JI Integration

Release Notes

Version 4.6

September 2026

This document applies to JI Integration Version 4.6 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 1999–2026 Software GmbH, Darmstadt, Germany and/or their suppliers. All rights reserved.

The name Software GmbH, and all Software GmbH product names are either trademarks or registered trademarks of Software GmbH. Other company and product names mentioned herein may be trademarks of their respective owners.

Document ID: JI-RN-46-20260923

Table of Contents

Overview **11**
About JI Integration 11
Organization 11

JI Integration 4.6 Release Notes **12**
About This Release 12
Installation and Upgrade Information 12
Java and Application Server Versions 12
License Configuration 12
New Features Included in JI 4.6 13
Software AG License Key Harmonization 13
Java 17 Compatibility Improvements 13
Digital Signing of Release Files 13
Detailed Description of Fixes Included in JI 4.6 14
License Names Containing Spaces 14
Known Limitations 14
Features That May Be Removed in a Future Release 14

JI Integration 4.5.8 Release Notes **15**
Installation and Upgrade Information 15
Supported Platforms 15
JI Development Environments 16
JI Production Environments 17
General Data Protection Regulation 18
New Features Included in JI 4.5.8 19
Updated Operating System Support 19
Java Version 17 Support 19
Azul Java 17 Support 19
Tomcat 9 Support 19
Detailed Description of Fixes Included in JI 4.5.8 20
Known Limitations 20
Native-Code Client Library Support 20
The JI Installer May Calculate Free Disk Space Incorrectly 20
ea_admin plist Returns an Empty Process List on AIX 20
Tomcat 10 Support 21
Java Version Support 21
MySQL Database Support 21
Features That May be Removed in a Future Release 21

JI Integration 4.5.7 Release Notes **22**
Installation and Upgrade Information 22
Supported Platforms 22
JI Development Environments 23
JI Production Environments 23

General Data Protection Regulation 25
New Features Included in JI 4.5.7 25
Removal of Native Wrappers for Command Line Utilities 25
Updated Operating System Support 26
Embedded H2 Database Server Upgrade 26
Tomcat Application Server Upgrade 26
Detailed Description of Fixes Included in JI 4.5.7 27
5145771 - Mapmaker writes extra-large map files when editing multiple maps 27
5166959 - Issue sending data elements to mainframe screen 27
5194406 - Getting error trying to convert MYSQL to H2 for upgrade to 4.5.6 27
5197260 - Problem converting MYSQL DB to H2 DB during JI4.5.6 upgrade 28
5200165 - Found 'WARNING: No server license was found!' 28
Known Limitations 28
Native-Code Client Library Support 28
The JI Installer May Calculate Free Disk Space Incorrectly 28
ea_admin plist Returns an Empty Process List on AIX 29
Features That May be Removed in a Future Release 29

JI Integration 4.5.6 Release Notes 30
Installation and Upgrade Information 30
Supported Platforms 30
JI Development Environments 31
JI Production Environments 31
New Features Included in JI 4.5.6 33
Updated Operating System Support 33
Embedded H2 Database Server 33
Introduction of Native 64-bit Support 34
Inclusion of 64-bit Native Code Executables 34
Support for 64-bit JVMs 34
Tomcat Application Server Upgraded to Version 6.0.35 34
Improved Environment Startup performance 34
Updated HTTP Gateway Tutorial 35
Console Installation 35
Detailed Description of Fixes Included in JI 4.5.6 35
5044451 - Trying to use program “ea_admin”, not getting any results from commands 35
5070984 - Automate deployments 36
5078004 - Upgrading service not working 36
5088442 - MapMaker debugger bug 36
5094287 - java.util.ConcurrentModificationException initializing method variables . . 36
5098578 - Unable to display online help with Windows 7 / JI 4.5.5 new installation . . 36
5109799 - Service Logs Don’t Get Created Unless I Set the Log Level in Env Mngr to 9 and Other Log Questions 36
Known Limitations 37
E/A 2.x Legacy Support 37
Features That Might be Removed in a Future Release 37

JI Integration 4.5.5 Release Notes 38
Installation and Upgrade Information 38
Supported Platforms 38
JI Development Environments 38
JI Production Environments 39
New Features Included in JI 4.5.5 41
Command Line Batch Mode Interface for MapMaker 41
Export of JI Maps to SDFX Format 42
Updated Operating System Support 42
Tomcat 6 Support 42
Product Rebranding 42
Detailed Description of Fixes Included in JI 4.5.5 43
692225 - Service Waiting Forever 43
695375 - Deployment Failure with Jacada Upgrade to 4.5.3 & Other Questions 43
705691 - JI 4.5.4 Fails to Read 4.5.3 Map 44
708487 - HTTP 500 Error (UA Environment) 44
700954 - Jacada Passing a Negative Number (String) Into a Numeric Host field Issue 44
703341 - Message Line in Playback Showing Blank. 44
1021131 - Upgrade Causing JI to Crash 44
5003304 - Help Key is Not Included in the List of AID Keys for MapMaker Action Inputs 44
Known Limitations 45
ASP.Net Client Generation is Not Supported with Microsoft .NET SDK Version 2.0 or Newer 45
JI Environments Running on a 64-bit O/S Must Use a 32-bit JVM 45
Resource Adapter Generation is No Longer Supported in this Release 45

JI Integration 4.5.4 Release Notes 46
New Features Included in Release 4.5.4 46
Product Name 46
Tomcat 4.1.39 Support 46
Java 1.6 Support 46
Logging Improvements 46
Standardize SSL Cipher Suites 47
Removal of MySQL from the JI Installation 47
Installation and Upgrade Information 47
Installation Process 47
Supported Platforms 48
JI Server 48
MapMaker 48
Software Requirements 48
Java Software Requirements 48
Database Requirements 48
JI Licenses 49
X11-Compatible Window Server 49
JDBC Driver Included in Installation 49
EA_ENV Environment Variable 49

The environment.ccf File 50
DISPLAY Environment Variable 50
THREADS_FLAG Environment Variable 50
TERMINFO Environment Variable 51
Localhost 51
Directory Names that Contain Spaces 51
Communication Protocol Support 51
Supported Client and Integration Environments 51
Before Installation 52
Installing MySQL 5.0 52
Supported MySQL Versions 52
Installing JI 4.5.4 52
Upgrade Procedures 54
Database Migration 54
Upgrading JI Services 55
Upgrading JI Clients 55
Migrating from Previous JI 4.x Releases 55
Migrating from JI 3.5 56
Detailed Description of Fixes Included in JI 4.5.4 56
ATL-29279 – OnFail issue 56
ATL-28937 – Environment Manager logging 57
EU-05900 – Service logging improvements 57
SR-674182 – Make service log timestamp format configurable 57
Timestamp 58
Property 58
SR-676838 – NoSuchElementException 58
SR-691021 – Error in XML Parser Unicode character 0x12 59
SR-695830 – TLS Encryption Support (SSL 3.1) 59
Known Limitations 59
JVM Selection during installation 59
Uninstall fails to remove all unmodified files 60
MapMaker Thread Step 60
Arrays in a MapMaker Set Step 60
JVM Path 60
Array Indexes 60
JAVA_HOME Variable 60
Window Sizes in MapMaker 61
Data Mapping Editor Columns 61
Modifying a Data Template 61
Modifying a Traverse Step 61
Incomplete Validation in ForCondition Step 61
Attributes in XPath Expressions 62
Deleting Global Variables 62
Modifying Structure Name of Thread Step 62
Choose a Datasource Dialog Box 62
Replacing Invoked Methods 62
MapMaker Comments Tab 62
Modifying MapMaker Look and Feel 62

MapMaker Properties Dialog Box . . . 63
MapMaker Update Map Button . . . 63
JClient3 Step . . . 63
MapMaker Set Step . . . 63
Copying Types in the Business Entity Editor . . . 63
OnFail Links . . . 64
Curly Braces in Code Generation . . . 64
Auto Complete Feature in Set Step Properties . . . 64
Deploy to Resource Server Dialog Box . . . 64
ASP.NET Business Entities . . . 64
ASP.NET Client . . . 65
Date Types in XSDs . . . 65

Jacada Integrator 4.5.3 Release Notes . . . 66
New Features Included In Release 4.5.3 . . . 66
Platform Support . . . 66
MapMaker Fixes . . . 66
MapMaker Multicast Resource Address . . . 66
Custom code generation error . . . 66
SOAP Gateway Fixes . . . 66
SOAP Gateway White Space Preservation Option . . . 66
SOAP Gateway NoFacilities Exception . . . 67
JClient3 Fixes . . . 67
Services are becoming stuck in a wait state . . . 67
Server Fixes . . . 67
Jacada Services Failure . . . 67
Problem trimming spaces . . . 68
JIStructureType - Assignment Exception . . . 68

Jacada Integrator 4.5.2 Release Notes . . . 69
New Features Included In Release 4.5.2 . . . 69
Log file enhancements . . . 69
Support multiple secure host connections . . . 70
XSL transformation support for 4.x XSD inputs and outputs . . . 70
Upgrading from 3.5 to 4.5.2 . . . 71
Code Generation Errors . . . 71
Change in JClient3 API from 3.5 to 4.5.2 . . . 71
Global Variables Cleared . . . 71
MapMaker Fixes . . . 72
Find' option opens method in tree panel . . . 72
Scroll tags with escape characters supported . . . 72
Server Fixes . . . 72
ConcurrentModificationException and NoSuchElementException problems fixed . . . 72
Check for null value for secure host configuration works . . . 72
NullPointerException problem fixed for threads that are not killed after timing out . . . 72
SOAP Gateway Fixes . . . 73
SOAP Gateway CPU Utilization . . . 73

Jacada Integrator 4.5.1 Release Notes 74
Upgrading from 3.5 to 4.5.1 74
Code Generation Errors 74
Change in JClient3 API from 3.5 to 4.5.1 74
Global Variables Cleared 74
MapMaker 74
Data Encryption Between Jacada Integrator and the Host 74
Generating Service Code That Includes User methods in Custom Classes 75
Evaluation of NOT_PRESENT and NO_VALUE Structures 75
Opening a Map Containing Set Steps Using an XPath Expression 75
Name Changes to Internal Data Types in XPath Expressions 75
XPath and Output Formatting 75
Formatting Empty Stings into Dates 76
Invalid Arrows in Map Removed 76
ASP .NET Client Code Generation Improvements 76
Tracing Host Data 76
Action Comments in the Perform Step Combo Box 76
MapMaker and the Debugger 77
Closing the Debugger's JTerm Windows When Loading a New Map 77
Global Actions Created for the Debugger 77
Tags in Map Recognized by the Debugger 77
Message Box Displayed Upon Exception 77
Input Values Not Passed to External Method 77
Stopping Methods Doesn't Write Extra Messages to the Log 77
MapMaker and JTERM 78
Support for Field- and Field+ Keys 78
Support for Negative Values in Signed Numeric Fields 78
Client 78
JTerm Window Instantiated 78
Server 78
Fputs Used to Log the Host Presentation Space 78
Internal debug comments no longer written to log 78
Mapping VT100 keys 79
SOAP Gateway 79
Messages with Long String Fields Passed Through the JIWSVC Gateway 79
WSDL Generation for Methods With Custom Code 79
Server Command Line Utilities 79
ea_start command line utility 79
ea_status command line utility 79
EAServiceBean 80
Retrieving the LU Name After Making a Connection 80
Session Name Retrieved by EAServiceBean.getSessionName() 80
getWriter() method Added 80

Jacada Integrator 4.5 Release Notes 81
New Features Included in Release 4.5 81
Documentation Reduction and Optimization 81

Hardware Platforms and Operating Systems 82
Windows NT® 4.0 Workstation Limitations 82
Minimum System Recommendations 83
Production System 83
Development System 84
Disk Space Requirements 84
Temporary Directory for Installation 85
System Information 85
TCP Ports 85
Software Requirements 86
Java Software Requirements 86
Jacada Integrator Licenses 86
X11-Compatible Window Server 87
JDBC Driver Included in Installation 87
SUNWi1of Package 87
Motif Requirements 87
EA_ENV Environment Variable 88
The environment.ccf File 88
DISPLAY Environment Variable 89
THREADS_FLAG Environment Variable 89
TERMINFO Environment Variable 89
Localhost 89
Directory Names that Contain Spaces 89
Communication Protocol Support 90
Supported Client and Integration Environments 90
Migrating from Jacada Integrator Release 3.5 to 4.5 90
Upgrading Jacada Integrator Clients and Services 91
Service Library Changes 92
Migrating from Jacada Integrator Release 4.0 to 4.5 92
Known Limitations 93
MapMaker Thread Step 93
Arrays in a MapMaker Set Step 93
JVM Path 94
Array Indexes 94
JAVA_HOME Variable 94
Window Sizes in MapMaker 94
Renaming Fields in IBEs 94
Data Mapping Editor Columns 94
Modifying a Data Template 95
Modifying a Traverse Step 95
Incomplete Validation in ForCondition Step 95
Attributes in XPath Expressions 95
Deleting Global Variables 95
Modifying Structure Name of Thread Step 95
Choose a Datasource Dialog Box 96
Replacing Invoked Methods 96
MapMaker Comments Tab 96
Modifying MapMaker Look and Feel 96

MapMaker Properties Dialog Box 96
MapMaker Update Map Button 96
JClient3 Step 97
MapMaker Set Step 97
Copying Types in the Business Entity Editor 97
OnFail Links 97
Curly Braces in Code Generation 97
Auto Complete Feature in Set Step Properties 97
Deploy to Resource Server Dialog Box 98
ASP.NET Business Entities 98
ASP.NET Client 98
Special Instructions 98
Database Changes and Database Import/Export Concerns 99
Character Mode 100
Double-Byte Data Streams 100
Windows NT 100
Solaris 100
User Interaction Mode 101
SOCKS 101
Java Runtime Environment (JRE) Compatibility 101
MapMaker Issues 101
Windows Java Compiler not Case Sensitive 101

Overview

About JI Integration

Welcome to the Release Notes for JI Integration, an application development tool and runtime server environment that provides an industry leading, enterprise-strength solution for real-time interaction between users in client/server networks and applications running on legacy mainframe computer systems. The JI Integration application development tool and Application Program Interfaces (APIs) allow customers with large amounts of data stored on legacy systems to create modern client/server interfaces to their data without having to re-engineer their legacy applications. The JI Integration runtime server environment allows for large-scale implementation of these clients and services to provide a solution for mainframe access that can be used in large enterprise networks.

Organization

The JI Integration Release Notes are organized as follows:

- JI Integration 4.6 Release Notes on page 13
- JI Integration 4.5.8 Release Notes on page 16
- JI Integration 4.5.7 Release Notes on page 24
- JI Integration 4.5.6 Release Notes on page 32
- JI Integration 4.5.5 Release Notes on page 40
- JI Integration 4.5.4 Release Notes on page 48
- Jacada Integrator 4.5.3 Release Notes on page 68
- Jacada Integrator 4.5.2 Release Notes on page 72
- Jacada Integrator 4.5.1 Release Notes on page 78
- Jacada Integrator 4.5 Release Notes on page 86

Note: Release notes for prior releases of JI are provided for historical purposes only. The current release notes should be referenced for current software and hardware requirements.

JI Integration 4.6 Release Notes

About This Release

JI Integration 4.6 introduces Software AG license key harmonization through signed XML license files. The release also improves Java 17 RMI compatibility, updates license-status handling, and refreshes digital signing for selected release files.

Installation and Upgrade Information

Before upgrading, back up the existing configuration and license files. Refer to the *JI Integration Installation and Configuration Guide* for installation and upgrade procedures.

Java and Application Server Versions

JI 4.6 continues to use Java 17, including the Azul Zulu distribution, and includes Apache Tomcat 9.0.109. Java 17 and Tomcat 9 support were introduced in JI 4.5.8; this release builds on that runtime baseline.

License Configuration

When adding a license in the Configuration Manager, select the signed Software AG XML license file supplied for your installation. The new workflow replaces manual entry of a license key and license name.

Imported XML files are stored in **config/licenses** and linked to records in **config/license.txt**. Include both when backing up the license configuration. The referenced XML file must remain available for its license record to load. If an installation is moved, verify the recorded file paths.

