

Java Card™ 2.1 Development Kit Release Notes



Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303 USA
650 960-1300 fax 650 969-9131

Version 1.0P
November 15, 1999

Copyright © 1999 Sun Microsystems, Inc.

901 San Antonio Road, Palo Alto, CA 94303 USA

All rights reserved. Copyright in this document is owned by Sun Microsystems, Inc.

Sun Microsystems, Inc. (SUN) hereby grants to you at no charge a nonexclusive, nontransferable, worldwide, limited license (without the right to sublicense) under SUN's intellectual property rights that are essential to practice the Java Card™ technology to use this document for internal evaluation purposes only. Other than this limited license, you acquire no right, title, or interest in or to the document and you shall have no right to use the document for productive or commercial use.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-1(a).

SUN MAKES NO REPRESENTATIONS OR WARRANTIES ABOUT THE SUITABILITY OF THE SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. SUN SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES.

TRADEMARKS

Sun, Sun Microsystems, the Sun logo, Java, Java Card, SunDocs, and SunExpress are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. UNIX® is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.

THIS PUBLICATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. SUN MICROSYSTEMS, INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.

Contents

1. Introduction	1
2. Contents of this Release	3
3. Java Card™ 2.1 Development Kit Release	5
The Java Card 2.1 Development Kit Tools Reference Implementation Executables	5
The Java Card 2.1 Framework and Sample Applets	5
The Java Card 2.1 Specification	6
Compile-time constants and the CAP file specification	6
Clarification of Section 6.13.2 field_descriptor_info, field_ref item	6
4. Quality Assurance	9
Testing	9
5. New Features in this Release	11
Converter	11
capgen	11
JCWDE	11
6. Known Limitations in this Release	13
General	13

API 13

javacardx.crypto 13

javacard.security 13

Converter 14

JCWDE 14

7. Problems Fixed in this Release 15

Introduction

This document contains the release notes for the Java Card™ 2.1 Development Kit.

You can use this release to:

- View and experiment with the reference implementation of
 - Java Card 2.1 Converter tool
 - Java Card 2.1 `capgen` tool
 - JCWDE (Java Card Workstation Development Environment) Simulator
 - Java Card 2.1 ApduTool utility
 - Java Card 2.1 framework classes.
- Exercise the reference implementation to
 - convert the sample applet packages using the Converter tool.
 - build a CAP file using the `capgen` tool
 - run the virtual machine in JCWDE and exercise the installer applet to install a set of sample applets and libraries.

The Java Card 2.1 Development Kit tools run on a workstation using a Java Virtual Machine (VM).

For more details on the reference implementation, please refer to the *Java Card 2.1 Development Kit User's Guide*.

Contents of this Release

This release contains:

- The Java Card 2.1 Converter tool reference implementation pre-built binary.
- The Java Card 2.1 `capgen` tool reference implementation pre-built binary.
- The JCWDE reference implementation pre-built binary.
- The Java Card 2.1 AduTool utility reference implementation pre-built binary.
- The Java Card 2.1 framework and sample packages to run against the cJDK.
- Java Card 2.1 Development Kit User's Guide.

Java Card™ 2.1 Development Kit Release

This chapter describes what is and what is not included in this release.

The major components are the Java Card 2.1 tools reference implementation pre-built executables, and the example applets, including source code and pre-built executables.

These tools implement the *Java Card 2.1 Virtual Machine Specification (Final Revision 1.1)*

The Java Card 2.1 Development Kit Tools Reference Implementation Executables

The executables in `bin` directory are the Unix shell scripts and the MS Windows batch files for running the Converter, and the `capgen` tool, and the `ApduTool` utility, and their corresponding `jar` files.

