IBM® **WebSphere**® **Everyplace**® **Deployment for Windows and Linux Version 6**

Client Platform Overview

@business on demand.

This presentation provides an overview of the client platform provided by IBM WebSphere Everyplace Deployment for Windows and Linux Version 6.

IBM

Goals

- Understand the client platform provided by IBM WebSphere Everyplace Deployment for Windows and Linux Version 6

The goal of this presentation is to understand the client platform provided by IBM WebSphere Everyplace Deployment for Windows and Linux Version 6.

Agenda

- Managed Client Container

- Client Services
  - ▶ Interaction Services
  - ▶ Managed Client Services
  - ▶ Platform Management
  - ▶ Access Services

The agenda of this presentation is to provide an overview of the managed client container and client services provided by IBM WebSphere Everyplace Deployment for Windows and Linux Version 6.

IBM

Section

Managed Client Container

Let's start with an overview of the Managed Client Container.

Managed Client Container
An architecture to enable the eco-system

Solutions

| Telematics | Mobile | Factory | Home | Office |

Platform

Interaction Services

Access Services

Platform Management System

Managed Client Services

J2EE

Device

Server

IBM WebSphere Everyplace Deployment for Windows and Linux 6.0 Client Platform Overview   © 2005 IBM Corporation
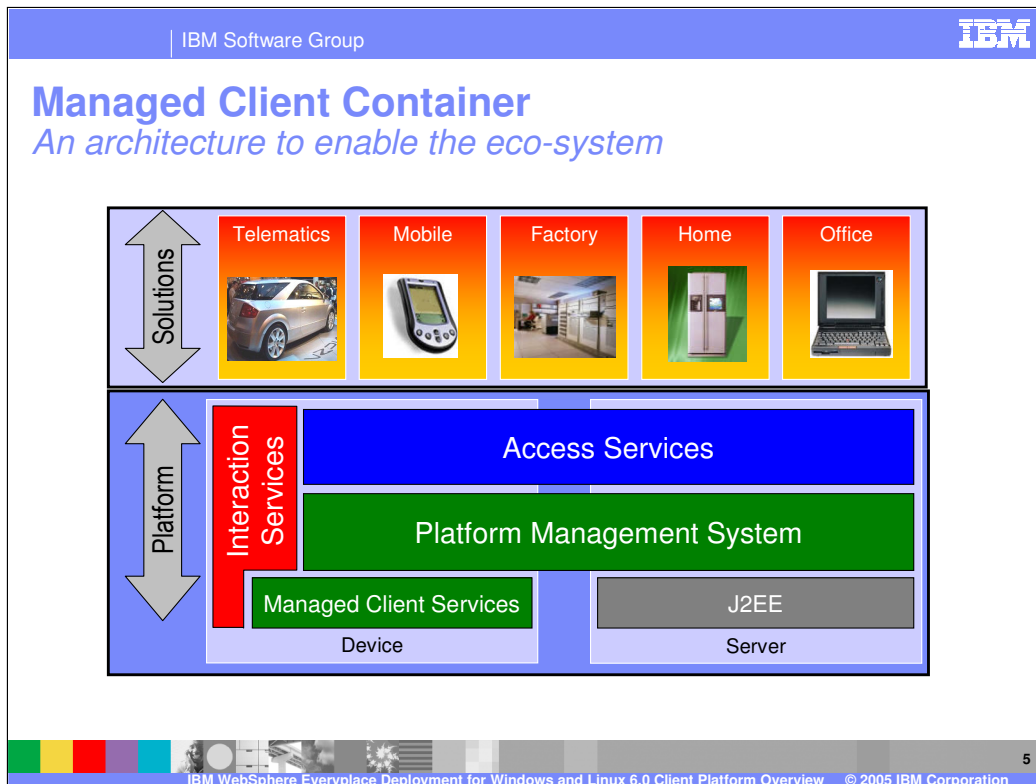
The managed client container provides a set of standards-based *Client Services* on which you can build your applications. These Client Services include:

• **Managed Client Services to** enable multiple applications and services to run on the same JVM, support life cycle management of these applications and services, and provide application portability across Windows and Linux operating systems. Managed Client Services form the foundation of the client platform.

