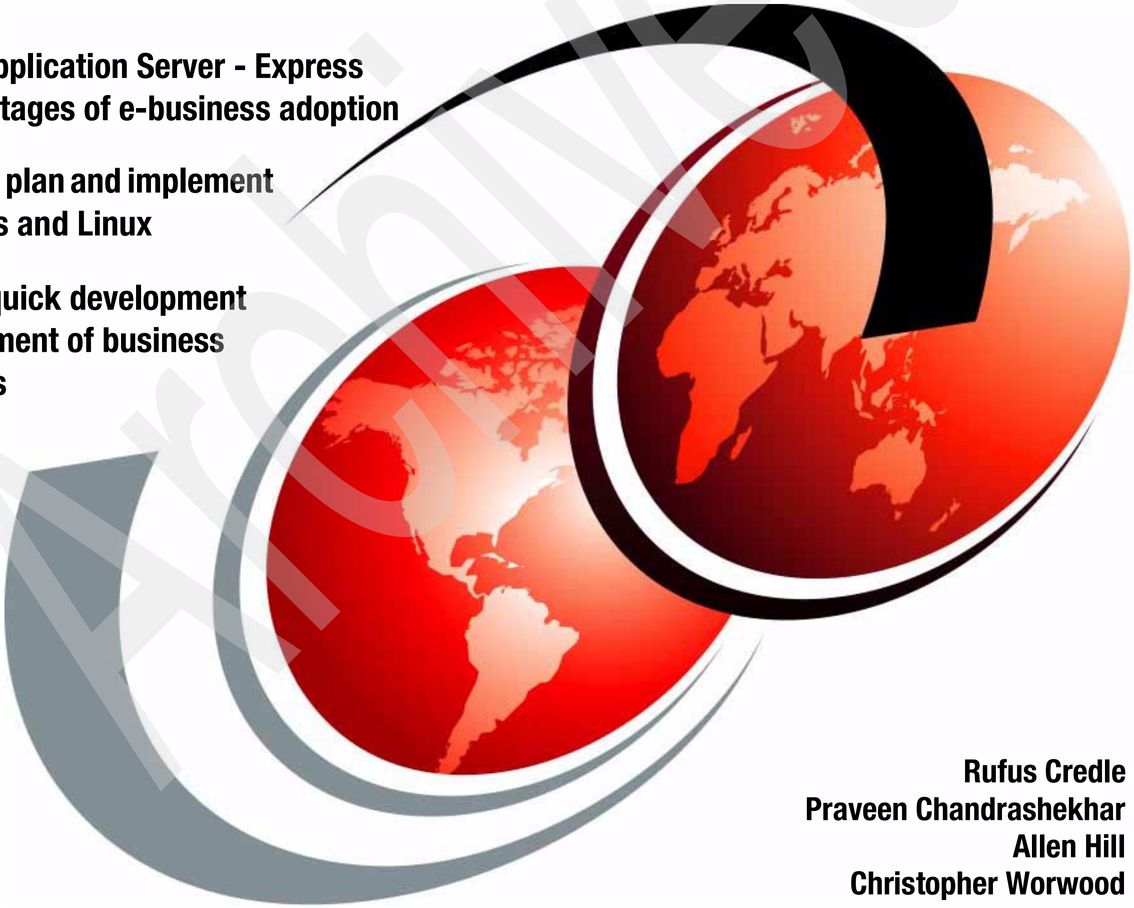
WebSphere. software

IBM

WebSphere Solution Guide: WebSphere Application Server - Express, Version 5.0



Positions Application Server - Express within the stages of e-business adoption

Helps you to plan and implement for Windows and Linux

Illustrates quick development and deployment of business applications

Rufus Credle
Praveen Chandrashekhar
Allen Hill
Christopher Worwood

Redbooks

ibm.com/redbooks

International Technical Support Organization

WebSphere Solution Guide: WebSphere Application Server - Express, Version 5.0

April 2003

SG24-6925-00

Note: Before using this information and the product it supports, read the information in "Notices" on page vii.

First Edition (April 2003)

This edition applies to IBM WebSphere Application Server - Express, Version 5.0.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other 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.

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 the 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.

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 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.

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

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

Cloudscape™
CT™
DB2®
eServer™
eserver™
IBM®
ibm.com®
IBM eServer™
Illustra™
iSeries™
Lotus Notes®
Lotus®
MQSeries®
Netfinity®
PartnerWorld®
pSeries™
Redbooks™
Redbooks(logo)™
Software Mall®
SLC™
SP™
Tivoli®
VisualAge®
WebSphere®
Word Pro®
xSeries™
XT™
z/OS™
zSeries™

The following terms are trademarks of other companies:

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, Windows 2000, SQL 7.0, SQL 2000 and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

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

Red Hat, the Red Hat "Shadow Man" logo, RPM, Maximum RPM, the RPM logo, Linux Library, PowerTools, Linux Undercover, RHmember, RHmember More, Rough Cuts, Rawhide and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.

SuSE is trademark or registered trademark of SuSE.

Linux is a registered trademark of Linus Torvalds.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

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

Preface

The *WebSphere® Solution Guide: WebSphere Application Server - Express, Version 5.0* was developed to support our IBM® Business Partners and independent software vendors.

The Solution Guide includes WebSphere channel-ready documentation and is broken into two separate documents:

1. A marketing and sales guide
2. An implementation and integration guide

The purpose of this implementation and integration guide is to simplify the planning and implementation of your e-business applications that are enabled for WebSphere Application Server - Express. It also takes into consideration the “Whole Product Concept”, which incorporates everything the customer needs to achieve the business goals that drive its purchase decisions, including consultation, design, configuration, implementation, OEM products and services, and on-going support.

In this IBM Redbook, we review and execute step-by-step instructions to set up and configure WebSphere Application Server - Express, Version 5.0 running Windows 2000 and Red Hat Linux 8.0. We also discuss how to successfully use the Application Server - Express wizard to create and customize a business application for your WebSphere Application Server - Express, V5 environment. This methodology provides a reference for a working solution that has been system-assured and can be quickly implemented. Additional information on education and support is included to help you to understand and manage your WebSphere e-business solution.

This book is intended for technical professionals, IT architects, Business Partners, independent solution developers, customers and IBM IT specialists. It is assumed that the reader has some knowledge of the IBM @server product line, WebSphere Application Server 4.0 and above, IBM DB2® UDB and Microsoft SQL Server.

During first half of 2003, WebSphere Application Server - Express will offer additional flexibility with the addition of browser-based administration. Users will have the option of using a more streamlined version of the browser-based administration console included in the more advanced configurations of WebSphere Application Server V5.0, including remote administration and greater administrative options. If users prefer, they can continue to use the simplified console integrated into WebSphere Studio for a single interface regardless of task.

The WebSphere Solution Guide: WebSphere Application Server - Express, Version 5.0 for marketing and sales can be obtained from the following Web sites:

- Business Partners: For the quick path to accessing the marketing and sales guide -- go to the WebSphere Innovation Connection Online Web site.
- IBMers: The marketing and sales guide can be accessed by going to the WebSphere Sales and Support intranet site

Please note that Red Hat Linux 7.2, SuSE 7.3, Red Hat Advanced Server 2.1, and SuSE SLES 7 were officially supported at the writing of this book.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Rufus Credle is a Senior I/T Specialist and certified Professional Server Specialist at the International Technical Support Organization, Raleigh Center. He conducts residencies and develops redbooks about network operating systems, ERP solutions, voice technology, high availability and clustering solutions, Web application servers, pervasive computing, and IBM and OEM e-business applications, all running @server xSeries™ systems. Rufus's various positions during his IBM career have included assignments in administration and asset management, systems engineering, sales and marketing, and IT services. He holds a BS degree in business management from Saint Augustine's College. Rufus has been employed at IBM for 22 years.

Praveen Chandrashekhar is an Advisory IT Availability Professional in IBM Global Services - E-Business Enablement, for Asia Pacific. He is based out of India and has 8 years of IT experience with 5 years of experience in IBM. He has almost 5 years of extensive experience of working on Aix, Linux and Internet applications based on the same. He has worked extensively on Internet security, Apache, BIND, and various other Internet applications. He has set up more than 10 Internet Service Providers and 15 Internet Data Centers across Asia Pacific as part of his job profile within IBM. He has written extensively on Linux and its related applications in various publications in India, and also has written a Redbook on Tivoli® Global Sign on.

Allen Hill is a Senior WebSphere Engineer for Software Information Systems (SIS), a IBM premier business partner based in Lexington, Kentucky. He holds a degree in software development and has eight years experience in software development. His areas of expertise are WebSphere Application Server, J2EE development, and iSeries™ application development.

Christopher Worwood is a senior IT Specialist in South Africa. He has 6 years of experience in e-business and Internet technology. His areas of expertise include WebSphere Application Server Java J2EE application development, DB2 and MQ Series. He has worked extensively in the financial services sector.

Thanks to the following people for their contributions to this project:

Gail Christensen, Cecilia Bardy, Margaret Ticknor, Tamikia Barrow
International Technical Support Organization, Raleigh Center

Yvonne Lyon
International Technical Support Organization, San Jose Center

Roger Cheung, WebSphere Software Services Planning
IBM Toronto

Rick Stanley, WebSphere Application Server Channel Development
IBM New York

Donna Zubrod, WebSphere Application Server Marketing
IBM Research Triangle Park

Robin Hackney, WebSphere Application Server - Express Marketing
IBM Research Triangle Park

Colin Richardson, Business Integration Programs
IBM Research Triangle Park

Janet Olausen, Technical Support Marketing Program Manager
IBM Minneapolis

Mark Edwards, WebSphere Application Server - Express
IBM Research Triangle Park

Timothy Deboer, WebSphere Studio Application Developer - Server Tools
IBM Toronto

Carla Sadtler, ITSO WebSphere Redbook Development
IBM Research Triangle Park

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

`ibm.com®/redbooks/residencies.html`

Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  `ibm.com/redbooks`

- Send your comments in an Internet note to:

  `redbook@us.ibm.com`

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HQ7 Building 662
  P.O. Box 12195
  Research Triangle Park, NC 27709-2195

1

WebSphere Application Server roadmap

Dynamic e-business is about adapting e-business processes and associated systems to support changing business strategies and tactics. As e-business continues to grow in speed and complexity, so must the IT infrastructure that supports it. To provide a competitive advantage in a constantly evolving marketplace, the foundation of any e-business must facilitate fast and efficient responses.

The need to improve both top and bottom-line results drives businesses to bring new products and services to market faster, to create a compelling Web experience that improves the quality and quantity of site traffic, to increase transaction volume and frequency, and to leverage reusable assets. Dynamic e-business is about creating business processes that can meet several of these goals with decreasing levels of incremental investment.

1.1 WebSphere Application Server - Express, V5 capabilities

Dynamic e-business demands a robust, flexible software infrastructure that can enable you to:

- Build applications based on open industry standards within multi-vendor environments
- Rapidly develop and deploy applications to meet your needs today and then extend those applications dynamically as your needs change
- Incorporate mobile devices, new languages and locales, and new trading partners, with a minimum amount of recoding or redeployment
- Provide high performance, scalability and security to maximize application integrity

The IBM WebSphere software platform for e-business is a comprehensive set of award-winning, integrated e-business solutions. It is a software platform based on industry standards, making it flexible and pluggable, which can allow you to adapt on-the-fly as markets shift and business goals change.

Building on this robust platform, you can integrate diverse IT environments to maximize current investments:

- Deliver core business applications to the Web.
- Grow these applications to meet changing needs and increasing demand.
- Create a differentiated e-business that sets your business apart from the competition.

For more information about the full line of WebSphere software platform products and solutions, visit:

http://www.ibm.com/WebSphere

1.2 WebSphere Application Server - Express, V5 positioning

IBM WebSphere Application Server - Express, Version 5 is a specialized configuration of IBM WebSphere Application Server, Version 5 that provides an easy on-ramp to e-business.

1.2.1 IBM WebSphere Application Server Version 5.0

WebSphere Application Server Version 5.0 is a proven, agile e-business platform designed to support today's business imperatives. WebSphere Application Server, V5 facilitates reducing overall costs, improving customer loyalty, and adapting quickly to exploit new business opportunities.

Serving as an operating system for the Internet, WebSphere Application Server V5 provides:

1. Comprehensive open standards-based integration platform
2. Highly integrated application development environment
3. Agile application deployment and administration
4. Intelligent end-to-end application optimization

WebSphere Application Server fully integrates with the WebSphere Studio application development platform, which is based on the Eclipse open source project. This combined development and deployment environment supports rapid time to market of new applications and dramatic improvements in asset reuse across the organization and over time.

WebSphere Application Server V5 provides several configurations to help a business seamlessly grow from basic to sophisticated as business needs require. Upward compatibility and migration paths are clear. The path includes:

- WebSphere Application Server - Express provides an easy on-ramp to e-business
- WebSphere Application Server - provides full J2EE 1.3 compliance within a distributed environment with diverse platforms and multiple enterprise systems on the "back-end".
- Network deployment configuration provides the option for clustering and workload management.
- WebSphere Application Server Enterprise - provides additional integration, application framework and deployment services designed to simplify complex mixed enterprise environments, provided by WebSphere Application Server Enterprise.

1.2.2 IBM WebSphere Application Server - Express, Version 5

IBM WebSphere Application Server - Express, V5 offers fast development, deployment, and management of dynamic Web experiences that engage customers and empower employees.

Medium-sized businesses and enterprise departments must squeeze every bit of productivity out of their IT resources to stay competitive. They need to utilize the Web to cost-effectively increase employee productivity, sales and customer satisfaction. Affordable start up costs, easy to use application development and management tools, and ready availability of skills, services and applications are all critical. At the same time, performance and security should not be compromised.

WebSphere Application Server - Express provides an integrated application development and deployment platform that was specially designed to meet these needs. WebSphere Application Server - Express helps enterprise departments, stores and branch locations, as well as the mid-sized suppliers they work with to reliably leverage existing database and application assets to create dynamic Web applications that drive up productivity and sales. In addition, applications deployed on WebSphere Application Server - Express can run without change on more advanced WebSphere Application Server configurations providing an extremely cost-effective growth path.

Designed to meet the unique needs of these companies, IBM WebSphere Application Server - Express, Version 5 combines the power and function of an industry leading application server with a smaller, up-front investment making WebSphere Application Server - Express, V5 a low-risk, affordable entry to e-business.

Supporting the latest specifications for Java Server Pages, Java Servlets and Web services, WebSphere Application Server - Express, V5 can help you build dynamic Web sites by viewing information in databases and performing simple updates - while also providing the ability to create and use Web services.

IBM WebSphere Application Server - Express, Version 5 supports the key capabilities and benefits described in the following sections.

Ready to go

It provides the tools necessary to create and run a simple dynamic Web site in a tightly integrated and affordable package including a simplified Web application server, a development environment based on IBM WebSphere Studio, which is powered by Eclipse technology, and wizards and samples that can help reduce complexity and deployment costs.

Easy to use

It is designed with your convenience in mind:

- It delivers an out-of-the-box solution for building static and dynamic Web sites
- It includes a quick, wizard-driven installation (only five clicks to install)

- It provides one-click application assembly and deployment that can help improve productivity
- It offers near-zero maintenance that can help minimize administration requirements

Migration ready

It offers a migration ready solution:

- Migration is available to other IBM WebSphere Application Server V5 and IBM WebSphere Studio V5 configurations, and other Eclipse technology based tools when more advanced development and deployment capabilities are required
- It leverages your investment in WebSphere Application Server - Express, V5 by offering a discount when you are ready to upgrade

1.3 WebSphere Application Server environments

Many firms with internal development projects have a three-tier scheme that may include the following phases:

- The initial development phase is undertaken on one system and sometimes on an isolated development LAN.
- Once the development is “complete”, a simulated live environment is created for the sole purpose of confirming the completion and its correct functionality. This is the Staging phase, and may likewise be on an isolated test LAN.
- Only after these two phases have run their course is an application “promoted” to the production environment.

Migrating from a staging to production environment means all development of e-business applications will take place on a server physically separate from than the production servers that will service end-user HTTP requests.

> **Note:** In this section, we discuss the different tiers for a development, staging, and production environments to help you understand the use of the IBM WebSphere Application Server. However, for the implementation of WebSphere Application Server - Express, Version 5.0, we focus on a single-tier environment.

1.3.1 Why we need separate environments

In the design of non-distributed or centralized applications, the process flow from development to production follows the stages shown in Figure 1-1. The application developer first works on the application, which is typically unstable and marginally close to the production site. Once the code is tested and proves to be stable, it is moved to the staging server.

The application on the staging server should be as close to 100 percent production quality as possible. The code and content in this environment is then tested thoroughly, and any final bugs are ironed out. The last step is moving from staging to production. Once on the production server, a site is live and accessible to the audience. When building centralized sites, the code is moved from a single server to another single server.

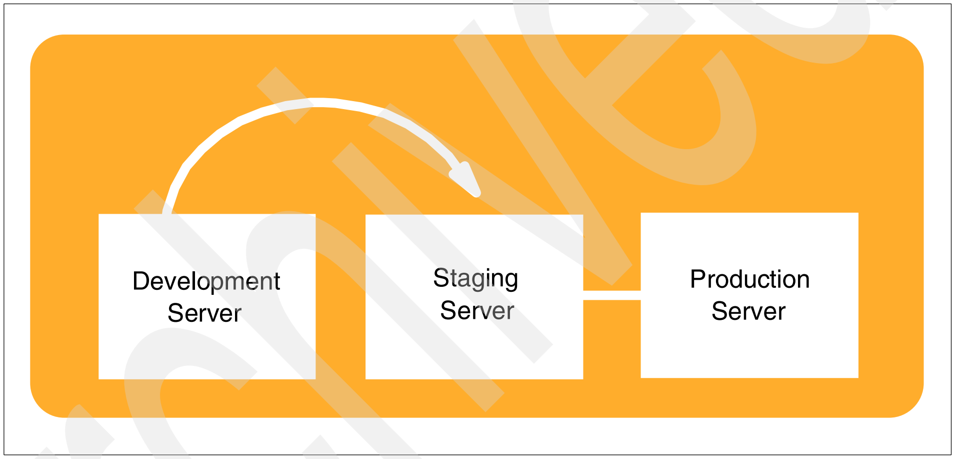


Figure 1-1 Process flow diagram

Distributed sites follow a different process, especially for the final two steps: staging and production. As shown in Figure 1-2, multiple applications must be updated with the relevant content. For example, there needs to be a concurrent update in two different applications running in production server 1 and 2, which are hosting two different applications.

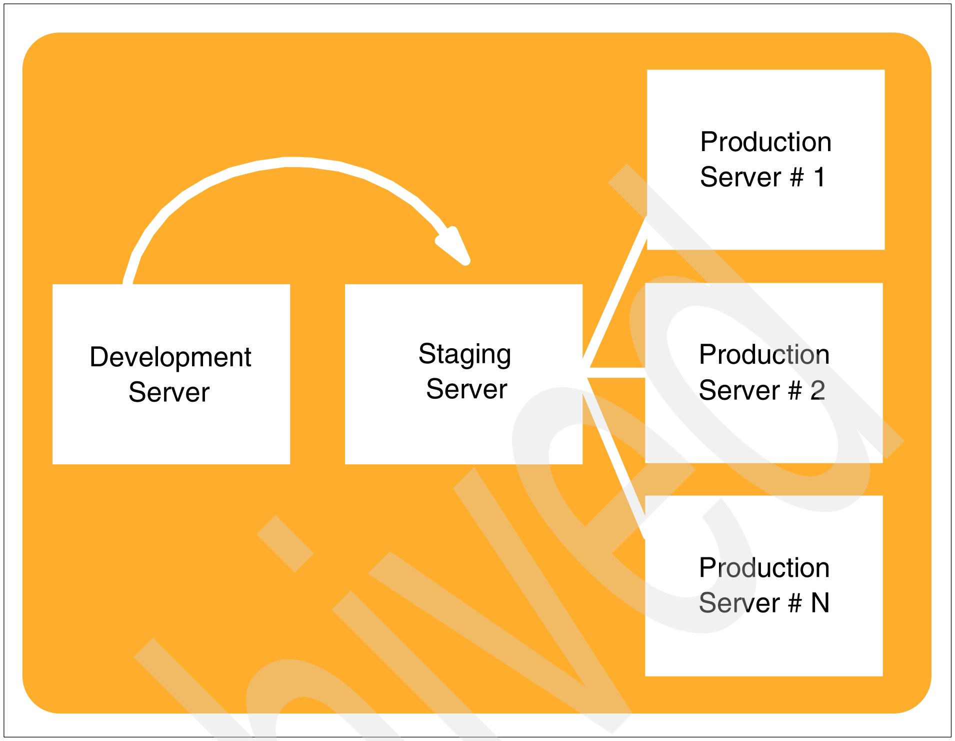


Figure 1-2 Process flow diagram

Development environment

A development environment is usually:

- Unstable
- Dynamic
- Proof-of-concept testing
- For rapid application development

Generally, for these purposes we would choose a Windows platform.

Staging environment

A staging environment is very similar to a production environment. It is mainly used to do quality assurance (QA) and load balancing. Here QA refers to functional testing, stress testing, and all other types of testing.

Production environment

This is the live environment that is exposed to the World Wide Web. In this environment, a change to any existing code needs to go through a cycle of a change requests. Hence it is a very stringent environment. Generally, production environments are UNIX flavors, which are very rugged and stable.

We have discussed the various phases of an e-business solution built from scratch. However, this may not always be the case. There may be applications that have already been built and need to be ported or migrated to a WebSphere Application Server environment. WebSphere Application Server is a J2EE-compliant Application Server. So for deploying a J2EE application, the following components need to be built for an enterprise e-business application:

- **WAR (Web Archive):** The Web-related components (HTML, JavaScript, JSP)
- **JAR (Java Archive):** The Java classes that make up the business logic components
- **EAR (Enterprise Archive):** The JAR files plus WAR files that make up an enterprise application

> **Note:** The minimum deployable unit in WebSphere Application Server is a WAR file. If the application is developing EJBs, then a JAR file and an EAR file are necessary.

2

IBM and ISV e-business applications and offerings

In this chapter we provide information regarding the appropriate e-business solution to address the needs of your customers. In specific, we address the WebSphere Application Server - Express, V5 solution offering.

2.1 Setting the foundation for stages of e-business adoption

Over the past year, IBM has done considerable work in understanding and identifying how customers are adopting e-business technologies to meet their tactical and strategic long-term business needs and requirements. IBM has worked with McKenna Group and some of its internal market research resources to interview over 21,000 customers to understand their e-business processes and associated systems to support their customer's changing business strategies and tactics.

They categorized e-business adoption into six distinct stages:

- There is an initial state of Web access, with users just accessing the Internet to find information. However, this is not a real use of e-business.
- The first real step of e-business is Web publishing, using the Web as a marketing channel for information about the company and/or its products, etc. and making that available to the outside world.
- The second stage is called e-business transactions, where external users, through Web applications, are allowed to connect to applications and data that were previously reserved for internal users. These transactions involve full update mode, full read/write modes, and read-only modes.
- The third stage is where companies start to focus more and more on internal integration, optimizing their internal processes to make sure that as their processes become more visible to the outside world, they become increasingly more integrated and more effective.
- Stage three evolves into stage four, external integration or creation of value networks, which are external application integration points where companies work together with other companies to deliver more value to their customers than they would be able to do alone.
- Where we see all this evolving towards is a model we call dynamic e-business. Dynamic e-business is a business model where a company can focus on its core activity and outsource all surrounding activities in a dynamic fashion over the Internet. This is the ability for one application to communicate with another application function over an Internet network and leverage and integrate with that application function. Basically, what we're seeing is a long-term trend towards digital businesses, the creation of increasingly more automated and more integrated business processes that are transforming the way business is done right now.

The business scenarios outlined in this guide are targeted at these stages of e-business adoption, using the stages as a foundation, and expressed as solution scenarios that map out to different customer pains from a business consultative perspective. A “Whole Product Concept” incorporates everything that a customer needs to achieve the business goals that drive its purchase decisions, including consultation, design, configuration, implementation, OEM products and services, and on-going support.

What does all this “road to e-business” and “stages of e-business adoption” mean from a “requirements for Web applications” perspective?

Web applications will evolve from just doing static Web serving and e-mail with limited use of dynamic content into applications where there is increasingly more of a separation of business logic and presentation logic and higher degrees of transactional integrity and integration with databases or existing transaction systems. As we move into the stage of internal integration, we see the need for applications that can coordinate transactions between different in-house applications, and this need evolves into the coordination of transactions between in-house and external applications. As we move into the world of dynamic e-business, there is a need for applications that can coordinate transactions between in-house applications and outsourced applications that are linked dynamically.

We clearly see a trend towards increasingly intelligent transactions — meaning transactions that can coordinate resources across a variety of systems and that can be long lived, not just for a couple of seconds, and can actually take weeks or even months to complete. They are also self-conscious — for example, transactions containing decisions on business logic as to which external provider should be leveraged in certain scenarios. Being able to develop these intelligent transactions will require much more integration work, tied together systems, applications, databases, etc., and all of this relies on a real, solid e-business infrastructure for all of these transactions to perform appropriately.

2.2 Whole Product Concept

The Whole Product Concept is based on everything the customer needs to achieve the business goals that drive its purchase decisions. It includes not only the core technology, products and services you offer, but the on-going support and third-party products and services necessary to solve the customer’s business problem.

2.3 Business scenario: Getting started in e-business

This business scenario applies to customers just getting started or customers moving from stage 1 to stage 2 of e-business adoption and the recommended product for this particular situation.

In the early stages of e-business adoption, the WebSphere Application Server - Express, V5 is the product that best meets their current needs, while at the same time providing a platform of products that will grow as they grow.

WebSphere Application Server - Express, V5 is for customers that need to provide a more valuable experience for their customers as well as more efficient business processes, thereby attracting new customers, growing customer loyalty, and reducing operating expenses.

This scenario applies to customers looking for a solution that will:

- Enable simple integration between their back-end business systems and Web sites
- Provide a total solution that meets their specific needs
- Allow them to implement this solution with their resources and skill base currently in house
- Provide them with the most value for their money on a solution and total cost of ownership basis

Why choose WebSphere Application Server - Express, V5?

IBM's experience as one of the leading database providers makes it uniquely able to build a smooth link between core business systems and Web sites.

WebSphere Application Server - Express, V5 includes an application server, development tools, and application technology such as customized end-user experiences, cataloging, and search capability, all paired with award-winning IBM support and three years of product updates.

It allows application development through JavaScript and wizards and requires no administration of the application server resulting in self implementation and easy maintenance. In addition, WebSphere Application Server - Express, V5 includes basic applications for immediate functionality, everything you need to get up and running to provide an immediate return on investment with no new skills development necessary. (See Table 2-1.)

Table 2-1 Getting started in e-business

Offering name	Target audience	Audience profile, in terms of e-business stages of adoption	Core products, those most often in the buying vision	Optional products, added based on customer pains
Getting started in e-business	► Primary = SMB ► Secondary = departments in large enterprises	Customers just getting started -or- Customers moving from stage 1 to stage 2 of e-business adoption	Multiple offerings - hardware platform dependent 1. pSeries™: StartNow based 2. pSeries: WDS, CA, iSeries wireless 3. zSeries™	WebSphere Application Server AEs

Key message: WebSphere Application Server - Express, V5 has everything you need to get up and running with dynamic Web site development and deployment.

Value proposition: With WebSphere Application Server - Express, V5, you realize an immediate return on your investment by being able to utilize the resources and skill base you currently have.

WebSphere Application Server - Express provides an integrated application development and deployment platform that was specially designed to meet the needs of enterprise departments, stores, and branch locations, as well as the mid-sized suppliers they work with to reliably leverage existing database and application assets to create dynamic Web applications that drive up productivity and sales. Note that WebSphere applications deployed on even the simplest configuration can run without change on more advanced configurations providing an extremely cost-effective growth path.

3

WebSphere Application Server - Express, Version 5.0

IBM WebSphere Application Server - Express, Version 5.0 offers a combination comprising both a development tool and an application server that, together, provide a single integrated package geared towards Web centric applications.

This product ships with the following components:

- WebSphere Studio Site Developer Version 5
- WebSphere Application Server - Express, Version 5.0
- IBM Agent Controller
- Five Web application examples and supporting documentation
- Cloudscape™ database pre-populated with data specific to examples
- A Getting Started Guide containing three tutorials that will help walk you through creating a Web application

3.1 Product overview

There are various methods of deploying the product, and to help you understand it better, we highlight some of the critical points here:

- WebSphere Studio Site Developer is the integrated development environment that you use to build, deploy, and test Web applications.
- WebSphere Studio Site Developer has a self-contained, fully functional, WebSphere Application Server that can be used for deploying and testing your Web applications. You always get this test environment when you choose to install WebSphere Studio Site Developer.
- WebSphere Application Server - Express includes a separately installable WebSphere Application Server. You can choose to install this separate application server along with WebSphere Studio Site Developer or as a stand-alone server on a different machine than you install WebSphere Studio Site Developer. You use WebSphere Studio Site Developer to deploy applications and to administer this separate application server.
- The WebSphere Application Server - Express examples characterize a simplified programming model focusing on tag libraries, Java Server Pages, JavaScript, and servlets. These examples represent commonly used functionality in Web applications, and are intended for use as reference or to be modified to develop and deploy customized applications.

Eclipse is used in WebSphere Application Server - Express, Version 5.

Eclipse is an open-source project that creates royalty-free technology and a universal platform for tools integration. Eclipse-based tools give developers freedom of choice in a multi-language, multi-platform, multi-vendor environment. Eclipse delivers a framework for developing plug-ins that makes it easier to create, integrate, and use software tools, saving time and money. By collaborating and sharing core integration technology, tool producers can concentrate on their areas of expertise and the creation of new development technology.

The Eclipse platform is written in the Java language and comes with extensive toolkits and examples for construction plug-ins. It has already been deployed on a range of operating system environments, including Linux, MAC OS X, QNX, and Windows-based systems. Full details of the Eclipse community and whitepapers documenting the design of the Eclipse platform are available at:

`http://www.eclipse.org`

Toolkits and complimentary middleware from IBM integrate with the WebSphere Studio environment, allowing you to quickly and easily add new functionality targeted at specific development needs. Plug-in tools from other vendors, also built using the open Eclipse platform, integrate with the WebSphere Studio environment to provide additional value.

