Integrating Host Applications with e-business Portals

- Access legacy application content from personalized portals
- Extend your Host Publisher applications as new portal content
- Deploy and use Host On-Demand portlets

Juan R. Rodriguez
Rodrigo P. Arantes
Luis E. Cabral
Guohui Du
Fabiola Rodriguez

ibm.com/redbooks
Integrating Host Applications with e-business Portals

June 2002
Take Note! Before using this information and the product it supports, be sure to read the general information in “Notices” on page v.

First Edition (June 2002)

This edition applies to WebSphere Host On-Demand V6.02 (available through the Host Access Client Package), WebSphere Host Publisher V4.0, and WebSphere Portal Server V2.1 for multiplatforms.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8  Building 662
P.O. Box 12195
Research Triangle Park, NC 27709-2195

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

© Copyright International Business Machines Corporation 2002
Note to U.S Government Users - Documentation related to restricted rights - Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.
# Contents

## Notices

Trademarks

## Preface

The team that wrote this Redpaper

Notice

Comments welcome

## Chapter 1. An architectural view of host integration for e-business portals

1.1 Portal environment

1.1.1 IBM WebSphere software products

1.2 A conceptual view of host integration using portal server

1.2.1 Host On-Demand portlet

1.2.2 IBM WebSphere Host Publisher portlet

1.3 Host integration scenarios

1.3.1 Business-to-employee (B2E)

1.3.2 Business-to-consumer (B2C)

1.3.3 Business-to-business (B2B)

1.3.4 Presentation services

1.3.5 Host access services

1.3.6 Personalization services

1.3.7 Services for security and administration

1.3.8 Support of wireless devices

1.4 Architectural considerations

1.4.1 Understanding the integration needs

1.4.2 Infrastructure considerations

1.4.3 Software considerations

1.4.4 Development considerations

1.4.5 Systems management considerations

## Chapter 2. Integrating WebSphere Host On-Demand with Portal Server

2.1 Host On-Demand portlet

2.2 Major HOD components

2.2.1 HTTP Server

2.2.2 HOD Server

2.2.3 LDAP support

2.2.4 HOD clients

2.2.5 Deployment Wizard

2.2.6 Screen Customizer interface

2.2.7 Language support

2.3 Client receive method

2.3.1 Host On-Demand cached client

2.3.2 Host On-Demand download client

2.4 Client execution (run) method

2.4.1 Embedded mode

2.4.2 Separated mode

2.5 SSL solutions

2.5.1 Firewall concepts

2.5.2 Secure Sockets Layer (SSL)
Chapter 3. Installing and using the WebSphere Host Publisher portlet

3.1 Host Publisher portlet ...................................................... 50
  3.1.1 Limitations .......................................................... 51
3.2 Major Host Publisher components ..................................... 51
3.3 What’s new in Host Publisher V4.0 ..................................... 52
3.4 Installation of Host Publisher V4.0 .................................... 52
  3.4.1 DB2 ................................................................. 53
  3.4.2 WebSphere Application Server V4.0 ............................ 53
  3.4.3 Host Publisher Server ............................................ 54
  3.4.4 Host Publisher Studio ............................................ 60
  3.4.5 WebSphere Studio Application Developer ...................... 60
3.5 Host Publisher applications ............................................. 61
  3.5.1 Creating Host Publisher applications .......................... 61
  3.5.2 Using WebSphere Studio Application Developer for further development ........................................... 65
  3.5.3 Other scenarios using macro looping, conditionals and chaining ......................................................... 70
  3.5.4 Recommendations ................................................ 78
3.6 Deploying HostPublisher applications ................................ 79
  3.6.1 Installing the Host Publisher application ....................... 79
  3.6.2 Installing the Host Publisher portlet in the Portal Server .... 85
  3.6.3 Configure the Host Publisher portlet to access applications ................................................................. 87
3.7 Security and administration ............................................. 91
3.8 Problem determination and troubleshooting .......................... 92

Related publications .......................................................... 97
IBM Redbooks ................................................................. 97
Referenced Web sites ........................................................ 97
How to get IBM Redbooks .................................................... 98
IBM Redbooks collections .................................................... 98
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:

- Redbooks(logo)™
- AIX®
- AS/400®
- CICS®
- DeveloperWorks™
- DB2®
- IBM®
- IMS™
- Informix®
- MQSeries®
- OS/2®
- OS/390®
- OS/400®
- Perform™
- Redbooks™
- S/390®
- SecureWay®
- SP™
- VisualAge®
- WebSphere®
- z/OS™

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

- Lotus®
- Domino™

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, 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.
- 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

This Redpaper will help you integrate existing host enterprise applications so that they can be accessed from portlets using the IBM WebSphere Portal Server product. Host applicationportlet support is provided by the current releases of IBM WebSphere Host On-Demand and IBM WebSphere Host Publisher products.

In this Redpaper, you will find step-by-step examples showing ways to rapidly integrate your host enterprise applications, such as database, 3270, 5250, and VT applications, into an IBM WebSphere Portal Server environment by implementing new and enhanced capabilities incorporated in the current releases of IBM WebSphere Host On-Demand and IBM WebSphere Host Publisher. You will find numerous scenarios describing recommended ways to integrate your legacy applications using the portlet support and sample portlets provided by these products.

A basic knowledge of Java technologies such as servlets, JavaBeans, and JavaServer Pages (JSPs), as well as HTML and XML markup languages and the terminology used in Web publishing, is assumed.

The team that wrote this Redpaper

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

Juan R. Rodriguez is a Consulting IT professional and Project Leader at the IBM ITSO Center, Raleigh. He received his M.S. degree in Computer Science from Iowa State University. He writes extensively and teaches IBM classes worldwide on such topics as networking, Web technologies, and data security. Before joining the IBM ITSO, he worked at the IBM laboratory in the Research Triangle Park (North Carolina, USA) as a designer and developer of networking products.

Rodrigo P. Arantes is a Technical Sales Specialist for the IBM Software Group in Rio de Janeiro, Brazil. He has two years of experience in the Host Integration area, and six years in IT and UNIX solutions. He holds a degree in Computer Science from Universidade do Estado do Rio de Janeiro (UERJ). His areas of expertise include WebSphere Host On-Demand, Host Publisher, Portal Server, Edge and Transcoding Publisher products.

Luis E. Cabral is an Advisory Technical Sales Specialist for the IBM Software Group in New York City. He has over five years of experience in the IT field. He has three years of experience in the field of web to host integration. His areas of expertise include Host On-Demand, Host Publisher, and Transcoding Publisher. He holds a degree in Computer Science from Hunter College and an MBA in Finance and Information Systems from Fordham University.
Guohui Du is a certified consultant with IBM UBG in Germany. He joined IBM in 1996, with an extensive background in object technology and application development. He has more than 10 years of experience in the IT architecture field. He received a Dr.-Ing. degree from University of Stuttgart in knowledge-based systems. His areas of expertise include advising corporate IT clients on emerging new technologies, object and component architecture, application development, and infrastructure design for e-business.

Fabiola Rodriguez is an IT Specialist for the IBM Global Services group in Caracas, Venezuela. She has three years of experience with Java and development of Web applications, and two years of experience with Host On-Demand applications. She holds a degree in Computer Engineering from Simon Bolivar University in Venezuela. Her areas of expertise include WebSphere Application Server, VisualAge for Java, Host On-Demand, and MQSeries products.

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

Gail Christensen, Margaret Ticknor, Jeanne Tucker
International Technical Support Organization, Raleigh Center

Julie Henderson, Hollins Williams, Rick Hardison, Sherri Midyette
IBM Research Triangle Park, North Carolina, USA

Charlotte Davis, Gabriel Cohen
IBM Research Triangle Park, North Carolina, USA

Notice

This publication is intended to help I/T specialists and architects to integrate host applications using IBM WebSphere Portal Server. The information in this publication is not intended as the specification of any programming interfaces that are provided by IBM WebSphere Host On-Demand, IBM WebSphere Host Publisher and IBM WebSphere Portal Server products. See the PUBLICATIONS section of the IBM Programming Announcement for these products for more information about what publications are considered to be product documentation.

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this Redpaper 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 the address on page ii.
Chapter 1. An architectural view of host integration for e-business portals

Today, we are surrounded by information and corporate data, for example business-related data, documents, e-mail, video, newspapers, and many others. As a way to better handle this information, many companies use portals, where an end user can find a great deal of information while also accessing other business applications.

The majority of business-critical data and applications still reside on the IBM host systems, such as CICS, IMS and other mainframe applications. As e-business evolves from simple Web publishing to more sophisticated transaction-oriented integration, Web-to-host integration becomes an essential part of many e-businesses. IBM offers corporate customers two host integration solutions, Host On-Demand and Host Publisher, to address different integration needs.

This chapter provides a conceptual overview of integrating host applications for e-business portals. We discuss the major components and services you need to implement typical host integration for portals, and consider several points that must be addressed during the architecture phase of host integration applications with portals.
1.1 Portal environment

In the Web environment we can distinguish between two types of applications:

- Web applications implemented for internal services in an intranet.
- Web applications implemented for services available to all users on the Internet.

These two application types are of a different nature and also intended for different types of users, but they have been put through the same consistent application design. These applications not only provide the implemented functions and features they support, but also have the required flexibility for easy maintenance and any other future requirements.

Portal technology is constantly evolving and therefore many already established applications on the Web will need to be enhanced to adapt to this new technology. There are many options to help you design an application with a high degree of flexibility as well as enhancing your application for portal use. Demands for applications are changing as new technologies are emerging every day and robust Web applications must be able to adapt to these new technologies to keep up with competition.

A portal provides a single point of access to diverse information and applications, secure interactions, a customized interface, personalized content, and much more.

Integrating host applications to e-business is an integral part of many companies. A high percentage of business-critical data and applications is on IBM host systems, such as AS/400, OS/390 and RS/6000. Making this information available for new intranet, Internet and extranet users in new ways enables you to reduce costs, improve services, generate new sources of revenue, and stay competitive.

When implementing a host application and portal integration solution, you can gather these applications into a portal page and therefore provide your clients and users with improved and simplified access.

1.1.1 IBM WebSphere software products

IBM provides software in the WebSphere family of products that allows you to develop solutions for portals. For example, WebSphere software products offer you solutions for Web-to-host integration in a portal environment. Integrating host applications with e-business portals becomes possible with the use of the following products:

- **IBM WebSphere Host On-Demand (HOD)** is an IBM product that provides Web users with access to host applications through emulation in a Web browser. Written entirely in Java, it is primarily server based, and it downloads its applets to client devices such as Web browsers. In other cases, applets can also be installed as stand-alone clients. Host On-Demand uses the following standard Internet protocols:
  - TCP/IP
  - HTTP
  - Telnet 3270
  - Telnet 5250
  - SSL

  Support for TN3270E, TN5250, VT52/100/220 and IBM CICS Java Gateway access provides a single interface to key host data. Because Host On-Demand is Java-based, its interface has the same look-and-feel across various types of operating environments.

- **IBM WebSphere Host Publisher** provides a quick and easy way to implement e-business host applications by extending its access to Web browsers for new users across the intranet and Internet, without changing the existing host applications. IBM WebSphere
Host Publisher allows you to integrate multiple sources of host data into a single Web page, giving users the appearance of a single Web site. It enables Web integration with existing 3270, 5250, VT, JDBC, and Java host applications without requiring any changes to those existing applications.

> **IBM WebSphere Application Server** is leading the way in support for industry open standards. Release 4.0 provides full Java 2 Platform, Enterprise Edition (J2EE) compliance with a rich set of enterprise Java open standards, making it production-ready for the deployment of enterprise Web services solutions.

Businesses can respond to changing markets without migrating to a different technology base or replacing existing technology investments. With WebSphere Application Server, you can move applications to more capable platforms or simply add to your existing infrastructure.

> **IBM WebSphere Portal Server** offers a framework to meet all the presentation services as well as security, scalability and availability issues for portals. It provides a flexible and open infrastructure for creating and deploying many kinds of portals that are accessible from many different types of mobile and desktop devices. It can also deliver Web content to WAP-enabled devices and i-Mode phones, as well as to various Web browsers.

WebSphere Portal Server is middleware and includes applications (called portlets) and development tools for building and managing secure business-to-business (B2B), business-to-consumer (B2C), and business-to-employee (B2E) portals.

Portlets are central to WebSphere Portal Server. As special reusable Java servlets that appear as defined regions on portal pages, portlets provide access to many different applications, services, and Web content. WebSphere Portal Server provides a rich set of standard portlets. It also provides an API that portlet developers can use to create new portlets.

> **SecureWay Directory (LDAP Server).** People and businesses are increasingly relying on networked computer systems to support distributed applications. These distributed applications may interact with computers on the same local area network (LAN), or anywhere on the Internet. To improve functionality and ease of use, and to enable cost-effective administration of distributed applications, information about services, resources, users, and other objects accessible from applications need to be organized in a clear and consistent manner. Much of this information can be shared among many applications, while at the same time it must be protected to prevent unauthorized modification or the disclosure of private information.

