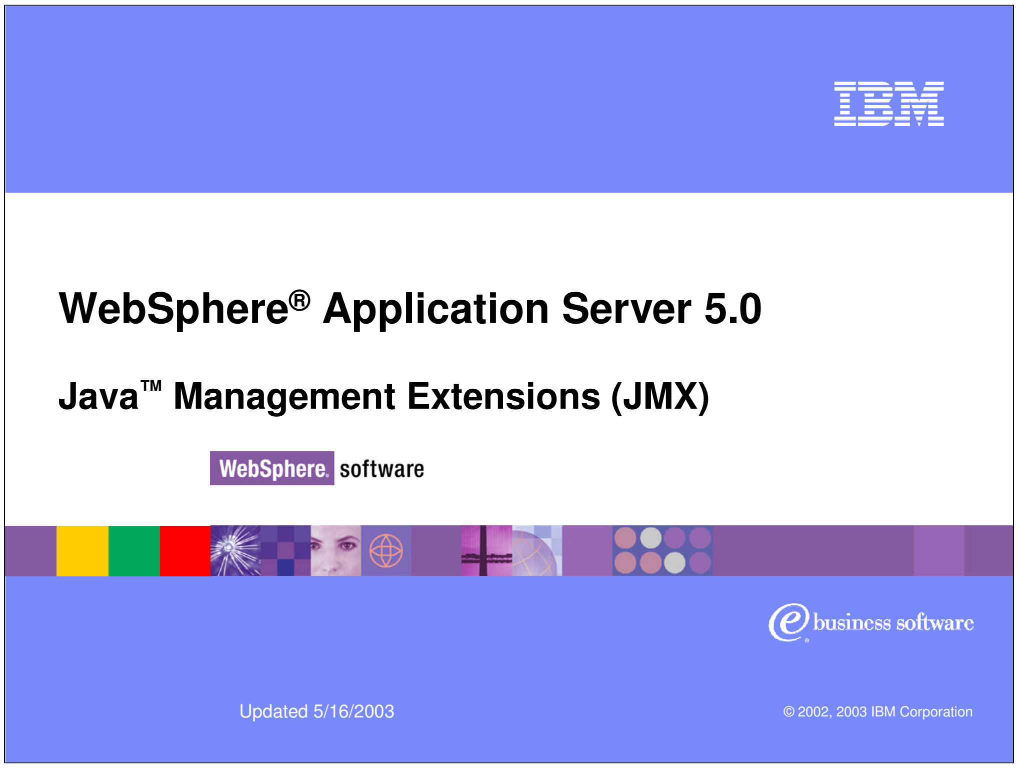


WebSphere® Application Server 5.0

Java™ Management Extensions (JMX)

WebSphere software

e business software

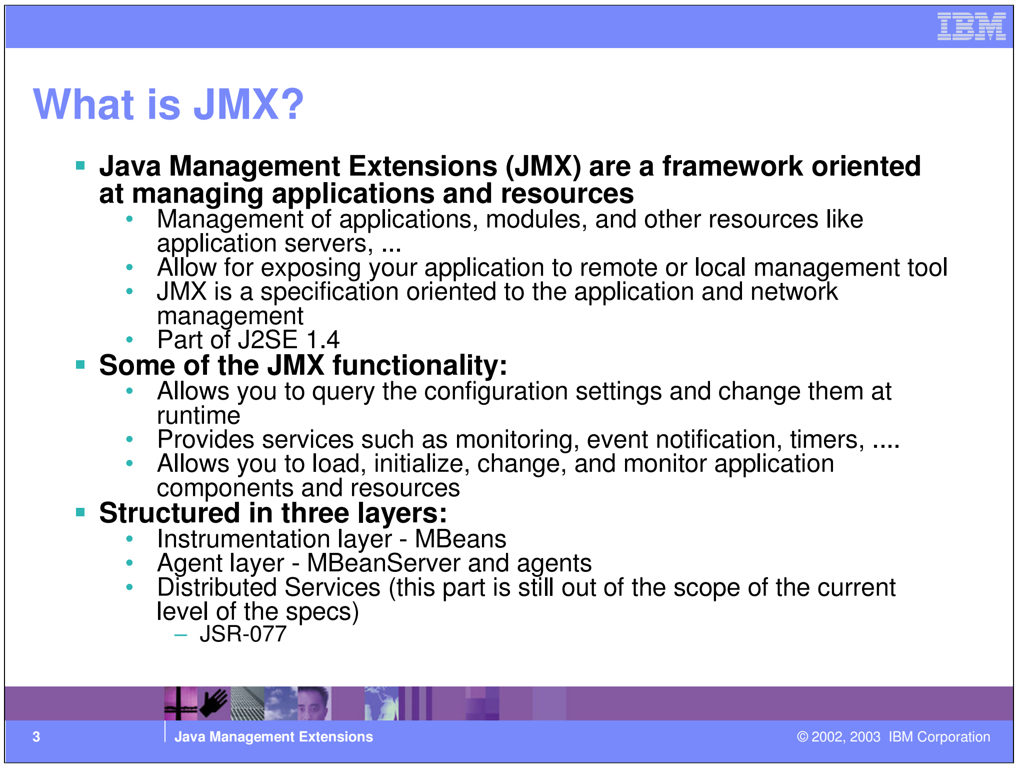
Updated 5/16/2003

© 2002, 2003 IBM Corporation

Objectives

- **Understand what the Java Management Extensions (JMX) are all about**
  - Instrumentation layer
  - Agent layer
  - Management layer
- **Describe WebSphere 5.0 JMX Support**
  - Moving from a proprietary to a standard administrative implementation
  - Usage scenarios
  - Internal product usage
  - External programmatic administration
  - External scripting administration via Bean Scripting Framework (BSF)

© 2002, 2003 IBM Corporation



What is JMX?

- **Java Management Extensions (JMX) are a framework oriented at managing applications and resources**
  - Management of applications, modules, and other resources like application servers, ...
  - Allow for exposing your application to remote or local management tool
  - JMX is a specification oriented to the application and network management
  - Part of J2SE 1.4
- **Some of the JMX functionality:**
  - Allows you to query the configuration settings and change them at runtime
  - Provides services such as monitoring, event notification, timers, ...
  - Allows you to load, initialize, change, and monitor application components and resources
- **Structured in three layers:**
  - Instrumentation layer - MBeans
  - Agent layer - MBeanServer and agents
  - Distributed Services (this part is still out of the scope of the current level of the specs)
    - JSR-077