WebSphere Application Server - Express, Version 5.0 is designed to create and run a simple dynamic Web site in a tightly integrated and affordable package. It is easy to install and offers wizards, samples, and application templates that help you write the code, or provide the code whenever possible. The development tool includes a list of valid alternatives for completing the current line of code to speed the development process.

It also offers you a Struts builder which offers a high-level view of an application, allowing you to map out the pages and actions within an application before writing any code.

The application server also has a built in debugger that enables you to detect and diagnose errors in your program running either locally or remotely. You can debug either live server-side code as well as programs running locally on your workstation.

WebSphere Application Server - Express also has sophisticated XML support. It provides a comprehensive visual XML development environment that allows you to:

- Debug and edit XSL with code assist
- Create XML documents from a DTD/ XML schema, and vice-versa
- Generate JavaBeans from XML, and vice-versa
- Debug XSL stylesheets using the XSL debugger and many more features.

The WebSphere Application Server - Express, Version 5 contents are depicted in Figure 3-1.

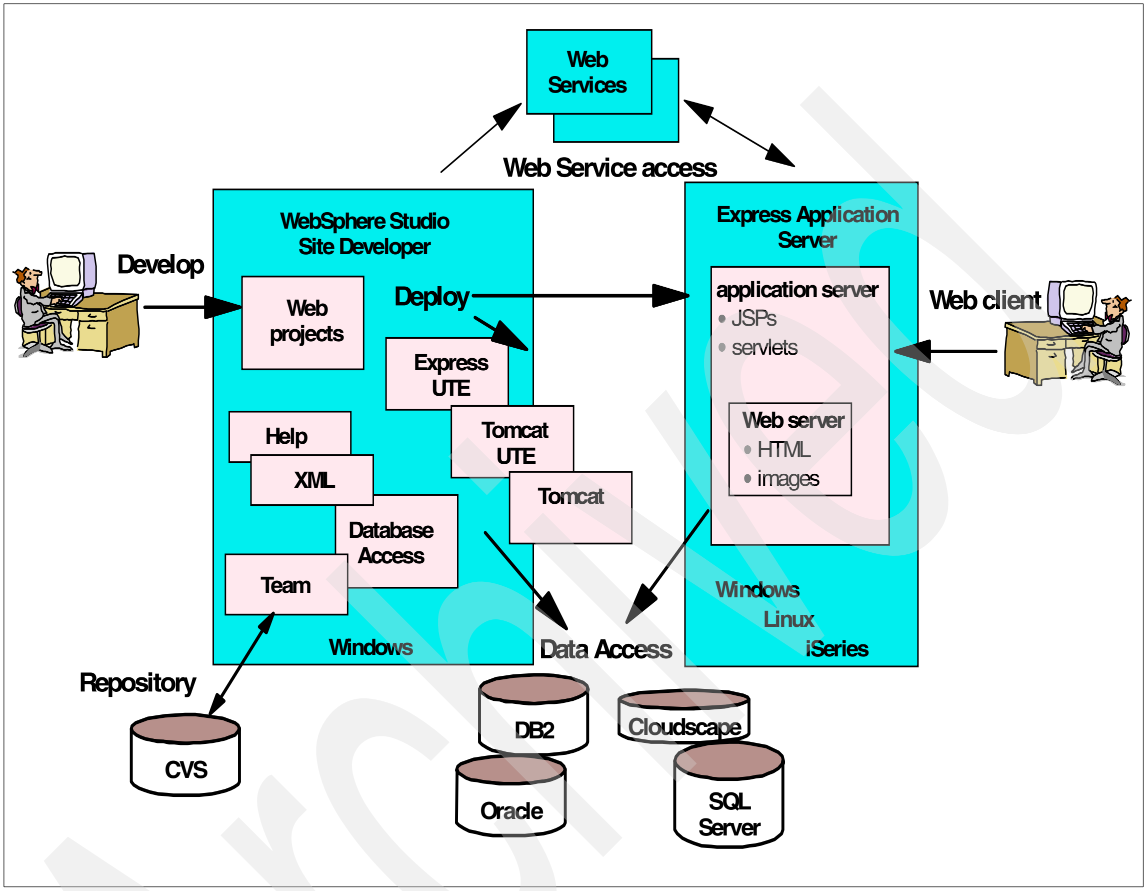


Figure 3-1 WebSphere Application Server - Express overview

3.2 Expected results

Upon completion of the base service, the customer's WebSphere Application Server - Express server will be able to respond to requests from World Wide Web browsers in both unencrypted and encrypted (SSL) modes. In addition, the customer will have obtained the following skills:

- Web server administration
- WebSphere Application Server administration
- Secure server (SSL) mode configuration
- Basic administration of LDAP, DB2, TCP/IP, Security

3.3 Information roadmap

This guide is divided into various chapters. Each chapter deals with a specific area of WebSphere Application Server - Express.

The main topics covered in this book are:

- Installation
- Application deployment
- Sample business cases

Other topics include hardware and software prerequisites, service provider skills, education and related publications.

3.4 Installation approach

The installation approach, as well as setting up a development environment for WebSphere Application Server - Express, Version 5, are depicted in Figure 3-2, which shows a typical development environment.

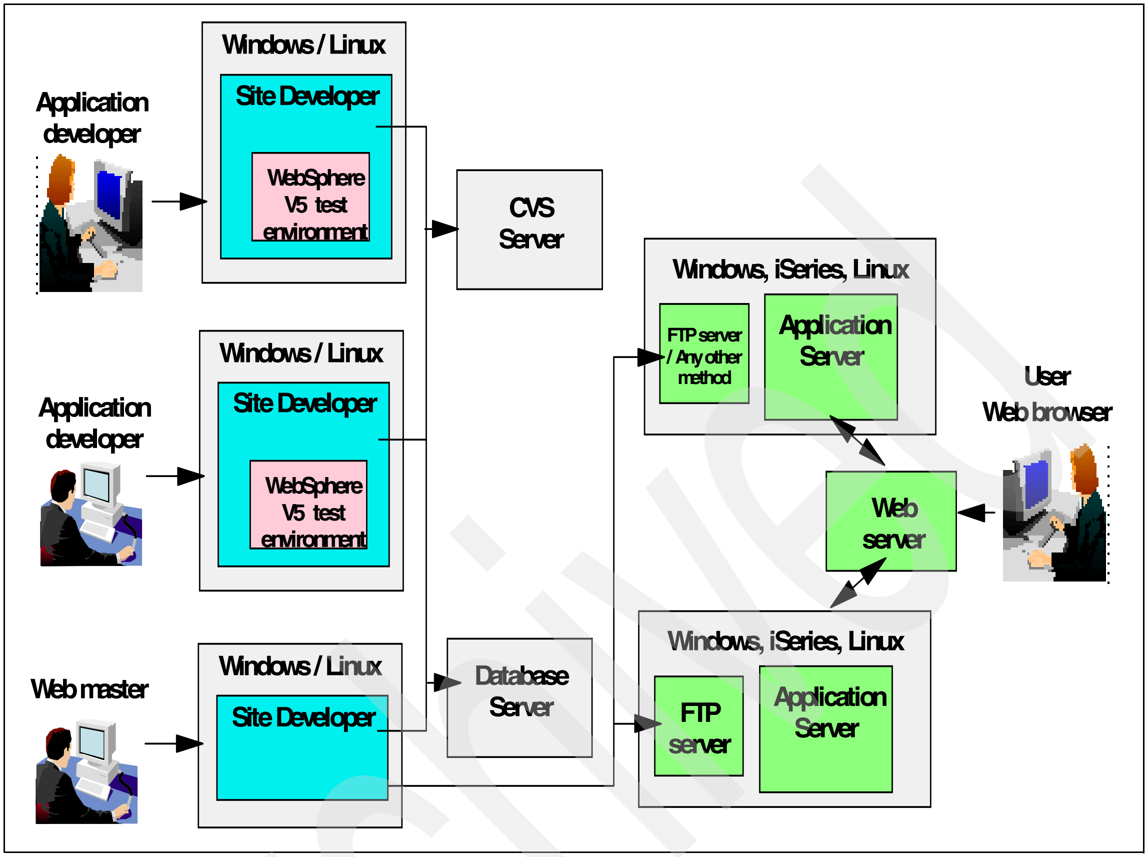


Figure 3-2 Installation approach for WebSphere Application Server - Express

3.5 Hardware requirements for installation

In this section, we list the hardware requirements for the installation and configuration of WebSphere Application Server - Express, Version 5. Following are the specifications for the video hardware component for the respective operating systems:

For Windows:	VGA (800 x 600) display minimum (1024 x 768 recommended)
For Linux:	SVGA (1024 x 768) display minimum

3.5.1 Installing only tools on one machine

For installing only the tools on one machine, these are the hardware requirements:

- Intel Pentium II processor minimum; Pentium III at 500 MHz, or faster, recommended.
- 512 MB RAM minimum (768 MB RAM recommended).
- Disk space: 650 MB minimum disk space for installing WebSphere Application Server - Express and additional disk space for your development resources. Additional disk space is required if you download the electronic image to install WebSphere Application Server - Express, Version 5.0.

3.5.2 Installing only tools and Remote Server on one machine

For installing only the tools and a Remote Server on one machine, these are the hardware requirements:

- Intel Pentium II processor minimum; Pentium III at 500 MHz, or faster, recommended.
- 512 MB RAM minimum (768 MB RAM recommended).
- Support for a communications adapter if Windows is the platform of installation.
- Support for TCP/IP and an adapter if Linux is the platform of installation.
  It should also be noted that running Linux Red Hat for the production server component requires one machine at 512 MB RAM minimum (768 MB RAM recommended) for the tools component and one additional machine with 256 MB RAM minimum (512 MB RAM recommended) for the production server component.
- Disk space: 750 MB minimum disk space for installing WebSphere Application Server - Express and additional disk space for your development resources. Additional disk space is required if you download the electronic image to install WebSphere Application Server - Express, Version 5.0.

3.5.3 Remote Server only on one machine for Windows NT or 2000

For installing only the Remote Server on one machine running the Windows operating system, these are the hardware requirements:

- Intel Pentium III at 500 MHz, or faster, recommended
- Support for a communications adapter
- 256 MB RAM minimum (512 MB RAM recommended)

- Disk space: 100 MB minimum disk space for installation including SDK, and additional disk space for your development resources. Additional disk space is required if you download the electronic image to install WebSphere Application Server - Express, Version 5.0.

3.5.4 Remote Server only on one machine running Linux

For installing the Remote Server only on one machine running Linux, these are the hardware requirements:

Note: You must have a separate machine for the server if you are using Red Hat Linux.

- Intel Pentium III at 500 MHz, or faster, recommended
- Support for a communications adapter
- 256 MB RAM minimum (512 MB RAM recommended)
- Disk space: 100MB minimum disk space for installation including SDK. and additional disk space for your development resources. Additional disk space is required if you download the electronic image to install WebSphere Application Server - Express, Version 5.

3.5.5 Software requirements

Next we list the software requirements for WebSphere Application Server - Express, Version 5.

Operating systems:

- One of the following operating systems are required:
  - Windows 2000 Professional with Service Pack 2, or later
  - Windows XP Professional for the studio tools component, including the unit test environment only
  - Windows NT Workstation or server V 4 with Service pack 6 a, or later
  - Linux Red Hat V 7.2 for the Studio tools component, including the unit test environment only
  - Linux Red Hat AS V 2.1 for the Remote Server component only
  - Linux - SuSE Version 7.2

- While the supplied DB2 personal edition Version 7.2 is not required to install WebSphere Application Server - Express, it is nevertheless required to work with certain WebSphere Studio applications and samples.
- To profile your application, ensure that JRE 1.3.0 or later, is installed.

Web browser:

- **Windows:** Microsoft Internet Explorer 5.5 with Service Pack 1, or later, or Netscape Navigator Version 4.76
- **Linux:** Netscape Navigator Version 4.5; Universal Test Client, and Web Services Explorer require Netscape Version 4.6 or Mozilla Version 0.7, or later.

Skills planning and education

IBM recognizes that the key to profitable services engagements is repeatability; so numerous classes are offered to help your technical resources skill-up on the various components of the WebSphere platform. Your technical people can become certified in a variety of WebSphere family products (such as the WebSphere Application Server or WebSphere Host Integration, to name just two) by completing a prescribed series of technical courses and then passing the appropriate exams.

WebSphere certification tells your customers that the technical resources you send out to work on their projects are trained experts, and clients expect to pay extra for such expertise. According to early surveys by IBM, 64% of IBM Business Partners indicated that having their IT professionals IBM certified for e-business increased their ability to close a sale; 59% reported an increase in sales volume.

4.1 Skills required

An individual experienced in using WebSphere Application Server - Express, Version 5.0 should possess the following skills to conduct various business engagements:

- Basic Linux installation, configuration, and administration
- Configuration experience with WebSphere Application Server - Express, Version 5
- SSL concepts and implementation, including knowledge of Certificate Authority concepts and operation
- Internet resources and resource types, including knowledge of Uniform Resource Locators and Uniform
- Resource Indicators (URLs and URIs)
- Basic TCP/IP concepts, including addressing, routing, and protocol layering
- Basic HTML coding and concepts
- Basic understanding of the purpose of GIF, JPG, MPG, WAV, and similar visual and audio encoding standards
- In-depth Java programming knowledge is not required, but some knowledge of the servlet programming model is required.
- Basic Java servlet concepts
- Security considerations involved in the use of a Web server
- Intermediate knowledge of WebSphere Studio Site Developer

4.2 Educational site references

For further information on how to get yourself educated on WebSphere Application Server - Express, Version 5, go to this URL:
`http://www-3.ibm.com/software/webservers/appserv/express/education/`

More information regarding the Business Partner programs and other trainings can also be found at: `http://www.developer.ibm.com/websphere/was.html`

For various support related issues, especially in regard to fixpacks, etc., you can find more information at `http://www.ibm.com/support/us/`

For further information on how to get yourself educated and certified on WebSphere Application Server Version 5, go to this URL:
`http://cgse3.cgselearning.com/zone-websphere.html`

5

Planning and installation

In this chapter we discuss the various steps involved in planning for a WebSphere Application - Express installation. In addition, we provide examples of WebSphere Application Server - Express installed on the Linux and Microsoft Windows platforms.

5.1 Planning prior to installation

Before we go through the installation process, it is very important that you plan the prerequisites for installation in advance.

5.1.1 Before you install

Before you install the product, check the following:

- If you are installing Windows 2000, in addition to the space required to install the product, you must have at least 50 MB free on your Windows system drive, and your environment variable TEMP or TMP must point to a valid temporary directory with at least 10 MB free.
- If you have a previous copy of WebSphere Application Server - Express on your system, you must uninstall it before proceeding.
- If you have a previous copy of IBM Agent Controller on your system, you must uninstall it before proceeding.
- If you have a previous copy of WebSphere Studio Site Developer on your system, you must uninstall it before proceeding.
- If you are installing Linux, the installation of WebSphere Application Server - Express, Version 5 requires the usage of X-Windows on the system. Hence it is advisable that you install X-Windows in the system.
- The WebSphere Application Server - Express, Version 5 for Linux installation does not give you the option of changing the directory path for installation, and the default installation directory is "*/opt/...*" Depending on the packages that you are going to install, there might be two - three directories under opt used for the purpose of WebSphere Application Server - Express, Version 5. If you are concerned about file system management and security, you would need to plan for the same in advance. Also note that all files in this directory would be owned by root.
- Our Linux setup was based on a Netfinity® 3000 Intel Pentium II with 256 MB RAM and 9 GB of hard disk space.
- Our Windows setup was based on a xSeries 230 Intel Pentium III with 1 GB RAM and 18 GB of hard disk space.

5.2 Installing WebSphere Application Server - Express for Linux using a CD Image

To install WebSphere Application Server - Express from the product CD, follow these steps:

1. Insert the Linux CD-ROM into your CD drive.
2. Log in as root.
3. Mount the CD-ROM drive: type `mount /dev/cdrom`.
4. Change to the CD-ROM drive: type `cd /mnt/cdrom`.
5. Run the following command:

   ```
   ./launchpad.sh
   ```

   The WebSphere Application Server - Express installation launcher window is displayed. See Figure 5-1.

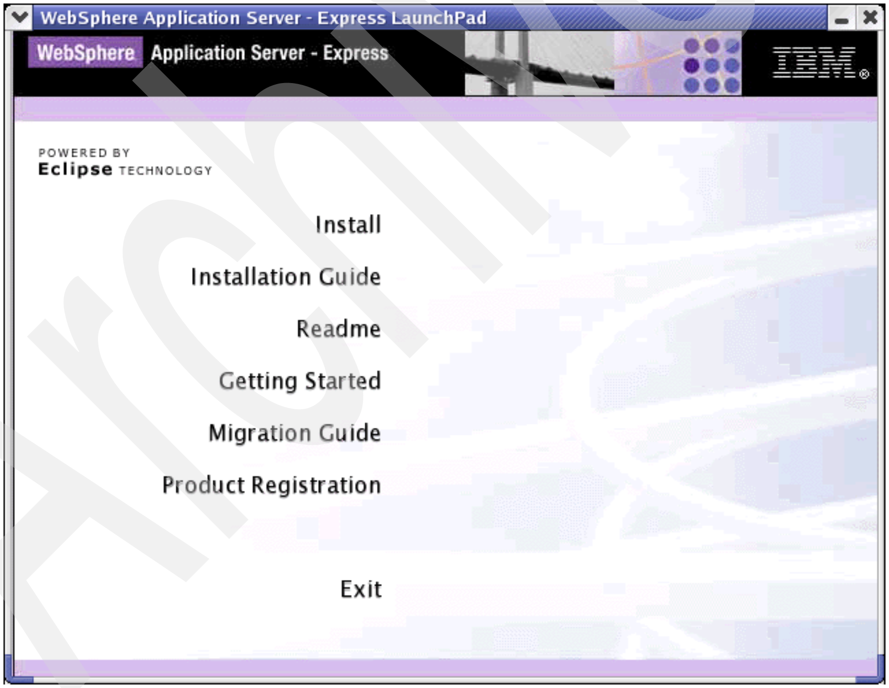

WebSphere Application Server - Express LaunchPad
WebSphere Application Server - Express
IBM
POWERED BY Eclipse TECHNOLOGY
Install
Installation Guide
Readme
Getting Started
Migration Guide
Product Registration
Exit

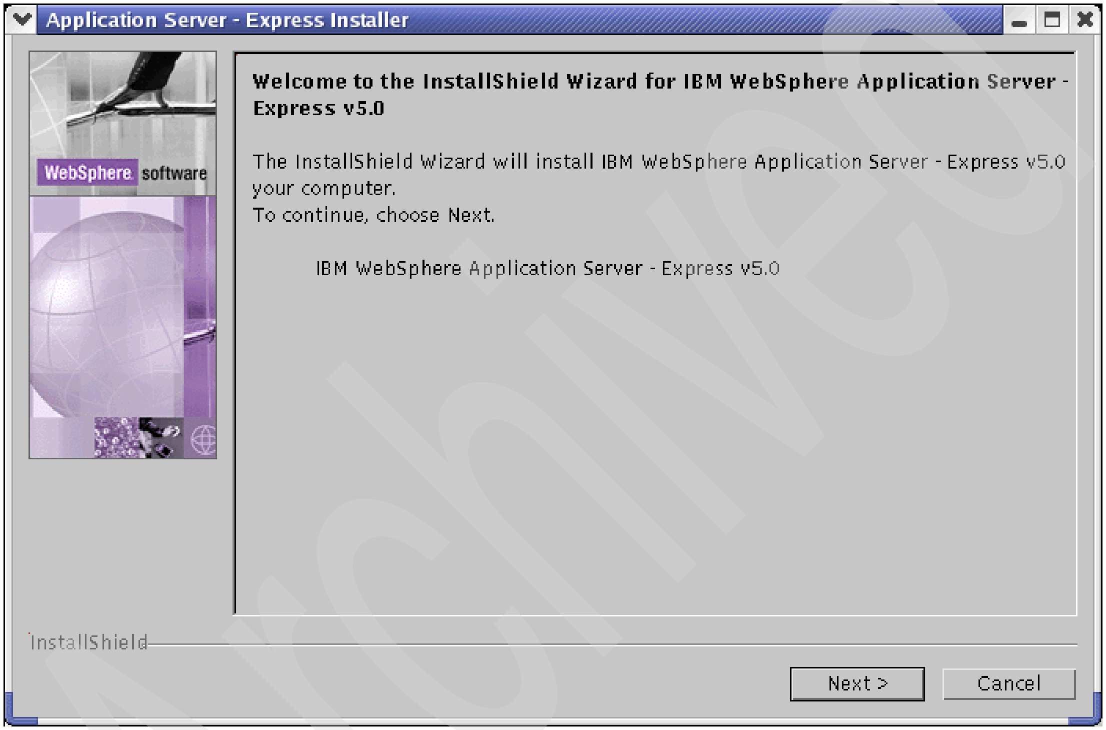
Figure 5-1 WebSphere Application Server - Express installation launcher

The WebSphere Application Server - Express installation launcher window contains several links that you can select to browse the readme file, the *Installation Guide*, the *Getting Started Guide*, or the *Migration Guide*.

6. Click **Install**.

   The Launchpad does not need to remain open during the install.

   The WebSphere Application Server - Express Installer window is displayed. See Figure 5-2.



Application Server - Express Installer

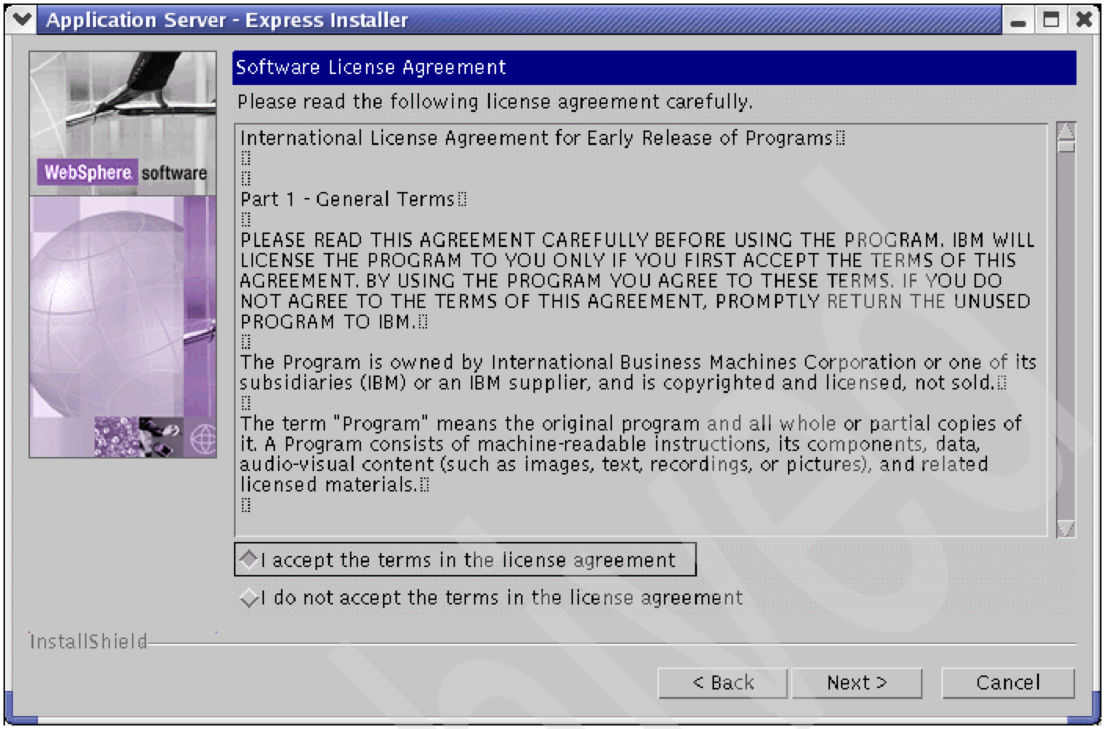
Welcome to the InstallShield Wizard for IBM WebSphere Application Server - Express v5.0

The InstallShield Wizard will install IBM WebSphere Application Server - Express v5.0 your computer.
To continue, choose Next.

IBM WebSphere Application Server - Express v5.0

Figure 5-2 Application Server - Express Installer

7. Click **Next**. The License Agreement window is displayed. See Figure 5-3.



Software License Agreement

Please read the following license agreement carefully.

International License Agreement for Early Release of Programs

Part 1 - General Terms

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THE PROGRAM. IBM WILL LICENSE THE PROGRAM TO YOU ONLY IF YOU FIRST ACCEPT THE TERMS OF THIS AGREEMENT. BY USING THE PROGRAM YOU AGREE TO THESE TERMS. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, PROMPTLY RETURN THE UNUSED PROGRAM TO IBM.

The Program is owned by International Business Machines Corporation or one of its subsidiaries (IBM) or an IBM supplier, and is copyrighted and licensed, not sold.

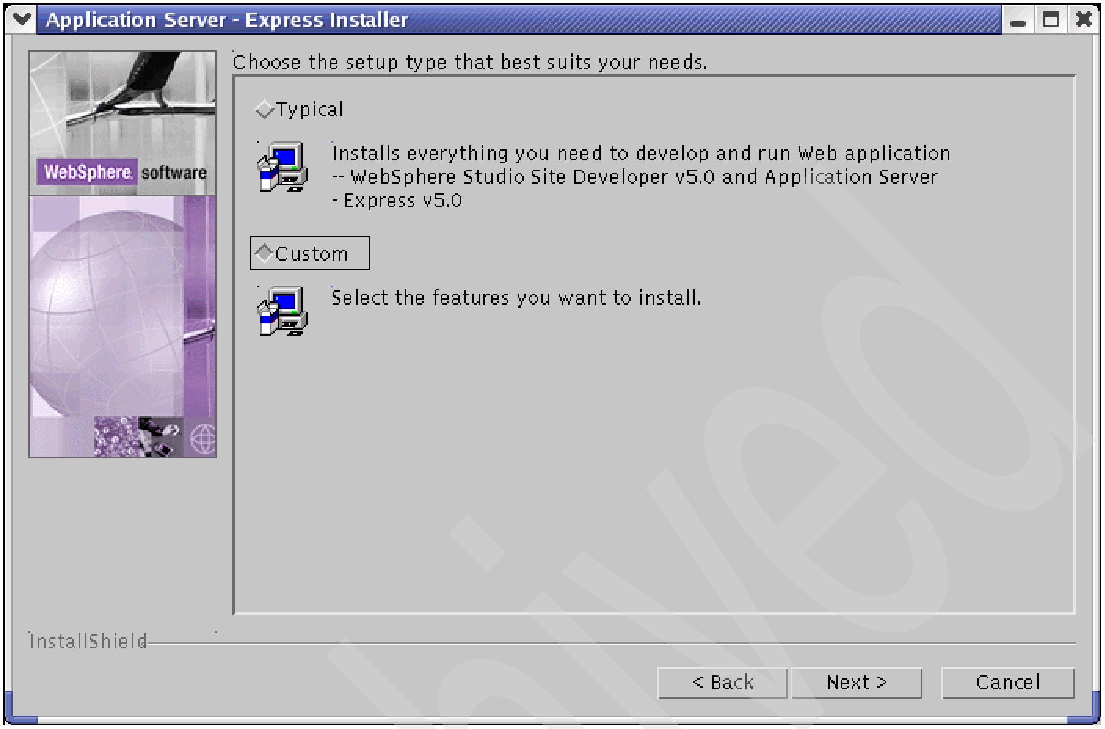
The term "Program" means the original program and all whole or partial copies of it. A Program consists of machine-readable instructions, its components, data, audio-visual content (such as images, text, recordings, or pictures), and related licensed materials.

◇ I accept the terms in the license agreement

◇ I do not accept the terms in the license agreement

Figure 5-3 Software License Agreement window

8. Select **I accept the terms in the licence agreement**.
9. Click **Next**. The installation type window is displayed. See Figure 5-4.



Choose the setup type that best suits your needs.

◇Typical

Installs everything you need to develop and run Web application -- WebSphere Studio Site Developer v5.0 and Application Server - Express v5.0

◇Custom

Select the features you want to install.

Figure 5-4 Installation type window

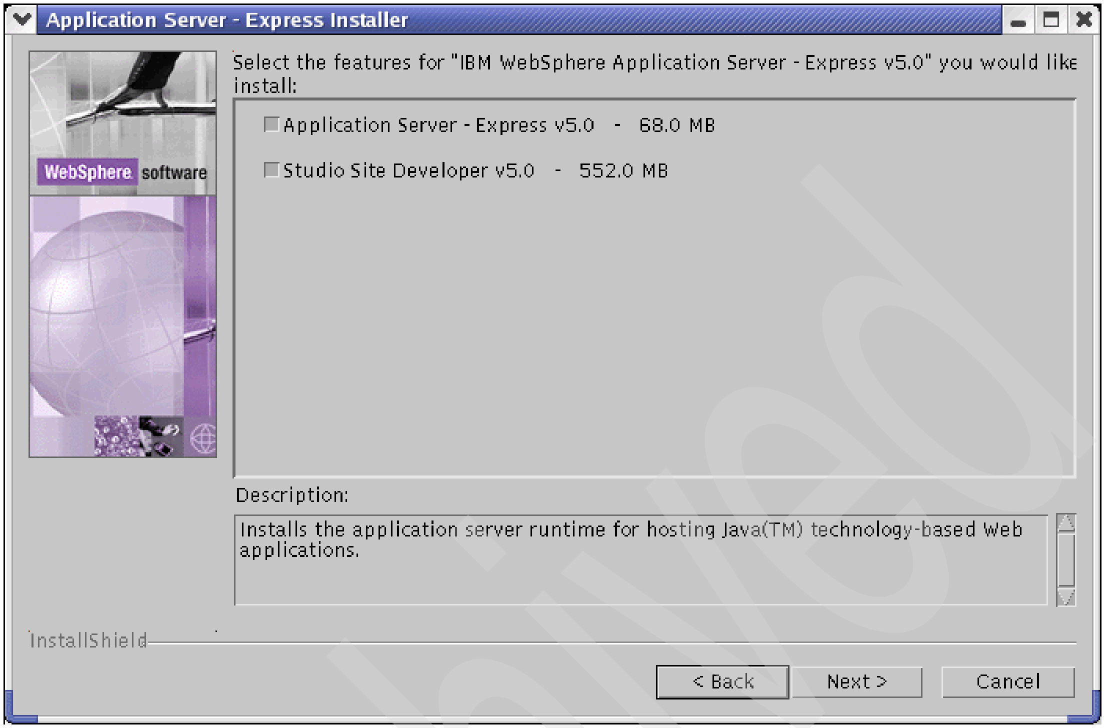
10. Select either **Typical** or **Custom**.

