IBM WebSphere Multichannel Bank Transformation Toolkit
Version 8.0.1

Product Overview

Note!
Before using this information and the product it supports, be sure to read the general information under "Notices" on page 17.

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

© Copyright IBM Corp. 1998, 2012

WebSphere Multichannel Bank Transformation Toolkit overview

This document provides a high-level introduction to the IBM® WebSphere® Multichannel Bank Transformation Toolkit, a multichannel application platform which helps accelerate channel application development for financial industry. This document describes the benefits of using the platform and gives a brief description of the architecture, each of the provided components, and the development model.

The audience for this document is business and sales professionals, project managers, and anyone else who is interested in a high-level introduction to the WebSphere Multichannel Bank Transformation Toolkit. Solution architects and anyone requiring more understanding of the architecture of this product should refer to the Solution architecture overview section.

Introduction

Financial institutions are diversifying their offerings and adapting their products and services to ensure that they are able to respond to future market challenges and support changing business operations in an increasingly competitive environment.

The traditional channel application is becoming obsolete. Many existing channel application systems such as teller, internet banking are based on old technologies, such as financial-specific controllers or basic PC systems. These systems are no longer adequate or appropriate for meeting the challenges of the new environment, which include competitive factors such as the following:

- Reduced margin, especially in traditional products
- Increased competition
- Multiple channel environments
- Better informed customers who are sensitive to price and service quality
- Faster product introduction and reduced product life cycles

Financial institution services are mainly supported by applications whose core logic and data reside on host or enterprise systems. For a bank teller application, access to these services (for example, to conduct a withdrawal transaction) requires delivery channels and a transaction posting engine that can handle the many tasks involved with transaction processing. The delivery channel and transaction posting engine must be able to manage the user interface, gather operation data, build host messages, process host responses, log transaction information into an electronic journal, access financial devices, and all other activities involved with processing the transaction. In runtime, the IBM WebSphere Multichannel Bank Transformation Toolkit is the transaction posting engine used by many financial institutions and other organizations for accessing back-end systems for banking delivery channels such as the traditional branch, call center, banking kiosk, Internet banking, and mobile access. The IBM WebSphere Multichannel Bank Transformation Toolkit also provide the real-time marketing engine to provide targeted marketing, cross selling and up selling capabilities for multi-channel applications.

In the development side, the Bank Transformation Toolkit provides a set of facilities to help with each of the processes and concepts mentioned above,

© Copyright IBM Corp. 1998, 2012

modeling the real-life components of a channel application system as objects and presenting them to development teams in a very familiar way. It accomplishes the following:

- Implements a simple but effective architecture that ties all components together in a loosely coupled fashion and makes them highly independent of each other.
- Uses normal object-oriented techniques that enable you to adapt to specific customer requirements; but is also highly parametric, which is a "must" requirement for channel application systems.
- Abstracts the commonalties of local branch operations for financial transactions in a way that is easy to understand, develop, and maintain.
- Provides a way to deliver financial transactions as reusable and easily maintainable "model" objects.
- Provides an architecture and a class library that facilitate the structuring and development of teller applications by promoting reuse and providing the services required for the transaction processes.

In summary, the IBM WebSphere Multichannel Bank Transformation Toolkit product is a pragmatic infrastructure designed and built so that existing mission-critical systems can evolve rather than be replaced. Its architecture provides an environment for high development productivity and great flexibility to meet the challenges of the new pace of change in both technology and the banking industry.

To help you understand BTT value proposition better, the following diagram is the Positioning of Selected IBM Products:

Positioning of Selected IBM Products

	Lotus Expeditor	WebSphere Portal	WebSphere MBTT	WebSphere ESB	WebSphere Process Server (Includes ESB capabilities)	FileNet	WebSphere Business Services Fabric (Includes WPS capabilities)	WebSphere Message Broker
General Purpose RCP* Extensions	☆							
Banking specific RCP* Extensions			☆					
Web Portlet Container		☆						
Web At-the-Glass Integration		☆						
Screenflow			☆					
Channel Application Microflow			☆	☆				
Service Composition					☆			
Services Orchestration					☆			
Macroflow (BPEL & BSM)						☆		
Content and Human Centric Process Control							☆	
Dynamic Selection of Web Services				☆				☆
WS Routing and Transformation								

☆ = Recommended SWG Product * RCP = Eclipse Rich Client Platform

Rationale behind WebSphere Multichannel Bank Transformation Toolkit

IBM WebSphere Multichannel Bank Transformation Toolkit provides the following features:

- A proven runtime framework and a transaction posting engine for bank multichannel applications.
- A flexible extensible platform to adapt different requirement and better accelerate project implementation through creation and extension.
- Efficient development tools to facilitate a rapid development of different clients such as Teller, Internet banking, mobile banking, and so on.
- Easy integration with legacy and newer banking systems (mainframe, or via ESB).
- A real-time marketing engine to provide targeted marketing, cross selling and up selling capabilities for multi-channel applications.
- BTT solution templates showing leading practices for implementing Internet banking, and mobile banking solutions.