Existing legacy key/name records remain readable and are not automatically converted to XML. Adding new licenses through the Configuration Manager uses the XML workflow.

A new license key is required to install JI Integration 4.6.0. To request a new key, complete the following form:
https://empower.softwareag.com/Products/OrderProducts/default.aspx
Contact Logistics at Generic.LogisticsServiceCenterGER@softwareag.com to obtain the new key. The license will be delivered in *.xml format.

New Features Included in JI 4.6

- Software AG License Key Harmonization
- Java 17 Compatibility Improvements
- Digital Signing of Release Files

Software AG License Key Harmonization

The Configuration Manager now provides an XML file chooser for importing Software AG licenses. JI verifies the license signature and reads the customer, product-version, licensed-feature, capacity and expiration information from the file.

The imported information is converted to the internal license representation used by JI services. The Configuration Manager retains the originating XML file path with the license record. Removing a license also removes its stored XML file when file permissions allow.

The Software AG licensing runtime is included with the installation; no separate licensing-library installation is required.

Java 17 Compatibility Improvements

RMI communication and monitoring components now use dynamic RMI stubs, removing their dependency on legacy generated stubs. The update covers Environment Manager monitoring, clustering, resource-server client callbacks and System Monitor.

RMI service startup has also been updated for the Java 17 execution path. RMI remains part of JI; this change modernizes its implementation without removing the communication mechanism.

Digital Signing of Release Files

Release packaging applies and verifies digital signatures for selected product JARs and Windows executable and library files, including the Windows installer. These signatures support verification of publisher identity and file integrity.

Detailed Description of Fixes Included in JI 4.6

License Names Containing Spaces

Startup license-status parsing now treats a license name containing spaces as one field. Previously, splitting the status line at every space could misinterpret the remaining license fields. Records that do not match the expected status format are ignored.

PDF Help Viewer Startup

PDF help files now open through the operating system's default PDF application without constructing a shell command, addressing a command-injection risk. Desktop integration and a registered PDF viewer are required.

Resource Database Query Handling

Resource database operations now use parameterized queries for field and key values, addressing SQL-injection risks. Updates that change a key use its previous value to locate the original record.

Known Limitations

The following limitations from JI 4.5.8 continue to apply:

- Java versions other than Java 17 and Apache Tomcat 10 are not supported.
- MySQL database support and conversion from MySQL to H2 are not supported from JI 4.5.8 onward.
- Native ActiveX, Standard C and thread-safe client libraries are not supported.
- The installer may calculate free disk space incorrectly. The existing workaround is to set CHECK_DISK_SPACE to OFF before running the installer.
- The ea_admin plist command may return an empty process list on AIX.

For further details, see the Known Limitations section in the JI 4.5.8 chapter. Retain imported XML license files with the license configuration; a license record referencing a missing XML file is not loaded.

Features That May Be Removed in a Future Release

The following features remain candidates for removal in a future release:

- JISOAP Gateway
- MQ Gateway
- MapMaker ASP.NET (C#) client generation
- MapMaker JCA Resource Adapter generation

JI Integration 4.5.8 Release Notes

Installation and Upgrade Information

JI Integration 4.5.8 is the next service pack of the JI Version 4.5 major release. Like previous service pack releases, JI 4.5.8 is delivered as a standalone installation.

JI 4.5.8 may be installed as follows:

- On a clean machine which does not contain a previous installation of JI.
- Side by side with an existing JI installation.

See Chapter 1 - "JI Integration Installation" on page 9 and “Upgrading From an Earlier Release of JI Integration” on page 51 in the *JI Integration Installation and Configuration Guide* for more detailed installation and upgrade instructions.

Note: Starting with JI Integration 4.5.8, the option “Converting a Database From MySQL to H2” is no longer supported. Refer JI Integration Installation and Configuration Guide for detailed instructions on upgrading from JI 4.5.5 and earlier versions and “JI Integration Upgrade Procedure” on page 51 for detailed instructions on upgrading from JI 4.5.6 and later.

Supported Platforms

Software GmbH provides support for operating system versions, Java versions, browser versions and application server versions supported by their respective manufacturers. Generally, when a provider stops supporting an OS version, Java version, browser version or application server version, Software GmbH will stop supporting that version as of the next JI Integration service pack level delivered by Software GmbH. Although it may be technically possible to run a new version of JI Integration using an unsupported OS, Java, browser or application server version, Software GmbH cannot continue to support configurations that are no longer supported by their respective vendors.

Platform support for JI Integration is separated into two categories:

- JI Development Environments
- JI Production Environments

JI Development Environments

A JI Integration development environment includes the JI Java GUI applications (MapMaker, Config Manager and System Monitor) and optionally, the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI development environment is primarily used for developing JI services but may also be used for managing a production environment. Services may be developed and deployed locally for testing on the development environment and then deployed remotely for production.

Supported Operating Systems

The JI Integration development environment is supported on the following operating systems / architectures:

- Windows 10 Professional 64-bit
- Windows 11 Professional 64-bit
- Red Hat Enterprise Linux (RHEL) 8.0
- Red Hat Enterprise Linux (RHEL) 9.0
- Windows Server 2016 Standard and Enterprise (64-bit)
- Windows Server 2019 Standard and Enterprise (64-bit)
- Windows Server 2022 Standard and Enterprise (64-bit)

Java Software Requirements

The following table lists the minimum JDK versions for running JI development environments:

Table 1. JI Development Environment Java Software Requirements

Operating System	Java Version
Windows 10	JDK 17
Windows 11	JDK 17
RHEL 8.0	JDK 17
RHEL 9.0	JDK 17
Windows Server 2016	JDK 17

Table 1. JI Development Environment Java Software Requirements

Operating System	Java Version
Windows Server 2019	JDK 17
Windows Server 2022	JDK 17

Note: To use the MapMaker graphical development environment's code generation capabilities, the Tomcat Server or the `ea_admin plist` command, the Java Development Kit (JDK) is required, and should be pre-installed before installing JI Integration.

JI Production Environments

A JI Integration production environment consists of the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI production environment is used for running the JI server applications. In addition, the JI Java GUI applications (MapMaker, Config Manager and System Monitor) may also be installed and run on Windows production environments. The JI Java GUI applications are only supported on Windows operating systems.

Supported Operating Systems

The JI Integration production environment is supported on the following operating systems / architectures:

- Red Hat Enterprise Linux 8 x86_64
- Red Hat Enterprise Linux 9 x86_64
- Windows Server 2011 Standard and Enterprise (64-bit)
- Windows Server 2016 Standard and Enterprise (64-bit)
- Windows Server 2019 Standard and Enterprise (64-bit)
- Windows Server 2022 Standard and Enterprise (64-bit)

Java Software Requirements

The following table lists the minimum JDK versions for running production environments:

Table 2. JI Production Environment Java Software Requirements

Operating System	Java Version
Linux / x86_64	JDK 17
Windows Server 2011	JDK 17
Windows Server 2016	JDK 17
Windows Server 2019	JDK 17
Windows Server 2022	JDK 17

Database Requirements

JI Integration 4.5.8 includes an embedded database and no longer requires or supports the use of MySQL for the JI resource database.

General Data Protection Regulation

While JI Integration does not collect or store any user personal information directly, the client, EnvMgr and service logs may contain personal information depending on the host application and JI Integration service methods.

JI Integration JClient3 clients running at log level 2 or higher will log client and server IP addresses and port numbers. In addition, JClient3 clients running at log level 3 or higher will log the contents of messages sent between the client and server. The JI Integration EnvMgr will log client IP addresses and ports at log level 3 or higher. JI Integration services will log host screens at log level 2 or higher and will log service / host interaction at log level 3 or higher.

Setting the JClient3 debug level to 1 or lower will disable access logging between the client and the server. As JClient3 applications, the JISOAP and JIWSVC web service gateways follow the same logging rules. Setting the EnvMgr log level to 2 or lower will disable client access logging. Setting the service log level to 1 or lower will disable logging of server / host interaction.

Any personal information in the JI server logs can be removed by deleting the EnvMgr, JCluster and service logs. JClient3 logging is configurable by the developer and the location and content of JClient3 client application log files, if any, is dependent on the JClient3 application. The JISOAP and JIWSVC web service gateways log to the logs directory of their respective web application server installation directories.

New Features Included in JI 4.5.8

The following new features have been added for JI 4.5.8:

- Updated Operating System Support
- Java Version 17 Support
- Azul Java 17 Support
- Tomcat 9 Support

Updated Operating System Support

Expanded compatibility to include Windows Server 2022. This update ensures the product can be seamlessly installed, configured, and run on Windows Server 2022.

Java Version 17 Support

JI Integration now supports Java 17 for both compile-time and run-time operations.

Azul Java 17 Support

JI Integration now supports Azul Java 17, enabling applications to be built and executed using the Azul Zulu distribution of Open JDK 17.

Tomcat 9 Support

JI Integration now supports Apache Tomcat 9 for deploying and running applications.

Detailed Description of Fixes Included in JI 4.5.8

This release brings product upgrades only, and no bug fixes are included.

Known Limitations

JI Integration 4.5.8 has the following limitations:

- Native-code client libraries support
- The JI installer may calculate free disk space incorrectly
- ea_admin plist returns an empty process list on AIX
- Tomcat 10 support
- Java version support
- MySQL database support

Native-Code Client Library Support

Due to the removal of all native-code components, the following features are no longer supported.

- ActiveX Client Library 32 and 64-bit
- Standard C Client Library 32 and 64-bit
- Thread-safe Client Library 32 and 64-bit

The JI Installer May Calculate Free Disk Space Incorrectly

In some cases, the JI installer may calculate the amount of free disk space incorrectly, requiring you to free disk space or quit the installation even though there is more than enough disk space available to install JI. To work around this problem, the environment variable "CHECK_DISK_SPACE" may be set to "OFF" prior to running the installer. See "Step 12: Pre-Installation Summary" on page 27 and "Step 12: Pre-Installation Summary" on page 46 of the *JI Integration Installation and Configuration Guide* for more details.

ea_admin plist Returns an Empty Process List on AIX

Due to problems with the IBM JVM implementation, the ea_admin plist command may return an empty process list even though there is a JI environment running.

Tomcat 10 Support

This release is certified for Apache Tomcat 9. Compatibility with Tomcat 10 has not yet been introduced.

Java Version Support

JI 4.5.8 works only with Java 17. Other Java versions are not supported.

MySQL Database Support

MySQL database is supported only in earlier versions (up to 4.5.7) and is not supported from 4.5.8 onwards.

Features That May be Removed in a Future Release

The following features may not be supported in a future release of JI Integration.

- JISOAP Gateway
- MQ Gateway
- MapMaker ASP.NET (C#) client generation
- MapMaker JCA Resource Adapter generation

JI Integration 4.5.7 Release Notes

Installation and Upgrade Information

JI Integration 4.5.7 is the next service pack of the JI 4.5 major release. Like previous service pack releases, JI 4.5.7 is delivered as a standalone installation.

JI 4.5.7 may be installed as follows:

- On a clean machine which does not contain a previous installation of JI.
- Side by side with an existing JI installation.

See Chapter 1 - "JI Integration Installation" on page 9 and “Upgrading From an Earlier Release of JI Integration” on page 51 in the *JI Integration Installation and Configuration Guide* for more detailed installation and upgrade instructions.

Note: The upgrade procedure for JI Integration 4.5.7 has changed from previous releases. See “Converting a Database From MySQL to H2” on page 52 in the *JI Integration Installation and Configuration Guide* for detailed instructions on upgrading from JI 4.5.5 and earlier versions and “JI Integration Upgrade Procedure” on page 51 for detailed instructions on upgrading from JI 4.5.6 and later.

Supported Platforms

Software GmbH provides support for operating system versions, Java versions, browser versions and application server versions supported by their respective manufacturers. Generally, when a provider stops supporting an OS version, Java version, browser version or application server version, Software GmbH will stop supporting that version as of the next JI Integration service pack level delivered by Software GmbH. Although it may be technically possible to run a new version of JI Integration using an unsupported OS, Java, browser or application server version, Software GmbH cannot continue to support configurations that are no longer supported by their respective vendors.

Platform support for JI Integration is separated into two categories:

- JI Development Environments
- JI Production Environments

JI Development Environments

A JI Integration development environment includes the JI Java GUI applications (MapMaker, Config Manager and System Monitor) and optionally, the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI deveopment environment is primarily used for developing JI services but may also be used for managing a production environment. Services may be developed and deployed locally for testing on the development environment and then deployed remotely for production.

Supported Operating Systems

The JI Integration development environment is supported on the following operating systems / architectures:

- Windows 7 Professional 32-bit / 64-bit
- Windows 10 Professional 32-bit / 64-bit

Java Software Requirements

The following table lists the minimum JDK versions for running JI development environments:

Table 1. JI Development Environment Java Software Requirements

Operating System	Java Version
Windows 7	JDK 1.8.0
Windows 10	JDK 1.8.0

Note: To use the MapMaker graphical development environment's code generation capabilities, the Tomcat Server or the `ea_admin plist` command, the Java Development Kit (JDK) is required, and should be pre-installed before installing JI Integration.

JI Production Environments

A JI Integration production environment consists of the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI production envionment is used for running the JI server applications. In addition, the JI Java GUI applications (MapMaker, Config

Manager and System Monitor) may also be installed and run on Windows production environments. The JI Java GUI applications are only supported on Windows operating systems.

Supported Operating Systems

The JI Integration production environment is supported on the following operating systems / architectures:

- AIX 7.1 POWER
- HP-UX 11v3 Itanium
- Red Hat Linux ELS 6 x86_64
- Red Hat Linux ELS 7 x86_64
- Solaris 10 SPARC
- Solaris 11 SPARC
- SUSE Linux Enterprise Server 11 x86_64
- Windows Server 2008 64-bit
- Windows Server 2012 64-bit

Java Software Requirements

The following table lists the minimum JDK versions for running production environments:

Table 2. JI Production Environment Java Software Requirements

Operating System	Java Version
AIX / POWER	JDK 1.8.0
HP-UX / Itanium	JDK 1.8.0
Linux / x86_64	JDK 1.8.0
Solaris / SPARC	JDK 1.8.0
Windows Server 2008	JDK 1.8.0
Windows Server 2012	JDK 1.8.0

Database Requirements

JI Integration 4.5.7 includes an embedded database and no longer requires or supports the use of MySQL for the JI resource database.

General Data Protection Regulation

While JI Integration does not collect or store any user personal information directly, the client, EnvMgr and service logs may contain personal information depending on the host application and JI Integration service methods.

JI Integration JClient3 clients running at log level 2 or higher will log client and server IP addresses and port numbers. In addition, JClient3 clients running at log level 3 or higher will log the contents of messages sent between the client and server. The JI Integration EnvMgr will log client IP addresses and ports at log level 3 or higher. JI Integration services will log host screens at log level 2 or higher and will log service / host interaction at log level 3 or higher.

Setting the JClient3 debug level to 1 or lower will disable access logging between the client and the server. As JClient3 applications, the JISOAP and JIWSVC web service gateways follow the same logging rules. Setting the EnvMgr log level to 2 or lower will disable client access logging. Setting the service log level to 1 or lower will disable logging of server / host interaction.

Any personal information in the JI server logs can be removed by deleting the EnvMgr, JCluster and service logs. JClient3 logging is configurable by the developer and the location and content of JClient3 client application log files, if any, is dependent on the JClient3 application. The JISOAP and JIWSVC web service gateways log to the logs directory of their respective web application server installation directories.

New Features Included in JI 4.5.7

The following new features have been added for JI 4.5.7:

- Removal of native wrappers for command line utilities.
- Updated operating system support
- Embedded H2 database server upgrade
- Tomcat application server upgrade

Removal of Native Wrappers for Command Line Utilities

Up to JI 4.5.7, some command line utilities used by JI for runtime operations such as ea_start, ea_shutdown, ea_status and ea_admin relied on code written in C (a.k.a. native code), mostly for command line parsing before invoking the actual

functions which were implemented in Java. This approach implied that porting JI to different server operating systems and different bit models (32/64-bit) required porting and building this native code component on multiple operating systems for multiple bit models. This has proven to be a daunting task for our maintenance team.

