

Multichannel Bank Transformation Toolkit
Version 8.0

*Migrating from WebSphere
Multichannel Bank Transformation
Toolkit version 4.3 to WebSphere
Multichannel Bank Transformation
Toolkit version 8.0*

IBM

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 19.

This edition applies to Version 8, Release 0, Modification 0, of *IBM WebSphere Multichannel Bank Transformation Toolkit* (5724-H82) and to all subsequent releases and modifications until otherwise indicated in new editions.

IBM welcomes your comments. You can send to the following address:

IBM China Software Development Lab
Bank Transformation Toolkit Product
Diamond Building, ZhongGuanCun Software Park, Dongbeiwang West Road No.8,
ShangDi, Haidian District, Beijing 100193 P. R. China

Include the title and order number of this book, and the page number or topic related to your comment.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright IBM Corporation 1998, 2012.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Migrating from WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0 1

Concepts	1
Migration overview	1
Benefits gained through the migration	2
Migration tool architecture and support	3
Migration procedures	4
Phase 1: Analysis and preparation	4
Phase 2: Customizing the migration tool	5
Phase 3: Migrating your applications	6
Phase 4: Fixing errors and manual steps	6
Using the migration tool	7
Tasks	7
Setting up the migration tool	7
Adding rules	8
Migrating the dse.ini file	9
Migrating XML file	9
Migrating Java code	9

Migrating context	10
Migrating formatter	10
Migrating Channel	11
Migrating exception handling	11
Migrating externalizer	11
Migrating Service	12
Migrating JSP files	13
Migrating APIs	14
Generating reports	14
References	15
Manual migration	15
Code changes in dse.ini after migration	15
Initializer and extFile	15
Processor definition.	16
Operation definition	16

Notices	19
Trademarks	21

Migrating from WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0

This migration guide, provided by WebSphere® Multichannel Bank Transformation Toolkit version 8.0, enables users of WebSphere Multichannel Bank Transformation Toolkit version 4.3 to move up to the full power and versatility of IBM® WebSphere Application Server. It provides the migration path to allow an enterprise to reuse much of its existing toolkit application code base and still take advantage of the WebSphere J2EE environment.

IBM recognizes that an enterprise cannot always upgrade its application software all at once. Because the enterprise can decide how much or how little toolkit functionality to retain, WebSphere Multichannel Bank Transformation Toolkit version 8.0 enables the enterprise to upgrade to a full J2EE environment incrementally. With WebSphere Multichannel Bank Transformation Toolkit version 8.0, the enterprise can bypass the toolkit functionality and access J2EE components directly upon requests.

WebSphere Multichannel Bank Transformation Toolkit version 8.0 provides a migration tool that helps you to migrate your applications. The migration tool can migrate WebSphere Multichannel Bank Transformation Toolkit version 4.3 definitions to WebSphere Multichannel Bank Transformation Toolkit version 8.0 definitions, and migrate WebSphere Multichannel Bank Transformation Toolkit version 4.3 class packages and APIs to the corresponding WebSphere Multichannel Bank Transformation Toolkit version 8.0 class packages and APIs.

This migration guide describes the concepts and tasks that are related to migrating your applications to WebSphere Multichannel Bank Transformation Toolkit version 8.0 with the migration tool.

Concepts

For detailed information on the concepts that are related to migrating from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0, see the following sections.

Migration overview

IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0 provides the following functions to help you to migrate your applications from WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0:

XML migration

This feature transforms the definition files from WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0. All the existing definition files are converted or migrated to the corresponding WebSphere Multichannel Bank Transformation Toolkit version 8.0 definition files. For example, the `dse.ini` definition file is converted to the `btt.xml` definition file.

Java code migration

The migration tool scans the code of your applications and automatically

replaces the code of WebSphere Multichannel Bank Transformation Toolkit version 7.0, or earlier, with WebSphere Multichannel Bank Transformation Toolkit version 8.0 code by renaming packages and some classes and methods. This reduces the workload for the application program migration.

JSP migration

The JSP migration tool identifies Java code in JSP pages and replaces the code of WebSphere Multichannel Bank Transformation Toolkit version 7.0, or earlier, with WebSphere Multichannel Bank Transformation Toolkit version 8.0 code by renaming packages and some classes and methods.

Migration report

After the migration, a migration report is generated that provides information on the successfully migrated code and the code that must be migrated manually.

Customer Extension Support

The migration tool includes default and common customer rules, which have been identified in real migration scenarios, to improve automatic migration. If you have extended or modified WebSphere Multichannel Bank Transformation Toolkit extensively, you can customize your own migration rules to meet your migration requirements.

