## "private native" \*; 152

com.sun.imageio.plugins.jpeg.JPEGImageReader(238): private native long initJPEGImageReader();

com.sun.imageio.plugins.jpeg.JPEGImageReader(335): private native void setSource(long structPointer);

com.sun.imageio.plugins.jpeg.JPEGImageReader(638): private native boolean readImageHeader(long structPointer,

com.sun.imageio.plugins.jpeg.JPEGImageReader(997): private native void setOutColorSpace(long structPointer, int id);

com.sun.imageio.plugins.jpeg.JPEGImageReader(1410): private native boolean readImage(int imageIndex,

com.sun.imageio.plugins.jpeg.JPEGImageReader(1442): private native void abortRead(long structPointer);

com.sun.imageio.plugins.jpeg.JPEGImageReader(1445): private native void resetLibraryState(long structPointer);

com.sun.imageio.plugins.jpeg.JPEGImageReader(1608): private native void resetReader(long structPointer);

com.sun.imageio.plugins.jpeg.JPEGImageWriter(1695): private native long initJPEGImageWriter();

com.sun.imageio.plugins.jpeg.JPEGImageWriter(1698): private native void setDest(long structPointer);

com.sun.imageio.plugins.jpeg.JPEGImageWriter(1703): private native boolean writeImage(long structPointer,

com.sun.imageio.plugins.jpeg.JPEGImageWriter(1760): private native void writeTables(long structPointer,

com.sun.imageio.plugins.jpeg.JPEGImageWriter(1833): private native void abortWrite(long structPointer);

com.sun.imageio.plugins.jpeg.JPEGImageWriter(1836): private native void resetWriter(long structPointer);

com.sun.swing.plaf.gtk.GTKEngine(121): private native void native\_paint\_arrow(

com.sun.swing.plaf.gtk.GTKEngine(124): private native void native\_paint\_box(

com.sun.swing.plaf.gtk.GTKEngine(127): private native void native\_paint\_box\_gap(

com.sun.swing.plaf.gtk.GTKEngine(131): private native void native\_paint\_check(

com.sun.swing.plaf.gtk.GTKEngine(134): private native void native\_paint\_expander(

com.sun.swing.plaf.gtk.GTKEngine(137): private native void native\_paint\_extension(

com.sun.swing.plaf.gtk.GTKEngine(140): private native void native\_paint\_flat\_box(

com.sun.swing.plaf.gtk.GTKEngine(143): private native void native\_paint\_focus(

com.sun.swing.plaf.gtk.GTKEngine(146): private native void native\_paint\_handle(

com.sun.swing.plaf.gtk.GTKEngine(149): private native void native\_paint\_hline(

com.sun.swing.plaf.gtk.GTKEngine(152): private native void native\_paint\_option(

com.sun.swing.plaf.gtk.GTKEngine(155): private native void native\_paint\_shadow(

com.sun.swing.plaf.gtk.GTKEngine(158): private native void native\_paint\_slider(

com.sun.swing.plaf.gtk.GTKEngine(161): private native void native\_paint\_vline(

com.sun.swing.plaf.gtk.GTKEngine(164): private native void native\_paint\_background(

com.sun.swing.plaf.gtk.GTKEngine(166): private native Object native\_get\_gtk\_setting(int property);

com.sun.swing.plaf.gtk.GTKEngine(167): private native void nativeSetRangeValue(int widgetType, double value,

com.sun.swing.plaf.gtk.GTKEngine(171): private native void nativeStartPainting(int w, int h);

com.sun.swing.plaf.gtk.GTKEngine(172): private native int nativeFinishPainting(int[] buffer, int width, int height);

com.sun.swing.plaf.gtk.GTKEngine(173): private native void native\_switch\_theme();

com.sun.security.auth.module.NTSystem(36): private native void getCurrent(boolean debug);

com.sun.security.auth.module.NTSystem(37): private native long getImpersonationToken0();

com.sun.security.auth.module.SolarisSystem(36): private native void getSolarisInfo();

com.sun.security.auth.module.UnixSystem(36): private native void getUnixInfo();

java.awt.AWTEvent(387): private native void nativeSetSource(ComponentPeer peer);

java.awt.Cursor(474): private native static void finalizeImpl(long pData);

java.awt.SplashScreen(413): private native static void \_update(long splashPtr, int[] data, int x, int y, int width, int height, int scanlineStride);

java.awt.SplashScreen(414): private native static boolean \_isVisible(long splashPtr);

java.awt.SplashScreen(415): private native static Rectangle \_getBounds(long splashPtr);

java.awt.SplashScreen(416): private native static long \_getInstance();

