

# **FUJITSU Software Interstage Studio**

A horizontal band with a red abstract graphic featuring glowing, overlapping lines and curves, creating a sense of motion and energy.

## **Release Notes**

B1WD-3156-02ENZ0(00)  
November 2013

# Preface

---

## Purpose of This Manual

This manual provides an overview of additional functions and information related to compatibility and fixes.

## Intended Readers

By reading this manual, you will understand the information about additional functions and the functions that are no longer provided in this version and level.

## Structure of this Manual

This manual consists of the following chapters:

- [Chapter 1 What's New in this Version](#)

This chapter explains the functions that have been added since the previous version.

- [Chapter 2 Compatibility Information](#)

This chapter explains changes in the operating methods and resource handling since the previous version.

- [Chapter 3 Functions No Longer Provided](#)

This chapter explains the functions that are no longer provided in this version and level.

- [Chapter 4 Fixes in this Version](#)

This chapter explains the fixes adopted in this version for faults that occurred in the previous version.



## Information

---

### **Additional functions, functions that are no longer provided, and fix information for Interstage Application Server**

Refer to the "Interstage Application Server Release Notes" for information on additional functions, functions that are no longer provided, and fix information for Interstage Application Server.

---

## Conventions

This product creates a group name of [Interstage Studio Vxx] on the Apps screen (in Windows 8 and Windows Server 2012) and the Start menu (in other Windows systems). Check the actual group names in the "Software Release Guide" and substitute those names when reading this document.

## Export Controls

This document or a portion thereof may not be exported (or re-exported) without authorization from the appropriate government authorities in accordance with the pertinent laws.

## Trademarks

- Microsoft, Active Directory, ActiveX, Excel, Internet Explorer, MS-DOS, MSDN, Visual Basic, Visual C++, Visual Studio, Windows, Windows NT, Windows Server, Win32 are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.
- Other company and product names in this documentation are trademarks or registered trademarks of their respective owners.

## Revision History

Date of Publication and Version	Manual Code
November 2013: 2nd Version	B1WD-3156-02ENZ0(00)/B1WD-3156-02ENZ2(00)

Date of Publication and Version	Manual Code
November 2012: 1st Version	B1WD-3156-01ENZ0(00)/B1WD-3156-01ENZ2(00)

**Copyright Notice**

Copyright 2012-2013 FUJITSU LIMITED

# Contents

---

Chapter 1 What's New in this Version.....	1
1.1 Java Integrated Development Environment.....	1
1.2 J Business Kit.....	2
1.3 JDK/JRE.....	2
1.4 FJQSS (Information Collection Tool).....	3
Chapter 2 Compatibility Information.....	4
2.1 Changes from the Functions of V11.0.....	6
2.1.1 Java EE 6 Development Functions.....	6
2.1.2 Apcoordinator Development Functions.....	7
2.2 Changes from the Functions of V10.0.....	7
2.2.1 Basic Functions.....	7
2.2.2 J2EE Development Functions.....	7
2.2.3 J Business Kit.....	8
2.2.4 Installer.....	10
2.3 Changes from the Functions of V9.2.....	10
2.3.1 Basic Functions.....	10
2.3.2 Installer.....	11
2.4 Changes from the Functions of V9.1.....	11
2.4.1 Installer.....	11
2.5 Changes from the Functions of V9.0.....	12
2.5.1 Basic Functions.....	12
2.5.2 J2EE Development Functions.....	13
2.5.3 Apcoordinator Development Functions.....	13
2.5.4 Installer.....	13
2.5.5 Online Manuals.....	13
2.6 Changes from the Functions of V8.....	14
2.6.1 Basic Functions.....	14
2.6.2 J2EE Development Functions.....	14
2.6.3 Java Development Functions.....	16
2.7 Changes from the Functions of V7 and Earlier.....	17
2.7.1 Basic Functions.....	17
2.7.2 J2EE Development Functions.....	17
2.7.3 Java Development Functions.....	18
2.7.4 J Business Kit.....	18
2.8 Migration from the Previous Version.....	23
Chapter 3 Functions No Longer Provided.....	24
Chapter 4 Fixes in this Version.....	25
4.1 Java Integrated Development Environment.....	25
4.2 J Business Kit.....	25
4.3 JDK/JRE.....	27
Index.....	29

# Chapter 1 What's New in this Version

This chapter describes the functions added to this product since the previous version.

## Notation

This chapter uses the following terms in headers of tables that outline the added functions:

No.	Version	Function name	Description	Reference manual
-----	---------	---------------	-------------	------------------

No.

Sequence number

Version

Version to which the function is added

Function name

Name of the added function

Description

Description of the added function

Reference manual

Section in the manual providing information on the added function

## 1.1 Java Integrated Development Environment

No.	Version	Function name	Description	Reference manual
1	V11.1	Function for the development of applications that use Java EE 6	Enables IJServer Cluster to be added to the Servers view in the Java EE 6 workbench.	Interstage Studio User's Guide "Developing Java EE 6 Applications" > "Common Subject Matter Regarding Java EE 6 Applications" > "Preparing to Create Applications"
2	V11.1	Function for the development of applications that use Java EE 6	Enables a remote server to be added to the Servers view in the Java EE 6 workbench.	Interstage Studio User's Guide "Developing Java EE 6 Applications" > "Common Subject Matter Regarding Java EE 6 Applications" > "Preparing to Create Applications"
3	V11.1	Interstage Management Service Operation Tool	Provides the Interstage PCMI(isje6) Service and the Interstage HTTP Server 2.2 Service, which use the Interstage Management Service Operation Tool.	Interstage Studio Release Notes "Compatibility Information" > "Changes from the Functions of V11.0"
4	V11.1	Java SE 7 function	Provides a support function in the Java EE 6 workbench for the development of Java SE 7 applications. This function can be used to edit Java Form, Applet, and JavaBeans.	Interstage Studio User's Guide "Developing Java Applications"
5	V11.0	Function for the development of	Provides support functionality in the workbench for the development of J2EE applications that use Java SE 6.	Interstage Studio User's Guide "Migrating J2EE 1.4 Applications"

No.	Version	Function name	Description	Reference manual
		J2EE applications that use Java SE 6		
6	V11.0	Database linkage function	<p>The database linkage function enables linkage with the following products:</p> <p>&lt;Standard workbench&gt;</p> <ul style="list-style-type: none"> <li>- Microsoft(R) SQL Server 2012</li> </ul> <p>&lt;Java EE 6 workbench&gt;</p> <ul style="list-style-type: none"> <li>- Oracle Database 11.2</li> <li>- Microsoft(R) SQL Server 2012</li> <li>- Derby 10.8</li> </ul> <p>Note: JDBC driver versions that support JDK 6 are supported.</p> <p>Note: The Java EE 6 workbench database linkage function should be used in the development of applications using JPA.</p>	<p>Interstage Studio User's Guide</p> <p>"Database Operation" &gt; "Tasks" &gt; "Supported Database Information"</p>

## 1.2 J Business Kit

No.	Version	Function name	Description	Reference manual
1	V11.0	GUI Library	Provides functionality for notification of automatic exit in character input parts as JFAutoEscapeEvent.	J Business Kit GUI Library User's Guide "Embedded character string field Bean" "Character string field Bean" "Text area Bean"
2	V11.0	GUI Library	Provides functionality for enabling double-click in JFImageButton.	J Business Kit GUI Library User's Guide "Image button Bean"

## 1.3 JDK/JRE

No.	Version	Function name	Description	Reference manual
1	V11.1	JDK/JRE	<p>The following versions of JDK/JRE are provided.</p> <ul style="list-style-type: none"> <li>- 7 Update 9</li> <li>- 6 Update 37</li> </ul> <p>However, Mozilla Rhino, Java DB, Java Web Start (JWS), and Java FX functions are not provided.</p>	<p>Java SE 7 Documentation</p> <p>Java SE 6 Documentation</p>
2	V11.0	JDK/JRE	<p>The following versions of JDK/JRE are provided.</p> <ul style="list-style-type: none"> <li>- 7 Update 3</li> </ul>	<p>Java SE 7 Documentation</p> <p>Java SE 6 Documentation</p>

No.	Version	Function name	Description	Reference manual
			- 6 Update 31 However, Mozilla Rhino, Java DB, and Java Web Start (JWS) functions are not provided.	