Exclusions

In WebSphere Multichannel Bank Transformation Toolkit version 8.0, Swing-based Java client is wholly compatible with WebSphere Multichannel Bank Transformation Toolkit version 4.3; therefore, no migration is required.

Constraints

A migration procedure can be automated only when the migration procedure has been written as rules, otherwise migration might need to be performed manually. This scenario occurs only if you have modified or extended WebSphere Multichannel Bank Transformation Toolkit code bases extensively.

Benefits gained through the migration

IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0 provides support for distributed architecture. The adoption of a distributed architecture provides benefits to the system planner and developer.

Migration provides the following benefits:

- Dramatically improved scalability. A version 8.0 system can scale from small- or medium-sized enterprises that have a single server to large enterprises with server farms that are distributed across multiple regional data centers.
- Performance can be tuned by balancing hardware and network usage against the expected user base. IBM Rational® Application Developer and IBM WebSphere Integration Developer provide many tools and guides for performing this tuning.
- Industry standards such as JCA, EJB, JMS, and Web services replace many previous architectures and facilities in WebSphere Multichannel Bank Transformation Toolkit. This adds flexibility to expanding enterprise applications that use third-party components that are built with the same industry standard architectures.
- To enhance its flexibility, WebSphere Multichannel Bank Transformation Toolkit maintains much of its previous external design philosophy. WebSphere Multichannel Bank Transformation Toolkit adds further configurability by

enabling deployment information to be configured. Scaling and performance tuning are all performed by using configuration settings rather than Java code changes.

- Platform flexibility is provided using this configurability as well. WebSphere Multichannel Bank Transformation Toolkit version 8.0 can be executed by using Rational Application Developer for small scale deployments or it can make use of the enhanced features of WebSphere Integration Developer to expand the available options to much larger enterprise deployments.

Migration tool architecture and support

IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0 provides a migration tool to help you migrate the applications that you developed with previous versions of the toolkit to the WebSphere Multichannel Bank Transformation Toolkit version 8.0 architecture.

A high scalability architecture has been applied to this rule-based migration tool. Under this architecture, you can easily extend the toolkit to meet your specific requirements. It is much like a generic migration toolkit, because it can be used in most Java application migration cases besides WebSphere Multichannel Bank Transformation Toolkit. Figure 1 shows the architecture of the rule-based migration tool.

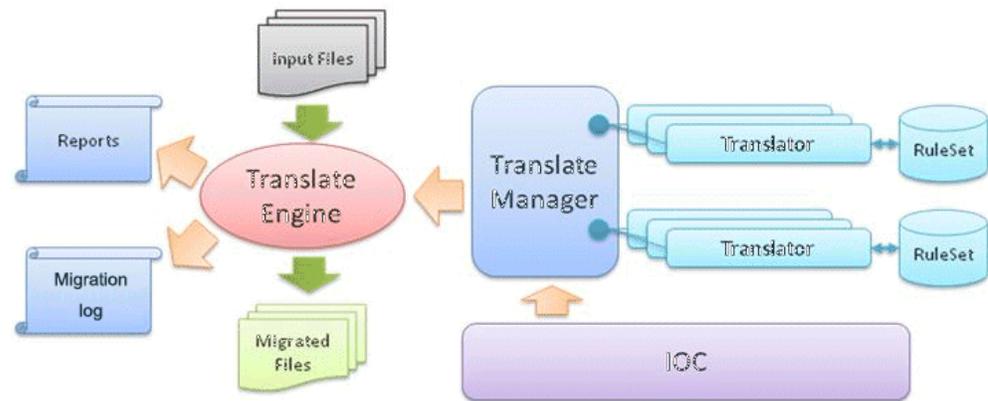


Figure 1. Migration tool architecture

The functional descriptions of the migration tool are provided in the following list:

- Migrate Java code:

From WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0, there are API changes to the same functions. It is time-consuming to replace the old API calls with the new ones. The Java code migration function of the migration tool enables Java code to be migrated automatically.

- Migrate the dse.ini file:

The dse.ini file contains most of the configurations for all of the WebSphere Multichannel Bank Transformation Toolkit components. In WebSphere Multichannel Bank Transformation Toolkit version 8.0, the dse.ini definition file has been converted to the btt.xml definition file. The migration tool helps you to migrate this file.