    A Typical install will install both the Application Server and the WebSphere Studio Site Developer. The Custom option gives you the choice to install either WebSphere Application Server - Express or WebSphere Studio Site Developer, or both.

    If you are installing just a development server, you should choose Typical. For a test or production server, you should choose Custom, which gives you the option to choose what software you would like to install.

    For our example, we selected **Custom**.

11. Click **Next**. If you choose Custom, the Product selection window will be shown. See Figure 5-5.



Select the features for "IBM WebSphere Application Server - Express v5.0" you would like install:

□ Application Server - Express v5.0 - 68.0 MB

□ Studio Site Developer v5.0 - 552.0 MB

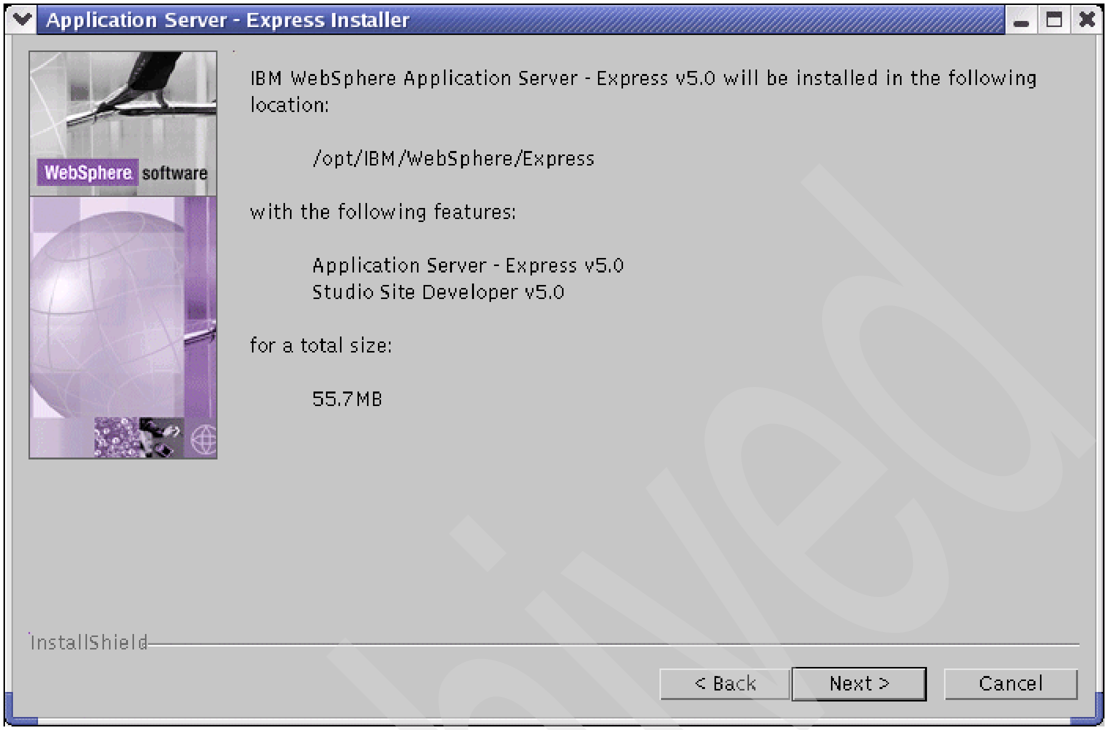
Description:

Installs the application server runtime for hosting Java(TM) technology-based Web applications.

Figure 5-5 Product selection window

This window allows you to select what products you wish to install. Typically for a developer workstation you would install only Studio Site Developer. For a development server, you would install both products; and for test or production servers, you would install only Application Server - Express remote server.

12. Select the products you wish to install.
13. Click **Next**. The installation options window is displayed. See Figure 5-6.



IBM WebSphere Application Server - Express v5.0 will be installed in the following location:

/opt/IBM/WebSphere/Express

with the following features:

Application Server - Express v5.0
Studio Site Developer v5.0

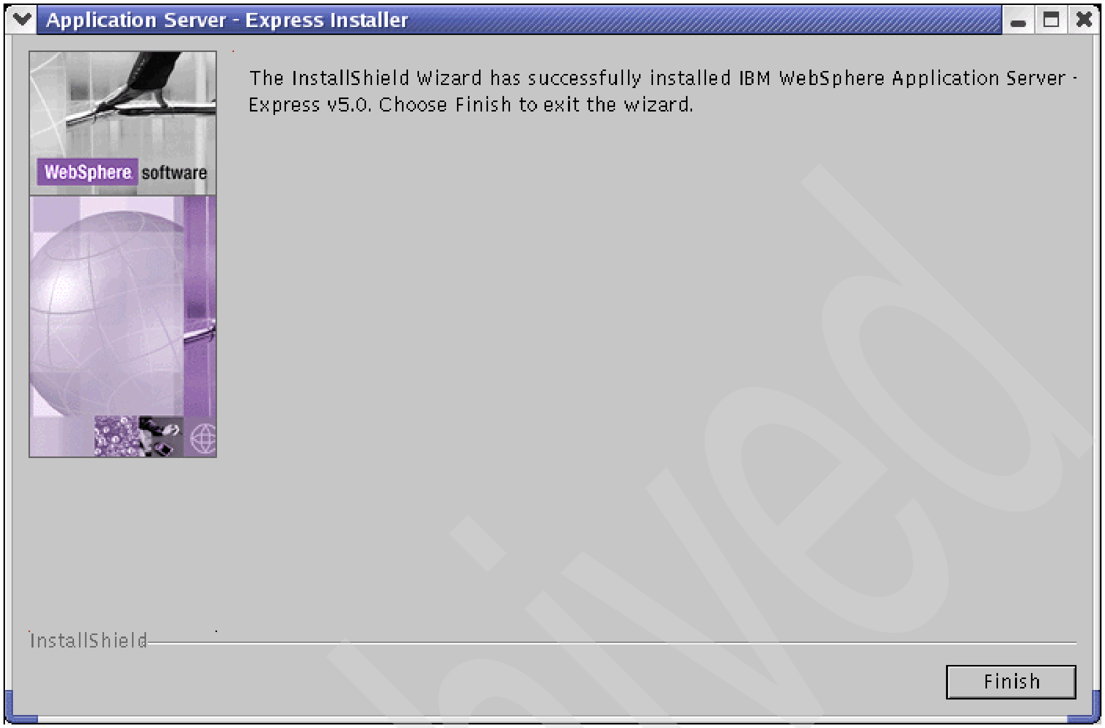
for a total size:

55.7MB

Figure 5-6 Installation options window

The installation options window shows what you are going to install as well as the directory where the products are going to be installed. You can review your install options here. You can go back and change your options if you are not happy by clicking the **Back** button.

14. Click **Next** to begin the installation. Once the installation is complete, you will see a window similar to Figure 5-7.



Application Server - Express Installer

The InstallShield Wizard has successfully installed IBM WebSphere Application Server - Express v5.0. Choose Finish to exit the wizard.

Figure 5-7 Installation complete window

15. Click **Finish.** This will close the installation program. The full installation takes between 10-30 minutes to complete, depending on your hardware configuration.

The Application Server is installed in the /opt/IBM/WebSphere/Express directory, the Studio Site Developer in the /opt/IBM/WebSphereStudio directory, and the IBM Agent Controller in the /opt/IBMRAC directory.

Review the log file for any errors reported during the installation. The log file is located in the /opt/IBM/WebSphere/Express directory and is called expressInstallLog.

Start and stop the IBM Agent Controller

The install will start the IBM Agent Controller to enable communication between Studio Site Developer and the Application Server. To start the IBM Agent Controller manually, complete the following steps:

1. Change to the /opt/IBMRAC/bin directory.
2. Type the following command:

   ```
   ./RAStart.sh
   ```

   This will check if the IBM Agent Controller is currently running and will start it if it is not running.

   To start from the Start menu, select **Start -> Extras -> Other -> Start Remote Agent Controller** on the X-Windows toolbar.

To stop the IBM Agent Controller, do the following:

1. Change to the /opt/IBMRAC/bin directory
2. Type the following command:

   ```
   ./RAStop.sh
   ```

Start the Studio Site Developer

To start the Studio Site Developer, type the following command:

```
wasexpress
```

To start from the Start menu, select **Start -> Extras -> Other -> Application Server - Express v5.0 Site Developer** on the X-Windows toolbar.

5.3 Installing from the downloadable images

This section details the installation from the downloadable images.

5.3.1 Before you install

Before you install the product, check the following. The download image of the Linux CD for WebSphere Application Server - Express is split into several parts to reduce download size. Each part is a tape archive (.tar) file and all the parts are required.

To install from these images, follow these steps:

1. Download each part to the same directory.
2. Unpack the files using the following command:

   ```
   tar -zxvf downloadpart.tar.gz
   ```

3. After you have unpacked all the parts, you are finished with the downloaded parts and may remove them to save disk space.
4. The unpacked files are the same as on the WebSphere Application Server - Express CD-ROM. Follow the instructions above on installing from CD-ROM to complete the install. You need to navigate to the directory where you unpacked the files and run the launchpad.sh program.

5.4 Verifying the installation

After you have installed WebSphere Application Server - Express, you should have the following directory structure:

- /opt/IBM/WebSphere/Express/_uninst

  Contains three files for uninstalling the product.

- /opt/IBM/WebSphere/Express/AppServer

  Contains approximately 1100 files. These are the Application Server files.

- /opt/IBM/WebSphere/Express/Express

  Contains approximately 380 files. These are miscellaneous files for databases and scripts.

- /opt/IBMRAC

  Contains approximately 65 files. These are the IBM Agent Controller files.

- /opt/IBM/WebSphereStudio

  Contains approximately 21,500 files. These are the Studio Site Developer files.

- /opt/IBM/WebSphere/Express/readme

  Contains the Readme, *Installation Guide* and the *Getting Started* PDF documents

If any of these directories contain a substantially different number of files, then you should suspect a problem during the installation. Look at the /opt/IBM/WebSphere/Express/expressInstallLog.txt file.

5.4.1 Verifying Agent Controller installation

Open the /opt/IBM/WebSphere/Express/racInstallLog.txt file. Check the install completion status at the bottom of the file. If the install was successful, the IBM Agent Controller should have started. You can check this by running the following command:

```
/opt/IBMRAC/bin/RAStart.sh
```

This will check if the agent has started. If not it will start the IBM Agent Controller. If you suspect a problem, look at the /opt/IBMRAC/service/servicelog.log file. If the install did not complete successfully, to determine the error, check the /opt/IBM/WebSphere/Express/racInstallLog.txt file.

5.4.2 Verifying Application Server installation

You can verify the basic operation of your Application Server installation by running the following command:

```
/opt/IBM/WebSphere/Express/bin/testServer.sh
```

This script will step through starting and stopping the server using a default configuration. This script can only be used to test the server installation. You will use Studio Site Developer for configuration, and starting and stopping your server for normal use.

5.4.3 Verifying Site Developer installation

Look at the /opt/IBM/WebSphere/Express/wssdInstallLog.txt file. Check the bottom of the file for the install completion status. If the installation completed successfully, start Studio Site Developer by running the following command:

```
/opt/IBM/WebSphereStudio/wasexpress
```

or select **Start -> Extras -> Other -> Application Server - Express v5.0 Site Developer** on the X-windows toolbar.

If the install did not complete successfully, look at the wssdInstallLog.txt file to determine the reason.

5.5 Using the silent install

The installation program for WebSphere Application Server - Express can be run in silent mode. This allows you to bundle the product with your own product. To install silently, the **`-silent`** command-line option is used. During this kind of install, you will not see the wizard panels after you invoke the command. An example of this command is:

```
setuplinux -is:silent -silent
```

You can use a response file for further options for use during the installation process. See the example response file on the installation CD in the util directory for a template. An example using a response file is as follows:

```
setuplinux -is:silent -options responsefile
```

5.6 Uninstalling WebSphere Application Server - Express

To uninstall WebSphere Application Server - Express, follow these steps:

1. Run the command:

   ```
   /opt/IBM/WebSphere/_uninst/uninstaller
   ```

2. Clean up the file system by removing the remaining files in the installation directory and remove these directories.

5.7 Install WebSphere Application Server - Express for Windows from CD

This section provides detailed instructions for installing and verifying installation of WebSphere Application Server - Express, V5 for Windows 2000 from CD.

5.7.1 Install WebSphere Application Server - Express

To install WebSphere Application Server V5.0 - Express, follow these steps.

1. Insert the CD into your CD-ROM drive.
2. Log in as an administrator whose Windows ID does not contain DBCS characters.
3. If autorun in enabled on your system, the launchpad program will automatically come up. If autorun is disabled on your system, run launchpad.exe from the root of the CD drive.

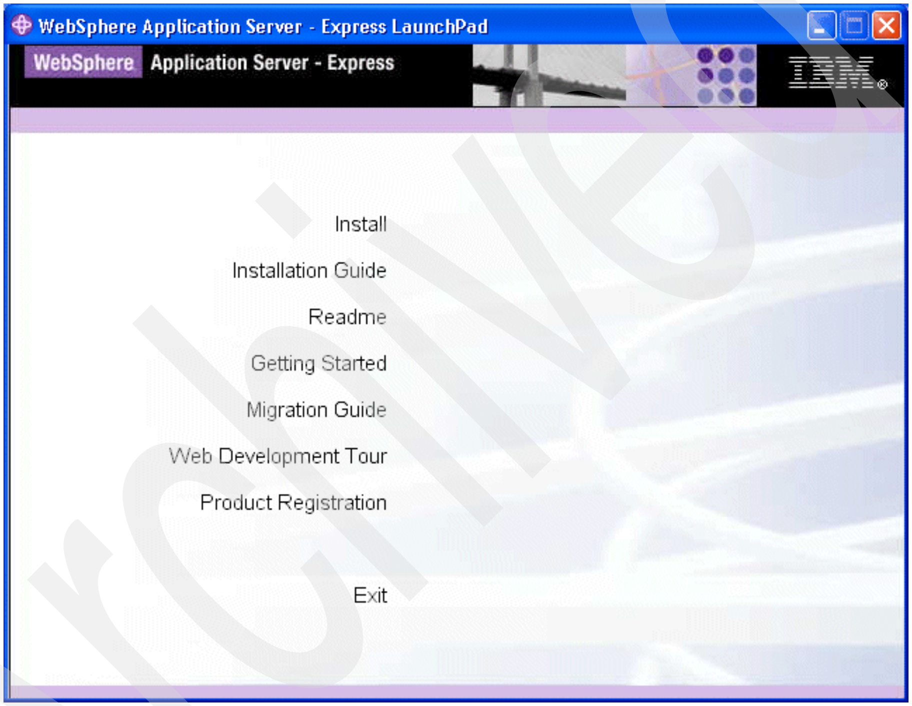
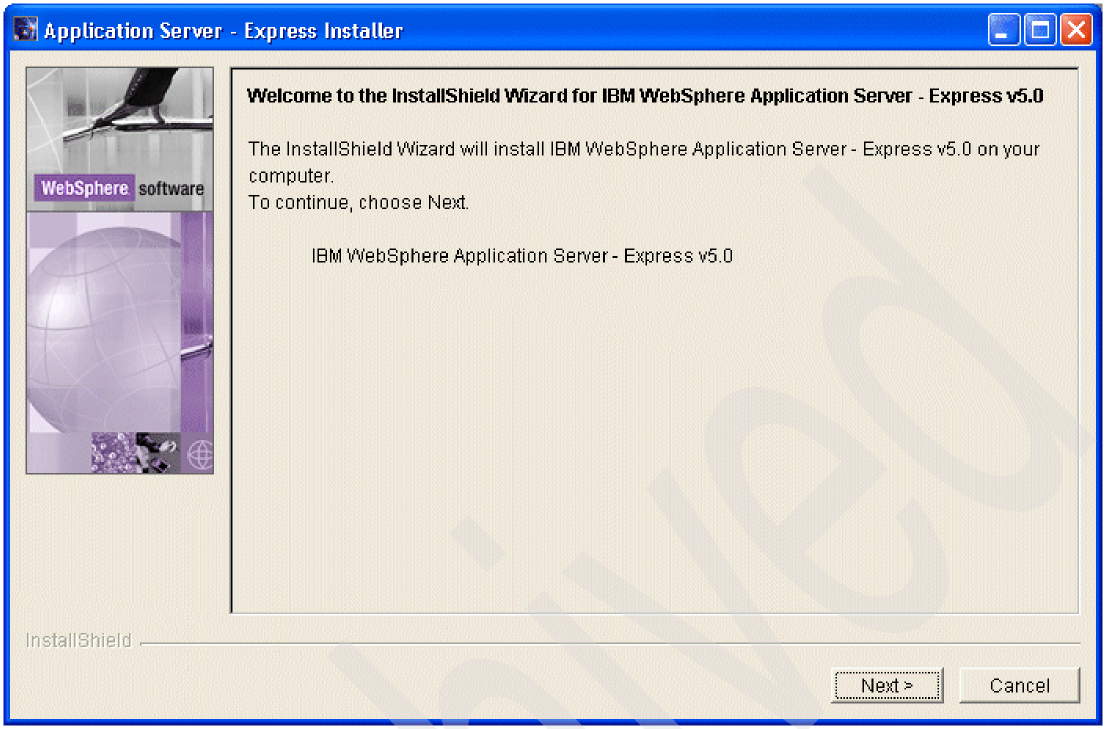


Figure 5-8 WebSphere Application Server - Express installation launcher

4. The WebSphere Application Server - Express installation launcher window contains several links shown in Figure 5-8. Select **Install** to begin installing the product.


Application Server - Express Installer

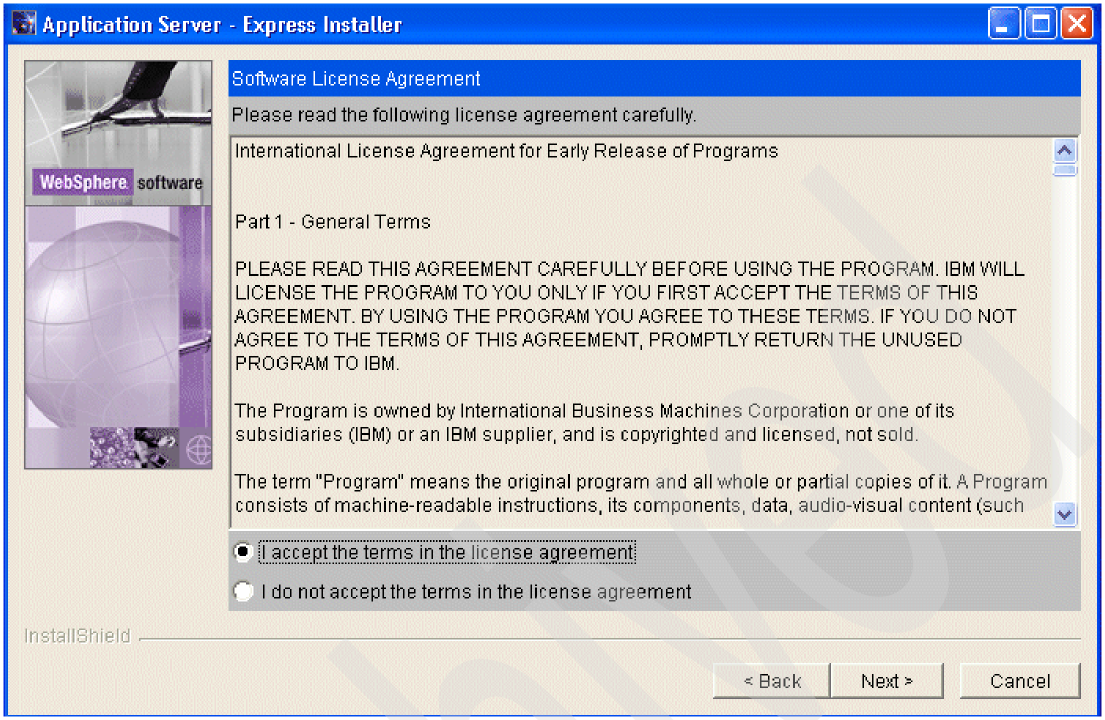
Welcome to the InstallShield Wizard for IBM WebSphere Application Server - Express v5.0

The InstallShield Wizard will install IBM WebSphere Application Server - Express v5.0 on your computer.
To continue, choose Next.

IBM WebSphere Application Server - Express v5.0

Figure 5-9 Express Installer

5. Click **Next** in the Express Installer Window (see Figure 5-9).



Software License Agreement

Please read the following license agreement carefully.

International License Agreement for Early Release of Programs

Part 1 - General Terms

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THE PROGRAM. IBM WILL LICENSE THE PROGRAM TO YOU ONLY IF YOU FIRST ACCEPT THE TERMS OF THIS AGREEMENT. BY USING THE PROGRAM YOU AGREE TO THESE TERMS. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, PROMPTLY RETURN THE UNUSED PROGRAM TO IBM.

The Program is owned by International Business Machines Corporation or one of its subsidiaries (IBM) or an IBM supplier, and is copyrighted and licensed, not sold.

The term "Program" means the original program and all whole or partial copies of it. A Program consists of machine-readable instructions, its components, data, audio-visual content (such

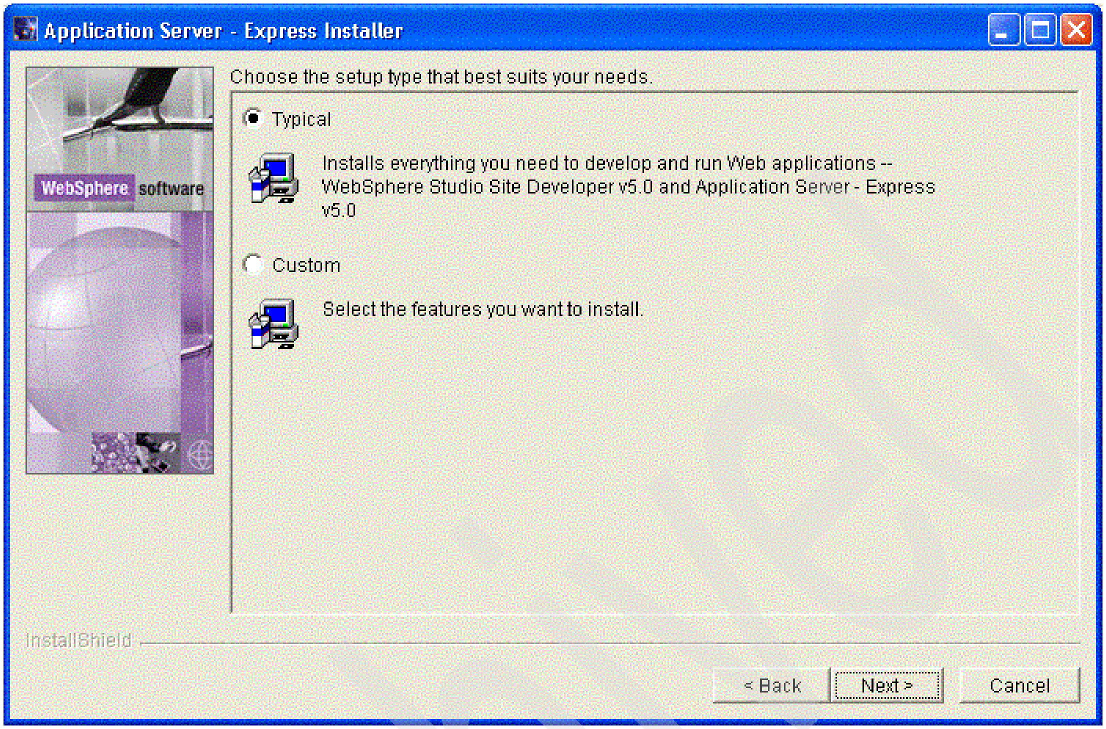
● I accept the terms in the license agreement

○ I do not accept the terms in the license agreement

Figure 5-10 Software License Agreement

6. Select **I accept the terms in the license agreement**, click **Next** (see Figure 5-10).

   A Typical install would include WebSphere Application Server - Express and WebSphere Studio Site Developer. The Custom option gives you the choice of the WebSphere Application Server - Express and/or WebSphere Studio Site Developer (see Figure 5-11).



Application Server - Express Installer

Choose the setup type that best suits your needs.

● Typical

Installs everything you need to develop and run Web applications -- WebSphere Studio Site Developer v5.0 and Application Server - Express v5.0

○ Custom

Select the features you want to install.

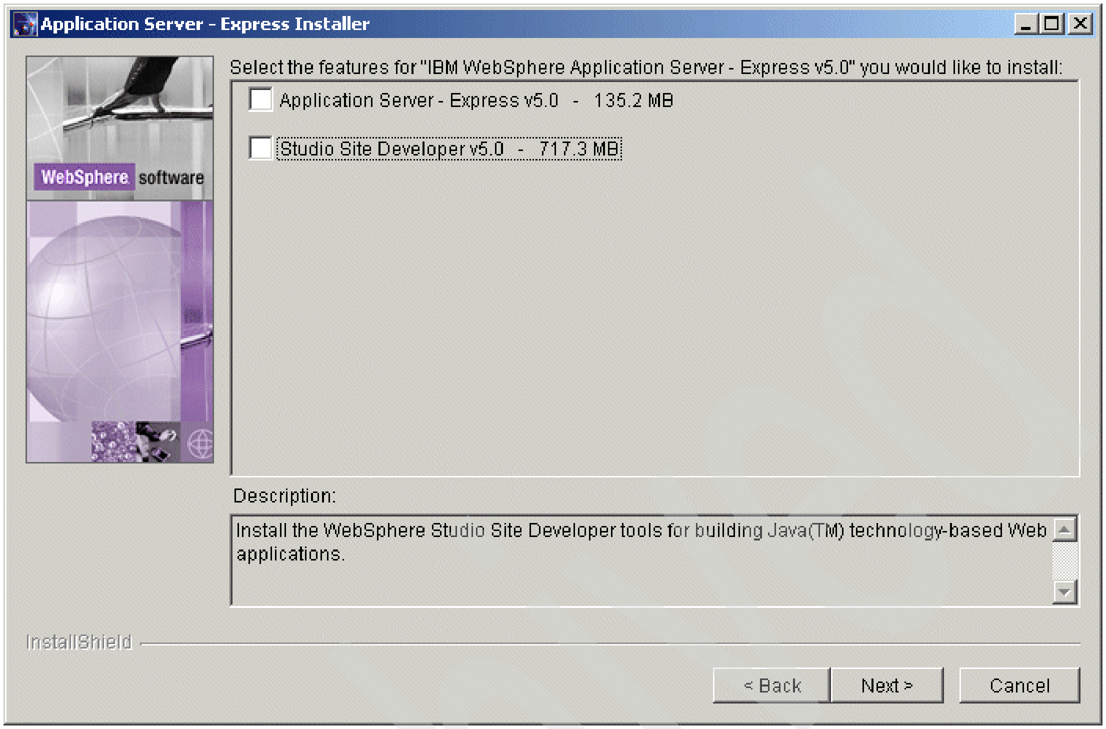
Figure 5-11 Setup type

7. Select either **Typical** or **Custom**.

   If you are installing a development server only, select Typical. If you are installing a test or production server, select Custom. In our example, we selected **Custom**.

8. Click **Next**.

   If you choose Custom, the product selection window appears (see Figure 5-12).



Select the features for "IBM WebSphere Application Server - Express v5.0" you would like to install:

☐ Application Server - Express v5.0 - 135.2 MB

☐ Studio Site Developer v5.0 - 717.3 MB

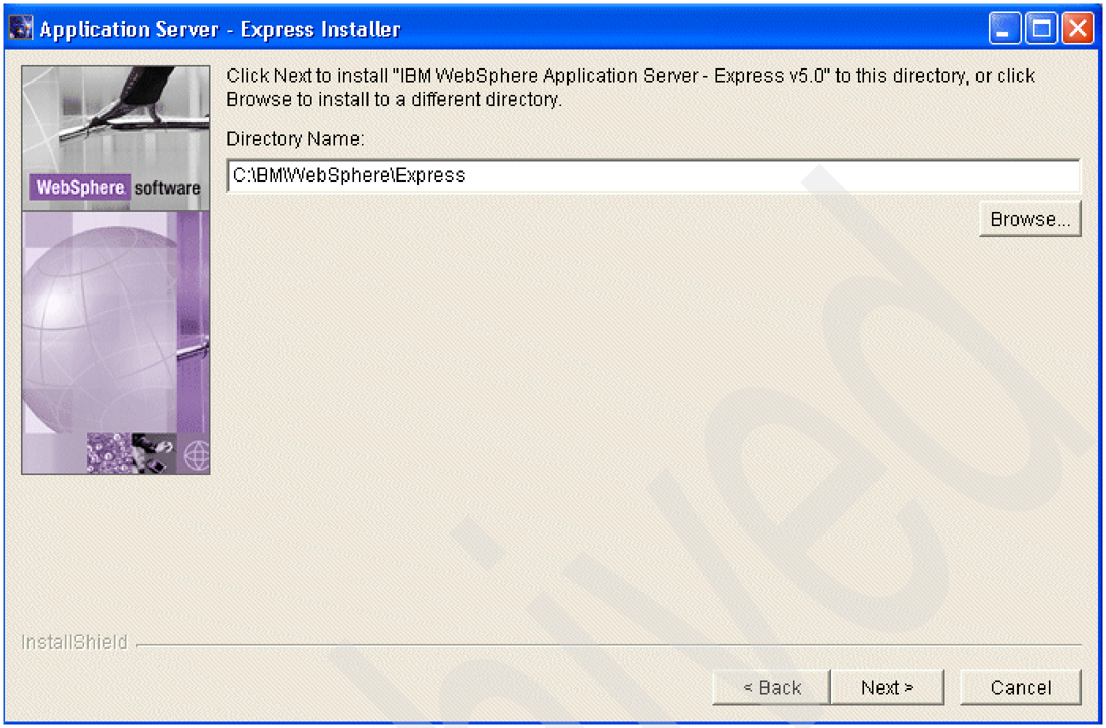
Description:

Install the WebSphere Studio Site Developer tools for building Java(TM) technology-based Web applications.