Therefore the focus of version 4.5.7 has been on porting this C code to Java in order to simplify the product maintenance and thus future proofing the product, while maintaining full backward compatibility with previous releases.

In addition, as of this release the product source code is entirely written in Java. All client components based on C and VisualBasic code were removed.

Updated Operating System Support

JI Integration development environments may now be run on Windows 10 (32 or 64-bit).

JI server environments may now be run on:

- Red Hat Enterprise Linux Server 7 (64-bit)
- Windows Server 2012 (64-bit)

Note: The following limitations apply to environments running on Windows Server 2012, Windows 2008 Server, Windows 7 and Windows 10:

- User Account Control must be turned off.
- The JI environment must be run using administrator priveleges.

Embedded H2 Database Server Upgrade

The H2 Database Engine has been upgraded to version 1.4.191 (2016-01-21). Migration from JI Integration 4.5.5 and earlier versions requires conversion of the database. Detailed instructions on migrating from from an older release of JI Integration to JI 4.5.7 can be found in the *JI Integration Installation and Configuration Guide*.

Tomcat Application Server Upgrade

The embedded Tomcat application server has been upgraded to version 8.0.14.

Detailed Description of Fixes Included in JI 4.5.7

The following customer fixes with Empower incident numbers are included in JI Integration 4.5.7:

- 5145771 - Mapmaker writes extra-large map files when editing multiple maps
- 5166959 - Issue sending data elements to mainframe screen
- 5194406 - Getting error trying to convert MYSQL to H2 for upgrade to 4.5.6
- 5197260 - Problem converting MYSQL DB to H2 DB during JI4.5.6 upgrade
- 5200165 - Found 'WARNING: No server license was found!'

5145771 - Mapmaker writes extra-large map files when editing multiple maps

Several MapMaker memory leaks that could cause saved maps to grow when opening, saving maps multiple times without restarting MapMaker have been fixed.

5166959 - Issue sending data elements to mainframe screen

Attempting to load a JI map with an unconnected method step would cause MapMaker to throw an exception and fail to load the map.

Non-list global variables passed to an external invoke were being wrapped in square brackets.

Entry of non-numeric characters in 3270 numeric fields is now configurable. See “TN3270 Configuration” on page 128 for details.

5194406 - Getting error trying to convert MYSQL to H2 for upgrade to 4.5.6

An uncaught error thrown during database conversion caused temporary tables to be left in place after the conversion. These temporary tables caused the next attempt to run the conversion to fail.

5197260 - Problem converting MYSQL DB to H2 DB during JI4.5.6 upgrade

A missing key specification for the ACLDetails table caused conversion of JI 4.5.5 and earlier databases with entries in the ACLDetails table to JI 4.5.6 to fail.

The PluginException thrown from MapMapaker at startup is no longer logged.

5200165 - Found 'WARNING: No server license was found!'

Initialization of licenses and feature codes in the Resource Server could fail due to concurrent access.

Known Limitations

JI Integration 4.5.7 has the following limitations:

- Native-code client libraries support
- The JI installer may calculate free disk space incorrectly
- ea_admin plist returns an empty process list on AIX

Native-Code Client Library Support

Due to the removal of all native-code components, the following features are no longer supported.

- ActiveX Client Library 32 and 64-bit
- Standard C Client Library 32 and 64-bit
- Thread-safe Client Library 32 and 64-bit

The JI Installer May Calculate Free Disk Space Incorrectly

In some cases, the JI installer may calculate the amount of free disk space incorrectly, requiring you to free disk space or quit the installation even though there is more than enough disk space available to install JI. To work around this problem, the environment variable “CHECK_DISK_SPACE” may be set to “OFF” prior to running the installer. See “Step 12: Pre-Installation Summary” on page 27 and “Step 12: Pre-Installation Summary” on page 46 of the *JI Integration Installation and Configuration Guide* for more details.

ea_admin plist Returns an Empty Process List on AIX

Due to problems with the IBM JVM implementation, the ea_admin plist command may return an empty process list even though there is a JI environment running.

Features That May be Removed in a Future Release

The following features may not be supported in a future release of JI Integration.

- JISOAP Gateway
- MQ Gateway
- MapMaker ASP.NET (C#) client generation
- MapMaker JCA Resource Adapter generation

JI Integration 4.5.6 Release Notes

Installation and Upgrade Information

JI Integration 4.5.6 is the next service pack of the JI 4.5 major release. Like JI 4.5.5, JI 4.5.6 is delivered as a standalone installation.

JI 4.5.6 may be installed as follows:

- On a clean machine which does not contain a previous installation of JI.
- Side by side with an existing JI installation.

See Chapter 1 - "JI Integration Installation" on page 9 and “Upgrading From an Earlier Release of JI Integration” on page 51 in the *JI Integration Installation and Configuration Guide* for more detailed installation and upgrade instructions.

> **Note:** The upgrade procedure for JI Integration 4.5.6 has changed from previous releases. See “Converting a Database From MySQL to H2” on page 52 in the *JI Integration Installation and Configuration Guide*.

Supported Platforms

Software GmbH provides support for operating system versions, Java versions, browser versions and application server versions supported by their respective manufacturers. Generally, when a provider stops supporting an OS version, Java version, browser version or application server version, Software GmbH will stop supporting that version as of the next JI Integration service pack level delivered by Software GmbH. Although it may be technically possible to run a new version of JI Integration using an unsupported OS, Java, browser or application server version, Software GmbH cannot continue to support configurations that are no longer supported by their respective vendors.

Platform support for JI Integration is separated into two categories:

- JI Development Environments
- JI Production Environments

JI Development Environments

A JI Integration development environment includes the JI Java GUI applications (MapMaker, Config Manager and System Monitor) and optionally, the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI deveopment environment is primarily used for developing JI services but may also be used for managing a production environment. Services may be developed and deployed locally for testing on the development environment and then deployed remotely for production.

Supported Operating Systems

The JI Integration development environment is supported on the following operating systems / architectures:

- Windows 7 32-bit / 64-bit
- Windows 8 32-bit / 64-bit
- Windows XP SP3 32-bit

Java Software Requirements

The following table lists the minimum JDK versions for running JI development environments:

Table 1. JI Development Environment Java Software Requirements

Operating System	Java Version
Windows 7	JDK 1.7.0
Windows 8	JDK 1.7.0
Windows XP SP3	JDK 1.7.0

JI Production Environments

A JI Integration production environment consists of the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI production envionment is used for running the JI server applications. In addition, the JI Java GUI applications (MapMaker, Config Manager and System Monitor) may also be installed and run on Windows production environments. The JI Java GUI applications are only supported on Windows operating systems.

Supported Operating Systems

The JI Integration production environment is supported on the following operating systems / architectures:

- AIX 6.1/7.1 POWER
- HP-UX 11v3 Itanium
- Red Hat Linux AS5 x86 / ELS 6 x86
- Solaris 10 SPARC
- Windows 2003 32-bit
- Windows 2008 32-bit/64-bit

Java Software Requirements

The following table lists the minimum JDK versions for running production environments:

Table 2. JI Production Environment Java Software Requirements

Operating System	Java Version
AIX / POWER	JDK 1.7.0
HP-UX / Itanium	JDK 1.7.0
Linux / X86	JDK 1.7.0
Solaris / SPARC	JDK 1.7.0
Windows Server 2003	JDK 1.7.0
Windows Server 2008	JDK 1.7.0

Database Requirements

JI Integration 4.5.6 includes an embedded database and no longer requires or supports the use of MySQL for the JI resource database.

New Features Included in JI 4.5.6

The following new features have been added for JI 4.5.6:

- Updated operating system support
- Embedded H2 database server
- Introduction of native 64-bit support
  - 64-bit native code executables
  - Support for 64-bit JVMs
- Tomcat application server upgraded to version 6.0.35
- Improved environment startup performance
- Updated HTTP Gateway tutorial
- Console installation

Updated Operating System Support

JI Integration development environments may now be run on Windows 8(32 or 64-bit).

JI server environments may now be run on:

- HP-UX 11v3 / Itanium
- Red Hat Enterprise Linux Server 6 (32 or 64-bit)
- Windows 2008 Server (32 or 64-bit)

Note: The following limitations apply to environments running on Windows 2008 Server, Windows 7 and Windows 8:

- User Access Control must be turned off.
- The JI environment must be run as the administrator user.

Embedded H2 Database Server

Due to licensing and platform support issues with MySQL, JI Integration 4.5.7 includes a new Resource Database implementation based on the the H2 Database Engine. The H2-based Resource Database replaces the MySQL-based Resource Database which is no longer required or supported. Migration from previous versions of JI Integration to Version 4.5.7 requires conversion of the database. Detailed instructions on migrating from from an older release of JI Integration to JI 4.5.7 can be found in the *JI Integration Installation and Configuration Guide*.

Introduction of Native 64-bit Support

JI Integration 4.5.7 introduces support for 64-bit systems in the following areas:

Inclusion of 64-bit Native Code Executables

The JI Integration 4.5.7 installer includes the option of installing 32 or 64-bit native code executables on supported 64-bit operating systems. The 64-bit installation option is only presented on 64-bit systems. Due to a limitation in InstallAnywhere, the Java launchers for 32 and 64-bit Windows systems are 32-bit executables. This limitation does not affect the Java launchers on UNIX systems.

Support for 64-bit JVMs

JI Integration 4.5.6 includes support for running the JI GUIs and servers on 64-bit JVMs on those systems that support them.

Items to consider when selecting a 32 or 64-bit JVM:

- All other things being equal, a 64-bit JVM will require more memory than a 32-bit JVM.
- Generally speaking, 32-bit JVM is limited to a maximum heap size of 2GB, a 64-bit JVM allows a maximum heap size of greater than 2GB.
- The amount of time required for JVM garbage collection increases as the JVM heap size increases.

In general, if your application is not limited by the constraints of running on a 32-bit JVM, there is probably no need to move to a 64-bit JVM. We recommend using a 64-bit JVM for new JI deployments.

Tomcat Application Server Upgraded to Version 6.0.35

The embedded Tomcat application server has been upgraded to version 6.0.35.

Improved Environment Startup performance

For JI Integration 4.5.6, the default multicast timeouts for the Environment Manager Resource Server lookup and Resource Server license service lookup have been reduced from 3 seconds to 1 second reducing the time it takes to start an environment by as much as 9 seconds each for the Resource Server and Environment Manager.

Updated HTTP Gateway Tutorial

The HTTP Gateway tutorial example.html has been updated to remove the dependency on the service generated from the obsolete tutorial.map, which has been removed. The HTTP Gateway example.html now depends on the service generated from the MajorBank.map.

Console Installation

JI Integration now includes support for console mode installations eliminating the need for for an X-server when installing on UNIX systems. Console mode installation is only suppported on UNIX systems. See the *JI Integration Installation and Configuration Guide* for detailed instructions on console mode installations.

Detailed Description of Fixes Included in JI 4.5.6

The following customer fixes with Empower incident numbers are included in JI Integration 4.5.6:

- 5044451 - Trying to use program "ea_admin", not getting any results from commands
- 5070984 - Automate deployments
- 5078004 - Upgrading service not working
- 5088442 - MapMaker debugger bug
- 5094287 - java.util.ConcurrentModificationException initializing method variables
- 5098578 - Unable to display online help with Windows 7 / JI 4.5.5 new installation
- 5109799 - Service logs don't get created unless I set the log level in Env Mngr to 9 and other log questions

5044451 - Trying to use program “ea_admin”, not getting any results from commands

The ea_admin command uses a Java child process to perform some of its commands. An incorrect argument in the child process command line was causing the child proces to fail, resulting in no output from the ea_admin command.

5070984 - Automate deployments

The MapMaker batch examples files are now included in the JI installation.

5078004 - Upgrading service not working

A limitation in screen sizes was causing MapMaker to disconnect from the host when attempting to traverse to a 5250 host screen which required a screen buffer larger than 7200 characters. The screen buffer size limit has been increased to 10240 characters.

5088442 - MapMaker debugger bug

A problem related to executing table template write steps while stepping through a method in the MapMaker debugger has been fixed.

5094287 - java.util.ConcurrentModificationException initializing method variables

Unsynchronized initialization of JI type mapping structures was causing JI type mappings to be lost, resulting in “processMappings failed : JITypeMapping not defined” messages being logged to the JCluster log.

5098578 - Unable to display online help with Windows 7 / JI 4.5.5 new installation

The JI Config Manager, MapMaker and System Monitor applications could no longer open the *JI Integration User’s Guide* when the Help key was pressed or when the Help / Help Topics menu item was selected.

5109799 - Service Logs Don’t Get Created Unless I Set the Log Level in Env Mngr to 9 and Other Log Questions

In some cases, the file separator character was being truncated from the log directory path passed to the JCluster, causing creation of the JCluster logs to fail. This problem did not occur if the Environment Manager log level was set to 9.

Known Limitations

JI Integration 4.5.6 has the following limitations:

- E/A 2.x Legacy support.

E/A 2.x Legacy Support

Due to the conversion of the JI resource database from MySQL to the embedded H2 server, Legacy C services are no longer supported in this release.

Features That Might be Removed in a Future Release

The following features might not be supported in a future release of JI Integration.

- ActiveX Client Library 32 and 64-bit
- Standard C Client Library 32 and 64-bit
- Thread-safe Client Library 32 and 64-bit
- MQ Gateway

JI Integration 4.5.5 Release Notes

Installation and Upgrade Information

JI Integration 4.5.5 is the next service pack of the JI 4.5 major release. Like JI 4.5.4, JI 4.5.5 is delivered as a standalone installation.

JI 4.5.5 may be installed as follows:

- On a clean machine which does not contain a previous installation of JI.
- Side by side with an existing JI installation.

See Chapter 1 - "JI Integration Installation" on page 9 in the *JI Integration Installation and Configuration Guide* for more detailed installation and upgrade instructions.

Supported Platforms

Software GmbH provides support for operating system versions, Java versions, browser versions and application server versions supported by their respective manufacturers. Generally, when a provider stops supporting an OS version, Java version, browser version or application server version, Software GmbH will stop supporting that version as of the next JI Integration service pack level delivered by Software GmbH. Although it may be technically possible to run a new version of JI Integration using an unsupported OS, Java, browser or application server version, Software GmbH cannot continue to support configurations that are no longer supported by their respective vendors.

Platform support for JI Integration is separated into two categories:

- JI Development Environments
- JI Production Environments

JI Development Environments

A JI Integration development environment includes the JI Java GUI applications (MapMaker, Config Manager and System Monitor) and optionally, the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI deveopment environment is primarily used for developing JI services but may also be used for managing a production environment. Services may be developed and deployed locally for testing on the development environment and then deployed remotely for production.

Supported Operating Systems

The JI Integration development environment is supported on the following operating systems / architectures:

- Windows 7 32-bit / 64-bit
- Windows XP SP3 32-bit

Java Software Requirements

The following table lists the minimum JDK versions for running JI development environments:

Table 1. JI Development Environment Java Software Requirements

Operating System	Java Version
Windows 7	JDK 1.6.0
Windows XP SP3	JDK 1.6.0

JI Production Environments

A JI Integration production environment consists of the JI server applications (Resource Database, Resource Server, Environment Manager and Tomcat application server). The JI production envionment is used for running the JI server applications. In addition, the JI Java GUI applications (MapMaker, Config Manager and System Monitor) may also be installed and run on Windows production environments. The JI Java GUI applications are only supported on Windows operating systems.

Supported Operating Systems

The JI Integration production environment is supported on the following operating systems / architectures:

- AIX 5.3/6.1 POWER
- HP-UX 11.23 Itanium
- Red Hat Linux AS5 x86
- Solaris 9/10 SPARC
- Windows 2003 32-bit
- Windows 2008 32-bit/64-bit

Java Software Requirements

The following table lists the minimum JDK versions for running production environments:

Table 2. JI Production Environment Java Software Requirements

Operating System	Java Version
AIX	JDK 1.6.0
HP-UX / Itanium	JDK 1.6.0
Linux / X86	JDK 1.6.0
Solaris	JDK 1.6.0
Windows Server 2003	JDK 1.6.0
Windows Server 2008	JDK 1.6.0

Database Requirements

A MySQL 5.1 database installation is required and should be pre-installed before installing JI 4.5.5. MySQL is available for all supported platforms from http://www.mysql.com and is distributed in two versions:

- MySQL Enterprise – Commercially licensed version.
- MySQL Community Server – Freely available GPL licensed version.

