



## **Distributed NetBeans for OpenVMS Version 1.1 Installation Guide and Release Notes**

December 2005

HP-AXPVMS-IDESERVER-V0101--1.PCSI\$COMPRESSED (IDE Server Kit on OpenVMS Alpha)  
HP-I64VMS-IDESERVER-V0101--1.PCSI\$COMPRESSED (IDE Server Kit on OpenVMS I64)  
distnbopenvms36\_v1\_1.nbm (Distributed NetBeans Client Module)

### **Contents**

- » 1 Overview of Distributed NetBeans
- » 2 Before Installing Distributed NetBeans
  - » Hardware Prerequisites - Client
  - » Hardware Prerequisites - IDE Server
  - » Software Prerequisites - Client
  - » Software Prerequisites - IDE Server on OpenVMS Alpha
  - » Software Prerequisites - IDE Server on OpenVMS I64
- » 3 Downloading and Installing Distributed NetBeans
  - » Downloading the NetBeans Client and IDE Server Kits
  - » Installing the Client Kit
  - » Installing the IDE Server Kit
- » 4 After Installing Distributed NetBeans
  - » IDE Server Startup and Shutdown
  - » Accounts, Quotas, and System Parameters
  - » Logical Names
- » 5 Release Notes
  - » Default Java SDK Version
  - » Define IDE\$ALLOW\_EXTAUTH Logical for Users with ExtAuth Flag Set in SYSUAF Record
  - » JSSE Keystore and Truststore
  - » IDE\$SERVER and IDE\$USER User Processes on OpenVMS
  - » Files Edited in NetBeans Require STREAM\_LF Record Format (SMB Filesystems Only)
  - » Remote Command Execution and Privileges
  - » Using Ant Outside of Distributed NetBeans
- » 6 Known Problems in the IDE Server and Distributed NetBeans Client
- » 7 Software Support

### **1 Overview of Distributed NetBeans**

Distributed NetBeans comprises two parts:

- **Distributed NetBeans Client for OpenVMS**, which is a plug-in for NetBeans running on your desktop. You install the NetBeans IDE (from NetBeans.org) and the Distributed NetBeans Client for OpenVMS on your desktop system.
- **IDE Server for OpenVMS**, which runs on OpenVMS and provides remote services for the client plug-in. You install the IDE Server on your OpenVMS system. (You do not need to install the NetBeans for OpenVMS IDE or any additional plug-in modules on your OpenVMS system.)

Communication between the client system and the remote server system is encrypted using SSL.

How to use Distributed NetBeans is explained in the **online help** that is available through the NetBeans JavaHelp system when you install the NetBeans client module.

This help is fully integrated with the standard NetBeans JavaHelp and may be opened from the Help menu. (The HelpSet option in the Help menu lists installed module help files.)

JavaHelp texts may be copied and pasted into an external text editor or printed. URL links provided in JavaHelp can be copied and pasted into a browser's address field.

## **2 Before Installing Distributed NetBeans**

### **Hardware Prerequisites - Client**

For NetBeans IDE desktop system hardware and software prerequisites, see the Readme file at <http://www.netbeans.org/community/releases/36/relnotes.html#SystemRequirements>.

### **Hardware Prerequisites - IDE Server**

The Distributed NetBeans IDE Server requires the following:

- » 500MHz minimum Alpha or I64 processor
- » 256MB of memory
- » 2000 blocks minimum free disk space on your system disk
- » ODS-5 or ODS-2 disk (ODS-5 is required to use the Java SDK with Distributed NetBeans)

### **Software Prerequisites - Client**

- » NetBeans 3.6 (from NetBeans.org) running on your desktop system
- » Java Standard Edition v 1.4.2-07 or higher (from java.sun.com) running on your desktop system
- » (Optional) X Windows Server (such as eXcursion) running on your desktop system

**Note:** An X Windows Server is required to run remote Java applications that use Swing, and to use some of the features of the Distributed NetBeans Client (such as "Use XTerm for Compile" and "Use XTerm for Execution").

