Build Overview

Solaris Build Instructions

Java™ 2 Platform Standard Edition, v5.0 fcs

Contents

Introduction

This README file contains build instructions for the Java™ 2 Platform Standard Edition, v5.0 (JDK 5.0) Community Source Release. Building the source code for the JDK requires a high level of technical expertise. Sun provides the source code primarily for technical experts who want to conduct research, port the platform to new operating systems and hardware, or add enhancements that require access to platform internals. If you are not a technical professional in one of these categories, this release is probably not for you.

Solaris System Setup

The official build platform for the 32-bit version of JDK 5.0 is Solaris 8.

The minimum recommended hardware for building the Solaris-SPARC version is an UltraSPARC with 512 MB of RAM. For building the Solaris-x86 version, a Pentium class processor or better and at least 128 MB of RAM are recommended. Approximately 1.4 GB of free disk space is needed for a 32-bit build.

64-BIT-ONLY: The official build platform for the 64-bit version of JDK 5.0 is a 64-bit installation of Solaris 8 on SPARC. You may run the command "isainfo -v" to verify that you have a 64-bit installation. An additional 7 GB of free disk space is needed for a 64-bit build.

The build uses the tools contained in `/usr/ccs/bin` and `/usr/bin` of a standard developer or full installation of the Solaris operating environment.

Minimum patch revisions are given in the tables below, though later patch revisions are expected to work also. Patches may be downloaded from the JDK download page. You should ensure that the latest patch cluster for your version of the Solaris operating environment has been installed prior to installing these patches. The JDK patch clusters are available for download on the SunSolve web site.

The term "required" means the patch is required for non-international build.

Patches for Building on Solaris 5.8

SPARC patch	x86 patch	Req/Opt	Description
109147-24	109148-24	Required	Linker patch
108652-66	108653-55	Required	Xserver patch
108940-52	108941-52	Required	Motif 2.1 patch
108989-02	108990-02	Required	Accounting patch
none	111307-04	Required	boot.bin, bootconf.exe, bootenv.rc and nbp patch
111310-01	111311-01	Required	libhcpagent.so.l patch
112396-02	112397-02	Required	fgrep patch
108987-13	108988-13	Required	patchadd, patchrm patch
111111-03	111112-03	Required	nawk patch
108528-20	108529-20	Required	Kernel update
108993-18	none	Required	LDAP2 Patch
none	110400-01	Required	RBAC Feature patch
none	111024-02	Required	/kernel/fs/mntfs patch
none	108994-18	Required	LDAP2 patch
109147-23	109148-23	Required	linker patch
111308-03	111309-03	Required	Performance for apps using memory alloc

Solaris Build Tools and Libraries Setup

Once the Solaris OS is set up, you will need to install additional tools and libraries for building the JDK.

Sun ONE Studio Compilers

Sun ONE Studio 8, Compiler Collection (containing version 5.5 of the C and C++ compilers) is required, with patches as defined below. You may download these compilers with a free 30-day "Try and Buy" license. A permanent license may be obtained from the Sun ONE Studio Developer Tools web site.

Set `ALT_COMPILER_PATH` to point to the location of the compiler binaries, and place this location in the `PATH`. These patches are available for download on the SunSolve web site.

Compiler Patches for Building on Solaris

SPARC patch	x86 patch	Req/Opt	Description
109505-06	109502-03	Required	For C 5.0, C++ 5.0
109513-05	109514-03	Required	For Forte Development 6 C compiler
109508-03	109509-03	Required	For Forte Development 6 update 1 C++ compiler
109510-03	109511-03	Required	For Forte 6.1 Debugger
109516-02	109517-02	Required	For Forte 6.1 Performance Analyzer
110480-01	110481-01	Required	For Forte TeamWare

Bootstrap JDK

You will need access to a JDK 1.4.2 for Solaris installation. The 1.4.2 binaries can be downloaded from Sun's JDK 1.4.2 download site. Set `ALT_BOOTDIR` to point to the location of the bootstrap JDK installation. The installation instructions for the bootstrap JDK include a list of required and recommended patches that are needed at runtime. The subset of these patches that are required for building JDK are listed in the System Setup section above. However, all the runtime patches should be installed to ensure proper behavior of all JDK functionality after the build is completed.

GNU Make

GNU make version 3.78.1 or later is required to build the JDK. Information on GNU make, including download sites, is available on the GNU Make web site. For convenience, place the GNU make binary in the `PATH`.

Motif 2.1

Motif version 2.1 headers and libraries are required for building the JDK.

Create a Motif library and header area that contains header files and libraries for Motif 2.1. Use the `ALT_MOTIF_DIR` environment variable to point to absolute path of the Motif directory. The top level of the directory must contain directory named `motif21`, which must have subdirectories `include` and `lib` with contents as shown here:

```
+- motif_area/  (set ALT_MOTIF_DIR to this level)
   +- motif21/
      +- include/
      |  +- Xm/  (from /usr/include/Xm)
      |     +- <many files>
      |
      +- lib/    (from /usr/dt/lib/)
         +- libXm.so (symbolic link to libXm.so.4)
         +- libXm.so.4
         +- sparcv9/    (64-bit Motif library)
            +- libXm.so (symbolic link to libXm.so.4)
            +- libXm.so.4
```

In the example above, the name of the top-level directory is not significant; it is not required to be named `motif_area`.

JDBC-ODBC Bridge

The DataDirect Connect ODBC 2.11 driver is needed for building the JDBC-ODBC Bridge. A copy of the driver is on the Desktop 1.1.1 Solaris CD-ROM, which is part of older Solaris distributions.