• **Platform Management** to install, maintain, and configure the applications and services on the client.

• **Access Services** to provide end-to-end services that enable client applications to connect to server applications through a set of standards-based middleware, which include selected J2EE services extended to the client platform, so your client applications can securely access e-business applications, services and data and continue to operate even when a device is offline.

• **Interaction Services** to support interaction with end-users through a traditional Web browser and a rich graphical user interface through Eclipse technology.
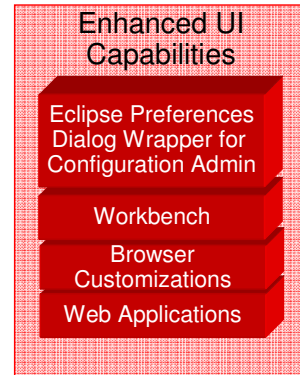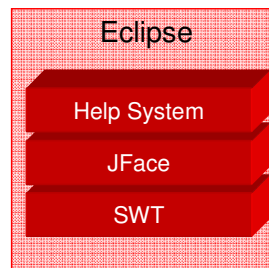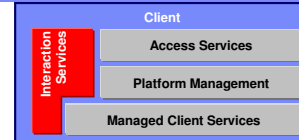
Section

Client Services

Next, let's highlight the value, standards, and major components for each of these Client Services.

IBM

Interaction Services

Client

Interaction Services	Access Services
	Platform Management
	Managed Client Services

- Value
  - ▸ Provides "fit for purpose" application user interface
  - ▸ Provides integrated workbench
- User Interface Models
  - ▸ Rich Client Application Model ("Reactive")
  - ▸ Web Application Model ("Request and Response")
- Standards
  - ▸ Eclipse
  - ▸ W3C

Enhanced UI Capabilities

Eclipse Preferences Dialog Wrapper for Configuration Admin

Workbench

Browser Customizations

Web Applications

Eclipse

Help System

JFace

SWT

7

The client platform includes key interaction services provided by Eclipse and adds enhanced user interface capabilities that result in an integrated Workbench window from which your end-users can install, launch and manage one or more applications. You can deliver rich client applications that use Eclipse to present a graphical user interface and Web applications that enable a Web browser to render the user interface.

Interaction Services
Major Components

- Workbench
  - Integrated application desktop window
  - Browser perspective
  - Application perspective
  - Branding
- Eclipse
  - Standard Widget Toolkit (SWT)
  - JFace
  - Help System
- Browser customizations
  - Control Web browser bars that appear in browser perspective
- Eclipse preferences dialog wrapper for configuration administration
  - Enables end-users to view and change application-specific preferences

Client
Access Services
Platform Management
Managed Client Services
Interaction Services

8

The **Workbench** provides an integrated application desktop window so end-users can install, manage and launch one or more applications within a single window. The Workbench provides the default look-and-feel that incorporates the base menus, images, and application launcher. The Workbench presents each application individually in its own perspective, only one of which is visible at any given time. When an end-user selects an application from the Workbench, the Workbench launches the perspective for that application. The Workbench supports a Web browser perspective for Web applications and an application perspective for all other applications. You can also modify the user interface of the client workbench to include your own **branding**. You can modify such elements as the title bar, splash screen, icons and images, and the About dialog.

You use the Eclipse **Standard Widget Toolkit (SWT)** and the **JFace toolkit** to develop the GUI for rich client applications. SWT provides a cross-platform API that tightly integrates with the native widgets of the operating system and, therefore, gives your applications a look and feel that makes them virtually indistinguishable from native applications. The JFace toolkit provides a set of components and helper utilities that simplify many of the common tasks in developing SWT user interfaces.

You use the **Help System** so end-users can select Help in the workbench to view helpful information about your applications.