Figure 5-12 Product Selection Window

The Product Selection Menu (see Figure 5-12) will allow you the choice of selecting the Application Server and Studio Site Developer. Typically, for the developer's workstation, you would choose Studio Site Developer only. For the development server, you would choose Application Server and Studio Site Developer. The test and production servers would only have the Application Server installed. For our example, we selected both.

9. Select the product you wish to have installed. Click **Next**. You will see a window similar to Figure 5-13.



Application Server - Express Installer

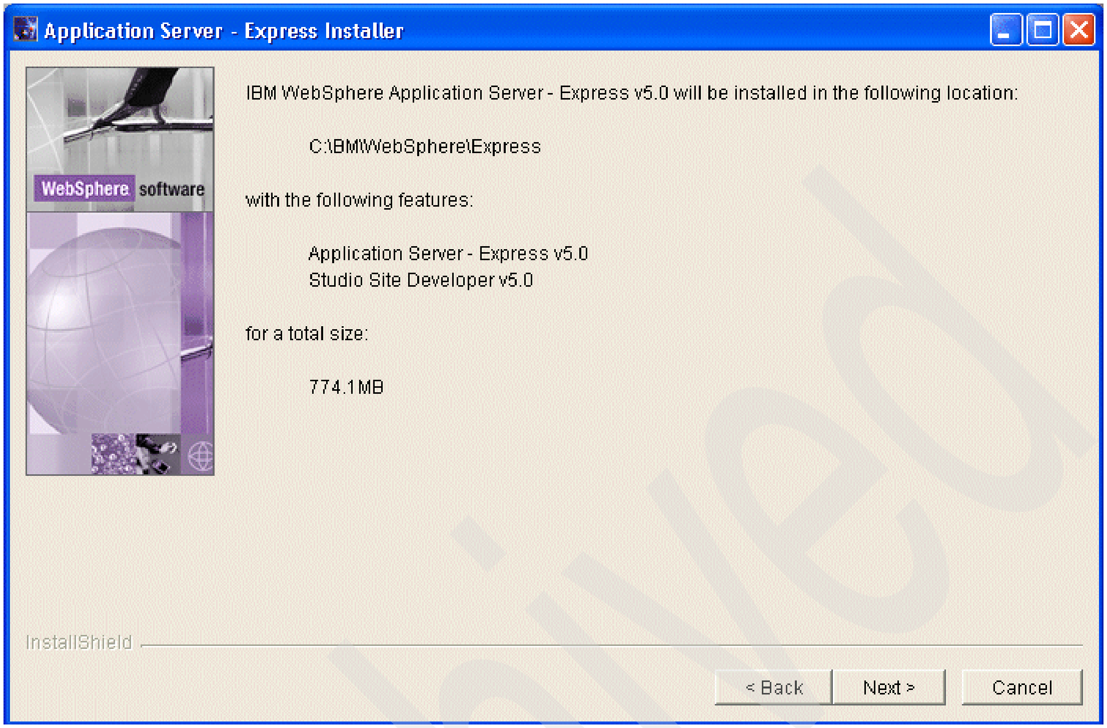
Click Next to install "IBM WebSphere Application Server - Express v5.0" to this directory, or click Browse to install to a different directory.

Directory Name: C:\IBM\WebSphere\Express

Figure 5-13 Application Installation Directory window

10.Click **Browse** or you may type in the directory into which you want WebSphere Application Server to be installed (see Figure 5-13).

11.Click **Next**. You will see a window similar to Figure 5-14.



IBM WebSphere Application Server - Express v5.0 will be installed in the following location:

C:\IBM\WebSphere\Express

with the following features:

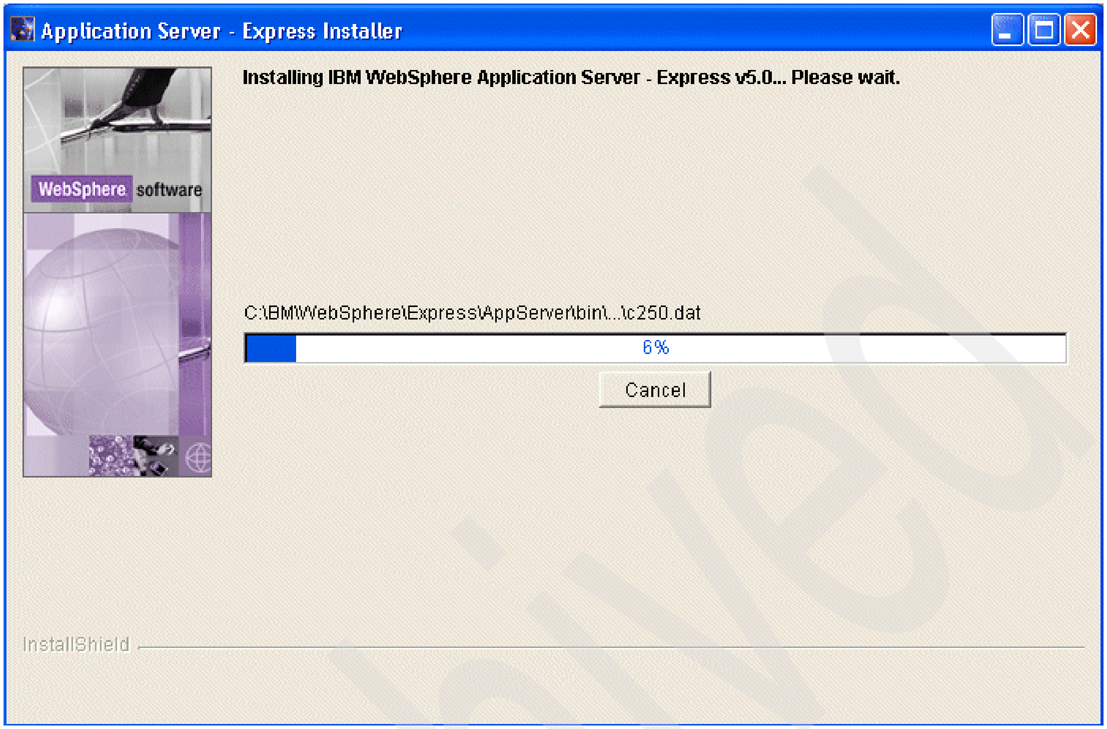
Application Server - Express v5.0
Studio Site Developer v5.0

for a total size:

774.1MB

Figure 5-14 Installation confirmation window

12. In the installation confirmation window, verify that all information shown is correct. Click **Next**. You will see a window similar to Figure 5-15.



Application Server - Express Installer

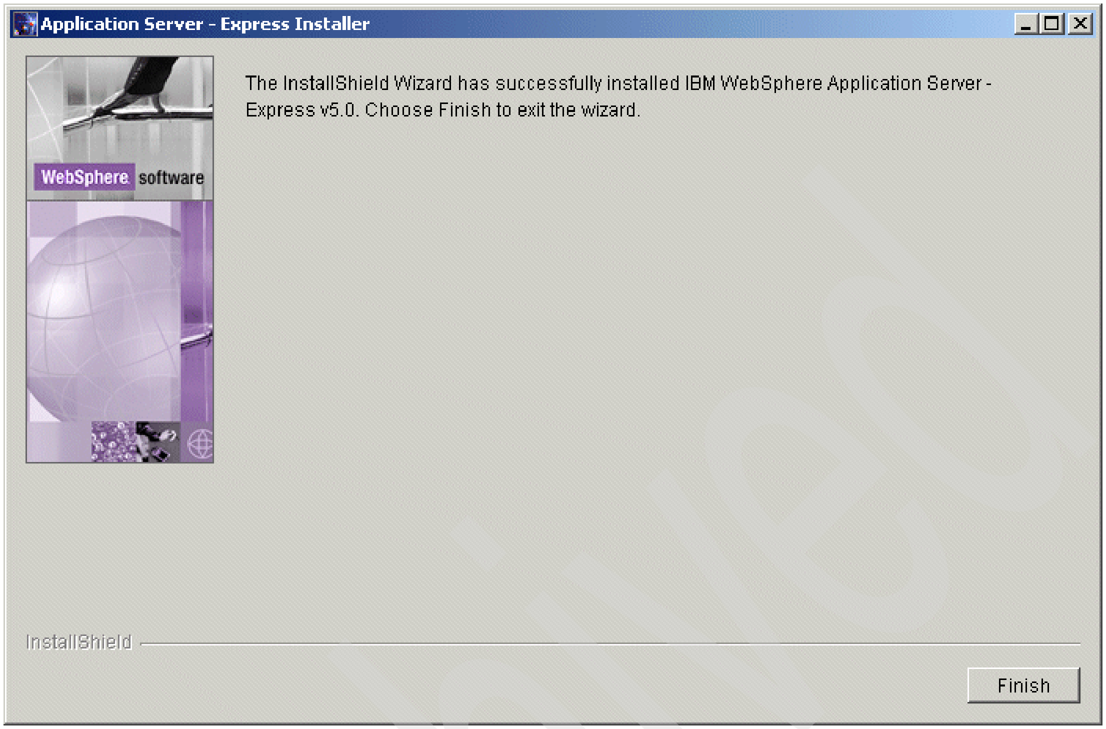
Installing IBM WebSphere Application Server - Express v5.0... Please wait.

C:\IBM\WebSphere\Express\AppServer\bin\...\c250.dat

6%

Figure 5-15 Installation Progress window

The Progress window (Figure 5-15) will indicate the progress of the installation. The progress bar is non-linear. If you have chosen to install Studio Site Developer, the progress bar will reach 99% before Studio Site Developer begins installing. You can click **Cancel** at any time to stop the installation.



Application Server - Express Installer

The InstallShield Wizard has successfully installed IBM WebSphere Application Server - Express v5.0. Choose Finish to exit the wizard.

Figure 5-16 Installation complete window

13. Click **Finish**.

    This will close the installation program. The installation will take 10-30 minutes depending on the applications being installed and the system configuration.

5.8 Installing WebSphere Application Server - Express from downloadable images

In this section we provide detailed instructions for installing and verifying installation of WebSphere Application Server - Express, V5 for Windows 2000 from downloadable images.

5.8.1 Before you install

Before you install the product, check the following:

- In addition to the space required to install the product, you must have at least 50 MB free on your Windows system drive, and your environment variable TEMP or TMP must point to a valid temporary directory with at least 10 MB free. The downloadable images require approximately 600 MB plus another 600 MB to unpack images.
- Do not install this version of WebSphere Application Server V5.0 - Express over an existing version. You must uninstall the previous version before installing this one.
- You must not have the HTTP server of WebSphere Application Server, Version 3.5 running.
- If you have previous copy of IBM Agent Controller on your system, you should uninstall it before proceeding.
- If you have a previous copy of Studio Site Developer on your system, you should uninstall it before proceeding.

5.8.2 Installing WebSphere Application Server - Express

The image of Windows CD for WebSphere Application Server - Express is split into several parts to reduce the download size. Each part is a self-extracting archive and all parts are required.

To install the electronic images of WebSphere Application Server - Express, follow these steps:

1. Click each part and save all images to the same Windows directory. You will need approximately 1.2 GB of disk space to download and unpack the images.
2. Navigate to the Windows directory where each part was saved. Select the same directory into which to unpack each part. You must unpack all parts to the have a valid image.
3. After you have unpacked all the parts, you are finished with the downloaded parts and may remove them to save disk space.
4. The unpacked files are the same as in the WebSphere Application Server - Express CD. See 5.7.1, “Install WebSphere Application Server - Express” on page 40.

5.9 Installing WebSphere Application Server - Express in silent mode

The WebSphere Application Server - Express installation can be run silently, if you wish to bundle the product with your own. During a silent installation, you will not see wizard panels after you invoke the **`install`** command.

5.9.1 Installing WebSphere in silent mode

To run the installation silently, accepting all the defaults, use the following command:

```
setupwin32.exe -is:silent -silent
```

If you want to change the default Express installation location, use the following command:

```
setupwin32.exe -is:silent -P wasBean.installLocation=c:\<yourdirectory>
```

You can also use a response file to set additional options for silent install. See the example response file on the installation CD for a template. To run the a silent install using a response file, use the following command:

```
setupwin32.exe -is:silent -options responsefile
```

5.10 Verifying your installation

After installing WebSphere Application Server - Express, you should have the following directory structure under the <Expressinstallpath> that you selected during the install.

_uninst	Contains three files for uninstalling Express
AppServer	Application Server files - approximately 950 files
RAC	IBM Agent Controller files - approximately 40 files
SiteDeveloper	WebSphere Studio Site Developer file - approximately 22,500 files
Tour	Flash tutorial for using Studio Site Developer - approximately 350 files
readme	Contains the readme, installation guide, and the Getting Started PDF

If these directories contain substantially different numbers of files, look at the expressInstallLog.txt file in <Expressinstallpath> to diagnose the problem.

5.10.1 Verifying Site Developer installation

To verify installation, go to the bottom of the wssdInstallLog.txt file in the <Expressinstallpath>. It should have an `Installation completed successfully` status message. If the Studio Site Developer installed successfully, start Site Developer by clicking **Start -> Programs -> WebSphere Application Server - Express V5.0 -> Studio Site Developer**.

If the installation did not complete successfully, look through the wssdInstallLog.txt to determine the reason.

5.10.2 Verifying Agent Controller installation

To verify installation, go to the bottom of the racInstallLog.txt file in the <Expressinstallpath>. It should have the `Installation completed successfully` status message. If the installation completed successfully, then check that the IBM Agent Controller is running. The IBM Agent Controller is always installed as a Windows service and is automatically started. The IBM Agent is the information broker between the Site Developer and the Application Server.

If the installation did not complete successfully, look through the racInstallLog.txt to determine the reason.

5.10.3 Verifying Application Server installation

Verification of the Application Server can be performed by running:

```
<Expressinstallpath>\Express\bin\testServer.bat file
```

The batch file will step through starting and stopping the Application Server using the default configuration. The batch file is only used for testing the server installation. Studio Site Developer will be used for starting, stopping, and configuring the Application Server during normal use.

5.11 Uninstalling WebSphere Application Server - Express

Use the Add/Remove Program application to remove WebSphere Application Server V5.0 - Express

1. Open Windows Control Panel and then open Add/Remove Programs.
2. Select **IBM WebSphere Application Server - Express V5.0** and click **Change/Remove** to start the uninstall.
3. You will notice that there are entries in the Add/Remove Programs for IBM Agent Controller and for IBM WebSphere Studio Site Developer (Express). You should not remove these products directly. They will be removed when you select the IBM WebSphere Application Server - Express V5.0 entry in Add/Remove Programs.
4. Follow the uninstall wizard to uninstall the product.
5. To completely clean up any remaining files, go to the installation directory and remove the remaining folders.

6

Build your WebSphere Application Server - Express

Before we start building a new application or start working on an existing sample of applications provided by the product, it is mandatory to understand how to configure the server post installation. In this chapter, we show you how to create various types of application servers, and specific scenarios which would require any of these three options:

- Express Remote Server
- Express Test Environment
- Remote Server Attach

WebSphere Application Server - Express does not have a separate administrative console and can be configured from the WebSphere Studio front-end.

6.1 New Application Server creation

In this section, we begin the necessary steps to set up and configure a new application server.

If using Linux, use these steps to start WebSphere Studio:

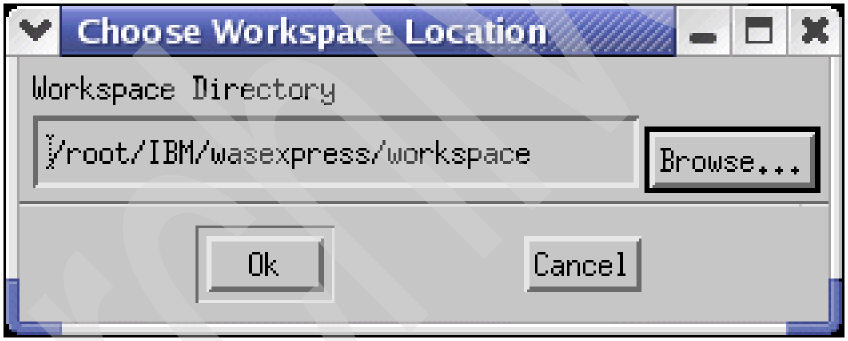
1. From a Linux window prompt, type `wasexpress` to start the WebSphere Studio console.

or

2. Go to the Gnome Menu (we used Gnome in our example) in the Red Hat Linux X-Windows toolbar and click:

   Start -> Extras -> Other -> Application Server - Express V5 Site Developer

3. You will be prompted to choose a Workspace Location. We did not change the default. If you want to change the location, you can do so, and then click **OK**. See Figure 6-1.



Choose Workspace Location

Workspace Directory

/root/IBM/wasexpress/workspace

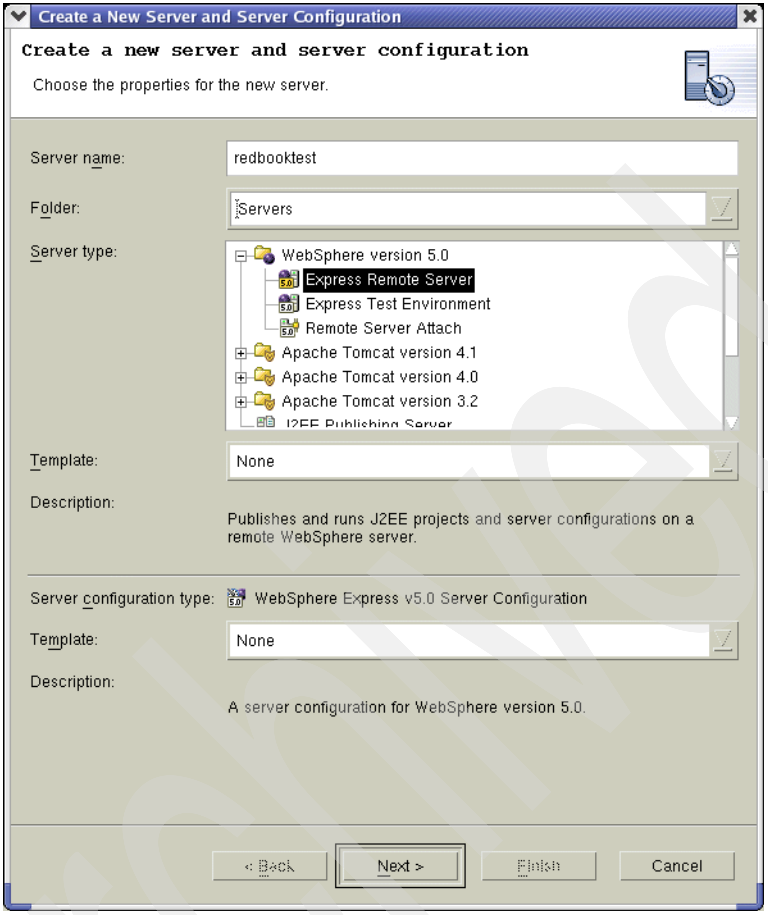
Figure 6-1 Choose Workspace Location

If using Windows, use these steps to start WebSphere Studio:

1. From the Windows desktop, click **Start -> Programs -> WebSphere Application Server - Express V5**.
2. Click **Studio Site Developer**.

The following steps are common for both platforms:

3. Once the WebSphere Studio Site Developer console opens, use your mouse to right-click **Servers** and select **New -> Server and Server configuration**. or go to **File -> New -> Server and Server Configuration.** You will see a window similar to Figure 6-2.



Create a New Server and Server Configuration

Create a new server and server configuration

Choose the properties for the new server.

Server name: redbooktest

Folder: Servers

Server type:

- WebSphere version 5.0
  - Express Remote Server
  - Express Test Environment
  - Remote Server Attach
- Apache Tomcat version 4.1
- Apache Tomcat version 4.0
- Apache Tomcat version 3.2
- J2EE Publishing Server

Template: None

Description: Publishes and runs J2EE projects and server configurations on a remote WebSphere server.

Server configuration type: WebSphere Express v5.0 Server Configuration

Template: None

Description: A server configuration for WebSphere version 5.0.

Figure 6-2 Create a new server and server configuration window

4. At this window, you will have to enter the server name to invoke the server. In our example, we used *redbooktest* as our server name. This name has no significance to your domain name, so this can be any name for identifying the server.

5. The server type is determined based on the configuration of your setup.

   a. **Express Remote Server** is chosen if the WebSphere Studio is running on an individual system with the WebSphere Application Server- Express on another system. This would mean that the publishing and the server configurations would be carried out on a remote system only.

   b. **Express Test Environment** is the option to choose if creating a test environment with the Application Server and WebSphere Studio running of the same system.

c. **Remote Server Attach** is chosen if there is already an existing WebSphere Application Server running in debug mode and you would like this WebSphere Studio server to attach itself to the WebSphere Application Server.

d. The other available options in the window are beyond the scope of this book and hence not covered.

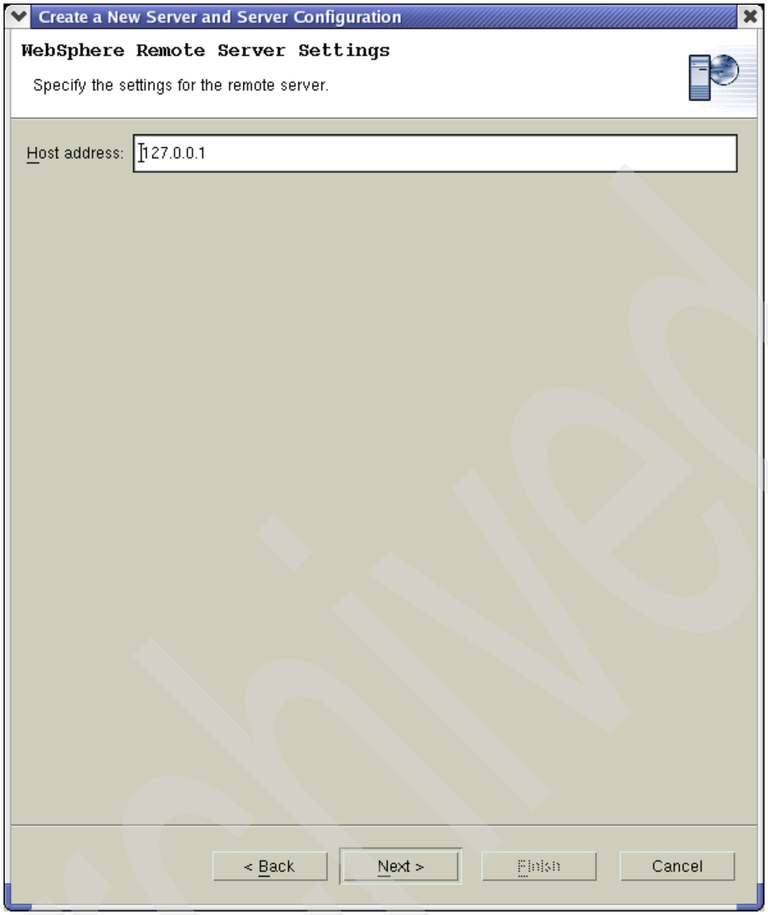
6. Once you choose the option applicable to your particular environment, click **Next**.

In the following sections, we cover the three options to be completed (Express Remote Server, Express Test Environment, and Remote Server Attach) in detail.

6.2 Create new server - Server type: Express Remote Server

There are three types of application servers that can be created and configured. This section will focus on the Express Remote Server Application Server creation. Complete the following instructions:

1. In Figure 6-2, choose **Express Remote Server** and click **Next**. You will be prompted for the Host IP address on which the Application Server - Express is running. See Figure 6-3.


Create a New Server and Server Configuration

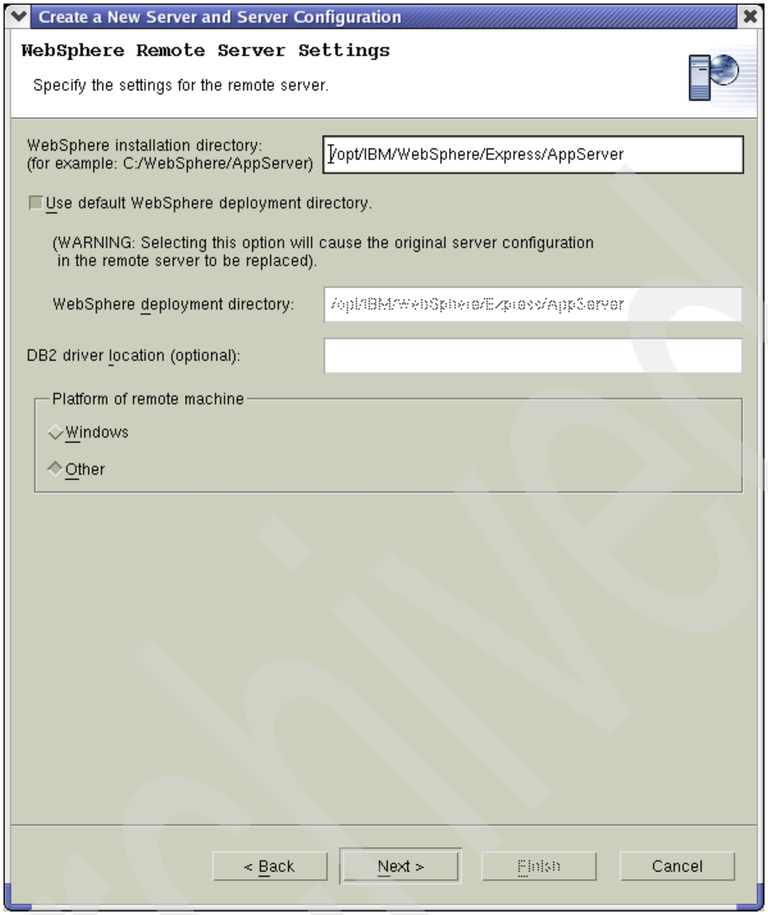
WebSphere Remote Server Settings

Specify the settings for the remote server.

Host address: 127.0.0.1

Figure 6-3 WebSphere Remote Server Settings window

2. In our case, since the application server is running on the same system, we accepted the default 127.0.0.1 as the IP address. Here it is necessary to mention the IP address on which the WebSphere Application Server - Express will run.
3. After entering the IP address of the application server, click **Next**. You will see a window similar to Figure 6-4.



Create a New Server and Server Configuration

WebSphere Remote Server Settings

Specify the settings for the remote server.

WebSphere installation directory:
(for example: C:/WebSphere/AppServer): /opt/IBM/WebSphere/Express/AppServer

□ Use default WebSphere deployment directory.

(WARNING: Selecting this option will cause the original server configuration
in the remote server to be replaced).

WebSphere deployment directory: /opt/IBM/WebSphere/Express/AppServer

DB2 driver location (optional):

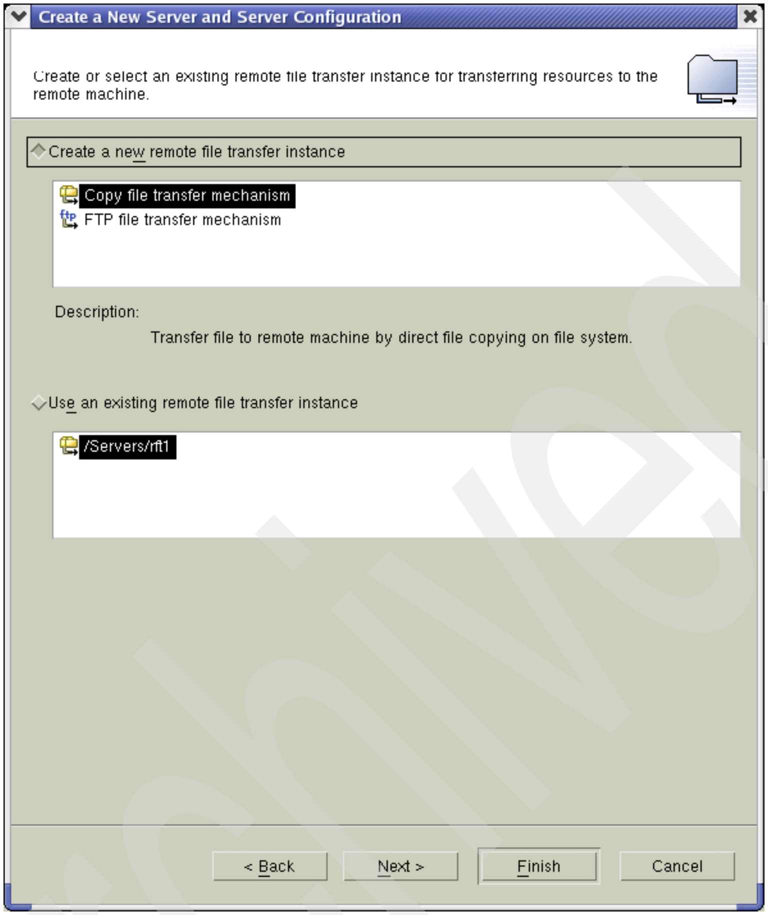
Platform of remote machine

○ Windows

● Other

Figure 6-4 Setting changes for new server configuration

4. Accept the defaults in this window. Click **Next** and you will be prompted for the method of transferring the files (see Figure 6-5).



Create a New Server and Server Configuration

Create or select an existing remote file transfer instance for transferring resources to the remote machine.