## 1.4 FJQSS (Information Collection Tool)

---

No.	Version	Function name	Description	Reference manual
1	V11.0	FJQSS (Information Collection Tool)	This tool is used to simplify the collection of resources required for troubleshooting.	Software Release Guide "Installation and Uninstallation"

## Chapter 2 Compatibility Information

This chapter describes the functions that were modified, in their method of operation or method of retaining resources, from the previous version. Refer to the description of changes in the section corresponding to the version and level of the previously used product. Note that Interstage Studio previous versions and levels include Interstage Apworks V8 and earlier.

Function	Contents	Version before Migration						
		V7	V8	V9	V9.1	V9.2	V10	V11
Basic	Development on the Compatible Workbench provided in V10	-	-	Y	Y	Y	Y	-
	Workbench virtual memory size	-	-	-	-	Y	Y	-
	Some of the Java Form features are now deprecated features	Y	Y	Y	Y	Y	Y	-
	Integration of Interstage Management Service Operation Tools	-	-	-	-	Y	-	-
	Changes to the database name used by the Interstage directory service	Y	Y	Y	Y	Y	-	-
	Managing the resident service required for the management of the local server	Y	Y	Y	-	-	-	-
	Predefined library "Fujitsu XML"	Y	Y	Y	-	-	-	-
	Project wizard configuration	Y	Y	-	-	-	-	-
	Reconfiguration of categories for new wizards	Y	-	-	-	-	-	-
	Projects outside the workspace folder	Y	-	-	-	-	-	-
Java EE 6	Using Interstage Management Service Operation Tool to operate services used by Java EE 6	-	-	-	-	-	Y	Y
	Default value of JDK version used in Java EE 6 function	-	-	-	-	-	Y	Y
J2EE	J2EE Perspective	-	-	Y	Y	Y	Y	-
	Deployment Descriptor (EJB) wizard	-	-	Y	Y	Y	Y	-
	DD (Deployment Descriptor) Editor	-	-	Y	Y	Y	Y	-
	EAR function for rapid deployment	-	-	Y	Y	Y	Y	-
	No support for the J2EE 1.3 format	Y	Y	Y	Y	Y	Y	-
	Changing the JServer type at the default deployment destination	Y	Y	Y	-	-	-	-
	Non-support of the J2EE 1.3 format application client	Y	Y	Y	-	-	-	-
	Project creation for J2EE application development	Y	Y	-	-	-	-	-
	Project creation for J2EE application client development	Y	Y	-	-	-	-	-
	Changes to J2EE-related wizards	Y	Y	-	-	-	-	-
	Changes to builders	Y	Y	-	-	-	-	-
	JSP tag input support in the JSP editor	Y	Y	-	-	-	-	-
	Breakpoint setting in the JSP editor	Y	Y	-	-	-	-	-
	Updating the deployment descriptor version	Y	Y	-	-	-	-	-
No support for the J2EE 1.2 format	Y	Y	-	-	-	-	-	



Function	Contents	Version before Migration						
		V7	V8	V9	V9.1	V9.2	V10	V11
	IJServer launch configuration	Y	Y	-	-	-	-	-
	Tomcat launch configuration	Y	Y	-	-	-	-	-
	Change of the default case for HTML tags to lowercase	Y	Y	-	-	-	-	-
	FTP client	Y	Y	-	-	-	-	-
	Integration of Web-application-related wizards	Y	-	-	-	-	-	-
	Changes to Web-service-related wizards	Y	-	-	-	-	-	-
	Servlet page template (SPT) editor	Y	-	-	-	-	-	-
	Web service deployment and debugging	Y	-	-	-	-	-	-
	Web site tools	Y	-	-	-	-	-	-
Java	Predefined library "Apworks Runtime"	Y	Y	-	-	-	-	-
	Libraries that do not support JDK/JRE 6 and JDK/JRE 7	Y	Y	-	-	-	-	-
	Changes to builders	Y	Y	-	-	-	-	-
	Changes to Java-related wizards	Y	Y	-	-	-	-	-
	Comment for the Graphical Editor	Y	Y	-	-	-	-	-
	Java applet launch configuration	Y	-	-	-	-	-	-
Apcoordinator	Predefined library "Fujitsu XML" of Apcoordinator project	Y	Y	Y	-	-	-	-
	Struts library when using the Struts linkage function	Y	Y	Y	Y	Y	Y	Y
J Business Kit	Changes to the specifications of the download installer	Y	Y	Y	Y	Y	Y	-
	Behavior of JFLightPanel when using JDK/JRE 7	Y	Y	Y	Y	Y	Y	-
	Focus movement with the focus traversal key when using JDK/JRE 7	Y	Y	Y	Y	Y	Y	-
	Changes to the storage location of the public source of the library	Y	Y	Y	Y	Y	Y	-
	Behavior of JFFieldDouble and JFFieldLong	Y	Y	Y	Y	Y	Y	-
	Behavior of setScrollbarDisplayPolicy	-	-	-	-	Y	Y	-
	JFInputLimitation	Y	-	-	-	-	-	-
Installer	Function installation folders (from V9.2 to V10)	-	-	-	-	Y	Y	-
	Function installation folders (V9.1 and earlier)	Y	Y	Y	Y	-	-	-
	Changes to the uninstallation procedure	Y	Y	Y	Y	Y	Y	-
	Creating IJServers, IJServer clusters, and CORBA WorkUnits	Y	Y	Y	Y	Y	-	-
	Installing Fujitsu XML Processor	Y	Y	Y	-	-	-	-
Manual	Changing manuals to be provided	Y	Y	Y	-	-	-	-

Y: Incompatibilities exist

-: No incompatibilities



## Example

---

### How to read compatibility information

If you were using Interstage Studio V9.2, read "Changes from the Functions of V9.2", "Changes from the Functions of V10.0" and "Changes from the Functions of V11.0". Read explanations in sequence, from the product you were using up to the most recent edition.

---

## 2.1 Changes from the Functions of V11.0

---

This section describes the changes from the functions of V11.0.

### 2.1.1 Java EE 6 Development Functions

---

#### Using Interstage Management Service Operation Tool to operate services used by Java EE 6

The Interstage Management Service Operation Tool can be used to operate the services that Java EE 6 uses and to manage a local Interstage Application Server. Services that can be started in the Interstage Management Service Operation Tool are as follows.

- If [Use only the Interstage Java EE DAS service] is selected:

- FJapache
- Interstage data store
- Interstage Operation Tool(FJapache)
- Interstage JServlet(OperationManagement)
- Interstage Operation Tool
- Interstage Java EE DAS

- If [Use the IJServer Cluster] is selected:

- FJapache
- Interstage data store
- Interstage Operation Tool(FJapache)
- Interstage JServlet(OperationManagement)
- Interstage Operation Tool
- Interstage Java EE DAS
- Interstage Java EE Node Agent

- If [Use J2EE execution environment] is selected:

- FJapache
- Interstage data store
- Interstage Operation Tool(FJapache)
- Interstage JServlet(OperationManagement)
- Interstage Operation Tool

- If [Use Java EE 6 execution environment] is selected:

- Interstage PCMI(isje6)
- Interstage HTTP Server 2.2

#### Default value of JDK version used in Java EE 6 function

The default value of the JDK version used in the Java EE 6 function has changed from "JDK 6" to "JDK 7".

## 2.1.2 Apcoordinator Development Functions

---

### Struts library when using the Struts linkage function

The Struts library is not bundled with Apcoordinator. For more information, refer to "Compatibility with Earlier Versions of Apcoordinator" in the "Apcoordinator User's Guide".

## 2.2 Changes from the Functions of V10.0

---

This section describes the changes from the functions of V10.0.

### 2.2.1 Basic Functions

---

#### Development on the Compatible Workbench provided in V10

The Compatible Workbench is no longer provided. Use the standard workbench to develop J2EE business systems and Java SE applications rather than the Compatible Workbench. Refer to "Migrating Resources from a Previous Version" in "Interstage Studio User's Guide" for information on migrating projects that were developed on the Compatible Workbench.

#### Workbench virtual memory size

The options used by the workbench are specified in the following file:

<Workbench installation folder>\eclipse\isstudio.ini

In here, the size of the JavaVM virtual memory set for new installations has been changed:

Value used up to V10.0	Value used in V11.0
-Xms128m -Xmx256m -XX:MaxPermSize=256m	When the Application Server is installed: -Xms128m -Xmx256m -XX:PermSize=128m -XX:MaxPermSize=256m
	When the Application Server is not installed: -Xms128m -Xmx256m -XX:PermSize=64m -XX:MaxPermSize=256m

The current value is not changed if the installation status of the Application Server is changed by overwrite installation.

#### Some of the Java Form features are now deprecated features

Some deprecated functions are unable to be displayed by default, unlike previous Java Forms.