3 Java Management Extensions © 2002, 2003 IBM Corporation

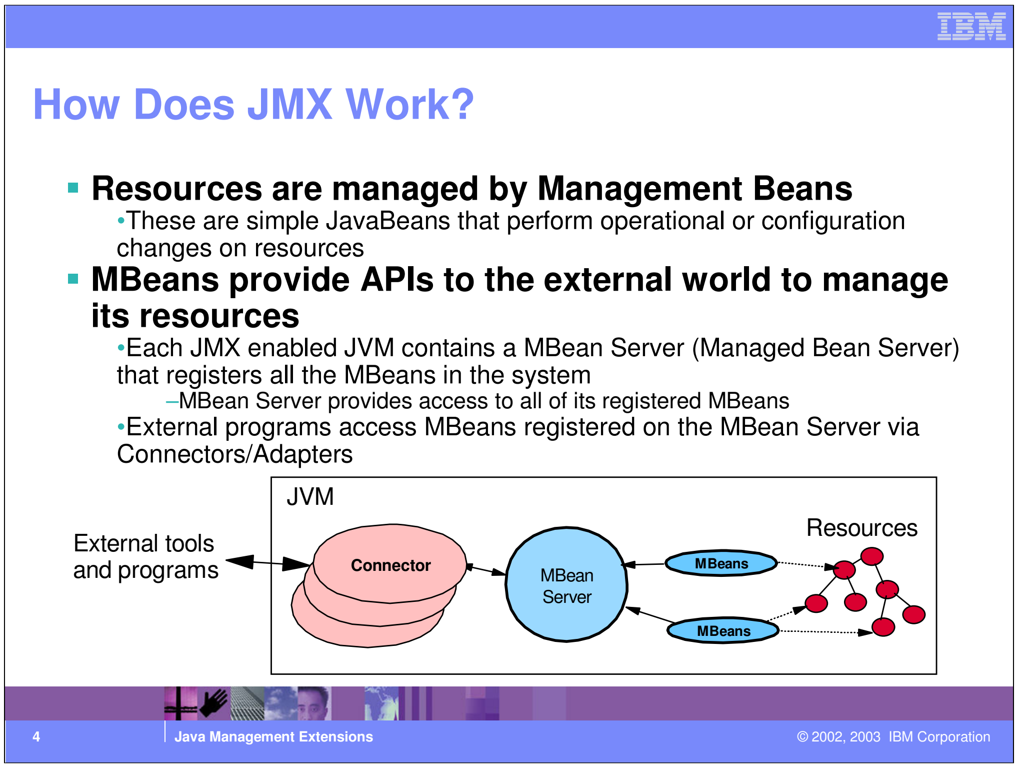
•The Java Management Extensions are a framework that provides a standard way of exposing Java resources to a systems management infrastructure.

•By Java resources we mean a wide range of objects, like Enterprise Applications, Web Modules, but also Application Servers, Clusters, and so on.

•The framework allows a provider to implement functions such as listing the configuration settings, and allowing users to edit them - it also includes a notification layer, that can be used by the management applications to monitor events such as the startup of an application server.

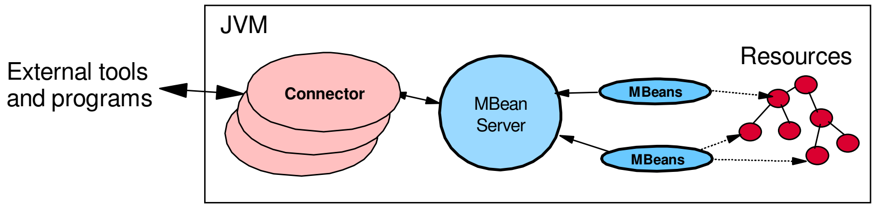
•There are three fundamental layers in JMX

> •The Instrumentation layer dictates how resources can be wrapped within special JavaBeans, called Management Beans or MBeans
>
> •The Agent layer, which implements the monitoring and event notification
>
> •The Distributed Services topmost layer - which is meant to define how external management applications can interact with the underlying layers, in terms of protocols, APIs, and so on. This last layer is yet under definition and it is the subject of the JSR-077 which WebSphere already provides as an anticipation of future levels of this body.



How Does JMX Work?

- **Resources are managed by Management Beans**
  - These are simple JavaBeans that perform operational or configuration changes on resources
- **MBeans provide APIs to the external world to manage its resources**
  - Each JMX enabled JVM contains a MBean Server (Managed Bean Server) that registers all the MBeans in the system
    - MBean Server provides access to all of its registered MBeans
  - External programs access MBeans registered on the MBean Server via Connectors/Adapters



•We mentioned in the previous chart that the central element of JMX is represented by MBeans (Management Beans) - they are NOT EJBs - they are simple JavaBeans that need to conform to certain design patterns outlined in the JMX specifications.

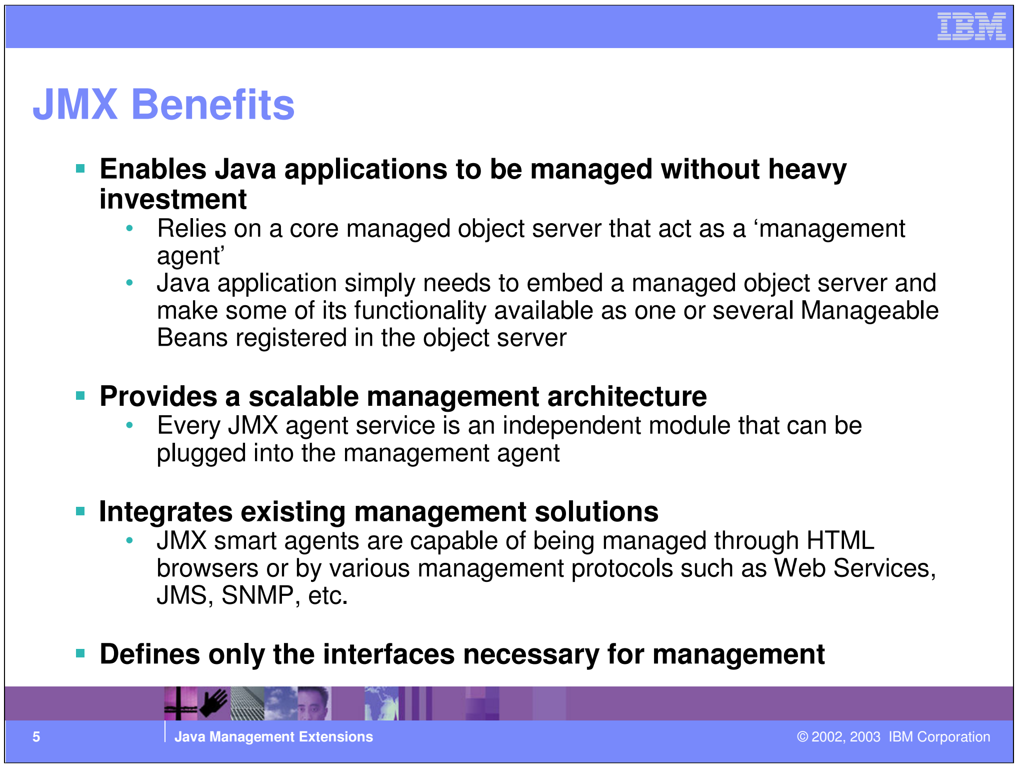
•Providers that need to instrument their systems with JMX need to provide a series of MBeans - each MBean is meant to wrap (or represent) a certain resource. For instance, in order to expose an Application Server as a manageable resource, WebSphere needs to provide an Application Server MBean.