### **Software Prerequisites - IDE Server on OpenVMS Alpha**

- » OpenVMS Alpha Version 7.3-2 or higher
- » Software Development Kit (SDK) v 1.4.2-4P2 (or higher) for OpenVMS Alpha  
*for the Java Platform*
- » All required ECOs for the Java SDK
- » HP TCP/IP Services for OpenVMS Version 5.4 ECO 4 or higher
- » A remote file access provider:
  - FTP Service in TCP/IP Services for OpenVMS enabled and started (see Notes)
  - or- Samba for OpenVMS 2.2.8 or higher (unsupported/optional) (see Notes)
  - or- Advanced Server Version 7.3A or higher for OpenVMS (see Notes)
- » (Optional) C/C++, FORTRAN, COBOL, or PASCAL compiler for OpenVMS
- » (Optional) Code Management System (CMS) for OpenVMS

## Software Prerequisites - IDE Server on OpenVMS I64

- » OpenVMS I64 Version 8.2 or higher
- » Software Development Kit (SDK) v 1.4.2-1 (or higher) for OpenVMS I64  
*for the Java Platform*
- » All required ECOs for the Java SDK
- » HP TCP/IP Services for OpenVMS Version 5.5 or higher
- » A remote file access provider:
  - FTP Service in TCP/IP Services for OpenVMS enabled and started (see Notes)
  - or- Samba for OpenVMS 2.2.8 or higher (unsupported/optional) (see Notes)
- » (Optional) C/C++, FORTRAN, COBOL, or PASCAL compiler for OpenVMS
- » (Optional) Code Management System (CMS) for OpenVMS

## Advanced Server Notes

- The TCP/IP PWIP Service must be started for PATHWORKS, if you are using Advanced Server as your remote file access provider. Enter the following command:

```
$ @SYS$MANAGER:TCPIP$CONFIG
  select 4 - optional
  select 1 - PWIP
  select 2 - Enable and Start
```

- The shares that you map to use in Distributed NetBeans remote filesystems should be created in STREAM\_LF format. For example:

```
$ administer add share /dir share_name share_path /host=(rms=stream_lf)
  /permission=(everyone=none,user_name=full)
```

- For better performance, especially when using CMS, disable the Advanced Server's Open File Cache, set the Files Per Client to 25 (or more), and set the Byte Range Locks equal to one half of the value for Files Per Client.

To turn off the open file cache, type the command ADMIN/CONFIG and select the Advanced button. Clear the Use Open File Caching selection, then restart the Advanced Server. For more information about these settings, see *Advanced Server for OpenVMS Server Administrator's Guide*.

## FTP Notes

Before you can use the FTP as your remote file access provider in Distributed NetBeans, the TCP/IP FTP Service must be enabled and started. Enter the following command:

```
$ @SYS$MANAGER:TCPIP$CONFIG
  select 3 - Server components
  select 5 - FTP
  select 2 - Enable and Start
```

See *HP TCP/IP Services for OpenVMS Installation and Configuration* for more information.

## Samba Notes

- The shares that you map to use in Distributed NetBeans remote filesystems should be created in STREAM\_LF format. You specify this parameter in the share section in the Samba configuration file (SMB.CONF). For example:

```
[test1$]
comment = test
valid users = user1
path=efs$:[test]
public = no
writable = yes
vms record format = stream_lf
create mask = 0777
```

- You can download Samba 2.2.8 for OpenVMS from <http://www.pi-net.dyndns.org/anonymous/jyc/>.

## 3 Downloading and Installing Distributed NetBeans

### Downloading the NetBeans Client and IDE Server Kits

Please fill out and submit the Distributed NetBeans for OpenVMS registration form. You will receive an email message with the URL of the location of the NetBeans client and IDE Server kits to download.

Save the IDE Server file HP-xxxVMS-IDESERVER-V0101--1.PCSI\_xxxEXE (where xxx is AXP or I64) to any location accessible to the OpenVMS system on which you will install the IDE Server. If applicable, copy or FTP the file(s) to the OpenVMS system.

Expand the IDE Server file by entering:

```
$ RUN HP-AXPVMS-IDESERVER-V0101--1.PCSI_AXPEXE ! for OpenVMS Alpha
$ RUN HP-I64VMS-IDESERVER-V0101--1.PCSI_I64EXE ! for OpenVMS I64
```

The file expands to HP-xxxVMS-IDESERVER-V0101--1.PCSI\$COMPRESSED.

Save the NetBeans client distnbopenvms36\_v1\_1.nbm to your desktop system.

### Installing the NetBeans Client Kit

The Distributed NetBeans client module contains the following modules, which are automatically installed when you install the Distributed NetBeans client module.