WebSphere Multichannel Bank Transformation Toolkit reduces your risk and cost and accelerates the development of your channel applications that might consist of large amount of transactions .

In runtime, the WebSphere Multichannel Bank Transformation Toolkit provides the infrastructure platform to run channel applications, a real-time marketing engine to provide targeted marketing and cross-selling campaign capability in channel applications. In development side, the WebSphere Multichannel Bank Transformation Toolkit provides a Java EE component-based framework that greatly accelerates multichannel application development for banks, providing an integrated development environment to development the mulitchannel applications.

Component-based application development is more cost-efficient and competitive than traditional methods. These benefits are realized through reduced requirements for software development skills and reduced development time. The value of these benefits continues to increase as the market demands increasingly sophisticated software applications at the same time that competitive pressures demand reduced time to market.

WebSphere Multichannel Bank Transformation Toolkit is well suited for building web-based financial services applications such as bank branch systems as well as building solutions for a wide variety of retail delivery channels, including Internet banking, call centers, stand-alone kiosks, automated teller machines (ATMs), and mobile access terminals such as wireless access protocol (WAP) capable cellular phones, or smart phones. The toolkit's multichannel support and dynamic component composition provide the foundation used to simultaneously meet the requirements of each of these retail delivery channels.

The toolkit is built on Java, an open industry standard and the object-oriented programming language of choice. Because the WebSphere software platform for e-business adheres to open industry and Internet standards, your investment is well protected. These standards include TCP/IP, HTML, HTTP, J2EE (Java, JSP, JCA, JDBC, EJB, and so on), Web 2.0 and Web services. The components of WebSphere Multichannel Bank Transformation Toolkit promote highly productive application development by supporting code reuse and the use of parameterization techniques to define business operations and their related objects. WebSphere Multichannel Bank Transformation Toolkit preserves investment in existing enterprise systems by providing specially designed components that can communicate with these systems.

WebSphere Multichannel Bank Transformation Toolkit is used to build applications with a multichannel architecture that extends the reach of a financial institution's information system services to all of its delivery channels. Financial institution services are most often supported by applications whose core logic and data reside on large-scale host systems. Financial service delivery channels, such as a bank teller application or an Internet banking application, must access transaction functions on these systems (for example, to transfer funds between accounts). The toolkit uses JCA connectors to integrate delivery channels with large-scale OLTP system. It includes components designed to handle all aspects of transaction processing for every channel: managing the user interface, providing navigation dialogs, gathering operation data, building host messages, processing host responses, logging transaction information, accessing financial devices, and more.

WebSphere Multichannel Bank Transformation Toolkit is highly customizable and its application is not limited to the financial services industry. Consider the toolkit as a potential solution to your transaction processing requirements no matter what your industry is.

The toolkit runtime provides National Language Support (NLS) for the following languages in group 1: Brazil Portuguese, French, Hebrew, Japanese, Korean, Simplified Chinese, Traditional Chinese, and Spanish. The toolkit also provides NLS for the following languages in group 2: Arabic. It also provides Bi-directional Languages Support (BIDI). The toolkit externalizes any end-user text or messages from runtime components in resource bundles.

WebSphere Multichannel Bank Transformation Toolkit provides pre-tested, user-configurable application components that can be quickly assembled into a complete financial services application. WebSphere Multichannel Bank Transformation Toolkit provides a number of tools that support the development of applications. All the tools are plug-ins of Eclipse.

Benefits of using WebSphere Multichannel Bank Transformation Toolkit

IBM WebSphere Multichannel Bank Transformation Toolkit is based on Java and other mature Internet technologies. This ensures that applications that are built with WebSphere Multichannel Bank Transformation Toolkit can be deployed with confidence as integral parts of robust production systems. The design of WebSphere Multichannel Bank Transformation Toolkit hides technical complexity from solution designers, which allows them to focus on business function rather than on the underlying technical details.

These features create benefits in the areas of project completion time, intermediate and long-term cost-effectiveness, and readiness for future changes, improvements, and evolution.

Cost-effective application development

IBM WebSphere Multichannel Bank Transformation Toolkit enables cost savings from the earliest stages of development planning all the way to deployment. Here are some of the ways that the toolkit reduces development costs:

Less reliance on high-level programming skills

From back-end connectors to user interface building blocks, the WebSphere Multichannel Bank Transformation Toolkit provides components that are easy to understand and use. This increases the size of the developer pool and reduces training costs.

