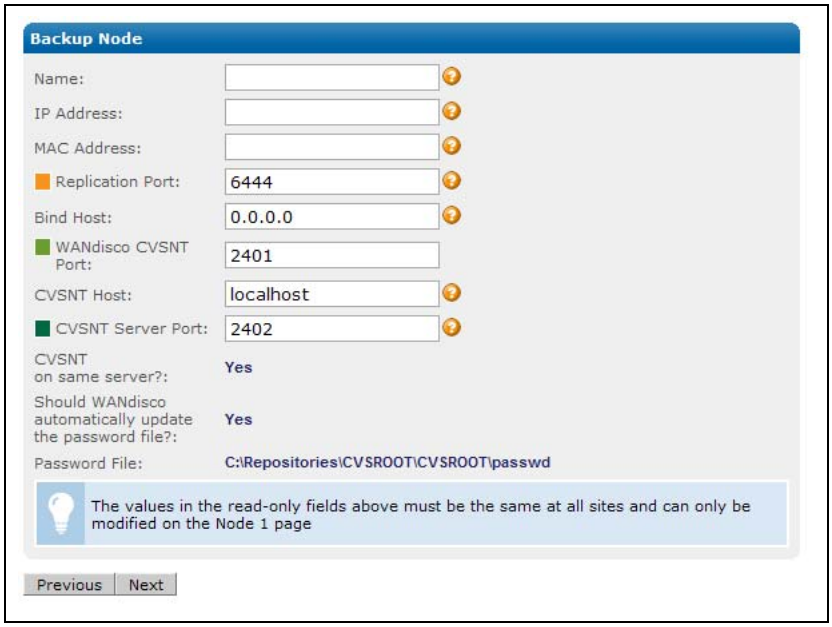
Step 43 For the CVS server port, WANdisco recommends you use a different port than the CVS port.

Step 44 WANdisco needs to know if it is sharing a server with CVS or not: answer yes or no to the *CVS on same server?* question.

Step 45 WANdisco needs to know if it is updating the CVS password file, or whether you (or a CVS administrator) is going to synchronize the password files: answer yes or no to the *Should WANdisco automatically update the password file?* question.

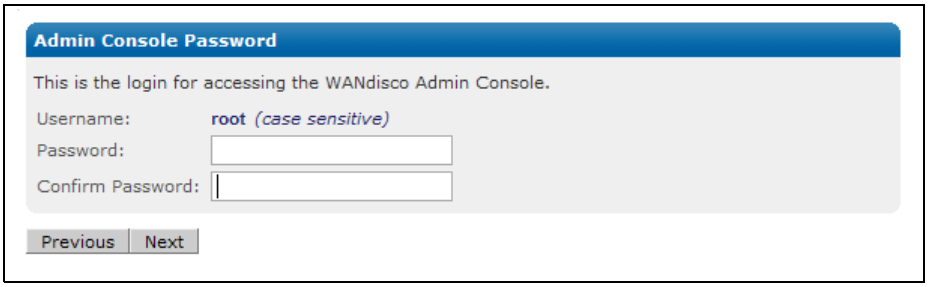
Step 46 If you answered yes to the previous question, enter the CVS password file path. Use the magnifying glass to browse.

3.1.12.2 Configuring the Backup Node



Step 47 Fill in the values. WANdisco recommends you use the same port numbers as you did for the previous node, as it can eliminate confusion.

Step 48 Since all the nodes must be configured identically, the install program uses the response from previous pages for the questions *CVS on same server?* and *Should WANdisco automatically update the password file?*



Step 49 Enter and confirm a password for WANdisco's username.

Step 50 Click **Next**.

Now you can complete the installation by proceeding with section 3.1.5, Looking Over the Summary Page.

3.2 Installing Upgrades

NOTES:

This procedure involves taking CVS offline. Please follow your company procedures about notifying CVS users of down time.

3.2.1 Disconnect CVS Users

Step 1 After notifying CVS users of the downtime, navigate to the Dashboard.

Step 2 Watch for **Total Transactions Pending** to reach **0**. The **Total Transactions Pending** must be 0. If you do not wait for the **0**, the databases will be out of synch.

Step 3 When the count reaches 0, shut down the Failover Agent.

Step 4 Stop all the HA nodes.

3.2.2 Back Up or Export CVS Data

Step 5 Backup or export the CVS repository.

3.2.3 Preparing the Install Node

Step 6 On the original install node, zip these directories:

```
config
```

Step 7 Delete this directory:

```
systemdb
```

Step 8 In the `cvs-ha-installer/config` directory, delete these directories:

```
membership
security
passwd
```

3.2.4 Preparing Subsequent Nodes

Step 9 On each node, zip the `cvs-ha` or `cvs-failover` directory.

Step 10 On each node, delete the `cvs-ha` or `cvs-failover` directory.

3.2.5 WANdisco-supplied .jar File

Step 11 Save the cvsha.tar.gz file.

Step 12 Verify the md5 checksum. For Unix, type

```
md5sum
```

Step 13 Copy the jar file to the `cvs-ha-installer/lib` directory.

Step 14 Unzip or untar the file.

3.2.6 Running the Install Program

Step 15 Run this command on the original install site. The install program automatically populates all previous configuration information.

```
cvs-ha-installer/bin/setup
```

The Setup page appears.

Step 16 Enter the number of nodes in the replication group.

Step 17 Click **Next** to continue.

Step 18 Make changes as needed.

Step 19 Verify the configuration on the Summary page. Click **Next**.

Step 20 Wait while the wizard creates packages for all the nodes. Click **Next**.

Step 21 Complete packaging by clicking **Finish**.

3.2.7 Installing and Running the HA Group

Step 22 Copy the `<node name>.zip` to each site.

Step 23 For all nodes, unzip the `<node name>.zip` file.

3.2.8 Validating the Installation

It is a good idea to take the time to validate the installation. That includes ensuring:

- all the databases are identical
- all database user privileges are identical at all nodes

3.2.9 Installing and Running the Failover Agent

Step 24 Copy the `cvs-failover.zip` to the Failover Agent site.

Step 25 Unzip the `cvs-failover.zip` file.

Step 26 Start the Failover Agent. In the `/bin` directory, type

```
failoveragent
```

The new installation is complete.

4 Using the Admin Console

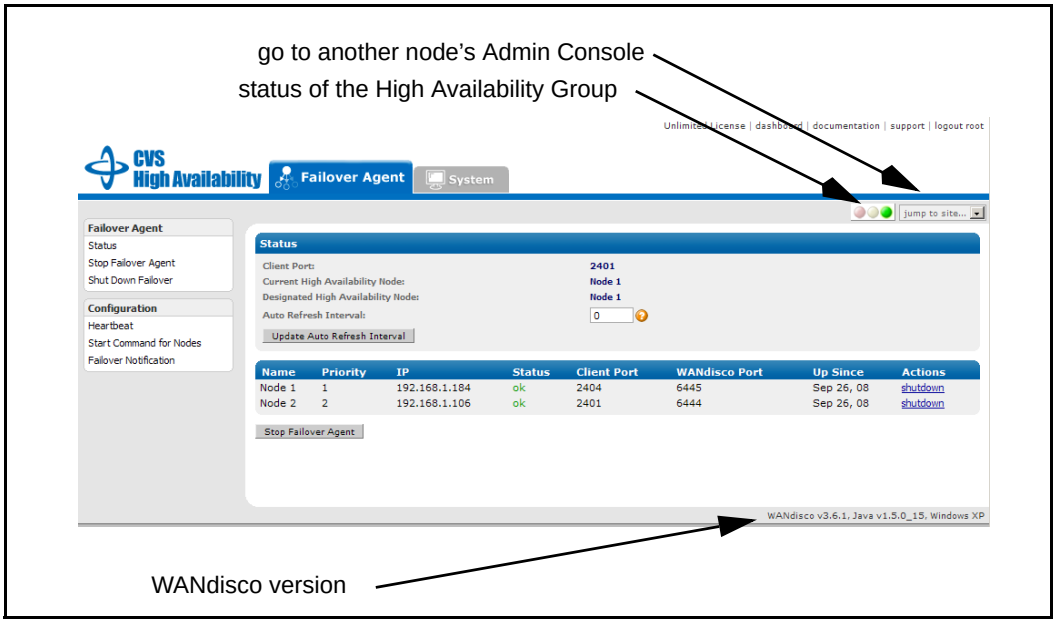
The Admin Console is a simple interface that allows you to monitor and perform simple tasks for your HA group and the Failover Agent. The Admin Console for the Failover Agent shows the other HA nodes onscreen, so you may want to have the Failover Agent's Admin Console running.

4.1 Starting the Admin Console

To start the Admin Console for the Failover Agent or any node, in a browser's address bar, type

http://<IP address>:<replication port number>

The Admin Console's Home page appears.



4.2 Failover Agent Pages

The Failover Agent's Admin Console has three pages: The Failover Agent page, the System page, and the Dashboard.

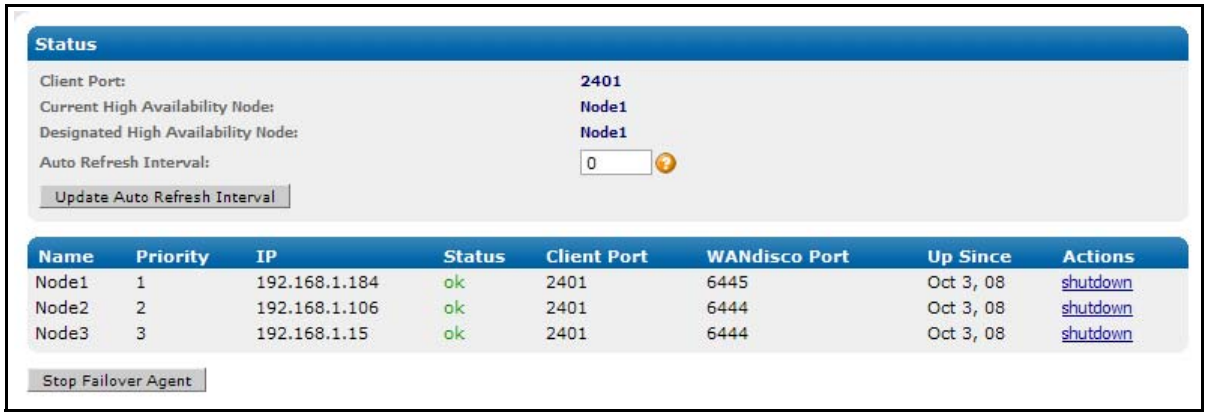
4.2.1 Failover Agent Page

The Failover Agent page displays the status of the other nodes in the HA group, gives information on the current node and designated node, and offers several commands in the left side menu.

The status of the High Availability group displays in the upper right. One of three circles pulses, giving the current status: green pulsing signifies all nodes are up, yellow pulsing signifies at least one node is not responding, and red pulsing signifies that all nodes are not responding.

The Status section names the Current Node (the node listening to the CVS client through the Failover Agent) and the Designated Node (priority level 1 in the prefs.xml file). These are the same node unless failover has occurred.

The Auto Refresh Interval offers you the ability to automatically update the page. The default is 0.



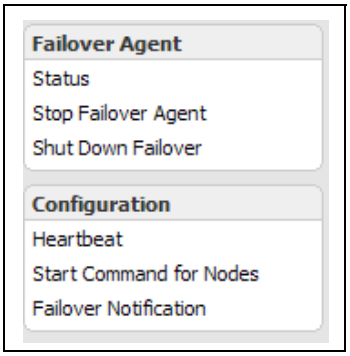
Each node is listed by name, priority order, IP address, current status (**ok** and **not responding**), client port (in use only by the current node), replication port, date of last start up, and action (**shutdown** or **start**).

NOTE:

If you shut down the current node, you trigger a failover. Shutting down any other node does not result in a failover.

A High Availability group of three or more nodes has a majority response quorum. If you shut down a majority of nodes in your HA group, replication stops, and can only continue when a majority of the HA nodes are running.

4.2.1.1 Left Side Menu



High Availability

Status	This command displays the HA group's status in the page's main area.
Stop Failover Agent	This stops CVS access to CVS clients. You must confirm your action.
Shut Down Failover	This shuts down the Failover Agent.

Config

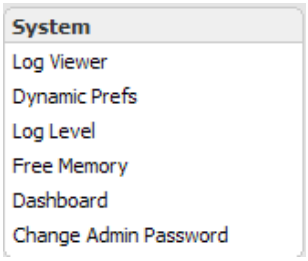
Heartbeat	For a complete discussion on the heartbeat, see 1.2, Failover and the Heartbeat.
Missing Heartbeat Count	The number of missing heartbeat responses the FA gets before marking a node as unavailable.
Interval	The time between queries from the HA to the nodes. You can specify a different number for each node.
Connection Timeout	The time to wait before assuming the query has failed. You can specify a different number for each node.
Save All	Save changes for all nodes.
Start Command	Enter in commands to start the nodes.
SSH Command	Enter syntax for the SSH command.
Start Command	Enter syntax to start the node. For Windows, in the Start Command field, enter `perl -x -S <pathname>\cvs-ha\bin\cvs-replicator`
Save All	Save changes for all nodes.
Failover Notification	If specified here, WANdisco sends emails when: a) the designated node dies, triggering failover; b) priority 1 node comes back online; c) in a two node group, when the second node dies. Email currently not supported in Windows.
Admin Email	Enter in the address to receive Failover notification emails.