SecureWay Directory Services implements the Lightweight Directory Access Protocol (LDAP), a fast growing technology for accessing common directory information. LDAP has been embraced and implemented in most network-oriented middleware. As an open, vendor-neutral standard, LDAP provides an extendable architecture for centralized storage and management of information that needs to be available for today’s distributed systems and services.

### 1.2 A conceptual view of host integration using portal server

IBM’s answer to Web-to-host integration is Host On-Demand and Host Publisher. They provide solutions to accessing the host applications on the Web.

Host On-Demand (HOD) is a client-based Web-to-host solution. One of the key components of HOD is a Java applet designed to emulate the 3270 terminal on a browser hosting a Java Virtual Machine. After the Java applet emulator is downloaded from the HOD server to the client, the client connects through a TN3270 server to access a host application directly. The main objective of HOD is to meet the needs of intranet and extranet users to access
mainframe legacy applications such as 3270 and 5250. These users are familiar with the original host application screens and can be considered as power users who require a full-function emulator. User desktop software is generally well managed and can be included in a Java-enabled browser. In this environment, users will typically connect via a LAN and for extended periods of time.

Host Publisher is a server-based Web-to-host solution. The client browser communicates with the Host Publisher server and the server makes a connection to a host, accesses a host application, and presents data back to the client browser in the form of an HTML page. Host Publisher is designed mainly to build applications for end users who are not familiar with typical host screens or how to navigate through legacy applications and for whom a new, easy-to-use graphical user interface is critical. It also addresses the needs of those who are familiar with host screens but who do not have Java-enabled browsers and therefore require HTML, or who prefer not to use the “green screen” interface. For these users, applications might need to access multiple hosts.

IBM Host Publisher has two major components: Host Publisher Server and Host Publisher Studio. The Host Publisher Server provides a runtime environment running Host Publisher-based Web applications on the base of IBM WebSphere Application Server. The Host Publisher Studio is a set of development tools used to define user interactions with the host applications, generate new user interfaces in the form of JSP pages, and package all the application components such as Java classes and JSP files in an enterprise application archive (.ear file), which is ready to deploy or import in a more sophisticated Integrated Development Environment (IDE) tool such as WebSphere Studio Application Developer for further enhancement or reuse.

As corporate portals emerge as a standard, there are increasing needs to make a variety of host applications available on portals that provide the capability of a secure, single point of access to diverse information and applications, personalized to the needs of individual users. A portal typically provides a set of portlets, which produce fragments of a HTML page that will be assembled by the portal server in a full HTML page sent back to the browser. How to render a Host Publisher-based Web application and HOD applet within certain specific portlets has to be resolved.

Figure 1-1 on page 5 shows a typical conceptual architecture of host integration for e-business portals using IBM Host On-Demand and Host Publisher. Depending on your integration needs, you can choose to deploy just one or both host integration solutions. To illustrate the basic concept of host integration, other details typical in an application server environment running e-business applications, such as security, work load balancing, etc., have been ignored. These topics will be addressed in the following sections.
The portal server provides a single point of access to the host applications to users who may be users on the Internet or employees within an intranet. The basic manageable elements on the portal are the installed portlets that provide unified access to individual information sources and applications, such as Web applications or other back-end applications. In addition the portal server also provides basic services for personalization and customization. A well-defined portlet can take advantage of these services and adapt accordingly based on the user’s preferences and environment. IBM provides two portlets that enable portal-based access to host applications. In the following section we explain how they work.

1.2.1 Host On-Demand portlet

Host On-Demand Server provides the management of the Host On-Demand environment, including configuration and user management, and supplies the client browser applets for download. At its core it is an HTML-based application running on top of a Web server. The second major function performed by the Host On-Demand Server is SSL support for the client browser, so that a secure communication between the client and Web server becomes possible. In this case Host On-Demand is a redirector that establishes an SSL session to the client and forwards TN32070 to the real host. An additional component, called Screen Customizer, can be used to customize the look and feel of the 3270 “green screens”. The customization data will be saved on the Host On-Demand Server and retrieved by the client browser at runtime, if needed.

To be able to use Host On-Demand on a portal, Host On-Demand provides a tool used to generate HOD portlets with different host session configurations. The Host On-Demand portlet enables the browser to run the Java applet emulator for 3270 applications on the portal. Figure 1-2 shows a view of how the Host On-Demand portlet enables access to host applications via a portal. After completing the download of the HOD applet to the client, the
HOD applet can now connect to the host application directly via pre-configured sessions without any further assistance of the portal server. The host sessions created by the Java applet emulator can be launched in a separate window or imbedded within the existing HOD portlet window.

1.2.2 IBM WebSphere Host Publisher portlet

Host Publisher Server is the runtime environment hosting Host Publisher-based Web applications which typically includes various macros, host access modules (for example Integration Objects), and JSP pages used to generate HTML-based front ends of the new applications. The host access macros are used to connect to the host application, extract data from the returned host screens, and finally disconnect from the host application after completing a conversation session. The Integration Objects are the key components hiding the complexity of host access. Basically, they are JavaBeans and can be used directly in the JSP pages. To support different types of applications several advanced access interfaces such as Web Services and EJBs for the Integration Objects can be used. This is especially useful if the business logic of host applications is part of the new business applications and the Host Publisher server serves only as an integration focal point for accessing the host applications.

To be able to access the Host Publisher-based Web applications on a portal, Host Publisher provides a specific portlet, called IBM WebSphere Host Publisher portlet, which understands the details of how to communicate with a Host Publisher-based Web application. Figure 1-3 on page 7 shows the architecture of how the IBM WebSphere Host Publisher portlet supports the access to the Host Publisher-based Web applications running on top of Host Publisher server via a portal.
After launching the portal start page on your browser, you can jump to the Host Publisher portlet configured to point to a specific Host Publisher application running on the Host Publisher server in order to gain access to the start page of the Host Publisher application. All the responses produced by the Host Publisher application afterwards will be rendered within the portlet window. The Host Publisher portlet is a generic portlet, basically derived from the Clipping portlet, which allows the portal server to clip the HTML content generated by the Host Publisher application. An additional button in the resulting page is provided for the user to clean up the host connection and restart the application within the portlet.

A wireless gateway is responsible for performing protocol conversion between Wireless Application Protocol (WAP) and HTTP. A transcoding server can also be used to support application content other than HTML, for example WML (WAP devices) and cHTML (i-Mode devices). To be able to support host integration via pervasive devices, such as cell phones or i-Mode specific portlets will be required. The transcoding server can be used in several different configurations:

- It can be configured as a proxy, so that it transforms outbound markup before the markup is sent to the browser. This configuration is useful when making the portal accessible through personal digital assistant (PDA) devices. It is also useful for WML/WAP devices, where the payload size is very limited.
- A portlet can be developed to call the transcoding service during page aggregation. This way, individual portlets can produce their markup automatically, therefore, custom JSP views or style sheets for each markup are not required.

In the following sections we will add more details to the conceptual architecture for host integration on the portals discussed above and make them more specific to some of the typical application scenarios in e-business portals.
1.3 Host integration scenarios

*IBM Patterns for e-business* are a group of reusable assets that can help speed the process of developing Web-based applications. Among these patterns, the integration patterns are tightly related to the host integration for e-business portals. Generally speaking, Integration patterns connect other Business patterns together to create applications with advanced functionality. Integration patterns are used to combine Business patterns in advanced e-business applications.

Integration patterns address two types of integration:

- **Access integration.** This pattern enables access from multiple channels (devices) and integrates the common services required to support a consistent user interface. The portal technology, with the support of industry-standard portlet API that facilitates functions such as single sign-on, multiple device support and personalization, can be used to implement this kind of access integration pattern.

- **Application integration.** This pattern allows for access of multiple applications and information sources without the user directly invoking them.

In order to show the differences between portals, the following sections describe the requirements that need to be addressed when you need to create proper architecture solutions for different host integration environments.

1.3.1 Business-to-employee (B2E)

A sophisticated B2E portal must address a number of requirements, as follows:

- Support collaboration and communities
- Provide capability of knowledge management to share valuable intellectual assets between employees
- Allow users to do business intelligence
- Enable legacy or host integration

An intranet is typically a secure environment with high-speed networking. Most of the employees are power users; some of them may know the existing host applications well. In this case, IBM Host On-Demand can be a foundation for providing the capability of accessing host applications via a portal without any change of host applications. You can deploy the integration solution quickly. Users who are acquainted with host applications can start using the portal solution without any significant education. However, if your long-term strategy is to provide a more user-friendly interface to the host applications and at the same time prepare the application infrastructure for further use, such as enabling Internet access to the business transactions implemented by the existing host applications, you may consider the IBM Host Publisher as the host integration platform. Other considerations are also described in 1.3.2, “Business-to-consumer (B2C)” on page 9.

To leverage the capability of single sign-on provided by the portal server you should consider using Windows domain logon for client authentication if possible, and the Express Logon Feature (ELF) for host application authentication, so that from the user's point of view access to host applications via the portal is also well-integrated in the portal single sign-on security environment.
1.3.2 Business-to-consumer (B2C)

A portal designed for B2C purposes normally has a far bigger reach than that of B2E portals. Portal users can be connected over the Internet from anywhere in the world. Some of the connections may suffer from low bandwidth. Users usually have different skill sets, and access the portal at different times. For those users, an attractive, easy-to-use user interface is of crucial importance. In this case you must first migrate your host business applications and publish them on your portal. IBM WebSphere Host Publisher is the tool that helps you address these requirements in a cost-effective way.

Using Host Publisher studio you are able to create straightforward HTML-based front ends for the host applications in a short period of time. You can also change the control flow of the legacy applications on the front end so that they are more in compliance with the usual Web applications regarding look and feel. However, in the majority of developing e-business applications, what you really need is just the business logic or transactions provided by the legacy applications. The user interface required by the new e-business application has to meet a number of Internet-typical requirements, such as short response time, and be capable of customer self-registration. In such a situation, Host Publisher can be used as integration middleware, wiring your Web applications and your host applications.

Using the single sign-on of the portal server infrastructure to enable the security check for the host applications is highly recommended.

1.3.3 Business-to-business (B2B)

B2B portals are usually designed to support and conduct business transactions over the Internet, especially between business partners who are connected via an extranet. Security enforcement and XML support are critical success factors.

In a well-controlled B2B environment, you can use the Host On-Demand solution to implement direct access to the host applications on your B2B portal. This allows your business partners to use your host business applications directly without any change. Firewalls might be an issue for some of enterprises, because to use the Host On-Demand cross enterprise boundaries you usually need to open a special port on the firewalls that allows telnet-based communication between HOD applets on the browser and the host.

To enable business automation, you can also use Host Publisher as an integration platform that provides host access services to other e-business applications. These services can be provided in the form of JavaBeans, XML, EJB, and Web Services. Supporting Web Services-based access to the host applications is a unique feature provided by Host Publisher V4.0. The major benefit is that it allows you to turn the business logic contained in the host systems into universally accessible business functions. This can be done in a reasonable period of development time.

In this B2B scenario you use Host Publisher Studio to create Integration Objects, with Web Services enablement and IBM WebSphere Studio Application Developer to create and test the Web Services implementation, which is basically a wrapper of the corresponding Integration Object infrastructure services for host integration.

To support host integration on the portals, several infrastructure services are required to enable host access and give the clients more flexibility in doing business. The typical services are presentation services, host access services, personalization services, services for security and administration, and support of pervasive devices. We discuss each of these services in the following sections.
1.3.4 Presentation services

Presentation services are the foundation of a universal desktop for all the Web-based applications of an enterprise. Simply speaking, an industry-standard browser such as Netscape Communicator or Microsoft Internet Explorer provides the necessary infrastructure to render the HTML-based applications on the client and launch a JVM to run the applets. The services typically include:

- HTML
- JavaScript
- Java Virtual Machine

To be able to run Host On-Demand applets you must ensure that the browser can provide the required Java support.

For wireless devices such as i-Mode and WAP-enabled cell phones, the following services are required:

- WML
- cHTML

1.3.5 Host access services

To be able to connect to the host applications, execute transactions, and extract the returned data, the following services may be required:

- Connection management for host sessions
- Access services and resource adapters such as those provided by the Host Publisher runtime modules (contained in the Host Publisher server)
- Transaction services, if you want to run host transactions in conjunction with other resource managers
- Java Connector Architecture (JCA) (might be required)
- Web Services support, if you publish your Integration Objects as Web Services on a Universal Description, Discovery and Integration (UDDI) repository
- XML support enabling the Integration Objects to return XML data

Note: Depending on the host environment, you may not need all of the services mentioned above.

1.3.6 Personalization services

Personalization service enable users or the enterprise to shape the choice, style and format of applications. Personalization may be done at any level (individual, group, role). It relies heavily on other services, such as presentation and security.

The portlet framework provides the capability to enable portlets to show different views, such as edit-view, to the client. Using this capability allows more flexibility in the host integration design and implementation.