Reduced development efforts by a rich set of development tools
WebSphere Multichannel Bank Transformation Toolkit provides a rich set of development tools that can help developers build WebSphere Multichannel Bank Transformation Toolkit transactions rapidly. The WebSphere Multichannel Bank Transformation Toolkit development tools also helps to shorten the development cycle.

Write once and deploy on several platforms
A toolkit-based application is portable across several platforms. Instead of a costly "from-the-ground-up" development effort for each target platform, you can define the application just once and then manage the deployment to any of the supported platforms.

Faster application development with fewer developers
The quicker an application can be developed the lower the cost. The toolkit's pre-built components reduce the number of person-hours that is required to complete an application.

Improved development team communication
Rational® Application Developer provides a team development environment. A common repository for the development products keeps the entire team synchronized and up-to-date. This avoids costly duplication of effort and rework.

Extendable and adaptable applications

Information technology is fast changing and so is the financial services market. An application developed for the needs of today can rapidly become obsolete. To protect you from this, every IBM WebSphere Multichannel Bank Transformation Toolkit solution has features that allow it to be easily extended and adapted. Changes can be made to match the evolution of IT systems, business expansion, business diversification, and other predictable or unforeseen shifts. Following are some of the ways that the toolkit allows you to compete instead of becoming obsolete:

Multichannel enabled
The multi-tier architecture enables component reuse among delivery channels. Transaction requests from every channel are handled by the enterprise systems in the same way as any other channel. This promotes uniform, consistent, and rapid deployment of financial services through all your delivery channels. New channels may be added as needed or as they become available.

Easy integration with existing (and upcoming) systems
WebSphere Multichannel Bank Transformation Toolkit integrates easily with present and future systems because it is based on open Internet standards such as HTML, SSL, HTTP, XML, TCP/IP, JavaBeans, Enterprise JavaBeans, JPA, WebServices, and JDBC. It includes JCA compliant SNA LU0 and LU62 resource adapters to facilitate its ability to develop applications that interact with other systems.

Platform portability and system scalability
Thanks to the portability of Java between operating systems and hardware platforms, toolkit applications can be ported from one operating system to another with minimal impact. You can easily change your branch platform (for example, from Windows to Linux), and you can also change the role played by your IT system components. For example, you may want to move part of the business logic from a branch system running on one

operating system to a regional or centralized system based on a different operating system; moving the logic between server levels and operating systems is a simple procedure.

Reduced risk

IBM WebSphere Multichannel Bank Transformation Toolkit provides a fast and competitive way to solve your application needs. However, being fast and competitive does not mean that you are left exposed to risk. Here are some of the ways that the toolkit reduces risk:

Systems work together
The extensive use of open computing industry standards (including Internet standards) protects against incompatibilities between systems.

Protection against obsolescence
The inherent flexibility and updateable nature of toolkit-based applications protects these applications from becoming obsolete.

Fast response to the business environment
The WebSphere Multichannel Bank Transformation Toolkit development environment allows quick changes to applications in response to changing business conditions.

Build it right the first time
The WebSphere Multichannel Bank Transformation Toolkit application development environment supports teamwork, and this in turn promotes dialog and sharing of ideas; fewer details will be overlooked.

Preserve stable IT infrastructures
The toolkit provides JCA LU0 and JCA LU62 Connectors for you to connect your toolkit applications to existing systems that have been providing reliable services. It also provides support for JMS, MQ, and Web services for modern IT infrastructures.

Faster time to market

The development approach that IBM WebSphere Multichannel Bank Transformation Toolkit promotes is designed to shorten development cycles and flatten the learning curve for the project team. The objective of this approach is to effectively save development effort, improve consistency, and reduce the time to market for all delivery channels. Following are some of the ways that WebSphere Multichannel Bank Transformation Toolkit reduces time to market:

Shortened development cycles
WebSphere Multichannel Bank Transformation Toolkit provides an environment that supports rapid application development by exploiting the benefits of component reuse. It does this by promoting the extensive use of object-oriented techniques and a high degree of application object parameterization.

Ready-to-use components
WebSphere Multichannel Bank Transformation Toolkit provides a set of pre-built infrastructure components with well-defined interfaces. The components are ready to be incorporated into delivery channel applications; a project team needs only to learn how to use them, not how to build them.

Leading practices solution template
WebSphere Multichannel Bank Transformation Toolkit provides solution sample for internet banking, and smart phone mobile banking, so project

team can build the complete internet banking and mobile banking by following leading practices of using toolkit and adding additional transactions.

Parametric application definition
WebSphere Multichannel Bank Transformation Toolkit reduces the effort that is required to add new functions to a toolkit-based application by providing the plug-ins to create the definitions for the function.

Flattened learning curve
The WebSphere Multichannel Bank Transformation Toolkit productivity tools hide the underlying technical details of the toolkit. This reduces the amount of time and effort needed by a project team to learn the features of WebSphere Multichannel Bank Transformation Toolkit and how to use them to deliver a solution. The development model creates a clear separation of roles that allows project team members to focus on their specific tasks.