You may optionally use **browser customizations** to control the browser bars that appear when the Workbench launches your Web applications.

You can use Eclipse **preferences** with OSGi configuration administration to persistently store your applications' preferences so end-users can view and set these preferences from the Workbench. Your applications can also be notified about configuration changes and System Administrators can remotely query and update this configuration.

Rich Client Application
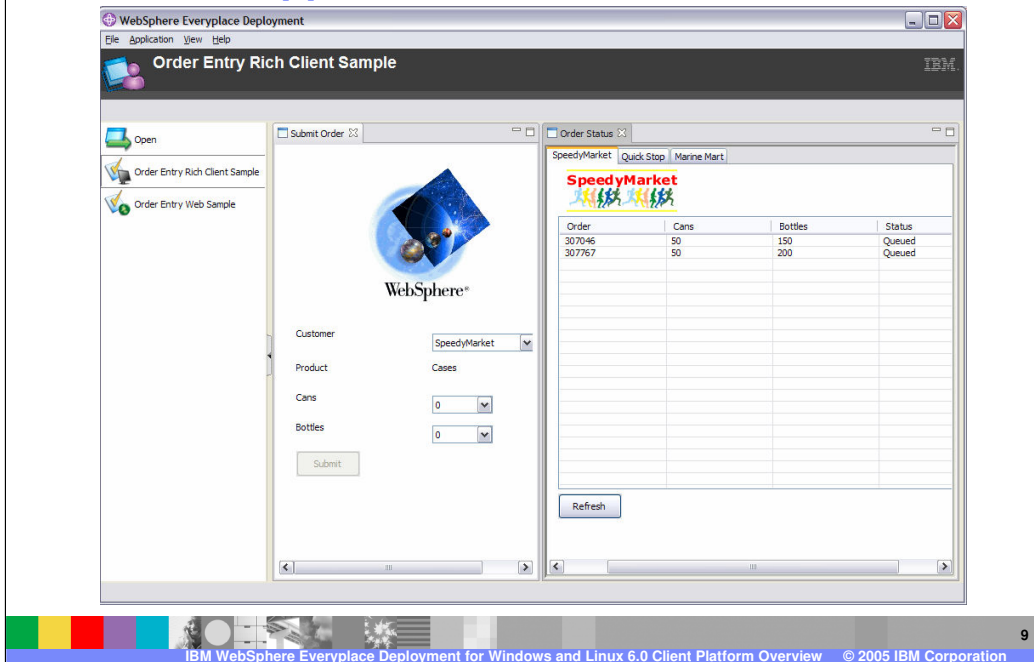
This screen shot shows a rich client application running in the workbench. The application has two views so you can submit orders while simultaneously viewing the status of orders they have already submitted.
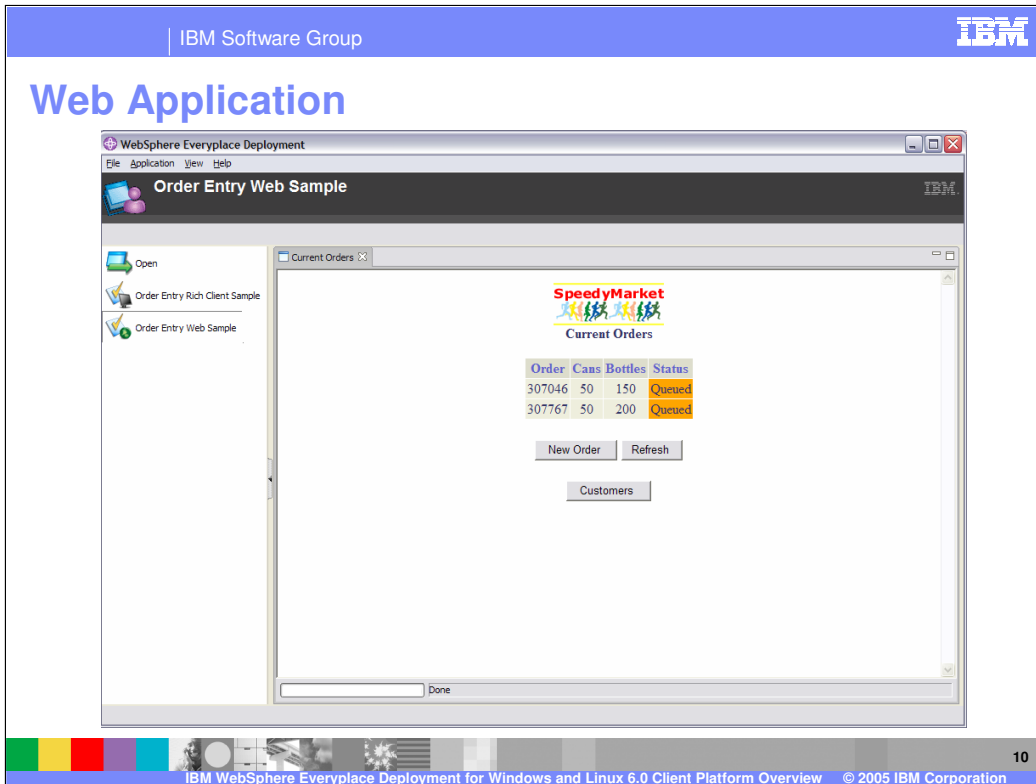
Web Application