java.awt.SplashScreen(417): private native static void \_close(long splashPtr);

java.awt.SplashScreen(418): private native static String \_getImageFileName(long splashPtr);

java.awt.SplashScreen(419): private native static String \_getImageJarName(long SplashPtr);

java.awt.SplashScreen(420): private native static boolean \_setImageData(long SplashPtr, byte[] data);

java.awt.SplashScreen(421): private native static float \_getScaleFactor(long SplashPtr);

java.awt.image.BufferedImage(284): static private native void initIDs();

java.awt.image.ComponentSampleModel(107): static private native void initIDs();

java.awt.image.IndexColorModel(138): static private native void initIDs();

java.awt.image.Raster(170): static private native void initIDs();

java.awt.image.SampleModel(102): static private native void initIDs();

java.io.Console(556): private native static boolean istty();

java.io.FileInputStream(187): private native void open0(String name) throws FileNotFoundException;

java.io.FileInputStream(210): private native int read0() throws IOException;

java.io.FileInputStream(219): private native int readBytes(byte b[], int off, int len) throws IOException;

java.io.FileInputStream(377): private native void close0() throws IOException;

java.io.FileOutputStream(259): private native void open0(String name, boolean append)

java.io.FileOutputStream(280): private native void write(int b, boolean append) throws IOException;

java.io.FileOutputStream(302): private native void writeBytes(byte b[], int off, int len, boolean append)

java.io.FileOutputStream(427): private native void close0() throws IOException;

java.io.ObjectStreamClass(1864): private native static boolean hasStaticInitializer(Class<?> cl);

java.io.RandomAccessFile(299): private native void open0(String name, int mode)

java.io.RandomAccessFile(340): private native int read0() throws IOException;

java.io.RandomAccessFile(349): private native int readBytes(byte b[], int off, int len) throws IOException;

java.io.RandomAccessFile(492): private native void write0(int b) throws IOException;

java.io.RandomAccessFile(502): private native void writeBytes(byte b[], int off, int len) throws IOException;

java.io.RandomAccessFile(561): private native void seek0(long pos) throws IOException;

java.io.RandomAccessFile(1148): private native void close0() throws IOException;

java.io.WinNTFileSystem(362): private native String getDriveDirectory(int drive);

java.io.WinNTFileSystem(445): private native String canonicalize0(String path)

java.io.WinNTFileSystem(458): private native String canonicalizeWithPrefix0(String canonicalPrefix,

java.io.WinNTFileSystem(566): private native boolean delete0(File f);

java.io.WinNTFileSystem(580): private native boolean rename0(File f1, File f2);

java.io.WinNTFileSystem(628): private native long getSpace0(File f, int t);

java.lang.Class(648): private native String getName0();

java.lang.Class(860): private native Class<?>[] getInterfaces0();

java.lang.Class(1069): private native Object[] getEnclosingMethod0();

java.lang.Class(1243): private native Class<?> getDeclaringClass0();

java.lang.Class(2319): private native java.security.ProtectionDomain getProtectionDomain0();

java.lang.Class(2528): private native String getGenericSignature0();

java.lang.Class(3138): private native Field[] getDeclaredFields0(boolean publicOnly);

java.lang.Class(3139): private native Method[] getDeclaredMethods0(boolean publicOnly);

java.lang.Class(3140): private native Constructor<T>[] getDeclaredConstructors0(boolean publicOnly);

java.lang.Class(3141): private native Class<?>[] getDeclaredClasses0();

java.lang.ClassLoader(857): private native Class<?> defineClass0(String name, byte[] b, int off, int len,

java.lang.ClassLoader(860): private native Class<?> defineClass1(String name, byte[] b, int off, int len,

java.lang.ClassLoader(863): private native Class<?> defineClass2(String name, java.nio.ByteBuffer b,

java.lang.ClassLoader(967): private native void resolveClass0(Class<?> c);

java.lang.ClassLoader(1019): private native Class<?> findBootstrapClass(String name);

java.lang.ClassLoader(1041): private native final Class<?> findLoadedClass0(String name);

java.lang.SecurityManager(363): private native ClassLoader currentClassLoader0();

java.lang.SecurityManager(472): private native int classLoaderDepth0();

java.lang.SecurityManager(1762): private native Class<?> currentLoadedClass0();

java.lang.Thread(731): private native void start0();

java.lang.Thread(972): private native boolean isInterrupted(boolean ClearInterrupted);

java.lang.Thread(1692): private native static StackTraceElement[][] dumpThreads(Thread[] threads);

java.lang.Thread(1693): private native static Thread[] getThreads();

java.lang.Thread(2044): private native void setPriority0(int newPriority);