- Distributed OpenVMS Client
- Distributed C/C++ Support for OpenVMS
- Distributed COBOL Support for OpenVMS
- Distributed FORTRAN Support for OpenVMS
- Distributed PASCAL Support for OpenVMS
- Distributed DCL Support for OpenVMS
- EDTkeypad Support Module
- CMS Support Module

These modules can be disabled individually by clicking the Tools menu, then choosing Options, System, and Modules.

You must have NetBeans 3.6 (from NetBeans.org) installed on your desktop system **before** you install the Distributed NetBeans for OpenVMS client.

Install the Distributed NetBeans client .nbm file on your desktop system as follows.

1. Start NetBeans on your desktop system.
2. Open the NetBeans **Update Center** wizard (choose Update Center from the Tools menu).
3. Choose the Install Manually Downloaded Modules option.
4. Follow the instructions on the wizard for completing the installation.
5. After clicking Finish, if you are prompted to restart NetBeans, you must choose Restart the IDE to complete installation of the module.

### Installing the IDE Server Kit

Before you install the IDE Server kit, you must **remove the previous version** (or previous field test kit) if it was installed, by entering the following command:

```
$ PRODUCT REMOVE IDESERVER
```

Wait until all of the IDE\$USER processes have stopped (this can take a minute or so). You can see these processes by entering the following command:

```
$ SHOW SYS/M
```

After the IDE\$USER\_ processes have stopped, install the server .PCSI\$COMPRESSED file by entering the following command from the SYSTEM account or another privileged account. (Do not expand the .PCSI\$COMPRESSED file before installing it; PCSI installs from the compressed kit directly.)

```
$ PRODUCT INSTALL IDESERVER
```

The IDE Server is installed into SYS\$COMMON by default. To specify a different installation location, use the /DESTINATION qualifier on the PRODUCT INSTALL command line, or set the PCSI\$DESTINATION logical name to a different location.

Following is a sample installation of the IDE Server on OpenVMS Alpha.

```
$ product remove ideserver
```

The following product has been selected:

```
HP AXPVMS IDESERVER T1.1-3A          Layered Product
```

Do you want to continue? [YES]

The following product will be removed from destination:

```
HP AXPVMS IDESERVER T1.1-3A          DISK$ALPHASYS:[VMS$COMMON.]
```

Portion done: 0%...10%...30%...40%...50%...60%...70%...80%...90%...100%

The following product has been removed:

HP AXPVMS IDESERVER T1.1-3A

Layered Product

\$ product install ideserver/dest=efs\$:[000000]

The following product has been selected:

HP AXPVMS IDESERVER V1.1 Layered Product

Do you want to continue? [YES]

Configuration phase starting ...

You will be asked to choose options, if any, for each selected product and for any products that may be installed to satisfy software dependency requirements.

HP AXPVMS IDESERVER V1.1: OpenVMS IDE Server

The IDE Server includes the JAR files for Jakarta Ant. You must agree to the terms of the Ant license agreement in order to use the IDE Server.

Press return to display the license

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:  
"This product includes software developed by the  
Apache Software Foundation (<http://www.apache.org/>)."  
Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [apache@apache.org](mailto:apache@apache.org).
5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Portions of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.

I have read and accept the above license.

[Y = I Accept, N = I Reject]: y

\* This product does not have any configuration options.

The installation will now continue without requesting further user input.

Execution phase starting ...

The following product will be installed to destination:

HP AXPVMS IDESERVER V1.1 EFSS:[000000.]

Portion done: 0%...10%...20%...30%...40%...50%...60%...70%...80%...90%

%PCSI-I-PRCOUTPUT, output from subprocess follows ...

%INSTALL-W-NOPREV, no previous entry exists - new entry created for  
OPNAPI\$DKA100:[000000.][IDE\$SERVER.SYSLIB]IDE\$VMS\_AUTH\_MOD.EXE;1

Portion done: 100%

The following product has been installed:

HP AXPVMS IDESERVER V1.1 Layered Product

HP AXPVMS IDESERVER V1.1: OpenVMS IDE Server

Insert the following lines in SYS\$MANAGER:SYSTARTUP\_VMS.COM:

@SYS\$STARTUP:IDE\$STARTUP.COM

Insert the following lines in SYS\$MANAGER:SYSHUTDOWN.COM:

@SYS\$STARTUP:IDE\$SHUTDOWN.COM

ACCOUNT QUOTAS

The IDE Server requires the modifications to system parameters as follows:

System parameter CHANNELCNT greater than or equal to 2000

USER ACCOUNTS that will use Distributed NetBeans will require quotas as follows in order for the IDE\$USER server to start:

WSEXTENT greater than or equal to 30000

JTQUOTA greater than or equal to 60000

PGFLQUO greater than or equal to 500000

Higher values of other quotas may be required depending on the commands you will be executing in your IDE\$USER server.