This screen shot shows a Web application running in the workbench. Notice that the user experience consists of a single view. Now, you must go to another view to submit new orders for the current customer or to select another customer.

Notice that the both the Order Entry Web Application and the Order Entry Rich Client Application are running in the same Workbench as shown in the left-most frame of the window, called the Application Switcher.

Managed Client Services

Client

Access Services

Platform Management

Managed Client Services

Interaction Services

- Value
  - ‣ Application portability across clients
  - ‣ Service-oriented platform
  - ‣ Extensible component model
  - ‣ Core services

- Standards
  - ‣ J2SE
  - ‣ Eclipse
  - ‣ OSGi

ICU4J

Eclipse Logging

Eclipse Preferences

OSGi User Admin Service

OSGi Log Service

OSGi Configuration Admin

Log Redirector

OSGi MetaType Service

OSGi Device Access

Core Eclipse

Core Extension Point Framework

OSGi Service I/F definitions

OSGi Utilities

OSGi Service Framework

J2SE 1.4.2 Class Libraries

IBM J2SE 1.4.2 JVM SR2

11
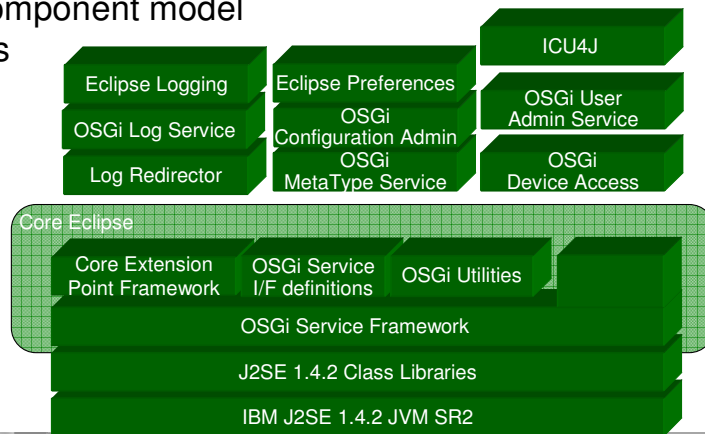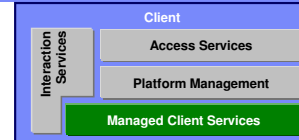
Managed Client Services enable multiple applications and services to run on the same JVM, support life cycle management of these applications and services, and provide application portability across Windows and Linux operating systems. The client provides an extensible component model through a service-oriented platform that enables you to add your own applications and services. The client platform provides additional services to assist you in the development of your applications including logging, preferences, globalization, and more.