.

4.2.2 The System Page

The System page offers several commands and utilities for the High Availability group.

4.2.2.1 Left Side Menu

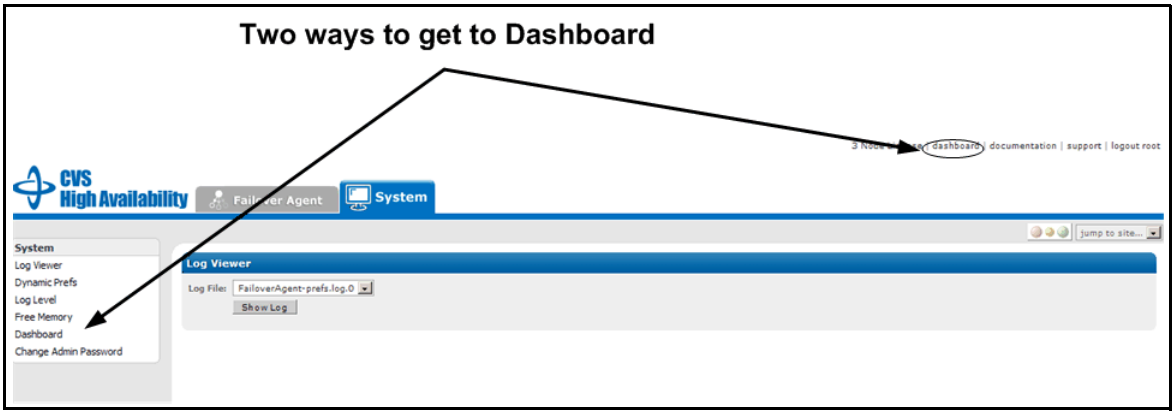


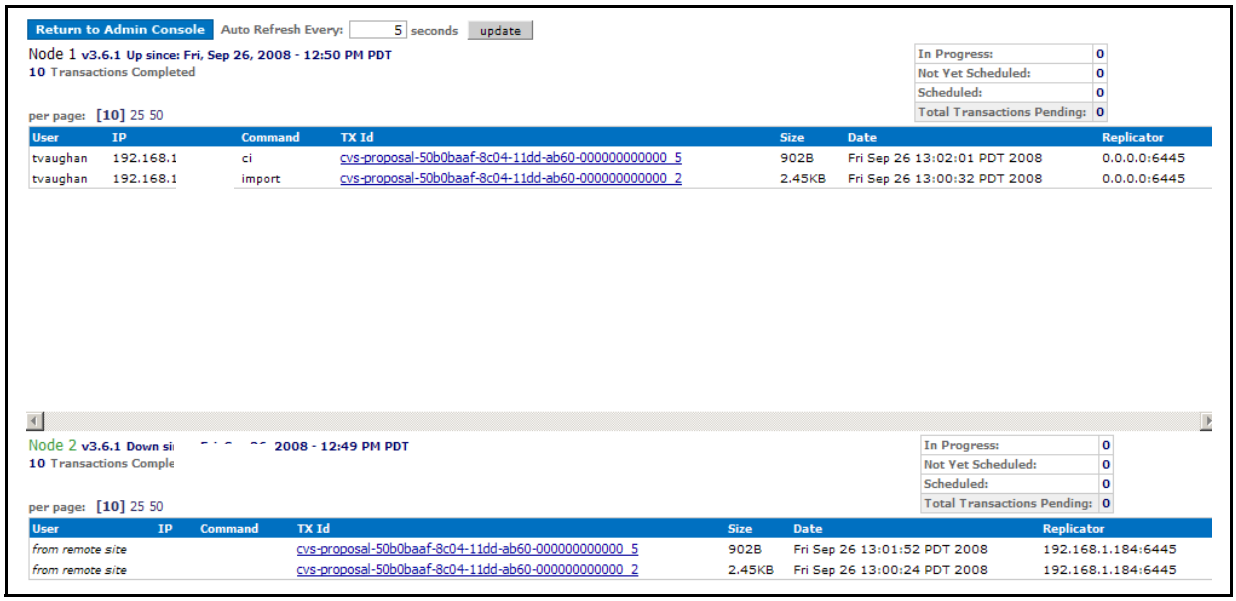
System

Log Viewer	
FailoverAgent-prefs.log	WANdisco's main log file for the Failover Agent.
ProxyServer-prefs.log	The installation log file.
Show Log	Click this to display the log in the Dashboard.
Dynamic Prefs	These fields are used internally and for customer support.
Log Level	WANdisco uses one log, and the default level is **info**. The levels vary from **severe**, where you get only the most severe warnings, to **finest**, which logs every action.
Free Memory	This command frees the memory (GB stands for garbage collection) for the current node. The command occurs when you click on this menu selection. The display shows information on the command that was just performed.
max mem used by JVM	the maximum memory that JVM can use on the current node
free memory before GC	the amount of free memory before you ran this command
free memory after GC	the amount of free memory after you ran this command
memory freed	the total amount of memory freed at the command's completion
Dashboard	offers another way to get to the Dashboard. The Dashboard is discussed in detail in section 4.2.3, The Dashboard.
Change Admin Password	You can change the WANdisco admin password here.

4.2.3 The Dashboard

There are two ways to get to the Dashboard from the Failover Agent pages. The Dashboard shows each node's transactions. As soon as High Availability receives a CVS transaction request, that transaction joins the replication group's queue. There is one queue for the replication group.) High Availability keeps track of which node it emanated from, but otherwise, the transaction joins the queue for the replication group





4.2.3.1 Pending Transactions

There are three statuses of replicated transactions before they are committed to CVS.

- **Not Yet Scheduled** - these transactions are in the queue
- **Scheduled** - these transactions are approved by the quorum and are waiting to be executed
- **In Progress** - these transactions are in the process of being completed

The **Total Transactions Pending** lists the total number of transactions in all statuses.

4.2.3.2 Completed Transactions

Once a transaction is completed, the Dashboard lists it in the transaction list, and that transaction is no longer considered in the **In Progress** count. The completed transaction joins the count in the **Transactions Completed** display. This display keeps count of transactions since replication began.

4.2.3.3 Refreshing the Dashboard Display

The Dashboard by default does not refresh. As High Availability is completing many transactions, a display that refreshes often would be very confusing to read. While the order of completed transactions is the same at all nodes, the real time of when a transaction is posted may vary from node to node. To refresh the Dashboard, click **Update**.

You can also set the Dashboard to refresh automatically by entering a number in seconds in the field.

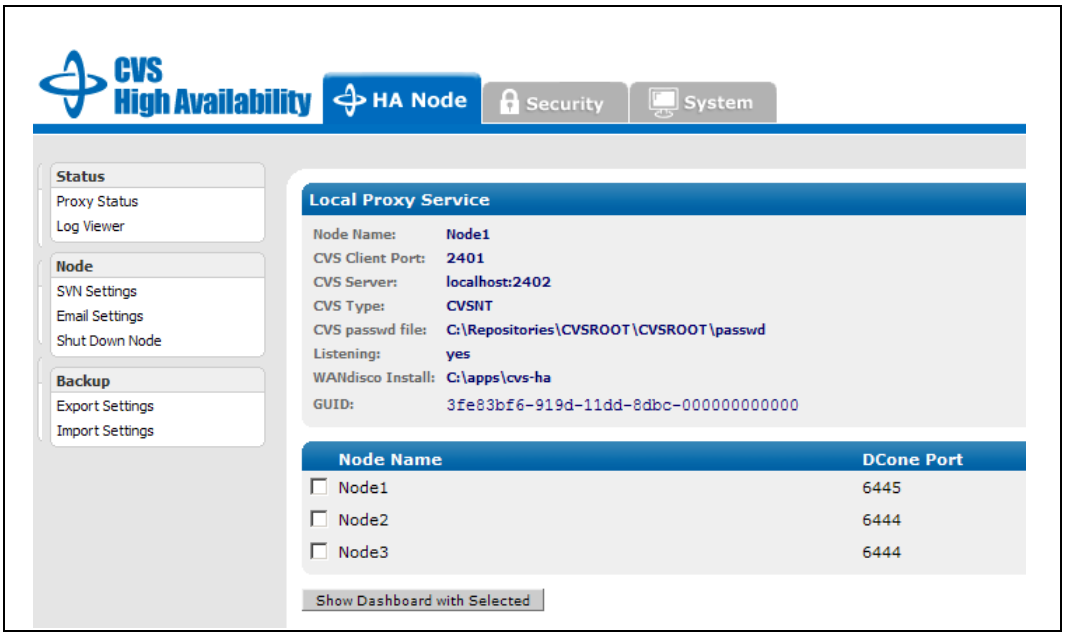
NOTE:

The value you enter in the Auto Refresh box is replicated throughout the HA Group and the Failover Agent. Whenever you change this value, at any node, that value becomes the Auto Refresh value for the HA group.

4.3 HA Group Pages

4.3.1 The Node Page

The HA Node tab offers the status of this node.



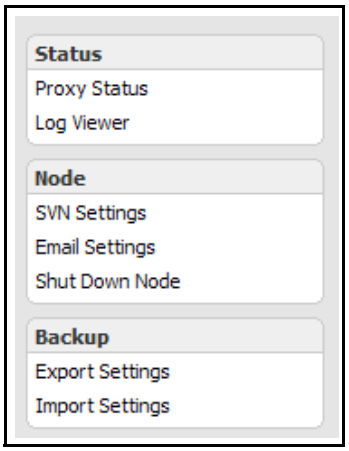
Local Proxy Service

Node Name	This node's name.
CVS Client Port	Set during installation. This node receives CVS client communications through the Failover Agent on this port.
CVS Server	This identifies which CVS server this node communicates with, and which port number.
CVS Type	The type of CVS specified upon WANdisco installation.
CVS passwd file	Specified upon WANdisco installation: either WANdisco manages the CVS passwd file or
Listening	This node is listening on the CVS Client Port. Only the current node is listening; all other nodes are not listening to the CVS client.
WANdisco install	The directory where WANdisco is installed.
GUID	This node's GUID (Globally Unique IDentifier). Also listed in prefs.xml file.

Node Name The HA group nodes are listed.

Show Dashboard with Selected
This is another quick way to see the Dashboard. Use the checkboxes to include HA nodes on the Dashboard display.

4.3.1.1 Left Side Menu



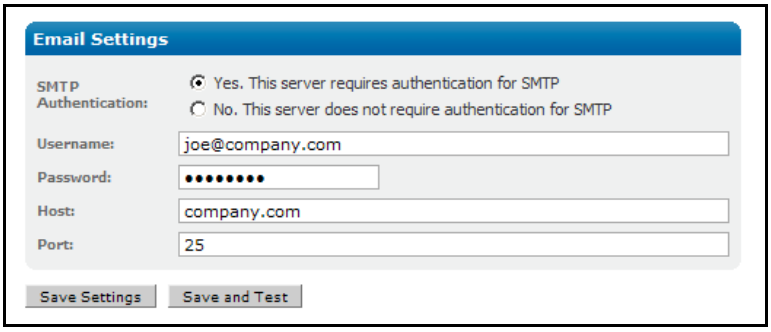
Status

Proxy Status	Described in the Local Proxy Service section.
Log Viewer	Select from two logs to view on the Dashboard.

Proxy

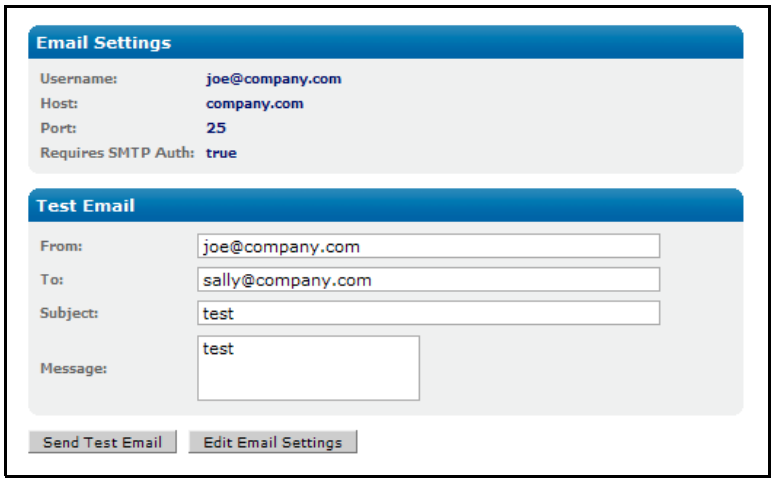
CVS Settings

CVS Executable	Identify the path to the CVS executables.
Username	Login credentials for CVS. For remote password management for Unix.
Password	Login credentials for CVS. For remote password management for Unix.
CVSROOT (s)	Credentials for CVS. For remote password management for Unix.
Default User	For remote password management for Unix.
Temp Directory	For remote password management for Unix.