◆ Create a new remote file transfer instance

- Copy file transfer mechanism
- FTP file transfer mechanism

Description:

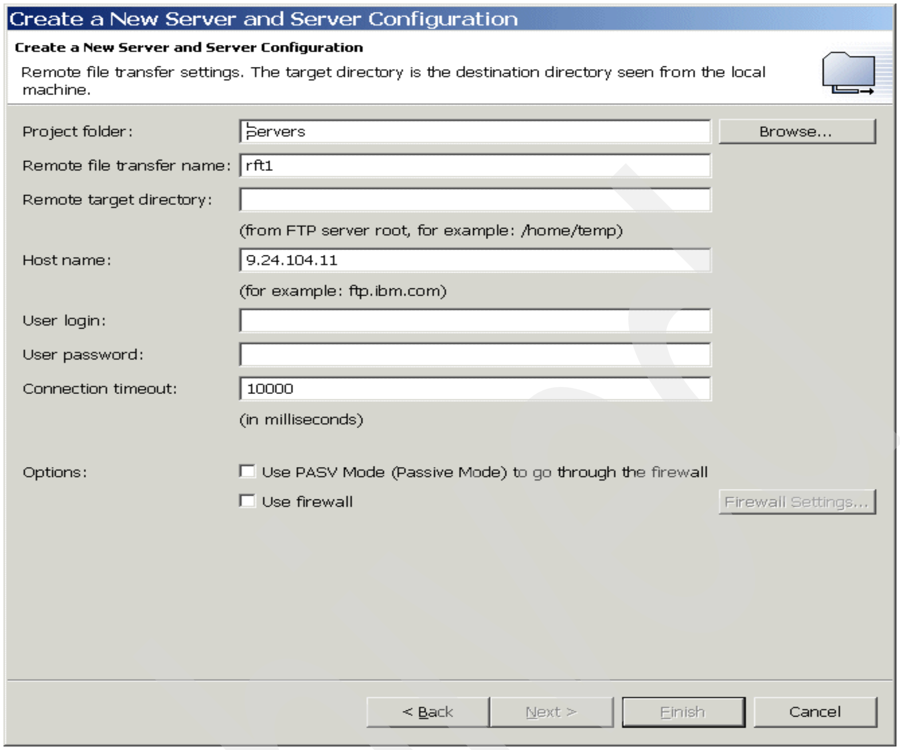
Transfer file to remote machine by direct file copying on file system.

◇ Use an existing remote file transfer instance

- /Servers/rft1

Figure 6-5   File transfer methodology window

5. At this window, if you choose FTP file transfer mechanism as the remote file transfer settings, you will see a window similar to Figure 6-6.



Create a New Server and Server Configuration

Create a New Server and Server Configuration

Remote file transfer settings. The target directory is the destination directory seen from the local machine.

Project folder: Servers

Remote file transfer name: rft1

Remote target directory:

(from FTP server root, for example: /home/temp)

Host name: 9.24.104.11

(for example: ftp.ibm.com)

User login:

User password:

Connection timeout: 10000

(in milliseconds)

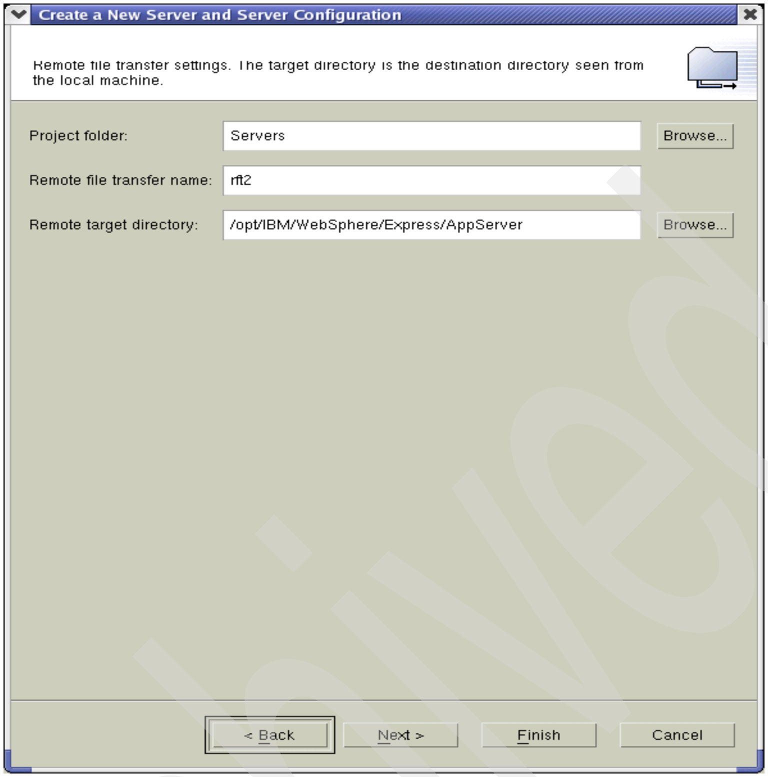
Options:

☐ Use PASV Mode (Passive Mode) to go through the firewall

☐ Use firewall

Figure 6-6 FTP transfer details

6. Type in the requisite FTP details asked for this window (Figure 6-6).
7. For our example, choose **Copy file transfer mechanism**. You will see a window similar to Figure 6-7.



Remote file transfer settings. The target directory is the destination directory seen from the local machine.

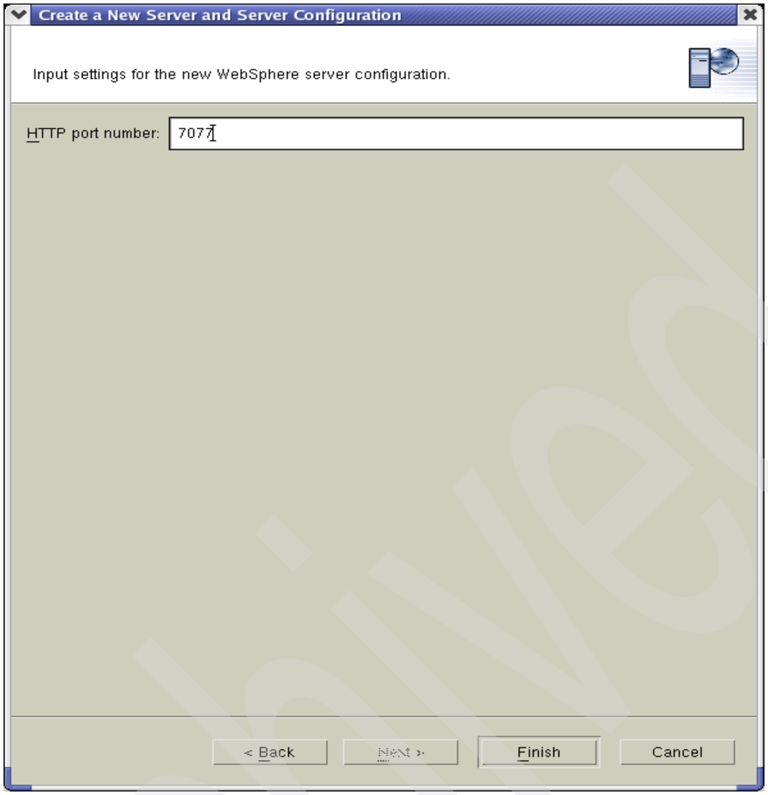
Project folder: Servers

Remote file transfer name: rft2

Remote target directory: /opt/IBM/WebSphere/Express/AppServer

Figure 6-7 Remote file transfer settings window

8. You will be asked to type the Remote target directory and the Remote file transfer name. The project folder should remain as Servers. The Remote target directory should point to the directory in which the WebSphere Application Server is installed on the Remote Server. For our example, since the server was installed in the default directory we did not change any settings on this page. Therefore, accept the defaults and click **Next**. You will see a window similar to Figure 6-8.



Create a New Server and Server Configuration

Input settings for the new WebSphere server configuration.

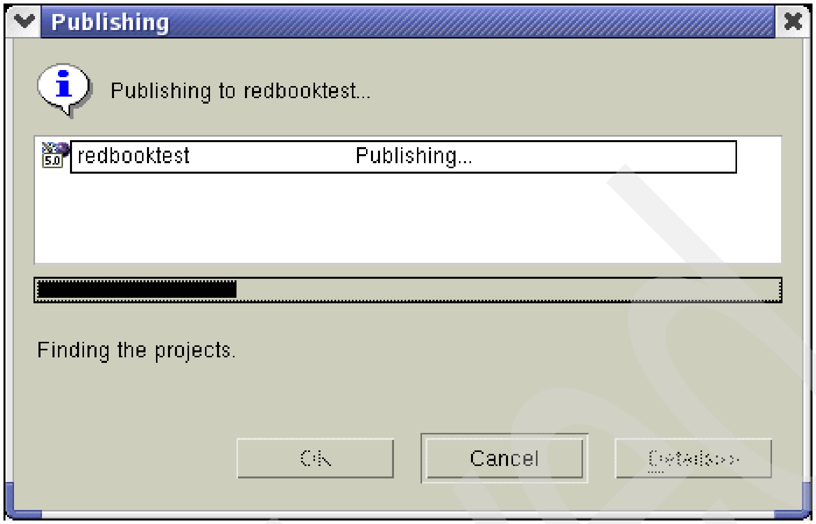
HTTP port number: 7077

Figure 6-8 HTTP port number window

9. You will be prompted to enter the HTTP port on where the WebSphere Application Server will run. This information is critical for the Web server to communicate with the application server. You can either retain the default setting, or enter a new port.

   After entering a new port number or retaining the one that originally existed, click **Finish**.

10. This completes the configuration of the Web server with the option of Express Remote Server as the server type. The application server would then be created and started with the configuration provided. The creation of the server would also enable a fresh window to be open during publishing, which would look similar to Figure 6-9.



Publishing to redbooktest...

redbooktest Publishing...

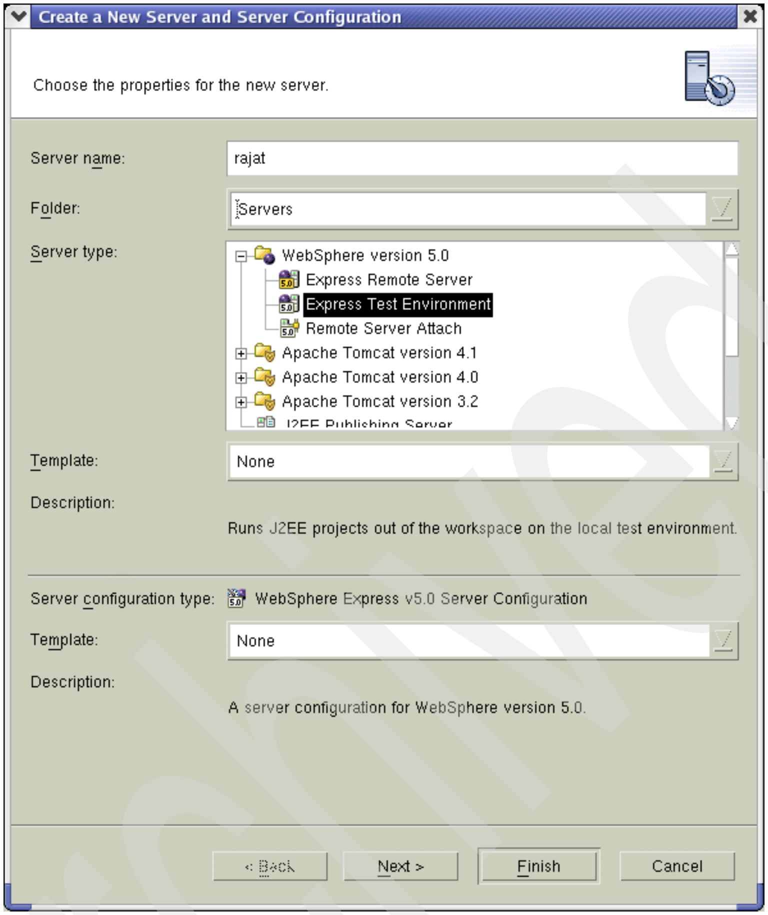
Finding the projects.

Figure 6-9 Server creation and publishing window

6.3 Create new server - Server type: Express Test Environment

In this section we focus on creating the Express Test Environment Application Server.

1. If you were to choose Express Test Environment as the server type, then the number of configuration options reduces drastically. See Figure 6-10.



Create a New Server and Server Configuration

Choose the properties for the new server.

Server name: rajat

Folder: Servers

Server type:

- WebSphere version 5.0
  - Express Remote Server
  - Express Test Environment
  - Remote Server Attach
- Apache Tomcat version 4.1
- Apache Tomcat version 4.0
- Apache Tomcat version 3.2
- J2EE Publishing Server

Template: None

Description:

Runs J2EE projects out of the workspace on the local test environment.

Server configuration type: WebSphere Express v5.0 Server Configuration

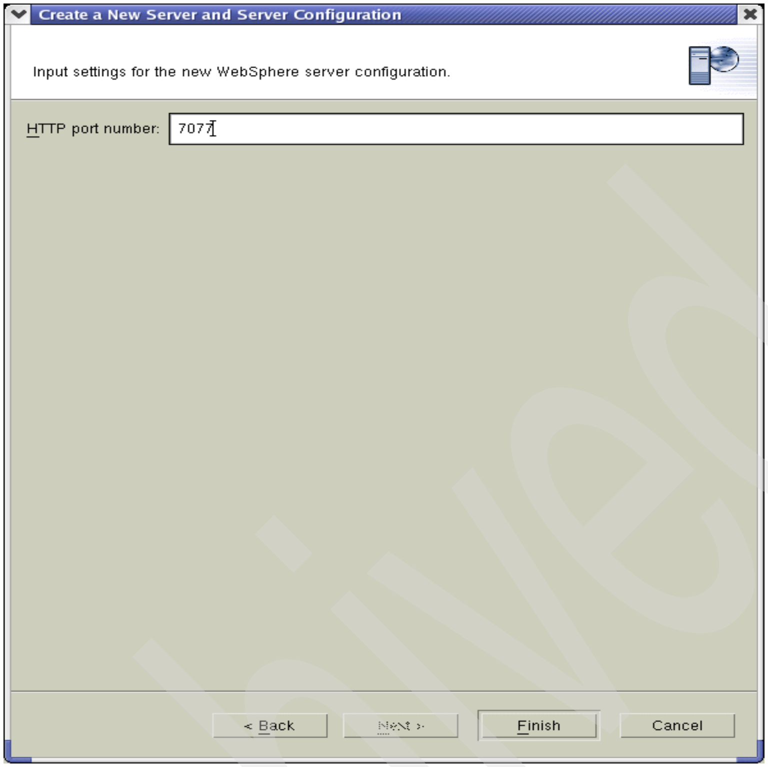
Template: None

Description:

A server configuration for WebSphere version 5.0.

Figure 6-10   Express Test Environment server type window

2. After choosing Express Test Environment as the server type, click **Next**. You will be prompted for the HTTP Port number. See Figure 6-11.



Create a New Server and Server Configuration

Input settings for the new WebSphere server configuration.

HTTP port number: 7077

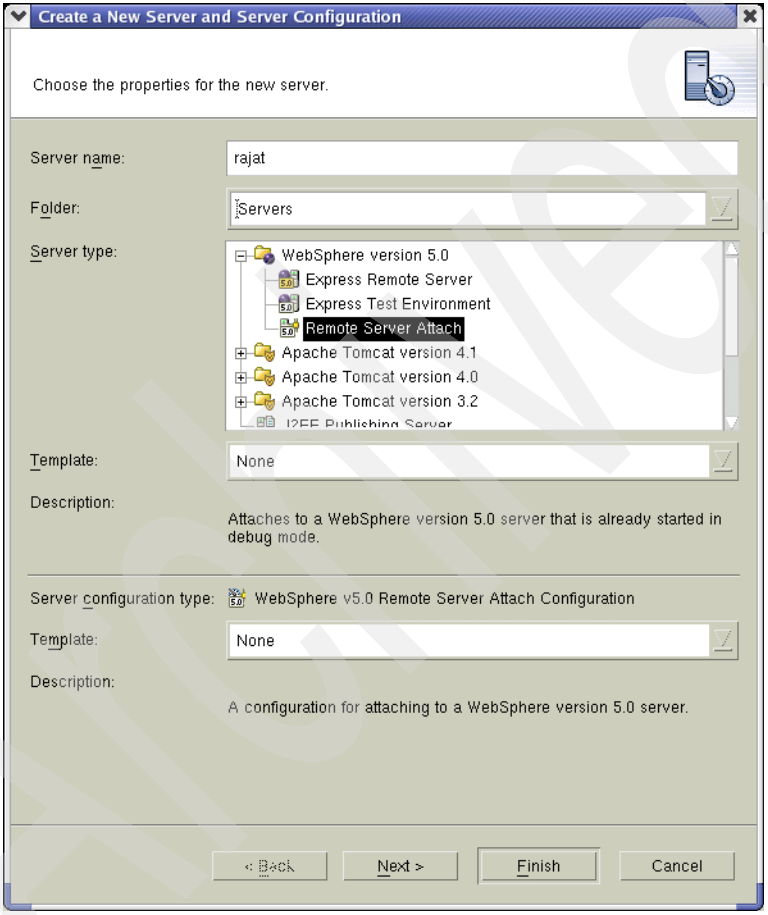
Figure 6-11 Entering HTTP port number

3. Here you can enter the port number of the application server or you can retain the default settings if you do not plan to run the application server on a different port. After defining the HTTP port, click **Finish**. The application server will be created and started with the new configuration on the same system.

6.4 Create new server - Server type: Remote Server Attach

In this section we focus on creating the Remote Server Attach Application server.

1. After selecting Remote Server Attach as the server type in the initial settings, you should see a window similar to Figure 6-12.



Create a New Server and Server Configuration

Choose the properties for the new server.

Server name: rajat

Folder: Servers

Server type:

- WebSphere version 5.0
  - Express Remote Server
  - Express Test Environment
  - Remote Server Attach
- Apache Tomcat version 4.1
- Apache Tomcat version 4.0
- Apache Tomcat version 3.2
- J2EE Publishing Server

Template: None

Description: Attaches to a WebSphere version 5.0 server that is already started in debug mode.

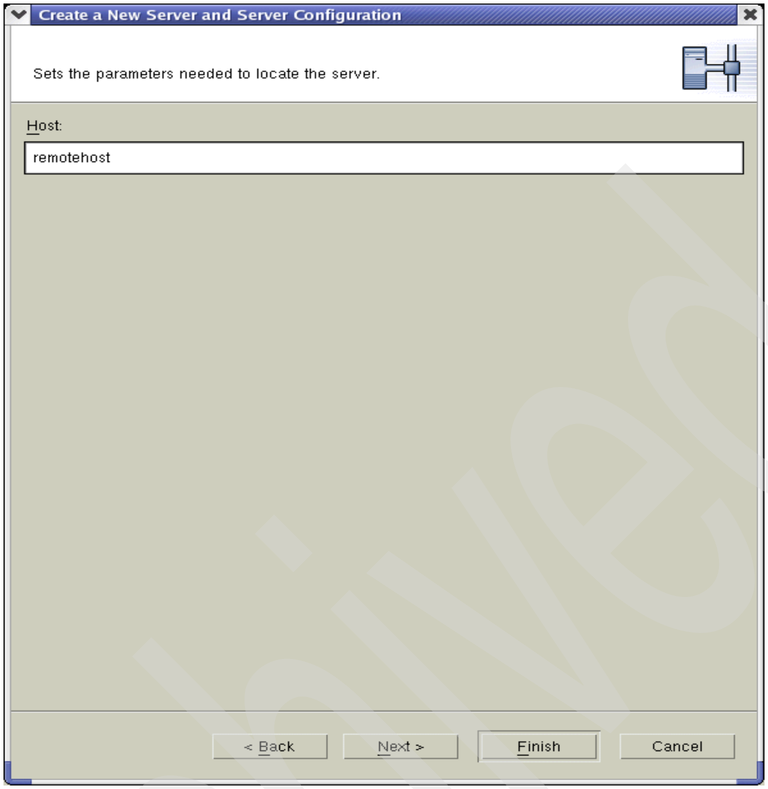
Server configuration type: WebSphere v5.0 Remote Server Attach Configuration

Template: None

Description: A configuration for attaching to a WebSphere version 5.0 server.

Figure 6-12 Remote Server Attach window

2. Click **Next**. You will see a window similar to Figure 6-13.



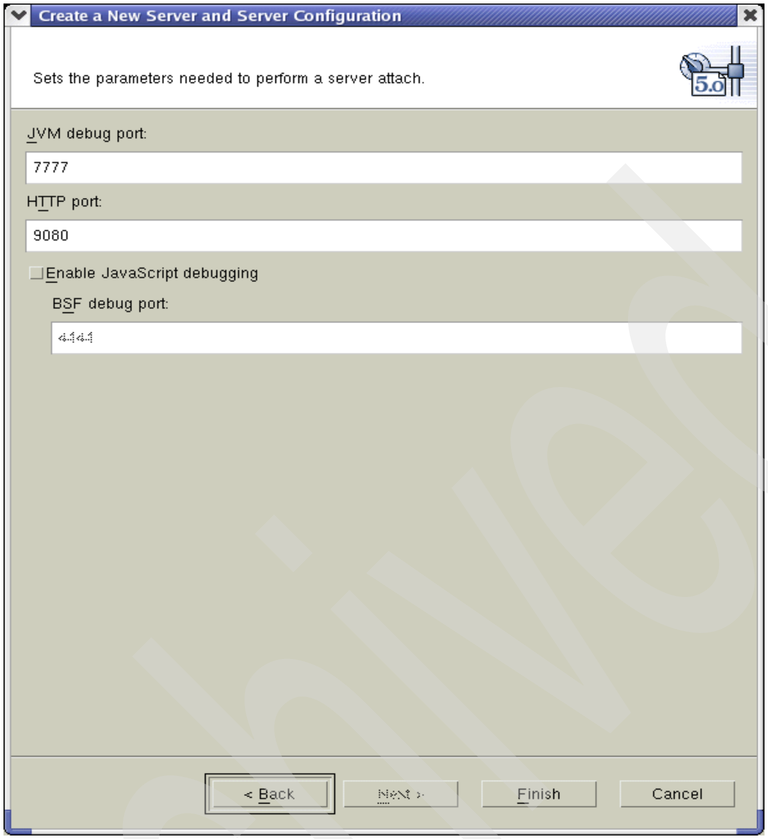
Create a New Server and Server Configuration

Sets the parameters needed to locate the server.

Host: remotehost

Figure 6-13 Host name for Remote Server Attach window

3. At this window, you are prompted for the host name on which the remote application server is running. You can either enter a host name or an IP address of the server where the WebSphere Application Server - Express is running. Make sure that if you are entering a host name, it is resolvable. Click **Next**. You will see a window similar to Figure 6-14.



Create a New Server and Server Configuration

Sets the parameters needed to perform a server attach.

JVM debug port: 7777

HTTP port: 9080

□ Enable JavaScript debugging

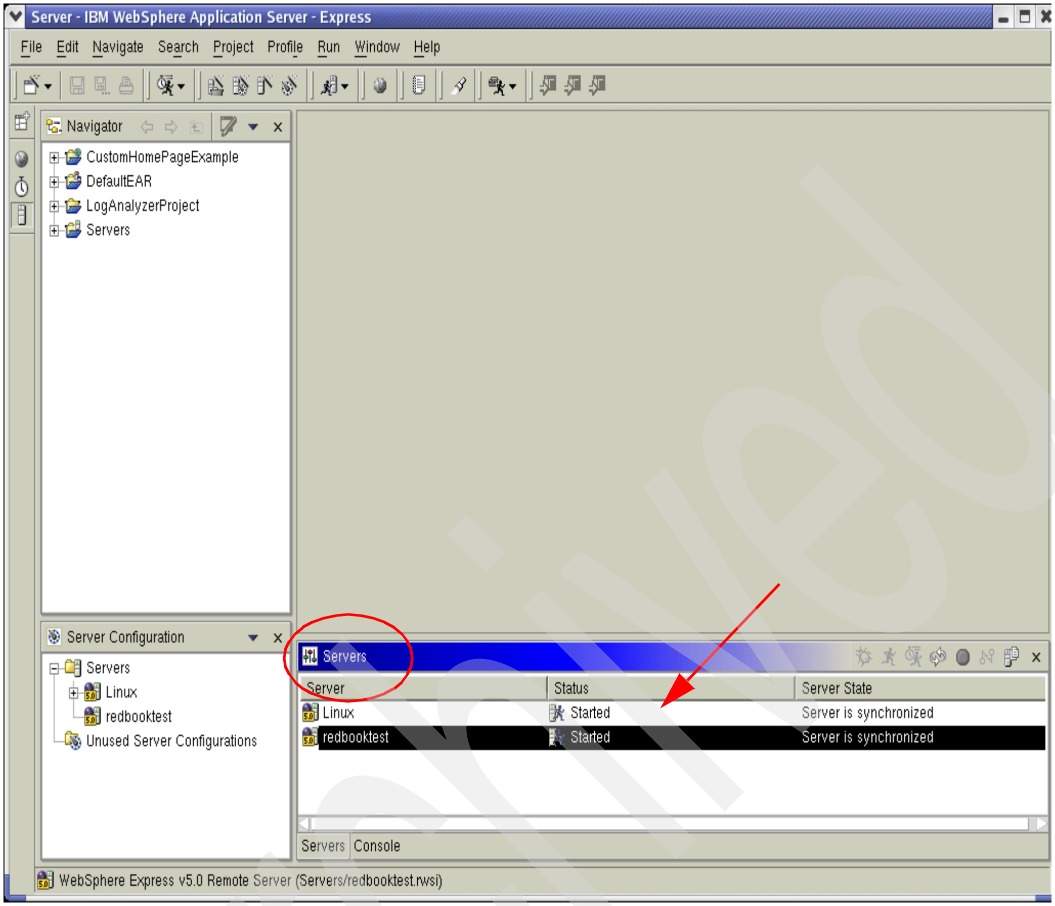
BSF debug port: 4444

Figure 6-14 Debugging and other details window

4. You will now be prompted for entering the port in which the JVM debug will execute along with the HTTP port to which the application server can be connected. Also, you are prompted for the BSF debug port in case JavaScript debugging is to be enabled.

   Once you enter these details, click **Finish** to connect to the Remote Server with the mentioned configuration.

> **Note:** This completes the configuration of server types. Once you have created the server based on one of the three server types selected and the specific configuration you require, it will appear at the bottom right of the window under the section Servers (Figure 6-15).



Server	Status	Server State
Linux	Started	Server is synchronized
redbooktest	Started	Server is synchronized

Figure 6-15 Server status menu

6.5 Starting and stopping the application server

After the creation of the application server based on one of the three server types, you must start and stop the server for normal maintenance functions. Perform the following procedures for either server type:

1. At the bottom of the window, it will indicate if the server has been started or not. If the server is not started, using the mouse right-click on the server name you want to start, and choose **Start** from the drop down menu. This will start the server.
2. To stop the server, right-click on the server you want to stop and then click **Stop** from the drop down menu that will appear.

6.6 Server command line interface (wsadmin)

All server configuration and communication for WebSphere Application Server - Express is usually done using Studio Site Developer. In the unlikely event that communication between Studio Site Developer and your application server cannot be established, the server command line interface provides several commands you can use to determine what is the problem.

Important: These commands must only be used in emergency situations. Studio Site Developer should be used for all server configurations and control.

The commands provided are as follows:

- startServer
- stopServer
- serverStatus

There is only one server per system in WebSphere Application Server - Express and it is called server1.

6.6.1 startServer Command

The startServer command can be used to start the Application Server. Depending on the option given, this command can either start the server or create a server start script. The command can be issued as follows:

```
startServer server1
```

```
startServer server1 -script
```

```
startServer server1 -trace
```

The -script command line option is used to create a start script file. The -trace option is used to create a log file of the start sequence of the server. If you do not specify any options, the command will create a new JVM and start the server. You must specify the server name in order to start the server.

6.6.2 stopServer Command

The stopServer command is used to stop the specified server. This command will wait until the server has completely stopped before returning control back to the command line. By using the -nowait option, control can be returned immediately to the command line. There are several command line options that can be used with the stopServer command:

```
stopServer server1

stopServer server1 -nowait

stopServer server1 -trace

stopServer server1 -username db2admin -password db2admin

stopServer server1 -port 8880
```

The first option is the server name and must be specified.

6.6.3 serverStatus Command

The serverStatus command is used to get the status of the specified server. It will list the status of all servers configured for a specified node.

```
serverStatus server1

serverStatus server1 -trace

serverStatus -?
```

The first argument is the server name and must be specified. The -trace option will create a trace file of the internal trace during the execution of the serverTrace command. The -? option will list the usage of the serverStatus command.

7

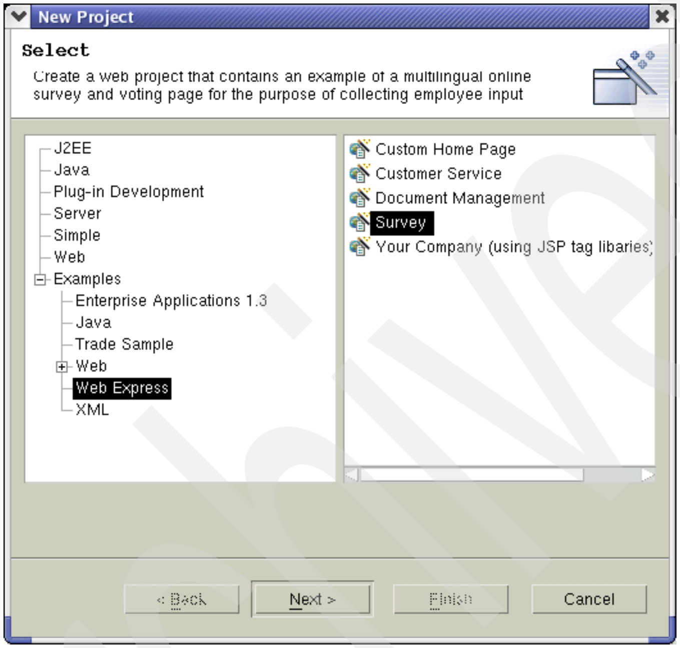
Creating business applications using the Application Server - Express wizard

In this chapter, we create a new Web project based on the wizard provided by WebSphere Studio Site Developer.