KEYSTORE CONFIGURATION

The keystore for SSL uses a default account and password. If you would like to modify these values, please see the Release Notes for information on changing the keystore values.

\$

## 4 After Installing Distributed NetBeans

### IDE Server Startup and Shutdown

After you install the IDE Server, add the following line to `SYS$STARTUP:SYSTARTUP_VMS.COM`:

```
$ @SYS$STARTUP:IDE$STARTUP.COM
```

Then add the following line to `SYS$MANAGER:SYSHUTDOWN.COM`:

```
$ @SYS$STARTUP:IDE$SHUTDOWN.COM
```

**Important:** Execute the `IDE$STARTUP.COM` command procedure only from the SYSTEM account.

### Accounts, Quotas, and System Parameters

Following are the quotas for the remote user account. (This is not the `IDE$SERVER` account, but the account that you specify when you create a remote server, called `IDE$USER`, from within the Distributed NetBeans client.)

Default	Quotas assigned when OpenVMS account is created.
Minimum	Minimum value required for Distributed NetBeans to run.
Recommended	Recommended value for optimal performance of Distributed NetBeans.

User Account Quota	Default	Minimum	Recommended
WSDEF	2000	2000	2000
WSEXTENT *	16,384	30,000	30,000
FILLM	100	100	500
BIOLM	150	150	512
DIOLM	150	150	512
ASTLM	250	250	300
TQELM	10	10	400
BYTLM	64,000	64,000	64,000
JTQUOTA *	4096	60,000	60,000
PGFLQUO *	50,000	500,000	500,000

\* Manually increase these quotas.

The IDE Server installation creates an account `IDE$SERVER` and a rights identifier (`IDE$SERVERRI`). The `IDE$SERVER` account is created with the following user quotas:

```
PRIVILEGES=TMPMBX, NETMBX, PRMMBX)
ASTLM=300
BIOLM=1024
BYTLM=2000000
DIOLM=1024
ENQLM=2000
FILLM=512
JTQUOTA=60000
PGFLQUOTA=1500000
PRCLM=10
TQELM=400
WSEXTENT=524288 (SYSGEN parameter WSMAX should be large enough to allow this)
WSDEF=2000
```



## Logical Names

Following are the logical names that can be used to control the IDE Server.

- **IDE\$ALLOW\_EXTAUTH** – This logical causes the IDE Server to ignore the ExtAuth flag and try local authentication using your password from the SYSUAF file.
- **IDE\$JDK\_HOME** – This logical forces a particular version of the Java SDK to be used by pointing to the directory of the Java installation on your OpenVMS system.
- **IDE\$JDK\_VERSION** – This logical forces a particular version of the Java SDK to be used by specifying the version number. For example:

```
$ DEFINE/SYSTEM IDE$JDK_VERSION 142    ! Use Java 1.4.2
$ DEFINE/SYSTEM IDE$JDK_VERSION 150    ! Use Java 1.5.0
```

**Note:** If both logical names are set, IDE\$JDK\_VERSION takes precedence over IDE\$JDK\_HOME. Also, you must restart the IDE Server after changing the IDE\$JDK\_VERSION or IDE\$JDK\_HOME logical names.

- **IDE\$HOST\_IPNAME** – If you have multiple Ethernet cards on your OpenVMS system, this logical allows you to choose which card the IDE Server uses for registration in the Java RMI registry. Set the logical in the SYSTEM table to the IP address or IP hostname of the card you choose. If you have multiple cards and you do not set this logical, the results are unpredictable. For example:

```
$ DEFINE/SYSTEM IDE$HOST_IPNAME buggy.malone.gangsters.com
```

- **IDE\$RMI\_PORT** – This logical defines the RMI port to be used. By default, 1099 is used. If this logical is defined, the value of the logical will be used as the port number. For example, if you enter the following command, the IDE Server will use port 999 for RMI communications:

```
$ DEFINE IDE$RMI_PORT 999
```

- **IDE\$VERBOSE\_LOG** – This logical turns on verbose logging in the IDE Server.