Refer to the "Interstage Studio User's Guide", "Developing Java Applications" > "Tasks" > "Performing Environment Setup for Java Applications" > "Setting Options" for information on these features and how to display the deprecated features.

## 2.2.2 J2EE Development Functions

---

### J2EE Perspective

The J2EE Perspective of the Compatible Workbench provided in V10 is no longer provided. Use the [Java EE] perspective instead when developing J2EE and Java EE applications.

## Deployment Descriptor (EJB) wizard

The deployment descriptor (EJB) wizard of the Compatible Workbench provided in V10 is no longer provided. The deployment descriptors are automatically generated using the EJB project wizard.

## DD (Deployment Descriptor) Editor

The DD Editor of the Compatible Workbench provided in V10 is no longer provided. Use an XML editor to edit the deployment descriptors.

XML editors provide a wealth of functionality to assist input, such as Content Assist and tool tips, and provide verification tools such as error correction during input. It is also possible to edit with forms in the design tags so that even those unfamiliar with XML will be able to edit with ease.

## EAR function for rapid deployment

The EAR function for rapid deployment of the Compatible Workbench provided in V10 is no longer provided.

By creating the stubs and skeletons used for remote communication when creating EAR, this function attempted to reduce the time for deployment. As the EJB2.1 deployment by Interstage Application Server only supports Web+EJB[1VM], there is no opportunity to use the function.

## No support for the J2EE 1.3 format

Development of J2EE 1.3 applications is no longer supported.

## 2.2.3 J Business Kit

---

### Changes to the specifications of the download installer

The download installer is a function for creating installers by selecting components distributed to Windows clients. Only the JRE version of the components to be distributed can be selected.

Refer to "J Business Kit User's Guide" > "JBK Download Installer" for details.

### Behavior of JFLightPanel when using JDK/JRE 7

When the JFLightPanel is operated as follows using JDK/JRE 7, the focus does not traverse to the part that has setFirstFocusComponent() specified.

- The follow settings are made in JFLightPanel>(\*1)
  - Specify the component to first set focus with setFirstFocusComponent().
  - Call setAutoFocus(true) to enable focus setting functions.
- Use the following procedure:
  1. Set focus on component in the JFLightPanel of (\*1).
  2. Activate another window.
  3. Click on JFLightPanel.

\* This is a limitation of JDK/JRE7.

### Focus movement with the focus traversal key when using JDK/JRE 7

There have been major changes in the focus traversal policy in JDK/JRE7. Focus control will be disabled with the focus traversal key when the following components are used:

- JFChoice
- JFTableView
- JFTabPanel
- JFTextArea

For this reason, the focus traversal policy is set independently in the following containers in the GUI components of the J Business Kit. Use the following containers to ensure that the focus traverses to the correct position with the focus traversal key:

- JFApplet
- JFDialog
- JFFrame
- JFTextInputWindow
- JFWindow
- JFCFrame
- JFCDialog

If java.applet.Applet, java.awt.Dialog, java.awt.Frame, or java.awt.Window container was used

Use the above containers.

Alternatively, call JFFocusTraversalPolicy.init() when the application starts and set the focus traversal policy of the JBK in the KeyboardFocusManager.

If an applet class other than JFApplet was used

Override the addNotify() method of the applet class as follows:

```
public void addNotify(){
    super.addNotify();

    //Set the focus traversal policy
    Container cycleRoot = this.getFocusCycleRootAncestor();
    if(cycleRoot!= null){
        cycleRoot.setFocusTraversalPolicy(new JFFocusTraversalPolicy());
    }
}
```

## Changes to the storage location of the public source of the library

The storage location of the public source of the library has changed.

- Up to V10.0

Location: JBK installation directory\src

- V11.0 or later

Location: JBK installation directory\gui6 / JBK installation directory\gui7

## Behavior of JFFieldDouble and JFFieldLong

If strings that cannot be converted to figures are entered in the input area, an exception is returned when the following methods are called.

- protected java.awt.Color getDisplayBackground()
- protected java.awt.Color getDisplayForeground()

From V11.0, changes have been made to return the color currently used for drawing.

## Behavior of setScrollbarDisplayPolicy

As the correspondence of PG88776, adjustments have been made to make it impossible to use the scrollbars when setScrollbarDisplayPolicy(int) of JFCScrollablePanel is called. This may cause a NullPointerException to occur because there is a disabled event process, but actual behavior is not affected.

\*PG88776: When a setScrollbarDisplayPolicy() call and the scrollbars operation are performed simultaneously, the Java process loops.

## 2.2.4 Installer

---

### Function installation folders

The functions (components) for which the installation destination differs from that of earlier versions, and the installation folders of these respective functions are shown below. If these function paths are included in user resources, change them to the paths for this product. Note that the installation folders below assume that the product installation folder is "C:\Interstage".

Function		Installation location from V9.2 to V10.0	Installation directories for this product
J Business Kit	for JRE 6	C:\Interstage\IDE\(\version number *1)\JBK6	C:\Interstage\IDE\(\version number *1)\JBK
	Download installer composition kit	C:\Interstage\IDE\(\version number *1)\jbkplgdi	C:\Interstage\IDE\jbkplgdi
JDK 6		C:\Interstage\IDE\(\version number *1)\JDK6	C:\Interstage\IDE\JDK6

\*1: The "*version number*" value depends on the product version. For V9.2 it is "0902". For V11.1 it is "1101".

### Changes to the uninstallation procedure

The uninstallation procedure has changed.  
Follow the procedure below to uninstall this product.

In Windows 8 and Windows Server 2012:

On the App Bar of the start page, go to [All Programs] > [Fujitsu] > [Uninstall (middleware)].

In other Windows systems:

On the start menu, go to [All Programs] > [Fujitsu] > [Uninstall (middleware)].

## 2.3 Changes from the Functions of V9.2

---

This section describes the changes from the functions of V9.2.

### 2.3.1 Basic Functions

---

#### Integration of Interstage Management Service Operation Tools

The Interstage Management Service Operation Tool and the Interstage Management Service Operation Tool (Java EE) are integrated in the Interstage Management Service Operation Tool. The J2EE and Java EE services required for management of a local Interstage Application Server can be run from the one tool. The services shown below can be started by the Interstage Management Service Operation Tool.

- If [Use only the Interstage Java EE DAS service] is selected:
  - FJapache
  - Interstage data store
  - Interstage Operation Tool(FJapache)
  - Interstage JServlet(OperationManagement)
  - Interstage Operation Tool
  - Interstage Java EE DAS
- If [Use the IJServer Cluster] is selected:
  - FJapache
  - Interstage data store

- Interstage Operation Tool(FJapache)
- Interstage JServlet(OperationManagement)
- Interstage Operation Tool
- Interstage Java EE DAS
- Interstage Java EE Node Agent
- If [Use J2EE execution environment] is selected:
  - FJapache
  - Interstage data store
  - Interstage Operation Tool(FJapache)
  - Interstage JServlet(OperationManagement)
  - Interstage Operation Tool

### Changes to the database name used by the Interstage directory service

The standard database name used by the Interstage directory service of Interstage Application Server has changed from Enabler to Interstage data store. In association with this name change, the service name has changed from Fujitsu Enabler to Interstage data store.

## 2.3.2 Installer

---

### Creating IJServers, IJServer clusters, and CORBA WorkUnits

Under this product, IJServers (MyDebug/My1VMDebug), IJServer clusters (MyDebugJEE), and CORBA WorkUnits (MyCORBADebug) are no longer created during installation.

- If IJServers (MyDebug/My1VMDebug) are required, create them referring to the "Interstage Studio User's Guide", "Migrating J2EE 1.4 Applications" > "J2EE Application Common Items" > "Preparing the Deployment Destination that is used for Application Operation Verification" for details.
- If IJServer Clusters (MyDebugJEE) are required, create them using the Interstage Java EE Admin Console or the asadmin command. Refer to the "Interstage Studio User's Guide", "Items Common to Java EE 5 Applications" > "Tasks" > "Preparing the Deployment Destination for Verifying Application Operation" for details.
- If CORBA WorkUnits (MyCORBADebug) are required, create them using the Interstage Management Console or the isaddwudef command. Refer to the "Interstage Application Server Operator's Guide" for details.

## 2.4 Changes from the Functions of V9.1

---

This section describes the changes from the functions of V9.1.

### 2.4.1 Installer

---

#### Function installation folders

The installation folders for the functions (components) of this product have been changed so that they do not conflict with those of related products with the same installation destination. Refer to the "Software Release Guide", "Folder Structure and Files" for the installation destinations of the various functions under this product.