7.1 Steps to create a Web project

From the WebSphere Studio Site Developer, complete the following steps to create a Web project:

1. Click **File -> New -> Project**. This will open a new window for the creation of a new project (Figure 7-1).

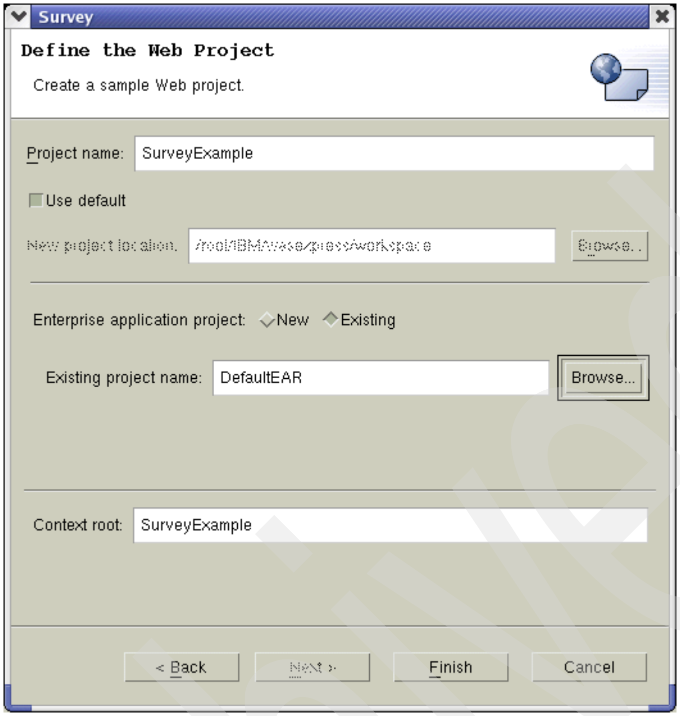


Select

Create a web project that contains an example of a multilingual online survey and voting page for the purpose of collecting employee input

Figure 7-1 Create new application window

2. Choose the type of project you want to create. In our case, choose **Web Express** and choose **Survey** application.
3. After choosing the application you want to create, click **Next**. You will see a window similar to Figure 7-2.



Survey

Define the Web Project

Create a sample Web project.

Project name: SurveyExample

□ Use default

New project location: /root/IBM/wase/press/workspace

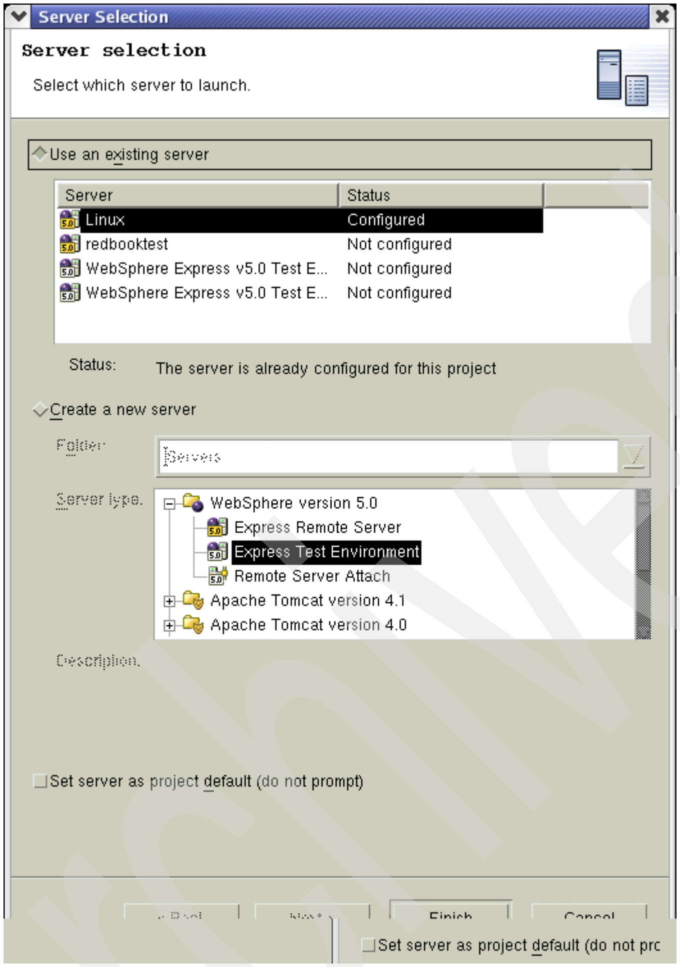
Enterprise application project: ◇ New ◆ Existing

Existing project name: DefaultEAR

Context root: SurveyExample

Figure 7-2 Step 2 Creation of application

4. Now you will be prompted for the new project name. The default name would appear under the Project name field. You can either retain the name provided or enter a new name. In our case, we retained the name provided by the system - SurveyExample.
5. You can either use an existing Enterprise application project or create a new one. In our example, we used an existing Enterprise application project named DefaultEAR. If you are creating the application for the first time, the window will show you DefaultEAR and have New selected by default. You can now change the context root or retain the system default. After performing the requisite changes, you must click **Finish**.
6. After creation of the application, right-click on the application and choose **Run on server**. You will see a window similar to Figure 7-3.



Server Selection

Server selection

Select which server to launch.

◆ Use an existing server

Server	Status
Linux	Configured
redbooktest	Not configured
WebSphere Express v5.0 Test E...	Not configured
WebSphere Express v5.0 Test E...	Not configured

Status: The server is already configured for this project

◇ Create a new server

Folder: Servers

Server type:

- WebSphere version 5.0
  - Express Remote Server
  - Express Test Environment
  - Remote Server Attach
- Apache Tomcat version 4.1
- Apache Tomcat version 4.0

Description:

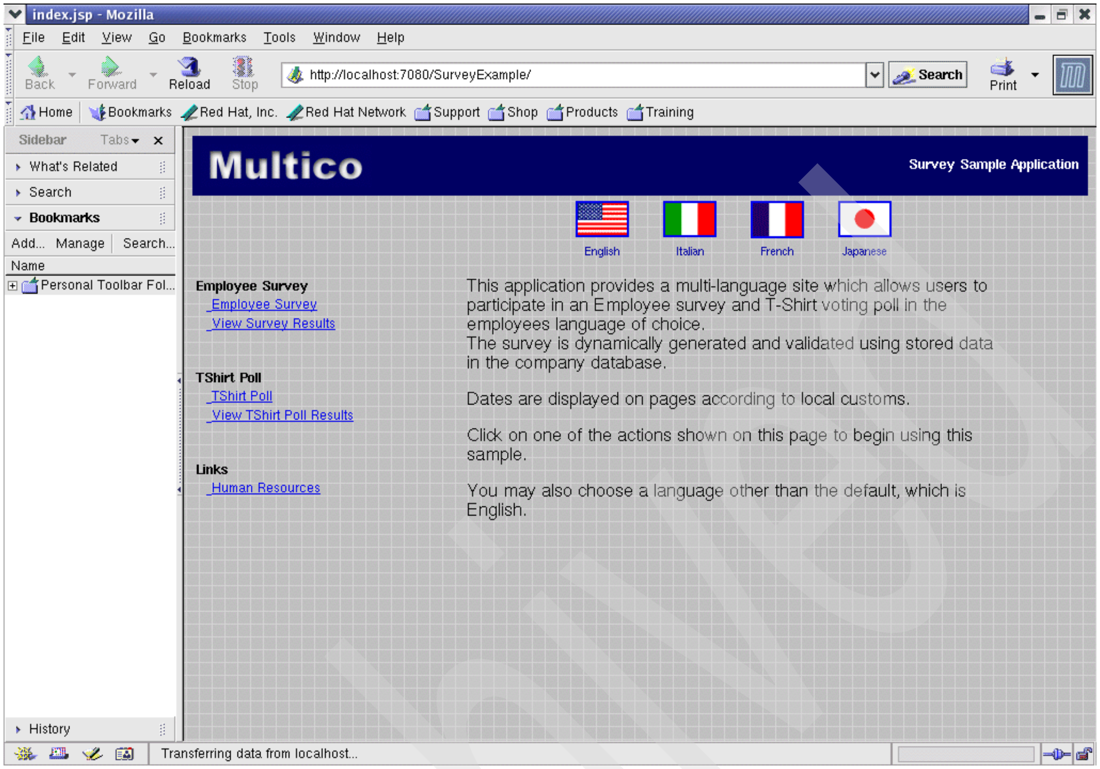
□ Set server as project default (do not prompt)

□ Set server as project default (do not prc

Figure 7-3 Choosing a Web Application Server

7. If you already have a Web application server created, and if it does not appear on the window, then click **Advanced**. Make sure that there is no server running at this time, as the application will now start the server after publishing the content. You can either choose one of the servers that you have already created, or create a new server based on your setup. In our case, we had already created a server as detailed in 6.2, “Create new server - Server type: Express Remote Server” on page 56.

8. If you have already created a Web application server, you can choose the application server you want the application to run on and click **Finish**. This would now start publishing the application on the application server you chose. You will see a window similar to Figure 7-4.



This application provides a multi-language site which allows users to participate in an Employee survey and T-Shirt voting poll in the employees language of choice.
The survey is dynamically generated and validated using stored data in the company database.

Dates are displayed on pages according to local customs.

Click on one of the actions shown on this page to begin using this sample.

You may also choose a language other than the default, which is English.

Figure 7-4 Survey Sample Application window

9. Once the application is published, it will be visible in a browser for usage. Now this Survey application is ready for use.

10. To access this application, you can either type in the domain name (if you have a DNS running) or an IP address with the port number on which the Application Server - Express is running from a Web browser, and the application will be appear as a normal HTML page.

8

Security

In this chapter we highlight the general security considerations that need to be taken care of, before and after installing WebSphere Application Server - Express, V5.

8.1 General security considerations

The WebSphere Application Server - Express, V5 for Linux installation does not give you the option of changing the directory path for installation and the default installation directory is "*/opt/...*" Depending on the packages that you are going to install, there might be 2 - 3 directories under opt used for the purpose of WebSphere application Server - Express. If you are conscious about file system management and security, you would need to plan for the same in advance. Also note that all files in this directory would be owned by root.

Consider the security issues related to operation of a Web server. The following issues should be discussed during the planning session:

- Is this a publicly accessible server, or will each user need to be authenticated? This raises the issue of how users will be authenticated - that is, how they will be able to prove who they say they are (Digital Certificates usually solve this problem)
- Will the server be attached directly to the Internet, or will it be connected only to a corporate intranet? This raises the issue of whether a firewall might be appropriate in the setup
- Will all users of the Web server have equal access to all information? This raises the issue of what information needs to be protected and how that information will be protected
- Will sensitive information be travelling over a non-secure network? Sensitive data may include user IDs, and passwords as well as confidential data supplied by an application.
- The customer's security policy should provide guidance on all of these issues. You may find that your customer's security policy has not been updated to reflect security considerations introduced by TCP/IP and Internet technology.

Once Linux is activated with TCP/IP support, it enables a lot of services, such as Sendmail, FTP, etc., which may not be required in the system and need to be stopped. If the customer does not have the expertise to do the same, and if the server is publicly accessible, the customer should consider installing a firewall and restrict certain ports accessible from the Internet to the Web server.

8.1.1 Securing the IBM Agent Controller

You will need to secure the IBM Agent Controller for a production environment. This is achieved by restricting access to the Agent to certain hosts or addresses. You can restrict access to the Agent using the Hosts setting in the serviceConfig.xml file in the config directory of the Agent Controller. Typically this would be /opt/IBM//RAC/config.

Modify the Hosts section to allow only certain hosts:

```
<Hosts>
<Allow host="ALL"/>
</Hosts>
```

This configuration would allow all hosts to access the agent controller. Use the keyword Allow or Deny to list host address you with to restrict access to. You can use masks to restrict access to certain network address ranges.

```
<Hosts>
    <Deny net="192.168.0.1" mask="255.255.255.255"/>
    <Allow net="192.168.0.0" mask="255.255.0.0"/>
    <Allow host="steamboat"/>
</Hosts>
```

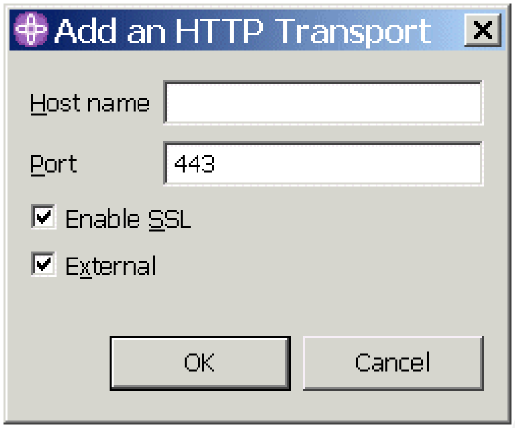
This configuration will allow access from all machines belonging to the 192.168.0.0 C-class network except 192.168.0.1. It will also allow access from the host steamboat. All other requests are denied.

8.1.2 Securing your application using SSL

WebSphere Application Server - Express supports the use of certificates and SSL. Your application needs to be developed using SSL techniques with either JSP-Tag libraries or values set in your JSP pages and servlets. The configuration of SSL for the server is done on the ports tab of the **Configure Server** window in WebSphere Studio Site Developer.

To enable SSL in WebSphere Studio Site Developer:

1. Select **Server Perspective**.
2. Double-click your server configuration to open it.
3. Select **Ports** tab.
4. In the server settings, click **Add**.



Host name

Port 443

☑ Enable SSL

☑ External

Figure 8-1 HTTP Transport configuration

5. Type in the details, for example, the port on which you which to use SSL, and check the **Enable SSL** check box.
6. Click **OK**.

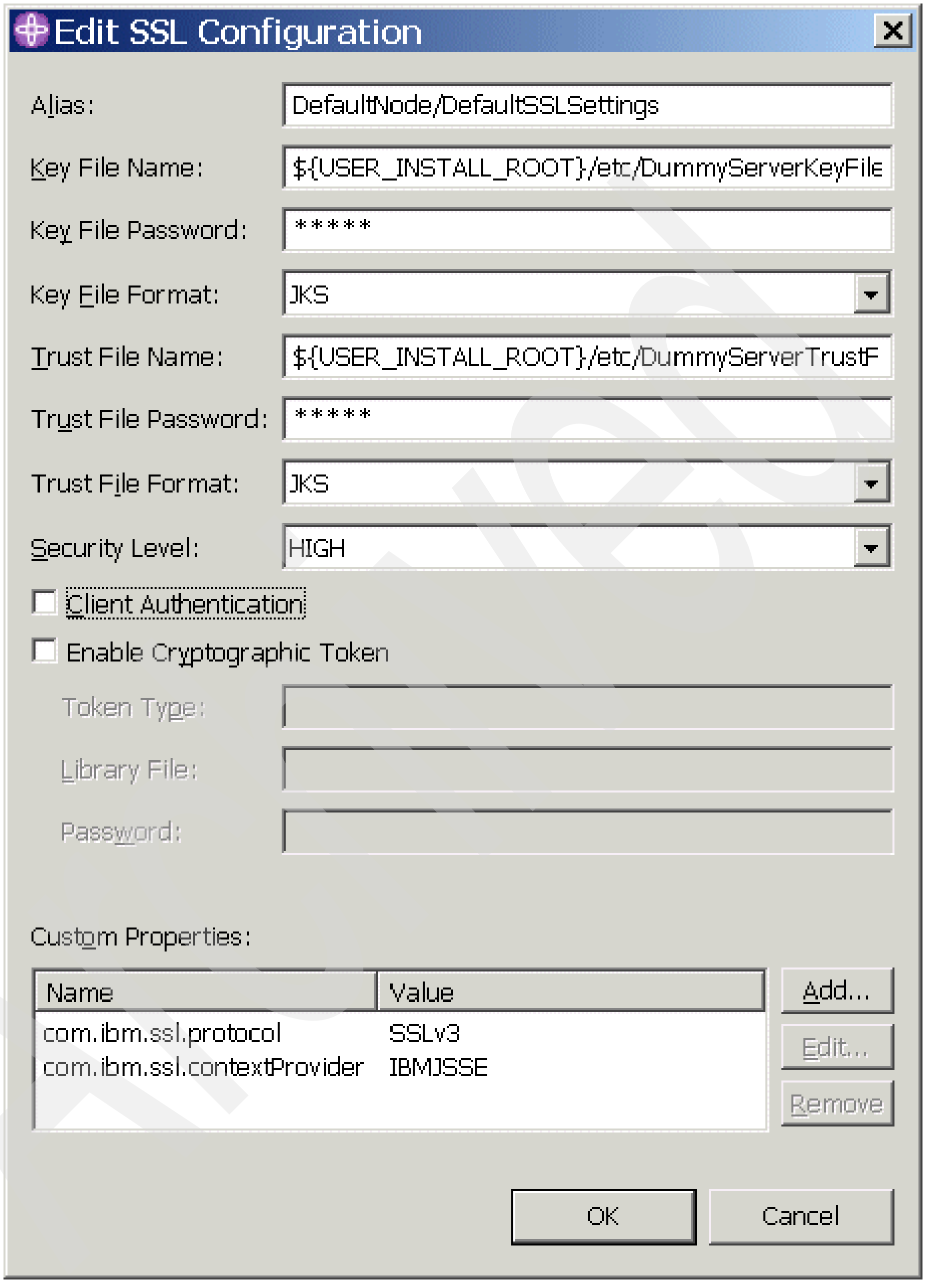
This will add a new port to your server. If the Enable SSL check box is checked, then this port will be used in a secure environment. Now the applications that are running on this port will be accessible using SSL only.

8.1.3 Adding a certificate to your Application Server

In order to use SSL, you will need to get a certificate. These can be obtained from certification authorities and service providers that provide security certificates.

To use this certificate, you need to add it to your WebSphere Application Server - Express server configuration:

1. Open the Server Perspective in WebSphere Studio Site Developer.
2. Select **Security Tab**.
3. Under SSL Configuration Properties, click **Add**.



Edit SSL Configuration

Alias: DefaultNode/DefaultSSLSettings

Key File Name: ${USER_INSTALL_ROOT}/etc/DummyServerKeyFile

Key File Password: *****

Key File Format: JKS

Trust File Name: ${USER_INSTALL_ROOT}/etc/DummyServerTrustF

Trust File Password: *****

Trust File Format: JKS

Security Level: HIGH

□ Client Authentication

□ Enable Cryptographic Token

Token Type:

Library File:

Password:

Custom Properties:

Name	Value
com.ibm.ssl.protocol	SSLv3
com.ibm.ssl.contextProvider	IBMJSSE

Figure 8-2 SSL Configuration window

4. Fill in the details of your certificate and click **OK**.
5. Save your server configuration. Republish and restart your Application Server.

8.1.4 Creation of a test certificate for testing SSL

Before you actually implement production certificates, which you may need to source from a commercial Certification Authority, you may want to test the SSL connectivity. This would require the creation of a sample SSL certificate and deploying the same in the test environment. Note that the steps below will explain how to get a test certificate and deploy it in the environment. However, this certificate should not be used in a production environment, as it will not be accepted by browsers across the Internet.

1. You will have to create a certificate pair for your server, and a certificate pair for the dummy CA who will certify the certificates. So we will generate a private key first for the server. It is assumed that you will have OpenSSL in your system for executing the following steps.

   In case you do not have OpenSSL, go to Appendix B, "Web server configuration for virtual domains and securing WebSphere Application Server - Express" on page 113. Read section B.5, "Uncompressing OpenSSL source" on page 117 and make sure you have the requisite software in your setup. Execute the following commands in the OpenSSL directory. In our case we executed these commands in the /usr/local/ssl directory. Executing this command from this directory is just for easy maintenance purposes. Or you could create a temp directory and execute the following commands from it:

   ```
   # openssl genrsa -des3 -out server.key 1024
   ```

   You will be prompted for a password here. Enter a password which you will remember easily. After entering the password once, you will be asked to re-enter the password for validation.

2. Now you need to generate a request with the private key. This is for the public key which you will certify later:

   ```
   # openssl req -new -key server.key -out server.csr
   ```

   Again you will be asked for a password. Enter the same password that you entered above. After entering this password, you will be asked a set of questions that are similar to the one you entered above in the openssl.cnf file.

3. The default entries appear in the bracket after the question. You can either press Enter to accept that entry, or enter a different parameter. Note that these parameters will be used to create the certificate for testing, so enter only your Web site URL (fully-qualified name of your Web server) for which this certificate is going to be used in the Common Name parameter.

4. After entering all the relevant information, you will be asked for a challenge password. It is best to leave the challenge password blank.

   Once you have created the server.csr file, you will have to create requisite certificates for the dummy CA. It should be noted that if you have a valid certificate with you, then you do not need to run through this entire certificate generation section.

5. Now generate a private key for your CA:

   ```
   # openssl genrsa -des3 -out ca.key 1024
   ```

   You will be asked for a password that will secure your CA. It is important to remember this password since you will be prompted for the same when certifying your server certificate.

6. To make the CA private key a complete usable CA, you will have to execute the following command.

   ```
   # openssl req -new -x509 -days 999 -key ca.key -out ca.crt
   ```

   You will be prompted for a password. This one is the same that you entered in step 5. You will be asked a series of questions similar to the one when you created the file server.csr. Enter details which are different from the server.csr create in step 4. Remember not to press Enter when the questions are prompted without entering any details as that will take the default entries which might be the one applicable to the server.csr file. Do not enter the same details that you entered when you created the server.csr file.

   Make sure you copy all the csr and crt files to the certs directory, and all the key files to the private directory.

7. You can do the same by executing the following commands from the OpenSSL directory. In our case this is /usr/local/ssl:

   ```
   # mv *.crt certs
   # mv *.csr certs
   # mv *.key private
   ```

At this point you will have a ca.crt and server.csr file in the certs directory and a server.key and ca.key file in the private directory.

Now you can use this CA to sign server certificates to secure your Web sites. You will need one for each domain that you plan to test. We planned to test only one domain with SSL, hence we created only one server.csr file and one CA for the same.

Signing the certificate

In this section, we will sign the certificate. Perform the following instructions:

1. Create the following files under the OpenSSL directory:

   ```
   touch /usr/local/ssl/ca.db.index
   touch /usr/local/ssl/ca.db.serial
   ```

2. Create the following director:

   ```
   mkdir /usr/local/ssl/ca.db.certs
   ```

   Create the ca.config file under the OpenSSL directory.

   The contents of the file will appear as shown in Example 8-1:

Example 8-1 ca.config file

```
[ ca ]
default_ca              = CA_own
[ CA_own ]
dir                     = /usr/local/ssl/
certs                   = /usr/local/ssl/certs
new_certs_dir           = /usr/local/ssl/ca.db.certs
database                = /usr/local/ssl/ca.db.index
serial                  = /usr/local/ssl/ca.db.serial
RANDFILE                = /usr/local/ssl/my_ca.db.rand
certificate             = /usr/local/ssl/certs/ca.crt
private_key             = /usr/local/ssl/private/ca.key
default_days            = 365
default_crl_days        = 30
default_md              = md5
preserve                = no
policy                  = policy_anything
[ policy_anything ]
countryName             = optional
stateOrProvinceName     = optional
localityName            = optional
organizationName        = optional
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional
```

Once you have created this file in the OpenSSL directory (in our case /usr/local/ssl), you can now sign the certificate.

3. Execute the following command:

   ```
   # openssl ca -config ca.config -out /usr/local/ssl/certs/server.crt
   -infiles /usr/local/ssl/certs/server.csr
   ```

   You are prompted for a password, which is the password you had used for generation of the CA file, in step 5 on page 85. After that you will be prompted for verification of the signature of this file and asked whether you would like to commit them. You will have to respond “yes” to these questions. This will now sign the certificate and create a server.crt file in the same directory.

4. Copy it to the certs directory under your OpenSSL installation. In our case, we executed the following command:

   ```
   # cp server.crt /usr/local/ssl/certs
   ```

5. Now, to verify whether your certificate generation was correct, execute the following command without changing your directory:

   ```
   # openssl verify -CAfile /usr/local/ssl/certs/ca.crt
   /usr/local/ssl/server.crt
   ```

You should get an OK prompt after the certificate is verified. This means that you have the certificates ready for usages in a SSL environment.

You can now use these certificates — server.crt, server.key, ca.crt in the SSL configuration of WebSphere Application Server - Express, as described in 8.1.3, “Adding a certificate to your Application Server” on page 82. Remember the server key password you entered during the certificate creation above, since these details must be entered in the window similar to Figure 8-2 on page 83.

8.2 Customer personnel resources

The customer’s systems programmer, security administrator, and database administrator (if there is a database in the setup) must be available for consultation by the IBM service provider during the engagement.

The customer needs to ensure that they have sufficient skilled people to:

- Operate WebSphere Application Server - Express and the HTTP server.
- Resolve operational and performance problems encountered during the use of the HTTP server and WebSphere Application Server - Express.
- Assist users who are connecting to the Web server with access problems or application problems.

8.3 Authentication certificate considerations

This service includes creation of a key ring and a self-signed authentication certificate for use in enabling SSL support. A self-signed certificate is useful only within the organization that created the certificate because generally, those outside of the organization that created the certificate will not trust the validity of the certificate.

General use (on the Internet) of SSL requires that a certificate from a recognized certificate authority be obtained and installed on the Linux/Windows system. Several recognized Certificate Authorities are available. The URLs for some of the widely recognized Certificate Authorities are listed below. The names of these Certificate Authorities are provided for information purposes only; IBM does not endorse the use of any particular Certificate Authority.

Verisign:	http://www.verisign.com
GTE Cybertrust:	http://www.cybertrust.gte.com
Entrust:	http://www.entrust.com

A

Microsoft SQL Server 2000 data source configuration

IBM WebSphere Application Server - Express, Version 5 provides a way to implement Microsoft SQL 2000 Server database connection. The process involves the following steps:

- Prerequisites
- Installing the Microsoft SQL Server 2000 drivers
- Configuring the data source
- Testing the connection

A.1 Prerequisites

Listed in this section are the prerequisites for the Microsoft SQL Server 2000 database connection.

A.1.1 Database prerequisite

For this example, the following information was used:

- The database created in Microsoft SQL Server 2000 is named: Test1.
- The table created in Microsoft SQL Server 2000 is named: employee.
- The table has the following columns: First_Name, Last_Name, and Department,
  and contains the data listed in Table A-1.

Table A-1 Test data

First_Name	Last_Name	Department
Mike	Bowman	Web
John	Coyne	Web
Mark	Hall	HR
Allen	Hill	Web
Chris	Worwood	Boss

A.1.2 Sample servlet

The following Java code is for the DataSourceTest servlet. This servlet will be used to test data source connections and perform SQL statements. The servlet needs to be created and installed to utilize the examples.

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class DataSourceTest extends HttpServlet {
```

```
public void doGet(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException {

   resp.setContentType("text/html; charset=UTF-8");
   PrintWriter out = resp.getWriter();
   StringBuffer html = new StringBuffer(200);

html.append(
   "<html><body><form method=\"POST\" action=\"DataSourceTest\"");
   html.append("<br><h1><center>Datasource Test Servlet</h1>");
   html.append(
   "<br>Enter your SQL here: <br><textarea name=\"sql\" cols=\"80\"
   rows=\"17\"></textarea>");
   html.append(
   "<br><INPUT type=radio name=\"insert\" value=\"INSERT\" >Update
   &nbsp;  <INPUT type=radio name=\"insert\" checked value=\"QUERY\"
   >Select</center>");
   html.append(
   "<br><br><br>Datasource name to be tested: <input type=\"text\"
   name=\"datasource\" value=\"jdbc/\">");

   html.append(
   "<br><br><br>Max Number of Rows to Return: <input type=\"text\"
   name=\"max\" value=\"50\" >");
   html.append("<br><br><input type=\"submit\" name=\"sub\">");
   html.append("</form></body></html>");
   out.println(html.toString());
   out.flush();

}

public void doPost(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException {

   try {
      boolean insert =
      req.getParameter("insert").equalsIgnoreCase("INSERT");
      if (insert)
      doUpdate(req, resp);
      else
      doQuery(req, resp);
   } catch (Exception e) {
      e.printStackTrace();
   }
}

   public void doQuery(HttpServletRequest req, HttpServletResponse
   resp)
throws ServletException, IOException {

   resp.setContentType("text/html; charset=UTF-8");
   java.sql.ResultSet resultSet;
   PrintWriter out = resp.getWriter();
   StringBuffer html = new StringBuffer(200);
   String datasource = req.getParameter("datasource");
   String sql = req.getParameter("sql");
   String smax = req.getParameter("max");
   int max = Integer.parseInt(smax);

   boolean insert =
   req.getParameter("insert").equalsIgnoreCase("INSERT");

   html.append(
   "<html><body><h1>SQL Results for:</h1><body><br>"
   + sql
   + "<br><br>");
   ResultSet r = null;
   DatabaseBroker broker= new DatabaseBroker(datasource);
   try {
      long startTime = System.currentTimeMillis();

      r = broker.getQuery(sql);
      long endTime = System.currentTimeMillis();
      long totalTime = endTime - startTime;
      resultSet = r;
      ResultSetMetaData rmd = r.getMetaData();
      html.append(
      "Query Execution Time: " + totalTime + "
      milliseconds<br><br>");

      html.append("<table border=\"1\"><tr>");

      for (int j = 0; j < rmd.getColumnCount(); j++) {
      html.append("<td>" + rmd.getColumnName(j + 1) + "</td>");
      }
      html.append("</tr>");
      int index = 0;

      while (r.next() && index < max) {
      html.append("<tr>");
      for (int i = 0; i < rmd.getColumnCount(); i++) {
      html.append("<td>" + format(r.getObject(i + 1)) + "</td>");

      }
      index++;
      html.append("</tr>");
      }
      html.append("</table>");
      html.append("</html>");
      out.println(html.toString());
      out.flush();

   } catch (Exception e) {
      e.printStackTrace(out);

   } finally {
      broker.closeConnection();
      }

}
public String format(Object inVal) {
   String sVal = "";
   if (inVal != null) {
   sVal = "" + inVal;
   if (sVal.trim().equals(""))
   sVal = "&nbsp";
```

```
      } else
      sVal = "null";

      return sVal;
   }

   public void doUpdate(HttpServletRequest req, HttpServletResponse
   resp)
      throws ServletException, IOException {
         resp.setContentType("text/html; charset=UTF-8");
         PrintWriter out = resp.getWriter();
         String datasource = req.getParameter("datasource");
         String sql = req.getParameter("sql");

         boolean insert =
         req.getParameter("insert").equalsIgnoreCase("INSERT");
         DatabaseBroker broker = new DatabaseBroker(datasource);

      try {

      int numUpdates = broker.updateData(sql);

      out.println("" + numUpdates + " Rows inserted");
      out.flush();

      } catch (Exception e) {
         e.printStackTrace(out);

      } finally {
         broker.closeConnection();
      }

   }

}
```

A.1.3 Sample data source broker class

The following Java code is for the DatabaseBroker class. This class is used to broker connections between the servlet and data source. The Database Broker class needs to be created and installed on the server to execute the examples.