Major Components

- IBM J2SE 1.4.2 JVM (service release 2)
- OSGi Service Framework
  - Allows simultaneous execution of applications and services on a single JVM instance
  - Enables applications to share services and packages
  - Separates service interface from service implementation
  - Supports independent life-cycle management
- Eclipse 3.0.2
  - Provides core Eclipse Framework to support plug-ins
  - Runs on the OSGi Service Framework
- Logging
  - Supports OSGi log service, Eclipse logging, and J2SE 1.4 logging
  - Collects all logged messages into a persistent log file
- Preferences
  - Allows applications to persistently store and process notifications for preferences
  - Enables System Administrators to remotely query and update preferences
- International Components for Unicode for Java (ICU4J)
  - Provides Java classes for Unicode and I18N to globalize applications

Client

Interaction Services

Access Services

Platform Management

Managed Client Services

The client platform ships IBM's Java 2 Standard Edition **(J2SE) 1.4.2 JVM** (service release 2) as the base Java runtime environment.

The client platform provides an **OSGi Service Framework** that implements the OSGi R3 framework specification and provides a service-oriented architecture on top of the JVM. The OSGI Service Framework enables multiple applications and services to run on the same JVM and supports life cycle management of these applications and services. OSGi Services provide the interface definitions for standard services defined by the OSGi R3 specification and OSGi Utilities provides an implementation of the OSGi utilities interface.

The **Eclipse** Core Extension Point Framework supports the Eclipse plug-in component model and, by running with the OSGi Service Framework, enables plug-ins to receive the corresponding benefits of OSGi. The client platform is built on Eclipse 3.0.2.
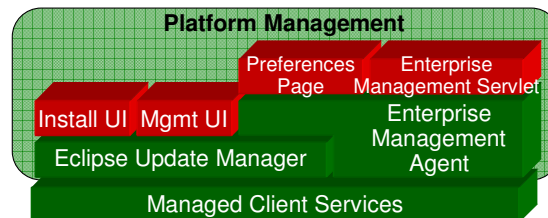
The client platform enhances **Logging** through a plug-in, called the logRedirector, that collects all messages logged in the client platform into one persistent log file regardless of whether you use the OSGi log service, Eclipse logging, or J2SE 1.4 logging.

You can use **Preferences** to persistently store your applications' preferences, allow your applications to process notifications of preferences changes, and to enable System Administrators to remotely query and update these configuration values.

International Components for Unicode for Java (ICU4J) is a set of Java classes that extend the capabilities provided by the J2SE class libraries in the areas of Unicode and internationalization support so you can globalize your applications.

IBM

Platform Management

Client

- Value
  - ▸ Enable "On Demand" management of applications

- Deployment options
  - ▸ Standalone - Eclipse Update Manager
  - ▸ Enterprise - Enterprise Management Agent and DMS

- Standards
  - ▸ Eclipse
  - ▸ OSGi
  - ▸ Open Mobile Alliance

Platform Management

Preferences Page

Enterprise Management Servlet

Install UI Mgmt UI

Enterprise Management Agent

Eclipse Update Manager

Managed Client Services

13
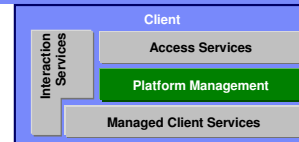
Platform Management installs, maintains, and configures applications and services on the client. End-users can directly deploy applications onto their Workbench ("standalone deployment") or a System Administrator may remotely deploy and configure applications on one or more clients from a central administrative console in the Enterprise ("Enterprise deployment").