•The systems management applications can interact with these MBeans through connectors and adapters.

•Connectors are used to connect an agent with a remote JMX-enabled management application - they are oriented to the transport mechanism. For instance, a certain provider may make available an RMI connector that allows Java applications to interact remotely with the MBeans.

•Protocol adapters provide a management view of the JMX agent through a given protocol

> •Management applications that connect to a protocol adapter are usually specific to the given protocol
>
> •For instance - a provider may want to provide an SNMP adapter that allows systems management facilities that use SNMP to interact with JMX.

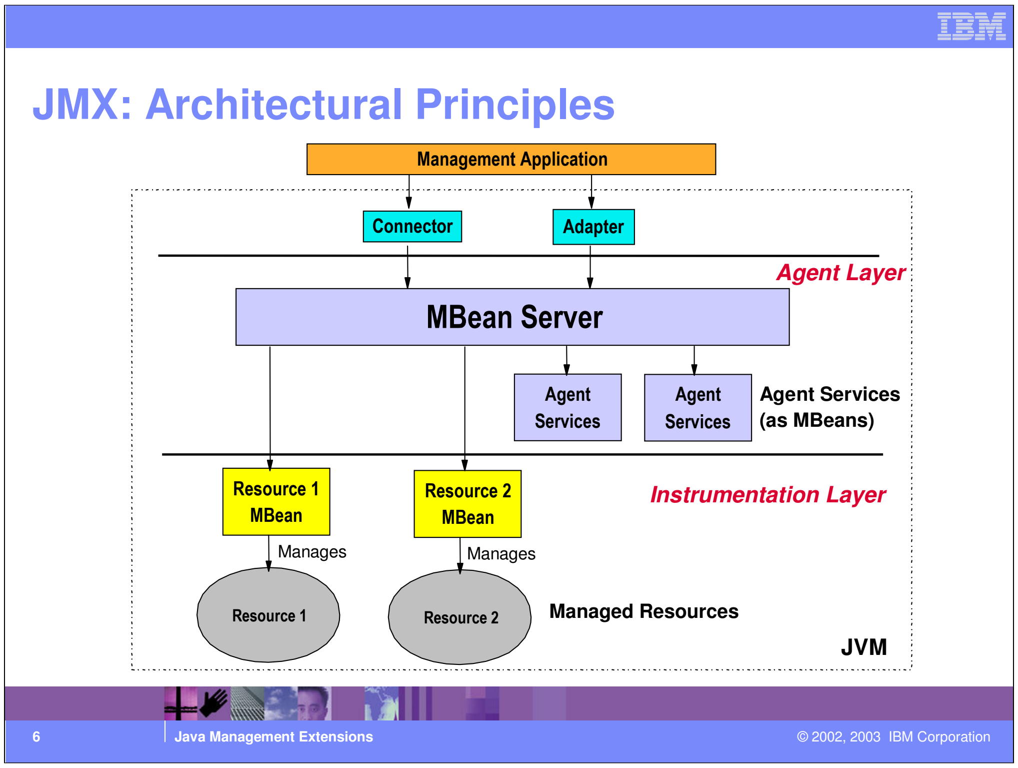


JMX Benefits

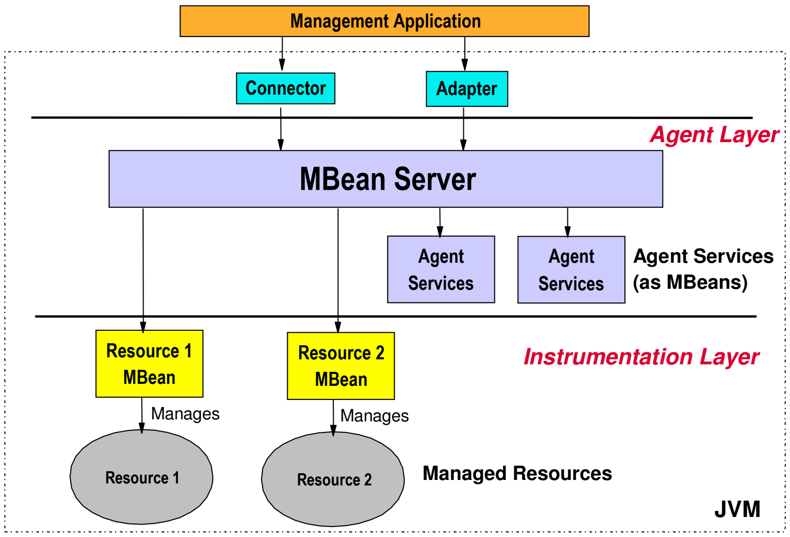
- **Enables Java applications to be managed without heavy investment**
  - Relies on a core managed object server that act as a ‘management agent’
  - Java application simply needs to embed a managed object server and make some of its functionality available as one or several Manageable Beans registered in the object server
- **Provides a scalable management architecture**
  - Every JMX agent service is an independent module that can be plugged into the management agent
- **Integrates existing management solutions**
  - JMX smart agents are capable of being managed through HTML browsers or by various management protocols such as Web Services, JMS, SNMP, etc.
- **Defines only the interfaces necessary for management**

5 Java Management Extensions © 2002, 2003 IBM Corporation

The benefits of JMX primarily reside in the standardization and in the isolation of roles.



JMX: Architectural Principles



JMX is structured in three layers:

Instrumentation layer - MBeans

Agent layer - MBeanServer and agents

Distributed Services (this part is still out of the scope of the current level of the specs) JSR-077