- Migrate other XML files:

In WebSphere Multichannel Bank Transformation Toolkit, contexts, formatters, services, data elements, and operations are externalized into XML files. Because

the referenced package names and class names of these XML files are different in different versions of WebSphere Multichannel Bank Transformation Toolkit, the migration tool automatically changes the package names and class names.

- Generate migration report:

After you run the migration tool, a report is generated to record the tasks that the migration tool performed. The migration report includes information on files that have been migrated by the tool and files that must be migrated manually.

The migration tool is rule-based, which means that the migration depends on the given rules. You can change the rules according your migration requirements.

Migration procedures

Migration procedures consist of tasks that must be executed to migrate to IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0. The procedures described in this section are procedures that are carried out in steps 3 – 5 of the migration process, which are described in the Migration process topic.

The migration procedures can be divided into the following four phases:

- Phase 1: Analysis and preparation.
- Phase 2: Customizing the migration tool.
- Phase 3: Migrating your applications.
- Phase 4: Fixing errors and manual steps.

The phases are iterative and contain tasks that might be refined during the project life cycle.

For detailed information on each phase, refer to the following sections.

Phase 1: Analysis and preparation

In this phase, you need to analyze the differences between IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 and WebSphere Multichannel Bank Transformation Toolkit version 8.0, especially the application logic layer.

About this task

You must identify the parts of IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 that can be migrated automatically and the parts that cannot be migrated automatically, and then make necessary preparation for the migration. To prepare for the migrate, you must do the following tasks:

1. Gather all the definition files of the existing application system version 4.3. The migration tool can migrate the definition files on the server side from WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0. The migration tool can process the definition contents based on the predefined rule. All unknown tags are copied to the new definition files, which may not is the correct place for new application. In this case, you need record the changes and manual review.
2. The migration tool can migrate Java code of the business logic of your applications to WebSphere Multichannel Bank Transformation Toolkit version 8.0. WebSphere Multichannel Bank Transformation Toolkit version 8.0 uses the processor, operation, and operation step on the server side of WebSphere Multichannel Bank Transformation Toolkit, which is the same as in WebSphere Multichannel Bank Transformation Toolkit version 4.3. The differences are in package names, class names, and changes of APIs. The migration tool can

migrate WebSphere Multichannel Bank Transformation Toolkit version 4.3 classes and APIs to the corresponding WebSphere Multichannel Bank Transformation Toolkit version 8.0 classes and APIs.

3. Migrate WebSphere Multichannel Bank Transformation Toolkit version 4.3 services. The services that are supported by WebSphere Multichannel Bank Transformation Toolkit version 8.0 on the server side does not support all of the WebSphere Multichannel Bank Transformation Toolkit version 4.3 services, and the SNA LU0/6.2 communication services are changed to JCA connectors. You must find a substitute solutions for these unsupported services or implement a new service based on the new service architecture. For example, IBM Lotus Notes® support. The migration tool can migrate the server side service definitions with customized migration rules, but you must find a solutions for the unsupported services.
4. The event mechanism is changed to fit the WebSphere Multichannel Bank Transformation Toolkit version 8.0 framework so that it can be distributed and span different servers. However, the event mechanism is still compatible with the event mechanisms of all releases of WebSphere Multichannel Bank Transformation Toolkit version 4. You must decide if they override the standard client server event code and standard.
5. Migration does not include Swing to WebSphere Multichannel Bank Transformation Toolkit XUI migration. You can select to stay with Swing or produce a migration roadmap with WebSphere Multichannel Bank Transformation Toolkit support.

Phase 2: Customizing the migration tool

The IBM WebSphere Multichannel Bank Transformation Toolkit migration tool is a rule-based tool and provides two set of rules: version 4.3 to version 8.0 rules, and version 5.2 to version 8.0 rules. By default, the migration tool uses the version 5.2 to version 8.0 rules. If you want to migrate from WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0, you must customize the migration tool to use the WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0 rules.

About this task

WebSphere Multichannel Bank Transformation Toolkit version 8.0 provides two sets of rules under the migration tool installation directory: `config\52to71` and `config\43to71`.

Procedure

To customize the migration tool to use the WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0 rules, do the following steps:

1. In Rational Application Developer, click **Window > Preferences**.
2. In the Preferences window, expand the **BTT** folder, and then click **Migration**.
3. In the Migration page, verify the following information:
 - a. In the **Java File Migration Configuration** field, you must select the `config\43to71` file.
 - b. In the **External File Migration Configuration** field, you must select the `config\43to 71` file.
 - c. In the **btt.xml Template File** field, you must select the `config\43to71` file.
 - d. In the **JSP File Migration Configuration** field, you must select the `config\43to71` file.