Each of these versions is packaged in various platform dependent ways but generally there is a compressed tarfile or zipfile version for each platform which can be installed. Detailed installation instructions for MySQL are available in the MySQL documentation at http://www.mysql.com.

The following MySQL package versions were tested for the JI 4.5.5 release:

Table 3. Database Requirements

Operating System / Platform	MySQL Package Version
AIX / Power	mysql-5.1.48-aix5.3-powerpc.tar.gz
HP-UX / Itanium	mysql-5.1.48-hpux11.23-ia64-64bit.tar.gz
Linux / X86	mysql-5.1.48-linux-x86_64-glibc23.tar.gz
Solaris / SPARC	mysql-5.1.48-solaris8-sparc.tar.gz
Windows / X86	mysql-5.1.48-win32.msi

New Features Included in JI 4.5.5

The following new features have been added for JI 4.5.5:

- Command line batch mode interface for MapMaker.
- Export of JI maps to SDFX format.
- Updated operating system support.
- Tomcat 6 support.
- Product Rebranding.

Command Line Batch Mode Interface for MapMaker

A command line batch mode interface has been added to MapMaker for this release. The command line batch mode interface can be used to automate the generation of services, clients and service definitions, building and deploying services to the resource database and deployment of web services. See the *Command Line Batch Mode* section in the *MapMaker* chapter of the *JI Integration User's Guide* and the *ea_mapmaker* manual page in the *JI Integration* Commands chapter of the *JI Integration Supplemental Reference Guide* for more information on the MapMaker command line interface.

Export of JI Maps to SDFX Format

It is now possible to export JI maps to the Software GmbH SDFX format from MapMaker. See the *Exporting* section of the *MapMaker* chapter in the *JI Integration User's Guide* for more information on exporting JI Maps.

Updated Operating System Support

JI Integration development environments may now be run on Windows 7 (32 or 64-bit).

JI server environments may now be run on:

- HP-UX 11.23 / Itanium
- Windows 2008 Server (32 or 64-bit)

Note: The following limitations apply to environments running on Windows 2008 Server and Windows 7:

- User Access Control must be turned off.
- The JI environment must be run as the administrator user.
- The JI environment must be run on a 32-bit JVM.

Tomcat 6 Support

The embedded Tomcat application server has been upgraded to Tomcat 6.0.24.

Product Rebranding

The Jacada Integrator product was acquired by Software GmbH and is now branded as JI Integration. The documentation has been updated to reflect this change except for historical references to prior versions of JI.

The following table maps the new terminology with the old terminology.

Table 4. Terminology Changes(Sheet 1 of 2)

Definition	New Terminology	Old Terminology
Company name	Software GmbH	Jacada

Table 4. Terminology Changes(Sheet 2 of 2)

Definition	New Terminology	Old Terminology
Product name	JI Integration	Jacada Integrator
Other references to Server, Client, and Application	JI	Jacada

Detailed Description of Fixes Included in JI 4.5.5

The following customer fixes with Empower incident numbers are included in JI Integration 4.5.5:

- 692225 - Service waiting forever
- 695375 - Deployment failure with Jacada upgrade to 4.5.3 & other questions
- 705691 - JI 4.5.4 Fails To Read 4.5.3 Map
- 708487 - HTTP 500 Error (UA environment)
- 700954 - Fix handling of negative numbers in 5250 signed numeric fields
- 703341 - 5250 Write Error Command not being handle correctly
- 1021131 - Upgrade causing JI to crash
- 5003304 - Help key is not include in the list of AID keys for MapMaker action inputs

692225 - Service Waiting Forever

A socket leak in the JClient3 library which could cause a JI service to be left in the WAITING state forever if the client timed out before the service completed its initialization has been fixed. In addition, the JCluster handling of incoming client connections has been improved.

695375 - Deployment Failure with Jacada Upgrade to 4.5.3 & Other Questions

A memory leak when deploying JI web services to the embedded Tomcat application server has been fixed.

705691 - JI 4.5.4 Fails to Read 4.5.3 Map

An incompatibility between JI 4.5.3 and JI 4.5.4 that could cause certain maps to fail to load in the JI 4.5.4 version of MapMaker has been fixed.

708487 - HTTP 500 Error (UA Environment)

A problem with the JI EnvMgr logging which could cause a NullPointerException to be thrown when EnvMgr logging is disabled has been fixed.

700954 - Jacada Passing a Negative Number (String) Into a Numeric Host field Issue

A problem with handling of negative numbers in 5250 signed numeric fields has been fixed.

703341 - Message Line in Playback Showing Blank.

A problem with handling the 5250 Write Error Command from a 5250 host which could cause the loss of error messages from the host has been fixed.

1021131 - Upgrade Causing JI to Crash

An enhancement was made to the JI service logging for failed Perform steps to aid in debugging service problems.

5003304 - Help Key is Not Included in the List of AID Keys for MapMaker Action Inputs

MapMaker has been enhanced to allow the use of the 5250 Help key in action inputs.

Known Limitations

JI Integration 4.5.5 has the following limitations:

- ASP.Net Client Generation is not support with Microsoft .NET SDK version 2.0 or newer
- JI Environments running on a 64-bit O/S must use a 32-bit JVM
- Resource Adapter generation is no longer supported in this release

ASP.Net Client Generation is Not Supported with Microsoft .NET SDK Version 2.0 or Newer

ASP.NET client generation requires the Microsoft .NET Framework Software Development Kit (SDK) version 1.1. Due to an incompatibility between the Microsoft .Net Framework SDK version 1.1 and newer versions, ASP.NET client code generated by MapMaker will not compile with newer versions of the .NET SDK.

Version 1.1 of the Microsoft .NET Framework SDK can be found on Microsoft's web site in the Microsoft Download Center: http://www.microsoft.com/downloads/en/default.aspx.

JI Environments Running on a 64-bit O/S Must Use a 32-bit JVM

JI Integration environments running on a 64-bit operating system platform must use a 32-bit JVM.

Resource Adapter Generation is No Longer Supported in this Release

Generation of JI services for deployment as Resource Adapters for application servers is no longer supported in JI 4.5.5. Resource Adapters are supported in JI 4.7.

JI Integration 4.5.4 Release Notes

New Features Included in Release 4.5.4

Product Name

The acquisition of the Jacada Integrator by Software GmbH required rebranding the product name. The new name is JI Integration, which can already be partially seen in the product and installation. The product documentation and tutorial have not been rebranded yet.

Note: *The name change does not influence the functionality except for the changes mentioned in this document.*

Tomcat 4.1.39 Support

The embedded application server has been upgraded to Tomcat 4.1.39.

Java 1.6 Support

In addition to the 1.4.2 JVM which is still supported, JI now supports the 1.6 JVM on Windows and Solaris.

Logging Improvements

With this release, log level 2 is designed to provide the necessary diagnostic information without creating very large log files and without significantly impacting performance. The following improvements have been made to JI logging:

- The service log level is now logged at service startup and when the log level is changed via the System Monitor.
- When a secure host connection is made from either MapMaker or a JI service, the list of defined secure hosts is logged.
- Service log time stamp format is now user configurable.
- The Environment Manager now logs the Resource Server URI, load balancing group and resource configuration name at startup.

- The Environment Manager now logs an error message when it detects the Resource Server has been disconnected.

Standardize SSL Cipher Suites

JI now accepts the following standard SSL cipher suites in addition to those previously supported for encrypted JI / Host communication:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_DSS_WITH_AES_128_CBC_SHA

Removal of MySQL from the JI Installation

MySQL 3.22 is no longer included in the JI installation. JI now requires MySQL 5.0 which must be installed separately. See detailed installation instructions below.

Installation and Upgrade Information

JI 4.5.4 is the next service pack (PTF) of the JI 4.5 major release. Unlike previous JI 4.5 PTFs, which were delivered as a differential installation and had to be installed on top of the JI 4.5 release installation, JI 4.5.4 is delivered as a standalone installation.

JI 4.5.4 may be installed as follows:

- On a clean machine which does not contain a previous installation of JI.
- Side by side with an existing JI installation.

Installation Process

In general, the process for installation of JI 4.5.4 consists of two steps:

- Installation of MySQL 5.0.
- Installation of the JI 4.5.4 environment.

Supported Platforms

JI Server

The JI server runtime environment is supported on the following platforms:

- Windows 2003 SP2, 32 bit kernel, 32 bit JDK, Sun JDK versions 1.4.2 and 1.6
- Solaris 10, 64 bit kernel, 32 bit JDK, JDK version Sun 1.4.2 and 1.6
- AIX 5.3 / 6.1, 64 bit kernel, 32 bit JDK, IBM JDK version 1.4.2

Note: *HP/UX, Linux x86 and Linux on Mainframe were not tested in this release.*

MapMaker

The MapMaker IDE, Configuration Manager and System Monitor are supported on the following platform:

- Windows XP, JDK 1.4.2 and 1.6

Software Requirements

The following software is required to use JI.

Java Software Requirements

A Java Virtual Machine (JVM) is required to run JI Java applications. See Supported Platforms for details on JVM versions.

To use the MapMaker graphical development environment's code generation capabilities and/or the Tomcat Server, the Java Development Kit (JDK) is required, and should be pre-installed before installing JI.

Note: *Due to the nature of Java technology where objects are exchanged across a network, it is recommended that all JI Java-based programs, regardless of platform, be run using the same version of the Java Virtual Machine, in order to avoid any JVM version incompatibilities.*

Database Requirements

A MySQL 5.0 database installation is required and should be pre-installed before installing JI 4.5.4. MySQL 5.0 may be acquired from http://www.mysql.com for all supported platforms. Database installation is described below.

JI Licenses

There are no changes to the JI licensing mechanism for JI 4.5.4. One or more JI licenses are required to use the JI environment. A licensing mechanism is included with JI that controls the following aspects of the JI environment:

- The capability of the MapMaker development tool to modify and generate JI Services.
- The number of concurrent servers (host machines) that are running JI Environment Managers. This limit is not tied to a specific server machine, which allows for flexibility in Environment Manager deployment. As long as the total number of Environment Managers in the User environment does not exceed the license limit, the Environment Managers can be running on any server in the network.
- The number of concurrent JI Services running on each server. (Optional license mechanisms are available for customer requirement flexibility.)

For more information, see the JI *User Guide*.

Note: *The licenses provided with JI 3.5 will not work with the 4.5.4 release. Consult JI Technical Support to obtain proper 4.5 licenses.*

X11-Compatible Window Server

To install JI on UNIX, an X11-compatible window server is required.

JDBC Driver Included in Installation

A required JDBC driver, tjFM, is included with installation of JI and is required to communicate with the Resource Database. The tjFM driver cannot be removed from JI and may not be used outside of JI except by acquiescence to tjFM's terms of agreement.

EA_ENV Environment Variable

In order to use the `ea_start` and `ea_shutdown` command line utilities, along with other administrative command line utilities, the `EA_ENV` environment variable should be set to the *<JI_install_dir>/config* directory. This is done automatically by the installation for Windows but must be done manually for UNIX. If this environment variable is not set, `ea_start` and `ea_shutdown` must be started using the command line option *-c <configuration directory>*.

Also, the `EA_ENV` environment variable must be set when using the compatibility kit, and can be used by certain JI clients.

The environment.ccf File

The *environment.ccf* file must be located in the directory referenced by the `EA_ENV` environment variable. The environment.ccf file must contain settings for the `EA_HOME, MYSQL_HOME, MYSQL_TCP_PORT, MYSQL_UNIX_PORT`, and `EA_ENVMGR` parameters:

- `EA_HOME`: The root directory of your JI installation. (For example: `C:\JI`)
- `MYSQL_HOME`: The root directory of your MySQL installation.
- `MYSQL_TCP_PORT`: The TCP port number of your MySQL daemon.
- `MYSQL_UNIX_PORT`: UNIX only. The UNIX domain socket for your MySQL daemon.
- `EA_ENVMGR`: The port number of the Environment Manager, used to determine if the Environment Manager is running.
- `START_TOMCAT`: Specifies whether the embedded Tomcat servlet engine is started and stopped, by default, with the `ea_start` and `ea_shutdown` commands. The default setting is "`true`".

Additional parameters in the *environment.ccf* file are used by the Compatibility Kit portion of JI and clients developed with certain JI client libraries.

Also, the `EA_HOME, MYSQL_HOME, MYSQL_TCP_PORT, MYSQL_UNIX_PORT` (UNIX only), and `EA_ENVMGR` parameters can be defined as environment variables, if the *environment.ccf* file is not used.

The installation will create an *environment.ccf file* with default values in the *<JI_install_dir>/config* directory.

DISPLAY Environment Variable

On UNIX, when using JI graphical interfaces or opening a terminal window, the DISPLAY environment variable must be set to the appropriate IP address or host name of the machine on which your X-server software is running. This variable is set using the format `<host>:0.0` where `<host>` is the name or IP address of the host machine.

For example:

```
export DISPLAY=127.0.0.1:0.0
```

THREADS_FLAG Environment Variable

On some UNIX platforms, the `THREADS_FLAG` environment variable must be set to native. For example:

```
export THREADS_FLAG=native
```

TERMINFO Environment Variable

On HP-UX, the `TERMINFO` variable must be set for proper function key behavior within PATerm terminal windows.

Set the `TERMINFO` environment variable to the *<JI_install_dir>/lib/terminfo* directory.

Localhost

The host *localhost* must be defined in your */etc/hosts* file in UNIX (or NIS/YP, if appropriate) or *c:\Windows\System32\drivers\etc\hosts* file in Windows. This is the default host for JI. The loopback IP address generally is 127.0.0.1.

Directory Names that Contain Spaces

It is recommended that you do not install JI into the *C:\Program Files* on Windows directory or any directory tree that has spaces in the names of the directories. This restriction applies to both Windows and UNIX installations.

Communication Protocol Support

JI supports TCP/IP communication protocols for client connections and the following communication protocols for mainframe connections:

- VT
- TN3270
- TN5250

See the *JI Supplemental Reference Guide Chapter 3* for keyboard mapping information. Protocol agent configuration information is located in the *<JI_install_dir>/lib/ProtoAgents* directory, where *<JI_install_dir>* represents the directory in which JI has been installed.

Supported Client and Integration Environments

The following client libraries and interfaces are available with JI:

- The Java Client Library (JClient3)
- The ActiveX Client Library (AXCL) [Windows only]
- The SOAP Gateway
- The MQ-Series Gateway
- The XML/HTTP Gateway

For more information about JI client libraries, see the JI Client Developer Guide.

Before Installation

On UNIX, before running the Installer, ensure that the 1.4.2 or 1.6.0 JVM you wish to use to run JI is in your search path.

Installing MySQL 5.0

MySQL is available from http://www.mysql.com and is distributed in two versions:

- MySQL Enterprise – Commercially licensed version.
- MySQL Community Server – Freely available GPL licensed version.

Each of these versions is packaged in various platform dependent ways but generally there is a compressed tarfile or zipfile version for each platform which can be installed.

Supported MySQL Versions

The following MySQL 5.0 package versions were tested for the JI 4.5.4 release:

- AIX mysql-5.0.77-aix5.2-powerpc.tar.gz
- Solaris myql-5.0.77-solaris10-sparc-64bit.tar.gz
- Windows mysql-5.0.77-win32.zip

Note: The MySQL Essentials installer for Windows does not include all the required utilities and is not supported for use with JI 4.5.4.

Detailed installation instructions for MySQL are available in the MySQL documentation at http://www.mysql.com.

Installing JI 4.5.4

Installation of JI 4.5.4 consists of the following steps:

- Run the installer. For Windows, the installer is called `setup.exe`. For UNIX the installer is called `setup.bin`.
- Accept the license agreement.
- Select an installation directory.
- Choose a JVM to run the JI environment.