Email Settings	Set the email settings for the watchdog contact. You have to set the email in watchdog also. See 5.24, About Watchdog Mode.
SMTP Authentication	Select whether you need SMTP Authorization.
Username	For authentication, enter a valid username.
Password	For authentication: enter a valid password.

Host	Enter the email host.
Port	Enter the emal port number.
Save Settings	Save any changes.
Save and Test	Save any changes, and run an email test. Selecting this displays the test page.



Test Email

From	Enter the sender address.
To	Enter in another address to send a test email.
Subject	Name a subject.
Message	Enter test text.
Send Test Email	Click to send the test email.
Edit Email Settings	Click to edit the email settings.

Shut Down Node	This command shuts down this node.

NOTE:

If you shut down the current node, you trigger a failover. Shutting down any other node does not result in a failover.

A High Availability group of three or more nodes has a majority response quorum. If you shut down a majority of nodes in your HA group, replication stops, and can only continue when a majority of the HA nodes are running.

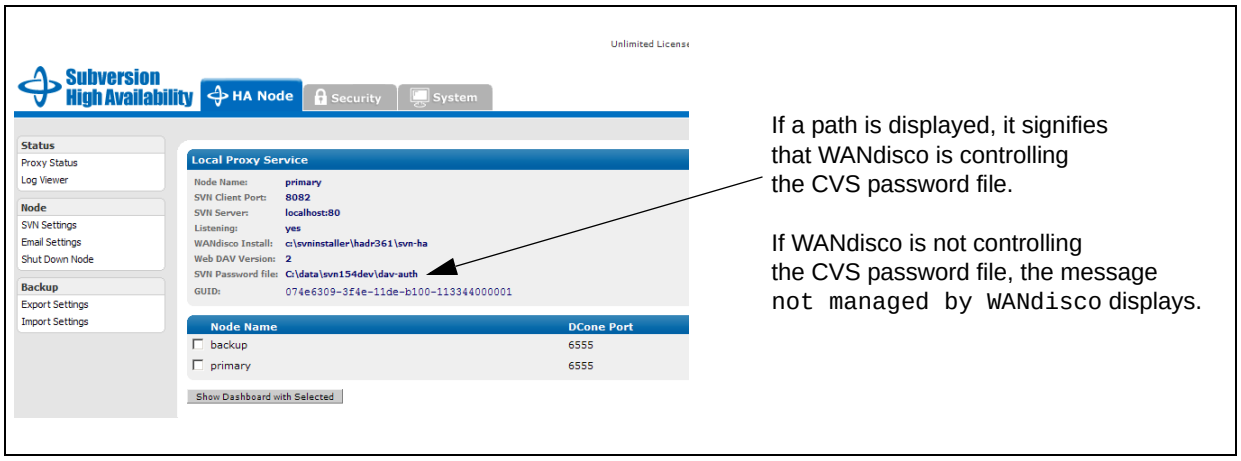
Backup

Export Settings	Export WANdisco settings with this command.
Import Settings	Import WANdisco settings with this command.

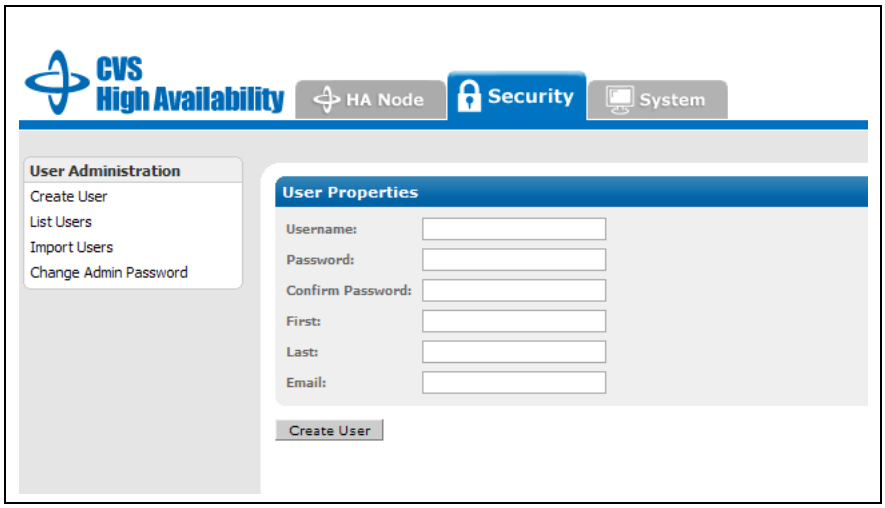
4.3.2 The Security Page

The Security tab offers password control for users across the HA group.

Most users, during installation, elect to have WANdisco control the CVS password files. You can verify this on the HA Node page. If the field **CVS password file** lists a path, WANdisco is controlling the CVS password file. If the message `not managed by WANdisco` displays, you, the WANdisco administrator, must control and synchronize the CVS password files at each node.



What Happens When WANdisco Controls the CVS Password File	What Happens When WANdisco Does Not Control CVS Password File
Any changes entered in the Security tab are replicated throughout the HA group.	Any changes entered in the Security tab are replicated throughout the HA group.
Changes are immediate and are reflected in the CVS password file, at each node.	The WANdisco administrator must update each node's CVS password file to reflect any changes in WANdisco.



User Administration

Create User	Create CVS users here.
Username	Enter the username.
First	Enter the user's first name.
Last	Enter the user's last name.
Email	Enter the user's email address.
Create User	Click when you have entered all information.
List Users	This command lists all the current users. You can order the users by userid, first name, last name or email, by clicking on the title in the title bar.
Import Users	You can import a text file of users, of the format `username,lastname,firstname,email`. Do not use spaces.
Change Admin Password	Change the Admin password here.

4.3.3 The System Page

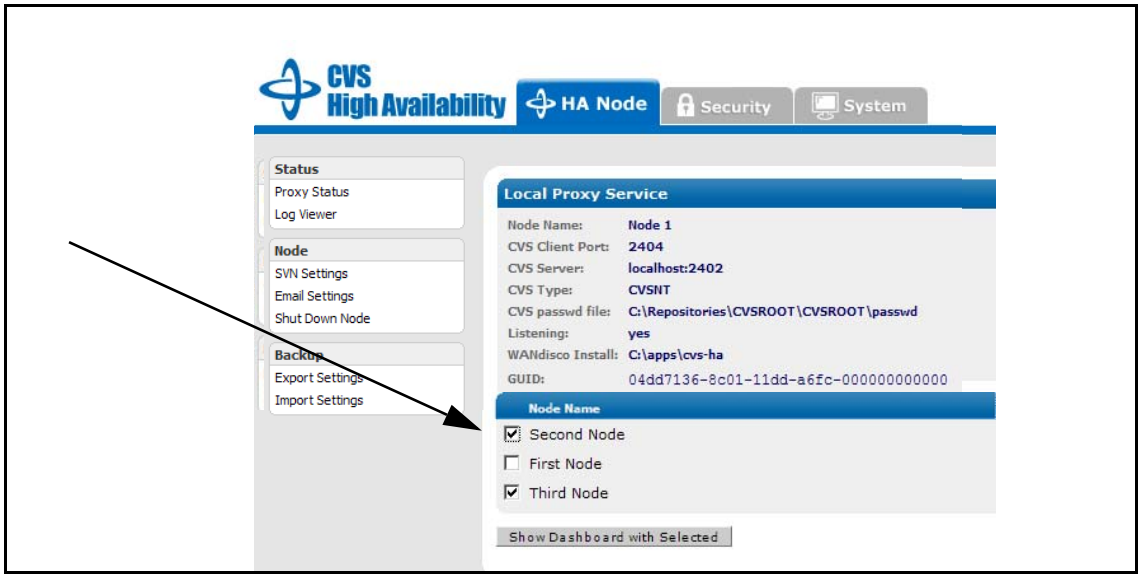
All but one of the commands and options on this page are described in 4.2.2, The System Page for the Failover Agent. There is one new command:

Transaction Status	You can check the status of a specific transaction number. You can select another node's transaction, in which case you are switched to that node's Admin Console.

4.3.4 The Dashboard

This page offers the same commands and options as the Failover System page. See 4.2.3, The Dashboard.

By default, the Dashboard displays all nodes. If you only want certain nodes displayed, check just those nodes and click **Show Dashboard with Selected**.



5 Procedures

5.1 Preventing CVS Users From Making Transactions

NOTE:

This procedure involves taking the High Availability offline. Please follow your company procedures about notifying CVS users of down time.

Step 1 After notifying your users of the downtime, navigate to the Dashboard.

Step 2 Watch for **Total Transactions Pending** to reach 0. The **Total Transactions Pending** must be 0. If you do not wait for the 0, you jeopardize the synchronization of the databases.

Step 3 Stop the Failover Agent. In the Admin Console, in the Failover Agent tab, click **Stop Failover Agent**.

Step 4 To start the Failover Agent, click **Start Failover Agent**.

5.2 Establishing a Baseline for Replication

Before starting WANdisco, you should ensure that all sites start with an identical copy of the repository (the cvsroots) - identical in all respects, except as noted below.

Depending on the size of your repository and available bandwidth to the remote sites, you can decide whether to copy or sync the repository over the network or ship a copy of the repository on a physical medium (for example, a CD, DVD or hard disk). Select the method that works best for your situation.

If you already have an older copy of the repository at the remote sites, for example, if, prior to deploying WANdisco, you were using a master-slave replication solution such as cvsup, choose the **Synchronize** procedure.

5.2.1 Copying the CVS Database

Otherwise, start by estimating how long it may take you to copy the repository over the network by determining the size of your repository and the bandwidth available to the remote sites. If you conclude that it takes too long, you will want to ship the repository to the remote sites on a physical medium.

Step 1 Determine the size of the repository. From a Unix command prompt, `cd` to your repository and type

```
du -s
```

This reports the size of your repository in kilobytes.

Step 2 Determine the network bandwidth. Copy a reasonable-sized file (say 100 megabytes) to the remote site using any means available (example, `scp` or `ftp`). Time the copy.

Step 3 Estimate how long the copy will take. Using the information gathered above, you can estimate how long it can take you to copy the repository to the remote sites over the network. For example, if copying a 100 megabyte file over the network took 10 minutes, copying your 5 gigabyte repository may take about 500 minutes (8 hours and 20 minutes).

5.2.2 Synchronizing with an older remote Copy

You can use `rsync` to sync up an older remote copy with your master copy. For example, from the machine with the master copy of myRepository, type

```
rsync -rvlHt /path/to/myRepository remoteHost:/path/to
```

Note that the final element, `myRepository`, is not specified in the remotehost's path. For further information, consult the `rsync` man pages.

5.2.3 Copying Over the Network

Use this procedure if:

- You do not have an older remote copy; i.e., you are copying the entire repository over.
- Your repository is small enough.
- You have enough network bandwidth to copy the repository to the remote sites in reasonable time.

Step 1 Ensure that the repository is not in use. If necessary, shut down the SCM server. For example, type

```
/etc/init.d/xinetd stop
```

Step 2 Package the master copy of your repository.

Step 3 Copy the package to the remote host.

Step 4 Log in to your remote host and unpackage the repository. For example, on the server with the master copy, type

```
cd /path/to
tar pzcf myRepository.tgz myRepository
ssh remoteHost mkdir -p /path/to
scp myRepository.tgz remoteHost:/path/to/
log into remoteHost
cd /path/to
tar pzxf myRepository.tgz
```

Step 5 Start the SCM server. For example, type

```
/etc/init.d/xinetd start
```

5.2.4 Shipping on a Physical Medium

If copying over the network may take too long, you can ship the repository to the remote destination on a physical medium, such as a CD, DVD or hard disk. Note that you do not have to wait for the baseline to be available at all sites before using WANdisco. Instead, you can follow the procedure below.

Using WANdisco before the baseline is available at all sites

Step 1 Deploy WANdisco as usual, but do not start the WANdisco server at the sites where the baseline is not yet available.

Step 2 When choosing a quorum, ensure that the sites where WANdisco can be started are sufficient to form a quorum. The simplest way to do this is to choose the Singleton Quorum policy, and choose the site that has the master copy of the repository as the distinguished node.

Safe Differences

The only things that can safely differ in the baselines across your sites are post-commit triggers. For example, if you generate email notifications from a post-commit trigger, it is a good idea to do that at only one site to avoid generating duplicate email notifications.

Common Pitfalls

It is important that the repositories are identical in all respects except as noted above. A common mistake when the desired baseline is an empty repository is to `init` a new empty repository at each site. Instead, you should `init` the repository at one site, and copy the empty repository to other sites.

For CVS, there may be no negative consequences to this mistake, but an empty repository is so small, you might as well play it safe.

5.3 Finding the Last Committed Transaction

Even though committed transactions are always in the same order for each node, the timing of the commits usually varies from node to node. So unless there are no CVS users logged in, you probably are going to have variations per node for committed transactions.

Go to any node's Dashboard. Type

```
http://<IP address>:<HA port number>/dashboard2
```

You see all the nodes on the Dashboard to compare the listed transactions.