For the functions shown in the table, the install destinations up to V9.1 are different from those under V9.2 and later. If these function paths are included in user resources, change them to the paths for this product. Note that the installation folders below assume that the product installation folder is "C:\Interstage".

Function	Installation folder up to V9.1	Installation folder under V9.2 and later
Java integrated development environment	C:\Interstage\IDE	C:\Interstage\IDE\(\version number *)

Function	Installation folder up to V9.1	Installation folder under V9.2 and later
Online manual	C:\Interstage\IDE>manual	C:\Interstage\IDE\( <i>version number</i> *)\manual
Application Server	C:\Interstage	C:\Interstage\APS

\*1: The "*version number*" value depends on the product version. For V9.2 it is "0902". For V11.1 it is "1101".

## 2.5 Changes from the Functions of V9.0

---

This section describes the changes from the functions of V9.

### 2.5.1 Basic Functions

---

#### Managing the resident service required for the management of the local server

There are changes in managing the resident services that are required when installing the [Interstage Application Server] functionality.

The resident services required for the management of the local Interstage Application Server (Herein after called the Interstage management services) were typically in start status after installation. However, it is in stop status in this version. This measure was taken to reduce the memory usage when the local server is not used.

- FJapache
- Fujitsu Enabler \*1
- Interstage JServlet(Operation Management)
- Interstage Operation Tool
- Interstage Operation Tool(FJapache)

\*1: The service name has changed from Fujitsu Enabler to Interstage data store.

Furthermore, their startup type is not set as "Automatic", but is set as "Manual". Therefore even if the machine is restarted, the services will not become resident.

Due to the above changes, operations on the workbench are changed as follows:

- If the user with administrator's authority launches the workbench, the local server (displayed as localhost) showed a connected status on the IJServer view from the beginning. In this version, however, the localhost shows a disconnected status on the Servers view.

To operate on the Servers view in this status, first click the Servers view, display the context menu, and select [Connect/Login] from the menu. The Interstage management services are started and the local server is connected. Then, the localhost displays a connected status on the Servers view. After that, the operation on the Servers view is performed as usual.

- Under the following operations, the Interstage was started when the Interstage was not already running. In this version, in addition to the above, the Interstage management services start when they are not already running. In this case, extra time is required for the Interstage management services to start.
  - Execution or debugging at a local server that uses the Interstage Application Server(J2EE) launch configuration or the Interstage Application Server V11.1 IJServer cluster (Java EE) launch configuration.
- The Interstage management services must be started beforehand if the Interstage management console is opened by selecting [Interstage Studio Vxx] > [Interstage Management Console] or [Interstage Java EE Management Console] from the Windows Start menu. To start the Interstage management services, use the Interstage management service operation tool, which will be described later.
- Once the Interstage management services have been started, they remain active even if the user exits the workbench. The Interstage management services are not restarted even if the workbench is repeatedly launched and exited. To stop the Interstage management services, use the Interstage management service operation tool, which will be described later, or restart the machine.

The Interstage management service operation tool is provided to operate the Interstage management services. Use this tool to start and stop the Interstage management services. In addition, the startup type of the services can be changed from "manual" to "automatic". The "automatic" setting allows the services to start simultaneously with the startup of the machine.



To start the Interstage management service operation tool, select [Interstage Studio Vxx] > [Interstage Management Service Operation Tool] from the Windows Start menu. For details about how to use the tool, refer to the Help document by clicking [Help] on the tool window.

### Predefined library "Fujitsu XML"

The Fujitsu XML Processor is not automatically installed when this product is installed. Therefore, the following error occurs in the product if the predefined library "Fujitsu XML" is used if the Fujitsu XML Processor has not been installed separately.

The project cannot be built until build path errors are resolved.  
Project <Project name> is missing required library: 'C:\Program Files\Common Files\FujitsuXML\xmlpro.jar'.

The Fujitsu XML Processor must be installed separately if the predefined library "Fujitsu XML" is used. If the Fujitsu XML Processor is not used, delete the predefined library "Fujitsu XML" from the project.

## 2.5.2 J2EE Development Functions

---

### Non-support of the J2EE 1.3 format application client

Since an application client of J2EE 1.3 is not supported, it cannot be created using wizard.

### Changing the IJServer type at the default deployment destination

The previous IJServer default deployment destination was "MyDebug" at the Web + EJB [separate VM] IJServer, but under this product IJServers must be created and deployed. Create IJServers referring to the "Interstage Studio User's Guide", "Migrating J2EE 1.4 Applications" > "J2EE Application Common Items" > "Preparing the Deployment Destination that is used for Application Operation Verification".

## 2.5.3 Apcoordinator Development Functions

---

### Predefined library "Fujitsu XML" of Apcoordinator project

Under this product, the Fujitsu XML processor is no longer installed automatically during installation.

In the Apcoordinator project created in the environment prior to V9.0, the predefined library "Fujitsu XML" is specified in the "Java Build Path" as the default. If the Fujitsu XML Processor is not installed, follow the procedure in the chapter "Basic Functions" > "[Predefined library "Fujitsu XML"](#)".

## 2.5.4 Installer

---

### Installing Fujitsu XML Processor

The Fujitsu XML Processor is not automatically installed when this product is installed. To develop an application that manages the XML format document using Fujitsu XML Processor, refer to "Add-on Components" in the "Installation Guide" and install it separately.

Installing the Fujitsu XML Processor does not set the following JAR files in the system environment variables CLASSPATH. Add them if they are required. For details, refer to the Software instructions of the Fujitsu XML Processor.

- xmlpro.jar
- xmltrans.jar
- xmltransx.jar

## 2.5.5 Online Manuals

---

### Changing manuals to be provided

The following workbench help documentation cannot be referenced from the manual viewer. To refer to these manuals, select [Help] > [Help Contents] from the workbench menu.

- Workbench User Guide

- Java Development User Guide

The following reference manuals (javadoc) with API specifications cannot be referenced from the manual viewer. Check these API specifications with "J2EE 1.4 API Specification"

- J2EE 1.3 API Specification
- JAX-RPC 1.1 API Specification
- SAAJ 1.2 API Specification

## 2.6 Changes from the Functions of V8

---

This section describes the changes from the functions of V8.

### 2.6.1 Basic Functions

---

#### Project wizard configuration

The project wizards provided in V8 or earlier required three steps, project settings, Java build path settings, and skeleton code selections. This has been simplified so that a project can be created by performing only the project-setting step. After creating the project, specify the Java build path in the project properties dialog box. Also after creating the project, generate code by invoking the required source wizard. Note that the Apcoordinator project wizards create a project in three steps as before.

### 2.6.2 J2EE Development Functions

---

#### Project creation for J2EE application development

In V8 and earlier, J2EE application development relies on the EAR file generation wizard. This wizard has become obsolete, and the [Enterprise Application Project] wizard is provided as an alternative. Accordingly, the EAR file editor was also removed.

You can use the [Enterprise Application Project] wizard to develop a J2EE application as a project. This simplifies the EAR file creation procedure.

- J2EE1.4 J2EE applications can be developed.
- You can add J2EE modules to an enterprise application project. Then, the modules are included in an EAR file.

#### Project creation for J2EE application client development

In V8 and earlier, J2EE application client development relies on the J2EE application client deployment descriptor generation wizard. This wizard has become obsolete, and the [Application Client Project] wizard is provided as an alternative.

- J2EE1.4 J2EE application clients can be developed.

#### Changes to J2EE-related wizards

The following J2EE-related wizards have been changed:

- Enterprise Bean wizard

It is now possible to generate EJB2.1 Enterprise Beans in the Enterprise Bean wizard. The wizard invocation method has changed to [EJB] > [J2EE] > [Enterprise Bean].

- HTML wizard

In V8 and earlier, there are two separate wizards: the HTML wizard and the HTML frame wizard. These wizards were integrated into one wizard. Users can customize HTML file templates for the new HTML wizard. By default, users can create either a typical HTML file or HTML frame.

- JSP wizard

The menu sequence for invocation of this wizard has been changed from [J2EE] > [JavaServer Pages] to [Web] > [JSP]. In addition, templates can now be used for the JSP files to be generated.

- CSS wizard

The menu sequence for invocation of this wizard has been changed from [J2EE] > [HTML] > [Cascading Style Sheets] to [J2EE] > [CSS]. In addition, templates can now be used for the CSS files to be generated.

- Servlet wizard

The wizard invocation method has changed to [Web] > [Servlet]. The information specified in the wizard and the format of the created files have been changed.