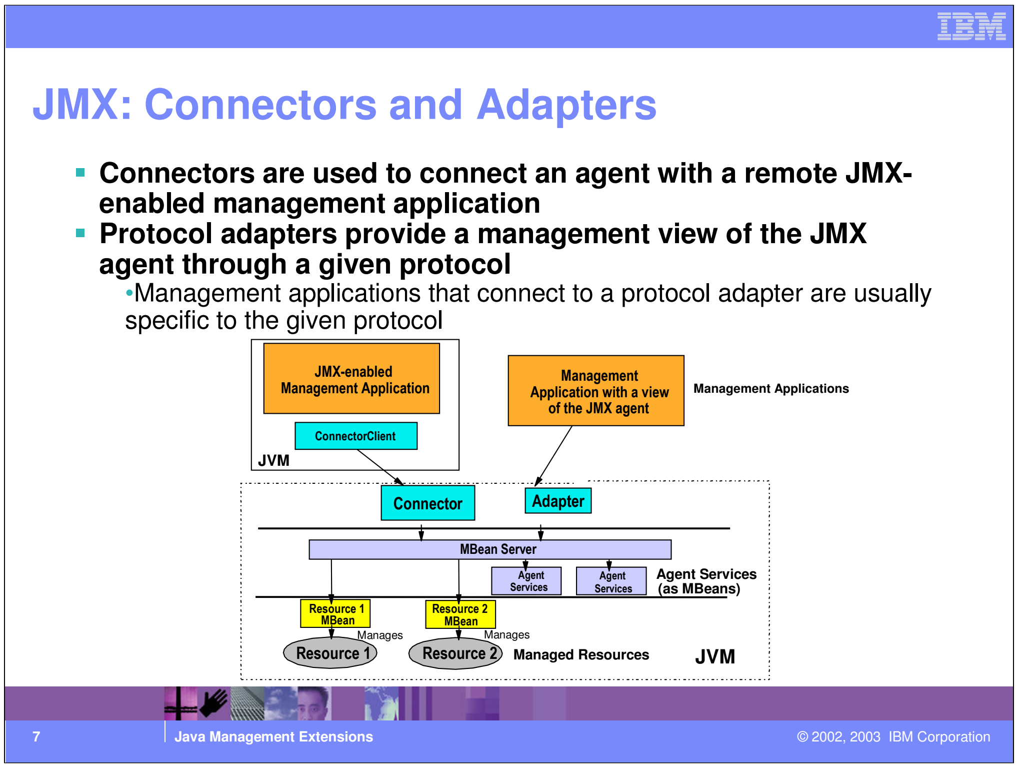
Difference between Connector and Adapter

Connector is a reflection of the MBean API to the outside world

Example: RMI, SOAP

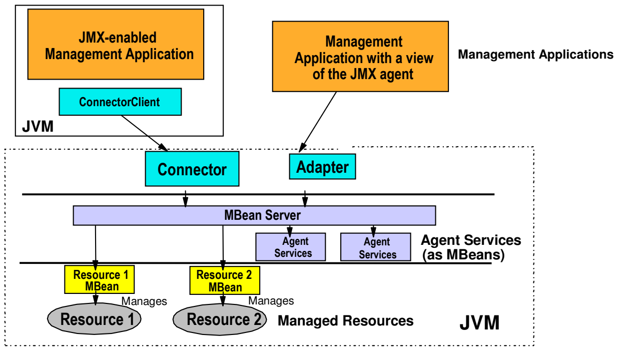
Adapter coverts a different protocol into appropriate MBean API call

Example: SNMP, HTTP



JMX: Connectors and Adapters

- Connectors are used to connect an agent with a remote JMX-enabled management application
- Protocol adapters provide a management view of the JMX agent through a given protocol
  - •Management applications that connect to a protocol adapter are usually specific to the given protocol



•The MBean server relies on protocol adapters and connectors (implemented as MBeans themselves) to make the agent accessible from management applications outside the agent's JVM

•Adapters and connectors make all MBean server operations to be available to a remote management application

•For an agent to be managed, it must include at least one protocol adapter or connector

•An agent can include any number of these, allowing it to be managed by multiple managers, through different protocols

•Connectors are used to connect an agent with a management application developed using the JMX distributed services

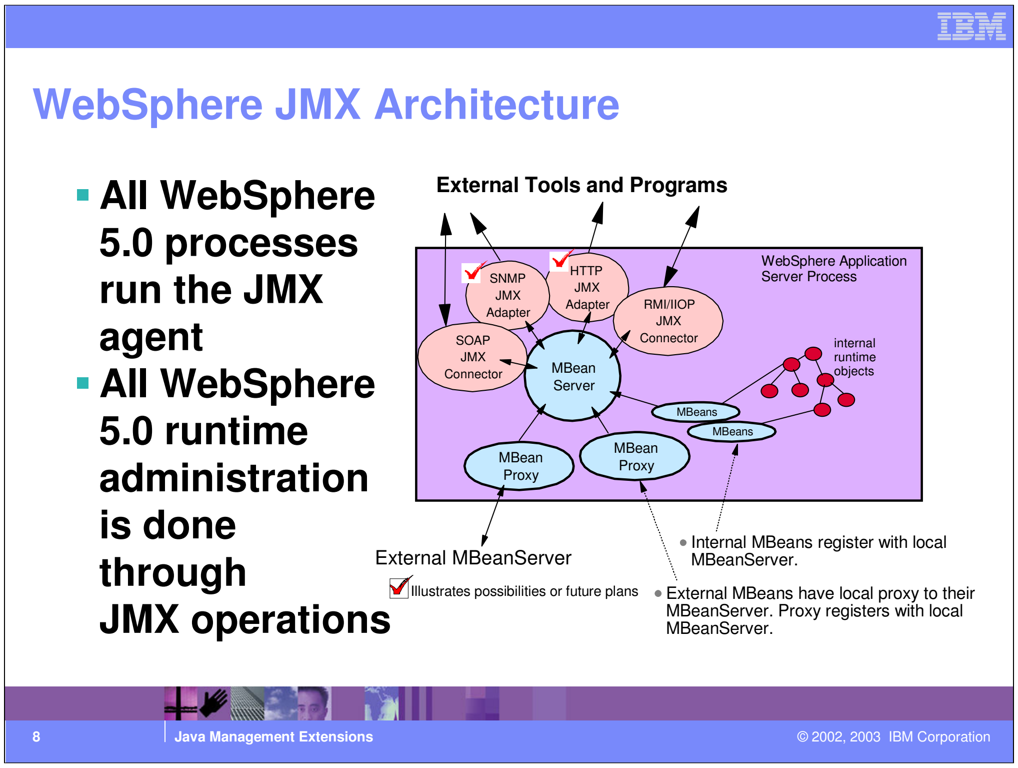
•This kind of communication involves a connector in the agent and a connector client in the management application

•Protocol adapters provide a management view of the JMX agent through a given protocol

•They adapt the operations of MBeans and the MBean server into a representation in the given protocol, and possibly into a different information model, for example SNMP

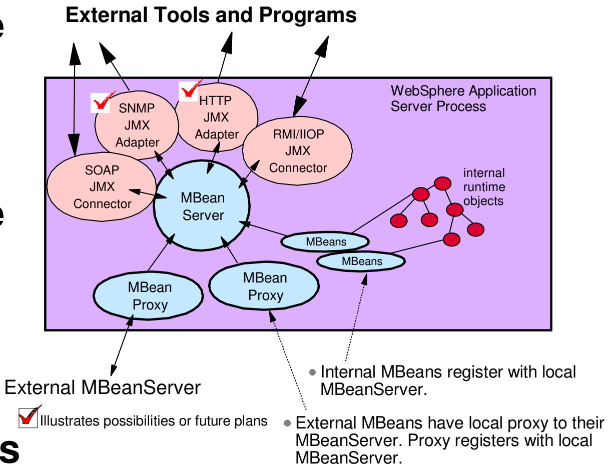
•Adapters and connectors provided by a JMX implementation should be implemented as MBeans

•Both connector servers and protocol adapters use the services of the MBean server in order to apply the management operation they receive to the MBeans, and in order to forward notifications to the management application.