- e. In the **Encoding Settings** field, select the encoding for your JSP application. For better compatibility, it is recommended that you select **UTF-8**.

Phase 3: Migrating your applications

After you have selected the correct configuration files, you can use the migration tool to migrate your applications.

Before you begin

Before you migrate your applications, you must make sure that you have selected the correct configuration files. For more information on selecting the correct configuration files, refer to the “Phase 2: Customizing the migration tool” on page 5 section.

Procedure

To migrate you applications from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0, do the following steps:

1. Replace the WebSphere Multichannel Bank Transformation Toolkit version 4.3 JAR files with WebSphere Multichannel Bank Transformation Toolkit version 8.0 JAR files, and change the build path of the project to the new JAR files.
2. Import the application that you want to migrate to the workspace as a project, and configure the migration tool. The migration tool will migrate the application directly based on the original project
3. Migrate the `dse.ini` configuration file to `btt.xml`. Because the configuration file in WebSphere Multichannel Bank Transformation Toolkit version 4.3 is `dse.ini`, the migration tool migrates the `dse.ini` file to `btt.xml`.
4. Migrate the definition files of the context, the data element, the type data, and the format.
5. 5. Migrate the business logic. WebSphere Multichannel Bank Transformation Toolkit version 8.0 uses process, operation, and operation step as channel aware business logic, which is similar to WebSphere Multichannel Bank Transformation Toolkit version 4.3. In WebSphere Multichannel Bank Transformation Toolkit version 8.0 the architecture is refined and some components have been redesigned, so the package names, class names, and some APIs in WebSphere Multichannel Bank Transformation Toolkit version 8.0 are different from those in WebSphere Multichannel Bank Transformation Toolkit version 4.3. The migration tool will automatically migrate these differences.
6. Manually migrate the components that cannot be completely migrated by the migration tool.

Phase 4: Fixing errors and manual steps

After you have migrated your applications from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0, you must add the business logic and fix errors.

About this task

To add the business logic and fix errors, take the following steps:

Procedure

1. Fix the Java build errors. There might be some API mismatch problems between WebSphere Multichannel Bank Transformation Toolkit version 4.3 and

WebSphere Multichannel Bank Transformation Toolkit version 8.0. The migration tool can handle the standard usage. If not, use Rational Application Developer to find the correct reference and fix the errors.

2. Build the services that WebSphere Multichannel Bank Transformation Toolkit version 8.0 does not support, and change the access to the services.
3. Change the access to the communication services to comply with JCA.

Using the migration tool

About this task

The Bank Transformation Toolkit (BTT) provides a migration tool to help you migrate your toolkit applications from version 4.3 to version 7.0. The migration tool can perform the following tasks to help you migrate your applications:

- Convert the file *dse.ini* from BTT version 4.3 to the file *btt.xml* in BTT version 7.0.
- Migrate the definition files in BTT version 4.3 to the definition files in BTT version 7.0, including context definitions, formatter definitions, and so on.
- Migrate the application Java codes, including package names, class names, APIs, and so on from BTT version 4.3 to BTT version 7.0.
- Migrate the JSP Tablib uri, BTT customized tag, page import, and Java code in JSP file from BTT version 4.3 to BTT version 7.0.

The migration tool is rule-based. It migrates BTT version 4.3 applications to BTT version 7.0 based on the given migration rules. You can customize the migration rules to meet your special requirements.

For information about how to use the migration tool, see Migration Tool.

Tasks

After you understand the concepts that are related to the migration process, you can use the migration tools to migrate your applications from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0. For information on how to set up the migration tool and how to use the migration tool to perform the migration tasks, refer to the following sections.

Setting up the migration tool

Before you can migrate your applications from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0, you must set up the migration tool.

Before you begin

If you installed WebSphere Multichannel Bank Transformation Toolkit before you installed IBM Rational Application Developer on your workstation, you must copy the following plugin files to the $\$D(RAD)/plugins$ directory manually after you install Rational Application Developer:

- **com.ibm.btt.tools.migration.0**
- **com.ibm.btt.core.0**

Procedure

To set up the migration tool, take the following steps:

1. Start Rational Application Developer.
2. Import the project that you want to migrate.
3. In Rational Application Developer, click **Window > Preferences**.
4. In the Preferences window, expand the **BTT** folder, and then click **Migration**.
5. In the Migration page, select the migration rule definition files that you want to apply. In the migration tool plug-in, there is a **config** category, and you can find WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0 migration rule definition files under the related category.
6. Click **OK**.
7. In the Project Explorer view, right-click the BTT definition file and select **BTT Migration**.