## Changes to builders

The following changes have been made to various tools used for build:

- EJB specification validation and JSP syntax check

The current functions for EJB specification validation and JSP syntax check by the JSP compiler are provided as validators. To achieve results equivalent to those of the previous version in the EJB specification verification, the EJB validator and Interstage EJB Container-managed persistence validator should be used. To achieve results equivalent to those of the previous version in the JSP syntax check, the HTML syntax validator and JSP syntax validator should be used.

- Servlet generation builder for JSP debugging

Because debugging on JSP files is now supported, servlet source files for JSP debugging are no longer generated.

- J2EE packager builder

The J2EE packager builder is no longer provided. It is no longer necessary to archive in order to deploy to the server from the workbenches. Use the export function to create the archive file to be deployed to the operating environment.

- Home/Component interface generation builder

The Home/Component interface generation build tool is not provided with this product, but the EJB validator can be executed to check the Home/Component interface.

## JSP tag input support in the JSP editor

The conditions under which a JSP tag is displayed as a candidate of code completion have been made stricter. Only JSP tags that are specified with the taglib directive in the relevant JSP file are displayed as candidates. This eliminates the problem of unused JSP tags being displayed as candidates.

## Breakpoint setting in the JSP editor

Now, you can set breakpoints only at the location of JSP elements in the JSP editor. No breakpoint can be set in a static area, such as that initiated by HTML tags.

## Updating the deployment descriptor version

In V8 and earlier, updating the deployment descriptor version relies on the version conversion function of the Web deployment descriptor editor, EJB deployment descriptor editor, or J2EE application client deployment descriptor editor. Now, a project menu command is used for such upgrading. To do so, execute [Update J2EE deployment descriptor] from the context menu of a project.

## No support for the J2EE 1.2 format

Development of J2EE 1.2 applications is no longer supported.

- Deployment descriptors for J2EE 1.2 cannot be generated.
- The EJB test client wizard cannot create a test client for J2EE 1.2.

## IJServer launch configuration

The IJServer launch configuration is not provided with this product.

## Tomcat launch configuration

The Tomcat launch configuration is not provided with this product.

## Change of the default case for HTML tags to lowercase

The default case for the HTML tags used in HTML files created by wizards and for the HTML tags entered using the code assist of the HTML editor has been changed from uppercase to lowercase.

## FTP client

This product does not provide the FTP client provided in V8 or earlier. To deploy Web application resources to a server, use the Servers view or a third-party FTP client software product. For details on the Servers view, refer to the "Interstage Studio User's Guide".

## 2.6.3 Java Development Functions

---

### Predefined library "Apworks Runtime"

The predefined library "Apworks Runtime" is not provided with this product.

### Libraries that do not support JDK/JRE 6 and JDK/JRE 7

The Java Form extension function (form extension function) library and Applet support library do not support JDK/JRE 6 and JDK/JRE 7. When using applets and Java Forms that use the form extension function, change to applets and Java Forms that use AWT or Swing. Refer to "Interstage Studio User's Guide" > "Migrating Resources from a Previous Version" for details.

### Changes to builders

The following changes have been made to various tools used for build:

- Build tools

All the build tools in the previous version have been integrated into builders.

- IDL compiler

The IDL compiler is no longer provided. If IDL compilation is required during build, add an IDL compiler as an external tool builder to the builders.

- JAR package

The JAR package is not provided with this product. To archive files, launch the export wizard from [File] > [Export], and then create a JAR to suit your needs.

- JavaBeans info generator and JavaBeans manifest creator

These two builders have been integrated into the BeanInfo builder.

### Changes to Java-related wizards

Java-related wizards have been changed as follows:

- Java application project wizard

The Pure Java project wizard and the Applet project wizard are not provided with this product. Use the Java project wizard.

- Renamed wizards

Some wizards have been renamed. They include the following.

V11	V8
DB Access Class	DB Access Source

### Comment for the Graphical Editor

The comment at the beginning of the source code of a file that can be edited with the Graphical Editor used to be "//APWORKS Form". This comment has been changed to "// Graphical Editor Form".

A file created with V8.0 or earlier can be opened by the Graphical Editor. When the opened file is saved, the comment at the beginning is changed to "// Graphical Editor Form".

## 2.7 Changes from the Functions of V7 and Earlier

---

This section describes the changes from the functions of V7.

### 2.7.1 Basic Functions

---

#### Reconfiguration of categories for new wizards

In V7 and earlier, the category [Java] contains various wizards. This version uses several additional types of categories to enable users to find the desired wizards easily. For example, the new categories include the following:

- [Apcordinator]

This category includes wizards used to develop applications based on the framework (Apcordinator).

- [Java]

This category includes wizards used to develop Java applications. It also includes wizards used to create database access classes.

- [XML]

This category includes wizards used to create XML files and XML schema files.

#### Projects outside the workspace folder

This version of the product enables projects to be placed outside the workspace folder. With this added flexibility, you can place your project in a specific folder for any reason, such as because it is required by your resource management tool.

### 2.7.2 J2EE Development Functions

---

#### Integration of Web-application-related wizards

In V7 and earlier, six types of wizards are provided for generating Web application code and six or seven types of wizards for creating JavaServer Pages. This version provides an integrated wizard, which can simplify handling.

As a result, simpler files are created, which are useful as the base from which users start editing.

#### Changes to Web-service-related wizards

Since Interstage Application Server 8.0, the Interstage Web service, a Web service execution environment, has been provided as a J2EE function. Furthermore, the following changes have been made to Web service development functions:

- The SOAP project wizard has been removed. Now, Web services can be developed as Dynamic web projects. This enables Web services to be deployed and debugged in an extremely simple manner as though they were Web applications.
- The SOAP server application code generation wizard and SOAP access source code generation wizard have been renamed to the Web service (JAX-RPC) wizard and Web service client (JAX-RPC) wizard, respectively. They generate source code for J2EE Web services.

#### Servlet page template (SPT) editor

An SPT is a template file used to create dynamic Web pages. Utilizing an SPT is one of the development techniques provided by Fujitsu for easy development of dynamic Web pages in the days when servlet-based Web application development was dominant. Now, SPTs have been replaced with JavaServer Pages (JSPs), as using JSPs has become a standard technique for Web application development. For this reason, we discarded the SPT editor to emphasize our policy of recommending use of JSPs.

#### Web service deployment and debugging

Now, since Web services are integrated into Web applications, you can deploy and debug Web services in the same way as for Web applications. A Web service that is locally debugged is automatically deployed to the local application server.

#### Web site tools

This product does not provide the Web site tools provided in V7 or earlier. The following functions can be used as alternatives to the functions provided by those Web site tools:

- JSP wizard
- HTML wizard
- HTML/JSP editor
- CSS editor
- XML editor
- Servers view

## 2.7.3 Java Development Functions

---

### Java applet launch configuration

The Java applet launch configuration provided in V7 and earlier has been integrated into the Java applet launch configuration provided by Eclipse. As a result, you can execute and debug applets from the workbench without creating any HTML file for applet invocation.

## 2.7.4 J Business Kit

---

### JFInputLimitation

In V6 and earlier versions, if the inputtable character type in JFInputLimitation is set to any of the following values, then characters that are not included in the characters defined in the JFInputLimitation can be input.

- TYPE\_ALL
- TYPE\_MULTIBYTE
- If the inputtable character type is set to include all the following character types
  - TYPE\_MULTIBYTE\_DIGIT
  - TYPE\_MULTIBYTE\_LOWERCASE
  - TYPE\_MULTIBYTE\_UPPERCASE
  - TYPE\_MULTIBYTE\_SPECIAL
  - TYPE\_MULTIBYTE\_HIRAGANA
  - TYPE\_MULTIBYTE\_KATAKANA
  - TYPE\_JIS\_KANJI
  - TYPE\_GAIJI
  - TYPE\_VENDOR\_SPECIAL

In V7 and later versions, characters that are not included in the character types mentioned above cannot be input, thus leading to the following incompatibility problems.

**The input results vary, depending on the setting of the inputtable character type in the JFInputLimitation.**

- When the inputtable character type in the JFInputLimitation is set to TYPE\_ALL
  - In V6 and earlier versions
    - All the characters can be input.
  - In V7 and later versions
    - Only the characters that match the following character types can be input.
      - TYPE\_MULTIBYTE\_DIGIT
      - TYPE\_MULTIBYTE\_LOWERCASE
      - TYPE\_MULTIBYTE\_UPPERCASE

- TYPE\_MULTIBYTE\_SPECIAL
- TYPE\_MULTIBYTE\_HIRAGANA
- TYPE\_MULTIBYTE\_KATAKANA
- TYPE\_JIS\_KANJI
- TYPE\_GAIJI
- TYPE\_VENDOR\_SPECIAL
- TYPE\_DIGIT
- TYPE\_LOWERCASE
- TYPE\_UPPERCASE
- TYPE\_SPECIAL
- TYPE\_KATAKANA

### [Resolution]

One way to resolve this issue is to override the processKeyEvent(KeyEvent) method in the character input component.