WebSphere JMX Architecture

- All WebSphere 5.0 processes run the JMX agent
- All WebSphere 5.0 runtime administration is done through JMX operations



- Internal MBeans register with local MBeanServer.
- External MBeans have local proxy to their MBeanServer. Proxy registers with local MBeanServer.

8 | Java Management Extensions | © 2002, 2003 IBM Corporation

•Connectors are used to connect an agent with a remote JMX-enabled management application

•Protocol adapters provide a management view of the JMX agent through a given protocol

  •Management applications that connect to a protocol adapter are usually specific to the given protocol

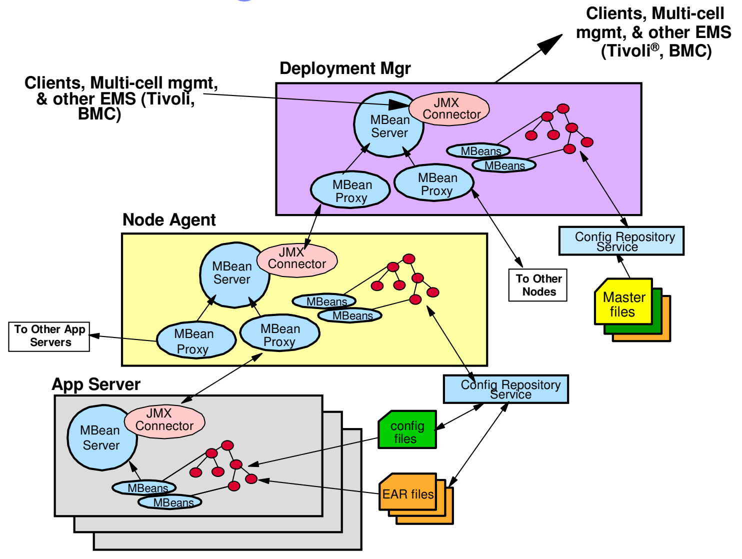
•MBean Proxy let the MBean Server pass the message/request to an external MBean Server (another Server, NodeAgent, Cell manager)

  •Node Agent has MBean Proxy for all Servers within its Node

    •They don't have MBean Proxies for other Node Agents

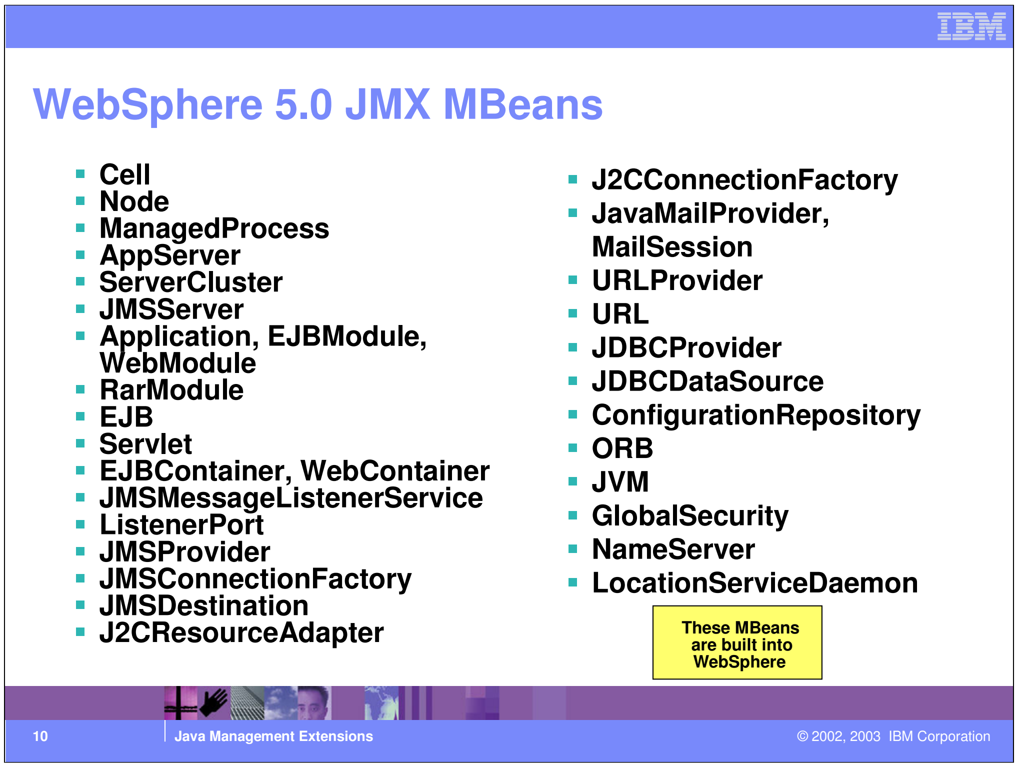
•Cell Managers have MBean Proxies for all the NodeAgent

WebSphere JMX Big Picture



•One of the great advantages of JMX is that each process is self-sufficient when it comes to managing the resources that are involved with it. There is no single and central point of control - in principle, you could connect a systems management client to any managed process and interact with the MBeans that "live" there.

•However - JMX still allows for a single, flat, "big picture" approach to systems management. Separate processes interact through MBean proxies allowing a single management client to seamlessly navigate through the network of managed processes.