Note: A JDK is required to generate services with MapMaker and to compile JSPs in Tomcat. It may be necessary to browse for a JDK if the installer lists only JREs in the Choose a Java Virtual Machine window.

- Select the MySQL installation directory. This step was added in JI 4.5.4 to allow selection of a previously installed MySQL database installation. JI 4.5.4 will not function without a correctly installed MySQL database. The MySQL installation dialog will look similar to this:

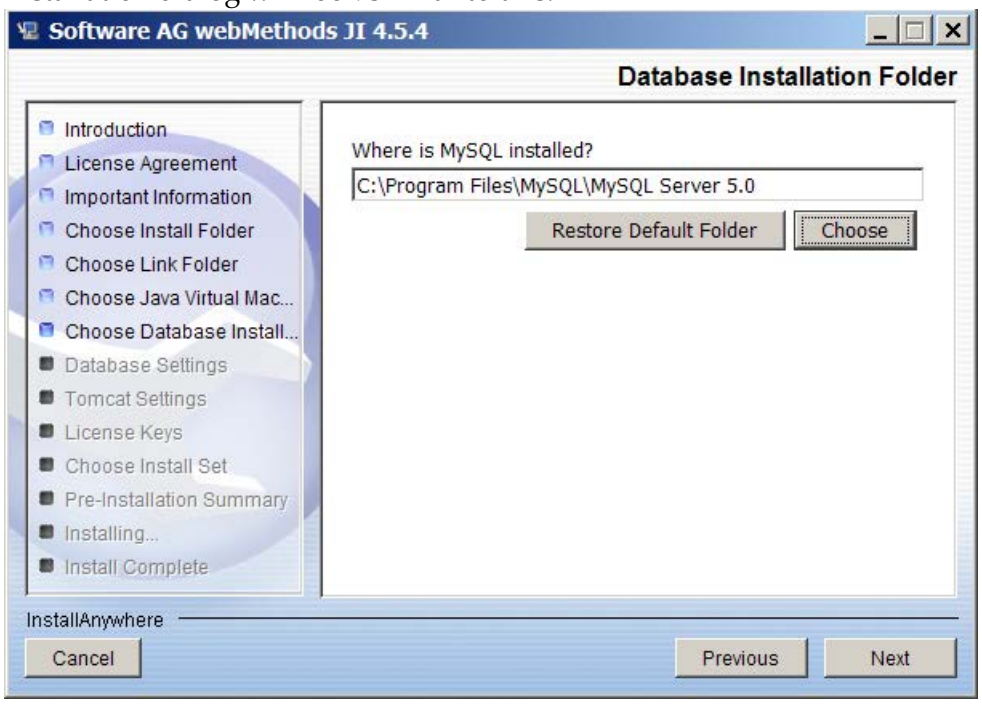

Figure 1. Select the MySQL installation directory

The initial default folder depends on the platform and environment. On Windows, the registry is checked first and if no MySQL installation is found, the environment variable MYSQL_HOME is checked. If neither of these is found, the default folder is set to: "<MYSQL_INSTALLATION_FOLDER>". On UNIX, the environment variable MYSQL_HOME is checked. If MYSQL_HOME is not set, the default folder is set to: "<MYSQL_INSTALLATION_DIRECTORY>".

The directory chosen should be the root of a MySQL 5.0 installation. The directory will be validated by the installation when the Next button is pressed. If the chosen directory is not a valid MySQL installation, an error message will be displayed and you will be returned to the Database Installation Folder selection dialog. It is not possible to continue the installation of JI 4.5.4 without selecting a valid MySQL installation.

- Select a database port and database name and choose whether or not to create a database.

- Select a Tomcat port.
- Enter JI license key and license name.
- Select an installation set. Only the “Full Server” option was tested by QA.

Upgrade Procedures

When migrating from earlier JI installations, databases must be exported and then imported into JI 4.5.4, services should be regenerated and redeployed and clients should be recompiled.

Database Migration

To migrate the database from an earlier JI release using MySQL 3.2.2 to JI 4.5.4 using MySQL 5.0 the following steps should be taken:

- Start the database daemon for the existing installation.

  ```
  ea_start -m
  ```

- Export the existing database to a file using the `ea_exportdb` utility.

  To export the standard data, use this command (all one statement):

  ```
  ea_exportdb ACLDetail ACLMaster EMConfig Hosts <export_file>
  ```

  The ACLDetail and ACLMaster tables may be omitted if there is no Access Control configured in the environment being upgraded. The JSvcDetail and JSvcMaster tables can also be exported; however, it is recommended that new service master records be created when the services are upgraded.

  If legacy C services have been configured, the legacy data should be exported with this command:

  ```
  ea_exportdb -l <legacy_export_file>
  ```

- Shut down the existing environment including the database daemon, Resource Server and Environment Manager processes.

  ```
  ea_shutdown
  ```

- Start the JI 4.5.4 database daemon:

  ```
  ea_start -m
  ```

- Import the database file(s) from the previous JI installation using the `ea_importdb` utility. If legacy services are being upgraded, import the legacy export file.

  ```
  ea_importdb <export_file>
  ea_importdb <legacy_export_file>
  ```

- Start the JI 4.5.4 environment, i.e., start the database, Resource Server and Environment Manager.

  ```
  ea_start -r -e
  ```

- Verify the export/import process by starting Configuration Manager and inspecting the relevant resources

  ```
  ea_cfgmgr
  ```

Upgrading JI Services

To upgrade existing JI services, the following steps should be taken for each map after adjusting the MapMaker properties as necessary for the new installation of JI:

- Load the existing map into the JI 4.5.4 version of MapMaker.
- Update the map by clicking the Update Map button or selecting File/Update Map from the menu.
- Generate the service.
- Deploy the service to the JI 4.5.4 resource database.

Upgrading JI Clients

To upgrade existing JI clients, the client code should be recompiled using the JI 4.5.4 versions of the required libraries. The recompiled clients should then be redeployed as necessary.

Migrating from Previous JI 4.x Releases

There are two options when upgrading to JI 4.5.4 from a previous release of JI 4.x:

- Install JI 4.5.4 along side of the existing installation.
- Back up and remove the existing installation and install JI 4.5.4.

Install JI 4.5.4 Alongside the Existing Installation

- Select a different directory than your existing JI 4.x installation when installing JI 4.5.4.
- Copy the .mapmaker40.cfg file from your previous JI 4.x installation into the JI 4.5.4 installation. The .mapmaker40.cfg file is created in the %USERPROFILE% directory by default but the location can be specified with the "-f" comand line option.
- Modify the properties for MapMaker in the JI 4.5.4 programs group menu to point to the JI 4.5.4 copy of .mapmaker40.cfg:
- From the Start menu, select the JI 4.5.4 program group.
- Right click the MapMaker entry and select properties.
- Select the Shortcut tab in the Properties dialog.
- Append the following to the target command line:
  -f "<JI454InstallDir>/.mapmaker40.cfg" -l "<JI454InstallDir>/ .mapmaker_license.cache"
  where <JI454InstallDir> is the installation directory for JI 4.5.4.

- Set the value of the EA_ENV environment variable to: <JI454InstallDir>/config. You can set the value of EA_ENV back to its previous setting to run the previous JI installation.
- Adjust the JI 4.5.4 MapMaker properties in the Properties dialog as necessary for the JI 4.5.4 installation by selecting Files/Properties from the menu.

Back Up and Remove the Existing Installation and Install JI 4.5.4

- Back up the existing JI 4.x installation.
- Uninstall the existing JI 4.x installation.
- Install JI 4.5.4.

Migrating from JI 3.5

In addition to the upgrade procedure described above, maps created with a JI 3.5 version of MapMaker need to be converted for use within JI 4.5.4. JI 4.5.4 includes a conversion utility for MapMaker 3.5 maps. This command-line utility is named `ea_convertmap` and is found in the <JI_install_dir>\bin directory.

Usage is:

```
ea_convertmap My_JI35_Map.map
```

The JI 3.5 map file will be backed up with a `.old` extension added to it. The converted map will retain the existing original name. The converter will output various diagnostic messages. The last couple of lines will indicate the success of the conversion. Once the conversion is complete, the converted map should be loaded into the JI 4.5.4 MapMaker, updated and the resulting services regenerated and re-deployed into the JI 4.5.4 environment.

The licenses provided with JI 3.5 will not work with the JI 4.5.4. Consult Software GmbH JI Integration Technical Support to obtain proper JI 4.5.4 licenses.

Detailed Description of Fixes Included in JI 4.5.4

ATL-29279 – OnFail issue

If an OnFail fired during an internal invoke from within a branch step, the error message fetched from the screen would get overridden by the branch condition error and not get mapped into the ErrorReportType structure.

The OnFail error is now included in a cause/caused by exception message stacked on top of the branch condition error.

ATL-28937 – Environment Manager logging

The Environment Manager did not record any log messages identifying the Resource Server, load balancing group or resource config name during startup making it difficult to determine which Resource Server an environment is connected to. If the Resource Server stopped while the environment was still running, the Environment Manager did not record any log messages or attempt to reconnect.

The Environment Manager now logs the Resource Server URI, load balancing group and resource configuration name at startup. The Environment Manager also logs an error message if it detects that the Resource Server has been disconnected and attempts to reconnect.

EU-05900 – Service logging improvements

Previously, in order to log certain minimum items, JI services had to be run at log level 9 which produces large amounts of log traffic potentially impacting performance.

With this release log level 2 is designed to provide the necessary diagnostic information without creating very large log files and without significantly impacting performance. In order to achieve this, JI service logging has been modified as follows:

- Host traces are enabled only if the log level is 3 or higher.
- Branch step return values are logged at log level 2 and higher.
- Service events are logged at log level 2 or higher.
- Method return values are logged at log level 2 or higher.
- Data fields sent to the host are logged at log level 2 or higher.
- AID keys sent to the host are logged at log level 2 or higher.
- Action inputs sent to the host are logged at log level 2 or higher.
- The current presentation space (host screen) is logged at log level 2 or higher.

SR-674182 – Make service log timestamp format configurable

The JI service log timestamp format is now configurable.

The JI service log timestamp can now be customized by modifying the javacmd property in the Environment Manager config file: envmgr.cfg. The following table describes the properties which can be used to modify JI service log timestamps:

Timestamp	Property
Host trace timestamps	com.jacada.brxtrace.BrxTrace.timeStampFormat
Non-Host trace timestamps	com.jacada.common.util.LogOutputWriter.timeStampFormat
All JI service log timestamps	com.jacada.timeStampFormat

The JI service log timestamps are divided into two parts, the Host Trace timestamps and the rest of the service log timestamps. The com.jacada.brxtrace.BrxTrace.timeStampFormat and com.jacada.common.util.LogOutputWriter.timeStampFormat properties take precedence over the com.jacada.timeStampFormat property. The format used for specifying the timeSampFormat is documented in the java.text.SimpleDataFormat JavaDoc page. For example, to produce log messages with a two digit month, two digit day of the month, hours, minutes, seconds and milliseconds the following should be added to the javacmd property:

```
-Dcom.jacada.timeStampFormt="MM-dd hh:mm:ss.SSSS"
```

The resulting javacmd property would look like this:

```
javacmd=C:\jdk1.5.0_10\bin\javaw.exe -Xmx256M -Xss1M -
Djava.security.manager  -
Djava.security.policy=C:\Integrator454\etc\.java.policy -
com.jacada.timeStampFormat="MM-dd hh:mm:ss.SSSS"
```

Note: There are no line breaks in the the javacmd property, it should be entered as one line.

SR-676838 – NoSuchElementException

Previously it was not possible to include spaces in the javacmd property in the Environment Manager config file envmgr.cfg, preventing setting of Java properties which required spaces.

It is now possible to include spaces in the javacmd property by quoting the parts of the property which require spaces. Single and double quotes may be used and the backslash character "\" may be used to escape a quote within a quoted string. For example, to set the java command to a path that includes spaces on windows, the following would be used:

```
javacmd="C:\Program Files\Java\JDK1.6.0\bin\java.exe"
```

Care must be taken when quoting strings which include backslashes. For example, the following would not give the same results as the example above:

```
javacmd=C:\"Program Files"\Java\JDK1.6.0\bin\java.exe
```

The \" sequence is seen as an escape allowing the quote to be inserted into the string resulting in a javacmd property set to: C:"Program Files"Java\JDK1.6.0\java.exe.

SR-691021 – Error in XML Parser Unicode character 0x12

The UTF-8 encoded XML returned from JI (non-Basic Data Operations) methods caused XML parsing failures on returned data that included non UTF-8 characters.

The XMLEncoding property in the Environment Manager config file has been extended to apply to non-Basic Data Operations methods allowing user configuration of the encoding for XML returned from JI methods on an environment-wide basis. For example, to set the XML encoding to ISO-8859-1, the XMLEncoding property in the envmgr.cfg file should be set as follows:

```
XMLEncoding=ISO-8859-1
```

SR-695830 – TLS Encryption Support (SSL 3.1)

The list of cipher suites supported for SSL encryption of JI service / host communication now includes the following:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_DSS_WITH_AES_128_CBC_SHA
- SSL_RSA_WITH_RC4_128_MD5
- SSL_RSA_WITH_RC4_128_SHA
- SSL_RSA_WITH_DES_CBC_SHA
- SSL_RSA_WITH_3DES_EDE_CBC_SHA
- SSL_RSA_EXPORT_WITH_RC4_40_MD5

Known Limitations

JVM Selection during installation

The JI 4.5.4 install may not present any JDKs in the Choose a Java Virtual Machine window, requiring the user to browse for a JDK by clicking the Choose Another button.

Uninstall fails to remove all unmodified files

In some cases, the JI 4.5.4 uninstaller my fail to remove all unmodified files and directories. These can be removed manually. On Windows, it is recommended to uninstall via the Add or Remove Programs applet in the Control Panel.

MapMaker Thread Step

An exception on the main Thread is always sent to the client, but an exception on a Thread branch is not. Use an OnFail step to handle Thread branch exceptions.

Arrays in a MapMaker Set Step

When using a Set step to fill an empty array, the Add operator appends the new value after the “NoValue” element in the first array index. Use the Set operator to replace the “NoValue” element with the new value.

The same principle applies when adding a structure array with 0 elements to a structure array with 1 element. This returns a structure array with 2 elements because a Global variable has a value of “Not Present”, and adding an additional value appends instead of overwrites the “Not Present” value. Use the Set operator instead of the Add operator to overwrite the “Not Present” value.

JVM Path

When installing JI, the path specified for the JVM must be a full path to a JDK or JRE installation. Symbolic links are not supported in this path.

Array Indexes

In an XPath expression, the first element in an array is element [1]. In a Global variable, the first element in an array is element [0].

JAVA_HOME Variable

The JAVA_HOME environment variable used in Tomcat startup is calculated from the envmgr.cfg file, using the envmgr.cfg java command for the JCluster and not the general environment JAVA_HOME variable.

Window Sizes in MapMaker

The window sizes of the Structure Relationship Editor and Business Entity Editor are in proportion to the size of the MapMaker main window. This means that when the MapMaker main window is made small, the Structure Relationship Editor and Business Entity Editor open in very small and unusable windows. These windows can only be enlarged after the MapMaker main window is enlarged.

Renaming Fields in IBEs

When renaming a field in an Internal Business Entity, the field name is not updated in some of the more complex XPath expressions referencing that field.

Data Mapping Editor Columns

When the data contained in the Mapped Data table is too long to be fully displayed in a column, it gets an ellipses rather than a horizontal scroll bar. The full version of the data can be viewed in the particular cell's tooltip.

Modifying a Data Template

Modifying data templates in screens that are already used by particular methods can affect those methods. You must update such methods to account for any changes made to the data template.

Modifying a Traverse Step

In a method containing a Traverse step linked to a Perform step, changing the current screen for the Traverse step does not modify the action in the Perform step. The action can be modified by clicking the **Reset Current Screen** button in the **General** tab of the Perform step properties.

Incomplete Validation in ForCondition Step

When entering the invalid combination of a string with an internal number for comparison in the **Conditional** tab of the ForCondition step properties, the **Numeric** checkbox is incorrectly enabled and selected. The correct behavior would be for an error message to be displayed, pointing out the invalid use of a non-numeric value.