Another way is to unset JFInputLimitation, add the characters that cannot be input because of the default setting to the inputtable characters, and then do not process the input events of repeated character keys.



### Example

#### Implementation example

```
public class SubField extends JFFieldString {
    //Add the characters that are not included in the default input setting to the inputtable
    characters.
    String enableChars = "\u00a6\u00a7\u00a9\u00ad\u00ae" +
        "\u00b0\u00b2\u00b3\u00b6\u00b9\u00bc\u00bd\u00be";
    //Add the repeated characters to the not-inputtable characters
    String disableChars = "\u221a\u2220\u2229\u222a\u222b" +
        "\u2235\u2252\u2261\u22a5";

    public SubField() {
        //Add the inputtable characters
        this.setEnableChars(enableChars.toCharArray());
    }

    protected void processKeyEvent(KeyEvent ke){
        if(ke.getID()==KeyEvent.KEY_TYPED){
            //Do not input the repeated characters
            if(disableChars.indexOf(ke.getKeyChar())!=-1){
                return;
            }else{
                super.processKeyEvent(ke);
            }
        }else{
            super.processKeyEvent(ke);
        }
    }
}
```

- When the inputtable character type is set to TYPE\_MULTIBYTE or the combination of all the following
  - TYPE\_MULTIBYTE\_DIGIT
  - TYPE\_MULTIBYTE\_LOWERCASE
  - TYPE\_MULTIBYTE\_UPPERCASE

TYPE\_MULTIBYTE\_SPECIAL  
TYPE\_MULTIBYTE\_HIRAGANA  
TYPE\_MULTIBYTE\_KATAKANA  
TYPE\_JIS\_KANJI  
TYPE\_GAJI  
TYPE\_VENDOR\_SPECIAL

- In V6 and earlier versions

All characters that are not TYPE\_HALFBYTE can be input.

- In V7 and later versions

Only the characters that match the following character types can be input.

- TYPE\_MULTIBYTE\_DIGIT
- TYPE\_MULTIBYTE\_LOWERCASE
- TYPE\_MULTIBYTE\_UPPERCASE
- TYPE\_MULTIBYTE\_SPECIAL
- TYPE\_MULTIBYTE\_HIRAGANA
- TYPE\_MULTIBYTE\_KATAKANA
- TYPE\_JIS\_KANJI
- TYPE\_GAJI
- TYPE\_VENDOR\_SPECIAL

#### [Resolution]

To resolve this issue, use `setEnabledUnicodeType(java.lang.Character.UnicodeBlock...enableUnicodeType)` of `JFInputLimitation`, allow the input of characters that are not included in the `BASIC_LATIN` character Block, and add characters included in `TYPE_SPECIAL` and `TYPE_KATAKANA` to the not-inputtable characters to allow the input of characters that are not `TYPE_HALFBYTE`.



#### Example

#### Implementation example

```
JFInputLimitation limit = new JFInputLimitation();

//Make all the characters that are not included in BASIC_LATIN inputtable.
limit.setEnabledUnicodeType(Character.UnicodeBlock.AEGEAN_NUMBERS
    ,Character.UnicodeBlock.ALPHABETIC_PRESENTATION_FORMS
    ,Character.UnicodeBlock.ARABIC
    ,Character.UnicodeBlock.ARABIC_PRESENTATION_FORMS_A
    ,Character.UnicodeBlock.ARABIC_PRESENTATION_FORMS_B
    ,Character.UnicodeBlock.ARMENTIAN
    ,Character.UnicodeBlock.ARROWS
    //,Character.UnicodeBlock.BASIC_LATIN Do not input halfwidth characters
    ,Character.UnicodeBlock.BENGALI
    ,Character.UnicodeBlock.BLOCK_ELEMENTS
    ,Character.UnicodeBlock.BOPOMOFO
    ,Character.UnicodeBlock.BOPOMOFO_EXTENDED
    ,Character.UnicodeBlock.BOX_DRAWING
    ,Character.UnicodeBlock.BRAILLE_PATTERNS
    ,Character.UnicodeBlock.BUHID
    ,Character.UnicodeBlock.BYZANTINE_MUSICAL_SYMBOLS
    ,Character.UnicodeBlock.CHEROKEE
    ,Character.UnicodeBlock.CJK_COMPATIBILITY
    ,Character.UnicodeBlock.CJK_COMPATIBILITY_FORMS
```