WebSphere 5.0 JMX MBeans

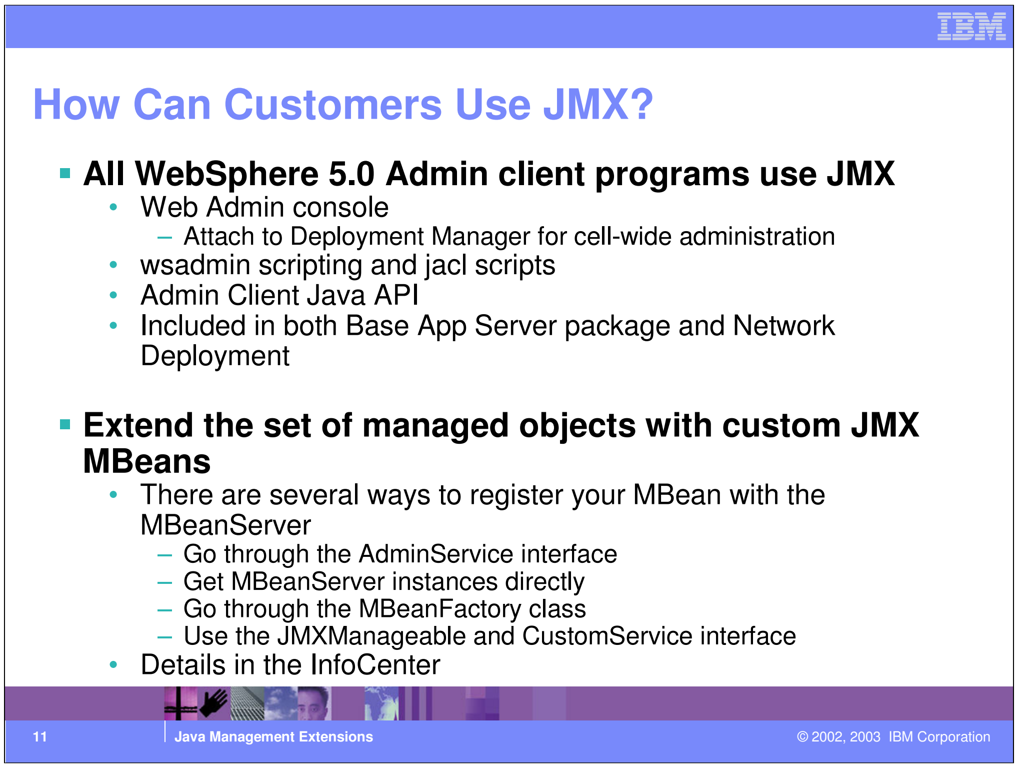
- Cell
- Node
- ManagedProcess
- AppServer
- ServerCluster
- JMSServer
- Application, EJBModule, WebModule
- RarModule
- EJB
- Servlet
- EJBContainer, WebContainer
- JMSMessageListenerService
- ListenerPort
- JMSProvider
- JMSConnectionFactory
- JMSDestination
- J2CResourceAdapter
- J2CConnectionFactory
- JavaMailProvider, MailSession
- URLProvider
- URL
- JDBCProvider
- JDBCDataSource
- ConfigurationRepository
- ORB
- JVM
- GlobalSecurity
- NameServer
- LocationServiceDaemon

These MBeans are built into WebSphere

10 Java Management Extensions © 2002, 2003 IBM Corporation

These are all the MBeans provided in WebSphere 5.0.

Each MBean may have different functions. For instance, an Application Server may provide functions such as start and stop - an Enterprise Application may expose functions such as install or uninstall.



How Can Customers Use JMX?

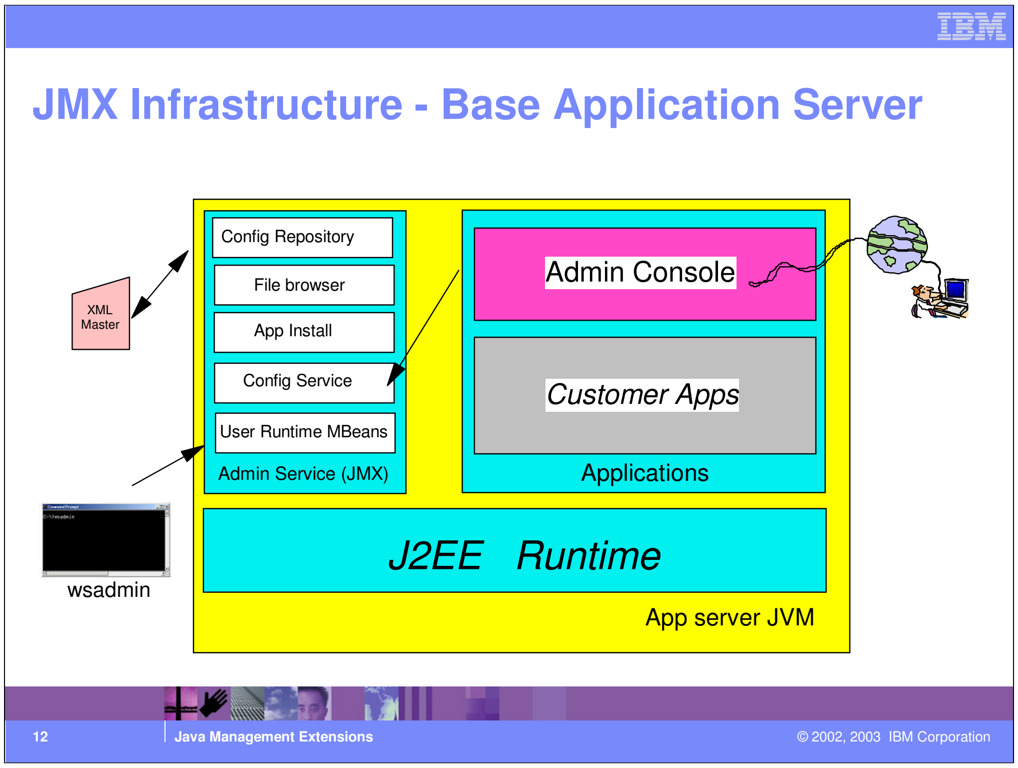
- **All WebSphere 5.0 Admin client programs use JMX**
  - Web Admin console
    - Attach to Deployment Manager for cell-wide administration
  - wsadmin scripting and jacl scripts
  - Admin Client Java API
  - Included in both Base App Server package and Network Deployment
- **Extend the set of managed objects with custom JMX MBeans**
  - There are several ways to register your MBean with the MBeanServer
    - Go through the AdminService interface
    - Get MBeanServer instances directly
    - Go through the MBeanFactory class
    - Use the JMXManageable and CustomService interface
  - Details in the InfoCenter

11 Java Management Extensions © 2002, 2003 IBM Corporation

•In general, WebSphere customers will experience very limited exposure to JMX. They will interact with it through the admin clients provided by WebSphere (the console, wsadmin, and so on).

•Conceivably, customers (or software vendors) may create their own Java-based, management infrastructure based on JMX and use it to administer WebSphere resources.

•Or- they could provide additional MBeans - for instance, to instrument application subsystems.