The IDE Server startup procedure creates the following logicals in the system logical name table.

IDE\$ROOT		Top level IDE Server directory (PCSI installation destination)
IDE\$ANT_HOME	IDE\$ROOT:[IDE\$SERVER.ANT]	Distributed Ant related files
IDE\$ANT_ROOT	IDE\$ROOT:[IDE\$SERVER.ANT.]	Root directory of Ant 1.6.2 files
IDE\$CMS	IDE\$ROOT:[IDE\$SERVER.CMS]	Distributed CMS support files
IDE\$COM	IDE\$ROOT:[IDE\$SERVER.COM]	Command procedures
IDE\$DOC	IDE\$ROOT:[IDE\$SERVER.DOC]	Documentation
IDE\$JARS	IDE\$ROOT:[IDE\$SERVER.JARS]	JAR files for the IDE Server
IDE\$JAVA_ROOT	Varies	Root of Java files
IDE\$LIB	IDE\$ROOT:[IDE\$SERVER.SYSLIB]	Contains the shareable images used by the IDE Server
IDE\$LOGS	IDE\$ROOT:[IDE\$SERVER.LOGS]	Log files from the IDE Server processes
IDE\$SCRATCH	IDE\$ROOT:[IDE\$SERVER.SCRATCH]	Scratch directory
IDE\$STORES	IDE\$ROOT:[IDE\$SERVER.STORES]	Keystore and truststore files

The IDE\$SERVER process writes log files into the IDE\$LOGS directory.

## 5 Release Notes

### Default Java SDK Version

The IDE Server requires Java SDK version 1.4.2-4 (or higher) on OpenVMS Alpha or v 1.4.2-1 (or higher) on OpenVMS I64. By default, the IDE Server assumes that Java v 1.4.2 is installed on the system, and the server attempts to use that version of the SDK.

To force the server to use a particular version of the SDK, define the logical `IDE$JDK_VERSION` to a three-digit version number of the SDK you want to use (for example, 142 or 150). You can also define `IDE$JDK_HOME` to point to the top level directory that contains the OpenVMS SDK distribution you want to use. If you define both logical names, `IDE$JDK_VERSION` takes precedence.

For example, entering **either** of the following commands causes the IDE Server to be started using the SDK (in this case, v 1.5.0) contained in the tree whose root is `SYS$COMMON:[JAVA$150]`:

```
$ DEFINE IDE$JDK_VERSION 150
$ DEFINE IDE$JDK_HOME SYS$COMMON:[JAVA$150]
```

User processes that are started by the IDE Server also use these logicals to choose the Java version. You must restart the IDE Server after changing the `IDE$JDK_VERSION` or `IDE$JDK_HOME` logical names.

### Define IDE\$ALLOW\_EXTAUTH Logical for Users with ExtAuth Flag Set in SYSUAF Record

The IDE Server does not support external authentication. If you have the `ExtAuth` flag set in your `SYSUAF` user record, you must define the system logical name `IDE$ALLOW_EXTAUTH` to cause the IDE Server to ignore the `ExtAuth` flag and try local authentication using your password from the `SYSUAF` file (these are usually synchronized with Advanced Server).

If the logical is defined (with any value) then the check for the `ExtAuth` flag is disabled. If the logical is not defined, then the `SYSUAF` record is checked for the `ExtAuth` flag and the Remote Server connection will fail from within the Distributed NetBeans client.

HP recommends that you define the logical in the `SYSTEM` table to ensure that it is visible to the IDE Server process.

### JSSE Keystore and Truststore

The IDE server uses JSSE (Java Secure Socket Extension) for secure network connections. Configuring the JSSE's Keystore and Truststore for the IDE server requires running `IDE$STORES:IDE$CONFIG.EXE`. It will encrypt the user provided information and store it as `IDE$STORES:IDE$KEYDATA`.

The IDE server is shipped with a preconfigured `IDE$STORES:IDE$KEYDATA` which assumes `IDE$STORES:IDE$_KEYSTORE` and `IDE$STORES:IDE$_TRUSTSTORE` as the Keystore and Truststore, respectively.

`IDE$STORES:IDE$_KEYSTORE` and `IDE$STORES:IDE$_TRUSTSTORE` were created with JDK's `keytool` and preconfigured to use with the IDE server. The password to access them is **`_keystore`**.