,Character.UnicodeBlock.CJK\_COMPATIBILITY\_IDEOGRAPHS  
,Character.UnicodeBlock.CJK\_COMPATIBILITY\_IDEOGRAPHS\_SUPPLEMENT  
,Character.UnicodeBlock.CJK\_RADICALS\_SUPPLEMENT  
,Character.UnicodeBlock.CJK\_SYMBOLS\_AND\_PUNCTUATION  
,Character.UnicodeBlock.CJK\_UNIFIED\_IDEOGRAPHS  
,Character.UnicodeBlock.CJK\_UNIFIED\_IDEOGRAPHS\_EXTENSION\_A  
,Character.UnicodeBlock.CJK\_UNIFIED\_IDEOGRAPHS\_EXTENSION\_B  
,Character.UnicodeBlock.COMBINING\_DIACRITICAL\_MARKS  
,Character.UnicodeBlock.COMBINING\_HALF\_MARKS  
,Character.UnicodeBlock.COMBINING\_MARKS\_FOR\_SYMBOLS  
,Character.UnicodeBlock.CONTROL\_PICTURES  
,Character.UnicodeBlock.CURRENCY\_SYMBOLS  
,Character.UnicodeBlock.CYPRIOT\_SYLLABARY  
,Character.UnicodeBlock.CYRILLIC  
,Character.UnicodeBlock.CYRILLIC\_SUPPLEMENTARY  
,Character.UnicodeBlock.DESERET  
,Character.UnicodeBlock.DEVANAGARI  
,Character.UnicodeBlock.DINGBATS  
,Character.UnicodeBlock.ENCLOSED\_ALPHANUMERIC  
,Character.UnicodeBlock.ENCLOSED\_CJK\_LETTERS\_AND\_MONTHS  
,Character.UnicodeBlock.ETHIOPIA  
,Character.UnicodeBlock.GENERAL\_PUNCTUATION  
,Character.UnicodeBlock.GEOMETRIC\_SHAPES  
,Character.UnicodeBlock.GEORGIAN  
,Character.UnicodeBlock.GOTHIC  
,Character.UnicodeBlock.GREEK  
,Character.UnicodeBlock.GREEK\_EXTENDED  
,Character.UnicodeBlock.GUJARATI  
,Character.UnicodeBlock.GURMUKHI  
,Character.UnicodeBlock.HALFWIDTH\_AND\_FULLWIDTH\_FORMS  
,Character.UnicodeBlock.HANGUL\_COMPATIBILITY\_JAMO  
,Character.UnicodeBlock.HANGUL\_JAMO  
,Character.UnicodeBlock.HANGUL\_SYLLABLES  
,Character.UnicodeBlock.HANUNOO  
,Character.UnicodeBlock.HEBREW  
,Character.UnicodeBlock.HIGH\_PRIVATE\_USE\_SURROGATES  
,Character.UnicodeBlock.HIGH\_SURROGATES  
,Character.UnicodeBlock.HIRAGANA  
,Character.UnicodeBlock.IDEOGRAPHIC\_DESCRIPTION\_CHARACTERS  
,Character.UnicodeBlock.IPA\_EXTENSIONS  
,Character.UnicodeBlock.KANBUN  
,Character.UnicodeBlock.KANGXI\_RADICALS  
,Character.UnicodeBlock.KANNADA  
,Character.UnicodeBlock.KATAKANA  
,Character.UnicodeBlock.KATAKANA\_PHONETIC\_EXTENSIONS  
,Character.UnicodeBlock.KHMER  
,Character.UnicodeBlock.KHMER\_SYMBOLS  
,Character.UnicodeBlock.LAO  
,Character.UnicodeBlock.LATIN\_1\_SUPPLEMENT  
,Character.UnicodeBlock.LATIN\_EXTENDED\_A  
,Character.UnicodeBlock.LATIN\_EXTENDED\_ADDITIONAL  
,Character.UnicodeBlock.LATIN\_EXTENDED\_B  
,Character.UnicodeBlock.LETTERLIKE\_SYMBOLS  
,Character.UnicodeBlock.LIMBU  
,Character.UnicodeBlock.LINEAR\_B\_IDEOGRAMS  
,Character.UnicodeBlock.LINEAR\_B\_SYLLABARY  
,Character.UnicodeBlock.LOW\_SURROGATES  
,Character.UnicodeBlock.MALAYALAM  
,Character.UnicodeBlock.MATHEMATICAL\_ALPHANUMERIC\_SYMBOLS  
,Character.UnicodeBlock.MATHEMATICAL\_OPERATORS  
,Character.UnicodeBlock.MISCELLANEOUS\_MATHEMATICAL\_SYMBOLS\_A  
,Character.UnicodeBlock.MISCELLANEOUS\_MATHEMATICAL\_SYMBOLS\_B  
,Character.UnicodeBlock.MISCELLANEOUS\_SYMBOLS

```

,Character.UnicodeBlock.MISCELLANEOUS_SYMBOLS_AND_ARROWS
,Character.UnicodeBlock.MISCELLANEOUS_TECHNICAL
,Character.UnicodeBlock.MONGOLIAN
,Character.UnicodeBlock.MUSICAL_SYMBOLS
,Character.UnicodeBlock.MYANMAR
,Character.UnicodeBlock.NUMBER_FORMS
,Character.UnicodeBlock.OGHAM
,Character.UnicodeBlock.OLD_ITALIC
,Character.UnicodeBlock.OPTICAL_CHARACTER_RECOGNITION
,Character.UnicodeBlock.ORIYA
,Character.UnicodeBlock.OSMANYA
,Character.UnicodeBlock.PHONETIC_EXTENSIONS
,Character.UnicodeBlock.PRIVATE_USE_AREA
,Character.UnicodeBlock.RUNIC
,Character.UnicodeBlock.SHAVIAN
,Character.UnicodeBlock.SINHALA
,Character.UnicodeBlock.SMALL_FORM_VARIANTS
,Character.UnicodeBlock.SPACING_MODIFIER_LETTERS
,Character.UnicodeBlock.SPECIALS
,Character.UnicodeBlock.SUPERSCRIPTS_AND_SUBSCRIPTS
,Character.UnicodeBlock.SUPPLEMENTAL_ARROWS_A
,Character.UnicodeBlock.SUPPLEMENTAL_ARROWS_B
,Character.UnicodeBlock.SUPPLEMENTAL_MATHEMATICAL_OPERATORS
,Character.UnicodeBlock.SUPPLEMENTARY_PRIVATE_USE_AREA_A
,Character.UnicodeBlock.SUPPLEMENTARY_PRIVATE_USE_AREA_B
,Character.UnicodeBlock.SYRIAC
,Character.UnicodeBlock.TAGALOG
,Character.UnicodeBlock.TAGBANWA
,Character.UnicodeBlock.TAGS
,Character.UnicodeBlock.TAI_LE
,Character.UnicodeBlock.TAI_XUAN_JING_SYMBOLS
,Character.UnicodeBlock.TAMIL
,Character.UnicodeBlock.TELUGU
,Character.UnicodeBlock.THAANA
,Character.UnicodeBlock.THAI
,Character.UnicodeBlock.TIBETAN
,Character.UnicodeBlock.UGARITIC
,Character.UnicodeBlock.UNIFIED_CANADIAN_ABORIGINAL_SYLLABICS
,Character.UnicodeBlock.VARIATION_SELECTORS
,Character.UnicodeBlock.VARIATION_SELECTORS_SUPPLEMENT
,Character.UnicodeBlock.YI_RADICALS
,Character.UnicodeBlock.YI_SYLLABLES
,Character.UnicodeBlock.YI_JING_HEXAGRAM_SYMBOLS
);

//Add halfwidth Katakana characters to the not-inputtable characters
//Halfwidth Katakana characters are included in the HALFWIDTH_AND_FULLWIDTH_FORMS Block, which
means the halfwidth Katakana characters can be input, so they shall be added to the not-inputtable
characters.
//from dd66 to ff9d
String disableChars =
"\uff66\uff67\uff68\uff69\uff6a" +
"\uff6b\uff6c\uff6d\uff6e\uff6f" +
"\uff70\uff71\uff72\uff73\uff74" +
"\uff75\uff76\uff77\uff78\uff79" +
"\uff7a\uff7b\uff7c\uff7d\uff7e" +
"\uff7f\uff80\uff81\uff82\uff83" +
"\uff84\uff85\uff86\uff87\uff88" +
"\uff89\uff8a\uff8b\uff8c\uff8d" +
"\uff8e\uff8f\uff90\uff91\uff92" +
"\uff93\uff94\uff95\uff96\uff97" +
"\uff98\uff99\uff9a\uff9b\uff9c" +
"\uff9d"+

```

```

//Add the halfwidth special characters that are not included in BASIC_LATIN to the not-inputtable
characters
"\u00a5\u203e\u203f\u2040\u2041\u2042\u2043\u2044\u2045\u2046\u2047\u2048\u2049\u204a\u204b\u204c\u204d\u204e\u204f" +
"\uff64\u2013\u2014\u2015\u2016\u2017\u2018\u2019\u201a\u201b\u201c\u201d\u201e\u201f" +

//Add the repeated characters to the not-inputtable characters
"\u221a\u2220\u2229\u222a\u222b" +
"\u2235\u2252\u2261\u22a5"+

//Add the control characters beyond the BASIC_LATIN scope to the not-inputtable characters
//from 0080 to 009f
"\u0080\u0081\u0082\u0083\u0084" +
"\u0085\u0086\u0087\u0088\u0089" +
"\u008a\u008b\u008c\u008d\u008e" +
"\u008f\u0090\u0091\u0092\u0093" +
"\u0094\u0095\u0096\u0097\u0098" +
"\u0099\u009a\u009b\u009c\u009d" +
"\u009e\u009f";

//Add the not-inputtable characters
limit.setDisableChars(disableChars.toCharArray());

//In versions before the revision, add the characters that are not included in TYPE_HALFBYTE to the
inputtable characters
String enableChars = "\u007e";//~

//Add the inputtable characters
limit.setEnableChars(enableChars.toCharArray());

```

.....

Refer to the "J Business Kit GUI Library Package Guide" for details on the JFInputLimitation function.

## 2.8 Migration from the Previous Version

---

Resource migration may be required for using a workspace or project created with the previous version of the product.

Refer to the "Interstage Studio User's Guide" > "Migrating Resources from a Previous Version" for information on migrating workbench resources.

## Chapter 3 Functions No Longer Provided

This chapter describes the earlier functions that are not provided under this product.

### Functions no longer provided in V11

- JDK/JRE 5.0
- J Business Kit JDK/JRE 5.0 support
- J Business Kit class files
- Compatible Workbench
- Database linkage function for linkage of workbench and Symfoware Server V8
- Database linkage function for linkage of Java EE 6 workbench and Derby 10.6

### Functions no longer provided in V10

- JDK/JRE 1.4
- J Business Kit JDK/JRE 1.4.2 support
- COBOL/CORBA application development functions
- Database linkage function for linkage with Oracle Database 9
- Development support functions for applications linked to Interstage Shunsaku
- Predefined library (Interstage Studio compatible runtime library, DB access, Interstage SOAP, JICK, Interstage Shunsaku)
- J Business Kit debug mode (specify `jbk.plugin.debugmode`)
- J Internet Communication Kit

## Chapter 4 Fixes in this Version

Faults that occurred with earlier versions and levels have been corrected in this product. Note that Interstage Studio previous versions and levels include Interstage Apworks V8 and earlier.

This chapter describes the bug fixes that apply to the product of this version.

### Notation

This chapter uses the following terms in headers of tables that list program fixes:

No.	Version	P number	Symptom
-----	---------	----------	---------

No.