Platform Management
Major Components

- Update Manager ("Standalone Deployment")
  - End-users can directly install applications onto their Workbench
  - Supports the Eclipse standard
- Install User Interface
  - Allows end-users to install applications onto their client, locally from their Workbench
- Application Management User Interface
  - Enables end-users to manage the current configuration of their client platform from their Workbench
- Enterprise Management Agent ("Enterprise Deployment")
  - Enables a System Administrator to remotely manage the client platform
  - Supports the Open Mobile Alliance (OMA) protocol with DMS
  - Supports the OSGi standard to locally manage client components
- Enterprise Management Agent Servlet
  - Allows a System Administrator to view and control the Agent via a browser
- Preferences Pages
  - Allows an end-user to locally configure Agent from their Workbench

Client

Interaction Services

Access Services

Platform Management

Managed Client Services

The **Update Manager** provides the capability to install applications and components from standard Eclipse update sites onto the Workbench.

The **Install user interface** enables end-users to use the Update Manager so they can install applications and components.

The **Application Management user interface** also runs with the Update Manager and enables end-users to manage the current configuration of the client platform, including applications that are installed on the client, from the Workbench.

The **Enterprise Management Agent** enables a System Administrator to manage the client platform remotely by using a Device Management Server, which is provided by the server platform.

The **Enterprise Management Agent Servlet** allows you to view and modify the configuration and control the features of the Enterprise Management Agent. You can launch it from within the IBM WebSphere Everyplace Deployment for Windows and Linux Version 6 workbench or from within a web browser.

You may also configure the Enterprise Management Agent after you install the client platform by using the **Preferences Page** from the Workbench.

Access Services

- Value
  - ▸ Extend backend programming model "out" to devices
- Services
  - ▸ Web Container
  - ▸ Transaction Container
  - ▸ JNDI
  - ▸ Messaging
  - ▸ Database
  - ▸ Web Services
  - ▸ Data Sync Framework
- Standards
  - ▸ J2EE subset, J2SE, J2ME
  - ▸ W3C
  - ▸ Web Services
  - ▸ OMA
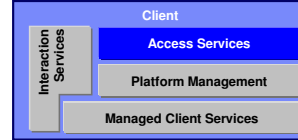
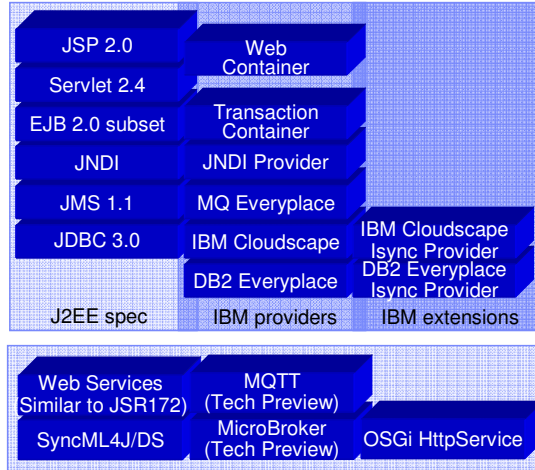**Client**

Interaction Services

| Access Services |
| Platform Management |
| Managed Client Services |

JSP 2.0	Web Container	
Servlet 2.4	Transaction Container	
EJB 2.0 subset		
JNDI	JNDI Provider	IBM Cloudscape Isync Provider
JMS 1.1	MQ Everyplace	DB2 Everyplace Isync Provider
JDBC 3.0	IBM Cloudscape	
	DB2 Everyplace	
J2EE spec	IBM providers	IBM extensions

| Web Services Similar to JSR172) | MQTT (Tech Preview) | |
| SyncML4J/DS | MicroBroker (Tech Preview) | OSGi HttpService |

15

Access Services provide a familiar programming model for J2EE developers so they can reuse their skills and components to develop applications that run on clients. Additionally, Access Services enable client applications to support offline operations. Access Services also enable you to move key components of your application to the client platform through the use of standard APIs.