## IDE\$SERVER and IDE\$USER Processes on OpenVMS

At system startup, the `SY$STARTUP:IDE$STARTUP.COM` command procedure creates a process named `IDE$SERVER`, which runs in the `IDE$SERVER` account. The `IDE$SERVER` process starts a subprocess named `IDE$RMIREG` for handling RMI. Both of these processes are detached processes.

When you create a remote server inside NetBeans running on your desktop, the NetBeans client contacts the `IDE$SERVER` process running on the OpenVMS machine. The `IDE$SERVER` process authenticates the user account login information you supplied, and creates a detached process running in your user account on the OpenVMS machine. This process is named `IDE$USER_XXXXXX`, where `XXXXXX` is a unique identifier. The `IDE$USER_XXXXXX` process is then fed DCL commands by the NetBeans client running on your desktop.

Because the `IDE$USER` process is running detached, it may not obtain all of the symbol and logical name definitions that you defined in your `SYLOGIN.COM` and `LOGIN.COM` files. A detached process has an `F$MODE()` of `OTHER`. `SYLOGIN.COM` has sections for each different process mode, and a `GOTO` at the beginning of the file. When a process has an `F$MODE()` of `OTHER`, the `GOTO` may bypass some important DCL commands. This is also a common DCL programming practice in `LOGIN.COM`.

HP recommends that you examine your `SYLOGIN.COM` and `LOGIN.COM` procedures to make sure that detached processes obtain the proper symbol and logical name definitions.

## Files Edited in NetBeans May Require STREAM\_LF Record Format

`STREAM_LF` RMS file format is required if you are using an SMB-based filesystem (Advanced Server or Samba). `STREAM_LF` is **not** required if you are using an FTP filesystem.

Distributed NetBeans automatically checks the record format of each file from an SMB share that you attempt to open in the editor, and asks if you want to convert the file to `STREAM_LF`. (Converting files to a record format of `STREAM_LF` does not alter the contents of the file in any way.) If you choose not to convert the file, the file is not opened in the NetBeans editor. (Using the NetBeans editor to open a non-`STREAM_LF` file that resides on an SMB filesystem will corrupt the file.)

The Java SDK contains an FDL file that you can use to **manually** convert text files of different formats to `STREAM_LF` record format. This FDL file is located in the `[.COM]` directory of the Java distribution. You perform this conversion only once per file. The following shows the location of the FDL file in the Java SDK v 1.4.2 distribution:

```
Directory SYS$COMMON:[JAVA$142.COM]

STREAM_LF.FDL;1      1  26-SEP-2000 16:05:06.22  (RWED,RWED,RE,RE)
```

Use the following command to convert a file (`MYFILE.C`) to `STREAM_LF` record format:

```
$ CONVERT/FDL=SYS$COMMON:[JAVA$142.COM]STREAM_LF.FDL MYFILE.C MYFILE.C
```

If you store the file in a CMS library, reserve the file, change the record format to `STREAM_LF`, then replace the file back into the CMS library. This ensures that subsequent fetches result in a file with the proper record format.

## Remote Command Execution and Privileges

Remote commands are not executed in a process that has the AUTHORIZED privilege mask from the remote user server.

The `IDE$SERVER` starts a detached user server process logged in to the account you specify when you create the remote server in NetBeans on your Distributed NetBeans desktop machine. This detached process spawns subprocesses to execute commands (at your control) on the OpenVMS machine. The spawned subprocess is given only the privileges that were enabled when the user server spawned the subprocess.

If you need extra privileges to execute commands in a DCL file, you must enable them in your user server process. You can do this by modifying your `LOGIN.COM` file to turn on the required privileges. You can restrict this action by turning on privileges only if your process name begins with `IDE$USER_` or by checking the `f$mode()` of the process. (All Distributed NetBeans servers are detached, so the `f$mode() == "OTHER".`)

## Using Ant Outside of Distributed NetBeans

Apache Ant version 1.6.2 is included with the IDE Server. This version of Ant has been patched to work on OpenVMS.

The logical `IDE$ANT_HOME` points to the root directory of the Ant files for OpenVMS. This directory contains the command procedure `IDE_ANT.COM`. If you define a symbol that points to this command procedure, you can issue Ant commands to DCL.