Reduced application operating costs

After an application is deployed, the costs of operating the application become an important measure of success. IBM WebSphere Multichannel Bank Transformation Toolkit offers cost savings that take effect at and continue beyond deployment. Following are some of the ways that the toolkit reduces operating costs:

Preservation of back-end systems
Deployment of a toolkit-based application does not require changes in existing business logic or transactions run in back-end systems. The toolkit uses JCA connectors and invokers to connect existing back-end systems and the application located on a middle-tier server.

Reduced maintenance and operational costs
The use of the network computing architecture, which is based on Internet technologies, results in immediate cost savings on client administration, code distribution, and server management. In addition, WebSphere Multichannel Bank Transformation Toolkit solutions minimize the code distribution that is required for incremental changes.

Operational portability
If operational conditions require that the application be moved to another platform, this can be quickly performed since the application is platform-independent.

Ease of maintenance
During operation, it is common to discover that application changes are required. The environment and the distributed nature of the application supports easy, quick, and universal application updates no matter how many application delivery channels and users are affected.

Adjustments to suit available system resources
Technology and systems are subject to change; toolkit-based applications can quickly be adapted to take advantage of more system resources or compensated for a reduction in resources.

Reduced workstation requirements
The distributed architecture of toolkit-based applications reduces the resources that are required to deliver the application to the user. User workstations need to do little more than support the application presentation and any directly connected peripherals. Adding workstations is extremely cost-efficient since the server-based application can be distributed to any number of client workstations.

Common functionality across channels

An application can be designed to provide a common set of functions across multiple delivery channels. This consistent approach to service delivery promotes user satisfaction and reduces the training time needed if the user moves between channels.

Architecture

This section describes the general architecture of IBM WebSphere Multichannel Bank Transformation Toolkitby the figures below.

Figure 1 shows the architecture of WebSphere Multichannel Bank Transformation Toolkit.