Attributes in XPath Expressions

The XPath Evaluator does not support attributes in XPath expressions.

Deleting Global Variables

In the Business Entity Editor, deletion of Global variables using the Delete key on the keyboard is not recommended.

Modifying Structure Name of Thread Step

The MapMaker Presentation view does not update the modification of a structure name in the **General** tab of the Thread step properties.

Choose a Datasource Dialog Box

The **Current Type Definition** field in the **Choose a Datasource** dialog box retains the value entered even when that value is invalid.

Replacing Invoked Methods

When replacing an invoked method with a method of a different external type, the data mappings are not updated. This is rectified by manually deleting the old data mappings and adding new data mappings in the **Data Mapping Editor**.

MapMaker Comments Tab

MapMaker does not retain modifications made in the **Comments** tab of the **Properties** view, unless the **Apply** button is clicked before clicking another area of the GUI.

Modifying MapMaker Look and Feel

The **Appearance** tab of the **Properties** dialog box enables you to modify the look and feel of MapMaker.

- In some cases, this feature displays dialog boxes incorrectly, in that the **OK** and **Cancel** buttons are partially out of the window's scope.

- When changing the look and feel from one option to a second option and then back again, the second look and feel remains in some dialog boxes. This is rectified by restarting MapMaker after changing the look and feel.

MapMaker Properties Dialog Box

The **Cancel** button in the MapMaker **Properties** dialog box (accessed from the **File** menu) does not work. Clicking the **Cancel** button applies changes. Changes can only be cancelled by reapplying the previous settings.

MapMaker Update Map Button

After clicking the **Update Map** button, the Properties view shows the properties of a Start method step regardless of the tab currently selected in the Tab view.

JClient3 Step

A service opened by a JClient 3 step is not disconnected when a service open fails.

If a client that is a JClient3 step in a JI service attempts to open a service without first connecting to the host, the service open fails leaving the service connected but not open. If there is a subsequent attempt to open the service, it will also fail because the service is already connected and there is no means of recovery without terminating the service.

MapMaker Set Step

The tooltip for a Set step does not show the specified array indexes for the left-hand side Global variable expression.

Copying Types in the Business Entity Editor

Copying a Type that contains a subtype, sometimes causes the subtype to be copied. When this happens, the subtype is copied without its sample values. Sample values can be added manually in the **TypeDataModel Viewer**.

OnFail Links

MapMaker incorrectly allows OnFail links to be attached to steps that are on branches. Do not use OnFail links in this way.

Curly Braces in Code Generation

If a line of code contains only a curly brace, "{", on its own, do not put a comment on the previous line. The comment can be placed two lines above the curly brace.

Auto Complete Feature in Set Step Properties

The Auto Complete feature is activated indiscriminately in the Set Step's **General** tab. The Auto Complete feature provides variable names beginning with the letter you type into the right-hand object name field.

However, if you select a type that is not a variable from the **Data Source** dropdown list, such as **InternalString**, and then start to enter the object name in the next field, the field incorrectly auto completes with variable names, even though the object you are specifying is not a variable.

Deploy to Resource Server Dialog Box

The **Web Services** tab in the **Deploy to Resource Server** dialog box contains the following limitations:

- When Services is selected in the **Manage** dropdown list, the **Undeploy** column incorrectly appears to be enabled.
  The **Undeploy** action does not apply to the list of services.
- When **Deployed Web Services** is selected in the **Manage** dropdown list, the **Deploy** column incorrectly appears to be enabled.
  The **Deploy** action does not apply to the list of deployed web services.

ASP.NET Business Entities

ASP.NET client services will not work if any method contains a Business Entity that contains two structures, both of which contain one element, and that one element has the same name in both cases.

For example, performing the Copy To action on the ErrorReport type Internal Business Entity, results in an External Business Entity that contains the following:

- An AutoFetchData structure containing only a Templates element.
- A UserFetchData structure containing only a Templates element.

This can be rectified by renaming the elements so that their names are different.

ASP.NET Client

If an XML type has attributes, the text element always appears as an array, even if it is not an array. This is rectified by modifying the code manually.

Date Types in XSDs

The SOAP Gateway does not support Date types in XSD External Business Entities.

Jacada Integrator 4.5.3 Release Notes

New Features Included In Release 4.5.3

Platform Support

The following new platforms have been certified as supported for Jacada HostFuse 4.5.3 Server:

- AIX 5.3, 64 bit kernel, 32 bit JDK, JDK version 1.4.2
- Solaris 10, 64 bit kernel, 32 bit JDK, JDK version 1.4.2

MapMaker Fixes

MapMaker Multicast Resource Address

MapMaker has been enhanced to use the res.mcastport property setting in the Resource Server configuration file, ressvr.cfg, to locate Resources Servers during service deployment. If the ressvr.cfg file cannot be read or the res.mcastport property is not defined, MapMaker will use the default Multicast address, 224.0.0.1:35999, to locate Resource Servers.

Custom code generation error

MapMaker now parses custom code with Strings with braces in them correctly.

SOAP Gateway Fixes

SOAP Gateway White Space Preservation Option

An option has been added to the JIWSVC SOAP gateway to allow preservation of white space in SOAP messages. The default value is false.

To enable preservation of white space in JIWSVC SOAP gateway messages, add or uncomment the `whiteSpacePreserve` property in the `JIWSVC.cfg` file and set its value to `true`:

```
whiteSpacePreserve=true
```

SOAP Gateway NoFacilities Exception

The JIWSVC SOAP Gateway now uses the default values of 2, 0 and 10 for the `jclient3Slope`, `jclient3InitialDelay` and `jclient3MaxRetries` properties if they are not explicitly set by the client, This causes the gateway to retry for up to 90 seconds instead of failing immediately if a service request returns a `NoFacilitiesException`.

Example values for jclient3Timeout, jclient3Slope and jclient3Retries The example values for `jclient3Timeout`, `jclient3Slope` and `jclient3MaxRetries` in the JIWSVC SOAP Gateway configuration file `JIWSVC.cfg` have been corrected. The corrected values are:

```
jclient3Timeout=60

jclient3Slope=2

jclient3MaxRetries=10
```

JClient3 Fixes

Services are becoming stuck in a wait state

The problem causing services to be left in the WAITING state when there is an exception thrown while a JClient3 client closes the service connection has been fixed.

Server Fixes

Jacada Services Failure

The HostFuse Environment Manager now uses multiple threads to handle incoming client connection requests, preventing a single client failure from locking up the Environment Manager.

Problem trimming spaces

HostFuse service no longer adds extraneous spaces to null, `NO_VALUE` or `NOT_PRESENT` values returned from methods.

JIStructureType - Assignment Exception

The message added to the service log for an input mapping failure now includes the name or value of the input variable that caused the input mapping failure.

Problem identifying a screen Methods that do more than one level of external invoke now correctly identify the current screen when the externally invoked method returns.

Jacada Integrator 4.5.2 Release Notes

New Features Included In Release 4.5.2

Log file enhancements

The ability to roll over service log files based on the file size has been added.

To set the log to enable rolling over:

- Add the following properties in `<JI_install_dir>/config/envmgr.cfg` file:

  serviceLogfileMaxFileSize - The maximum size to allow the service log to grow to, in bytes, before rolling it over to a backup file.

  serviceLogfileMaxBackupIndex - The maximum number of backup files (starting with 1) to keep for each service log.

  Both properties must be set in order for rollover to be enabled.

> **Note:** Setting `serviceLogfileMaxFileSize` to a small value or setting `serviceLogfileMaxBackupIndex` to a large value has a negative impact on performance.

For example, the following `envmgr.cfg file` is set to enable log rollover with each log file being allowed to grow to 1MB, with up to 4 backup files:

```
#Environment Manager Configuration
serviceLogfileMaxFileSize=1048576
serviceLogfileMaxBackupIndex=4
configname=eaconfig
license=
logfile=envmgr.log
loglevel=0
javahome=C:\Java\jre
javacmd=.\jre\bin\javaw.exe -Xmx256M -Xss1M -
Djava.security.manager -
Djava.security.policy=C:\JI\etc\.java.policy
resourcename=eardb://JI
resourceMCastAddr=239.192.0.0:35999
envMCastAddr=239.192.0.0:35998
```

Support multiple secure host connections

You can now configure multiple secure host connections for MapMaker, both at design time and for the HostFuse runtime environment.

Secure host connections are also supported for 5250 hosts.

To configure multiple secure host connections at runtime:

- Add the following property in `<JI_install_dir>/config/envmgr.cfg` file:

  secureHostAddr - A comma separated list of secure host addresses. For example:

  secureHostAddr=s390p2:992,sslhost:992

  To configure multiple secure host connections at design time:
- Add the following property in `<JI_install_dir>/bin/ea_mapmaker.lax` file:

  secureHost - A comma separated list of secure host addresses. For example:

  ```
  lax.nl.java.option.additional=-
  Dcom.jacada.mapstudio.config.secureHost="s390p2:992,sslhost:992"
  ```

The cipher suite and truststore properties set for the secure host connections apply to all secure host addresses.

XSL transformation support for 4.x XSD inputs and outputs

Support is provided for XSL transformations. If an error occurs when transforming input or output variables, the error is logged and processing continues with the untransformed XML.

XSL transformation of 4.x MLM input and output variables is **not** supported.

To enable XSL transformation of a 4.x XSD input variable:

- The client message must include either the `EA_XSL_IN` or `EA_XSL_IN_URI` variable, where:

  EA_XSL_IN - A string with the XSL stylesheet to be used for transforming the input variables.

  EA_XSL_IN_URI - A string with a URI reference to the XSL stylesheet to be used for transforming the input variables.

Note: Only the first instance of `EA_XSL_IN` or `EA_XSL_IN_URI` are used to specify the input and output stylesheets. Subsequent instances are ignored.
Make sure the input XSL stylesheet generates XML that is valid against the input variable schema.

To enable XSL transformation of a 4.x XSD output variable:

- The client message should include either the `EA_XSL_OUT` or `EA_XSL_OUT_URI` variable, where:

  EA_XSL_OUT - A string with the XSL stylesheet to be used for transforming the output variables.

  EA_XSL_OUT_URI - A string with a URI reference to the XSL stylesheet to be used for transforming the output variables.

> **Note:** Only the first instance of `EA_XSL_OUT` or `EA_XSL_OUT_URI` are used to specify the input and output stylesheets. Subsequent instances are ignored.

Upgrading from 3.5 to 4.5.2

Code Generation Errors

The package name prefix `com.cnt` was changed to `com.jacada` from version 4.0. This caused a number of upgrade problems when generating code:

- Compiling JClient3 code that tried to reference `com.cnt.ea.jclient3/ParameterMetadata.MODE_IN, MODE_OUT` or `MODE_INOUT` failed.
- Compiling user-generated or modified code caused a stack overflow error.

These problems have been fixed.

Change in JClient3 API from 3.5 to 4.5.2

The underlying implementation of the `com.jacada.ea.jclient3.ServiceConnection.invokeMethod` return object was changed from `com.jacada.util.OrderedMap` to `com.cnt.util.OrderedMap` in order to maintain backwards compatibility with version 3.5.

Global Variables Cleared

Maps converted from Jacada Integrator version 3.5 now return the updated value of global variables to the caller and not their 3.5 default values from "Basic Data Operations" methods regardless of what changes have been made to them at runtime.

MapMaker Fixes

Find' option opens method in tree panel

The find option in MapMaker has been fixed so that when a method step is found, the method is opened in the designer panel with the method step selected and the method is opened in the tree panel with the method expanded and the method selected.

When you select a method step in the designer panel; the method is expanded with the method step selected in the tree panel.

> **Note:** This behavior is the same as existed in Jacada Integrator version 4.0.

Scroll tags with escape characters supported

Escape characters, such as non-SPACE whitespace; double quote; backslash; upper ASCII and non-ASCII Unicode characters; double quote and backslash, in scroll tags do not break literal strings when generating code for the map.

Server Fixes

ConcurrentModificationException and NoSuchElementException problems fixed

The concurrency problem in maps loaded in a JCluster has been fixed.

Check for null value for secure host configuration works

Null values entered for the secure host or port now return an error.

NullPointerException problem fixed for threads that are not killed after timing out

The NullPointerException thrown when threads were not killed after timing out has been fixed.

SOAP Gateway Fixes

SOAP Gateway CPU Utilization

The JIWSVC version is always logged at startup, regardless of the logging level set.

Jacada Integrator 4.5.1 Release Notes

Upgrading from 3.5 to 4.5.1

Code Generation Errors

`COM.CNT` was changed to `COM.JACADA` from version 4.0. This caused a number of upgrade problems when generating code:

- Compiling JClient3 code that tried to reference `com.cnt.ea.jclient3/ParameterMetadata.MODE_IN, MODE_OUT` or `MODE_INOUT` failed.
- Compiling user-generated or modified code caused a stack overflow error.

These problems have been fixed.

Change in JClient3 API from 3.5 to 4.5.1

The underlying implementation of the `com.jacada.ea.jclient3.ServiceConnection.invokeMethod` return object was changed from `com.jacada.util.OrderedMap` to `com.cnt.util.OrderedMap` in order to maintain backwards compatibility with version 3.5.

Global Variables Cleared

Maps converted from Jacada Integrator version 3.5 now return the updated value of global variables to the caller and not their 3.5 default values from "Basic Data Operations" methods regardless of what changes have been made to them at runtime.

MapMaker

Data Encryption Between Jacada Integrator and the Host

The data transferred to and from Jacada Integrator by the Host is now encrypted, using SSL encryption.

Generating Service Code That Includes User methods in Custom Classes

When generating service code, and the code to be generated includes custom classes with user methods, the user methods are removed from the generation when the **Overwrite Custom Classes** check box is checked, enabling the code to generate successfully.

> **Note:** If you have user methods, backup the code before generating if you have the **Overwrite Custom Classes** check box checked. After generating add back the user methods.

Evaluation of NOT_PRESENT and NO_VALUE Structures

In a condition step, a value could not being compared to NOT_PRESENT or NO_VALUE structures to determine if it was empty or NULL. This has been fixed.

In addition, both NOT_PRESENT and NO_VALUE now equate to NULL.

Opening a Map Containing Set Steps Using an XPath Expression

Set steps which use an XPath expression containing a method variable as an array index into a global variable array are now valid when the map is first loaded.

Name Changes to Internal Data Types in XPath Expressions

An XPath expression now tracks the name changes made to internal data types that make up the expression.

For example, if the data target of a Set step was an XPath expression: `FOO_IBE/BAR_List[0]/BAZ` and the name of the `BAR_List` element of the `FOO_IBE` structure was changed to `FOOB_LIst`, the `setJITypeDefinition` would be generated incorrectly with the `BAR_List` element name.

XPath and Output Formatting

A modification to the code in the previous version, which was meant to fix a problem with XPath number functions and make it work with booleans, caused output formats to be ignored. This has been fixed.

Formatting Empty Stings into Dates

A NULL pointer exception was thrown when attempting to format an empty string into a date. This has been fixed.

Invalid Arrows in Map Removed

The GUI is automatically cleaned up to remove any invalid arrows.

ASP .NET Client Code Generation Improvements

The following improvements have been made to the ASP.NET client code generation:

- The ASP .NET client code generation would fail if there were spaces in the path of the wsdl/code directories. This has been fixed.
- Client code generation uses the generated WSDL, so it no longer requires deployment of the Web service for client generation.
- Definitions are generated before clients, so that **Generate All** can be used to create an ASP.NET client.
- Headers are created in Web services and set manually.
- The session name in the client generation dialog box is no longer ignored when creating an ASP.NET client.

Tracing Host Data

You can now trace host data within MapMaker.

Action Comments in the Perform Step Combo Box

Action comments in the combo box of a perform step and choose action dialog are now displayed.

MapMaker and the Debugger

Closing the Debugger's JTerm Windows When Loading a New Map

When you change the map, a current connection in the Debugger, if one exists, is now closed.

Global Actions Created for the Debugger

When Global Actions were created for the Debugger an exception was thrown. This has been fixed.