5.4 Adding a Node to the HA Group

You can add a node to an existing HA group. Notify WANdisco of the IP address of the new node or nodes, and WANdisco sends you a new installation file and license key file.

Follow the instructions in 3.2, Installing Upgrades. The instructions are identical; you are just adding more nodes to the group.

5.5 Deleting a Node from the HA Group

You can delete a node or nodes from an existing HA group. Follow the instructions in 3.4, Installing Upgrades, but just eliminate the node from the group.

5.6 Changing a prefs.xml File

The prefs.xml files for nodes are located in `cvs-ha/config`, and for the Failover Agent in `cvs-failover agent/config`. Each file contains all preference information for the nodes in the group.

If you make changes that affect more than one node, you must change each node's specific file. But if your change affects just one node, you can change just that node's prefs.xml file.

5.7 Performing a Synchronized Stop

NOTE:

This procedure involves taking CVS offline. Please follow your company guidelines on notifying CVS users of downtime.

Step 1 Bring up the Admin Console for each site.

Step 2 Stop the Failover Agent. On the Failover Agent page, click on **Stop Failover**.

Step 3 Watch the count for Pending Transactions go to 0. View all the nodes on the Dashboard.

NOTE:

You must wait for the count to reach 0 for all nodes. If you shut down a node before the count is 0, the nodes are no longer in sync and must be re synchronized.

Step 4 When the Pending Transaction count is 0 for all nodes, stop all the nodes. Go to the Failover Agent page and click on **Shut Down Node** for each node.

Step 5 If desired, shut down the Failover Agent. Click **Shutdown Failover Agent**.

Step 6 (Optional) Manually verify that the repositories are in sync. Make sure that there are no lingering lock files in the CVS repository.

Lingering CVS locks can happen for a variety of reasons. This discussion is quite illuminative, and includes a script for fixing the problem: http://lists.gnu.org/archive/html/info-cvs/2005-06/msg00211.html.

5.8 Performing a Clean Restart After a Synchronized Stop

WARNING:

This procedure involves running the `reset` script. You must always reset all the nodes in the group.

Step 1 At each node's command line, run the `reset` script in `cvs-replicator/bin`. (For Windows, type `perl reset`.) The command resets the system database, and the transaction count resets to 0.

Step 2 Start up all the nodes. At each node's prompt, type

```
cvsreplicator
```

Step 3 Start up the Failover Agent. At the Failover Agent's prompt, type

```
failoveragent
```

5.9 Verifying That the Replicator is Working

There are two ways you can check. You can make a minor change in CVS on one client, wait a minute, and go to another client to ensure the change is reflected.

Another way to check if High Availability is replicating, is to verify there are commit transactions posted to the log file `cvs-ha/logs/ProxyServer-prefs.log`.

```
INFO:  [listen-1] Listening on port : 0.0.0.0/0.0.0.0:6445
1219077847375 org.nirala.communication.transport.DConeNet.AsyncConnector
makeConnection
INFO:  [main] Connection request to Node Id = c66b6db9-6a50-11dd-8675-
001aa036534c, host = 192.168.1.15, port = 6666, timed out in 500ms
1219077847875 org.nirala.communication.transport.DConeNet.AsyncConnector
makeConnection
INFO:  [main] Connection request to DFTPEndpoint - Node Id =
192.168.1.156666, host = 192.168.1.15, port = 6666, timed out in 500ms
1219077848578 org.nirala.admin.DiskMon start
INFO:  [main] Diskmon is monitoring C:\Thursday\cvs-ha\systemdb every
15min
1219077849000 org.nirala.communication.transport.DConeNet.ListenReactor
setupListener
INFO:  [listen-1] Listening on port : 0.0.0.0/0.0.0.0:2403
1219077849000 org.nirala.communication.transport.cvsproxy.ProxyServer
onStartedProxyListen
INFO:  [main] CVS Proxy listener is now turned ON at port :2403
```

```
1219077853765 org.nirala.communication.transport.DConeNet.Listen-
Stage$TCPStopListening onStop
INFO:  [listen-1] Host: 0.0.0.0, Port: 2403 Stopped Listening.
1219077853765 org.nirala.communication.transport.cvsproxy.ProxyServer
onStopProxyListener
INFO:  [p-queue-1] CVS Proxy listener is now turned OFF at port :2403
1219077872328 org.nirala.communication.transport.DConeNet.ListenReactor
setupListener
INFO:  [listen-1] Listening on port : 0.0.0.0/0.0.0.0:2403
1219077872328 org.nirala.communication.transport.cvsproxy.ProxyServer
onStartedProxyListen
INFO:  [mqueue-1] CVS Proxy listener is now turned ON at port :2403
```

5.10 Installing a .jar File Patch

Follow these steps to install a `cvs-replicator.jar` patch to an existing High Availability installation. You are going to copy the same jar file to the Failover Agent's and each node's `lib` directory.

NOTE:

Always read the `readme` file and `Release Notes` first.

Step 1 Download the `cvs-replicator.jar` file.

Step 2 Verify the md5 checksum.

Step 3 At the Failover Agent and all the nodes, move the existing jar file to a back up directory. (All jar files in the `/lib` directories are in the WANdisco `CLASS` path.)

Step 4 Perform a synchronized stop. See 5.7, Performing a Synchronized Stop.

Step 5 Copy the new jar file to each node's `lib` directory.

Step 6 Restart the group. See 5.8, Performing a Clean Restart After a Synchronized Stop.

Step 7 Confirm the upgrade by checking the dashboard for the newer version, and check the log file under `cvs-ha/logs` for the start header with the new version.

5.11 Setting Replicator to Start Up on System Boot

You can easily have the replicator start up on system boot. To start up on boot, edit the `init.d` scripts. You must make modifications based on your operating system. Make this change at the Failover Agent and at each node, and edit the script accordingly.

For instance, here is an `/etc/init.d/cvsreplicatord` script for Gentoo Linux.

```
#!/sbin/runscript
#
# Gentoo Linux dist compatible rc script for
# starting/stoppping cvsreplicator
#
# Copyright WANdisco
#

<start command = cvsreplicator or failoveragent>

REP_HOME="/home/admin0/cvs-replicator"
REP_OPTS="-wdog -email admins@foo.com"
export JAVA_HOME="/export/share/apps/jdks/1.5.0"
USER="admin0"

pidfile="my.pid"

depend() {
need net
}


checkconfig() {
if [ ! -f $REP_HOME/bin/<start command> ]; then
eerror "No ${REP_HOME}/bin/cvsreplicator present"
return 1
fi
prog=$"<start command>"
}

start() {
checkconfig || return 1
ebegin "Starting $prog:"
ulimit -S -c 0 >/dev/null 2>&1
ulimit -n 65000 >/dev/null 2>&1
RETVAL=0
start-stop-daemon --start --quiet -u ${USER} --chuid ${USER} --exec
${REP_HOME}/bin/<start command> -- ${REP_OPTS}
RETVAL=$?

if [ "$RETVAL" -gt 0 ]; then
eend $RETVAL "Failed to bring up cvsreplicator/failover agent"
return $RETVAL
fi
eend $RETVAL
```

```
}


stop() {
checkconfig || return 1
ebegin "Shutting down $prog:"
su ${USER} -c \"${REP_HOME}/bin/shutdown\" >/dev/null 2>&1
start-stop-daemon --stop --quiet -u ${USER} --pidfile ${REP_HOME}/logs/
${pidfile}
RETVAL=$?
if [ "$RETVAL" -gt 0 ]; then
eend $RETVAL "Failed to shutdown cvsreplicator/failover agent"
return $RETVAL
fi
eend $RETVAL
}
```

5.12 Setting the Replicator Up as a Windows Service

To set the replicator to run as a Windows service, perform the following command at the command prompt:

```
sc create CVS-Replicator binpath= C:\perl\bin\perl.exe –x –S C:\cvs-rep-
licator\bin\cvsreplicator start= auto
```

Substitute the path for Perl in your environment, and give a different path to the CVS replicator perl script, depending on where that was installed. You may want to also set `type= share`. The MicroSoft knowledge base article (http://support.microsoft.com/kb/251192) indicates that that is the default, but the `sc.exe` help for create indicates that `type= own` is the default. Note that there is a space between the equals sign, `=`, and the parameter's value.

The Services Control Panel indicates that the service has not started, because our Perl script is currently not exiting because the watchdog is running to restart the replicator. This is actually fine, because the Perl script really takes over.

5.13 Verifying System Integrity

WANdisco recommends performing periodic incremental diffs to ensure all the remote CVS repositories are in sync. This could be done periodically using a cron job, for example. But do remember that because of the way WANdisco replication works, it is possible for two CVS repositories to appear to be different while updates are happening. In other words, if you take a snapshot while updates are being applied, the snapshots could differ from each other. This does not indicate a failure of WANdisco replication.

So, when making snapshots for comparing, please make sure repositories are not being updated. The easiest way is to -

- disable client access to the replicator using Admin Console
- wait briefly for the replication group to quiesce

To do the actual diffs you are welcome to download a fast compare tool from the WANdisco support web site. Go to http://www.wandisco.com/php/support_downloads.php. This tool uses MD5 checksums to speed up comparison of CVS branches at multiple sites on a low-bandwidth, slow network link.

Alternatively you can :

- Do a local CVS checkout without the RCS keywords expansion, using the -kk option. This will help avoid spurious conflicts.
- Bring the snapshot to the machine where you would be doing the diff, perhaps using rsync.

5.14 Restricting PSERVER Access From Fail Agent to CVS Replicator Host

You can restrict clients' direct access to the CVS server. Use one of these methods.

Method 1

The `only_from` attribute of `xinted.conf(5)` is used to restrict a remote access to a particular service. The value for the 'only_from' attribute should be either 'failover agent' or '127.0.0.1' if the CVS replicator is running on the same server as the CVS pserver process. The value needs to either be a valid IP address for the server running the CVS replicator host, or the IP address that the CVS replicator is using to communicate to the CVS pserver when the CVS replicator is running on a separate host from the CVS pserver.

Following is an example of a `/etc/xinetd.d/cvspserver` that restricts the access to localhost. It assumes that the cvsreplicator is running on localhost, and cvs server is listening on port 2402.

```
service cvspserver
{
disable = no
socket_type = stream
wait = no
user = root
log_type = FILE /var/log/cvspserver
protocol = tcp
only_from = failover agent
port = 2402
server = /usr/bin/cvs
server_args = --allow-root=/data/cvs pserver
```

```
}
```

Method 2 — Disable ssh forwarding. This is done by modifying `/etc/ssh/ssh_config` and setting `ForwardAgent` to `no`. From the manual page of ssh: ForwardAgent specifies whether the connection to the authentication agent (if any) is forwarded to the remote machine. The argument must be `yes` or `no`. The default is `no`.

Agent forwarding should be enabled with caution. Users with the ability to bypass file permissions on the remote host (for the agent's Unix-domain socket) can access the local agent through the forwarded connection. An attacker cannot obtain key material from the agent, however, they can perform operations on the keys that enable them to authenticate using the identities loaded into the agent.

Method 3 — By setting the CVS repository permission to a local user and local group, access to the CVS repository can also be restricted. For example, create a local user **cvsuser** and a local group **cvsgroup**. By changing the permission on the CVS repository to this local user and group, no other user can directly write to the repository. The following configuration in `CVSROOT/config`, `CVSROOT/checkoutlist` and by adding entries in `CVSROOT/passwd` allows the users to commit to the repository.

Modify `CVSROOT/config` and set `SystemAuth` to `no`.

Add `passwd` line in the `CVSROOT/checkoutlist` file. For more details, read http://ximbiot.com/cvs/manual/cvs-1.11.21/cvs_18.html section 181.

For each user accessing the repository, add a line in the `CVSROOT/passwd` file. For more details, read Setting up the server for password authentication at http://ximbiot.com/cvs/manual/cvs-1.11.21/cvs_2.html section 30.

```
wandisco:$1$mS0G8tBk$T6K1/sHmmEPiM6sgi6rH6.:cvsuser
```

Please also read the following article: *Managing CVS Security* http://www.linux.ie/articles/tutorials/managingaccesswithcvs.php?style=printer.

5.15 Replicating User/Auth Database

NOTE:

> This procedure is valid only if WANdisco is not managing the user password files.

Let's walk through a solution based `CVSROOT/passwd` file, and then we will generalize it. Please note that CVSNT does not support the passwd file being submitted into version control.

This discussion presumes you have a replicated CVS repository already set up using WANdisco CVS High Availability.

By default, the `CVSROOT/passwd` file is not under revision control in the CVS repository. First, make sure you add it to `CVSROOT/checkoutlist` (see the CVS manual) and commit the checkoutlist administrative file into the repository.

This way, whenever you add or commit the `CVSROOT/passwd` file, the CVS server performs a rebuild of administrative files and checks out a copy of the `passwd` file in the `CVSROOT` directory automatically.

If these commands were committed via the replicator, you have a replicated password file. Any changes you make (user add/delete, passwd modify, etc.) is automatically propagated to all replicas and the user database stays synchronized.

Now let's try to generalize this to any user database file, say `/etc/passwd`.