```
import java.io.*;
import java.util.*;
import java.sql.*;
import java.text.*;
import javax.naming.*;
import javax.servlet.ServletConfig;
import javax.sql.DataSource;
import com.ibm.*;

public class DatabaseBroker {

      private String datasource;
      private ResultSet rs = null;
      private Connection con = null;
      private DataSource ds = null;
      private Statement stmt = null;
      private PreparedStatement pStmt = null;

      public DatabaseBroker(String datasource) {

         this.datasource = datasource;

         try {
         Hashtable env = new Hashtable();
         env.put(
         Context.INITIAL_CONTEXT_FACTORY,
         "com.ibm.ejs.ns.jndi.CNInitialContextFactory");
         InitialContext ctx = new InitialContext(env);
         ds = (javax.sql.DataSource) ctx.lookup(datasource);
         ctx.close();
         } catch (Exception ex) {
         ex.printStackTrace();
         }
      }
      public ResultSet getQuery(String statement) {

         rs = null;
         con = null;
         pStmt = null;

         try {

         con = ds.getConnection();
         pStmt = con.prepareStatement(statement);
         rs = pStmt.executeQuery();

         } catch (Exception ex) {
         ex.printStackTrace();
         }

         return rs;

}

      public int updateData(String statement) {

         int returnInt = 0;

         con = null;
         pStmt = null;

         try {

         con = ds.getConnection();
         pStmt = con.prepareStatement(statement);
         returnInt = pStmt.executeUpdate();

         } catch (Exception ex) {
         ex.printStackTrace();
         }
         return returnInt;

      }

      public void closeConnection() {

         try {

         if (pStmt != null) {
         pStmt.close();
         pStmt = null;
         }

         if (con != null) {
         con.close();
         con = null;
         }

         if (rs != null) {
         rs.close();
         rs = null;
         }

         if (ds != null) {
         ds = null;
         }

         } catch (SQLException ex) {
         ex.printStackTrace();

         }
      }

}
```

A.2 Install Microsoft SQL Server 2000 driver

The Microsoft SQL Server 2000 driver can be retrieved from the Microsoft Web site. The following URL is given for the SQL Server JDBC driver downloads page at the time this document was written:

http://msdn.microsoft.com/library/default.asp?url=/downloads/list/sqlserver.asp

To install the JDBC drivers, follow the Microsoft installation instructions.

A.3 Configure data source

This section guides you through configuring a data source for Microsoft SQL Server 2000:

1. Go to your Server Perspective.
2. Create a new server and server configuration (see Figure A-1).

   Select **File > New > Server** and **Server Configuration**.

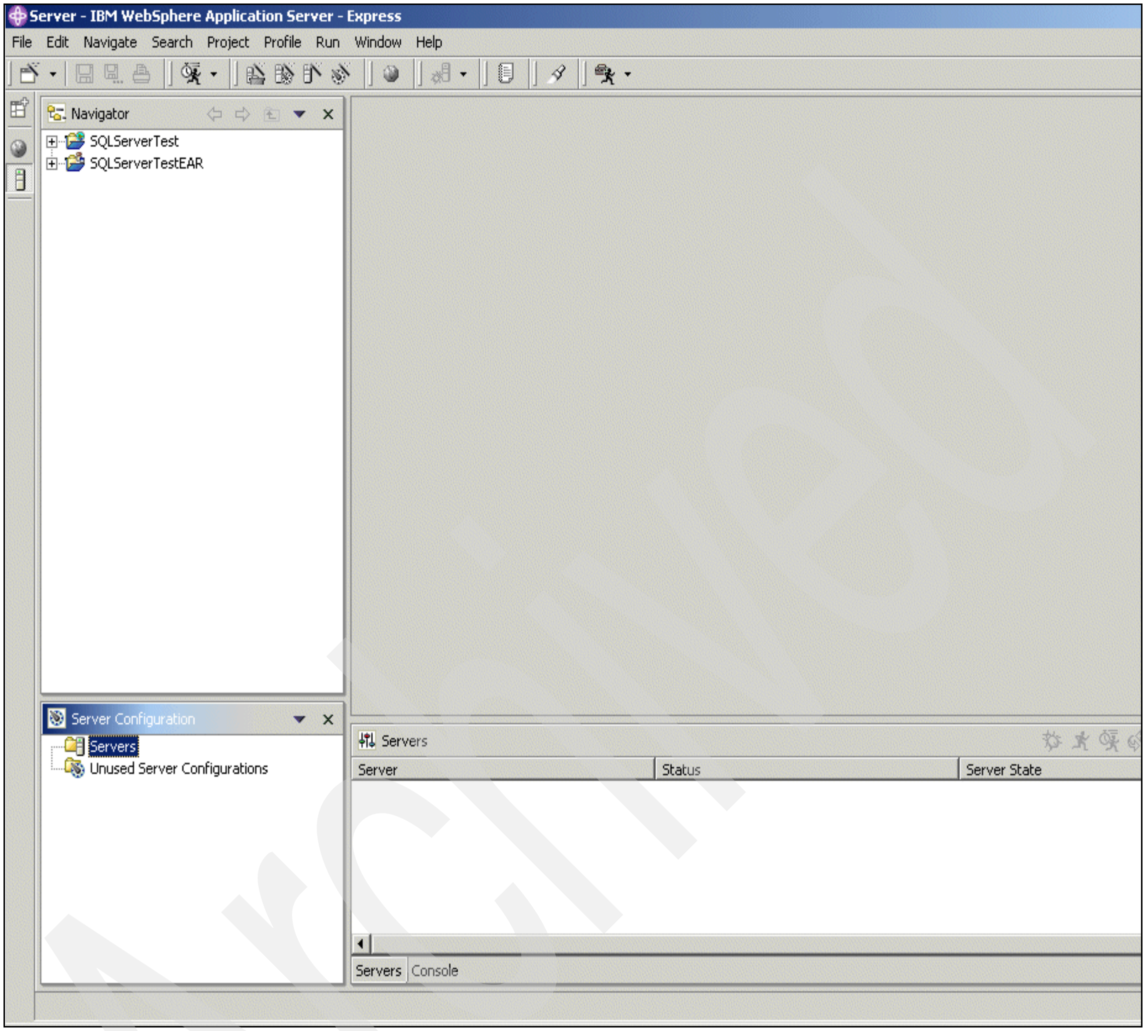
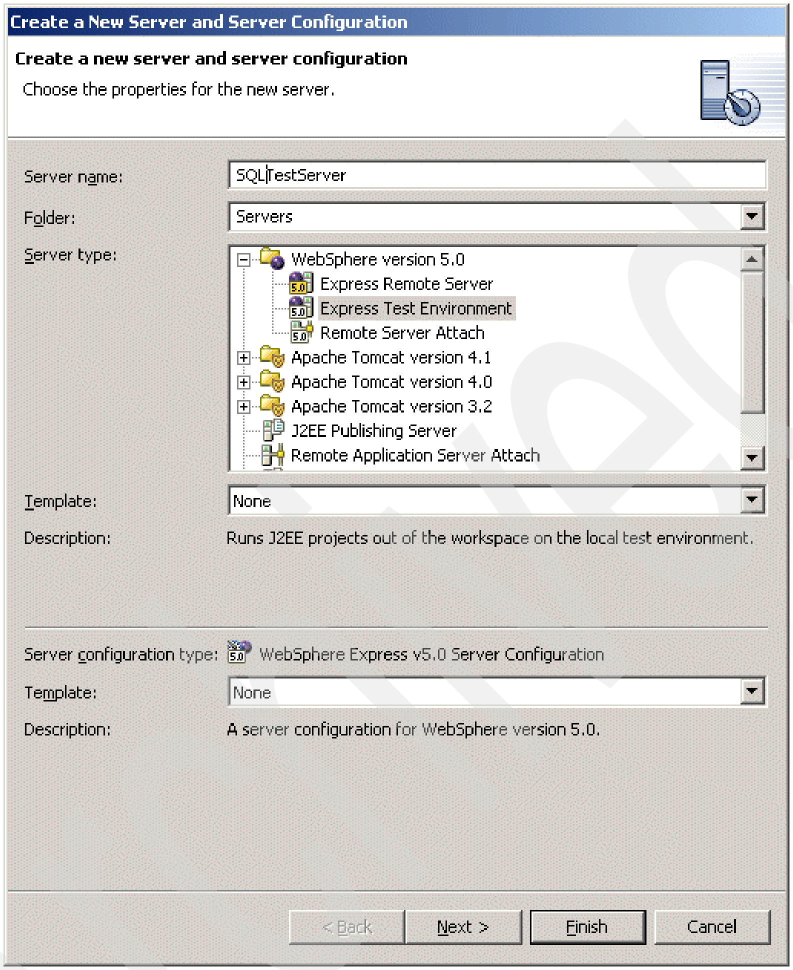

Server - IBM WebSphere Application Server - Express
File Edit Navigate Search Project Profile Run Window Help
Navigator
SQLServerTest
SQLServerTestEAR
Server Configuration
Servers
Unused Server Configurations
Servers
Server
Status
Server State
Servers
Console

Figure A-1 Server Perspective

3. Give your new server a name. For example, see Figure A-2.



Create a New Server and Server Configuration

Create a new server and server configuration

Choose the properties for the new server.

Server name: SQLTestServer

Folder: Servers

Server type:

- WebSphere version 5.0
  - Express Remote Server
  - Express Test Environment
  - Remote Server Attach
- Apache Tomcat version 4.1
- Apache Tomcat version 4.0
- Apache Tomcat version 3.2
- J2EE Publishing Server
- Remote Application Server Attach

Template: None

Description: Runs J2EE projects out of the workspace on the local test environment.

Server configuration type: WebSphere Express v5.0 Server Configuration

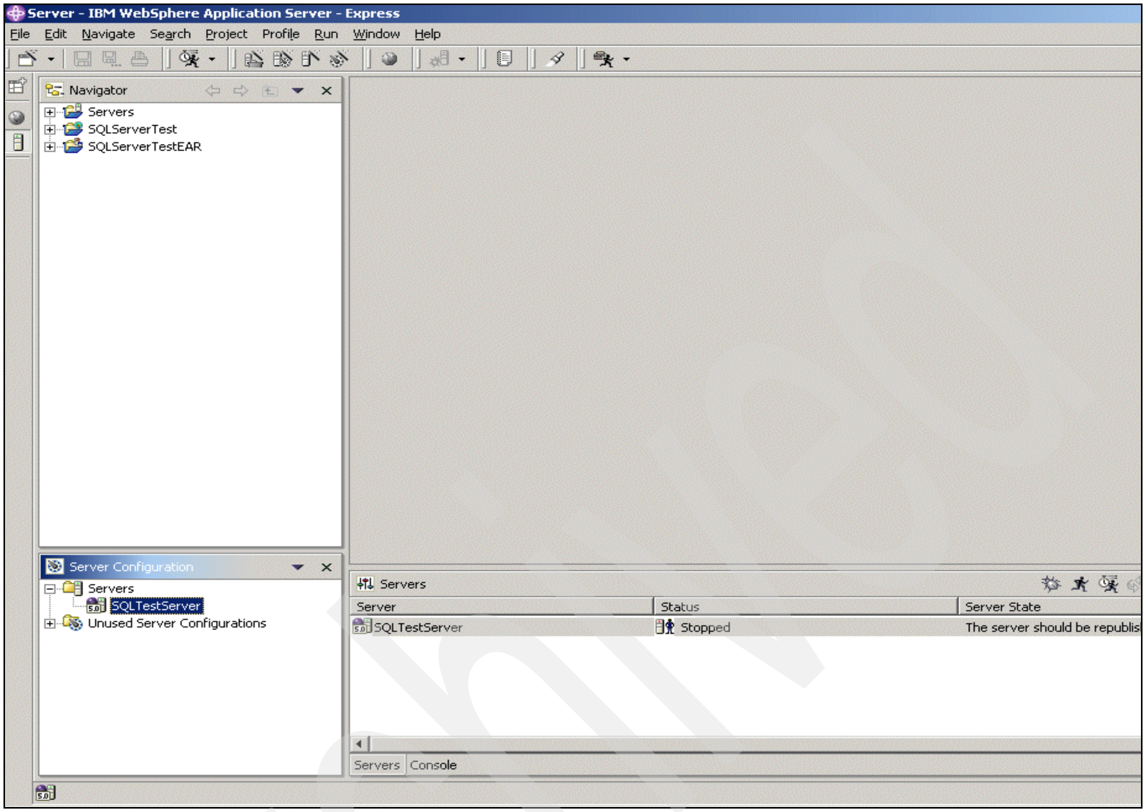
Template: None

Description: A server configuration for WebSphere version 5.0.

Figure A-2 Create new server and server configuration

4. Click **Finish**.

5. Double-click your new server (see Figure A-3).

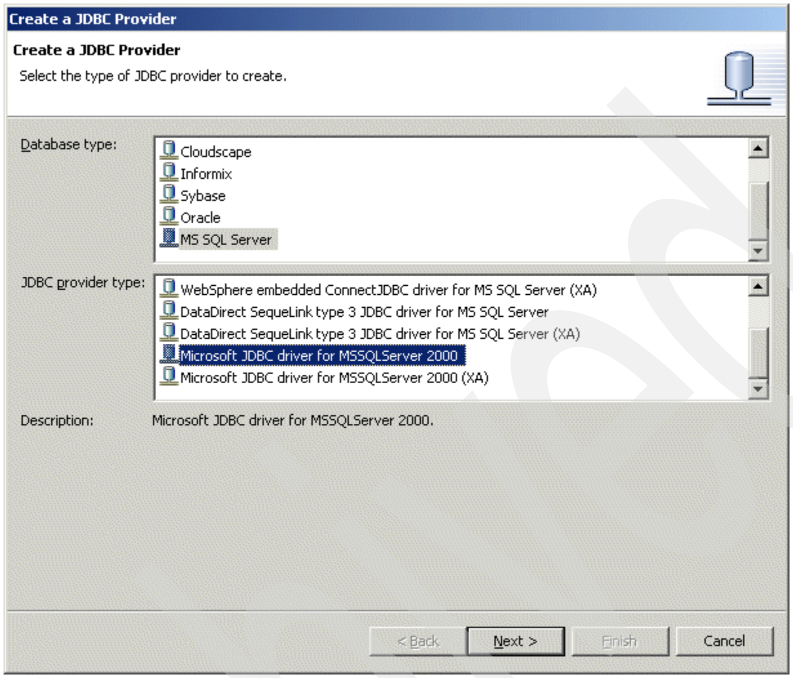

Server - IBM WebSphere Application Server - Express
File Edit Navigate Search Project Profile Run Window Help
Navigator
Servers
SQLServerTest
SQLServerTestEAR
Server Configuration
Servers
SQLTestServer
Unused Server Configurations
Servers
Servers Console

Server	Status	Server State
SQLTestServer	Stopped	The server should be republis

Figure A-3 Server Perspective with new server

6. Click **Data source** tab.
7. Click **Add** under the JDBC provider list.
8. Click **MS SQL Server** under the Database type.

9. Click **Microsoft JDBC driver MSSQLServer 2000** under the JDBC provider type (see Figure A-4).



Create a JDBC Provider

Create a JDBC Provider

Select the type of JDBC provider to create.

Database type:

- Cloudscape
- Informix
- Sybase
- Oracle
- MS SQL Server

JDBC provider type:

- WebSphere embedded ConnectJDBC driver for MS SQL Server (XA)
- DataDirect SequeLink type 3 JDBC driver for MS SQL Server
- DataDirect SequeLink type 3 JDBC driver for MS SQL Server (XA)
- Microsoft JDBC driver for MSSQLServer 2000
- Microsoft JDBC driver for MSSQLServer 2000 (XA)

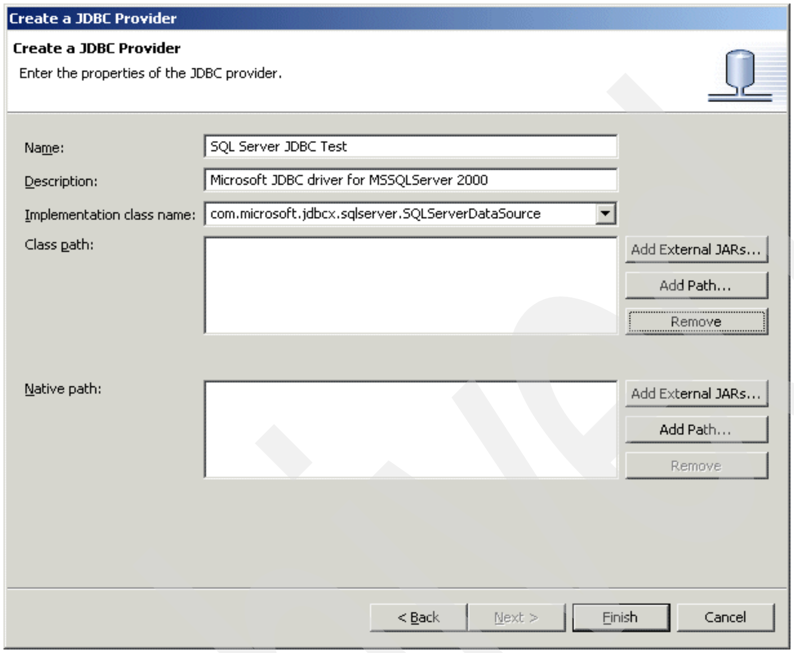
Description: Microsoft JDBC driver for MSSQLServer 2000.

Figure A-4 JDBC provider

10.Click **Next**.

11. Give your JDBC provider a name.

12. Remove all Class path entries (see Figure A-5).



Create a JDBC Provider

Create a JDBC Provider

Enter the properties of the JDBC provider.

Name: SQL Server JDBC Test

Description: Microsoft JDBC driver for MSSQLServer 2000

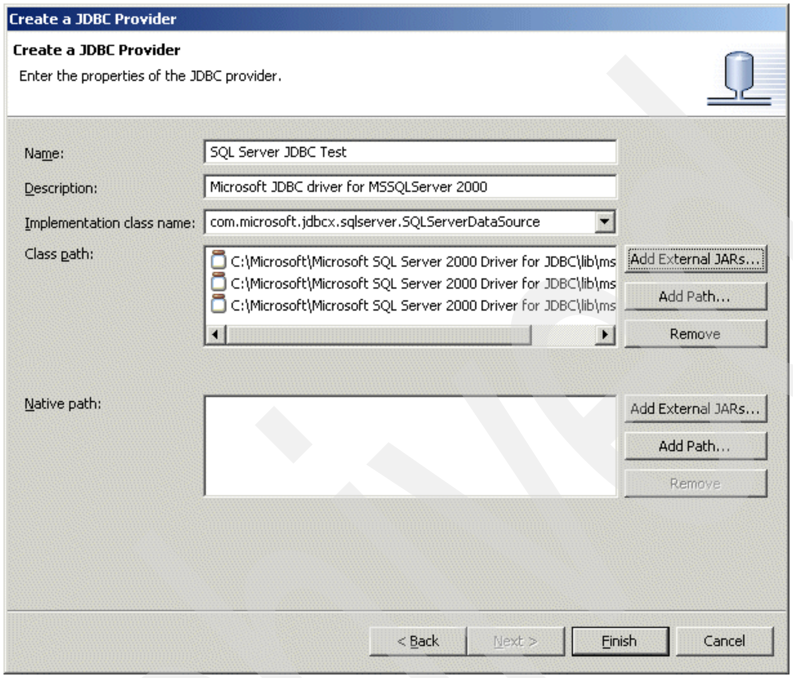
Implementation class name: com.microsoft.jdbcx.sqlserver.SQLServerDataSource

Class path:

Native path:

Figure A-5 JDBC properties

13. Add External Jars. Navigate to where the Microsoft installed mbase.jar, mssqlserver.jar, and msutil.jar. The default location is: C:\Microsoft\Microsoft SQL Server 2000 Driver for JDBC\lib (see Figure A-6).



Create a JDBC Provider

Create a JDBC Provider

Enter the properties of the JDBC provider.

Name: SQL Server JDBC Test

Description: Microsoft JDBC driver for MSSQLServer 2000

Implementation class name: com.microsoft.jdbcx.sqlserver.SQLServerDataSource

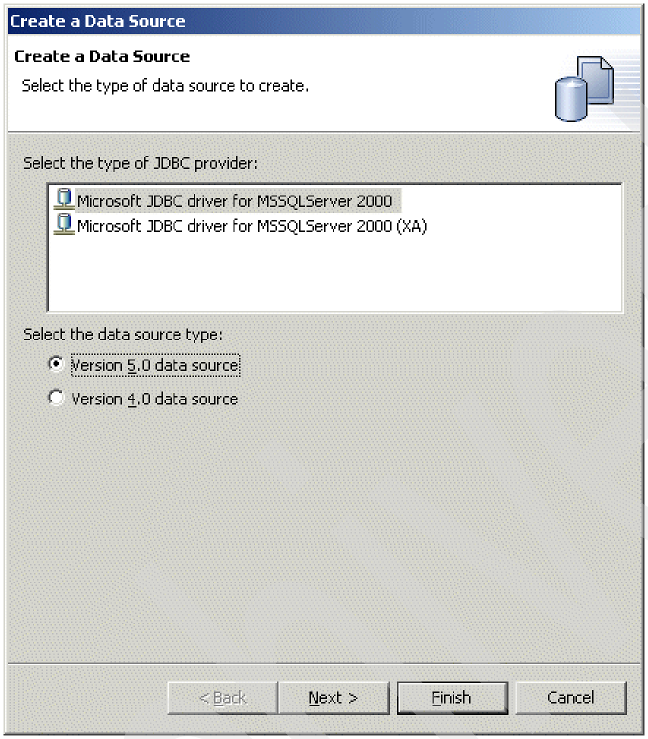
Class path:
C:\Microsoft\Microsoft SQL Server 2000 Driver for JDBC\lib\ms
C:\Microsoft\Microsoft SQL Server 2000 Driver for JDBC\lib\ms
C:\Microsoft\Microsoft SQL Server 2000 Driver for JDBC\lib\ms

Native path:

Figure A-6 JDBC provider new entries

14. Click **Finish**.

15. With the new JDBC provider highlighted, click **Add** under Data source defined in the JDBC provider selected above (see Figure A-7).



Create a Data Source

Create a Data Source

Select the type of data source to create.

Select the type of JDBC provider:

- Microsoft JDBC driver for MSSQLServer 2000
- Microsoft JDBC driver for MSSQLServer 2000 (XA)

Select the data source type:

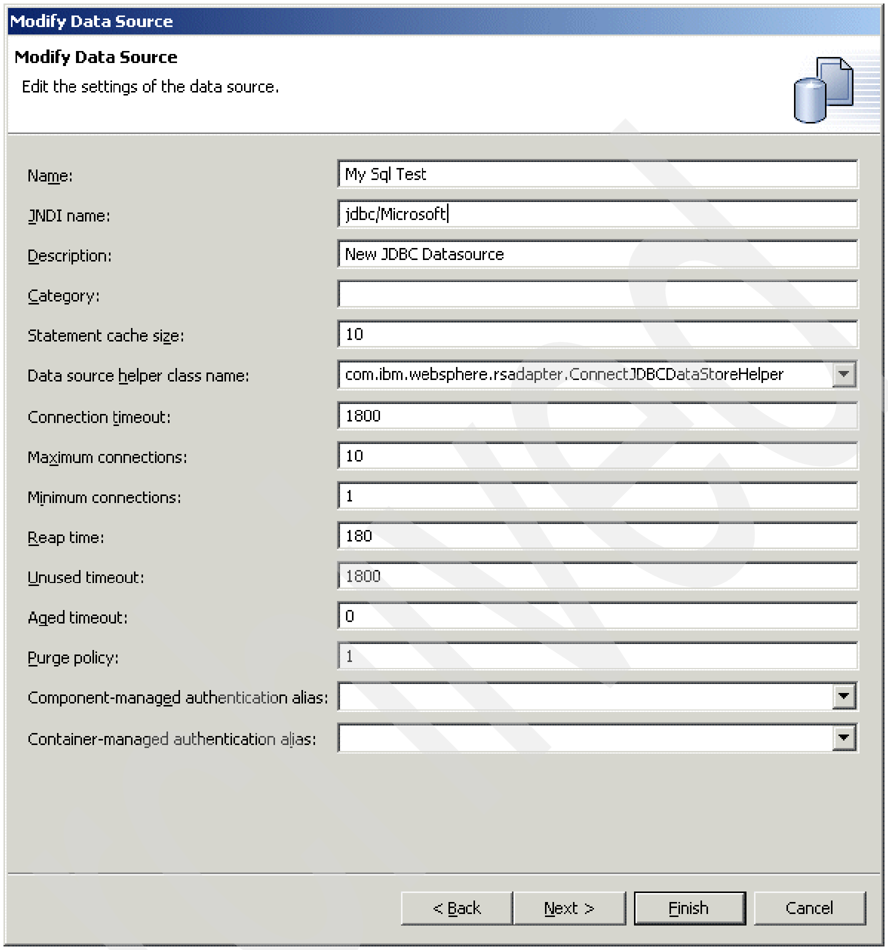
● Version 5.0 data source

○ Version 4.0 data source

Figure A-7 Create a Data Source

16. Accept the defaults. Click **Next**.

17. Give your Data Source a name (see Figure A-8).



Modify Data Source

Modify Data Source

Edit the settings of the data source.

Name:	My Sql Test
JNDI name:	jdbc/Microsoft
Description:	New JDBC Datasource
Category:	
Statement cache size:	10
Data source helper class name:	com.ibm.websphere.rsadapter.ConnectJDBCDataStoreHelper
Connection timeout:	1800
Maximum connections:	10
Minimum connections:	1
Reap time:	180
Unused timeout:	1800
Aged timeout:	0
Purge policy:	1
Component-managed authentication alias:	
Container-managed authentication alias:	

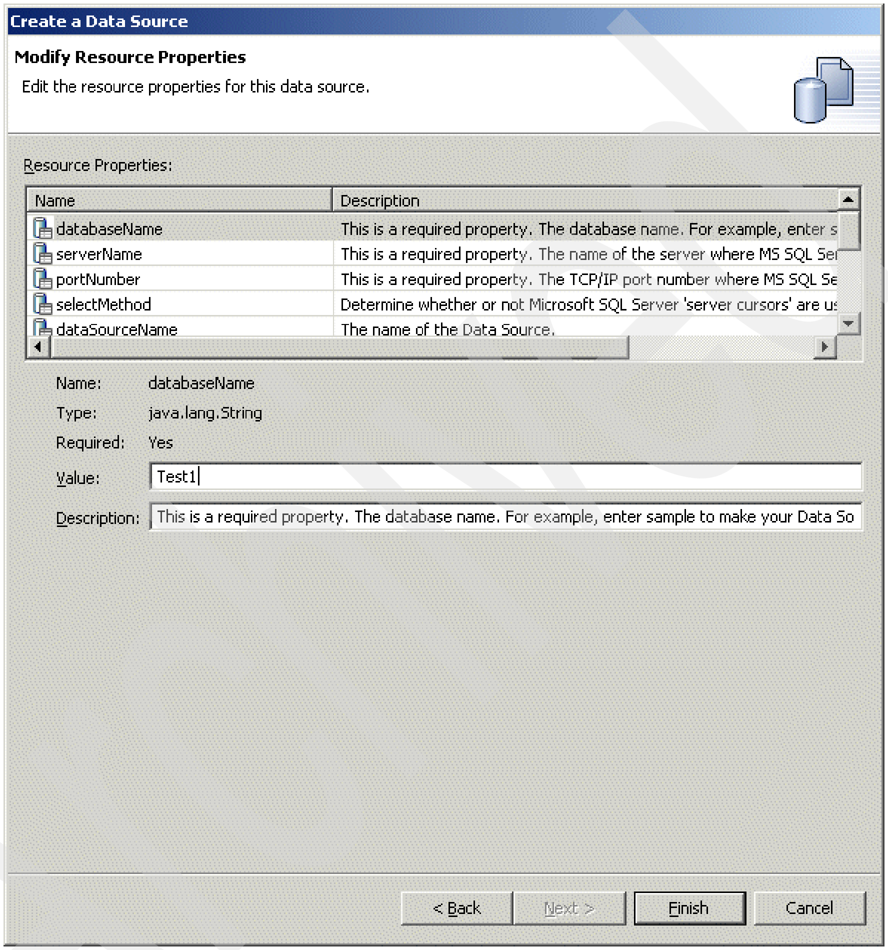
Figure A-8 JNDI name

18. Modify the JNDI name to be something relevant. In our example, we used the name jndi/Microsoft as the JNDI name.

19. Click **Next**.

20. In the value column for databaseName, enter the name of the database that you are using. The example will be using Test1 (see Figure A-9).

21. In the value column for serverName, enter the server name or IP address for your SQL server.



Create a Data Source

Modify Resource Properties

Edit the resource properties for this data source.

Resource Properties:

Name	Description
databaseName	This is a required property. The database name. For example, enter s
serverName	This is a required property. The name of the server where MS SQL Se
portNumber	This is a required property. The TCP/IP port number where MS SQL Se
selectMethod	Determine whether or not Microsoft SQL Server 'server cursors' are u
dataSourceName	The name of the Data Source.

Name: databaseName

Type: java.lang.String

Required: Yes

Value: Test1

Description: This is a required property. The database name. For example, enter sample to make your Data So

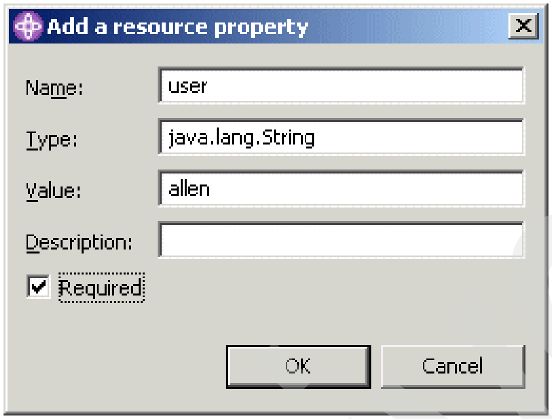
Figure A-9 Database Name

22. Click **Finish**.

23. Under the Resource properties, click **Add**.

24. Add the following values (see Figure A-10):

- Name: user
- Type: java.lang.String
- Value: <the user name for the database access>
- Required: checked



Name: user

Type: java.lang.String

Value: allen

Description:

☑ Required

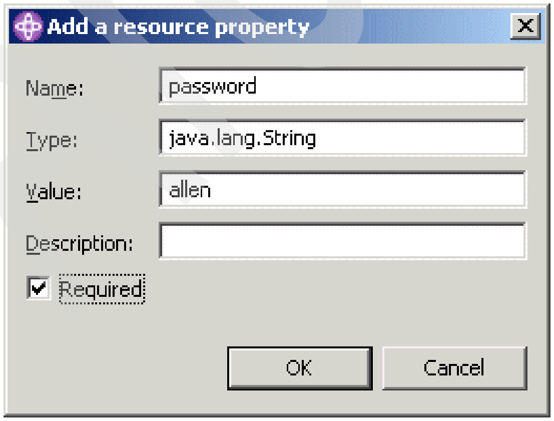
Figure A-10 Add a resource property: user

25. Click **OK**.

26. Under the Resource properties, click **Add**.

27. Add the following values (see Figure A-11):

- Name: password
- Type: java.lang.String
- Value: <the password for the database access>
- Required: checked



Name: password

Type: java.lang.String

Value: allen

Description:

☑ Required

Figure A-11 Add a resource property: password

28. Click **OK**.

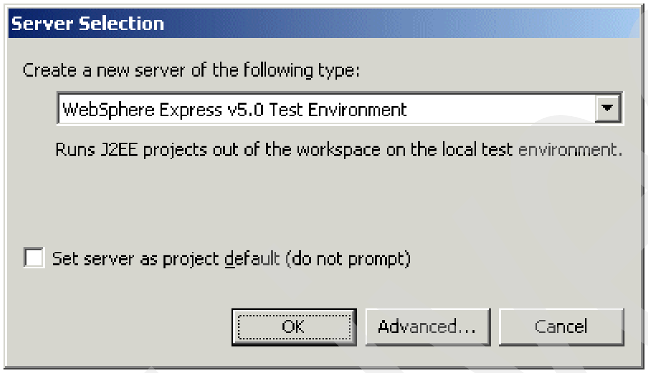
29. Save your server configuration. **File > Save**.

30. Close your server configuration file.

31. Right-click the servlet, DataSourceTest.

32. Click **Run on Server**.

33. Click **Advanced** (see Figure A-12).



Server Selection

Create a new server of the following type:

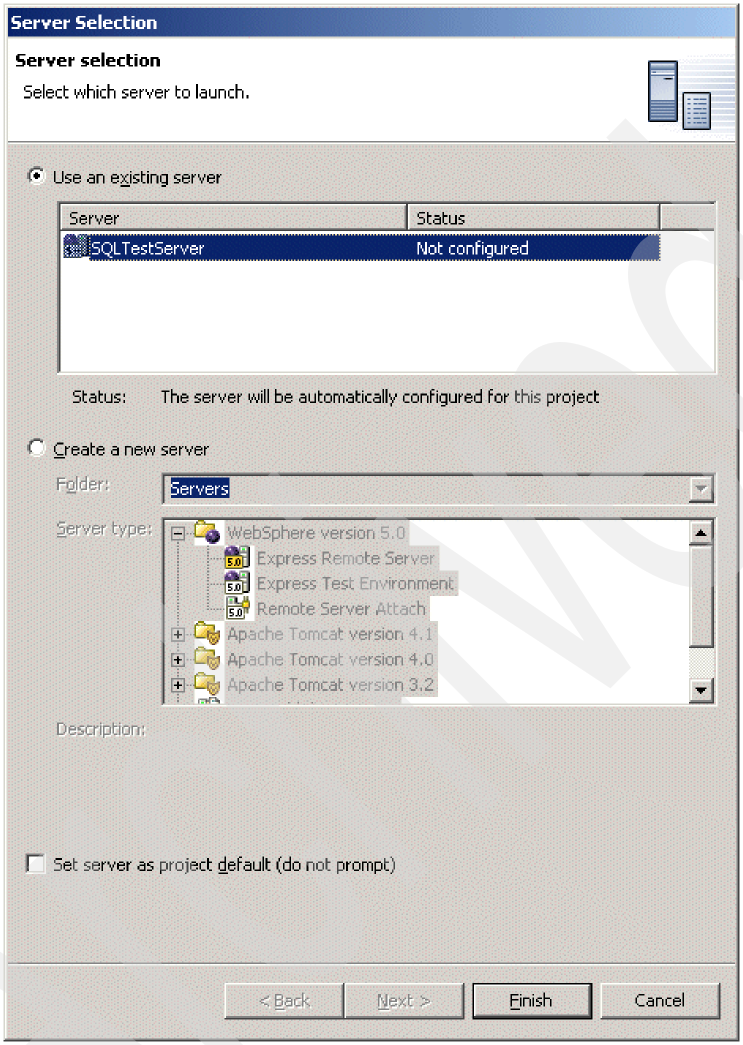
WebSphere Express v5.0 Test Environment

Runs J2EE projects out of the workspace on the local test environment.

☐ Set server as project default (do not prompt)

Figure A-12 Server Selection

34.Click the server that was just created (see Figure A-13).



Server Selection

Server selection

Select which server to launch.

● Use an existing server

Server	Status
SQLTestServer	Not configured

Status: The server will be automatically configured for this project

○ Create a new server

Folder: Servers

Server type:

- WebSphere version 5.0
  - Express Remote Server
  - Express Test Environment
  - Remote Server Attach
- Apache Tomcat version 4.1
- Apache Tomcat version 4.0
- Apache Tomcat version 3.2

Description:

□ Set server as project default (do not prompt)

Figure A-13 Server Selection

35. Click **Finish**.

    The server that you created will be started. Once the server is up, it will display the DataSourceTest servlet in a Web browser.

36. Enter an SQL statement. For example, type:

    ```
    select * from employee
    ```

37. Enter a data source name. For example, type:

    ```
    jdbc/Microsoft
    ```

38. Click **Submit Query**.

The result will display the time it took to perform the database operation. If you click the **Back** function and click **Submit Query**, you will see an increase in performance of the time it took to complete the SQL operation the second time.

Web server configuration for virtual domains and securing WebSphere Application Server - Express

The IBM WebSphere Application Server - Express, Version 5 provides a single server environment where the application server and WebSphere Studio are on the same server, or a double server environment which separates only the application server and WebSphere Studio. The application server does not provide the ability to create multiple virtual domains or sub-domains in the setup.

These issues sometimes are critical in a production environment. In order to overcome such issues, we tested a setup which had a reverse proxy server both with SSL and without SSL support. This setup enabled us to provide virtual domain support and provide additional security to the WebSphere Application Server - Express.

B.1 Configuration details

We took cost into consideration in configuring a multi-server setup, and hence used minimal hardware and free software wherever possible to achieve this setup. The configuration included:

- WebSphere Application Server - Express installed on a Windows 2000 server (Pentium III with 1 GB RAM)
- WebSphere Application Server - Express installed on a Linux server (Pentium II with 128 MB RAM) and a database server (Pentium III with 1 GB RAM).

This setup provided us a scalable environment but still cut down on our investment in expensive software.

The software used for the configuration:

Application server	WebSphere Application Server - Express V5
Reverse proxy server	Apache 1.3.27
Database server	Microsoft SQL Server

The Database server was used in order to demonstrate the database support of the software and is not mandatory for such a setup.

First, we will take you through the steps involved in compiling Apache for your setup and then take you through the configuration involved in setting up the reverse proxy.

It is to be noted that since you will be compiling Apache with SSL support, you should have valid certificates for implementing in a production environment. In our test, we will show you how to create test certificates and execute a test with the same environment. CA in this section refers to Certification Authority.

Important: A valid certificate from one of the certification authorities is mandatory, and it is not advisable to run the site with a self-generated test certificate if you are planning to run the site with SSL support. It should also be noted that, although we have tested the reverse proxy with Apache and on Linux, it should be workable on other platforms with other software which supports similar features.

Apache can be used with SSL either by compiling it with the Apache-SSL patch or by using mod-ssl. In our test environment, we used mod-ssl, as it does most of the configuration when you do an Apache install. If you were to use Apache-SSL, then you would have to enter most of the configuration details in the httpsd.conf yourself. However, we will show the configuration in the httpd.conf, even though we used mod-ssl with Apache.

You can either install OpenSSL using the RPM Package Manager (RPM) available with Red Hat or compile the source. We will list in detail how to compile the source for OpenSSL.

The source packages required are listed in Figure B-1.

Table B-1 Source packages

Software package	Version
Apache	1.3.27
mod_ssl	2.8.12
OpenSSL	0.96h or above

We assume that Perl, sh, gzip, make, gcc (version 3.x), and tar are already installed in your system. These packages are important for compiling packages. In case you want to install these packages, you would find all of them in your Red Hat CD. So you can go to the install packages option in Red Hat and install these packages.

After you obtain all the sources, save them in the same directory for ease of maintenance and as some packages will need access to other packages's sources and it's easier if they are all in the same location.

In our example, all the downloaded source files were under /root/downloads.

These were the files that we downloaded:

- apache_1.3.27.tar.gz from http://www.apache.org
- mod_ssl-2.8.12-1.3.27.tar.gz from http://www.apache.org
- openssl-0.9.6h.tar.gz from http://www.openssl.org

You must un-tar these files to their respective directories. Do all the steps mentioned below after logging in as root.

B.2 Host details

In this section, we will discuss the host details.

This is the host on which Apache (Reverse proxy) is installed:

IP Address	9.24.104.11
HostName for SSL	testing.itso.raleigh.ibm.com
HostName for non-ssl	testing1.itso.raleigh.ibm.com

This is the host on which WebSphere Application Server - Express, V5 (Application Server) is installed:

IP Address	9.24.104.173
Port on which application is running	7080

B.3 Uncompressing Apache source

In this section, we will uncompress the Apache source. Perform the following instructions:

1. Run the command