java.lang.Thread(2045): private native void stop0(Object o);

java.lang.Thread(2046): private native void suspend0();

java.lang.Thread(2047): private native void resume0();

java.lang.Thread(2048): private native void interrupt0();

java.lang.Thread(2049): private native void setNativeName(String name);

java.lang.Throwable(789): private native Throwable fillInStackTrace(int dummy);

java.lang.reflect.Executable(437): private native Parameter[] getParameters0();

java.lang.reflect.Field(1159): private native byte[] getTypeAnnotationBytes0();

java.net.DatagramPacket(387): private native static void init();

java.net.Inet4AddressImpl(38): private native boolean isReachable0(byte[] addr, int timeout, byte[] ifaddr, int ttl) throws IOException;

java.net.Inet6AddressImpl(44): private native boolean isReachable0(byte[] addr, int scope, int timeout, byte[] inf, int ttl, int if\_scope) throws IOException;

java.net.NetworkInterface(366): private native static NetworkInterface[] getAll()

java.net.NetworkInterface(369): private native static NetworkInterface getByName0(String name)

java.net.NetworkInterface(372): private native static NetworkInterface getByIndex0(int index)

java.net.NetworkInterface(375): private native static NetworkInterface getByInetAddress0(InetAddress addr)

java.net.NetworkInterface(491): private native static boolean isUp0(String name, int ind) throws SocketException;

java.net.NetworkInterface(492): private native static boolean isLoopback0(String name, int ind) throws SocketException;

java.net.NetworkInterface(493): private native static boolean supportsMulticast0(String name, int ind) throws SocketException;

java.net.NetworkInterface(494): private native static boolean isP2P0(String name, int ind) throws SocketException;

java.net.NetworkInterface(495): private native static byte[] getMacAddr0(byte[] inAddr, String name, int ind) throws SocketException;

java.net.NetworkInterface(496): private native static int getMTU0(String name, int ind) throws SocketException;

java.net.SocketInputStream(94): private native int socketRead0(FileDescriptor fd,

java.net.SocketInputStream(291): private native static void init();

java.net.SocketOutputStream(87): private native void socketWrite0(FileDescriptor fd, byte[] b, int off,

java.net.SocketOutputStream(183): private native static void init();

java.net.TwoStacksPlainDatagramSocketImpl(215): private native static void init();

java.nio.MappedByteBuffer(208): private native boolean isLoaded0(long address, long length, int pageCount);

java.nio.MappedByteBuffer(209): private native void load0(long address, long length);

java.nio.MappedByteBuffer(210): private native void force0(FileDescriptor fd, long address, long length);

java.util.jar.JarFile(207): private native String[] getMetaInfEntryNames();

java.util.zip.Adler32(137): private native static int update(int adler, int b);

java.util.zip.Adler32(138): private native static int updateBytes(int adler, byte[] b, int off,

java.util.zip.Adler32(140): private native static int updateByteBuffer(int adler, long addr,

java.util.zip.CRC32(134): private native static int update(int crc, int b);

java.util.zip.CRC32(135): private native static int updateBytes(int crc, byte[] b, int off, int len);

java.util.zip.CRC32(137): private native static int updateByteBuffer(int adler, long addr,

java.util.zip.Deflater(563): private native static long init(int level, int strategy, boolean nowrap);

java.util.zip.Deflater(564): private native static void setDictionary(long addr, byte[] b, int off, int len);

java.util.zip.Deflater(565): private native int deflateBytes(long addr, byte[] b, int off, int len,

java.util.zip.Deflater(567): private native static int getAdler(long addr);

java.util.zip.Deflater(568): private native static void reset(long addr);

java.util.zip.Deflater(569): private native static void end(long addr);

java.util.zip.Inflater(398): private native static void initIDs();

java.util.zip.Inflater(399): private native static long init(boolean nowrap);

java.util.zip.Inflater(400): private native static void setDictionary(long addr, byte[] b, int off,

java.util.zip.Inflater(402): private native int inflateBytes(long addr, byte[] b, int off, int len)

java.util.zip.Inflater(404): private native static int getAdler(long addr);

java.util.zip.Inflater(405): private native static void reset(long addr);

java.util.zip.Inflater(406): private native static void end(long addr);