Results

The migration report and migration results will be displayed in the Project Explorer view if there are any.

Adding rules

About this task

The migration tool enables different types of migration: Java migration, WebSphere Multichannel Bank Transformation Toolkit definition XML file migration, `dse.ini` file migration, and JSP file migration. The migration is implemented according to the configured migration rules.

Two sets of migration rules are provided: version 4.3 to version 8.0 migration rule set and version 5.2 to version 8.0 migration rule set. The rule sets are in the `$(RAD)\plugins\com.ibm.btt.tools.migration.0\config` directory. By default, version 5.2 to version 8.0 migration rule set is applied.

The following files are configuration files for the rule-based Migration tool:

- `javaRule.xml`: rules for Java code migration
- `cfgRule.xml`: rules for XML and `dse.ini` file migration
- `bttTemplate.xml`: `btt.xml` template.
- `jspRule.xml`: rules for JSP file migration

If the default rules do not meet your migration requirements, you can add new rules and configure them as the migration rules.

Procedure

To configure your own migration rules, do the following steps:

1. In IBM Rational Application Developer, click **Window > Preferences**
2. In the Preferences window, expand the **BTT** folder, and then select **Migration**. The Migration page displays in the Preferences window.
3. In the Migration page, select the migration configuration files that you require.

- If the default migration rules that are provided by the WebSphere Multichannel Bank Transformation Toolkit migration tool meet your migration requirements, select the default migration rules.
- If the default migration rules that are provided by the WebSphere Multichannel Bank Transformation Toolkit migration tool do not meet your migration requirements, define your own migration rules, and then click **Browse** to locate your migration configuration file.

Note: If you do not specify any migration rules in the Migration page, the default migration rules will be used during migration.

Migrating the dse.ini file

This section describes how to migrate the `dse.ini` file.

Procedure

In IBM Rational Application Developer, right-click the `dse.ini` file, and then click **BTT Migration > Migrate dse.ini**.

Results

The `dse.ini` file is migrated to the **config.migrated** folder.

Migrating XML file

This section describes how to migrate an XML file.

Procedure

In the Enterprise Explorer pane of IBM Rational Application Developer, right-click the XML file that you want to migrate, and then click **BTT Migration > Migrate XML File**.

Note: If you want to migrate multiple XML files at the same time, press Shift and select the XML files that you want to migrate, and then right-click the selected XML files.

Results

The XML files are migrated to the **config.migrated** folder. If you selected multiple XML files for migration, all the XML files that you selected are migrated to the **config.migrated** folder at the same time.

Migrating Java code

About this task

When you migrate Java code, you can select to migrate either individual Java files or packages.

Procedure

To migrate Java code, do the following steps:

In the Enterprise Explorer pane of IBM Rational Application Developer, right-click the Java file or package that you want to migrate, and then click **BTT Migration > Migrate Java Files**.

Results

The Java code is migrated.

Migrating context

This section describes how to migrate an IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 context to a WebSphere Multichannel Bank Transformation Toolkit version 8.0 context.

About this task

You do not need to change the context, type, and data definitions of WebSphere Multichannel Bank Transformation Toolkit version 4.3 when you migrate them to WebSphere Multichannel Bank Transformation Toolkit version 8.0 because these definitions are the same in version 4.3 and version 8.0. In WebSphere Multichannel Bank Transformation Toolkit version 8.0, a class table tag for context implementation class and an initializer tag are added in the CHA configuration in the `btt.xml` definition file.

The APIs in WebSphere Multichannel Bank Transformation Toolkit version 4.3 and WebSphere Multichannel Bank Transformation Toolkit version 8.0 are almost the same except the way of constructing the context instance. In WebSphere Multichannel Bank Transformation Toolkit version 8.0, the context is constructed from `ContextFactory`. The service reference of WebSphere Multichannel Bank Transformation Toolkit version 4.3 is no longer applicable in WebSphere Multichannel Bank Transformation Toolkit version 8.0. An example of migrating a WebSphere Multichannel Bank Transformation Toolkit version 4.3 context to a WebSphere Multichannel Bank Transformation Toolkit version 8.0 context is provided in the procedure in this topic.