Set up the following directory structure for the ODBC driver, and set the `ALT_ODBCDIR` environment variable to point to it.

```
+- odbc/           (set ALT_ODBCDIR to this level)
   +- ISLIodbc/
      +- 2.11/
         +- odbc files and directories (lib/, include/, etc.)
```

On SPARC systems you may use the DataDirect ODBC 3.7 driver in place of the 2.11 driver: use the directory structure above (including the `2.11` directory) if doing so.

An alternative to using a DataDirect driver is to build a dummy driver of your own. Create and then "cd" to the directory `$ALT_ODBCDIR/ISLIodbc/2.11/lib`, copy over the source file `j2se/make/sun/jdbc/dummyodbc.c`, and then compile as follows using the Sun ONE Studio 8 C compiler:

```
cc -G -h libodbc.so -o libodbc.so dummyodbc.c
cc -G -h libodbcinst.so -o libodbcinst.so dummyodbc.c
```

zip

The build requires `zip` version 2.2 (November 3rd 1997) or later. Set `ALT_DEVTOOLS_PATH` to point to the location of this binary. Information on zip, including download sites, is available on the info-zip web site.

cacerts

A certificates file named "cacerts" represents a system-wide keystore with CA certificates. In JDK and JRE binary bundles, the "cacerts" file contains root CA certificates from several public CAs (e.g., VeriSign, Thawte, and Baltimore).

The source bundles contain a cacerts file without CA root certificates. JDK builders will need to secure permission from each public CA and include the certificates into their own custom cacerts file. Failure to provide a populated cacerts file will result in verification of a certificate chain during runtime.

The `ALT_CACERTS_FILE` should be set to point to the location of the populated cacerts file.

GNU C Compiler (32-bit build only)

GNU gcc version 2.95.2 is required for building the Plug-in. The source code for gcc is available from http://www.gnu.org/software/gcc/, and some pre-built binaries are available from sunfreeware.com. Set `ALT_GCC_COMPILER_PATH` to point to the location of the gcc binary.

Mozilla Headers (32-bit build only)

Mozilla headers are required for building Java Plug-in. Download and unpack the headers into a directory similar to the one shown below, and set the `ALT_MOZILLA_PATH` environment variable to the absolute path of the top-level directory.

```
+- devtools/          (set ALT_MOZILLA_PATH to this level)
   +- share/
      +- plugin/
         +- mozilla_headers_ns7/
```

The name of the top-level directory is not significant; it is not required to be named `devtools`.

Solaris Build Environment Variables

This section describes environment variables that you can set to influence various aspects of the build. Some of these variables are mentioned specifically in the setup and build instructions above. Others you may set at your discretion.

Environment variables may be set in the shell environment or on the GNU make command line.

`PATH`
: Set the `PATH` to contain:
  - The location of the GNU make binary
  - The location of the Sun ONE Studio 8 Compilers (`ALT_COMPILER_PATH`)
  - `/usr/bin`

`ALT_BOOTDIR`
: The location of the JDK 1.4.2 bootstrap installation.

`ALT_OUTPUTDIR`
: An override for specifying the (absolute) path of where the build output is to go.

`ALT_UNIXCOMMAND_PATH`
: An override for specifying where the Unix command set are located. You usually do not need to set this variable: the default location is `/usr/bin`.

`ALT_UNIXCCS_PATH`
: An override for specifying where the Unix CCS command set are located. You usually do not need to set this variable: the default location is `/usr/ccs/bin`

`ALT_COMPILER_PATH`
: The location of the Sun ONE Studio 8 compilers and tools. See Sun ONE Studio Compilers for details.

`ALT_DEVTOOLS_PATH`
: The location of the `zip` binary. See zip for details.

`ALT_CACERTS_FILE`
: The location of the `cacerts` file. See cacerts file for details.

`ALT_MOTIF_DIR`
: The location of the Motif 2.1 headers and libraries. See Motif 2.1 for details.

`ALT_MOZILLA_PATH`
: The location of the Mozilla headers. See Mozilla Headers for details.

`ALT_ODBCDIR`
: The location of the ODBC driver. See JDBC-ODBC Bridge for details.

`ALT_GCC_COMPILER_PATH` (32-bit build only)
: The location of the GNU C compiler and tools, for building the Plug-in. See GNU C Compiler for details.

`ARCH_DATA_MODEL`
: The `ARCH_DATA_MODEL` variable is used to specify whether the build is to generate 32-bit or 64-bit binaries. The Solaris build supports either 32-bit or, on SPARC platforms only, 64-bit builds. Leave `ARCH_DATA_MODEL` unset or set to `32` for generating 32-bit binaries, or set to `64` for generating 64-bit binaries.

`MILESTONE`
: The milestone name for the build (*e.g.* "beta").

`BUILD_NUMBER`
: The build number for the build (*e.g.* "b27").

Solaris Build

1. cd into the `control/make` directory.
2. Start the build with the command:

```
make scsl [ARCH_DATA_MODEL=32 or 64] [ALT_OUTPUTDIR=directory-name-for-output] [MILESTONE=milestone_name]
[BUILD_NUMBER=build_number] [other "ALT_" overrides]
```

Please be sure to use the GNU version of `make`.

64-BIT-ONLY: Before 64-bit binaries can be used, they must be merged with the binaries from a separate 32-bit build. The merged binaries may then be used in either 32-bit or 64-bit mode, with the selection occurring at runtime.