1.3.7 Services for security and administration

Security and administration services enable users to access multiple applications deployed on the Host Publisher server and Host On-Demand Server.
1.3.8 Support of wireless devices

To enable wireless devices to access the host applications deployed on a Host Publisher server, transcoding services such as those provided by IBM WebSphere Transcoding Publisher will be required. The specific portlets used to access the Host Publisher-based applications should be enhanced with respect to the capability of generating proper markup language content, for example cHTML for i-Mode devices and WML for WAP-enabled phones.

1.4 Architectural considerations

This section lists the most important considerations during the architecture phase. Typical questions that must be addressed in the early stages of any integration project are:

1. Publish host applications as-is or give users entirely new front ends?
2. Which technologies can best serve the integration objectives?
3. What does a thoroughly integrated security concept look like?

1.4.1 Understanding the integration needs

Understanding why you need host integration is crucial in making proper architecture decisions. The following factors must be taken into consideration:

- Who is going to use the host applications via portal?
- Is it a strategic or tactic solution?
- What time frame is available to develop and implement a host integration solution that can meet your business needs?
- What available skills are needed to implement the solution?

The following options can be regarded as a reasonable starting point:

1. Publish the host applications directly on the portal. Available resources are a Host On-Demand portlet in conjunction with Screen Customizer, Host Publisher portlet for Host Publisher-based applications and the XML gateway. It is also expected that new technologies will be implemented at a later date in future product releases.

2. Use the business logic via Integration Objects and integrate it into portlet-based applications. Available resources are Integration Objects in the form of JavaBeans, Web Services, and XML interface.

1.4.2 Infrastructure considerations

To ensure that a host integration solution can be deployed and operate properly, an infrastructure capable of providing certain levels of service has to be put in place. The following resources must be examined:

- Network for LAN and WAN
- Firewall
- Client environment (browser and JVM)
- Server platforms, such as Linux, UNIX or z/OS
- Enablement of Express Logon Feature on the host
Except for Host On-Demand applet-related network issues, there are no other special issues in deploying and running Host Publisher-based Web applications that need to be taken into account, because they are just normal Web applications running in the Web container on top of an application server. Of course, any concerns regarding development, management, and operation of Web applications apply to Host Publisher applications as well.

1.4.3 Software considerations

There are several software resources that can be used to implement host integration for e-business portals. The following are the most important:

- Use applets connecting to TN3270 directly or via HTTP tunneling
- Use Host Publisher to generate Web applications
- Use XML Gateway to access the host applications
- Use Integration Objects and publish them as Web Services for use of other portlet-based applications

1.4.4 Development considerations

Host On-Demand is a ready-to-go solution as there is no additional development required. Using Host Publisher to make host applications available on the portals may require substantial development efforts, in comparison with deployment of a HOD-based solution.

Host Publisher Studio provides significant support for creating Host Publisher-based Web applications. For example:

- Allows you to record host access macros, generate Integration Objects, and generate simple JSP pages using the data fields already defined in the Integration Objects as properties.
- Allows you to package your Host Publisher application in an enterprise application archive as EAR files. The resulting EAR file can then be imported in the more sophisticated development tools such as WebSphere Studio Application Developer for further development.

Note: It is highly recommended that you use WebSphere Studio Application Developer for application development, testing and debugging. As a result, you will finish your development faster, and the quality of your applications will be better.

The Host Publisher Studio is a tool that is easy to use and capable of generating simple HTML forms that will be associated with the Integration Objects. It also provides the ability to sequence the resulting JSP pages. This tool can only be used to make the first cut of your application. That means you cannot easily synchronize the parallel development with both Host Publisher Studio and WebSphere Studio Application Developer.

1.4.5 Systems management considerations

For most IT corporations, host integration means integrating a variety of system components in a heterogeneous environment. Effectively managing those components becomes a serious challenge. To be able to more effectively in manage the entire production portal environment afterwards, several systems management points have to be taken into consideration prior to finalizing the overall integration architecture. The most important concerns to be taken into account are security management, deployment, administration and monitoring, and required availability. In the following sections, each of these aspects will be discussed.
Security

Security is normally the top issue for most corporate customers. To address security in host integration for e-business portals properly, several security solutions can be taken into consideration. In 1.2, “A conceptual view of host integration using portal server” on page 3, the following security options may be implemented:

- Use the Single Sign-On (SSO) capability of the portal server
- Trust the platform environment such as Sign-On in the Windows security domain
- Use the Express Logon Feature (ELF) of Host On-Demand to enable access to the host applications without prompting individual users to sign on.
- Use the sophisticated functions of supporting user management and access control provided by the WebSphere Portal Server. To be able to effectively manage the registration of a large number of users, a thorough registration process has to be put in place.
- Open special ports for TN3270 on the firewall when using HOD applets that will go across the Internet.

Deployment

Host Publisher Studio and WebSphere Studio Application Developer provide sophisticated features supporting a seamless deployment of Host Publisher-based Web applications to the Host Publisher Server environment. Version control and management of the Host Publisher portlet and diverse Host Publisher Web applications should be planned and executed in a well-controlled manner.

Administration and monitoring

Basically, IBM WebSphere Portal Server provides the required capability of managing Host On-Demand portlets or Host Publisher portlets. You can use the functions of the portal server to monitor the usage of all the portlets running on top of the application server, and those of the application server to monitor the status of the active Host Publisher applications running on the Host Publisher Server.

Managing the required availability

The availability of a host integration solution depends on the availability of components such as Host On-Demand Server or Host Publisher Server. Assuming that the underlying infrastructure can provide the required availability, the following configurations to achieve high availability apply:

- **Host On-Demand.** Use LDAP directory services to store configurations for multiple Host On-Demand servers. If one of the servers failed, the Network Dispatcher can dispatch the requests to another server that shares the same configuration with the failed server. For more information see *IBM Host Access Client Package*, SG24-6182.

- **Host Publisher.** Use the capability of cloning provided by the application server for single and multiple machines, so that the network dispatcher can dispatch the requests to available Host Publisher server instances. For more information on how this work, refer to *WebSphere Scalability: WLM and Clustering Using WebSphere Application Server Advanced Edition*, SG24-6153 and *A Comprehensive Guide to IBM WebSphere Host Publisher Version 3.5*, SG24-6281.
Integrating WebSphere Host On-Demand with Portal Server

This chapter describes the basic WebSphere Host On-Demand components, as well as how these components work in a Portal Server solution. In this chapter, we discuss the process of creating the required configuration for your portal environment, and more specifically the use of the mkpar tool to create a portlet.

In this chapter we discuss the Host On-Demand portlet solution on the WebSphere Portal Server Version 2.1.
2.1 Host On-Demand portlet

The Host On-Demand portlet is an interface inside the WebSphere Portal Server and requires a Host On-Demand license. Because Host On-Demand is an applet, users need a Java Virtual Machine (JVM) enabled in their browsers. Figure 2-1 illustrates the basic process started from the client device to access and run the Host On-Demand portlet.

![Diagram of Host On-Demand portlet](image)

**Figure 2-1  Basic Host On-Demand portlet**

The following description simplifies the process of accessing a telnet server using the Host On-Demand portlet:

- The first time, the user accesses the Portal Server with the Host On-Demand portlet. This portlet contains the configuration created by the administrator with all sessions and restrictions required for this user. The configuration will also include where the Host On-Demand Server is located and which protocol (http or https) will be used to receive the applet. The user sign-on is not mentioned here because you can add the customized Host On-Demand portlet into one of the public pages.

- Once the user has accessed the HOD portlet, the client will receive the applet from the Host On-Demand Server. The Host On-Demand Server address has been already configured by the Portal Server Administrator during the deployment process of this portlet. This configuration also specifies the HOD model that will be used for this session as follows:
  - Download or cached client
  - Imbedded or running in a separated window

- Finally, the Host On-Demand applet will make a connection to a destination telnet server. The configuration values required to access all the required Telnet Servers, such as IP address, port, and SSL configuration, have been defined already in the portlet.
2.2 Major HOD components

In short, Host On-Demand uses the Java environment, industry-standard TN3270, TN5250, VT, SSL and other Internet protocols to provide secure, platform-independent host access through a Web browser.

The features and functions that Host On-Demand provides have been updated with each version of the product by improving the product to support new requirements. The most used clients are display and printer sessions for 3270 and 5250, VT terminals and FTP clients.

2.2.1 HTTP Server

The Host On-Demand server requires a Web server, installed on the same machine, to provide the applet download to the client.

If the configuration created by the Deployment Wizard is an HTML-based model, then the definitions for all the sessions are stored in HTML format and the configuration files will be stored in the HODData directory. This type of configuration does not require the Host On-Demand Service Manager to validate the user inside Host On-Demand and the TCP/IP port number 80 must be used. You will find more information about the required ports in 2.5.1, “Firewall concepts” on page 24.

2.2.2 HOD Server

The HOD Server provides functions to send the applet to the client, and manage users and groups, as well as keep the environment up to date.

Other provided services, such as redirector, OS/400 Proxy Server and licenses, are managed by the Host On-Demand Server.

2.2.3 LDAP support

If you are planning to implement more than one Host On-Demand server, LDAP support is recommended.

Using an LDAP structure allows you to store a configuration from a Host On-Demand server and when required the same configuration can be accessed and updated from other Host On-Demand servers. However, the default implementation of Host On-Demand uses a private data store model that does not provide sharing across servers.

Note: We do not recommend using the same LDAP Directory used by the Portal Server, with Host On-Demand. The Host On-Demand structure inside the LDAP Directory is basically to be shared with other Host On-Demand servers only.

2.2.4 HOD clients

Host On-Demand provides several types of pre-configured clients. Each client is composed of several components. In addition, you can use the Deployment Wizard to create a custom emulation client with all components and features you would like to make available to the user.
2.2.5 Deployment Wizard

The Deployment Wizard is a guide that enables you to configure every function necessary for your environment. This software also helps you to define which functions will be available, and which components will be used by the users.

With the Deployment Wizard (in the Host On-Demand product), you can identify and configure the Host On-Demand sessions that you wish to make available to your users. The result of this process is an HTML page that can be accessed by the user.

**Note:** To create a portlet, the mkpar tool must be used.

2.2.6 Screen Customizer interface

IBM Screen Customizer Version 2.0.62 is a Java client for Host On-Demand that provides a graphical user interface alternative to the host application “green screens”. It has the ability to change a standard 3270 or 5250 emulator application into something that’s comparable to a typical Web application, making 3270 and 5250 host applications easier to use. Screen Customizer can combine data from multiple screens, hide unneeded information from the user, and change cryptic mainframe input fields into user-friendly forms, such as radio buttons, check boxes, drop-down lists and valid value lists, depending on the input required.

In short, Screen Customizer can provide a better user presentation to applications, providing new life for 3270 and 5250 applications whose only problem is their user interface. Screen Customizer can extend the life of the mainframe’s terminal-based applications while a Web-based replacement is being built, or even serve as an end-of-life substitute for little-used applications.

The following are some of the immediate benefits that can be derived upon deploying Screen Customizer:

- Users unfamiliar with traditional host application green screens typically find interfaces similar to Web pages easier to use.
- Customization can be established for individual users or groups, so that the graphical screens provide controlled application access and flow.
- Customization can be done by a graphic artist and not necessarily by a system administrator.
- Host On-Demand provides all Telnet connectivity and security, enabling its use in most TCP/IP network environments.

For more information about the Screen Customizer components and the process necessary to configure maps and templates, refer to *IBM Host Access Client Package*, SG24-6182.

2.2.7 Language support

**WebSphere Portal Server**

WebSphere Portal Server displays its content using the user-preferred language configured in the LDAP Directory. The user chooses the language during the enrollment process, and this language is effective once the user logs in to the portal.
If the user has not selected a language or has not yet logged in to the portal, WebSphere Portal Server first looks for the language defined by the user's browser and, if one is defined, the content is rendered in that language.
Otherwise, the portal uses the default language specified in the TurbineResources.properties file.

**Host On-Demand and Screen Customizer**

The Host On-Demand client only looks for the language defined by the user's system and requests this package from the Host On-Demand Server. If the user's location is not defined on the server, the client browser will use the default language, English. The same thing occurs when you are installing the Host On-Demand Server and Screen Customizer: the installation process will use the language defined in your system.
If you are planning to support Host On-Demand clients in other languages or countries, it is necessary to select all these languages during the Host On-Demand Server and Screen-Customizer installation process. Table 2-1 shows the languages supported in the current release for both products.

**Table 2-1. Language support for Host On-Demand and Portal Server**

<table>
<thead>
<tr>
<th>Language Support</th>
<th>Host On-Demand and Screen Customizer</th>
<th>Portal Server 2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Brazilian Portuguese</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Portuguese (Standard)</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Czech</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Chinese (Simplified)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Chinese (Traditional)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Danish</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Dutch</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Finnish</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>French</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>German</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
### 2.3 Client receive method

The client receive method that will be used to receive the Host On-Demand applet.