Since you are allowed to add any administrative file into `CVSROOT`, you could follow the above process and have, say `/etc/passwd`, linked to a specific `CVSROOT/passwd`. Any changes you make and commit are replicated and `/etc/passwd` is auto updated with version control.

If you have a non file-based user repository like LDAP, you can contact WANdisco for adding bidirectional replication with the WANdisco SDK.

5.16 Changing CVS Port on Unix Flavor

To change the CVS port on any Unix flavor platform, follow these steps.

Prerequisites:

- CVS is installed
- `xinetd` is installed

Change the CVS port:

Step 1 Modify the `/etc/services` file.

Step 2 Modify the `xinetd` file for CVS.

Step 3 Restart `xinetd`.

Change the `/etc/services` file:

Step 1 Change to the root user, for example: `su root`.

Step 2 Edit the `/etc/services` file: vi /etc/services. In the line

```
cvspserver 2401/tcp
```

change the port number to the new port CVS is to run on, typically `cvspserver 2402/tcp`

Step 3 Save the file.

Modify the `xinetd` file for CVS:

Step 1 Change to the root user, for example: `su root`.

Step 2 Edit the cvs or cvspserver file, located in the directory: `/etc/xinetd.d`.

Modify the line `port = [current port]`.

For a typical installation, the modification changes the port number to `2402`.

Step 3 Save the file.

Restart `xinetd`:

Step 1 Change to the root user, for example: `su root`.

Step 2 Execute:

```
/sbin/service xinetd restart
```

5.17 Changing CVS Port on CVSNT

On the CVS server, change the value in the CVS Server Port field in the Server Settings tab of the CVS NT Server window. The CVS NT Server window is available at Start > Control Panel > CVS NT Server.

5.18 Using CVS Triggers for Sending E-mails

Many administrators like to set up CVS backend triggers that fire whenever a CVS user commits a set of file changes. With a single/master CVS server setup, e-mails can be initiated once when the `loginfo` trigger fires.

However, with the addition of WANdisco replicator, unless some safeguards are put in place, all your CVS replicas may fire the `loginfo` trigger. This could potentially cause multiple e-mail notifications. Most likely, developers do not want several e-mails for the same transaction.

The easiest way to remedy this is to designate any one node as the “e-mail hub.” Just enable the `loginfo` trigger to fire from a single site within the replication group. Alternatively, you could use the time of day to fire the e-mail alerts from a specific site. For example, you could modify the

`loginfo` trigger to send e-mails from India during 9:00 a.m. to 5:00 p.m. IST, and from the US during 9:00 a.m. to 5:00 p.m. PST.

It is allowable to have asymmetry in the e-mail triggers, but make sure not to disable the `commit-info` trigger on any node. That may cause a CVS commit transaction to abort at some sites but commit at other sites. The `commitinfo` trigger behavior at each site should be deterministic and should not cause the replicas to go out of sync.

When sending e-mail, it is important to set up the e-mail configuration to avoid long blockages or delays. Many times, an administrator uses the default SMTP settings on the CVS host. These settings by default try to use the organization domain specific e-mail server to send e-mails (by looking up the MX records corresponding to the organization's domain).

The organization-wide SMTP server may be located on a remote WAN, or it may have throttling policies for e-mails originating from the same IP address to cut down on spam. This can cause it to block or reject e-mails, which may in turn cause scripts (like the `loginfo` script) to hang or terminate. To avoid such problems with e-mail triggers, WANdisco recommends that you set up a local e-mail hub or a local SMTP agent/server. The local SMTP server should preferably be on the same host as the CVS server. It should be set up to forward/relay e-mails to the organization-wide SMTP server. This ensures the e-mail triggers are a lot faster and just need to enqueue the e-mails to the local SMTP server.

5.19 Using CVS admin -l to Lock Parts of CVS Repository

You can use the `cvs admin -l` command in conjunction with a script that puts metadata in the CVS RCS files to indicate whether a file revision or branch has been locked. One such script that is bundled with CVS sources is `cvs/contrib/rcslock`. The specific steps are:

Step 1 Add `rcslock` to the `CVSROOT/checkoutlist` file, and commit the file. The file gets replicated.

Step 2 Add the `rcslock` script to `CVSROOT` and commit it. This allows the rcslock file to be replicated to all CVS repositories.

Step 3 Add the rcslock script to the `commitinfo` trigger: for example, `ALL $CVSROOT/CVSROOT/rcslock`.This causes the `commitinfo` trigger to be checked in and replicated.

Step 4 Now, you can issue the cvs `admin -l` or `-u` command to lock and unlock. The admin command is replicated and causes all the repositories to be locked in unison.

Note that `rcslock` is an open source script, and not maintained or supported by WANdisco. The cvs `admin` command works in conjunction with a `rcslock`-like script. This has nothing to do with the WANdisco CVS replicator, per se. We just replicate the command and CVS does the rest.

5.20 Managing CVS Security

We would like to refer you to the following article in addition to the CVS manual.

Managing Access with CVS http://www.linux.ie/articles/tutorials/managingaccesswith-cvs.php?style=printer.

5.21 check-valid-dirs.pl

The `commitinfo` script that deals with the cvs add bug as well as permissions issues when committing into a directory. The pre-commit script fires in the **verify** phase of a commit transaction and aborts the commit before it even replicates. This allows the administrator to catch corrupted sandboxes before it causes the replicator to go into read-only mode.

The script can be downloaded from our support site: http://support.wandisco.com/php/support_downloads.php.

The steps for installing it are:

Step 1 Copy the script to the same location at all sites. For example, copy it to `/usr/wandisco/bin`.

Step 2 Setup execute permissions on the script file.

Step 3 Change the CVSROOT/commitinfo file to invoke this script. For example:

```
ALL /usr/wandisco/bin/check-valid-dirs.pl
```

Step 4 Check in the `commitinfo` script and make sure it is replicated to all sites.

When the CVS user tries to commit from a bad sandbox that has a directory locally added, but missing at the CVS repository, they would get an error message. This would occur prior to any replication. The message may look like:

```
[user1 ~/<3>dir0/bad]$ cvs ci -m new foo
WANdisco caught a CVS fatal error - The parent directory: (/home/cvsroot/
dir0/bad) was not added in the CVS repository. Please clean the CVS sand-
box by deleting the directory 'bad/CVS' and then run 'cvs add' on the
directory again before committing the files in the directory. You have
encountered this problem due to a know bug in 'cvs add'.
cvs commit: Pre-commit check failed
cvs [commit aborted]: correct above errors first!
```

5.22 Configuring (x)inetd Limits

For the CVS replicator, the demands on (x)inetd and its optimal configuration differ from standard CVS.

When clients access the CVS server directly, each client typically connects to the CVS server from a different address. In contrast, when a client accesses the replicator, the replicator in turn connects to the CVS server on behalf of the client. Thus, all connections to the CVS server originate from the same IP address. To address this, we recommend an appropriate increase of the per-source connections-per-second limit. For example, in `xinetd.conf`,

```
defaults
{
....
per_source = UNLIMITED
....
}
```

When a replicator instance recovers from a crash, it goes through both a local recovery process, where CVS transactions are replayed from its log, and a distributed recovery process, where CVS transactions are learned from remote nodes and replayed. During this process, the transaction rate on the CVS server can be much higher than normal. To support this higher transaction rate, WANdisco recommends an appropriate increase of the number of servers that can be simultaneously active, and the maximum allowed connection rate. For example, in `xinetd.conf`,

```
defaults
{
....
instances = UNLIMITED
cps = 10000 30
....
}
```

5.23 Using SSH Access With NIS-based User Database

You can use NIS. There are several ways to configure SSH and CVS security.

Question

As I was going through the SSH support section in the CVS Replicator Administration Guide, I found that if I use the SSH Direct method that uses the cvsrelay executable that in turn gets invoked by sshd, I need to also set up the CVS server's password database file in `$CVSROOT/CVSROOT/passwd`. Is it mandatory to set up this file? Can I avoid setting up this file and instead use the centralized NIS authentication to verify user credentials?

Answer

Yes, you can use NIS. Basically, if you are using SSH, it is really SSHd that authenticates the user's incoming password. SSHd can be set up via `/etc/ssh/sshd_config` (or wherever your config file is) to permit NIS/NIS_PLUS authentication. Note this is independent of the CVS replicator.

Once SSHd successfully completes NIS authentication, it then invokes the `cvsrelay` executable. Without `cvsrelay`, SSHd forks a CVS process with `setuid` to the user it just authenticated. There is no password checking done by CVS itself then. In other words, CVS does not consult the `CVSROOT/passwd` file or NIS when using SSHd! This is normal CVS behavior, without the CVS replicator in the picture.

With `cvsrelay`, we avoid the forking of CVS replicator. That requires `cvsrelay` to pass a valid user and a dummy password to replicator. WANdisco can supply you with scripts that add entries in the `CVSROOT/passwd` file without manually modifying it. Alternatively, you can have the `CVSROOT/passwd` file with an empty password, and then no password checking is done at the CVS level (this is only secure if that user only comes through SSH).

If you want to use both SSHD-NIS authentication and CVS-NIS authentication (in case some users come through SSH and some directly using CVS pserver), that is also possible to set up. CVS server looks for user's password first in `CVSROOT/passwd`. If it is not found, it then uses the system PAM. PAM can be configured to use NIS, LDAP, etc.

In summary, NIS can be used. There are several ways to configure SSH and CVS security.

5.24 About Watchdog Mode

By default, WANdisco starts in watchdog mode. Whenever the replicator goes down, the watchdog mode restarts it. In watchdog mode, the replication process automatically disassociates from the terminal and becomes a daemon process, so you should not try running it in the background (with &).

NOTE:

> Watchdog mode is not supported in Windows, but it is in Windows Cygwin.

You can turn off watchdog by typing `-nowdog`.

If WANdisco is unable to start up, for example if it terminates several times in quick secession, watchdog starts WANdisco in read-only mode.

```
$ ./bin/cvsreplicator -h
Usage: cvsreplicator [-v] [-verbose] [-nowdog] [-pause time]
[-email email-address]
```

-v	Print the cvsreplicator version
-verbose	Verbose, console messages go to STDOUT/STDERR instead of logs/console.txt
-nowdog	Turn off watchdog mode. WANdisco will not restart automatically if it terminates. Use this option for testing.
-pause	Time in seconds that the watchdog pauses for, before restarting service. Defaults to 0 seconds.
-email	Specify an email address to send an alert to, whenever the Watchdog restarts or shuts down WANdisco. WANdisco generates an email per local replicator activity. Set up the email account for each site with the Email Settings, described in Chapter 4, Using the Admin Console.

Use the `-email` option to generate email alerts whenever WANdisco restarts. For instance:

```
$ cvs-replicator/bin/cvsreplicator -pause 5 -email “admin@blueand-
gold.com, scmuser@blueandgold.com”
```

In order to have WANdisco Subversion Replicator automatically started on system reboots, see 5.11, Setting Replicator to Start Up on System Boot.

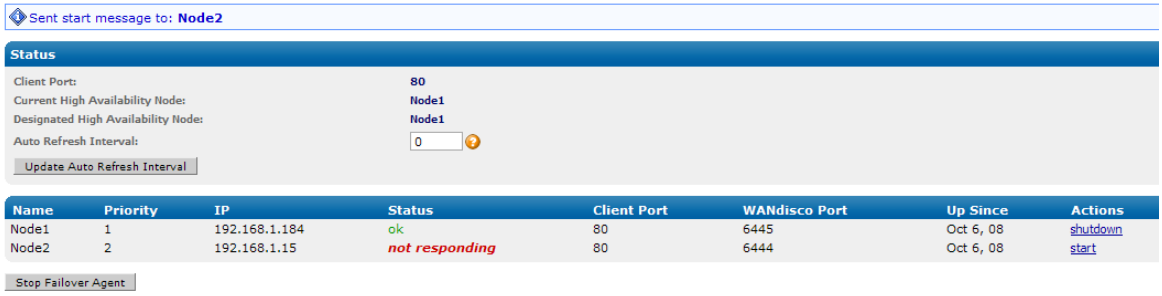
6 Procedures for Two-Node HA Groups

6.1 Recovering from Primary Node Failure

When the primary node fails, the Failover Agent sends transactions to the second node. This sets a failover flag. Use the onscreen wizard to clear the flag. WANdisco ensures the two repositories are in sync.

6.2 Recovering from Backup Node Failure

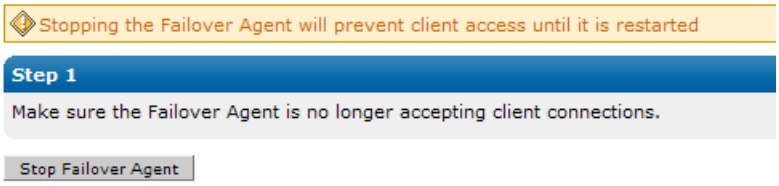
When the backup node fails, there are a few more steps to complete.



Upon a CVS commit, you see this error message about going into Unilateral mode.

Click **Click here to restore the High Availability Group**.

Step 1 Click **Stop Failover Agent**.