Procedure

1. Creating a WebSphere Multichannel Bank Transformation Toolkit version 4.3 context.

```
Context ctxt = (Context) Context.readObject("myContext");
Context ctxt2 = new com.ibm.dse.base.Context();
ctxt2.setName("myContext2");
```

2. Creating a WebSphere Multichannel Bank Transformation Toolkit version 8.0 context.

```
Context diiTestCtx = ContextFactory.createContext("myContext");
Context ctxt2 = new com.ibm.btt.base.Context();
ctxt2.setName("myContext2");
```

Migrating formatter

About this task

To migrate a formatter, you must migrate the configuration that is related to the formatter in the `dse.ini` file in IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to the `btt.xml` file in WebSphere Multichannel Bank Transformation Toolkit version 8.0. Do not change the other code in the file.

Migrating Channel

About this task

The channel component has been restructured in IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0. In addition to `ChannelRequest` and `ChannelResponse`, which help to achieve protocol and channel independency, there are some other changes in structure and code, including:

- In WebSphere Multichannel Bank Transformation Toolkit version 4.3, `CSReqServlet` implements the `ChannelDriver` interface. In WebSphere Multichannel Bank Transformation Toolkit version 8.0, `CSReqServlet` is only a servlet and does not implement the `ChannelDriver` interface.
- In WebSphere Multichannel Bank Transformation Toolkit version 7.0, the `BTTChannelDriver` class was introduced, which implements the `ChannelDriver` interface. In WebSphere Multichannel Bank Transformation Toolkit version 8.0, the `BTTChannelDriver` class is still used to implement the `ChannelDriver` interface.
- Because the `BTTChannelDriver` class implements the `ChannelDriver` interface in WebSphere Multichannel Bank Transformation Toolkit version 8.0, some functions and methods in `CSReqServlet` are not in `ChannelDriver`.

To migrate channel code, you must change some methods and logic to the classes in WebSphere Multichannel Bank Transformation Toolkit version 8.0.

Migrating exception handling

About this task

In IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3, APIs such as `context.getKeyedCollection()`, `context.getValue()` do not throw out exceptions; while in WebSphere Multichannel Bank Transformation Toolkit version 8.0, these APIs throw exceptions.

As a result, you must migrate exception handling and you must trace the exceptions.

The following code is the code sample before migration:

```
return utb.getContext().getKeyedCollection().getElements().entrySet();
```

After migration, the code is as follows:

```
try {
return utb.getContext().getKeyedCollection().getElements().entrySet();
} catch (DSEInvalidRequestException e) {
// TODO: Exception handling
//Trace the exception.
}
return null;
```

Migrating externalizer

About this task

To migrate the externalizer, you must migrate the `getSetting()` parameter and the `iniFile` path.

- Migrating the `getSetting()` parameter:

In IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0, each component has its own initializer; therefore, you need to use the initializer of each component to retrieve the parameter.

Following is the code sample before migration:

```
String startupOpName = (String) Settings.getSettings().getValueAt("channelHandlers." +
    channelContext.getDeviceType() + ".startupOp");
```

After you migrate to WebSphere Multichannel Bank Transformation Toolkit version 8.0, the code is as follows:

```
String startupOpName = (String) ChannelInitializer.getSettings().
    getValueAt(channelContext.getDeviceType() + ".startupOp");
```

- Migrating iniFile path:

WebSphere Multichannel Bank Transformation Toolkit version 8.0 supports following types of iniFile:

- JAR path iniFile, such as jar:///package/dse.ini;
- File path iniFile, such as file:///path/dse.ini
- URL path iniFile, such as http://127.0.0.1:9080/BankWeb/path/btt.xml

If you use the file path method, you must migrate the iniFile in the StartServerServlet class:

```
String res=null;
if (!iniFileParameter.startsWith("/WEB-INF")) {
File iniFile=new File(iniFileParameter);
if (iniFile.isAbsolute() && iniFile.exists()) {
res=iniFile.getAbsolutePath();
}
}
```

```
res = "file:\\\\\\" + res;
}
```

Migrating Service

About this task

IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0 supports the definition of a service in a context. Following is the code example in the service.xml file:

```
<CSServer id="realCSServer" inactivityClientTimeout="900000"
timeBetweenSessionCheck="60000"/>
```

The code example in the context.xml file is as follows:

```
<context id="rootCtx" type="root">
<refKColl refId="rootData"/>
<refService refId="realCSServer" alias="CSServer" type="cs"/>
<refService refId="JDBCIB2ConnectionManager" alias="connectionManager"
type="jdbcManager"/>
<refService alias="proximaService" refId="proximaService" type="prx"/>
<refService alias="dummyService" refId="dummyService" type="prx"/>
<refService alias="messageResourcesSrv" refId="realMessageResourcesService"
type="multilan"/>
</context>
```

The following code is the Java code before migration:

```
Service.readObject(getCSServerServiceName());
```

After migration, the Java code is as follows:

```
((CSServerService)ContextFactory.getCSServer())
```

Migrating JSP files

About this task

You can migrate JSP Taglib uri, WebSphere Multichannel Bank Transformation Toolkit customized tags, page import, and Java code in JSP files. The migration rules of taglib uri and WebSphere Multichannel Bank Transformation Toolkit customized tags are defined in the JSP migration rule file. The migration rules of Java code are defined in the Java migration rule file.