#### 2.3.1 Host On-Demand cached client

Using the Cached Client, the components will be stored on the users machine, avoiding one another download in the successive use.

This is the Host On-Demand recommended method because startup is faster and it makes the best use of the network. However, when implementing a Portal Server solution where clients can possibly use different versions of Host On-Demand to access multiple suppliers, this can be a problem because multiple versions of the client cache are not supported by Host On-Demand.

**Important:** After the download of the cached client, users must restart the browser to complete the installation process.

As a rule, use the wizard to select which components will be downloaded initially. All other components will be downloaded as needed. By doing this, you are not excluding the user from accessing other functions. To disable functions, you should use the Disable Functions option in the wizard.

#### 2.3.2 Host On-Demand download client

The download client feature does not require a browser restart to use the function stored, like the cached client do, but in another access to the server, the components selected in the wizard will be downloaded again.

Using the download client you will be ready to access the server without any need to restart the browser and consequently it will be ready to be used by the Portal Server users.

<table>
<thead>
<tr>
<th>Language Support</th>
<th>Host On-Demand and Screen Customizer</th>
<th>Portal Server 2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greek</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Hungarian</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Italian</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Japanese</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Koran</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Norwegian</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Polish</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Russian</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Slovenian</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Spanish</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Swedish</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Turkish</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
With the Deployment Wizard you can select the components that users will receive during the download process. If you would like to prevent the download of other components, it is recommended that you first disable the unwanted function.

2.4 Client execution (run) method

The Host On-Demand applet can be executed in two different modes:

- Embedded
- Separated

2.4.1 Embedded mode

Using the embedded mode, the Host On-Demand applet will run under the Host On-Demand portlet session. All embedded emulation sessions use the same applet area, and this size is defined by the wizard in the Advanced Options.

Figure 2-5 illustrates a Host On-Demand portlet with embedded sessions.

![Host On-Demand portlet with embedded sessions](image)

If you desire a larger emulation session, update your HTML file in the c:\hostondemand\hod\ directory as suggested in Example 2-1.

Since Host On-Demand is an applet, and consequently is executed by the Java Virtual Machine, a portal logoff will not be sufficient to disconnect the already connected sessions. These sessions will remain connected until they receive a disconnected signal from the host or until the user closes the browser.

If the user returns to the portal server and makes a second sign-on, then the user will see the Host On-Demand sessions in the same state they were in before logging off.
Example 2-1  Changing the applet size

```html
<!doctype html public "//W3//DID HTML 3.2 Final//EN">
<!-- HOD WIZARD HTML -->
<html>
<head>
...
</head>
<body background="hodbkgnd.gif">
<center>
<img src="hodlogo.gif">
<p>
<!-- APPLET Begin -->
<applet archive="..." code=com.ibm.eNetwork.HOD.HostOnDemand.class width=920 height=505>
<param name=cabinets ...>
<param name=parameterfile ...>
...
</applet>
<!-- APPLET End -->
</center>
</body>
</html>
```

2.4.2  Separated mode

When using this mode, the emulation window will not run inside the portlet, even though it will still be using the browser Java Virtual Machine. This can give you more flexibility to change the size of the emulation session screen, but all other functions will still work in the same mode.

**Note:** Since the applet is running in a separate window, it will still be working after the user logs off from the portal.

2.5  SSL solutions

This section discusses the network security techniques available to implement a secure solution with Host On-Demand and Portal Server, and provides an overview of a number of solutions for addressing security issues in networks. The field of network security in general and of TCP/IP security in particular is very wide, so this section concentrates on the most recent and most widely used security techniques.

2.5.1  Firewall concepts

If you are planning to implement a Host On-Demand solution with Portal Server over the Internet, you will need to make some changes in your firewall.

A firewall is a system (or group of systems) that enforces a security policy between a secure internal network and an untrusted network such as the Internet. Firewalls tend to be seen as a protection between the Internet and a private network. But generally speaking, a firewall should be considered a means to divide the world into two or more networks: one or more secure networks and one or more non-secure networks. A firewall can be a PC, a router, a midrange, a mainframe, a UNIX workstation, or a combination of these that determines what
information or services can be accessed from the outside and who is permitted to use the information and services from the outside. Generally, a firewall is installed at the point where the secure internal network and untrusted external network meet, which is also known as a choke point.

It is important to note that even if the firewall is designed to permit the trusted data to pass through, deny the vulnerable services, and prevent the internal network from being attacked, an attack may penetrate the firewall at any time. The network administrator must examine all logs and alarms generated by the firewall on a regular basis. Otherwise, it is generally not possible to protect the internal network from outside attacks.

We know that the Host On-Demand applet will access the telnet server from the user’s machine. Therefore one port for this service must exist on the firewall. Another solution is using the redirector service inside the Host On-Demand Server to access the telnet server, but the port to the redirector service needs to be available on the firewall. The redirector service can implement SSL support for cryptography between the client and the redirector service.

If you are implementing Host On-Demand with a native user, in other words, defining users and groups inside the Host On-Demand server, another port will be necessary to validate users and after that receive the emulation session. The default port for this service is 8999, and can be substituted by the Host On-Demand Servlet inside a WebSphere Server. With this you can validate users using port 80 of the WebSphere Server.
2.5.2 Secure Sockets Layer (SSL)

If you have clients outside the firewall connecting to any telnet server or Host On-Demand redirector inside your firewall, you should configure SSL connections to ensure your data is secure.

Starting with Communications Server for OS/390 V2R6, the TN3270 telnet server provides support for the Secure Sockets Layer (SSL) protocol. This provides secure data transmission between a secure port and an SSL-enabled client. The SSL authentication mechanism known as server authentication is used.

Important: The crucial step in the process is when the client checks its list of trusted Certificate Authorities (CAs) and self-signed certificates. For a locally installed client, on which Host On-Demand is loaded directly from the client's hard disk, that list is kept on its local hard disk. This is considered adequately secure.

However, for a download client, on which the client is really just a browser that downloads all its code from the server using HTTP, the only place the browser can look for the list of trusted CAs or self-signed certificates is on the server from which it has just downloaded the certificate. If that server is an intruder, security is breached. One way to avoid this problem is to use HTTPS rather than HTTP, because HTTPS ensures that the browser really is connected to the correct server.

2.5.3 Redirector

The redirector is a telnet proxy that may be used to protect internal telnet servers and ports from unauthorized Internet users or to provide security for telnet servers that do not support security natively.

The redirector is a telnet proxy that is able to accept connections from clients and pass them on, through a different port, to the next stage in the link. The redirector can serve as a barrier between clients and the target telnet server. If you do not want large number of clients connecting directly to your host system because of a security risk, you can have the clients connect to one or more redirectors. The redirectors pass the connection on to the host, allowing you to hide the address of the host from the client users. On Windows NT and AIX, the redirector provides the support for Secure Sockets Layer (SSL) security between clients and the server.

The redirector acts as a transparent telnet proxy that uses port remapping to connect Host On-Demand to other telnet servers. Each defined server is given a local port number. Instead of connecting directly to the target telnet server, a Host On-Demand session connects to the Host On-Demand server. The redirector maps the local port number to the host-port number of the target and makes a connection.

Redirectors can be connected to each other (in a cascaded configuration). In that case, SSL security is also available between the redirectors.

Figure 2-7 shows how the redirector works. Secure connections are possible between the client and Host On-Demand server.
The redirector gives Host On-Demand secure access to a wide range of hosts. Typically a Java applet, such as Host On-Demand, is made secure by preventing access to all local and network resources except the host that directly supports the applet.

The redirector sets security for each host. Security choices are no data-stream modification (pass-through), client-side encryption, host-side encryption, and encryption on all data flowing between the Host On-Demand emulator session and the secure server (both).

2.5.4 HTTPS

HTTPS is a unique protocol that combines SSL and HTTP. You need to specify https:// as an anchor in HTML documents that link to SSL-protected documents. A client user can also open a URL by specifying https:// to request an SSL-protected document.

Because HTTPS (HTTP + SSL) and HTTP are different protocols and use different ports (443 and 80, respectively), you can run both SSL and non-SSL requests at the same time. As a result, you can choose to provide information to all users using no security, and specific information only to browsers that make secure requests. This is how a retail company on the Internet can allow users to look through the merchandise without security, but then fill out order forms and send their credit card numbers using security.

This feature is very important if you are sending the certificate for a SSL Server to the client. And of course this port needs to be available through the firewall too.

For more details on the concepts covered in this section, please see Tutorial and Technical Overview, GG24-3376 and IBM Host Integration in a Secure Network: A Practical Approach, SG24-5988.

2.5.5 Express Logon Feature (ELF)

The Express Logon Feature is an IBM cross-product solution that allows a user with a TN3270 client session and an X.509 certificate to log on to a SNA application residing on a host system without having to enter a user ID and password.

This new feature has several advantages:

- It helps reduce the time administrators spend maintaining user IDs and passwords.
It helps reduce the number of user IDs and passwords that users have to remember.

It helps remove a potential security risk of users writing down user IDs and passwords, losing them, or sharing them with someone else.

It helps remove a potential security risk of users using trivial or easily guessed passwords.

Details on using this feature can be found in WebSphere Host On-Demand: Library, found at http://www-3.ibm.com/software/webservers/hostondemand/library.html and in IBM Host Access Client Package, SG24-6182.

2.6 Installation of HOD

In this section we introduce information related to the installation of the Host On-Demand product.

2.6.1 Supported server operating systems

A Host On-Demand server can be installed on the following operating systems:

- Windows NT 4.0 with SP5 or later
- Windows 2000 (Professional, Server and Advanced Server)
- AIX Version 4.3.3, 4.3.4, and 5.1
- OS/2 Warp Server Version 4 and OS/2 Warp Server for e-business 4.5
- Novell NetWare Version 4, 5, and 6
- Sun Solaris 2.6, 2.7, and 2.8
- OS/400 V4R4, V4R5 and V5R1
- HP-UX 10.20 and 11.00
- Red Hat Linux 6.2, 7.0, and 7.1
- SuSE 6.4, 7.0, and 7.1
- OS/390 Version 2 Releases 5, 6, 7, 8, 9, 10
- z/OS Version 1 Releases 1 and 2
- Caldera 2.3
- TurboLinux 6.0 and 6.1
- UnixWare 7
- Linux on S/390

Note: Windows XP support is available for client only.

Host On-Demand V6.02 client support includes the following:

- Netscape 4.x with JVM 1.1.8 only
- Netscape 6.x with IBM or Sun JRE 1.3 or later plug-in
- Internet Explorer 6.0 with Microsoft JVM 1.1(3802 recommended)
- Internet Explorer 6.0 with IBM or Sun JVM 1.3 or later plug-in.

2.6.2 Software prerequisites

During the Host On-Demand Server installation on Windows platform, the Java support will automatically be installed and the existence of one Web Server will be verified. If your HTTP server was not recognized at this time (see Figure 2-8), you can update the configuration with the Host On-Demand alias after the installation.
For UNIX platforms, verify the correct version of the Java that you need to install to run the Host On-Demand Service Manager. It is not required that the Java directory be on the path, because you will need to configure the NCServicemanager script with the correct Host On-Demand library and Java directory.

To create a Host On-Demand portlet, Version 6.02 needs an additional tool, the mkpar tool, which will read the HTML created by the wizard.

For more information, see “Creating the HOD PAR file” on page 32.

To execute a Host On-Demand applet on the client machine, the Java Virtual Machine must be enabled on the browser. You can find the latest Java Virtual Machine on the Microsoft download page: http://www.microsoft.com/downloads/search.asp? Its size is around 5 MB.

### 2.6.3 Installation procedure

All information necessary to install Host On-Demand on z/OS and OS/390 can be found in The Program Directory for IBM WebSphere Host On-Demand Version 6 for OS/390 and z/OS, found at http://www-3.ibm.com/software/webservers/hostondemand/library.html.

Also you can find the installation process for all platforms in IBM Host Access Client Package, SG24-6182.

English is the default language. If you are planning to support another language, install this support during the installation process.

### 2.6.4 Verifying an installation

After the installation, you need to restart your Web Server to recognize the changes made by the Host On-Demand, and of course, start the Host On-Demand Service Manager.

Execute the following procedure to verify if the server is working properly:

1. **Administration.** Open the Administration Window (http://your_server_host_name/HOD/HODMain.jsp) and select the Administration client. You need to receive the sign-on page. The default user ID is admin and the password is password.
If you have received a message about the service manager (see Figure 2-9), verify that the server is active or there is a firewall between you and the server.

![Host On-Demand](image)

Figure 2-9  Service Manager unavailable

After you have logged on, go to the users and groups in the HOD administration window, and add a user and session under the HOD group. You will need the telnet server IP address and port, and in some cases when required the specific LU name for your session.

2. **Client.** Open the http://your_server_host_name/HOD/HODMain.jsp page, select the **Download Client** option, sign on with the user created by the administrator, and start the session. This is enough to confirm that everything is running well.