For example:

```
$! First, set up Java
$!
$ @SYS$MANAGER:JAVA$142_SETUP FAST
$!
$! Now define our symbol
$!
$ Ant ::= "@IDE$ANT_HOME:IDE_ANT.COM"
$!
$! Now test...
$!
$ ant "-version"
Apache Ant version 1.6.2 compiled on March 7 2005
$
```

## 6 Known Problems and Restrictions in the IDE Server and Distributed NetBeans Client

Following are the known problems and restrictions in Distributed NetBeans Version 1.1.

- Remove Previous Version or Field Test Kit Before Installing Version 1.1

Manually remove the previous version of the IDE Server before you install a new kit. This ensures that all old files are completely removed and the new kit is installed properly.

- Unmount and Remount Remote Filesystems After Upgrade

If you are upgrading from Distributed NetBeans Version 1.0 to 1.1, after you complete the upgrade installation procedure you should unmount and remount your remote filesystems. This will allow the Remote Filesystems wizard to verify that your filesystems are in the proper format.

- Remotely Debugging Java Programs

Debugging Java programs remotely does not work with the Java SDK 1.5 Beta 1 on OpenVMS Alpha. This problem will be corrected in a future Java SDK 1.5 release.

- Multiple Ethernet Cards on IDE Server

If you have multiple Ethernet cards on your OpenVMS system, you must choose which card the IDE Server uses for registration in the Java RMI registry. Set the IDE\$HOST\_IPNAME logical in the SYSTEM table to the IP address or IP hostname of the card you choose. For example:

```
$ DEFINE/SYSTEM IDE$HOST_IPNAME buggy.malone.gangsters.com
```

If you have multiple cards and you do not set this logical, the results are unpredictable.

- Multiple Ethernet Cards on Desktop Machine

Having more than one IP address or hostname on your desktop machine can cause the connection to the IDE Server to fail. To solve this problem, choose a host name or IP address for your client machine and set the RMI hostname property on the command line used to launch NetBeans as follows:

```
-J-Djava.rmi.server.hostname=<host name or ip address>
```

- Client Tested on Windows Only

The Distributed NetBeans client has been tested on Windows only. However, HP believes that the client software will run on other desktop platforms, such as Linux and HP-UX.

- Router Using NAT Times Out

If the Distributed NetBeans client initiates the connection to the remote server via a router that uses Network Address Translation (NAT), the connection attempt will time out. This happens because the Java RMI implementation obtains the virtual IP address (private IP address) of the client from the IP packet body, and not from the actual address in the IP packet header. There is currently no workaround for this problem.

- C/C++ Error Marking

Errors in C/C++ source files are marked with red underlining and a red x in the margin. These error markings are cleared only if you modify the line they mark.

- Process Termination from Runtime Tab

Explicitly terminating a process from the NetBeans Runtime tab sometimes does not stop the

corresponding remote process on the server.

- Open Files in Editor Delays Subsequent Startup

If you exit NetBeans on your desktop with several files from the remote OpenVMS system open in the editor, your startup time when you next invoke NetBeans will be longer than usual. At startup, NetBeans restores the state of the previous editing session, which includes opening all of the files that were open when you last exited NetBeans.

## 7 Software Support

### On OpenVMS Alpha:

- Support for Distributed NetBeans is provided by HP under OpenVMS Alpha service agreements through standard support channels.
- Download is an official distribution mechanism for Distributed NetBeans, including updates in addition to the OpenVMS Alpha media kit.
- Source code kits are provided in the spirit of the open source community, but are not supported by HP.

### On OpenVMS I64:

- Support for Distributed NetBeans is provided by HP under OpenVMS I64 Foundation Operating Environment (FOE) service agreements through standard support channels.
- Download is a courtesy distribution mechanism Distributed NetBeans. Kits or updates other than security patch kits obtained via download are unsupported.
- The OpenVMS I64 Operating Environments media kit is the official distribution mechanism for Distributed NetBeans. Supported kits and updates must be obtained from this mechanism or from software product update services.

A variety of service options are available from HP Support. For more information, contact your local HP account representative or distributor. Information is also available from Software Support Services.

General NetBeans support is provided by the NetBeans user group at <http://blog.gmane.org/gmane.comp.java.ide.netbeans.user>.

You can informally exchange information with other users in the OpenVMS newsgroup `comp.os.vms`.

To contact the NetBeans for OpenVMS engineering team, please send mail to **NetBeans@hp.com**.