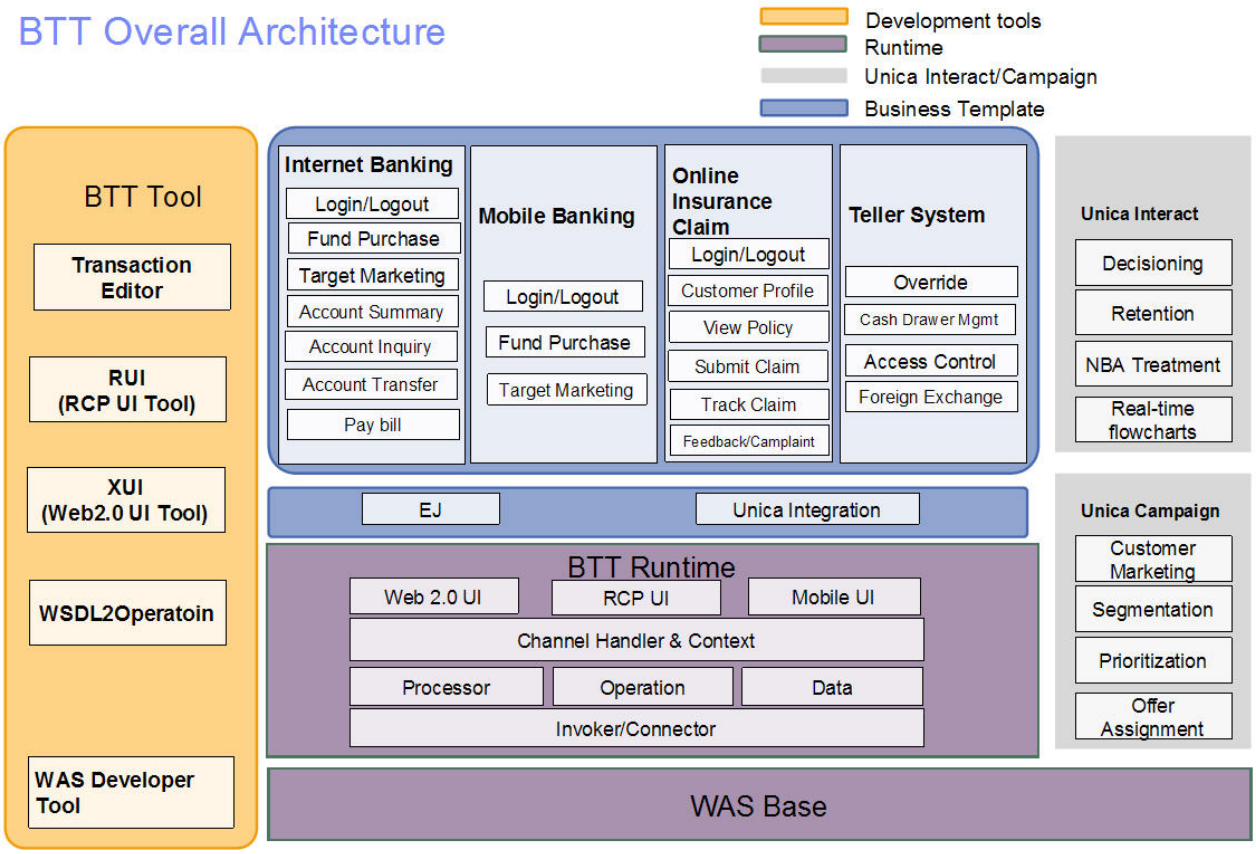


Figure 1. **The architecture of WebSphere Multichannel Bank Transformation Toolkit 8.0.1**.

Figure 2 on page 9 shows the tooling of WebSphere Multichannel Bank Transformation Toolkit.

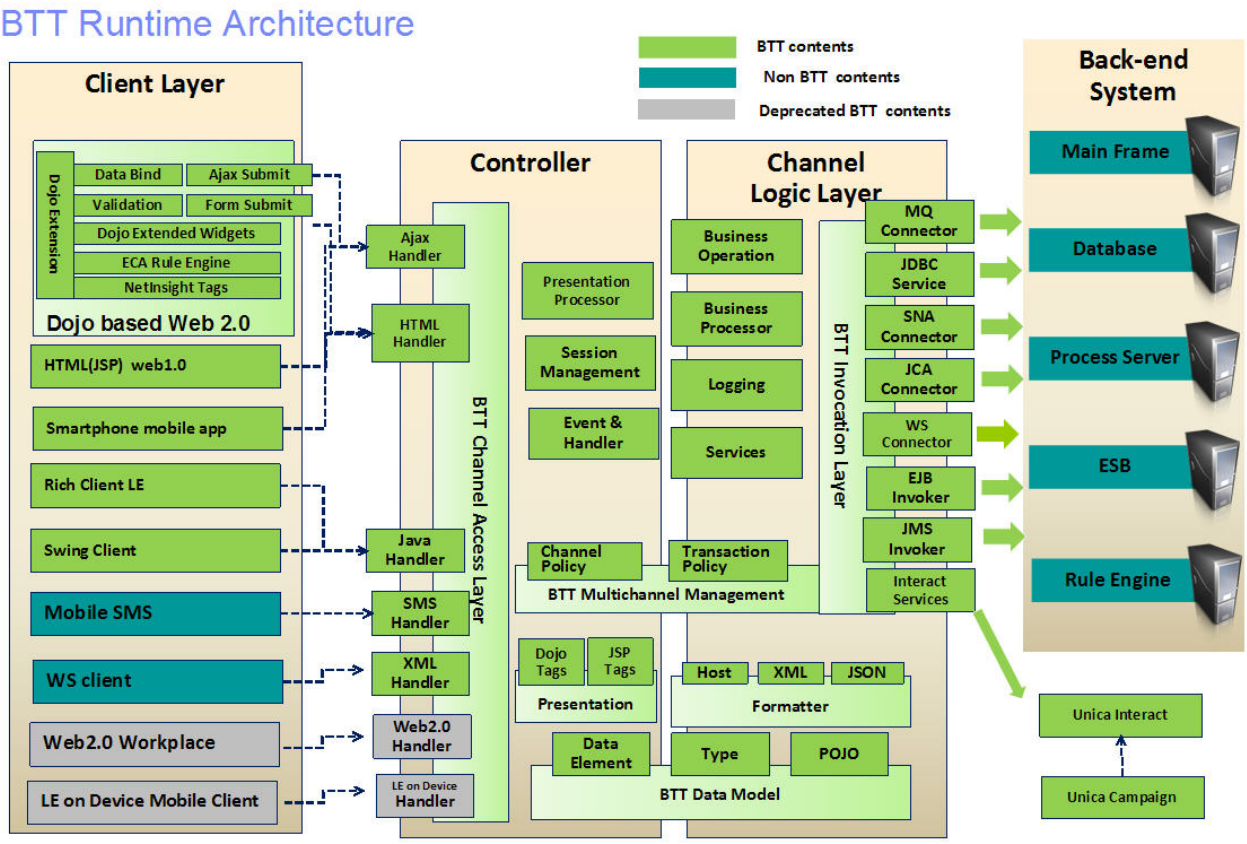


Figure 2. **The runtime architecture of WebSphere Multichannel Bank Transformation Toolkit8.0.1**.