```
downloads ]# gzip -d apache_1.3.27.tar.gz
```

2. You will be returned to the # prompt after this command has unzipped the file. Run the following command

```
downloads ]# tar -xvpf apache_1.3.27.tar
```

3. This command would now untar all the respective files to a directory called apache_1.3.27

B.4 Uncompressing mod_ssl source

In this section, we will uncompress the mod_ssl source. Perform the following instructions:

1. Run the command:

```
downloads ]# gzip -d mod_ssl-2.8.12-1.3.27.tar.gz
```

2. You will be returned to the # prompt after this command has unzipped the file. Run the following command:

```
downloads ]# tar -xvpf mod_ssl-2.8.12-1.3.27.tar
```

3. This command would now untar all the respective files to a directory called **mod_ssl-2.8.12-1.3.27**

B.5 Uncompressing OpenSSL source

In this section, we will uncompress the OpenSSL source. Perform the following instructions:

1. Run the command:

   ```
   downloads ]# gzip -d openssl-0.9.6h.tar.gz
   ```

2. You will be returned to the # prompt after this command has unzipped the file. Run the following command:

   ```
   downloads ]# tar -xvpf openssl-0.9.6h.tar
   ```

3. This command will now untar all the respective files to a directory called **openssl-0.9.6h.**

B.6 Compilation and configuration process

Once all the source files are available, it is now important to compile the applications *only* in the order that is mentioned below, since there are application dependencies.

B.6.1 OpenSSL compilation

You are assumed to be in the directory where all the sources were downloaded. In our case, this directory is /root/downloads.

1. Change to the source directory for OpenSSL:

   ```
   # cd /root/downloads/openssl-0.9.6h
   ```

2. Run the config script to automatically configure your system

   ```
   # ./config
   ```

   There are other options available with ./config and you can tune it based on your system requirements. More information on this can be found at www.openssl.org. This command will build a file called Makefile with the requisite information related to your system and configure the system to install OpenSSL in its default directory, for example, /usr/local/ssl.

3. Build OpenSSL by running:

   ```
   # make
   ```

This process will build the necessary binaries and configuration files which will be used when you install OpenSSL.

4. After a successful build, the libraries should be tested. Run the following command to test them:

```
# make test
```

5. Everything should run OK. If not, check the FAQ at http://www.openssl.org and resolve any issues. Now, install OpenSSL by performing the following command:

```
# make install
```

This will now install OpenSSL to /usr/local/ssl and the binary (executable of OpenSSL) to /usr/bin. This completes the installation of OpenSSL.

Configuration of OpenSSL and creation of test certificates

OpenSSL gives you the option of creating a test certificate pair and a CA test certificate for certification using specific commands. You can either follow the steps mentioned in this section, or use the standard command provided by mod_ssl in Apache, which allows you to create a certificate using a script. In our example, we created the certificates first, and then proceeded to compiling Apache with mod_ssl.

1. Open the OpenSSL config file. In our example, since we had chose to install OpenSSL in the default directory, the configuration file will be available at /usr/local/ssl. If you chose to install OpenSSL in some other directory, you will find the configuration file there. Perform the following:

```
# vi /usr/local/ssl/openssl.cnf
```

This will open the configuration file for editing.

2. Go to the specific section - CA_default. Do not change the default_ca attribute. Change the other parameters as per your setup. Based on our setup, we did the following:

Table B-2 Parameter settings

Variable in file	Parameter supplied by us	Use of this variable
dir	/usr/local/openssl	Where openssl and related info is kept
serial	$dir/ca.db.serial	The current serial number
crl	$dir/crl.pem	The current CRL
private_key	$dir/private/ca.key	The CA's private key
RANDFILE	$dir/ca.db.rand	private random number file

Variable in file	Parameter supplied by us	Use of this variable
certs	$dir/certs	Where the issued certs are kept
crl_dir	$dir/crl	Where the issued crl are kept
database	$dir/ca.db.index	database index file.
new_certs_dir	$dir/ca.db.certs	default place for new certs
certificate	$dir/certs/ca.crt	The CA certificate
serial	$dir/ca.db.serial	The current serial number
crl	$dir/crl.pem	The current CRL
private_key	$dir/private/ca.key	The CA's private key
RANDFILE	$dir/ca.db.rand	private random number file

3. Next, edit this section:

   [req_distinguished_name]

Table B-3 Parameter settings

Variable in file	Parameter supplied by us	Use of this variable
countryName_default	US	This is the country to which your Web site belongs.
stateOrProvinceName_default	North Carolina	This mentions the state to which your Web site belongs.
localityName_default	Research Triangle Park	This mentions the locality to which your Web site belongs.
0.organizationName_default	IBM	This mentions the default organization name to which this Web site belongs.
organizationalUnitName_default	ITSO	This mentions the default organizational unit name to which this Web site belongs
commonName_default	testing.itso.ral.ibm.com	This mentions the Web site name. Do not mention any other name than the Web site that you are going to use the SSL certificate.
emailAddress_default	webmaster@itso.raleigh.ibm.com	The default email address used by your site.

It should be noted that there are other parameters in this file which have not been filled in by us. They are all optional parameters we chose not to change. If you feel it necessary to complete these parameters then do so.

4. Now we must create the certificates. We will generate a private key first for the server. Then, execute the following commands in the OpenSSL directory.

   > **Note:** In our example, we executed these commands in the /usr/local/ssl directory. Executing this command from this directory is just for easy maintenance purposes. You could also create a temp directory and execute the following commands from there.

   ```
   # openssl genrsa -des3 -out server.key 1024
   ```

   You will be prompted for a password here. Enter a password which you will easily remember. After entering the password, you will be asked to re-enter the password for validation.

5. Now you need to generate a request with the private key. This is for the public key which you will certify later.

   ```
   # openssl req -new -key server.key -out server.csr
   ```

   Again you will be asked for a password. Enter the same password that you entered above. After entering this password, you will be asked a set of questions. These questions will be similar to the questions you entered above in the openssl.cnf file.

6. The default entries would appear in the bracket after the question and you can either press Enter to accept that entry, or enter a different parameter. It is noted that these are the parameters that will be used to create the certificate for testing, so enter only your Web site URL (fully-qualified name of your Web server) for which this certificate is going to be used in the Common Name parameter.

7. After entering all the relevant information, you will be asked for a challenge password. It is best to leave the challenge password blank.

   Once you have created the server.csr file, you will have to create requisite certificates for the dummy CA. It should be noted that if you have a valid certificate with you, then you do not need to run through this entire certificate generation section.

8. Now generate a private key for your CA. Perform the following command:

   ```
   # openssl genrsa -des3 -out ca.key 1024
   ```

   You are asked for a password that will secure your CA. It is important to remember this password since you will be prompted for the same when certifying your server certificate.

9. To make the CA private key a complete usable CA, you will have to execute the following command:

   ```
   # openssl req -new -x509 -days 999 -key ca.key -out ca.crt
   ```

   You will be prompted for a password. This one is the same that you entered in step 8. You will be asked a series of questions similar to the one when you created the file server.csr. Enter details which are different from the one you created the server.csr file in step 5. Remember not to press Enter when the questions are prompted without entering any details as that will take the default entries which might be the one applicable to the server.csr file. Do not enter the same details that you entered while creating the server.csr file.

   Make sure you copy all the csr and crt files to the certs directory, and all the key files to the private directory.

10. You can do the same by executing the following commands from the OpenSSL directory. In our example, /usr/local/ssl.

   ```
   # mv *.crt certs
   # mv *.csr certs
   # mv *.key private
   ```

   You will have a ca.crt and server.csr file in the certs directory and a server.key and ca.key file in the private directory.

   Now you can use this CA to sign server certificates to secure your Web sites. You will need one for each domain that you plan to test. We tested only one domain with SSL, hence, we created only one server.csr file and one CA for the same.

Signing the certificate

In this section, we will sign the certificate. Perform the following steps:

1. Create the following files under the OpenSSL directory.

   ```
   touch /usr/local/ssl/ca.db.index
   touch /usr/local/ssl/ca.db.serial
   ```

2. Create the following directory:

   ```
   mkdir /usr/local/ssl/ca.db.certs
   ```

   Create the ca.config file under the OpenSSL directory. The contents of the file will look similar to Example B-1.

Example: B-1   ca.config file example

```
[ ca ]
default_ca            = CA_own
[ CA_own ]
dir                   = /usr/local/ssl/
certs                 = /usr/local/ssl/certs
```

```
new_certs_dir          = /usr/local/ssl/ca.db.certs
database               = /usr/local/ssl/ca.db.index
serial                 = /usr/local/ssl/ca.db.serial
RANDFILE               = /usr/local/ssl/my_ca.db.rand
certificate            = /usr/local/ssl/certs/ca.crt
private_key            = /usr/local/ssl/private/ca.key
default_days           = 365
default_crl_days       = 30
default_md             = md5
preserve               = no
policy                 = policy_anything
[ policy_anything ]
countryName            = optional
stateOrProvinceName    = optional
localityName           = optional
organizationName       = optional
organizationalUnitName  = optional
commonName             = supplied
emailAddress           = optional
```

Once you created this file in the OpenSSL directory (in our example /usr/local/ssl), you can now sign the certificate.

3. Execute the following command:

   ```
   # openssl ca -config ca.config -out /usr/local/ssl/certs/server.crt
   -infiles /usr/local/ssl/certs/server.csr
   ```

   You are prompted for a password, which is the password you had used for the generation of the CA file, in Step 8 on page 120. After that you will be prompted for verification of signature of this file and whether you would like to commit them. You must say yes to these questions.This will now sign the certificate and create a server.crt file in the same directory.

4. Copy the server.crt file to the certs directory under your OpenSSL installation. In our example, we executed the following command:

   ```
   # cp server.crt /usr/local/ssl/certs
   ```

5. Now for verifying whether your certificate generation was correct, execute the following command without changing your directory:

   ```
   # openssl verify -CAfile /usr/local/ssl/certs/ca.crt
   /usr/local/ssl/server.crt
   ```

   You should get an OK prompt after the certificate is verified. This means that you have the certificates ready for usage in an SSL environment.

B.6.2 Compiling mod_ssl

In compiling the mod_ssl, the first step is to make sure that the Apache source tree understands where the SSL related files are located and to make itself SSL enabled. Complete the following steps:

1. Apply mod_ssl to the Apache source tree. Change to the mod_ssl source directory:

   ```
      # cd /root/downloads/mod_ssl-2.8.12-1.3.27
   ```

2. Now configure mod_ssl to make the requisite Makefile and make Apache aware of where the SSL related files are located:

   ```
      # ./configure --with-apache=/root/downloads/apache_1.3.27 \
   --with-ssl= /usr/local/ssl --with crt=/usr/local/ssl/certs/server.crt \
   --with-key=/usr/local/ssl/private/server.key
   ```

   This command will now exit after creating the requisite files in the requisite directories. The mod_ssl patch is now applied to Apache with the certificate information stored.

B.6.3 Compiling and configuring Apache

The following steps will demonstrate how to compile and install Apache in this setup and subsequently configure it for SSL and a reverse proxy.

1. Change to the Apache source directory.

   ```
   # cd /root/downloads/apache_1.3.27
   ```

2. Execute the following command for compiling Apache with proxy and ssl support under the /usr/local/apache directory. If you want it installed in some other directory, then change the --prefix option to the directory you want Apache installed in.

   ```
   # SSL_BASE=/usr/local/openssl \
   ./configure \
   --enable-module=so \
   --enable-module=proxy \
   --enable-shared=proxy \
   --enable-module=rewrite \
   --enable-shared=rewrite \
   --prefix=/usr/local/apache \
   --enable-module=ssl \
   --enable-module=so \
   ```

   This will configure Apache with the requisite module support. Now you will have to create the binaries and make it ready for execution.

3. Run the following command so that Apache will create the requisite binaries for its operations.

   ```
   # make
   ```

   This step will create the requisite binaries required by Apache for its operations.

> **Note:** In case you did not follow the steps mentioned under “Configuration of OpenSSL and creation of test certificates” on page 118, then you can create the certificate using the scripts provided by mod_ssl.

4. Run the following command to create the certificates. If you followed the steps mentioned under “Configuration of OpenSSL and creation of test certificates” on page 118, then ignore this step and move to the next one.

   ```
   # make certificate test
   ```

   You will be prompted to answer some questions about your certificate, which are similar to the questions in “Configuration of OpenSSL and creation of test certificates” on page 118.

5. Install the Apache server. Run the following command:

   ```
   # make install
   ```

   This will install all the requisite binaries of Apache under /usr/local/apache (if you have specified a different directory, it will be installed under that directory).

   Prior to starting Apache, make sure the configurations in the httpd.conf file located in the /usr/local/apache/conf are correct. (If you have installed Apache under a different directory, look for a conf directory under the directory you installed Apache).

6. Perform the following command:

   ```
   # cd /usr/local/apache/conf
   ```

7. Edit the httpd.conf file by running the following command:

   ```
   # vi httpd.conf
   ```

8. Make sure the following configuration is as per your setup. All the details mentioned in Example B-2 are specific to our setup.

Example: B-2 httpd.conf file

```
LoadModule rewrite_module     libexec/mod_rewrite.so
LoadModule proxy_module       libexec/libproxy.so

<IfDefine SSL>
AddModule mod_ssl.c
```

```
</IfDefine>

Port 80

<IfDefine SSL>
Listen 80
Listen 443
</IfDefine>

<IfDefine SSL>
<IfModule mod_ssl.c>
SSLPassPhraseDialog  builtin
SSLSessionCache         dbm:/usr/local/apache/logs/ssl_scache
SSLSessionCacheTimeout  300
SSLMutex  file:/usr/local/apache/logs/ssl_mutex
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
SSLLog      /usr/local/apache/logs/ssl_engine_log
SSLLogLevel info
</IfModule>
<IfDefine SSL>

<VirtualHost _default_:443>
ServerName testing.itso.raleigh.ibm.com
ServerAdmin root@wsexplnx.itso.ral.ibm.com
ErrorLog /usr/local/apache/logs/error_log
TransferLog /usr/local/apache/logs/access_log
ProxyRequests On
ProxyPass / http://testserver.itso.raleigh.ibm.com:7080/datasourceTest/
ProxyPassReverse /
http://testserver.itso.raliegh.ibm.com:7080/datasourceTest/
NoCache *
SSLEngine on
SSLCipherSuite
ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLCertificateFile /usr/local/apache/conf/ssl.crt/server.crt
SSLCertificateKeyFile /usr/local/apache/conf/ssl.key/server.key

<Files ~ "\.(cgi|shtml|phtml|php3?)$">
    SSLOptions +StdEnvVars
</Files>

<Directory "/usr/local/apache/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>

SetEnvIf User-Agent ".*MSIE.*" \
         nokeepalive ssl-unclean-shutdown \
         downgrade-1.0 force-response-1.0
```

```
</VirtualHost>
</IfDefine>
```

We have defined a second host accessible from the same machine but which does not use SSL. This is accessible on the same backend server, using the same reverse proxy in the front with the same IP address, but on an insecure port. In the event, you want to run multiple virtual hosts with SSL, then you will need one certificate pair for each domain that you plan to run.

```
NameVirtualHost 9.24.104.11:80
<VirtualHost 9.24.104.11:80>
ServerName testing1.itso.raleigh.ibm.com
Port 80
ErrorLog logs/test.error.log
CustomLog logs/test.access.log common
LogLevel debug
ProxyRequests On
ProxyPass / http://testserver.itso.raleigh.ibm.com:7080/datasourceTest/
ProxyPassReverse /
http://testserver.itso.raliegh.ibm.com:7080/datasourceTest/
NoCache *
</VirtualHost>
```

This completes the configuration of Apache as a reverse proxy. If necessary, please perform other changes in the Apache configuration file specific to your site's performance and security needs. Since this Redbook is not about Apache, the steps for making these changes are not documented.

9. You can now start Apache by running the command:

   ```
   # /usr/local/apache/bin/apachectl sslstart
   ```

   In case you have installed Apache in a different directory, you will have to start if from there.

10. Now you can access the Web site by typing in the URL that you have defined in the ServerName in the configuration above. In our example, we can access the sites:

   ```
   https://testing.itso.raleigh.ibm.com
   http://testing1.itso.raleigh.ibm.com
   ```

   It should be noted that one site is secure, and the other is a non-secure site. However, both point to the same application server in the backend. In case you have two different applications running on the same server in the backend, you could configure the proxy server accordingly. The only lines that would change in the configuration file mentioned above are:

   ```
   ProxyPass / http://testserver.itso.raleigh.ibm.com:7080/datasourceTest/
   ProxyPassReverse /
   http://testserver.itso.raliegh.ibm.com:7080/datasourceTest/
   ```

Just to briefly explain the comments above, ProxyPass takes two arguments, one is the sourceURL that needs to be processed, and the other is the finaldestination URL / server that the sourceURL needs to fetch data.

For example, we want users to access https://testing.itso.raleigh.ibm.com, however, the content is supposed to come from http://testserver.itso.raleigh.ibm.com. Therefore, we have just given a / as the source URL. This indicates that if someone tries to hit the URL https://testing.itso.raleigh.ibm.com without any further directories, the source from that request is to come from http://testserver.itso.raleigh.ibm.com:7080/datasourceTest/.

11. If you want to stop Apache, you must execute the following command:

```
# /usr/local/apache/bin/apachectl stop
```

If Apache is installed in a different directory, you will have to stop if from there.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see "How to get IBM Redbooks" on page 130.

- *WebSphere Application Server - Express, Version 5.0 Handbook*, SG24-6555

Other resources

These publications are also relevant as further information sources:

- *WebSphere Application Server - Express, Version 5.0 Installation Guide*
- *WebSphere Application Server - Express, Version 5.0 Getting Started Guide*

Referenced Web sites

These Web sites are also relevant as further information sources:

- CVSNT Enhanced CVS Server

  http://www.cvsnt.com
- Developersworks forum

  http://www-106.ibm.com/developerworks/forums/dw_lforums.jsp
- WebSphere Developer Domain Support - forums

  http://www-1.ibm.com/support/search.wss?tc=&rs=493&apar=exclude&q=WebSphere&dc=D700
- IBM Learning Services

  http://www.ibm.com/services/learning
- The Apache Software Foundation

  http://www.apache.org
- OpenSSL project

http://www.openssl.org

- Microsoft support

  http://support.microsoft.com

- IBM WebSphere products

  http://www.ibm.com/WebSphere

- WebSphere Application Server - Express education

  http://www-3.ibm.com/software/webservers/appserv/express/education/

- IBM PartnerWorld Software Development

  http://www.developer.ibm.com/websphere/was.html

- Eclipse technology

  http://www.eclipse.org

- Computer Generated Solutions - WebSphere Application certification

  http://cgse3.cgselearning.com/zone-websphere.html

- Verisign:

  http://www.verisign.com

- GTE Cybertrust:

  http://www.cybertrust.gte.com

- Entrust:

  http://www.entrust.com

How to get IBM Redbooks

You can order hardcopy Redbooks, as well as view, download, or search for Redbooks at the following Web site:

ibm.com/redbooks

You can also download additional materials (code samples or diskette/CD-ROM images) from that site.

IBM Redbooks collections

Redbooks are also available on CD-ROMs. Click the CD-ROMs button on the Redbooks Web site for information about all the CD-ROMs offered, as well as updates and formats.

Index

IBM Redbooks

WebSphere Solution Guide: WebSphere Application Server - Express, Version 5.0

(0.2"spine)
0.17"<->0.473"
90<->249 pages

WebSphere Solution Guide: WebSphere Application Server - Express, Version 5.0

Positions Application Server - Express within the stages of e-business adoption

Helps you to plan and implement for Windows and Linux

Illustrates quick development and deployment of business applications

The WebSphere Solution Guide: WebSphere Application Server - Express, Version 5 was developed to help the marketing and support efforts for WebSphere Channels Enablement.

This IBM Redbook represents one of two documents, together known as the Solution Guide for WebSphere Channel Enablement. The Solution Guide includes WebSphere channel-ready documentation in the form of a marketing and sales guide and an implementation and integration guide.

The purpose of this implementation and integration guide is to simplify the planning and implementation of IBM Business Partners' and independent software vendors' e-business applications that are enabled for WebSphere Application Server - Express. It also takes into consideration the “Whole Product Concept”, which incorporates everything the customer needs to achieve the business goals that drive its purchase decisions, including consultation, design, configuration, implementation, OEM products and services, and on-going support.

In this book, we review and execute step-by-step instructions to set up and configure WebSphere Application Server - Express, Version 5 running Windows 2000 and Red Hat Linux 8.0. We also discuss how to successfully use the Application Server - Express wizard to create and customize a business application for your WebSphere Application Server - Express, Version 5 environment.

SG24-6925-00 ISBN 0738429481