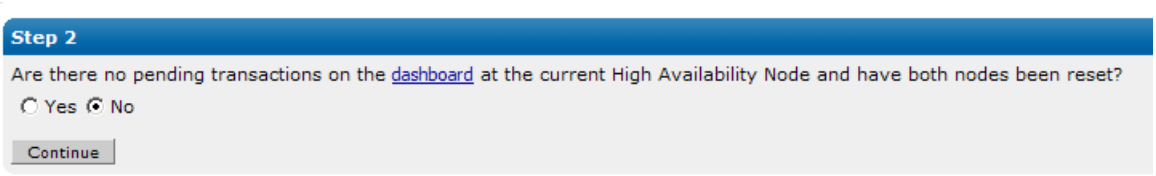
Step 2 Make sure there are no pending transactions on the dashboard, and reset both nodes. Reset the nodes by typing in the /bin directory

```
reset
```

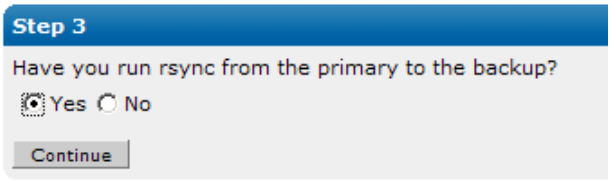
On Windows, type

```
perl reset
```

When you have done this, click **Yes**.



Step 3 Make sure the repositories are in sync. Click **Continue**.

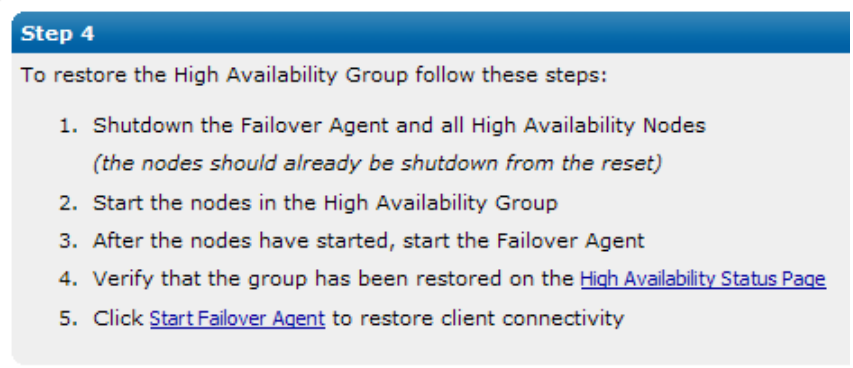


Step 4 Start the nodes. In the /bin directory, type

```
cvsreplicator
```

Start the Failover Agent. Type

```
failoveragent
```



Your HA group is restored.

7 Troubleshooting

7.1 How Do I Get WANdisco Support?

Before opening a ticket or submitting a new issue, always search the Knowledge base on *http://www.wandisco.com/php/support_login.php*.

If you want to open a ticket, you can do so at that URL.

7.1.1 How Do I Run the Talkback Script?

When you do contact WANdisco with a problem, the first thing WANdisco support asks for is the talkback file. Run the file by typing, for Unix

```
cvs-replicator/bin/talkback
```

For Windows, type

```
cvs-replicator\bin\perl talkback
```

Type in the pathname to cvsroot when prompted. The output looks like this:

```
Please open a ticket by visiting http://support.wandisco.com and upload
the /talkback-<machine name>.zip, with a description of the issue.

Note: do not email the talkback files, only attach them via the web
ticket user interface.
```

The zip file is located at the root directory. Do not email the .zip file, just attach it to an issue at *http://www.support.wandisco.com*.

7.2 General CVS High Availability

7.2.1 Connection Request Timeout Messages

Sometimes in the WANdisco logs, you see connection request timeout information messages logged. These are informational messages and should be ignored unless it is guaranteed that the connection can be established in xxx milli-seconds and happens often.

In normal operation of WANdisco, two connections are established between each of the replicated machines, a DConE connection and a DFTP connection. These two connections were established when High Availability started and are used when required. A keep-alive signal is sent on the DConE port periodically. There is no traffic on DFTP until a file transfer.

It takes between 300 to 400 milli-seconds to establish a network connection even on a slow Wide Area Network (WAN). By default, High Availability waits for 500 milli-seconds before giving up that a connection cannot be established to a peer machine and prints this informational message. What if the establishments of connection always take 501 milli-seconds. In this case, a connection is never established. To solve this problem, the timeout value is adjusted in 10% increments of the last timeout, starting at 500 milli-seconds, to a maximum of 10 seconds for each timed out connection. Upon establishment of a successful connection, this timeout value is used for subsequent connection establishment unless an adjustment is required for failed attempts.

7.3 Replication Processes

7.3.1 One-to-One Mapping For cvsroot and Replicator?

Each member of the replication group runs its own CVS pserver (identified by `cvs pserver host:port`). Within a CVS pserver installation, you can have multiple cvsroots (the `--allow-root` option in `inetd.conf` or `/etc/xinetd.d/cvspserver`). Within a cvsroot, you can have multiple modules or projects.

The mapping is one-to-one with CVS pserver rather than cvsroots. If you have multiple CVS servers, then you need to map each CVS server to its own replicator.

7.3.2 In Admin Console - Lock That's Held a Long Time

What should I do?

First, note that some CVS operations can take quite a while. For example, a CVS import of a large directory takes a long time. If you are sure this is not what you are seeing, read on...

The CVS replicator has invoked the CVS server to execute a CVS command that requires this lock. For some reason, the CVS server process executing the command is hung up. There are several possible reasons for a hung up CVS process, including a lingering lock, or a hung CVS trigger (a bug in the CVS trigger script).

What should I do when this happens?

- If possible, investigate the cause of the hang up
- Resume normal operation

To investigate the cause of the hang up, temporarily disable client connections to all CVS Replicators through the Admin Console.

If you use any of the CVS interceptor mechanisms, such as `commitinfo` scripts or `loginfo` scripts, on the CVS server, look for a hung instance of the scripts. For example, if you have a `commitinfo` script called `myCommitInfo.sh`, use `ps -leaf | grep myCommitInfo.sh`. If you succeed, you have found your problem.

If you do not use the CVS interceptor mechanisms, or if you conclude that the problem is not related to CVS interceptors, look for lingering locks in the CVS replicator's scheduler. From the CVS replicator's Admin Console, click on **Scheduler Locks**. Note all the directories where the CVS replicator's scheduler is holding locks; i.e., all entries that look like:

```
Lock: relative/path/to/a/directory
Txn Holding lock : <details of a transaction>
```

Determine the CVS server's locks directory. Look at `$CVSROOT/CVSROOT/config` to see if `LockDir` is specified. If it is not specified, the default LockDir is `$CVSROOT`.

In `LockDir/relative/path/to/a/directory`, look for CVS lock files. These files have names of the form `#cvs.lock_type.host_name.process_ID_of_creator:` for example, `#cvs.pfl.myserver.3528`.

Determine if the CVS processes that created these locks are still running. For the above example, examine the output of `ps -leaf | grep 3528`.

If you find that the CVS process that created a lock is not running (the lock was orphaned), then you have found the problem. The CVS process crashed before it could clean up the locks it created. Manually remove the lock files and directories and ensure the repositories are in sync.

If you did not find an orphaned lock, investigate networking problems between the CVS server and the CVS replicator.

Examine the TCP connections from the CVS replicator to the CVS server. Let's say your CVS server is configured to use port 2402; you have the line `<serverPort>2402</serverPort>` in `cvs-replicator/config/prefs.xml`. Run `netstat -tn | grep 2402` on the CVS replicator host (and on the CVS server host, if different from the CVS Replicator host).

On the CVS replicator host, you may see entries with the following structure:

```
Replicator Host netstat output Proto Recv-Q Send-Q Local Address Foreign
Address State
tcp A B ReplicatorHost:37751 CVSServerHost:2402 ESTABLISHED
```

On the CVS server host, you may see entries with the following structure:

```
Replicator Host netstat output Proto Recv-Q Send-Q Local Address Foreign
Address State
tcp C D CVSServerHost:2402 ReplicatorHost:37751 ESTABLISHED
```

A, B, C and D in the above samples are numerical values indicating the corresponding queue sizes. Non-zero values indicate problems as follows:

If A is non-zero, the CVS Replicator is hung. Contact WANdisco support.

If B is non-zero, the TCP connection is hung on the CVS Replicator host.

If C is non-zero, the CVS process is hung.

If D is non-zero, the TCP connection is hung on the CVS Server host.

To resume normal operation:

Step 1 After ensuring the replicator at the site experiencing the problem doesn't have any open CVS connections on the Dashboard (other than possible hung transactions), shut down the CVS replicator. This is to minimize disruption of service to CVS users.

Step 2 After the CVS replicator is shut down, if there are any CVS interceptor scripts (such as `commitinfo` or `loginfo` scripts) running, kill them.

Step 3 After the CVS replicator is shut down, if there are any CVS processes running, kill them.

Step 4 Delete any lingering locks in the CVS repository.

Step 5 Restart the CVS replicator.

7.3.3 When I Commit, Why is Replicator Being Bypassed?

Question:

I modified `/etc/services` cvspserver port to point to 2402. I am running the replicator on port 2401. Everything works fine from a remote machine (not the same as cvspserver machine). How-

ever, whenever I do a commit from the host on which cvspserver is running, it seems to go directly into cvs without getting replicated. Why is this happening?

Answer:

Many CVS clients, including GNU cvs, look up the `/etc/services` for the port to use to connect with CVS. When you run such a CVS client on the same host as cvspserver, what ends up happening is port 2401 (default) is not used, as the /etc/services port has a higher precedence. The cvs client happily connects to 2402 as is the case with your cvspserver port setting in `/etc/services`, thus bypassing the CVS replicator.

To verify, just run with the trace on. For example:

```
$ cvs -t commit -m new
```

And you see the port as 2402 and not 2401.

There are several ways to deal with this, although WANdisco recommends the last option, as it is the least error prone from an operational perspective.

- setup an environment variable `CVS_CLIENT_PORT` to 2401
- setup CVSROOT to explicitly specify 2401: e.g., `pserver:user@:2401/my/cvsroot`
- rename cvspserver to cvspserver2 in `/etc/services` and in the inetd.conf or rename `/etc/xinetd.d/cvspserver` to `/etc/xinetd.d/cvspserver2`, thus avoiding any interference.

7.3.4 CVS: Deferred Writes with Long Transactions

The replicator by default waits for 60 seconds when a write transaction is initiated by the CVS client, and then issues a message to the CVS client console if the transaction has not completed in that 60 seconds. The transaction time may exceed 60 seconds due to a couple of reasons: slow network, large checkins, the local replicator that the client is connecting to may be back logged. In other words, after getting the deferred write message, the client can disconnect and the replicator attempts to apply the changes in the background. This doesn't guarantee that the write will not be aborted by the CVS server due to an **up-to-date check failed** condition. All it means is the user CVS sandbox doesn't need to block. This behavior can be turned off.

The message that the replicator puts out when it detects a slow write operation (an operation that exceeds the default 60 seconds) :

```
cvs write: Replicator has accepted the change-set. You can disconnect the
CVS client if you want, changes will be applied automatically on your
behalf.
```

This behavior can be controlled in the prefs.xml file.

```
<CVSProxy>
..........
<!-- Defaults to true -->
<UseDeferredWrites>true</UseDeferredWrites>
<!-- Defaults to 60s or 60000 milliseconds -->
<ClientWaitTimeout>60000<ClientWaitTimeout>
</CVSProxy>
```

7.3.5 WinCVS Client Responds “Exited Normally with Code 1”

Why am I getting this message when the normal exit code is 0? (Seen in WinCVS client version 2.0.2.4 build 4, using compression level 5.)

I added a 300 mbyte file (cvs add) successfully. Upon commit, the client responded, after a long delay, with the message `exited normally with code 1`. I confirmed that WANdisco successfully committed the transaction (the large file was in the repository, and the WANdisco logs indicated a successful commit).

The sandbox still shows the large file as `added but not committed`. The updates from WinCVS failed with "exited normally with code 1". I deleted the large file from the sandbox and tried to update, but I still had the same problem.

What finally worked was that I deleted the sandbox, and checked it out again. Everything works fine now.

7.3.6 WinCVS Wants Passphrase - SSH Access to Replicator

Why is WinCVS prompting me for a passphrase when I use SSH access to the replicator?

This is not a replicator issue. Ensure `pageant` or `ssh-agent` is running with private keys already loaded. A less secure way is to have no passphrase on the private keys at creation time.

7.3.7 With WinCVS and SSH, Commits Not Replicated

Ensure `cvsrelay` is being picked up as the cvs executable by `sshd`. This can be accomplished by setting up the `CVS_SERVER` environment variable in the Windows system variables dialog box or renaming cvs to point to `cvsrelay`. A write or a commit operation is replicated only if it is submitted to the replicator. Bypassing the replicator changes the CVS repository directly and cause other replicas to go out of sync.