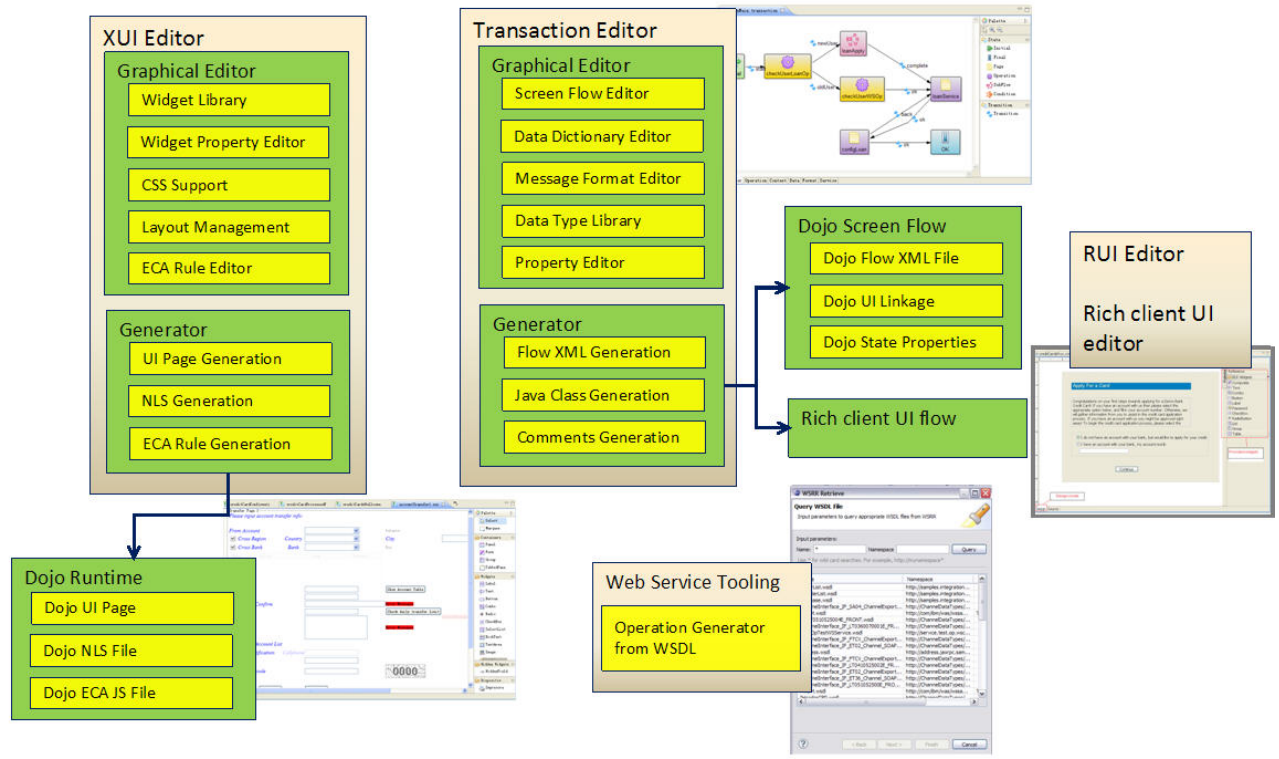


Figure 3. **The tooling architecture of WebSphere Multichannel Bank Transformation Toolkit8.0.1**.

Components

IBM WebSphere Multichannel Bank Transformation Toolkit contains the following components:

Core components

The Core components are the main entities of the IBM WebSphere Multichannel Bank Transformation Toolkit.

The core components include the following:

- Operation
- Flow
- Contexts
- Formatter
- Data element
- Initialization manager
- Element factory
- Exception
- Events
- Externalizer
- Trace

Presentation components

IBM WebSphere Multichannel Bank Transformation Toolkit provides components that facilitate the construction of the client presentation logic. The toolkit includes a set of SWT Visual beans and support for JavaServer Pages (JSPs).

The following presentation components are provided by WebSphere Multichannel Bank Transformation Toolkit:

- Web 1.0 support, which includes custom JSP tags and utility beans to enable applications to retrieve information from the operation context hierarchy, get resources, and handle errors. If your application requires additional behavior, you can build new tags by using the `JspContextServices` interface.
- Web 2.0 support, which includes Dojo-based UI support, a tag library, AJAX support and server-managed page flow.
- Mobile support, which includes Dojo mobile library.
- Short Messaging Service (SMS) support when integrated with an SMS gateway.
- XML UI Engine, which simplifies Dojo based Web2.0 UI design by defining the configuration files.
- RCP UI Engine, which simplifies RCP based rich client UI design by defining the configuration files.

Service components

IBM WebSphere Multichannel Bank Transformation Toolkit provides a set of service objects that enable an application to complete an operation. These services include host communications, journaling, store-and-forward for offline operations, financial devices for input and output operations, and more.