If you do not receive the emulation session, verify that the IP address and port of the telnet server are correct. Also, if using a dedicated LU, make sure it is not already in use.

### 2.6.5 Running the demo programs

Use the Host On-Demand Wizard to create an HTML page that better complies with the required needs of your environment.

Use the HTML file created by the Deployment Wizard to test the configuration and the session to the telnet server.

### 2.7 Enabling portal based access to HOD servers

A common requirement for many organizations is the ability to extend the reach of their legacy applications to users on intranets, extranets or the Internet. Through the use of industry standard technologies, such as Java and TCP/IP, WebSphere Host On-Demand (HOD) provides organizations with a means to accomplish this. From within a Web browser, users download an applet from an HOD server, launch an emulation window, and connect to legacy applications using TN3270 and TN5250 protocols.
WebSphere Portal Server is designed to provide different users a single point of access to various types of information. In addition to access, WebSphere Portal Server also gives users the ability to customize the type and layout of this information. By leveraging this capability (information access and personalization) an organization can provide a tool that will lead to higher productivity from their employees.

By integrating the functionality of HOD with WebSphere Portal Server, you can more effectively manage access to legacy applications.

The applications that run in WebSphere Portal Server and provide this customizable access are called portlets. These portlets provide a wide range of functionality. They range from information look up (for example, stock quotes) to communication and productivity tools (for example, chat, calendars). You can either modify generic portlets or write your own.

The basic steps required to create a portlet are:

- Building the code needed to run a portlet
- Creating a portlet descriptor
- Creating a PAR file
- Deploying the PAR file


In order to enable portal-based access to HOD servers from within WebSphere Portal Server you will need to create and deploy an HOD portlet (PAR file). The next sections will step you through this process.

### 2.7.1 Creating an HOD portlet

We do not need to write any code in order to develop our HOD portlet. Instead we can use existing tools to create it. First, we will use the Deployment Wizard in HOD to create an HTML file. This file will contain the type of client (that is, download or cached) as well as the session information required to connect to our legacy system. Then, we will use the mkpar tool to take this HTML file, and from it, create and package the PAR file we need for WebSphere Portal Server.

We then need to install and deploy the portlet in order to make it available for use. Users could then sign in to the Portal Server and select it from the list of available portlets. As illustrated in Figure 2-10 once they complete this process, accessing the Host On-Demand Server through the portlet would work as follows:

1. The user logs in with his user ID and password into the Portal Server. The Portal Server authenticates the user and presents them with their initial portal page.
2. The user's customized set of portlets is downloaded to his machine and is displayed within his browser.
3. The Host On-Demand portlet initiates a download of the Host On-Demand applet to the browser.
4. The Host On-Demand Telnet 3270 session is established.
Creating the HOD HTML file

Start the Deployment Wizard in HOD by selecting Start -> Programs -> IBM Host On Demand -> Administration -> Deployment Wizard. Follow the prompts to create and customize the HTML file. If you are not familiar with using the Deployment Wizard, refer to IBM Host Access Client Package, SG24-6182 for details.

Note: The HOD Deployment Wizard can only run from a Windows system. If you have installed HOD server on a different platform you can always run the Deployment Wizard from the HOD CD-ROM. For details on how to do this, see for example IBM Host Access Client Package, SG24-6182.

Creating the HOD PAR file

Download the mkpar.zip utility from the IBM WebSphere Host Integration Solution page http://www.ibm.com/software/webservers/hostintegration. Once you have downloaded the file, you will need to extract it, using a normal unzip tool such as WinZip or the JAR utility. Extract all files (including directory information) to a directory of your choice, for example, C:\mkpar. This target directory is used to configure the mkpar tool.

Configuring the mkpar tool

The mkpar tool is a Java program that runs on Windows operating systems and requires Version 1.1.8 or higher of the Java Software Development Kit (SDK) and Java Runtime Environment (JRE). Before using it, you need to specify the location of the Java executable and the home location of the mkpar tool files. For example:

```
set JAVA=c:\hostondemand\bin\java.exe
set MKPAR_HOME=c:\mkpar
```
To configure the mkpar tool, open the mkpar.cmd file in a text editor and set the values of the following variables:

**JAVA**

Set this variable to the directory path and file name of the Java executable (for example, C:\hostondemand\bin\java.exe).

**MKPAR_HOME**

Set this variable to the directory where the mkpar tool resides (for example, C:\mkpar).

**Note:** If you are using mkpar on Windows ME, Windows 98, or Windows 95, you must rename the mkpar.cmd file to mkpar.bat.

### Running the mkpar tool

Now that we have the HOD HTML file and configured the mkpar tool, we will generate the PAR file containing the HOD portlet. The syntax of the mkpar tool is as follows:

```
mkpar [-options] codebase input_file output_file
```

where options includes one or more of the following:

- `-name=PortletName`
  
  Specifies the name of the portlet. Use quotation marks to enclose the *entire* -name option. If a name is not specified, the default name is HOD Portlet.

- `-version`
  
  Displays the version number.

- `-h`
  
  Displays help for the mkpar tool.

The other parameters are:

- **codebase** is the Uniform Resources Locator (URL) of the Host On-Demand publish directory.

- **input_file** is the HTML file or ZIP file generated by the Deployment Wizard.

- **output_file** is the name of the PAR file that contains the portlet code.

**Note:** If you are running the mkpar utility tool under Version 1.3 of the Java 2 SDK, ignore the following output message when processing a ZIP file:

- Allocating ZIP comments array
- Added ZIP comments HODCDLABEL

In the following example, the Host On-Demand server is installed on myserver.ibm.com and the Host On-Demand publish directory is under the alias *hod*. The HTML file created with the Deployment Wizard is my3270.html. The portlet output is portlet3270.par.

```
mkpar http://myserver.ibm.com/hod/ c:\hostondemand\hod\my3270.html portlet3270
```

### 2.7.2 Deploying and configuring an HOD portlet

If you are familiar with adding portlets to WebSphere Portal Server, then adding an HOD portlet will be very simple. The process consists of installing the PAR file into WebSphere Portal Server and then making it available to your portal end users.
Installing the HOD portlet in WebSphere Portal Server

1. Browse to your default WebSphere Portal Server page and click the Login icon. The URL will be something like: http://[yourservename]/wps/portal.

2. Login with the default administrator user ID and password: wpsadmin - wpsadmin. See Figure 2-11.

3. Next, select the Administration tab as shown in Figure 2-12.
4. Now click the **Install** button from the Portal Application Administration window (as shown in Figure 2-13).

5. If you copied the PAR file to the installation directory in WebSphere Portal Server then you will see it in the list. Otherwise, click the **Browse** button to specify your PAR file location. Look for the PAR file you created with the mkpar utility.
6. Highlight this file and click the **Open** button.

7. Once you see the PAR file name in the edit box click the **Import** button. See Figure 2-14.

![Figure 2-14 Portlet look-up](image)

8. Next, click the **Continue** button, as shown in Figure 2-15.
9. Finally, click the **Finish** button to bring you back to the Portal Application Administration window, shown in Figure 2-16.

10. Your HOD portlet is now installed but needs to be activated. To do this, look for your HOD portlet from the list of installed portlet applications and highlight it. Next, click the **Activate** button to begin the activation process, as shown in Figure 2-17.
11. Next click **Continue** from the portlet activation window shown in Figure 2-18.

12. Finally, click **Finish** to complete the activation process, as shown in Figure 2-19.
You should now see that the HOD portlet is active (see Figure 2-20).

**Deploying your HOD portlet**

While still in the Administration Page, scroll down to the Access Control Administration window and click the Go arrow icon, as shown in Figure 2-21.
Look for your HOD portlet and select the **View** check box (Edit and Manage functions are not currently available for the HOD portlet) and click the **Save** button.

Your HOD portlet is now available for your end users. Host On-Demand portlet permissions are illustrated in Figure 2-22.
2.7.3 Adding an HOD portlet

In order to test your HOD portlet, you will need to add it to a new or existing portlet page in WebSphere Portal Server. In this example we will be adding it to a new page we're calling HOD Page. In order to do this, you first need to sign in to WebSphere Portal Server. If you have not yet created a profile in WebSphere Portal Server, you must do this first before proceeding. For details on how to create a profile, refer to the WebSphere Portal InfoCenter.

1. Go to your default WebSphere Portal Server page by entering the following URL: http://[server_name]/wps/portal. If you are not sure of the server name or you encounter an error, consult your systems administrator.

2. Click the **Sign-In** icon, as shown in Figure 2-23.

![Figure 2-23 WebSphere Portal Server Login Screen](image)

3. Next, enter your user ID and password.

4. Select the **Customize my portal pages** icon from the WebSphere Portal Server welcome page, as shown in Figure 2-24.
Figure 2-24  WebSphere Portal Server welcome page
Figure 2-25 details the flow for adding a portlet.

1. Type a name for your new page in the current page edit box.
2. Click the Add button.
3. Highlight your HOD portlet name from the list of available portlets.
4. Click the Add to page button.
5. Click the Save Page button.
6. Click the Close button to go to your newly created portlet page. This will start the download of the HOD Applet and establish your connection to the legacy system.

Your configured sessions will be displayed as shown in Figure 2-26.
2.7.4 Current limitations

At this time, the following limitations apply when using Host On-Demand through WebSphere Portal Server versus directly from the Host On-Demand Server:

- Multiple Host On-Demand portlets cannot be run on the same portal page.
- If you configured a cached version of the HOD applet, each machine that is used to access the portlet caches the Host On-Demand client.
If you are employing a configuration server model where session configuration data is saved on a user's local machine, then an update made by a user from one machine is not available if that user accesses the portlet from a different machine.

Host On-Demand bookmarking is not supported in the WebSphere Portal Server environment and is disabled by the mkpar tool.

If the applet size is not configured in the Deployment Wizard, it will default to fixed size, medium.

Note: For more details about limitations, see the readme file provided with the Host On-Demand product.

### 2.8 Security and administration

The HOD Logon is designed not just for security but also to allow an administrator to deploy and manage specific session options for different users. We refer to this as the Configuration Server Model. The reason why you might use this is that there may be a need for a subset of users to use customized keyboard settings. These settings are different from what the rest of the population will use. In order to accomplish this, an administrator would create two groups, each with a session configured with the desired keyboard settings. They would then assign users to the group that contained the session with the proper settings. When a user launches HOD, they are prompted for a user ID and password. After entering this information, they get the session configured for their group. For more details, refer to *IBM Host Access Client Package*, SG24-6182.

**Basic authentication**

Authentication of a user is performed by WebSphere Portal Server. Once authenticated, a user can add the HOD portlet to his/her customized page(s). The WebSphere Portal Server Administrator can decide the group(s) or user(s) that should have view access to the HOD portlet.

**Windows domain logon**

When using the configuration model, users are prompted to log in. If you don’t want users to have to see this prompt, you can enable the integrated Windows domain login. By enabling this feature, access to HOD will be determined by a successful login to the Windows domain. Refer to *IBM Host Client Access Redbook* for requirements and details on setting this up.

**Express logon**

The ability to automate the login to the OS/390 was delivered with the Version 5 release of Host On-Demand. By implementing this functionality along with the HOD portlet it is possible to add the same consistency users see when logging on to other Web applications. For details on how to set up this feature, refer to *IBM Host Access Client Package*, SG24-6182.

**HOD portlet administration**

The HOD portlet can be managed like any other portlet in WebSphere Portal Server. It can be distributed to all user(s) or only specific groups. Refer to WebSphere Portal Server InfoCenter for details. If session configuration changes are required then you can use the Host On-Demand Wizard to update or create new HTML files. These can then be converted to PAR files and redeployed as explained in 2.7.2, “Deploying and configuring an HOD portlet” on page 33.
2.9 Capacity planning and performance tuning

Since you will be running more than just the HOD portlet in WebSphere Portal Server, capacity planning for only this application is beyond the scope of this Redpaper. Please refer to the WebSphere Portal Server InfoCenter for details on this topic.

There are a number of configuration options available for tuning Host On-Demand. For details on capacity planning and performance tuning for HOD, refer to IBM Host Integration: A Practical Approach to Performance and Planning, SG24-5960-00.

2.10 Problem determination and troubleshooting

The implementation of a portal solution requires a number of different components. It is vital that you check the latest documentation available for each product before starting your installation. Refer to the WebSphere Portal Server InfoCenter for specific product requirements.

After completing your installation of WebSphere Portal Server load the portal_test.html file and follow the instructions in that file. In addition check the WebSphere Administrative Console for settings specific to WebSphere Portal Server.