Tags in Map Recognized by the Debugger

Tags used in a map were not recognized by the debugger. This has been fixed.

Message Box Displayed Upon Exception

When a method step fails, a message box opens, indicating an exception has occurred, and directing the user to the location where the exception is written in the session log.

Input Values Not Passed to External Method

Input values were not being passed to the InVars of invoked Methods when using the ExternalInvoke step in the debugger. This has been fixed.

Stopping Methods Doesn't Write Extra Messages to the Log

When a method was stopped in the debugger, the steps were invoked but did nothing, causing unnecessary messages to be written to the session log. Now, method execution stops as soon as it can with no further logging.

MapMaker and JTERM

Support for Field- and Field+ Keys

5250 terminal emulation now supports field- and field+ keys.

Support for Negative Values in Signed Numeric Fields

5250 terminal emulation now supports entering negative values to signed numeric fields.

Client

JTerm Window Instantiated

When running a client with a JTerm, the JTerm window is now instantiated.

Server

Fputs Used to Log the Host Presentation Space

The API for logging the host presentation space has been changed to allow % characters to be handled (fprintf has been changed to fputs, which allows the use of the % character as a text character).

Internal debug comments no longer written to log

Some debug comments were sent to the standard error log every time a client connected. This has been fixed.

Mapping VT100 keys

Enabling *Alternate Function Key Mapping* in the *Advanced Host Communication configuration* dialog box now works both during design time and when running the service. Previously, the runtime service used the default function key mapping regardless of the checkbox setting.

SOAP Gateway

Messages with Long String Fields Passed Through the JIWSVC Gateway

Text fields longer than 70 characters, with embedded spaces, can now be passed to a method in a Jacada service and the message is passed "as is" through the JIWSVSC gateway, without line breaks being added after the spaces. This feature can be controlled using the new configuration property, `indentXMLRequest`, which was added to the configuration file in the JIWSVC .war file. The default value is true, which enables indentation or "prettyprinting" of XML requests sent to the JI service.

WSDL Generation for Methods With Custom Code

The incorrect generation of WSDL for a method that included custom code has been fixed. The method can now be run using the WebServices gateway.

Server Command Line Utilities

ea_start command line utility

In order to start the MySQL database daemon using the `ea_start` command, even when the server is heavily loaded, `ea_start` now waits and retries up to ten times with two-seconds waits between retries.

ea_status command line utility

The exception thrown when running `ea_status` under 1.4.2_03 JDK on Solaris or 1.4.2 ca1420-20040626 build on AIX has been fixed.

> **Note:** The fix also applies to the `ea_ping` command, which is called by ea_status to determine if Tomcat is running.

EAServiceBean

Retrieving the LU Name After Making a Connection

The LU Name negotiated or assigned by the host can now be retrieved using the `EAServiceBean.getLuName()` method.

A private negotiatedLuName field was added to the EAServiceBean class to store the LU Name negotiated by the host without overwriting the configured LU Name.

Session Name Retrieved by EAServiceBean.getSessionName()

The session name in EAServiceBean is now set when initializing a JService (a single thread within a larger process, called a JCluster).

getWriter() method Added

A `getWriter()` method was added to the EASessionBean interface. Thus, you can change the JService to call `EASessionBean.getWriter()` to set its own logger after the `EASessionBean.setWriter(...)` method was called. This makes sure that the JService and the EAServiceBean use the same logger.

Jacada Integrator 4.5 Release Notes

New Features Included in Release 4.5

Following is a list of the new features included in the release of Version 4.5:

- **Debugger:** Enables you to locate and correct errors in a method, without deploying a service. See *JI Integration User's Guide* page 359.
- **General MapMaker enhancements:**
  - **Method variables:** Store data being transferred between the client and the legacy screen, and apply only to the specific method in which they are defined. See *JI Integration User's Guide* page 278.
  - **Method duplication:** Copies a method along with all its properties. See *JI Integration User's Guide* page 274.
  - **Screen Compare:** Compares one screen image with another, or with all screens currently defined in a map. See *JI Integration User's Guide* page 167.
  - **XPath Evaluator:** Verifies the result of an XPath expression, without the need to generate a method or deploy a service. See *JI Integration User's Guide* page 253.
  - **Field properties:** Retrieve properties such as **Name**, **Value**, **Length**, **Color** and **Style** from the legacy screen. See *JI Integration User's Guide* page 195.
- **Remote web services deployment:** Deploys and undeploys web services in any SOAP Gateway running on a remote system. See *JI Integration User's Guide* page 419.
- **ASP.NET client generation:** Enables you to deploy ASP.NET clients to a Web Server with .NET support. See *JI Integration User's Guide* page 412.
- **LINUX support:** The JI documentation contains instructions for UNIX operating systems throughout. These instructions also apply for the SUSE 8 LINUX operating system.

Documentation Reduction and Optimization

The printed versions of the *JI Integration Compatibility Kit User's Guide* and the *JI Integration Compatibility Kit API Function Reference* have been eliminated. The PDF versions of these documents are easily accessible from the Documentation Navigator in the JI installation.

Hardware Platforms and Operating Systems

The following table shows the hardware platforms on which you can run the Jacada Integrator software:

Manufacturer	Series	Operating System
Intel	Pentium-III or equivalent	Windows 2000® with Service Pack 4 Windows XP® Professional with Service Pack 1 (Clients and Java GUIs only, non-production runtime) Windows 2003®
Sun Solaris	UltraSPARC	Solaris 9
HP HP-UX	PA-RISC	HP HP-UX 11.0
IBM AIX	POWER2 and PowerPC Systems	IBM AIX 5.1 / 5.2
IBM MF	z/OS or S390	SUSE 8 LINUX

Windows NT® 4.0 Workstation Limitations

Jacada Integrator is supported on both the server and workstation versions of Windows 2000 and on Windows 2003 server. However, the Windows 2000 Professional version has a designed “hard” limit on the number of simultaneously queued client connections. If this limit is exceeded, some connections will be refused. These operating system versions are only supported for development environments; they are not supported for production deployments.

Note: This limitation also applies to Windows XP Professional installations.

Minimum System Recommendations

The following lists minimum recommended system configurations for both production and development systems for running Jacada Integrator:

Production System

The following lists minimum recommended system configurations for production systems for running Jacada Integrator:

Operating System	CPU Speed	Memory (RAM) Recommendation
Windows 2000 Windows 2003 Windows XP	500 Megahertz Pentium III	512 MB
Sun Solaris 9	Any UltraSPARC System	512 MB
HP HP-UX 11.0	Any PA-RISC System	512 MB
IBM AIX 5.1 / 5.2	Any POWER2 or PowerPC System	512 MB
IBM MF	z/OS or S390	512 MB

Development System

The Jacada Integrator Java GUIs are only supported on Windows. MapMaker Service development and Jacada Integrator configuration and monitoring must be done from a Windows machine.

Operating System	CPU Speed	Memory (RAM) Recommendation
Windows 2000	500 Megahertz	512 MB
Windows 2003	Pentium III	
Windows XP Professional		

Disk Space Requirements

The following table lists disk space requirements for a typical installation based on the operating system:

Operating System	MegaBytes Required
Windows 2000	200 MB
Windows 2003	
Windows XP Professional	
Sun Solaris 9	200 MB
HP HP-UX 11.0	200 MB
IBM AIX 5.1 / 5.2	200 MB
IBM MF SUSE 8 LINUX	200 MB

Please verify that there is enough disk space on your selected drive prior to installing the product.

Note: If the provided Windows JRE is used, please add an additional 20 MB to the **MB required** figure.

Temporary Directory for Installation

The Jacada Integrator installation program temporarily extracts data into a temporary directory. Approximately twice the disk space listed above must be available in the temporary directory. These files are removed from the temporary directory after installation is successfully completed.

- **UNIX.** In UNIX, the installation program checks to see if enough space is available in the */tmp* directory. If there is not enough space, an error message is printed to the screen instructing the user to clear space in the */tmp* directory.
- **Windows.** In Windows, the installation program extracts the file to the directory identified in the %TMP% environment variable. If there is not enough disk space on the temporary directory's disk partition to extract the data, the user is prompted to select another drive and directory for temporary extraction.

System Information

TCP Ports

Jacada Integrator requires three TCP ports for Jacada Integrator service components. Default port numbers are provided during installation, but these defaults may need to be modified to ensure that they are set to available ports on your server. These ports can also be given symbolic names in your *services* file (or NIS/YP, if appropriate), which serves to "allocate" these port numbers as far as the server is concerned. However, Jacada Integrator programs cannot use the symbolic names directly; port numbers must be used.

The following table lists the default port numbers allocated during installation for each service component:

Service Component	Default Port at Installation
Resource Database	30000
Environment Manager system port	30001
RMI port	30002

Service Component	Default Port at Installation
Tomcat Server (optional)	8080

Software Requirements

The following software is required to use Jacada Integrator.

Java Software Requirements

The Java Virtual Machine (JVM) 1.4.2_05, or compatible JRE/JDK, is required to run Jacada Integrator Java applications. On Windows, the 1.4.2_05 JRE can be installed during the Jacada Integrator installation.

- For Windows, Sun's JDK/JRE version 1.4.2_05 is required (use of the optional HotSpot Server VM is not recommended).
- For Solaris, Sun's JDK/JRE version 1.4.2_05 is required (use of the optional HotSpot Server VM is not recommended).
- For HP-UX, HP's JDK version 1.4.2 is required.
- For AIX, IBM's JDK version 1.4.2 is required.
- For SUSE 8 LINUX, IBM's JDK version 1.4.2 is required.

To use the MapMaker graphical development environment's code generation capabilities and/or the Tomcat Server, the Java Development Kit (JDK) is required, and should be pre-installed before installing Jacada Integrator.

Note: Due to the nature of Java technology where objects are exchanged across a network, it is recommended that all Jacada Integrator Java-based programs, regardless of platform, be run using the same version of the Java Virtual Machine, in order to avoid any JVM version incompatibilities.

Jacada Integrator Licenses

One or more Jacada Integrator licenses are required to use the Jacada Integrator environment. A licensing mechanism is included with Jacada Integrator that controls the following aspects of the Jacada Integrator environment:

- The capability of the MapMaker development tool to modify and generate Jacada Integrator Services.
- The number of concurrent servers (host machines) that are running Jacada Integrator Environment Managers. This limit is not tied to a specific server

machine, which allows for flexibility in Environment Manager deployment. As long as the total number of Environment Managers in the User environment does not exceed the license limit, the Environment Managers can be running on any server in the network.

- The number of concurrent Jacada Integrator Services running on each server. (Optional license mechanisms are available for customer requirement flexibility.)

For more information, see the *JI Integration User's Guide*.

Note: The licenses provided with Jacada Integrator 3.5 will not work with the 4.5 release. Consult Jacada Integrator Technical Support to obtain proper 4.5 licenses.

X11-Compatible Window Server

UNIX only: To use Jacada Integrator Graphical User Interfaces (GUIs), an X11-compatible window server is required.

JDBC Driver Included in Installation

A required JDBC driver, tjFM, is included with installation of Jacada Integrator and is required to communicate with the Resource Database. The tjFM driver cannot be removed from Jacada Integrator and may not be used outside of Jacada Integrator except by acquiescence to tjFM's terms of agreement.

SUNWi1of Package

Sun Solaris only: In order to use Jacada Integrator GUIs with Sun Solaris, we strongly recommend that you install the SUNWi1of package. This package is a patch provided by Sun to install additional fonts that are used by MapMaker.

Motif Requirements

If you are running Solaris, you must install Motif runtime libraries to enable Jacada Integrator GUIs.

Note: For Solaris, add /usr/dt/lib to your `LD_LIBRARY_PATH` environment variable.

EA_ENV Environment Variable

In order to use the *ea_start* and *ea_shutdown* executables, along with other administrative executables, the EA_ENV environment variable should be set to the *<JI_install_dir>/config* directory. If this environment variable is not set, *ea_start* and *ea_shutdown* must be started using the command line option `-c <configuration directory>`.

Also, the EA_ENV environment variable must be set when using the compatibility kit, and can be used by certain Jacada Integrator clients.

The environment.ccf File

The *environment.ccf* file must be located in the directory referenced by the `EA_ENV` environment variable. The *environment.ccf* file must contain settings for the `EA_HOME`, `MYSQL_TCP_PORT`, `MYSQL_UNIX_PORT`, and `EA_ENVMGR` parameters:

- **EA_HOME:** The root directory of your Jacada Integrator installation. (For example: *C:\JI*)
- **MYSQL_TCP_PORT:** The TCP port number of your MySQL daemon.
- **MYSQL_UNIX_PORT:** UNIX only. The UNIX domain socket for your MySQL daemon.
- **EA_ENVMGR:** The port number of the Environment Manager, used to determine if the Environment Manager is running.
- **START_TOMCAT:** Specifies whether the embedded Tomcat servlet engine is started and stopped, by default, with the `ea_start` and `ea_shutdown` commands. The default setting is `"true"`.

Additional parameters in the *environment.ccf* file are used by the Compatibility Kit portion of Jacada Integrator and clients developed with certain Jacada Integrator client libraries.

Also, the `EA_HOME`, `MYSQL_TCP_PORT`, `MYSQL_UNIX_PORT` (UNIX only), and `EA_ENVMGR` parameters can be defined as environment variables, if the *environment.ccf* file is not used.

DISPLAY Environment Variable

On UNIX, when using Jacada Integrator graphical interfaces or opening a terminal window, the DISPLAY environment variable must be set to the appropriate IP address or host name of the machine on which your X-server software is running. This variable is set using the format `<host>:0.0` where `<host>` is the name or IP address of the host machine.

For example:

```
export DISPLAY=127.0.0.1:0.0
```

THREADS_FLAG Environment Variable

On UNIX, the THREADS_FLAG environment variable must be set to native.

For example:

```
export THREADS_FLAG=native
```

TERMINFO Environment Variable

On HP_UX, the TERMINFO variable must be set for proper function key behavior within PATerm terminal windows.

Set the TERMINFO environment variable to the *<JI_install_dir>/lib/terminfo* directory.

Localhost

The host *localhost* must be defined in your *etc/hosts* file in UNIX (or NIS/YP, if appropriate) or *c:\Winnt\System32\drivers\etc\hosts* file in Windows NT. This is the default host for Jacada Integrator. The loopback IP address generally is 127.0.0.1.

Directory Names that Contain Spaces

Because of the way the MySQL daemon and Java interpret spaces in directory paths, do not use spaces in the name of the directory for your Jacada Integrator installation. You should also not install Jacada Integrator into the Windows *c:\Program Files* directory or any directory tree that has spaces in the names of the directories. This restriction applies to both Windows and UNIX installations.

Communication Protocol Support

Jacada Integrator supports TCP/IP communication protocols for client connections and the following communication protocols for mainframe connections:

- Telnet
- TN3270
- TN5250

See chapter 3 in the *JI Integration Supplemental Reference Guide* for keyboard mapping information. Protocol agent configuration information is located in the *<JI_install_dir>/lib/ProtoAgents* directory, where <JI_install_dir> represents the directory in which Jacada Integrator has been installed.

Supported Client and Integration Environments

The following client libraries and interfaces are available with Jacada Integrator:

- The Java Client Library for Java 2 (JClient3)
- The ActiveX Client Library (AXCL) [Windows only]
- The SOAP Gateway
- The MQ-Series Gateway
- Siebel Integration (using XML/HTTP Gateway)
- PeopleSoft Integration (using XML/HTTP Gateway)

For more information about Jacada Integrator client libraries, see the *JI Integration Client Developer's Guide*.

Migrating from Jacada Integrator Release 3.5 to 4.5

When migrating from earlier Jacada Integrator installations, databases must be exported and then re-imported into Release 4.5.

To upgrade from Jacada Integrator 3.5 to 4.5 the following steps should be taken:

1. Start the 3.5 version of MySQLD.
2. Export the 3.5 database to a file using the utility `ea_exportdb`.

   To export the standard data, use this command (all one statement):
   ```
   ea_exportdb ACLDetail ACLMaster EMConfig Hosts
   <export_file>
   ```
   The ACLDetail and ACLMaster tables may be omitted if there is no Access Control configured in the environment being upgraded. The JSvcDetail and

JSvcMaster tables can also be exported; however, it is recommended that new service master records be created when the services are upgraded.