Sequence number

Version

Oldest version at which the error may occur

P number

Unique number assigned to the error

Symptom

Symptom of the error

## 4.1 Java Integrated Development Environment

No.	Version	P number	Symptom
1	V9.0.0	PG86855	A "java.lang.NullPointerException" error may occur when copying and pasting components in the Graphical Editor.
2	V9.0.0	PG90121	The required libraries may not be recognized even when the build path is set correctly, causing compile errors.
3	V9.0.0	PG90122	Errors occur with files created using Microsoft Office 2007 or 2010 (".docx", ".xlsx", etc) when they are updated in a workbench.
4	V9.2.0	PG90124	When a project in an EAR project is modified, the EAR project may not be redeployed even if [Publish] is performed in Servers view.
5	V9.2.0	PG95925	In the Symfoware Server, if the same schema name and the same table appear in different databases, multiple tables will be shown.
6	V10.0.0	PG94708	XSD files acquired from the [WebService(JAX-WS)] wizard may contain unreadable codes.
7	V10.1.0	PG95772	This workbench and the Java EE 6 workbench use different words to refer to global headers on the [Preferences] page. The global header is always shown when the Java EE 6 workbench is started, no matter what the setting value is.
8	V11.1.0	PG96815	When the services needed for application development are operated in the Interstage Management Service Operation Tool and the workbench, the operation fails.
9	V11.1.0	PG96817	The Java EE 6 application deployed to the IIServer Cluster in the localhost may not run properly.

## 4.2 J Business Kit

No.	Version	P number	Symptom
1	V1.0L10	PG91717	On the JFCScrollablePanel, the page display during the UI Screen transition and the scrollbar operation may cause component overlapping during the page rendering.
2	V1.0L10	PG93388	When the applet that runs on JBK Plugin is started, PluginAppletEvent of PLUGINAPPLET_ACTIVATE may be notified twice continuously.
3	V3.0L10	PG94505	Ensure the notification conditions of PluginAppletEvent for the start and termination of an Applet are clear, and change the notification methods of PluginAppletEvent.
4	V4.0L20	PG88738	Web browsers hangs when trying to display a window with multiple applets pasted on it.
5	V4.0L20	PG93336	When the <object> tag is inserted using JavaScript, page offset and a smaller display area may occur for the applet that runs on JBK Plugin.
6	V5.0L10	PG86438	Focus traverses and the window flickers when the updateScrollPosition (Point pos, boolean pause) method of JFCScrollablePanel is called.
7	V5.0L10	PG90794	Strings edited with JFFieldDouble and JFFieldLong may not be drawn correctly.
8	V7.0L10	PG89802	Exception occurs when manipulating an item after hiding JFChoice when a JFChoice popup is being displayed.
9	V8.0.3	PG89086	The underlining in selected parts, and parts not selected, appear the same.
10	V9.0.0	PG85736	When using a Java applet focus does not return to the Java applet when the print dialog box is closed.
11	V9.0.0	PG87285	Focus is not set when the activateApplet() method is called.
12	V9.0.0	PG88871	Modal dialog boxes cannot be controlled properly when an applet that runs on JBK Plugin is run in Internet Explorer 9.
13	V9.0.0	PG94729	Improved internal interface of JBK Plugin.
14	V9.0.0	PG95992	Improved internal interface of GUI Library.
15	V9.2.0	PG86563	Performance of the cache function for JBK Plugin improved.
16	V9.2.0	PG87318	Improved internal interface of JBK Plugin.
17	V9.2.0	PG88776	When a setScrollbarDisplayPolicy() call and the scrollbars operation are performed simultaneously, the Java process loops.
18	V9.2.0	PG89914	When installers using JRE6 are created for JBK Download Installer, JBK Plugin may not be able to start.
19	V9.2.0	PG90061	Focus may move to an undetermined position when the transferFocus() method is called in a JBK GUI component.
20	V9.2.0	PG86079	Improved internal interface of GUI Library.
21	V9.2.0	PG86435	Improved internal interface of GUI Library.
22	V9.2.0	PG86436	Improved internal interface of GUI Library.
23	V9.2.0	PG86439	Improved internal interface of GUI Library.
24	V9.2.0	PG87219	Improved internal interface of GUI Library.
25	V9.2.0	PG92112	When a Java applet is running in standard-mode HTML in Internet Explorer 9, the calling JavaScript from an applet fails.
26	V9.2.0	PG92113	The frame is displayed from the applet, and the modal control of the browser window is controlled at the composition to which the frame displays the modal dialog.
27	V9.2.0	PG91679	Improved internal interface of GUI Library.

No.	Version	P number	Symptom
28	V9.2.0	PG94390	Improved internal interface of GUI Library.
29	V10.0	PG94984	When JTextArea is set to setBlink(true) in the JBK Plugin Applet, the strings do not flash and an exception occurs.
30	V10.0	PG95270	The strings of the JTextArea do not flash in the Graphical Editor.
31	V10.1	PG95498	Even though the jbk.plugin.ie.connection.max attribute has been added to the jbkplugin.properties files, the attribute value will not be reflected.
32	V10.1	PG90839	When a GUI component is used on an applet, java.security.AccessControlException occurs.
33	V11.0	PG95500	In the Java applet that uses JBK Plugin that runs in JDK7, PluginAppletEvent will be notified for every move of the focus.

### 4.3 JDK/JRE

No.	Version	P number	Symptom
1	V4.0L10	PG89521	During the running of Java application, the JIT may temporarily use a large amount of virtual memory.
2	V8.0.3	PG86013	The performance degradation might occur when using the DataInputStream which wraps the GZIP stream.
3	V9.0.0	PG94053	Unexpected termination of the Java VM may occur during the drag and drop operations in the GUI.
4	V9.2.0	PG87337	A redundant high-surrogate code is output when an XML with supplemental characters is converted to HTML.
5	V9.2.0	PG89082	JDK/JRE6 supports NIO with IPv6 on Windows.
6	V9.2.0	PG89624	Fujitsu Java VM includes the HotSpot VM maintenance release fixes of Oracle JDK/JRE 6 Update 30 to Update 31.
7	V9.2.0	PG89625	Enhancements to, and implementation of functions for troubleshooting, maintaining VM, and performing preventative monitoring for JDK/JRE 6 Java VM.
8	V9.2.0	PG93545	Fujitsu Java VM includes the HotSpot VM maintenance release fixes of Oracle JDK/JRE 6 Update 32 to Update 37.
9	V9.2.0	PG93546	Enhanced and implemented functions for troubleshooting and maintaining Java VMs in JDK/JRE 6.
10	V9.2.0	PG94021	The library revisions absorbed by Oracle JDK/JRE 6 Update 37 and prior updates are reflected in the Fujitsu JDK/JRE.
11	V9.2.0	PG94055	Internal interface improvement.
12	V9.2.0	PG95269	During the use of jheap, an error dialog may be displayed with the jheap terminated.
13	V10.0	PG85121	Improved messages about the JDK tools included in Java SE 6 and Java SE 7.
14	V10.0	PG86723	The result specified for -XX:ParallelGCThreads=0 may be incorrect when using a parallel GC with CMS.
15	V10.0	PG89378	An illegal value is displayed for MaxNewSize in "jmap -heap".
16	V10.0	PG89838	Reflects to the Fujitsu JDK/JRE the fixes absorbed up to the Oracle JDK/JRE 6 Update31.

No.	Version	P number	Symptom
17	V10.0	PG93057	Icons obtained through getIcon in the java.beans.Introspector may become null.
18	V11.0	PG93159	A troubleshooting tool jcmd has been added to the Fujitsu JDK.
19	V11.0	PG93166	The Library revisions absorbed by Oracle JDK/JRE 7 Update 9 and prior updates are reflected in the Fujitsu JDK/JRE.
20	V11.0	PG93547	Fujitsu Java VM includes the HotSpot VM maintenance release fixes of Oracle JDK/JRE 7 Update 4 to Update 9.
21	V11.0	PG93548	Enhanced and implemented functions for troubleshooting and maintaining Java VMs in JDK/JRE 7.
22	V11.0	PG94054	During the rendering GUI components with the Graphics object to which the xor color is set through java.awt.Graphics#setXORMode(Color c1), a java.lang.InternalError may occur.
23	V11.0	PG94736	During the use of JDK/JRE 7 fjtrace, the exception stack trace will be output, and the Java process will be terminated.



# Index

---

[A]	
Applet support library.....	16
Apworks Runtime.....	16
[B]	
builders.....	15,16
[F]	
form extension function.....	16
FTP client.....	16
Fujitsu XML.....	13
Fujitsu XML Processor.....	13
[G]	
Graphical Editor.....	16
[I]	
IJServer.....	13
IJServer launch configuration.....	15
Interstage management service operation tool.....	6,12
Interstage management services.....	12
[J]	
Java applet launch configuration.....	18
JDK/JRE 6.....	16
JDK/JRE 7.....	16
[R]	
resident service.....	12
[S]	
SOAP.....	17
SPT.....	17
[T]	
Tomcat launch configuration.....	15
[W]	
Web service.....	17
Web site tools.....	17