Below is some troubleshooting information from the WebSphere Portal Server InfoCenter:

<table>
<thead>
<tr>
<th>Problem</th>
<th>Action</th>
</tr>
</thead>
</table>
| Installation fails during Database Server Configuration | Check the portal log and install log files for exception errors caused by insufficient disk space:  
  ▶ Verify there is enough space to install the portal (50 MB).  
  ▶ Verify that there is an additional 50 MB of disk space for your temporary directory. For Windows systems, the temporary directory is usually the C drive. For AIX and Solaris systems, it is the partition where /tmp is mounted. |
| Unable to log in to the Administrative Console for WebSphere Application Server | Perform the following steps:  
  1. If you installed SecureWay Directory, the WebSphere Administrative login user ID and password changes to wpsbind automatically  
  2. Type wpsbind for both user ID and password in the login at the target server window. |
| HTTP errors shown instead of default portal page | Perform the following steps:  
  1. Verify that WebSphere Application Server is running.  
  2. Start or restart WebSphere Application Server. |
| When accessing the portal on Windows system, the following error message is displayed: Requested Action not found: IBMSessionValidator Auth | The jaas.jar file is missing. During a previous uninstall of WebSphere Portal Server, the jaas.jar file could not be deleted because the file was locked by the Java Virtual Machine. The file was scheduled to be deleted after the next system restart, but the new installation was started before the system was restarted. Consequently the jaas.jar file for the new installation was deleted.  
  1. Uninstall WebSphere Portal Server.  
  2. Restart the system.  
  If reinstalling is not an option then contact product support for a copy of the jaas.jar file. |
2.11 Deployment scenarios

Below are some typical scenarios and possible implementation of the HOD portlet within WebSphere Portal Server.

Intranet

In this scenario a company has WebSphere Portal Server deployed in a call center within their organization. It provides a “one stop” resource for information. In addition, they have legacy application emulation or a Web-based emulation that these representatives use for back-end connectivity. These users have also some predefined macros that allow them to move through the application with efficiency. This company wants to integrate these two tools because having one interface will allow the representatives to respond quickly to customer inquiries.

A possible implementation for the HOD portlet would be the cached client with the configuration server. In this solution the HTML file created with the Deployment Wizard would:

- Cache the HOD applet
- Provide predefined macros for the sessions as buttons on the toolbar
- Prevent users from making session changes
- Download only 3270 emulation functions
- Run embedded sessions within the portal window

In this scenario when the call center representatives need access to a legacy application, they just go to their HOD page from within WebSphere Portal Server. The first time that this page is selected the applet is downloaded and stored on their local machine. The preconfigured emulator sessions, with any required macros, start up automatically. Users are not allowed to make or save any changes to these sessions. However, they can select other portlets or pages within WebSphere Portal Server and the state of their HOD sessions remain intact.

Extranet

In this next scenario there is a financial services firm that has a WebSphere Portal Server for their brokers. This portal acts as a central access point for research and marketing material. In addition, this firm also provides secure mainframe access for transaction processing.

A possible implementation for the HOD portlet would be the cached client that starts an SSL configured session with preferences saved on the local workstation. In this solution the HTML file created with the Deployment Wizard would:

- Cache the HOD applet
- Establish a SSL TN3270 session
- Permit users to make session changes
- Run within the portal frame in WebSphere Portal Server

2.12 Where to find more information

For a more comprehensive review of the topic covered, please refer to the following publications:

- IBM Host Access Client Package, SG24-6182
- IBM Host Integration in a Secure Network: A Pratical Approach, SG24-5988
- IBM Host Integration: A Practical Approach to Performance and Planning, SG24-5960-00
Installing and using the WebSphere Host Publisher portlet

This chapter provides information about how you will prepare your Host Publisher server in order to install, activate, configure and run the Host Publisher portlet.

We include several scenarios where the Host Publisher portlet is used to access various types of Host Publisher applications using, for example, chaining, composite applications, looping, conditionals and other important features related to mainframe applications, such as connection pooling and security.

Note: The Host Publisher portlet (HostPubPortlet.par) was not included in the Host Publisher V4.0 general availability release and you should visit the Host Publisher Web site for information about how to obtain this portlet. The Host Publisher Web site is http://www.ibm.com/software/webservers/hostpublisher/.
3.1 Host Publisher portlet

The IBM Host Publisher portlet allows for the integration of applications created in Host Publisher Version 4.0 with WebSphere Portal Server Version 2.1. The benefit of using the provided portlet is that you can now provide legacy application content along with Web-based portal content. In addition, the portlet leverages some of the capabilities provided by Host Publisher. For example, you can combine information from different legacy applications and present them within the same portlet.

A key feature of the Host Publisher portlet is that it provides state management. What this means is that you can have a portlet that interacts with existing legacy applications and does not lose the connection to the session when users navigate to other portlets or pages within WebSphere Portal Server.

Figure 3-1 illustrates a sample configuration used in the scenarios included in this chapter where the Host Publisher portlet is used to access mainframe (CICS) applications. Notice that the portlet runs on WebSphere Portal Server V2.1 under WebSphere Application Server V3.5.4, whereas Host Publisher V4.0 runs under WebSphere Application Server V4.0.2.
3.1.1 Limitations

The following limitations apply to this version of the Host Publisher portlet:

- Applications using frames are not supported.
- The Express Logon Feature (ELF) is not supported. This is mainly because in this environment Host Publisher Server does not have direct access to the required client certificate.
- Secure Sockets Layer (SSL) restrictions. A secure session cannot be established directly (end-to-end) from the browser to the Web server in the Host Publisher Server machine. However, you can establish an SSL session from the browser to the Web server in the WebSphere Portal Server machine and optionally enable a second SSL session from the Portal Server to the Host Publisher Server.

**Note:** Secure sessions (using SSL) cannot be established end-to-end between browser and Host Publisher Server.

- NLS restrictions. The Host Publisher portlet is English only.
- HTML content only. The Host Publisher portlet supports HTML content only. However, you can use transcoding to support other markup languages such as WML (for WAP devices) and cHTML (for i-Mode devices).
- WebSphere Portal pages contain nested HTML pages and portlets are rendered within a HTML table cell. When running Host Publisher applications presenting HTML tables, some nested tables may not render properly when using a Netscape browser prior to Version 6.

3.2 Major Host Publisher components

There are two major components in WebSphere Host Publisher, the Host Publisher Studio (application development tool) and the Host Publisher Server (runtime). In this section, we briefly describe some of the most important functions related to the Host Publisher portlet.

**Host Publisher Studio**

Host Publisher Studio is a collection of task-oriented, easy-to-use graphical user interfaces that assist the Web application builder in managing and creating Web-to-host publishing projects. It uses task-oriented prompts to guide the user through the creation process, including recording host and database interactions, identifying desired data, and labeling that data for retrieval.

Host Publisher Studio automatically generates JavaBeans called Integration Objects, which encapsulate the interactions and data retrieval logic. You can use Host Publisher Studio to generate fully customizable Web pages for modeling interactions with the Integration Objects and rendering the resulting data. You can enhance the generated Web pages with your favorite Web authoring tool such as WebSphere Studio Application Developer.

**Host Publisher Server**

Host Publisher Server provides the runtime environment for supporting J2EE applications created with Host Publisher Studio. Running on the application server with the IBM WebSphere Application Server product (WebSphere), it includes components such as connection management, license monitoring, runtime administration, express logon, XML Gateway, and log and trace management.
3.3 What’s new in Host Publisher V4.0

As part of the WebSphere Version 4.0 environment, this new release of Host Publisher contains several new enhancements. In this section we provide a high-level overview; for more details see Host Publisher V4.0 User’s Guide provided with the product.

J2EE application support
Applications produced by Host Publisher Studio comply with J2EE, an industry-standard architecture that is intended to reduce the cost and complexity of developing enterprise applications. J2EE applications can be deployed rapidly and enhanced easily as the enterprise responds to competitive pressures. A J2EE application takes the form of an EAR (Enterprise Archive) file into which all the application's pages, Java objects, and resources are assembled.

JavaServer Pages (JSP) V1.1 support
Host Publisher Studio now produces JSP pages at the JSP 1.1 level. Applications with JSP V1.0 tags will still run, but applications with JSP V0.91 tags (created prior to Host Publisher Version V3.5) need to be migrated. Two migration tools, one on the Studio machine and one on the server, are provided with the product.

Enterprise JavaBeans (EJB) V1.1 support
Host Publisher now builds EJB-based applications supporting the EJB 1.1 specification level. EJB Access Beans developed with an earlier version of the product must be migrated to the 1.1 level; a migration tool is provided as part of Host Publisher Studio.

Web services
When creating your Host Publisher Integration Object you have the option to also enable them for Web services. You can then bring these objects into WebSphere Studio Application Developer to complete and deploy the actual Web service.

Multi-language support
In Host Publisher Server V4.0, you can now use Host Publisher Server Administration and view the Host Publisher documentation in a language other than the server’s default language.

Serviceability
The Software Maintenance Utility, a new command-line tool, can help you apply software fixes. For situations that require the involvement of the IBM Support Center, this tool also scans the product and creates a package containing documentation and files for the IBM support team to use in troubleshooting.

Note: If you have Host Publisher applications on the server that were developed with an earlier version of the product, they must be migrated to Version 4.0.

3.4 Installation of Host Publisher V4.0

In this section we briefly describe the requirements, software and hardware for the basic structure necessary to set up a Host Publisher Server V4 including the installation of the Host Publisher Server and Host Publisher Studio, as well as the WebSphere Application Server Version 4 prerequisites.
Host Publisher Studio V4.0 runs under Windows 98, NT V4, 2000, XP and ME. However, Windows 95 is not supported.


**Note:** You should read the product announcement letter to verify that you are using the correct product and level required by Host Publisher Server.

### 3.4.1 DB2

In a Host Publisher environment, a relational database such as DB2 is used by the WebSphere Application Server and it can also be used to access data from Host Publisher database Integration Objects (JavaBeans).

The following database servers are supported using the JDBC interface:
- DB2
- Informix
- Oracle
- SQL Server
- Sybase

We recommend verifying the WebSphere Application Server and WebSphere Host Publisher documentation for more details about required versions and JDBC drivers.

The requirements for these servers vary according to the version and edition of the Database Server. The configuration requirement for the DB2 Enterprise - Extended Edition Version 7 is for example:

- Memory - 128 MB recommended if you plan to use the administration tools
- Disk space - approximately 420 MB for a typical install, including the JRE

**Tip:** Use InstantDB for samples only; for a production environment a real database such as DB2 server is required.

### 3.4.2 WebSphere Application Server V4.0

Host Publisher V4.0 uses IBM WebSphere Application Server Advanced Edition V4.0 to provide a consistent, reliable execution environment for the Host Publisher applications, for example, servlets or EJB-based applications, and HTML/JSPs across platforms.

**WebSphere Application Server requirements**

The following software is required:

- Web server (one of the following):
  - Apache Server
  - IBM HTTP Server (included with WebSphere Application Server)
  - iPlanet Enterprise Server
  - Lotus Domino Application Server
  - Microsoft Internet Information Server
- WebSphere Application Server (one of the following):
  - Advanced Edition (AE) V4.0.2
  - Advanced Edition Single Server (AEs) V4.0.2
  - Enterprise Edition (EE) V4.1
  - Advanced Edition for Developers (AEd) V4.0.2
The following hardware is required:

- **Intel**
  - 500 MHz Pentium
  - 384 MB RAM minimum, 512 MB recommended
  - 180 MB free disk space for the base application server, 2 GB recommended for a full installation (including DB2 and HTTP server)

- **Non-Intel platforms**
  - AIX: RS/6000
  - HP-UX: HP 9000
  - Solaris: Sparc
  - 384 MB RAM minimum, 512 MB recommended
  - 220 MB free disk space for the base application server, 2 GB recommended for a full installation (including DB2 and HTTP server)

### 3.4.3 Host Publisher Server

The Host Publisher Server is available in the following operating systems:

- AIX
- Solaris
- Windows NT
- Windows 2000
- Other versions are also available in i-Series and z/OS systems.

**Note:** An FTPD product must be installed and enabled to remotely transfer applications from Host Publisher Studio to the Server.

On Windows NT and Windows 2000, with Host Publisher Server and Host Publisher Studio installed on the same machine, you can use the localhost option. See the *IBM WebSphere Host Publisher Administrator's and User's Guide* for more information.

#### Hardware requirements

In addition to those required by WebSphere Application Server, the following resources are required:

- 130 MB of disk space, add more for your applications if desired
- 128 MB RAM minimum, 256 MB RAM recommended

**Note:** These requirements are in addition to the WebSphere Application Server requirements. For example, if WebSphere Application Server requires 512 MB RAM, and Host Publisher Server requires 128 MB RAM, you must have a total of 640 MB RAM installed in your machine.

#### Installation process

Before you start the Host Publisher Server installation process, it is important to verify that WebSphere Application Server has been started and running.

**Note:** You should also be certain that the correct WebSphere Application Server maintenance has been properly installed.

#### Verifying WebSphere Application Server

The easiest way to verify that WebSphere Application Server is up and running is by using its administration tool to view and monitor its resources.
1. Click **Start -> Programs -> IBM WebSphere -> Application Server V4.0 AE -> First Steps - > Launch the Administrative Console**, found on the welcome window shown in Figure 3-2.

![Figure 3-2  WebSphere Application Server - First Steps](image)

This option opens the WebSphere Application Server console and you can start the Default Server Application to test the environment.