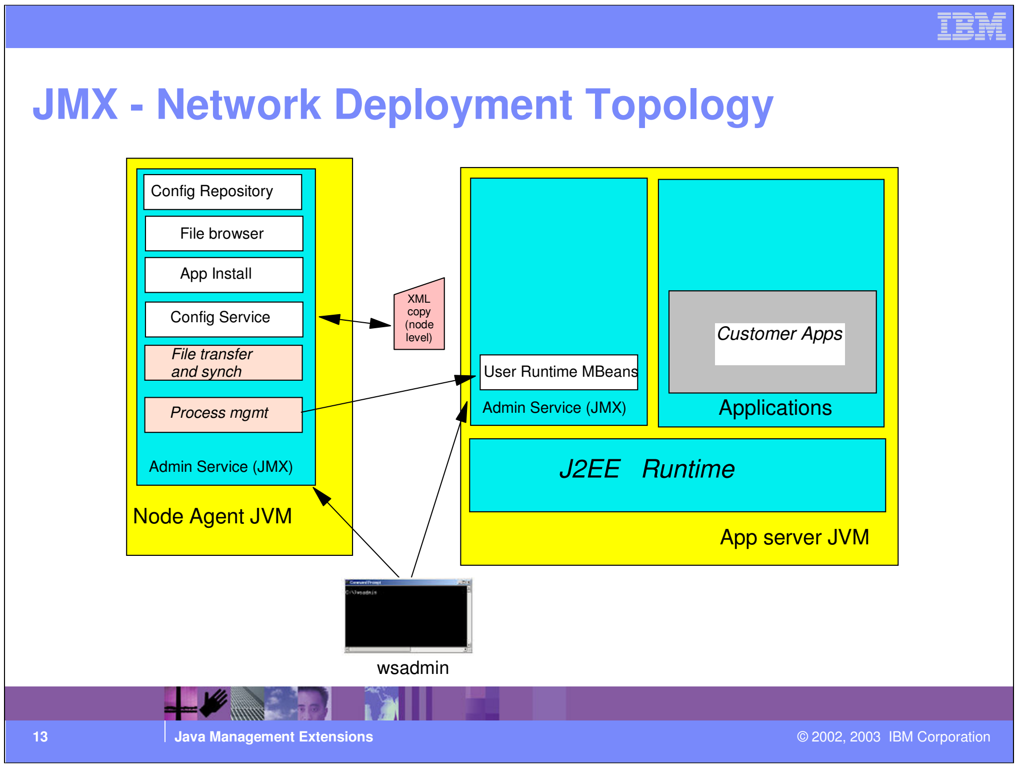
JMX Infrastructure - Base Application Server

•This chart illustrates the JMX components that are available in the base application server. The base application server implements the complete J2EE runtime and supports J2EE customer applications. The Admin Console, which is a "system" application, is also installed on the base app server. You need to take care not to accidentally remove the admin console - since WebSphere doesn't differentiate between "user" applications and "system" applications that may be installed on the application server.

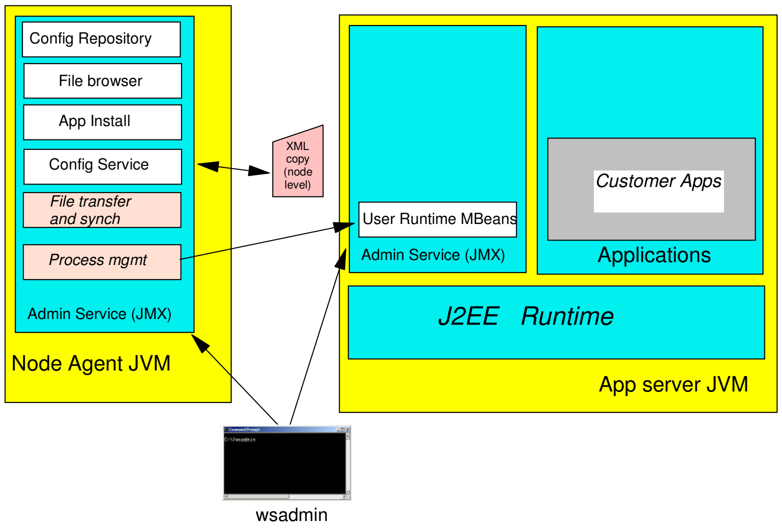
•The Base App Server also includes the "administrative service" implemented through JMX. This component supports a series of MBeans, including the MBeans for configuration management, that allow you to modify the contents of the master repository.

•It also includes the application install MBean, and the user runtime MBeans that allow the administration of applications and resources that you may have configured in WebSphere.

•The console and the wsadmin scripting interface can utilize this JMX layer.



JMX - Network Deployment Topology



13 | Java Management Extensions | © 2002, 2003 IBM Corporation

•When Network Deployment is installed and a node is added to the cell, there are changes that occur on that node.

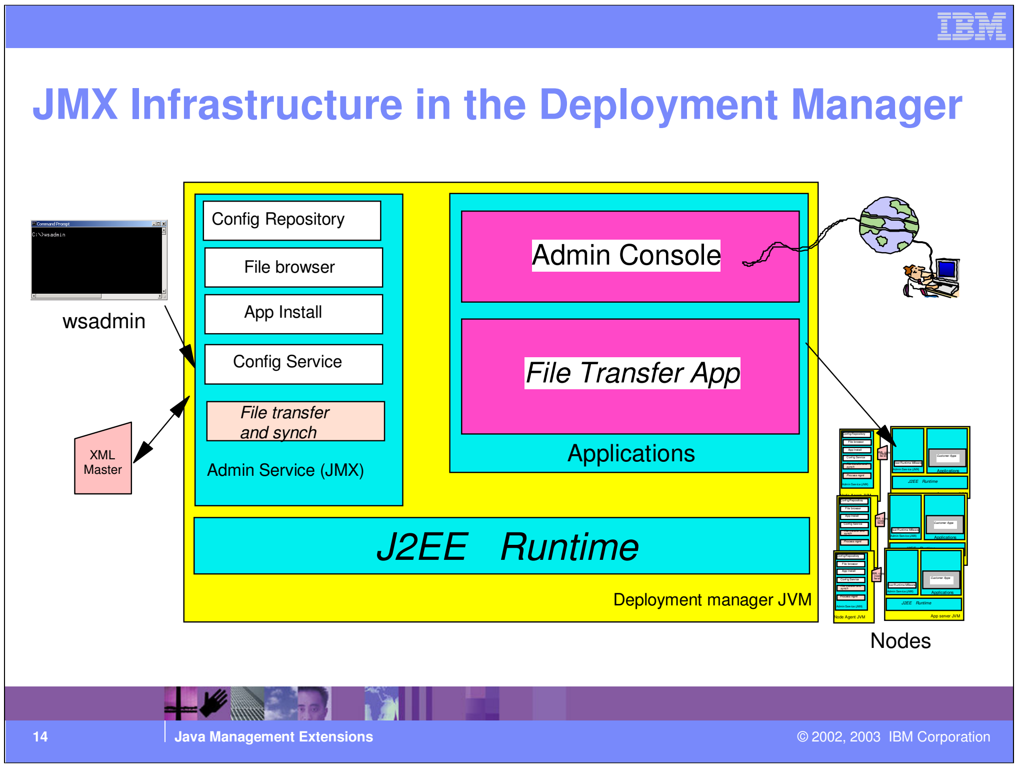
•The admin console application is uninstalled from the application server JVM and the MBeans for configuration management are also removed. As a consequence, you won't be able to connect the admin console directly to an app server that is part of an ND cell.

•wsadmin can be connected directly to an application server, but you'll only be able to perform administration of "live" resources (in other you won't be able to change the configuration files).

•The Node Agent is a thinner JVM that doesn't support the full J2EE programming model. It does support the JMX admin service, where the MBeans for config management are installed. The MBeans for file transfer and synchronization are also available, allowing the node agent to keep the node-level repository in synch with the master at the cell level.

•The process management MBeans allow the node agent to restart app servers in case they unexpectedly fail. Notice that no "user runtime MBeans" are configured in the Node Agent - meaning that no user-defined resources and applications can be installed there.