Remember to look for the following pattern in the logs/ProxyServer*.log to know if the replicator even got the write or commit changes:

```
...
INFO: [reader-2] Submitted after verification: cvs-proposal-2f195a4b-
dea9-11d9-b140-00065b193ef5_4
1118955513564 org.nirala.communication.transport.cvsproxy.CVSProposal out-
put
INFO: [gsn-1] Executing cvs-proposal-2f195a4b-dea9-11d9-b140-
00065b193ef5_4
1118955513575 org.nirala.communication.transport.cvsproxy.CVSProposal
onCommit
INFO: [reader-1] Committed CVS transaction cvs-proposal-2f195a4b-dea9-
11d9-b140-00065b193ef5_4

...
```

If you don't see a `Submitted` line in the log file, the CVS write operation never made it to the WANdisco replicator, and so it cannot be replicated.

Also check the host and port setting in `/etc/prefs.conf`.

If you still have trouble, run `cvsrelay` with a `-v` option. Just set the Windows environment variable `CVS_SERVER` with `-v` option to `cvsrelay` (for example, `..../bin/cvsrelay -v`). This causes a good amount of logging to `syslog` on the host on which `sshd` is running. You can diagnose the problem by looking at the syslog.

7.3.8 No Submitted Lines in Log File After Commits

The most common reason for this situation is an incorrectly set up `CVSROOT` environment variable. For example, if the CVSROOT environment variable is set up as `/home/cvs`, then the CVS client directly modifies the repository using the local access protocol.

The `CVSROOT` environment variable needs to be set up as:

```
$ setenv CVSROOT :pserver:@:/
```

If the replicator is running on port 2401, then the replicator port setting can be skipped unless you are running on the same host as cvs pserver repository.

Also see the article in 8.5, Should I Worry About Time Changes?

7.4 Error Messages

7.4.1 Replicator Aborting with System Error Message

The symptoms are as follows in one of the logs/Proxy* log file:

```
SEVERE: [reader-1] Encountered a system error from CVS server : error 0 /
home/cvsroot/dir2: no such repository The CVS command file is:.... Above
error constitutes a system error according to the CVS error catalog.
Shutting down. Please verify setup and ensure repositories are in sync
before starting
```

or

```
SEVERE: [reader-1] Encountered a system error from CVS server : E cvs
[tag aborted]: could not open lock file `/cvsd/root/module/,Run.java,':
Permission denied
```

The same transaction commits at other sites where the repository structure and file level permissions are fine.

The cause for this is the directory does not exist, or write permissions are missing at the other replicator site. The repository structure must be the same at all sites for replication to occur. Create the same repository at the errored site and re-start the replicator. It then applies pending transaction to the repository.

7.4.2 Aborted CVS transaction with 'I HATE YOU' message

Question

We are seeing the following stack trace in the log file and the CVS replicator is shutting down.

```
1122522454973 org.nirala.communication.transport.cvsproxy.state.
ResponseGatherState writeModeException
SEVERE: [reader-1] Exception encountered in write mode: previous line:
null
current line: I HATE YOU
countResponses: 0
countOKsPriorToFirstWrite: 2
countOKsAtCompression: 0
totalOKs: 3, server state :STATE : (ResponseGatherState) , txn# - null,
terminator is ON : true, null
```

```
 org.nirala.commmunication.transport.DConeNet.exceptions.SocketException:
java.io.IOException: Connection reset by peer
at sun.nio.ch.FileDispatcher.read0(Native Method)
at sun.nio.ch.SocketDispatcher.read(SocketDispatcher.java:21)
at sun.nio.ch.IOUtil.readIntoNativeBuffer(IOUtil.java:233)
at sun.nio.ch.IOUtil.read(IOUtil.java:206)
at sun.nio.ch.SocketChannelImpl.read(SocketChannelImpl.java:207)
at org.nirala.commmunication.transport.DConeNet.TCPPeerConnection.read(TCP-
PeerConnection.java:410)
...........
...........
erRunnable.run(ReadReactor.java:164)
at java.lang.Thread.run(Thread.java:595)

1122522454974 org.nirala.exceptions.BaseException onFatalException

SEVERE: [reader-1] Aborted CVS transaction cvs-proposal-d7bc8409-ff51-
11d9-8d6a-0011110ee979_18 due to errors from CVS repository.
```

Resolve the errors from CVS and then restart. Most of the time a restart fixes the problem. If a restart doesn't work, please verify that the repositories are in sync, delete `logs/tmp` at all replicas before resuming operation.

Answer

Whenever you see the log ending with an `Aborted CVS transaction...` please look at the previous log line. (Each line starts with a long 13 digit timestamp). In this case the previous line indicated the replicator received an `I HATE YOU` message from the CVS server. This happens if the CVS user or password do not match across replicas. In other words, a commit with a password X on site 1 and password Y on site 2 for the same user causes the replicated commit to be rejected in the second replica.

Don't worry, the replicator preserves the commit in its transaction log. So simply fix the password, restart, and the changes are applied.

When the replicator encounters such an error, it does an automatic shutdown to prevent further repository evolution until the administrator can fix the password or user name.

7.4.3 Missing License Key File

CVS High Availability depends on a license key file being present in the `cvs-failover/config` and `cvs-ha/config` directories for each ndoe. Please get a valid license from WANdisco and copy the file to the config directory. The Replicator does not start without the license file.

7.4.4 I'm Getting a SEVERE Exception

I'm getting a SEVERE exception, and replicator is aborting the CVS transaction and shutting down.

If you get a message in the logs/Proxy*.log file similar to

```
SEVERE: [reader-1] Encountered a system error from CVS server : E cvs
[commit aborted]: could not find desired version 1.16 in /development/
software/cvsroot/libcobra/Build,v
```

it means the replicator has detected an out of sync condition. Remember the replicator continuously monitors your repository for any out of sync issues. If it detects this has occurred, it triggers an automatic shutdown to prevent further corruption.

This could happen if some one accidently committed directly to CVS, bypassing the replicator, and ramped up the version in one site without giving the replicator any chance of replicating. This can be easily resolved by following the reset procedure (See 7.5.1, I Directly Committed to CVS, How Do I Rsync?).

Follow all precaution to avoid bypassing the replicator:

Step 1 Ensure only cvsreplicator host/IP address is allowed to connect to cvspserver.

Step 2 Protect direct logins in cvs replicator or cvs pserver box from end user.

Step 3 When administering, ensure a valid `CVSROOT` that points to the replicator.

Step 4 Avoid the cvspserver port issue, (see 7.3.3, When I Commit, Why is Replicator Being Bypassed?).

7.5 Oops!

7.5.1 I Directly Committed to CVS, How Do I Rsync?

If you bypassed the replicator, you can reset the replicator state with these steps:

Step 1 Shut down all replicators.

Step 2 Reset each replicator. For Unix, type

```
$ cvs-ha/bin/reset
```

For Windows, type

```
cvs-ha\bin\perl reset
```

Step 3 If this happened during an initial setup/evaluation stage, delete the old project in CVS and create a new one.

If this happened on a production repository, just re-sync all the repositories to the same state/data.

Step 4 Restart all the replicators.

NOTE:

It is very important that you take all precautions to avoid directly checking in or committing to the backend CVS repository.

7.5.2 I Pressed Ctrl-C During a CVS Command!

If you were executing a read command (a command that does not modify the CVS repository), you do not have to do anything.

If you were executing a write command, update your sandbox after the replicator has applied the command to the repository.

In addition, if you were adding files to the repository (either cvs import or cvs add, followed by cvs commit), wait until you update your sandbox before you continue to use it.

7.6 General CVS Issues

7.6.1 CVS Login Appears to Hang

After setting up a new replicator instance, you may try to do a CVS login via the replicator, but the login seems to hang. You may see a message such as:

```
[user@pee ~/tmp/src]$ cvs -d :pserver:user@tao:2801/home/cvsroot login
Logging in to :pserver:user@tao:2801/home/cvsroot
CVS password:
```

After you type the password, the login appears to hang.

The vast majority of the time, an invalid IP or host name causes this. For example, here we have the user specifying `tao` as the CVS replicator host name instead of `pee`. This causes the CVS client to do a connect to `tao` replicator, which may be behind the firewall, or port 2801 on `tao` is not exposed through the firewall. The TCP connection never completes. It can take several minutes to detect a connection failure, unless the host with the IP address itself is down. If the external IP address points to a firewall or a router, from the TCP perspective, this is just a slow connection issue, and the TCP stack would send the `SYN` packet and block.

You can confirm this by running with `-t` option to the CVS client.

```
[user@pee ~/tmp/src]$ cvs -t -d :pserver:user@tao:2801/home/cvsroot
login
-> main: Session ID is 630242e861f94567
-> main loop with CVSROOT=/home/cvsroot
Logging in to :pserver:user@tao:2801/home/cvsroot
CVS password:
-> Connecting to tao(67.160.228.194):2801.
cvs [login aborted]: connect to tao(67.160.228.194):2801 failed: Connec-
tion timed out
```

You can see the CVS client timing out. The user really wanted to connect with the replicator on `pee` but mistakenly specified `tao`.

7.7 CVSNT Issues

7.7.1 CVSNT Generates CVSLock Error Message

Problem

When I connect to the replicator for my CVSNT repository, I get the message

```
cvs [login aborted]: unrecognized auth response from localhost: CVSLock
2.2 Ready
```

Solution

Make sure the replicator is configured to connect to the CVSNT server port and NOT the CVSNT lock daemon port. This message is coming from the CVSNT lock daemon. You can go to the CVSNT control panel and change the server and lock daemon ports.

8 Frequently Asked Questions

8.1 Why Are So Many Java Processes Running?

On older versions of Linux, every thread is listed as a process by the `ps` command. This does not affect the operation of JIRA Clustering. WANdisco does not support the older versions of Linux.

8.2 How Do I Deal with Failover Agent Failure?

If the failover agent fails, the watchdog script immediately restarts it. If the machine crashes, service is unavailable until the machine is rebooted and the failover agent is restarted.

You could also run the failover agent on a hardware cluster. The Veritas Cluster Server is an example of a commercial solution. See http://www.symantec.com/business/products/overview.jsp?pcid=pcat_business_cont&pvid=20_1.

Linux-HA is an example of an open-source solution. See http://www.linux-ha.org/.

8.3 Can I Store Logs or Content on NFS?

NFS (Network File System) allows files and directories to be accessed remotely over a network using NFS clients. NFS clients are typically built into the operation system kernel these days. However, some operations, like renaming a file, are not guaranteed to be atomic over NFS. Here is a snippet from the rename function's `man` page on Linux, for example:

```
BUGS
On NFS filesystems, you can not assume that if the operation failed the
file was not renamed. If the server does the rename operation and then
crashes, the retransmitted RPC which will be processed when the server is
up again causes a failure. The application is expected to deal with this.
See link(2) for a similar problem.
```

Code management systems such as CVS make heavy use of the `rename` operation to modify the underlying databases. Independent of WANdisco, it is a risky practice to store CVS database content on NFS. The code management community at large recommends not using NFS for storing repositories.

WANdisco High Availability is bundled with a built-in transactional journal and an object database. These are by default stored in the `cvs-ha/systemdb` and `cvs-ha/config` directories. These directories should not be mounted on an NFS drive. The replicator itself may be installed on an NFS drive but the systemdb and config directories should be on direct storage (non-NFS

options like RAID, SCSI, SAN, etc). Replicator's transactional integrity can be compromised if writes to an NFS server are lost due to a potential NFS client cache crash after the NFS server has indicated IO completion.

8.4 Why is Set Up Configuring IP Addresses as 0.0.0.0?

The address 0.0.0.0 is a special IP address, treated as a wild-card IP address. In other words, on a machine with multiple NICs (Network Interface Cards), it binds to all interfaces. The advantages of using wild-card IP address include:

- It avoids binding to a fixed IP address. If the host's IP address changes, (for example, the subnet changes, or the machine is moved to a different location) you don't have to change the wild-card IP in the prefs.xml file to the new IP address.
- There is wider bandwidth to TCP clients. Now TCP clients can connect to any NIC, because JIRA Clustering is listening on multiple NICs.

The disadvantage to using the wild-card IP is that it gives coarser access control at the IP address level, as all address are being listened to at the specified port.

You can always switch from the wildcard IP address to a fixed, static IP address or a DNS hostname, though for the most part, WANdisco recommends you stick with wild-card addressing.

8.5 Should I Worry About Time Changes?

Time changes have no effect on the operation of High Availability.

8.6 Does WANdisco Support Dynamic DNS?

Yes, WANdisco supports dynamic DNS, but strongly discourages its use.

If a hostname is specified during the setup process, WANdisco requires that it should be able to connect to a valid DNS and resolve the hostname to valid IP address upon startup. If the hostname cannot be resolved to an IP (either by not being able to connect to DNS, or no entry is found at the given hostname), WANdisco dies gracefully. This has never been a problem during production and with static IPs.

However, if dynamic DNS support is required, please modify the `prefs.xml` file at each site and set `UseDynamicDNS` to `true` in `DConeNet` element.

```
<Preferences>
  ...
  <DConeNet>
```

```
    ...
    <UseDynamicDNS>true</UseDynamicDNS>
  </DConeNet>
```

In addition, the following Java security properties should be set to different Time-to-live (TTL).

```
networkaddress.cache.ttl
networkaddress.cache.negative.ttl
```

Please read InetAddress Caching for more details.