2. Expand the tree by selecting **WebSphere Administrative Domain -> Nodes -> your machine hostname -> Application Servers**, as shown in Figure 3-4.

3. Select the **Default Server** and click the **Start** button.
4. As illustrated in Figure 3-4, once the default server is up and running you will receive a message indicating that the server was successfully started and the default server icon will turn green. If this does not happen, you will need to verify the Event Message window to identify the possible cause of the problem.

5. An easy way to verify that the WebSphere Application Server default server is by invoking the snoop servlet from a browser using the following URL:

   http://your_server_host_name/servlet/snoop

6. If the operation is successful, you will see the window shown in Figure 3-5.
The fact that the Snoop servlet runs successfully can be taken as an indication that WebSphere Application Server is up and running. You can then start the Host Publisher Server installation.

**Installing the Host Publisher Server**

To install the Host Publisher Server, do the following:

1. Start by inserting the Host Publisher Server CD-ROM in the CD drive and the autorun program will start the Host Publisher installation process as shown in Figure 3-6.

2. Select the **Install** option to start the installation process (see Figure 3-7).
3. After accepting the license agreement and entering the target directory (default value is `c:\HostPub`), enter the number of licenses acquired. The number of the licenses is only for reference; it does not impact the application after exceeding the maximum number. You can change it after the installation using the License Management feature under the Host Publisher Administrator.

4. On the next window (see Figure 3-8) you have the opportunity to migrate applications to the new version of the Host Publisher.

![Figure 3-7 Host Publisher Install window](image)

**Figure 3-7** Host Publisher Install window

![Figure 3-8 Migrating applications](image)

**Figure 3-8** Migrating applications

**Important:** Previous Host Publisher applications must be migrated to Host Publisher Version 4.0

5. After the installation process you have an opportunity to register the product and reboot your computer to complete the Host Publisher Server installation. See Figure 3-9.
Chapter 3. Installing and using the WebSphere Host Publisher portlet

Figure 3-9  Registration and reboot window

**Executing Host Publisher Server**

After the Host Publisher Server machine is restarted, in a Windows system you can bring up the IBM WebSphere AdminServer V4.0 service and start the WebSphere Administrator’s console again to start the HostPubServer application.

Invoke the Host Publisher Server Administration and verify the server status to make sure that Host Publisher Server has been successfully installed (see Figure 3-10). Click **Start -> Programs -> HostPublisher -> HostPublisher Administration**.

Figure 3-10  Host Publisher - Server Status
3.4.4 Host Publisher Studio

One of the following operating systems is required to run Host Publisher Studio:

- Windows NT 4.0 (Workstation or Server)
- Windows Me
- Windows 98
- Windows 2000 (Professional, Server, or Advanced Server)
- Windows XP

**Note:** Host Publisher Studio does not support Windows 95.

Host Publisher Studio has the following hardware requirements:

- Pentium II 366 processor or higher, Pentium II 450 processor recommended
- Minimum of 128 MB RAM, 256 MB RAM recommended
- 110 MB of free disk space, add more for your applications, 1 GB recommended
- A display with a color depth of 256 colors or higher, and a minimum screen resolution of 800 x 600 pixels

**Installing the Host Publisher Studio**

Host Publisher Studio installation is very straightforward and installs directly to a Windows target directory that you specify. Default value is c:\HostPub. The installation process does not require that you reboot the Windows machine.

**Executing Host Publisher Studio**

There are three major components in Host Publisher Studio that you can start from the desktop by clicking **Start -> Programs -> Host Publisher Studio**. The Host Publisher Studio components are:

- Application Integrator. Allows you to create Host Publisher applications by generating JavaServer Pages (JSPs). This component was called Studio or Web Bridge in previous releases.
- Host Access. Allows you to create host Integration Objects (JavaBeans).
- Database Access. Allows you to create database Integration Objects (JavaBeans).

3.4.5 WebSphere Studio Application Developer

The WebSphere Studio Application Developer is optional software that allows you to create your own Java application using the JavaBean Integration Objects created by HostPublisher Studio.

The use of this tool is highly recommended for developers who want to run and test their applications in their local machine before deploying them to the HostPublisher Server.

The WebSphere Studio Application Developer is also recommended if you want to create a Web Services application. A sample procedure is described on the **Host Publisher V4.0 Implementation Guide** on the HostPublisher Library session. See:

WebSphere Studio Application Developer has the following software and hardware prerequisites:

- Windows 2000 Professional, Server, or Advanced Server
- Windows Me, Windows 98, and Windows XP
- Windows NT 4.0 with Service Pack 6a or higher
- Microsoft Internet Explorer 5.5 with Service Pack 1 or higher
- TCP/IP installed and configured
- A mouse or an alternative pointing device
- Pentium II 300 processor or higher recommended
- SVGA (800 x 600) display or higher (1024 x 768 recommended)
- 256 MB RAM minimum
- Disk space requirements: 400 MB minimum (based on NTFS, actual disk space on FAT depends on hard drive size and partitioning)

3.5 Host Publisher applications

In this section we briefly describe how you will go about developing a Host Publisher application in order to be accessed using the Host Publisher portlet. In a typical Host Publisher application you will need to create Integrated Objects using Host Publisher Studio and then create JSPs for your Web application.

Note: In many cases you will need to change the application look and feel in order to make it more friendly to the user. For this you can use WebSphere Studio Application Developer.

3.5.1 Creating Host Publisher applications

When you want to develop a Host Publisher application you must first define Integration Objects (JavaBeans) using Host Access or Database Access in Host Publisher Studio. You will then import the Integration Objects into the Application Integrator to create Host Publisher applications using JSPs. In this section a very simple sample application is developed to access CICS transactions. It is assumed that you are familiar with the development of applications using Host Publisher.

Other sample scenarios using macro looping, conditionals and chaining Integration Objects are discussed in 3.5.3, “Other scenarios using macro looping, conditionals and chaining” on page 70.

Host Access

Use this tool to create Integration Objects and connection pools. Connection pooling is an option when using host Integration Objects in IBM WebSphere Host Publisher. By default, Host Publisher uses non-pooled connections (dedicated connections). When a user requests information from a host access Integration Object on a Web page, Host Publisher connects to the host, logs on, extracts the information, and logs off. To reduce the time it takes to return information, Host Publisher allows you to keep pooled connections in place once the application that requested a connection is no longer communicating.

Note: You can start Host Access from the desktop by clicking Start->Programs->Host Publisher Studio->Host Access.
Figure 3-11  Host Access Studio - Connection Pool

Define the Connect, Data and Disconnect Macro for the Integration Object using the connection pool that was previously defined. Figure 3-12 on page 63 shows the main path for a simple Integration Object. The created sample application is an inquiry for an account number in a credit account application running in a CICS host application.
Once the Integration Object JavaBean is created, the next step is to create an application using the Application Integrator in the Studio.

**Application Integrator**

The Studio provides a development environment where you can create Web applications and manipulate the Integration Objects created using the Host Access and Database Access tools. It is useful to note that you are not required to use the Application Integrator to create a Host Publisher application. However, it is recommended for a quick and easy way to initiate the application development process.
Figure 3-13, is the first window you see when using the Application Integrator. In this window select the **Create Application** button to start the Host Publisher development, which will typically include an input JSP, and output JSP and an error JSP.

Follow the wizard’s prompts, where the options for the JSPs will be specified and the Integration Objects properties will be defined. You will need to create an input form including the required input fields, an output page to execute the Integration Object JavaBean and display the results, and an error page to display and show any problems. When you finish with this wizard, you will have the input, output, and error JSPs required by the Host Publisher application.

For this sample scenario, the results are shown in Figure 3-14 where the created input page includes the input form used to post the output page once the user enters the required input fields and submits the request.
In this scenario the created JSPs are very simple and if you want, you could edit additional changes, and therefore provide a more user-friendly interface for the application. Additional options you probably would like to include (manually or using a tool) in the JSPs are:

- Font
- Style
- Alignment of text to be displayed in the page
- Elimination of the blanks rows in tables
- Images

### 3.5.2 Using WebSphere Studio Application Developer for further development

WebSphere Studio Application Developer is an excellent tool to update JSPs developed with Host Publisher Studio. It allows you to define a test environment where you could run and verify any updates you made to the application, without the need to deploy the application into a Host Publisher Server.

In addition, WebSphere Studio Application Developer provides a debug facility you can use in case of problems.

When you create an application in Host Publisher, an EAR file (enterprise application) is generated. You can then use this file to import it inside of WebSphere Studio Application Developer. Figure 3-15 shows the option to import the EAR file. You will need to select a project of this type in the window and then follow the required steps until completed.
Figure 3-15 Importing an enterprise application into WebSphere Studio Application Developer

The imported Host Publisher enterprise application (EAR file) is shown in Figure 3-16 where the application input page (input.jsp) has been selected. The page can now be updated with new features such as fonts, styles and so on.

Figure 3-16 Updating a Host Publisher page in Application Developer
Once you are done with any additional enhancements to the Host Publisher application pages you can locally run the application using the WebSphere Test Environment provided by the WebSphere Application Developer.

**Host Publisher Server servlets**

Before you start execution of the application, it is necessary that you make a few settings in the WebSphere Studio Application Developer environment so that Integration Objects can be recognized and executed. Host Publisher Studio provides two servlets that allow you to create and enable a Host Publisher test environment. The servlets are used to start and stop the Host Publisher Server instance in the WebSphere Studio tools environment (WebSphere Studio Application Developer).

- StartHPRTE.java servlet initializes and starts the Host Publisher Server instance.
- StopHPRTE.java servlet stops the Host Publisher Server instance.

You will need to add these servlets to a new project for execution. The servlets can be found in the `\HostPub\SDK\Server\*` directory. Figure 3-17 shows the installed Host Publisher servlets in the Application Developer.

![Server - Application Developer](image)

Figure 3-17 Installed Host Publisher servlets

**Configuring the Java path for Host Publisher**

You also need to configure the classpath before you can execute the Host Publisher application inside the Application Developer. That is, the classpath needs to be updated to include the Host Publisher JAR files. Failing to do this will give an error when executing the Host Publisher servlets.

In the properties for the Host Publisher servlet project you need to update the Java Build Path with the JAR files in directory: `c:\HostPub\Common\*\*.jar`. Figure 3-18 shows the libraries for the project. Click Add Externals JARs.
Executing Host Publisher applications

At this time you are now ready to run your Host Publisher application in WebSphere Studio Application Developer but first you will need to start the WebSphere Test Environment. Therefore, if not already started, right-click the WebSphere Test Environment server and start it. Figure 3-19 shows the Test Environment started.

The next step is to start Host Publisher before you actually run your application. For example, right-click `StartHRPRTE.java` servlet and select the option Run on Server. Figure 3-20 illustrates Host Publisher Run Time Environment (RTE) successfully started in the Application Developer.
Once Host Publisher is available to run applications, start your application by right clicking on the input page and selecting Run on Server. Figure 3-21 shows this process.

Next, you will enter any required input and submit the request to post the Host Publisher application output page (JSP). The output page instantiates the Integration Object JavaBean and gets the output properties to be displayed on the window. Figure 3-22 shows a sample result for this scenario.
3.5.3 Other scenarios using macro looping, conditionals and chaining

Using Host Access in Host Publisher Studio, you can develop applications to access 3270 host applications using looping, conditionals and chaining features. In this section we describe these features and show you how can also integrate them into a Portal Server environment.

Macro looping

If your host application response results in multiple screens, you will need to use macro looping to be able to extract and display output data, typically in the form of a table. In this section, we will use Host Publisher to capture data shown in multiple screens and present that data in one single HTML page. To make this possible, you will need to use the looping feature resource in Host Access.

The sample loop macro for this scenario is shown in Figure 3-23 on page 71. The Integration Object will capture the data of all screens until it finds the last one in the loop. The sample application creates a query for a variable list of data. For example, the type of information the user is looking for is a list of Redbooks available within the IBM ITSO catalog system, based on a keyword search.
The generated macro is illustrated in Figure 3-24.
Perform the following steps to create the application:

1. Create the Integration Object including the connect, data and disconnect macros.
2. Generate and save the Integration Object.
3. Import the Integration Object into the Application Integrator in Host Publisher Studio to create an application using JavaServer Pages.
4. Create the application JSPs, that is input, output, and error pages.
5. Create the enterprise application in an EAR file.
6. Import the application EAR file into WebSphere Studio Application Developer. This step is explained in 3.5.2, “Using WebSphere Studio Application Developer for further development” on page 65.
7. If required, use WebSphere Studio Application Developer to improve the look and feel of the Host Publisher application.
8. Test your application in the WebSphere Studio Application Developer Test Environment.
9. Deploy your application into WebSphere Application Server. This step is explained in 3.6, “Deploying HostPublisher applications” on page 79.
10. Execute the final version of the application.

Figure 3-25 shows the input page for the looping application where you have to enter an input value to be used as a search argument in the Redbooks database.