Access Services
J2EE Affinity Components

- Web Container
  - Servlet 2.4 / JSP 2.0
  - Servlet 2.3 / JSP 1.2
  - JSF, JSTL, and Struts
- Embedded Transaction Container
  - EJB 2.0 subset
- Database
  - JDBC 3.0
  - DB2e or Cloudscape
  - Supports offline operation via sync with Enterprise databases
- Point-to-Point Messaging
  - JMS 1.1
  - WebSphere MQe
  - Supports offline operation through local queuing of messages
  - Provides bridge to WebSphere MQ
- JNDI

J2EE spec	IBM providers	IBM extensions
JSP 2.0	Web Container	
Servlet 2.4	Transaction Container	
EJB 2.0 subset		
JNDI	JNDI Provider	
JMS 1.1	MQ Everyplace	IBM Cloudscape Isync Provider
JDBC 3.0	IBM Cloudscape	DB2 Everyplace Isync Provider
	DB2 Everyplace	

16
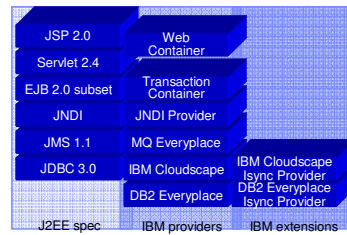
Here are the specific Access Services that support affinity with the J2EE / WebSphere programming model.

The client platform provides an embedded **Web container** from WAS to run Web applications that consist of JSP's and servlets. The Web container also supports applications that use Java Server Faces (JSF), JSP Standard Tag Library (JSTL), and Struts. As a result, the Web container enables you to move your Web applications from the server to clients to preserve the existing browser user interface, leverage your existing Web components, and provide a richer user experience through support of local and offline operations.

The client platform also provides an **embedded Transaction Container** to run Embedded Transaction Applications that conform to a subset of the EJB 2.0 specification, which includes support for stateless session beans and entity beans. This container enables you to move your business logic from the server to clients so you can leverage your existing beans to make business logic available to client applications, and support local and offline operations.

There are two key services that support local and offline operations.
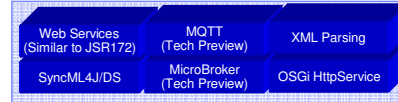
First, you can use the **JDBC 3.0** API with DB2 Everyplace or IBM® Cloudscape™ as a local SQL database when more advanced data manipulations are required than can be supported by placing data in a local file store. These databases can periodically synchronize with Enterprise databases to capture data on the client for use by the client application when the user is offline. These databases can also protect local data through data encryption.

Second, you can also use the Java Message Service (**JMS**) 1.1 API with WebSphere MQ Everyplace (MQe) to send and receive messages. MQe provides once-only, assured messaging and supports offline operations with local message queues that hold messages when the device is offline and then sends these queued messages to Enterprise applications when the device is back online. Similarly, messages destined for client applications are held in server-side message queues and then sent to the client applications when the device is back online. MQe encrypts messages to protect content over the network. As a result, the client platform enables your users to conduct secure e-business transactions.

WebSphere MQe provides a bridge to WebSphere MQ to connect your client applications to the Enterprise Service Bus.

JNDI provides a name space so applications can utilize named objects.