To migrate JSP file, perform the following steps:

Procedure

To migrate a JSP file, do the following steps:

1. In the Enterprise Explorer pane of IBM Rational Application Developer, right-click the JSP file that you want to migrate. If you want to migrate multiple JSP files at the same time, press Shift and select the JSP files that you want to migrate.
2. Click **BTT Migration > Migrate JSP file**.
3. The JSP file is then migrated to the `jsp.migrated` folder.

Results

The JSP file is migrated to the **jsp.migrated** folder. If you selected multiple JSP files for migration, all the JSP files are migrated to the **jsp.migrated** folder at the same time.

The following tables describe the samples for each migration type:

Table 1. Taglib uri migration

Before migration	Migration rule	After migration
<pre><%@ taglib uri="/WEB-INF/dse.tld" prefix="dse" %></pre>	<pre><taglibRule oldTagUrl="/WEB-INF/ dse.tld" newTagUrl="/WEB-INF/ btt.tld" oldTagPrefix="dse" newTagPrefix="btt"/></pre>	<pre><%@ taglib uri="/WEB-INF/btt.tld" prefix="btt" %></pre>

Table 2. Tag migration

Before migration	Migration rule	After migration
<pre><dse:label text="jspMigrationToolTest"/></pre>	<pre><tagRule oldTitle="dse:label" newTitle="btt:label" oldKey="text" newKey="text_new" oldVal="jspMigrationToolTest" newVal= "jspMigrationToolTest_new" /></pre>	<pre><btt:label text_new= "jspMigrationToolTest_new" /></pre>

Table 3. Page import migration

Before migration	Migration rule	After migration
<%@page import="com.ibm.dse" %>	<pageImport oldPattern="com.ibm.dse" newPattern="com.ibm.btt"/>	<%@page import="com.ibm.btt"%>

Table 4. Java code migration

Before migration	Migration rule	After migration
com.ibm.dse.base. JavaExtensions. getAlphaUniqueCode()	<simpleRule oldItem="com.ibm.dse" newItem="com.ibm.btt" />	com.ibm.btt.base. JavaExtensions. getAlphaUniqueCode()

Migrating APIs

About this task

To migrate APIs, do the following changes:

- Use `Context ctx=(Context)ContextFactory.createContext(name, false)` to create context, where `false` refers to local context.
- Change `Context.reset()` to `ContextFactory.reset()`.
- Change `Context.getRoot()` to `ContextFactory.getRoot()`.
- Change `Context.getTIDForSession(s)` to `CSSessionHandler.getTIDForSession(string)`.
- Change `Context.removeSession(sessionId)` to `CSSessionHandler.removeSession(sessionId)`.
- Change `Context.getCurrentContextForSession(sessionId)` to `CSSessionHandler.getCurrentContextForSession(sessionId)`.
- Change `Context.setTimeStampForSession()` to `CSSessionHandler.setTimeStampForSession()`.
- Change `Context.setTypeForSession()` to `CSSessionHandler.setTypeForSession()`.
- Change `Context.addSession()` to `CSSessionHandler.addSession()`.

Generating reports

About this task

Migration reports are generated after migration that identify the tasks that were completed during the migration process. Separate migration reports are generated for Java code migration, JSP file migration and WebSphere Multichannel Bank Transformation Toolkit XML definition file migration.

Note: The migration report files will not be generated for the migration of the `dse.ini` file.

All the report files are generated in a folder named `report`. The `changehistory_*.txt` file records all of the items that have been migrated during the migration process. The `error_*.txt` file provides a record of all of the errors that occurred during the migration process. The `summary_*.txt` file provides a summary of all of the migrated files and its status.

References

For the reference information related to migration from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0, see the following topics.