•You can connect wsadmin (but not the admin console) to the node agent and will be able to make config changes through it - however, this is a practice that is discouraged, because local changes will be wiped away at the next synchronization.



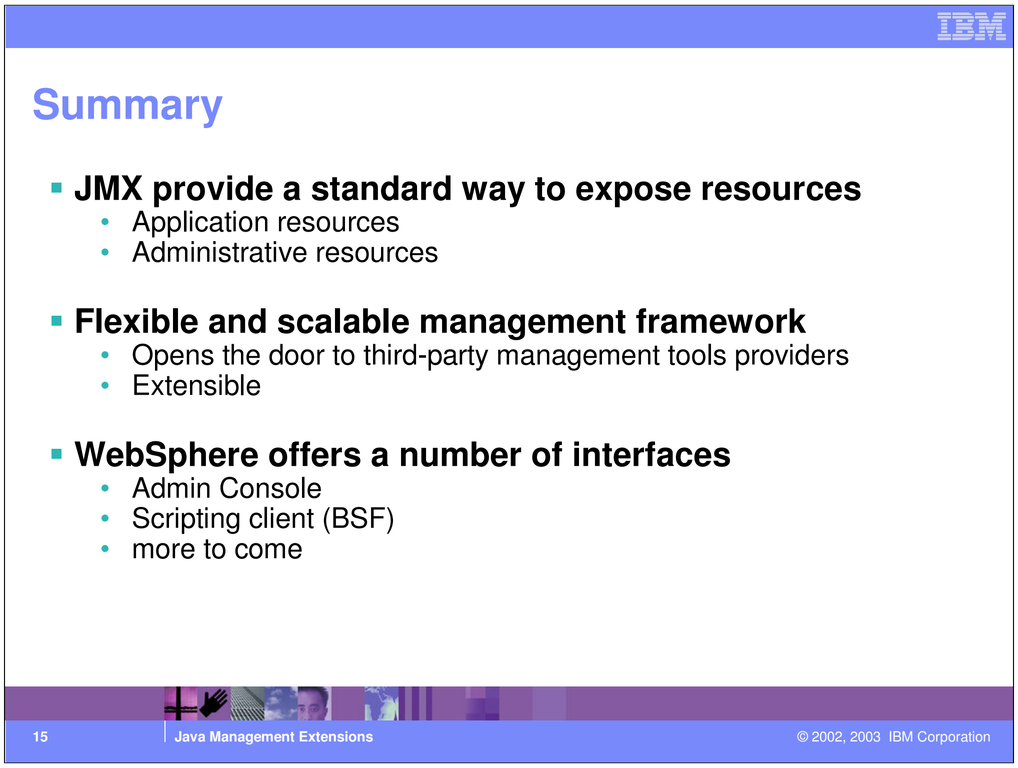
JMX Infrastructure in the Deployment Manager

•This chart describes the functional contents of the Deployment Manager server.

•The Deployment Manager is significantly "thicker" than the node agent. It supports the J2EE runtime - in fact, two J2EE applications are installed on the deployment manager - the admin console and the file transfer application. The latter makes it possible to move xml files from the central repository down to the nodes when the resynchronization occurs.

•A number of MBeans are also available on the Deployment Manager - allowing the configuration management and the synchronization process to take place.

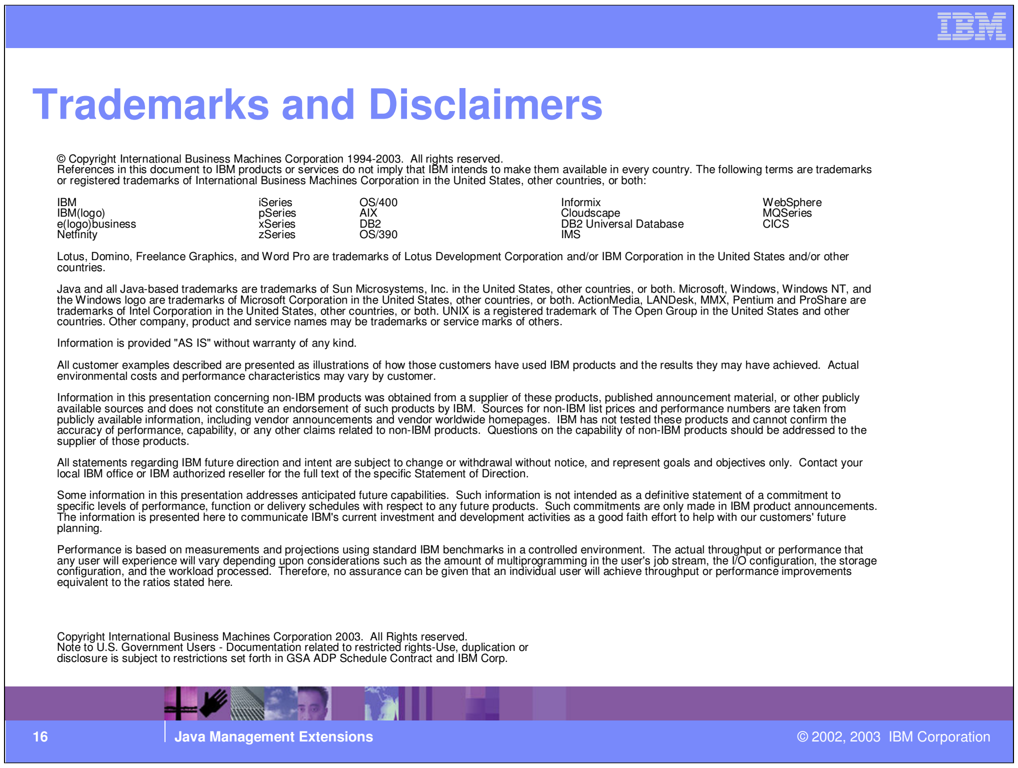
•The Admin Console and the wsadmin client can be connected to the Deployment Manager for centralized administration.



Summary

- **JMX provide a standard way to expose resources**
  - Application resources
  - Administrative resources
- **Flexible and scalable management framework**
  - Opens the door to third-party management tools providers
  - Extensible
- **WebSphere offers a number of interfaces**
  - Admin Console
  - Scripting client (BSF)
  - more to come

© 2002, 2003 IBM Corporation



Trademarks and Disclaimers

© Copyright International Business Machines Corporation 1994-2003. All rights reserved.
References in this document to IBM products or services do not imply that IBM intends to make them available in every country. The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

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

Lotus, Domino, Freelance Graphics, and Word Pro are trademarks of Lotus Development Corporation and/or IBM Corporation in the United States and/or other countries.

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 trademarks of Microsoft Corporation in the United States, other countries, or both. 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. Other company, product and service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information in this presentation concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

Some information in this presentation addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. 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.

Copyright International Business Machines Corporation 2003. All Rights reserved.
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.