The Java Card 2.1 Framework and Sample Applets

The `API21` directory comprises the Java Card 2.1 `java.lang` package and a complete implementation of the Java Card 2.1 `javacard.framework` package. `jcwde` comprises the workstation development environment scaffolding code to

allow Java Card applet development on a workstation. `samples` includes five sample Java Card 2.1 applets: `NullApp`, a basic applet which only returns error status; `HelloWorld`, a command APDU echo applet; `Wallet`, a simple electronic purse; `JavaPurse`, a smart card cash applet; `JavaLoyalty`, a loyalty applet which interacts with `JavaPurse`; and `SampleLibrary`, which contains the shared interface between `JavaPurse` and `JavaLoyalty`.

The Java Card 2.1 Specification

This release is compliant with the Java Card 2.1 specifications. The two items below are clarifications of the “Java Card 2.1 Virtual Machine Specification.”

Compile-time constants and the CAP file specification

Static final fields of primitive types (`int`, `short`, `byte`, and `boolean`) are not represented in a CAP File. This includes both the Static Field Component and Descriptor Component. This is described explicitly in the specification in section 6.1 Static Field Component, but is not mentioned in section 6.13, Descriptor Component. These compile-time constants are placed inline in bytecode sequences.

The values of static final fields of primitive types that are defined as part of an API of a package are represented in Export Files. These values are used when generating appropriate bytecode sequences.

Clarification of Section 6.13.2 `field_descriptor_info`, `field_ref` item

The `field_descriptor_info` structure of the CAP File’s Descriptor Component refers to an `instance_field_ref` structure. Such a structure is not defined in the specification.

The intent of the specification is to define the `field_ref` item as the content of a `CONSTANT_InstanceFieldref` without the constant’s tag item. The second paragraph of the `field_ref` item should read as follows:

If the `ACC_STATIC` flag is equal to 0, this item represents a reference to an instance field. It contains a `class_ref` item and an instance field token item. These items are defined in that same manner as in the `CONSTANT_InstanceFieldref_info` structure.

Quality Assurance

Testing

This release has been installed and tested on Win NT 4.0 and Solaris 7 systems, using Sun's JDK 1.1.8. Preliminary testing has been completed under Sun's JDK 1.2.2.

New Features in this Release

Converter

The Java Card Converter tool has been added, allowing you to do Java programming language subset checking, and with the `capgen` tool, to create CAP files.

`capgen`

The Java Card `capgen` tool has been added, allowing you to create CAP files from JCA files created by the The Java Card Converter tool.

JCWDE

The Java Card JCWDE tool has been enhanced, allowing applet creation (instantiation) and multiple instances of applets.

Known Limitations in this Release

General

- The reference implementation does not include the `javacardx.crypto` API packages. Export files are included however.
- The reference implementation does not support default Java Card applet packages. Applet and library packages must be named.

API

`javacardx.crypto`

Sources and binaries are not included.

`javacard.security`

`getInstance()` methods of `KeyBuilder`, `MessageDigest`, `Signature` and `RandomData` throw `NO_SUCH_ALGORITHM` exception.

Converter

- The class files input to the converter must have been compiled with the `-g` option.

Static fields can only be initialized to primitive compile-time constant values, or arrays of primitive compile-time constants. The converter supports static field initialization as in the following formats:

```
public static byte a = 1;
protected static short b = 4;
private static int c = 0xFFFF;
public static byte[] d = {1, 2, 3};
public static final short[] = { 0x11, 0x22, 0x33 };
public static int[] = new int[5];
```

The Converter reports errors on other formats of static field initialization and static field initialization types that are not compliant to the Java Card virtual machine specification.

For instance, the following code is not currently supported.

```
public static int[] a;
static {
a = new int[3]; for (int i = 0; i < 3; i++)
a[i] = i;
}
```

JCWDE

- No transactions or firewall.
- No download; all applets configured in mask.
- Execution simulates first time power up of card.

Problems Fixed in this Release

All logged problems except those listed in the below have been corrected in this release.

- Bug id 4275845. `Api21` – `SecurityException` is incorrectly thrown when the method `APDU.waitExtension()` is called.