8.7 Can I Use SSH Tunnel to Navigate a Firewall?

You can use SSH tunnels to test connectivity to a replicator's DConeNet port through a firewall.

NOTE:

> SSH tunnels are not recommended for a production environment.

SSH tunnels are temporarily created using a secure shell. If the shell hangs up for any reason, the tunnel goes away. You don't want the connectivity to a replicator's WAN port to be dependent on a transient shell. We recommend using permanent IPsec tunnels (VPN/NAT devices can help) for navigating firewalls.

For deferred writes with long transactions: The replicator by default waits for 60 seconds when a write transaction is initiated by the CVS client, then issues a message to the CVS client console if the transaction has not completed in 60 seconds. The transaction time may exceed 60 seconds due to a couple of reasons: slow network, large chickens, local replicator that the client is connecting to may be back logged. In other words, the client, after getting the deferred write message, can disconnect, and the replicator attempts to apply the changes in the background. This doesn't guarantee that the write will not be aborted by the CVS server due to the "upto-date check failed" condition. All it means is the user CVS sandbox doesn't need to block. This behavior can be turned off.

The message that the replicator puts out when it detects a slow write operation (exceeds the default 60 second time): cvs write: Replicator has accepted the change-set. You can disconnect the CVS client if you want, changes will be applied automatically on your behalf.

This behavior can be controlled in the Prefs.xml file. Look for `UseDeferredWrites`.

```
<CVSProxy>
..........
<!-- Defaults to true -->
<UseDeferredWrites>true</UseDeferredWrites>
<!-- Defaults to 60s or 60000 milliseconds -->
```

```
<ClientWaitTimeout>60000<ClientWaitTimeout>
</CVSProxy>
```

8.8 CVS Tagging and Branching Support

Question

Tagging and branching end up touching all the files in a module. Does each file have to be resynched between repositories? Does this affect the time it would take to tag or branch across replicas?

Answer

First of all, replication is invisible to the user: the user operates CVS normally. Replication is a transparent network proxy from a CVS client perspective.

Tagging and branching are treated as a write operation from the CVS perspective, as they change state on disk. The operations are propagated to all replicas.

The way the WANdisco CVS replicator works, files are not directly synced between repositories. This is vastly different from CVSup or rsync based approaches.

The logical tag or branch command is replicated to all the CVS repositories. From a CVS client perspective, you only block for your local CVS repository to finish the CVS command (in the example, that would be tag or branch). So any latency you are likely to see is local access only.

This also reduces the network bandwidth utilization, as the tag/branch command being replicated is typically a lot smaller compared to the modified file state on disk that would have to be copied (even using incremental copy) with rsync based approaches. For instance, a repository with 10000 files (10k each) on tagging causes rsync or any disk based tool to copy 10000 files, as each changed due to new tagging. This could be hundreds of megabytes in network transfer. With the replicator, the tag command itself is replicated, which is just a few hundred bytes.

The remote replicas (there can be any number of them) apply the tag or branch command in the correct sequence. When another write operation (tag,commit,admin, etc.) arrives at the remote repository, it is already properly ordered to ensure all replicas stay consistent with each other.

8.9 How Do I Know cvsrelay is Being Invoked?

First check the system log (on Linux: `/var/log/messages`), for any error messages from cvsrelay. A sample error:

```
tao /var/log# pwd
/var/log
tao /var/log# grep cvsrelay messages
```

```
Dec 1 21:02:04 tao cvs-replicator/bin/cvsrelay[15105]: failed to open /
etc/prefs.conf
Feb 24 22:03:07 tao ./cvsrelay[5348]: cvs: delagated login failure.
Check syslog
for further info. Server response - "error 0 /tmp/cvstest: no such repos-
itory"
```

You can also setup `cvsrelay` to run in verbose mode by passing an optional `-v` parameter.

```
setenv CVS_SERVER “...../bin/cvsrelay -v”
```

This causes the system log (on Linux: `/var/log/messages`) to contain detailed messages from cvsrelay.

8.10 Advisory: CVS Add Bug in All CVS Client Versions

A CVS add operation can corrupt the CVS user's sandbox if the CVS server fails or aborts the command due to a permissions issue, server crash, etc. The CVS add does not wait for the server to finish the CVS add operation on the server side; it adds the CVS directory locally, thus confusing the sandbox if the server fails to add. If the user later tries to commit files in the directory that was never added to the cvs server , the cvs server issues a **no such directory** exception. This can cause the replicated repository to go out of sync. See the cvs bug report http://savannah.nongnu.org/bugs/?func=detailitem&item_id=15797 for more details.

To recover from such an error, add the missing directory to the CVS server manually and restart the replicator.

WANdisco also provides a `commitinfo` trigger that can catch the add bug and report an error to the CVS client before any replication has occurred. It is highly recommended that you use or incorporate this trigger in your existing pre-commit commitinfo trigger. For the download link, see 5.21, check-valid-dirs.pl.

8.11 Advisory: Zlib Compression Bug in 1.12.12, 1.12.13

WANdisco recently discovered a bug in CVS 1.12.13 that causes several CVS clients, including the WANdisco replicator, to block indefinitely, as CVS server's zlib implementation gets hung while parsing zlib compressed files. A bug and a fix has been posted with the CVS project team: see https://savannah.nongnu.org/bugs/?func=detailitem&item_id=14840 for more details.

The CVS community is to release a fix in version 1.12.14 to address the bug. Until then, we recommend the following workaround.

- Recompile CVS 1.12.12 sources with configure -with-external-zlib after ensuring the libz.so on the system is not 1.2.2. On Red hat linux, the default version is typi-

cally 1.1.14. This can be determined for example by looking at the link `/usr/lib/libz.so` points to.

We have tested with the latest zlib (1.2.3 at the time of writing) from http://www.zlib.net/ and it works with CVS 1.12.12 and configure --with-external-zlib.

If you compile zlib, make sure you run configure with --shared option and ensure `LD_LIBRARY_PATH` points to the path of the newly compiled libz.so.

8.12 Advisory: Avoid Force Commit with CVS Commits

The CVS commit command supports a `-f` or a `Force Commit` option that can be used to force the revision of the file to be incremented, even if the file has not changed. It is a dangerous option, as the developer may sometimes hit commit twice from the CVS client if for any reason they do not get communication back from the CVS server that the command was completed. This causes unnecessary incrementation of revisions.

The situation in the previous scenario is benign. However, if the replicator detects a CVS server crash, it can retry the command at the site where the crash was detected. If the CVS server had in fact completed the command, but failed to notify the replicator before it crashed, this would cause revision numbers to go out of sync across the replication group.

A future release of the replicator will work around this problem. Until then, we recommend not using the force commit option.

8.13 Advisory: CVS 1.12.13 Bug In Import

We have recently discovered a bug in cvs server 1.12.13 that causes "cvs import with a -n option" to fail. A bug has been filed with the CVS project team, see http://savannah.nongnu.org/bugs/?func=detailitem&item_id=15703 for more details.

If you are using cvs server 1.12.13, you will want to switch off the verify mode (impact performance degradation) in the replicator to avoid the bug. See http://www.wandisco.com/techpubs/cvsmanual/node18.html for more details.

Alternatively you could move to cvs server 1.11.21 with WANdisco keyword patch http://support.wandisco.com/index.php?_m=downloads&_a=view&parentcategoryid=4&pcid=0&nav=0.

8.14 WANdisco Authentication

Authentication is the process of determining whether someone or something is, in fact, who or what it is declared to be. Authorization is the process of giving someone permission to do or have something.

WANdisco is compatible with CVS, among other SCM tools. There is a variety of forms of authentication for each. WANdisco in CVS can be handled through two forms of authentication, CVS Password and System Auth (NIS, LDAP).

CVS Authentication

CVS Password

CVS Password is the WANdisco-recommended method of authentication. The CVS user/passwords on all repository hosts must match. WANdisco recommends that the CVS passwords are set in the `username;password;user` schema.The CVS password files should be under CVS control, so that they are replicated throughout the replication group. This is required because the WANdisco CVS replicator creates a peer-to-peer replication system. Any replica of the CVS repository is accessible by every valid CVS user. To ensure that the passwd files stay in sync at all sites, navigate to the `checkoutlist` file, which is in the `CVSROOT` directory, and insert the line `passwd Could not update CVSROOT/passwd`. This causes an error message if the file cannot be checked out.

8.15 How WANdisco Handles CVS Authentication

WANdisco supports the following protocols: pserver, ext (ssh) and gserver. Authentication is always done by CVS, regardless of the protocol being used. This authentication is done at each site individually. Any PAM can be used with CVS plus WANdisco as long as you use one of the mentioned protocols.

As authentication is done at each site, a user has to exist with correct credentials at all the sites. With pserver protocol and **SystemAuth** set to **no**, CVS consults `CVSROOT/passwd` file. The entries defined in this password file have the format `username:password[:run-as-username]`. This user does not have to be created at all the sites, but the user does need to be defined in the `CVSROOT/passwd` file at all the sites.

If we follow the A:B:C format, that is, run-as-username, then only one physical user has to be created, and it becomes easy to manage the users using the `CVSROOT/passwd` file. The CVS daemon runs as **cvsuser**, and this **cvsuser** is the only user that has any read/write privileges to the repository. Note that in the cvs-log, the entry is made for the actual user doing the changes, not **cvsuser**.

You can change the user back to **root**, but make sure that the user has read/write permission to the CVS repository, and the user physically exists, and belongs to the correct group. This may cause extra administrative work in the distributed environment, but if you choose to do it, WANdisco does not prevent you.

For the authorization part, WANdisco needs to find out the username from the input byte stream. This information is readily available with pserver protocol, but not with ext and gserver protocol.

To support the later protocols, WANdisco has written appropriate relays. These relays interpret the authentication portion of the input stream, and pass it to CVS for authentication. After successful authentication, this relay gets the username from either CVS or the system, replaces the

authentication portion of the input stream to pserver protocol with the username (and a dummy password) and replicates.

Note that the user has already been authenticated with ext and gserver (kerbrose, LDAP, etc.). CVS is configured to use the `CVSROOT/passwd` file for pserver protocol, and now the password file has username with dummy password. The fake authentication succeeds and WANdisco replicates it at all sites. That is how ext and gserver protocol are implemented.

The *WANdisco CVS Replicator installation/User guide* discusses in detail how to set up the system for ext (ssh) protocol. The same setup applies to gserver protocol. The relay for SSH is distributed with the standard product and you should have it. The relay for gserver is not distributed and has to be coordinated with the sales team.

Appendix A - Installing Java and Perl

You should have already installed Java and Perl at all the sites in your replication group for your trial evaluation. However, any new site you add to the replication group needs Java and Perl installed as well.

Installing Java

Step 1 Install JDK 1.5 and define the `JAVA_HOME` environment variable to point to the directory where the JDK is installed. You can download JDK 1.5 from the URL below.

```
http://java.sun.com/javase/downloads/index_jdk5.jsp
```

Step 2 Add `$JAVA_HOME/bin` to the path and ensure that no other java (JDK or JRE) is on the path.

```
$ which java
/usr/bin/java

$export JAVA_HOME="/usr"
```

or

```
$which java
/export/share/apps/jdk/1.5.0/bin/java

$export JAVA_HOME="/export/share/apps/jdk/1.5.0"
```

Step 3 Ensure the full JDK is installed, not just the JRE. This can be confirmed by running `java -server-version`. If it generates a **not found** error, repeat Steps 1 and 2.

If you find package management problems or conflicts with the JDK version you are downloading (for example, rpm download for Linux), you may want to use the self-extracting download file instead of the rpm (on Linux) package. The self-extracting download easily installs in any directory without any dependency checks.

Installing Perl

Step 1 On UNIX or Cygwin, install perl version 5.6 or greater and ensure that the perl executable is on the system path.

Step 2 On Windows, install ActivePerl version 5.8 or greater and ensure that the perl executable is on the system path. You can download the MSI installer for ActivePerl from the URL below.

`http://activestate.com/Products/Download/Download.plex?id=ActivePerl`.

You may want to install Perl modules if you are doing any advanced work with JIRA MultiSite. For example, if you run the `logtimefix` utility, you may get an error such as:

```
Can't locate Date/Manip.pm in @INC (@INC contains: /usr/perl5/5.8.4/lib/
i86pc-solaris-64int /usr/perl5/5.8.4/lib /usr/perl5/site_perl/5.8.4/
i86pc-solaris-64int /usr/perl5/site_perl/5.8.4 /usr/perl5/site_perl /
usr/perl5/vendor_perl/5.8.4/i86pc-solaris-64int /usr/perl5/vendor_perl/
5.8.4 /usr/perl5/vendor_perl .) at ../bin/commitrate line 175.
```

In this case, you would need the Perl module for `Date::Manip`. To install Perl modules, just follow the instructions from *http://cpan.org/misc/cpan-faq.html#How_install_Perl_modules*.