Access Services
Additional Components

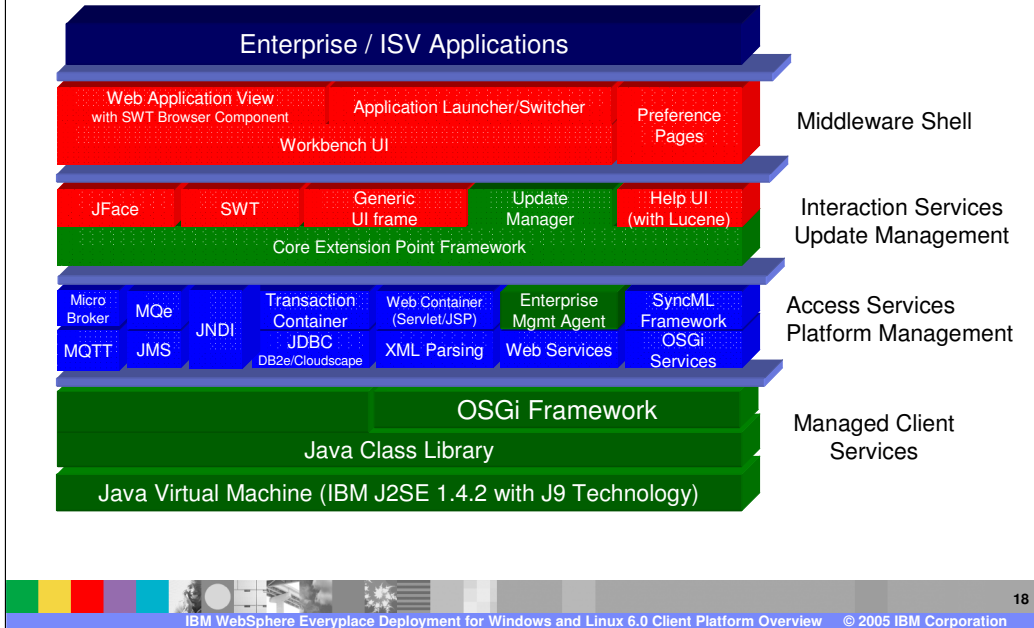|		
Web Services (Similar to JSR172)	MQTT (Tech Preview)	XML Parsing
SyncML4J/DS	MicroBroker (Tech Preview)	OSGi HttpService

- **Mobile Web Services**
  - ▸ Web Services client (similar to JSR 172)
  - ▸ Web Services provider (host Web Services)
  - ▸ WS-Security
  - ▸ Enables support of additional data types

- **Publish and Subscribe Messaging**
  - ▸ MQ Telemetry Transport (MQTT) (Tech Preview)
  - ▸ Microbroker (Tech Preview)
  - ▸ Provides bridge to WebSphere MQ

- **SyncML4J Framework**
  - ▸ Open Mobile Alliance SyncML specification

For online operations, the client platform supports **Mobile Web Services** so client applications can consume and provide Web Services in a secure manner. As a result, your users have access to a broad range of business data and consumer information.

The client platform also supports publish and subscribe messaging through a technical preview of the **MQ Telemetry Transport** and the **MicroBroker**. The Microbroker also provides a bridge to WebSphere MQ to connect your client applications to the Enterprise Service Bus.

The **SyncML4J** (SyncML for Java) toolkit enables you to develop data synchronization client applications.

Complete Client Stack

Enterprise / ISV Applications			

Web Application View with SWT Browser Component	Application Launcher/Switcher	Preference Pages
	Workbench UI	

Middleware Shell

JFace	SWT	Generic UI frame	Update Manager	Help UI (with Lucene)
		Core Extension Point Framework		

Interaction Services
Update Management

Micro Broker	MQe	JNDI	Transaction Container	Web Container (Servlet/JSP)	Enterprise Mgmt Agent	SyncML Framework
MQTT	JMS		JDBC DB2e/Cloudscape	XML Parsing	Web Services	OSGi Services

Access Services
Platform Management

OSGi Framework
Java Class Library
Java Virtual Machine (IBM J2SE 1.4.2 with J9 Technology)

Managed Client
Services

WebSphere Everyplace Deployment for Windows and Linux client as shown in this picture is a pre-integrated, pre-tested set of components that are available as a Licensed Program Product (LPP). However, you can also build a customized client platform that consists of a selected set of components.

For more information…

- For more information on any of the Client Services, please select from the following topics under the <u>Client platform</u> task:
  - ▶ Managed Client Services
  - ▶ Platform Management
  - ▶ Access Services
  - ▶ Interaction Services

For more information on any of the Client Services, please select from the following topics under the <u>Client platform</u> task:

•Managed Client Services

•Platform Management

•Access Services

•Interaction Services

Template Revision: 9/28/2005 5:50 PM

Trademarks, Copyrights, and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel 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.

Linux is a registered trademark of Linus Torvalds.

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

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

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

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

Page 20 of 12