The service components include the following services:

- Communication services:
  - JCA LU0
  - JCA LU62
  - MQ connector
  - Web services connector
- Database services:
  - Database Table Mapping
  - Electronic Journal
  - Store
- LDAP Access service
- Generic Pool
- Unica service

Development tools

IBM WebSphere Multichannel Bank Transformation Toolkit provides a set of tools that enable you to develop applications. All the tools are plug-ins of IBM Rational Application Developer.

The following development tools are provided by WebSphere Multichannel Bank Transformation Toolkit:

- Deployment Descriptor editor
- Transaction editor

- XUI editor
- RUI editor
- Web services tooling
- Formatter Simulator
- Application Wizard
- Migration Tool

Runtime tools

Runtime tools are used to fetch or record information of the BTT runtime.

The Runtime tools include the following:

- Server Runtime Monitor
- Trace Facility

Supported platforms and technical requirements

This section describes the supported platforms and software that are required by each component of IBM WebSphere Multichannel Bank Transformation Toolkit. Because the toolkit is built with Java, any additional platform that provides the corresponding Java Virtual Machine is supported by the toolkit architecture.

A new solution with additional platforms may require changes to the toolkit to make it more generic so that the new solution can cope with the current platform as well as the new one. These changes may involve enabling the toolkit interfaces or components to support the new platform, and may be required for both the hardware and software components of the solution. In cases where native interfaces are required, a gap analysis is needed to support the new specific modules not provided by the toolkit. The components that are actually used depend on the specific requirements of each customer.

Components and platforms

In the following table, an X indicates that the service or component can be installed on that particular platform. Note that an application can access a service or component that is installed on another platform.

Table 1. Client components

Component name	Windows Vista	Windows 7	Windows XP	LinuxIntel
WebSphere Multichannel Bank Transformation Toolkit Core	X	X	X	X
WebSphere Multichannel Bank Transformation Toolkit Rich Client	X	X	X	X

Table 2. Application server components

Component name	Windows Server 2003/2008	AIX®	Solaris	Linux	
WebSphere Multichannel Bank Transformation Toolkit Core	X	X	X	X	
WebSphere Multichannel Bank Transformation Toolkit Channels	X	X	X	X	X
WebSphere Multichannel Bank Transformation Toolkit Business Components	X	X	X	X	X
WebSphere Multichannel Bank Transformation Toolkit Database Services	X	X	X	X	X
WebSphere Multichannel Bank Transformation Toolkit Invoker	X	X	X	X	X

Table 2. Application server components (continued)

Component name		Windows Server 2003/2008	AIX®	Solaris	Linux	
WebSphere Multichannel Bank Transformation Toolkit Services	LDAP Service	X	X	X	X	X
	MQ Service	X	X	X	X	X
	Interact Service	X	X	X	X	X
Communications	JCA LU0 or LU62	X	X	X		X
Database services	Database Table Mapping	X	X	X	X	X
	Electronic Journal	X	X	X	X	X

Table 3. Tools

Component Name	Windows Vista	Windows 7	Windows XP	LinuxIntel
Application Wizard	X	X	X	X
Transaction Editor	X	X	X	
Deployment Descriptor Editor	X	X	X	X
XUI Editor	X	X	X	
RUI Editor	X	X	X	
Formatter Simulator	X	X	X	X
WebSphere Multichannel Bank Transformation Toolkit Migration Tool	X	X	X	X

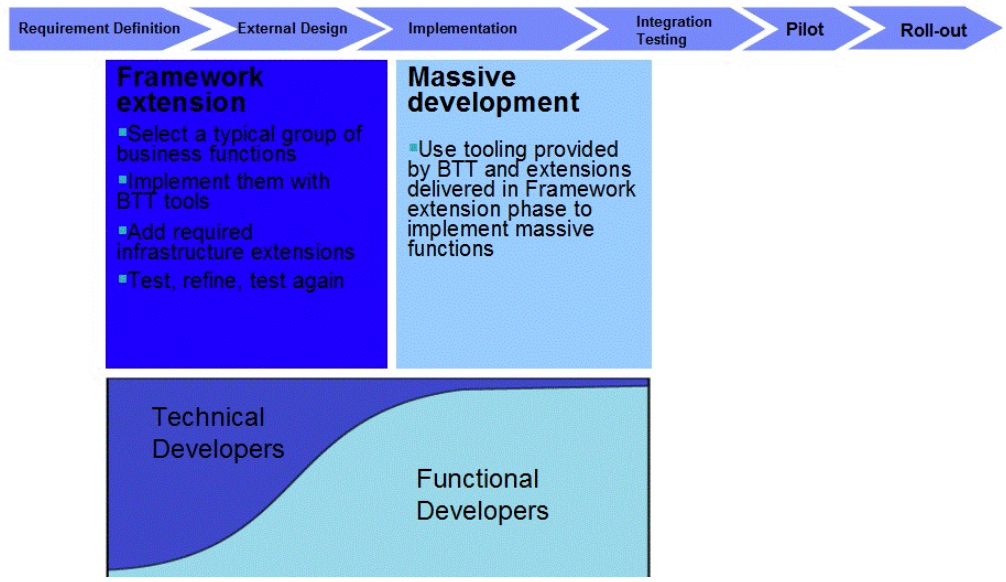
Components and technical requirements

The following table shows the additional technical prerequisites of the IBM WebSphere Multichannel Bank Transformation Toolkit components. For information version numbers, see the **** MISSING FILE **** topic.