Manual migration

The migration tool cannot migrate everything from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to WebSphere Multichannel Bank Transformation Toolkit version 8.0 automatically. The following features and components must be migrated manually to migrate to WebSphere Multichannel Bank Transformation Toolkit version 8.0:

- New features in WebSphere Multichannel Bank Transformation Toolkit version 8.0. The base functionality of the migration tool only migrates the existing applications to WebSphere Multichannel Bank Transformation Toolkit version 8.0 and makes sure that they can run in WebSphere Multichannel Bank Transformation Toolkit version 8.0. You must redesign your applications and implement them manually.
- Some services that are not within the scope of WebSphere Multichannel Bank Transformation Toolkit version 8.0. These services in WebSphere Multichannel Bank Transformation Toolkit version 4.3 application systems are not included in the migration tool, so you must migrate them manually, for example, notes services.
- The server side event mechanism of WebSphere Multichannel Bank Transformation Toolkit version 8.0 has been changed and is compatible with standard usage. You need pay attention to the overridden part of event mechanism
- The communication service access uses JCA connector in WebSphere Multichannel Bank Transformation Toolkit version 8.0. You must modify it manually

Code changes in dse.ini after migration

This section lists the code changes in the dse.ini file after you migrate your application from IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3 to version 8.0.

You can use the migration tool to migrate the dse.ini file. After migration, the file name is changed from dse.ini to btt.xml.

Initializer and extFile

In IBM WebSphere Multichannel Bank Transformation Toolkit version 8.0, the configuration of each component is defined in the component's kColl. Each component has its own initializer. All unknown first level tags will be put into settings clause which is compatible with WebSphere Multichannel Bank Transformation Toolkit version 4.3 usages.

In the following example, you can see that after the migration, each data's definition kColl has extFile and initializer:

Before migration, the code is as follows:

```
<kColl id="data">
<field id="field" value="com.ibm.btt.base.DataField"/>
<field id="iColl" value="com.fortis.be.rbaa.common.data.IndexedCollection"
```

```

    description="compound"/>
<field id="kColl" value="com.ibm.btt.base.KeyedCollection" description="compound"/>
<field id="operDef" value="com.ibm.btt.base.OperField"/>
<field id="refData"/>
<field id="sessionEntry" value="com.ibm.btt.base.SessionEntry"/>
<field id="sessionTable" value="com.ibm.btt.base.SessionTable"/>
<!-- Ban WSBC Common -->
<field id="finalField" value="com.bank.be.common.data.FinalDataField"/>
</kColl>

```

After migration, the code is changed to the following:

```

<kColl id="data">
<field id="extFile" value="dsedata.xml" />
<field id="initializer"
value="com.ibm.btt.base.DataInitializer" /><kColl id="classTable">
<field id="field" value="com.ibm.btt.base.DataField"/>
<field id="iColl" value="com.fortis.be.rbaa.common.data.IndexedCollection"
description="compound"/>
<field id="kColl" value="com.ibm.btt.base.KeyedCollection" description="compound"/>
<field id="operDef" value="com.ibm.btt.base.OperField"/>
<field id="refData"/>
<field id="sessionEntry" value="com.ibm.btt.base.SessionEntry"/>
<field id="sessionTable" value="com.ibm.btt.base.SessionTable"/>
<!-- Bank WSBC Common -->
<field id="finalField" value="com.bank.be.common.data.FinalDataField"/>
</kColl>
</kColl>

```

Processor definition

About this task

In IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3, the processor definition is as follows:

```

<kColl id="processors">
    <procDef id=".." value=".." path=".."/>
    ....
</kColl>

```

After you migrate to WebSphere Multichannel Bank Transformation Toolkit version 8.0, the processor definition is changed to the following:

```

<kColl id="processors">
    <kColl id="files">
        <procDef id=".." value=".." path=".."/>
        ....
    </kColl>
    ....
</kColl>

```

Operation definition

About this task

In IBM WebSphere Multichannel Bank Transformation Toolkit version 4.3, the operation definition is as follows:

```

<kColl id="operation">
    <operDef id=".." value=".." path=".."/>
    ....
</kColl>

```

After you migrate to WebSphere Multichannel Bank Transformation Toolkit version 8.0, the operation definition is changed to the following:

```
<kColl id="operation">
  <kColl id="files">
    <operDef id=".." value=".." path=".."/>
    ....
  </kColl>
  ....
</kColl>
```

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Lab Director
IBM China Software Development Lab
Diamond Building, ZhongGuanCun Software Park, Dongbeiwang West Road No.8,
ShangDi, Haidian District, Beijing 100193 P. R. China

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples may include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java is a trademark of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.