If legacy C services have been configured, the legacy data should be exported with this command:
`ea_exportdb -l <legacy_export_file>`

3 Shutdown the 3.5 server processes including the MySQL daemon, Resource Server, and Environment Manager processes.

4 Copy the appropriate configuration file settings from the 3.5 installation to the 4.5 configuration files. The configuration files can be found in the `<JI_install_dir>\config` and `<JI_install_dir>\lib\ProtoAgents` directories.

5 Startup the 4.5 version of the MySQL daemon.

6 Import the 3.5 database file into the 4.5 MySQL database using the utility `ea_importdb`. If legacy services are being upgraded, import the 3.5 legacy database file.

7 Start the 4.5 environment; i.e., start up the 4.5 version of MySQLD, the Resource Server, and the Environment Manager.

Note: If the 3.5 and 4.5 environments are to be run in parallel, the Environment Manager configuration (configuration name, load balance group name, and ports) must be modified before starting the Environment Manager.

8 Start the 4.5 GUIs as necessary.

9 Upgrade the Jacada Integrator Clients and Services.

Upgrading Jacada Integrator Clients and Services

Pre-existing Jacada Integrator clients and services must be recompiled with the required release 4.5 versions of the respective client or service libraries. Pre-existing MapMaker maps should be opened within the 4.5 version of MapMaker, updated, and new Java services should be generated and deployed.

Please contact Jacada Integrator Technical Support for more information regarding upgrading to Release 4.5 from a pre-4.5 environment.

Service Library Changes

The following Service API changes exist in Jacada Integrator 4.5:

- `GBMethod.invokeMethodObject()`
  `GBBasicMethodObject.invokeMethodObject()`
  `GBBasicMethod.invokeMethodObject()`

  A three parameter version of the `invokeMethodObject()` method call has been added. The two parameter method will still be available for compatibility. However, Jacada Integrator now calls the three parameter version by default. This change may require a slight change to the user's custom code.
- `GBBasicMethodObject.next()`
- MapMaker 4.5 now features explicit Start and Stop steps. This may require a code change for user's custom code that uses this method.
- `GBStreamDataTemplate.isWriteField()`
- A slight change was made to this method. If the user's custom code creates a `GBStreamDataTemplate`, or there were no writable fields in the DataTemplate, `isWriteField` returned TRUE. The API now returns FALSE. This is the correct behavior for this API. This change may require a slight change to the user's custom code.
- `GBBasicMethod.getMethodSteps()`
- MapMaker 4.5 now features explicit Start and Stop steps. This may require a code change for user's custom code that uses this method.
- `GBBasicMethod.getInitialMethodObject()`
- MapMaker 4.5 now features explicit Start and Stop steps. This may require a code change for user's custom code that uses this method.

Migrating from Jacada Integrator Release 4.0 to 4.5

The recommended procedure is to install JI version 4.5 on top of version 4.0, and not in the default path provided in the installation program. However, if you choose not to follow this recommendation, you have two options:

- The first option is to keep 4.0 in its original location, and install 4.5 in a different location.
- The second option is to uninstall version 4.0, and then install 4.5 in a different location.

Note: In all cases, you must click the **Restore Default** button in the **Java** tab of the MapMaker **Properties** dialog box, after performing the installation.

To keep JI 4.0 in its original location, and install JI 4.5 in a different location:

1 Perform the installation procedure.
2 Copy and rename the *.mapmaker40.cfg* file, located in your temporary directory, as an additional installation requires an additional configuration file.
3 Edit the *<JI_install_dir>\bin\ea_mapmaker.lax* file, as follows:

   In the `lax.command.line.args` section, add the `-f` parameter followed by the path to the new configuration file, as shown in the following example:

   lax.command.line.args=$CMD_LINE_ARGUMENTS$ -f "c:\Documents and Settings\username\.mapmaker45.cfg"
4 In the MapMaker **Properties** dialog box, replace the old installation path with the new installation path in the following locations:
   - In the **General** tab, replace the path in the **3270 KeyMap Enabled** field, the **5250 KeyMap Enabled** field and the **Character Mode Terminfo File** field.
   - In the **Default Directories** tab, replace the path in the **MapMaker Install Directory** field.
5 Also in the MapMaker **Properties** dialog box, click the **Restore Default** button in the **Java** tab.

After uninstalling JI 4.0, and then installing JI 4.5 in a different location:

- Edit the information in the MapMaker **Properties** dialog box, as described in steps 4 and 5 of the previous procedure.

Known Limitations

MapMaker Thread Step

An exception on the main Thread is always sent to the client, but an exception on a Thread branch is not. Use an OnFail step to handle Thread branch exceptions.

Arrays in a MapMaker Set Step

When using a Set step to fill an empty array, the Add operator appends the new value after the “NoValue” element in the first array index. Use the Set operator to replace the “NoValue” element with the new value.

The same principle applies when adding a structure array with 0 elements to a structure array with 1 element. This returns a structure array with 2 elements because a Global variable has a value of “Not Present”, and adding an additional value appends instead of overwrites the “Not Present” value. Use the Set operator instead of the Add operator to overwrite the “Not Present” value.

JVM Path

When installing Jacada Integrator, the path specified for the JVM must be a full path to a JDK or JRE installation. Symbolic links are not supported in this path.

Array Indexes

In an XPath expression, the first element in an array is element [1]. In a Global variable, the first element in an array is element [0].

JAVA_HOME Variable

The JAVA_HOME environment variable used in Tomcat startup is calculated from the envmgr.cfg file, using the envmgr.cfg java command for the JCluster and not the general environment JAVA_HOME variable.

Window Sizes in MapMaker

The window sizes of the Structure Relationship Editor and Business Entity Editor are in proportion to the size of the MapMaker main window. This means that when the MapMaker main window is made small, the Structure Relationship Editor and Business Entity Editor open in very small and unusable windows. These windows can only be enlarged after the MapMaker main window is enlarged.

Renaming Fields in IBEs

When renaming a field in an Internal Business Entity, the field name is not updated in some of the more complex XPath expressions referencing that field.

Data Mapping Editor Columns

When the data contained in the Mapped Data table is too long to be fully displayed in a column, it gets an ellipses rather than a horizontal scroll bar. The full version of the data can be viewed in the particular cell's tooltip.

Modifying a Data Template

Modifying data templates in screens that are already used by particular methods can affect those methods. You must update such methods to account for any changes made to the data template.

Modifying a Traverse Step

In a method containing a Traverse step linked to a Perform step, changing the current screen for the Traverse step does not modify the action in the Perform step. The action can be modified by clicking the **Reset Current Screen** button in the **General** tab of the Perform step properties.

Incomplete Validation in ForCondition Step

When entering the invalid combination of a string with an internal number for comparison in the **Conditional** tab of the ForCondition step properties, the **Numeric** checkbox is incorrectly enabled and selected. The correct behavior would be for an error message to be displayed, pointing out the invalid use of a non-numeric value.

Attributes in XPath Expressions

The XPath Evaluator does not support attributes in XPath expressions.

Deleting Global Variables

In the Business Entity Editor, deletion of Global variables using the Delete key on the keyboard is not recommended.

Modifying Structure Name of Thread Step

The MapMaker Presentation view does not update the modification of a structure name in the **General** tab of the Thread step properties.

Choose a Datasource Dialog Box

The **Current Type Definition** field in the **Choose a Datasource** dialog box retains the value entered even when that value is invalid.

Replacing Invoked Methods

When replacing an invoked method with a method of a different external type, the data mappings are not updated. This is rectified by manually deleting the old data mappings and adding new data mappings in the **Data Mapping Editor**.

MapMaker Comments Tab

MapMaker does not retain modifications made in the **Comments** tab of the Properties view, unless the **Apply** button is clicked before clicking another area of the GUI.

Modifying MapMaker Look and Feel

The **Appearance** tab of the **Properties** dialog box enables you to modify the look and feel of MapMaker.

- In some cases, this feature displays dialog boxes incorrectly, in that the **OK** and **Cancel** buttons are partially out of the window's scope.
- When changing the look and feel from one option to a second option and then back again, the second look and feel remains in some dialog boxes. This is rectified by restarting MapMaker after changing the look and feel.

MapMaker Properties Dialog Box

The **Cancel** button in the MapMaker **Properties** dialog box (accessed from the **File** menu) does not work. Clicking the **Cancel** button applies changes. Changes can only be cancelled by reapplying the previous settings.

MapMaker Update Map Button

After clicking the **Update Map** button, the Properties view shows the properties of a Start method step regardless of the tab currently selected in the Tab view.

JClient3 Step

A service opened by a JClient 3 step is not disconnected when a service open fails.

If a client that is a JClient3 step in a JI service attempts to open a service without first connecting to the host, the service open fails leaving the service connected but not open. If there is a subsequent attempt to open the service, it will also fail because the service is already connected and there is no means of recovery without terminating the service.

MapMaker Set Step

The tooltip for a Set step does not show the specified array indexes for the left-hand side Global variable expression.

Copying Types in the Business Entity Editor

Copying a Type that contains a subtype, sometimes causes the subtype to be copied. When this happens, the subtype is copied without its sample values. Sample values can be added manually in the **TypeDataModel Viewer**.

OnFail Links

MapMaker incorrectly allows OnFail links to be attached to steps that are on branches. Do not use OnFail links in this way.

Curly Braces in Code Generation

If a line of code contains only a curly brace, "{", on its own, do not put a comment on the previous line. The comment can be placed two lines above the curly brace.

Auto Complete Feature in Set Step Properties

The Auto Complete feature is activated indiscriminately in the Set Step's **General** tab. The Auto Complete feature provides variable names beginning with the letter you type into the right-hand object name field.

However, if you select a type that is not a variable from the **Data Source** dropdown list, such as **InternalString**, and then start to enter the object name in the next field, the field incorrectly auto completes with variable names, even though the object you are specifying is not a variable.

Deploy to Resource Server Dialog Box

The **Web Services** tab in the **Deploy to Resource Server** dialog box contains the following limitations:

- When **Services** is selected in the **Manage** dropdown list, the **Undeploy** column incorrectly appears to be enabled.
- The **Undeploy** action does not apply to the list of services.
- When **Deployed Web Services** is selected in the **Manage** dropdown list, the **Deploy** column incorrectly appears to be enabled.
- The **Deploy** action does not apply to the list of deployed web services.

ASP.NET Business Entities

ASP.NET client services will not work if any method contains a Business Entity that contains two structures, both of which contain one element, and that one element has the same name in both cases.

For example, performing the Copy To action on the ErrorReport type Internal Business Entity, results in an External Business Entity that contains the following:

- An AutoFetchData structure containing only a Templates element.
- A UserFetchData structure containing only a Templates element.

This can be rectified by renaming the elements so that their names are different.

ASP.NET Client

If an XML type has attributes, the text element always appears as an array, even if it is not an array. This is rectified by modifying the code manually.

Special Instructions

Convert/Update MapMaker Maps - Maps created with a Jacada Integrator 3.5 version of MapMaker need to be converted for use within Jacada Integrator 4.5. Jacada Integrator 4.5 includes a conversion utility for MapMaker 3.5 maps. This command-line utility is named “ea_convertmap” and is found in the <JI_install_dir>\bin directory.

Usage is:

```
"ea_convertmap My_JI35_Map.map"
```

The MapMaker 3.5 file will be backed up to the same name with a ".old" extension added to it. The converted map will retain the existing original name. The converter will output various diagnostic messages. The last couple lines will indicate the success of the conversion. Once the conversion is complete, the converted map should be loaded into the 4.5 MapMaker, updated using the Update Map feature, and the resulting services regenerated and re-deployed into the Jacada Integrator 4.5 environment.

Service Re-compilation - All services and all clients must be recompiled using the Jacada Integrator version 4.5 libraries. Services and clients must also be redeployed.

XML CDATA Handling for Jacada Integrator 3.x Methods - Text data appended within XML CDATA areas, such as `<![CDATA[<EM>BOLD</EM>]]>`, is currently appended to the text immediately preceding the CDATA item.

Java Heap and Stack Size Settings - These options are the standard Java command line options -Xmx and -Xss and they are set within the .lax file for the associated Jacada Integrator command. Most Jacada Integrator commands are found in the *<JI_install_dir>\bin* directory. Options for the JCluster are set in the *<JI_install_dir>\config\envmgr.cfg* file. These settings are set larger than the Java defaults for some of the memory intensive Jacada Integrator programs. These settings may require further tuning in some situations. See the appropriate Java documentation for more information about these Java options. For more information about editing LAX files, see the *JI Integration Supplemental Reference Guide*.

Configuration File Settings - Items in Jacada Integrator configuration files are generally case-sensitive; all settings must be entered with the proper case.

DNS Lookup Option - An option is available in the Environment Manager configuration file. This option turns off reverse DNS lookups for Jacada Integrator client connections in the Environment Manager and JCluster. Turning this option off usually results in better server performance, however client host names will be output to log files in IP address format. This option is named `dnsLookup` and it should be set to true for normal DNS lookup behavior and to false for disabled lookups.

HP-UX Alert - For users running Jacada Integrator software on HP-UX, it is strongly recommended that the Jacada Integrator software be installed on a non-NFS mounted partition.

Database Changes and Database Import/Export Concerns

Existing customers may export and re-import their database(s) when upgrading to Jacada Integrator release 4.5. See the *JI Integration Installation and Configuration Guide* for information on exporting and importing databases.

The ea_exportdb and ea_importdb utilities on Windows have a base directory of *<JI_install_dir>\bin*. This is due to an InstallAnywhere launcher limitation. Relative path names specified to these utilities will be relative to *<JI_install_dir>\bin*. It is recommended that these utilities be run from the *<JI_install_dir>\bin* directory, or that absolute path names be used, such as `C:\my_rdb.txt`.

Character Mode

Support for the DEC VT emulation for special characters and line drawing character set is not enabled by JDK from Sun. These characters are displayed using pixel drawing. This method has a small performance impact at design time (when using MapMaker).

Certain types of host applications, such as the vi text editor, are not suitable for MapMaker's trail-based representation.

Double-Byte Data Streams

Only Japanese double-byte data streams are supported at this time, and only locales supported by Java 1.2+ are supported. Support may also be limited based on the hardware and its accompanying operating system. Such as:

Windows NT

On Windows, 8-bit European characters and 16-bit Asian characters are entered using an escape sequence, or by using an international keyboard. Escape sequences are usually as follows:

Hold down right-Alt and enter a 4-digit value using the numeric keyboard.

Solaris

On Solaris, 8-bit European characters and 16-bit Asian characters are entered using "compose" sequences. Using an international keyboard may make this process easier.

Japanese versions of Windows and Solaris will have an Input Method of some type to allow the user to compose a character outside of MapMaker and send the composed key sequence into MapMaker when it is finished. The Input Method consist of a pop-up dialog box (supplied by the OS) where the user composes the Asian character. Each O/S may supply different Input Methods, but they all operate in generally the same way. The user should be familiar with the Input Methods on their particular system.

> **Note:** It is difficult to find English documentation for this functionality.

User Interaction Mode

The User Interaction Mode functionality does not provide for automatic starting of the PATerm. The PATerm must be running before User Interaction Mode is needed.

SOCKS

SOCKS does not provide for an encryption. All client server data is passed “as is”. Also, SOCKS does not provide a C-based library on Windows; the JClient3 is the recommended client for using the SOCKS protocol.

Java Runtime Environment (JRE) Compatibility

Due to a known problem with serialization incompatibility between different versions of Java, it is recommended that all Jacada Integrator Java-based programs throughout your network, regardless of platform, should be run using the same version of Java to avoid any version incompatibilities.

MapMaker Issues

Windows Java Compiler not Case Sensitive

In Windows only, the Java compiler is not case sensitive when replacing existing *.*class* files. After generating code, when MapMaker objects are renamed by changing only the case of characters in the object’s name, User must delete the *.*class* files from the output directory before regenerating code. This will assure that the compiler uses the correct case for the characters of the class name when in recreates the *.*class* file.