Table 4. Additional technical prerequisites

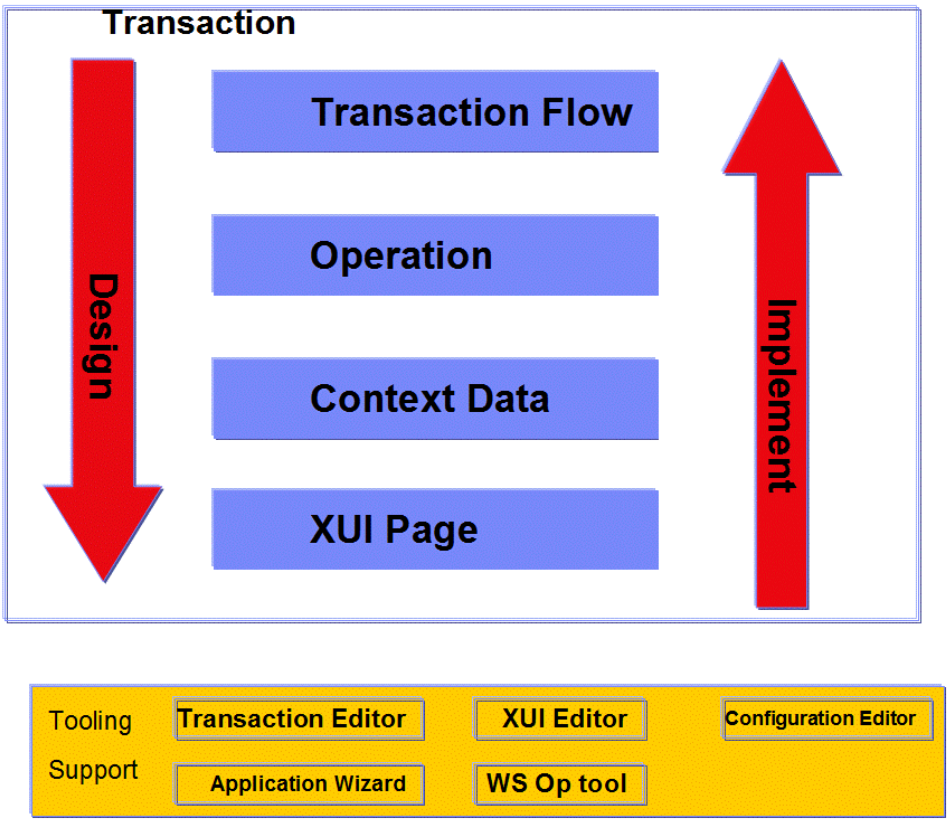
Component name	Technical requirements
LDAP Service	IBM Tivoli® Directory Server
MQ Service	IBM WebSphere MQ

Application development



You can develop applications rapidly by using the WebSphere Multichannel Bank Transformation Toolkit tooling. You can also extend the WebSphere Multichannel Bank Transformation Toolkit to suit your requirements; for example, you can extend WebSphere Multichannel Bank Transformation Toolkit to implement your own data types of UI widgets.

Application development with WebSphere Multichannel Bank Transformation Toolkit reduces the level of skills that are required of a development team. A high level of technical knowledge is required only from those developers who are responsible for extending the WebSphere Multichannel Bank Transformation Toolkit framework and development platform building. Functional developers can then focus on creating the business logic and the UI with WebSphere Multichannel Bank Transformation Toolkit tooling. WebSphere Multichannel Bank Transformation Toolkit generates the technical infrastructure code of WebSphere Multichannel Bank Transformation Toolkit framework and transaction definition and classes. In many cases, a function developer might not even be required to write any code. We call it zero code development of massive transactions.



WebSphere Multichannel Bank Transformation Toolkit application development involves first top down design and a bottom up implementation with development tooling. Firstly, you need to design the business flow of the transaction; we need to break down the transaction into operations and views, and the transition between them. Then you need to design the data context of the flow and operations and the data map between flow and operation. At the same time, we will design the UI layout and data binding between the UI widgets and data context of flow and operations.

In the implementation phase, you build an application by using UI widgets, data, and operations. Then you assemble the operation and XUI pages into a transaction flow by using the Transaction flow editor.

For more information on the design and implementation phases of the application development process, see the **** MISSING FILE ****.

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.

© Copyright IBM Corp. 1998, 2012

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.