## "public native" 42

java.io.FileDescriptor(154): public native void sync() throws SyncFailedException;

java.io.FileInputStream(282): public native long skip(long n) throws IOException;

java.io.FileInputStream(301): public native int available() throws IOException;

java.io.RandomAccessFile(537): public native long getFilePointer() throws IOException;

java.io.RandomAccessFile(569): public native long length() throws IOException;

java.io.RandomAccessFile(590): public native void setLength(long newLength) throws IOException;

java.io.WinNTFileSystem(521): public native int getBooleanAttributes(File f);

java.io.WinNTFileSystem(524): public native boolean checkAccess(File f, int access);

java.io.WinNTFileSystem(527): public native long getLastModifiedTime(File f);

java.io.WinNTFileSystem(530): public native long getLength(File f);

java.io.WinNTFileSystem(533): public native boolean setPermission(File f, int access, boolean enable,

java.io.WinNTFileSystem(539): public native boolean createFileExclusively(String path)

java.io.WinNTFileSystem(543): public native String[] list(File f);

java.io.WinNTFileSystem(546): public native boolean createDirectory(File f);

java.io.WinNTFileSystem(549): public native boolean setLastModifiedTime(File f, long time);

java.io.WinNTFileSystem(552): public native boolean setReadOnly(File f);

java.lang.Class(483): public native boolean isInstance(Object obj);

java.lang.Class(510): public native boolean isAssignableFrom(Class<?> cls);

java.lang.Class(520): public native boolean isInterface();

java.lang.Class(530): public native boolean isArray();

java.lang.Class(561): public native boolean isPrimitive();

java.lang.Class(732): public native Class<? super T> getSuperclass();

java.lang.Class(927): public native Class<?> getComponentType();

java.lang.Class(957): public native int getModifiers();

java.lang.Class(968): public native Object[] getSigners();

java.lang.ClassLoader(1731): public NativeLibrary(Class<?> fromClass, String name, boolean isBuiltin) {

java.lang.Object(100): public native int hashCode();

java.lang.Runtime(635): public native int availableProcessors();

java.lang.Runtime(646): public native long freeMemory();

java.lang.Runtime(659): public native long totalMemory();

java.lang.Runtime(670): public native long maxMemory();

java.lang.Runtime(688): public native void gc();

java.lang.Runtime(733): public native void traceInstructions(boolean on);

java.lang.Runtime(751): public native void traceMethodCalls(boolean on);

java.lang.String(3167): public native String intern();

java.lang.Thread(1218): public native int countStackFrames();

java.net.Inet4AddressImpl(34): public native String getLocalHostName() throws UnknownHostException;

java.net.Inet4AddressImpl(35): public native InetAddress[]

java.net.Inet4AddressImpl(37): public native String getHostByAddr(byte[] addr) throws UnknownHostException;

java.net.Inet6AddressImpl(40): public native String getLocalHostName() throws UnknownHostException;

java.net.Inet6AddressImpl(41): public native InetAddress[]

java.net.Inet6AddressImpl(43): public native String getHostByAddr(byte[] addr) throws UnknownHostException;

## "protected native" 17

java.lang.Object(212): protected native Object clone() throws CloneNotSupportedException;

java.lang.SecurityManager(316): protected native Class[] getClassContext();

java.lang.SecurityManager(422): protected native int classDepth(String name);

java.net.TwoStacksPlainDatagramSocketImpl(170): protected native void send(DatagramPacket p) throws IOException;

java.net.TwoStacksPlainDatagramSocketImpl(179): protected native void setTimeToLive(int ttl) throws IOException;

java.net.TwoStacksPlainDatagramSocketImpl(181): protected native int getTimeToLive() throws IOException;

java.net.TwoStacksPlainDatagramSocketImpl(184): protected native void setTTL(byte ttl) throws IOException;

java.net.TwoStacksPlainDatagramSocketImpl(187): protected native byte getTTL() throws IOException;

java.net.TwoStacksPlainDatagramSocketImpl(189): protected native void join(InetAddress inetaddr, NetworkInterface netIf)

java.net.TwoStacksPlainDatagramSocketImpl(192): protected native void leave(InetAddress inetaddr, NetworkInterface netIf)

java.net.TwoStacksPlainDatagramSocketImpl(195): protected native void datagramSocketCreate() throws SocketException;

java.net.TwoStacksPlainDatagramSocketImpl(197): protected native void datagramSocketClose();

java.net.TwoStacksPlainDatagramSocketImpl(199): protected native void socketNativeSetOption(int opt, Object val)

java.net.TwoStacksPlainDatagramSocketImpl(202): protected native Object socketGetOption(int opt) throws SocketException;

java.net.TwoStacksPlainDatagramSocketImpl(204): protected native void connect0(InetAddress address, int port) throws SocketException;

java.net.TwoStacksPlainDatagramSocketImpl(206): protected native Object socketLocalAddress(int family) throws SocketException;

java.net.TwoStacksPlainDatagramSocketImpl(208): protected native void disconnect0(int family);

## TOTAL 211