![Figure 3-25 Input page for looping application](image)

Figure 3-26 shows the search results in an HTML table. The table contains data captured from several screens for all entries found satisfying the search word and until the application found the last entry in the loop.
Integration Object chaining

The Integration Object chaining function in Host Publisher Studio enables multiple Integration Objects to execute in sequence, each using the same connection to back-end resources such as host applications and databases. You can use chaining to break up a complex application into multiple tasks, each task represented by an Integration Object. Host Publisher Studio ensures that the order in which Integration Objects are invoked is correct.

In this section, a sample scenario is presented where two host Integration Objects are used in a chain (first and last). This Web application is created that will ask the user to enter an account number, display information about that account using the first transaction type, allow the user to change information for an account, and allow a user to open a new account and to delete an existing account by using one of the following transaction types:

- Show account details.
- Add a new account.
- Update an account.
- Delete an account.

Figure 3-27 and Figure 3-28 show the two Integration Objects for this application, that is the first-in-chain and the last-in-chain Integration Objects. Notice that middle-in-chain Integration Objects are also possible but for simplicity they are not used in this scenario.
Figure 3-27  First-in-chain Integration Object macros

Figure 3-28  Last-in-chain Integration Object macros
Figure 3-29 shows the flow for the Web application using Integration Object chaining. The JavaServer Pages in this application were also improved using the WebSphere Studio Application Developer. The sample transaction shows the input and output pages when the user wants to display details for an specific account.

Using conditionals

In some situations, you will find application flows that need to run using an alternate path of the application and your macro needs to handle both the application main path and any other alternate path. To make this possible, you need to use the conditional macro resource of Host Access when building your Integration Object. In general, conditionals occur any time two or more screens can be received from the host applications. This occurrence is unpredictable most of the time.

For example, the application shown in Figure 3-30 on page 76 will flow most of the time from screen 1 to screen 2 (main path) and in some cases a new screen 1.1 will also need to be considered (alternate path). In other words, these applications have a main path most of the time, but sometimes an exception occurs and an alternate path is required.
In this scenario, we have the Credit Account application, and an exception can occur if the user types an invalid input account. The application must be prepared to recognize when that exception occurs, redirect the user to the input page again and not go the main path that is the usual flow process.

This scenario presents the following paths:

- A main path: Input page -> Output page
- In cases when the user enters an invalid input account the alternate path is: Input page -> Input page with error message

Figure 3-31 on page 77 shows the Integration Object using a conditional macro created with Host Access.
Figure 3-31 Integration Object including a conditional option

Figure 3-32 shows the application JavaServer Pages including improvements made with WebSphere Studio Application Developer. If the user enters an invalid account, the alternate path is invoked and a message is displayed in the input page. The error message can be passed from the output page to input page by using the Java HttpSession class. In addition, when the user enters a valid account the main path is executed and the output page with valid results is displayed.
3.5.4 Recommendations

There are several issues you may want to consider when you develop Host Publisher applications to be used in a WebSphere Portal Server environment so that they operate without problems.

Using JavaScript

In general, we recommend using JavaScript for validation of fields in a form page only and not to make reference to URL links, since these links will be used inside the Portal Server. The Host Publisher portlet containing the application validates references to links. For this reason, it is recommended that you use a specific URL in your links so that the portlet will recognize it. Example 3-1 shows an incorrect use of reference links.

Example 3-1 Incorrect use of JavaScript when using the Host Publisher portlet

```html
<SCRIPT LANGUAGE="javascript">
  //Call input page with the error msg
</SCRIPT>

<A HREF=JavaScript:history.go(-2)>Back</A>
```

Example 3-2 illustrates the correct use of reference links in a Host Publisher application page.

Example 3-2 Correct use of reference links when using the Host Publisher portlet

```html
<A HREF="input.jsp">Back</A>
```
HTTP session
Use the HttpSession class to pass data between pages in your Host Publisher application. For example, when you are working with conditionals and Integration Object chaining, the HttpSession object can be used to simplify the interaction between JavaServer Pages.

3.6 Deploying HostPublisher applications

The Host Publisher application deployment procedure has changed for this version of the HostPublisher. You will see that the HostPublisher application is more integrated with the WebSphere Application Server, and some new useful features have been implemented.

The process to deploy a HostPublisher Application is actually a file transfer of the created EAR file into the HostPublisher directory on the server. The application in the HostPublisher Studio remains the same, but during the transfer process the Studio creates a J2EE 1.2 EAR file or enterprise application.

3.6.1 Installing the Host Publisher application

In the WebSphere Administrative Console Wizard, use the EAR file in the Install Enterprise Application option to install the application in the WebSphere Application Server. The same EAR file can be used within WebSphere Studio Application Developer, allowing you to compose a new application using the Integration Objects and the JSP pages previously created with the Host Publisher Application Integrator.

Host Publisher Application Integrator
Use the HostPublisher Application Integrator to transfer your application to the server. This process creates an EAR file and sends it, using the FTP protocol, to the following directory:

    c:\WebSphere\AppServer\installableApps\HostPublisher\n
As an alternative, you can create the EAR application locally and send it to the server using a different method. In this case, click File -> Create J2EE Archives to save your file in the following directory:

    C:\Hostpub\Studio\Applications\your_application_name\
WebSphere Studio Application Developer

After importing the basic EAR file to the WebSphere Studio Application Developer and making all the changes and enhancements to your application, click **File -> Export**, or right-click the name of the application and select **Export EAR File**.

As illustrated in Figure 3-34, select your application in the What resources do you want to export? field and choose a directory and name for your new EAR file in the Where do you want to export resources to? box.
Installing the application

The installation process of the Host Publisher application is performed using the WebSphere Advanced Administrative Console (see Figure 3-35).

From the desktop, start the WebSphere Administrative Console by selecting Start -> Programs -> IBM WebSphere -> Application Server V4.0 AE -> Administrator's Console and select Console->Wizards->Install Enterprise Application as shown in Figure 3-35.
When selecting the option to install the application, use the browse option to locate the application EAR file and add the .ear extension to the application name as shown in Figure 3-36. Click Next to continue the installation.
Next, you will need to select the application server where your application will be installed. Therefore, click **Next** until the application server selection window appears and as shown in Figure 3-37 select the appropriate module, click **Select a Server or Server Group** -> **HostPubServer**. You will need to click **Next** to confirm the entered information and **Finish** to complete the installation process. You will receive an information window showing the results.

![Selecting Application Servers](image)

**Figure 3-37  Selecting the HostPubServer**

**Starting the application**

After you have installed the application you will need to regenerate the plug-in and start the application. As shown in Figure 3-38, select your node and right-click to select the **Regen Webserver Plugin** option. At this time you may want to verify the messages in the Event Message window for any errors.
The last task is to start the application by right-clicking the application name and clicking **Start**, as shown in Figure 3-39. After this, you will receive an information window showing that your application was successfully started.
As shown in Figure 3-40, once the application is ready you may run the Host Publisher application. For example use the following URL:

http://server_host_name/your_enterprise_application_name/your_default_page.jsp

![Running the sample Host Publisher application](image)

3.6.2 Installing the Host Publisher portlet in the Portal Server

The process to install and activate the Host Publisher portlet is simple, and consists of installing the portlet archive (PAR) file into WebSphere Portal Server and then making it available to your portal server end users. The Host Publisher portlet is available from the following Web site:

http://www7b.software.ibm.com/webapp/portlets/portletemarketplace

![Host Publisher portlet download file (PAR file)](image)

The process for installing the Host Publisher portlet into WebSphere Portal Server is similar to the HOD portlet installation. For details see 2.7.2, “Deploying and configuring an HOD portlet” on page 33.

**Note:** After you install the Host Publisher portlet, make sure you also make it active.
Deploying your Host Publisher portlet

In order to make your portlet available to your end users, you have to assign permissions to your portlet. This process is performed by accessing the Administration page. For example, while still in the Administration page you may want to scroll down to the Access Control Administration window and follow these steps (see Figure 3-42):

1. Select the proper Group or User to assign permissions (for example all users)
2. Select Portlets as the object for permissions.
3. Click the Go arrow icon.
4. In the list of installed portlets, look for your Host Publisher portlet and select the View and Edit check boxes and click the Save button. Your portlet is now available to be accessed by your selected end users.

**Note:** The manage mode is not currently available in this version of the Host Publisher portlet.

![Figure 3-42 Portlet permission window](image_url)
3.6.3 Configure the Host Publisher portlet to access applications

The Host Publisher V4.0 portlet supports view and edit modes of portlet operations. In edit mode, the configurable parameters will be presented for the user to modify, and in the view mode the user will be taken to the Host Publisher application. The goal of this portlet is that once the Host Publisher application has been accessed, the portlet will allow you to interact with it in the same fashion as if a browser window had been opened to access the application.

Adding a Host Publisher portlet

In order to run the Host Publisher portlet you need to add it to a new or existing portal page in WebSphere Portal Server. You will need to sign in to WebSphere Portal Server with a proper and authorized user ID. Also, if you have not yet created a profile in WebSphere Portal Server you must do this first before proceeding. For more details on how to create a profile, refer to the WebSphere Portal InfoCenter provided with the product.

Go to your default WebSphere Portal Server page by entering the following URL:

http://[server_name]/wps/portal

As shown in Figure 3-43, click the Sign-In icon which is the first icon in the upper right corner of the page.

![WebSphere Portal Server login window](image)

**Figure 3-43  WebSphere Portal Server login window**

Enter your user ID and password and select the option Customize my portal pages icon in the upper right corner of the WebSphere Portal Server welcome page as illustrated in Figure 3-44.
Next, you will need to customize the portal page by adding the Host Publisher portlet. Figure 3-45 on page 89 shows the details of the procedure to make the portlet available to a user or group of users.

1. Type a name for your new page in the current page edit box.
2. Click the Add button.
3. Highlight the Host Publisher portlet name from the list of available portlets.
4. Click the Add to page button.
5. Click the Save Page button.
6. Click the Close button to go to your newly created portlet page. This will make the Host Publisher portlet available in this page.
Chapter 3. Installing and using the WebSphere Host Publisher portlet

Figure 3-45   WebSphere Portal Server customization page

Configure the portlet in Edit mode
Once you have added the portlet to a page in your portal, you will need to configure it by selecting the edit mode option in the upper right corner of the portlet. This is shown in Figure 3-46.

Figure 3-46   Edit mode in Host Publisher portlet

In the edit mode window you configure the portlet parameters. You will need to provide the following information as illustrated in Figure 3-47:

- In the Target Host Publisher Application URL field, enter the URL of the application you want to run. It is a required parameter.
- In the Target Host Publisher Application Description field, enter a description or name for your application. This is an optional parameter.
Select the Using Forms Based Security check box if WebSphere Security is enabled on the server where the application is deployed.

When you click the Save button you will be switched into view mode where the target URL will be displayed and your Host Publisher application will be invoked (see Figure 3-48). If you wish to change to a different target application, you must select edit mode again and modify the target URL information.

The first invocation of the portlet will cause the configured URL (which points to the Host Publisher application) to be used. Subsequent invocations with the portlet will be interactions with the Host Publisher application.
However, if the user wants to start again at the beginning (with the configured URL), there is a Reset Host Publisher Portlet button, which will cause the portlet to begin again with the configured URL. When this reset button is clicked, there will be no automatic cleanup from the portal that will clean up the application and the session.

With Host Publisher Portlet you can have several different Host Publisher applications in the same page of your WebSphere Portal Server, and each one will work independently.

Figure 3-49 shows a page in the WebSphere Portal Server with four different Host Publisher applications. They use different Host Publisher features and options such as macro looping, conditionals, Integration Object chaining, and so on. For details on how these applications were created, see 3.5.3, “Other scenarios using macro looping, conditionals and chaining” on page 70.

![Host Publisher applications in the portlet](image)

3.7 Security and administration

Once you have transferred the applications to the Host Publisher Server, the deployment and administration of that application is done through the WebSphere Application Server. Note that this is different from previous releases of the product where application administration was done through the Host Publisher Server. For details on how to see 3.6, “Deploying HostPublisher applications” on page 79.

The administrative functions that are included in Host Publisher are for problem determination and connection management. The following is a list of functions that can be performed. For details, refer to the WebSphere Host Publisher Administrator’s and User’s Guide.
- Selecting a server, and an instance of Host Publisher Server in that server, to administer
- Monitoring server status: starting and stopping Host Publisher Server
- Supplying passwords required by Host Publisher Server
- Managing licenses: changing the allowed number of licenses on a server running one or more instances of Host Publisher Server
- Administering connections and connection pools: displaying pool definitions, pool status, and the status of active connections for Host Publisher Server
- Displaying user lists
- Performing problem determination: viewing trace and log files, and setting various options for tracing and logging
- Administering the XML Gateway: configuring the XML Gateway to access hosts and host applications, and creating a portal for accessing hosts and host applications

### 3.8 Problem determination and troubleshooting

For problem determination with your Host Publisher applications, you can set trace options in the Host Publisher server administration console. These traces can also be set to work with the application when they run in the WebSphere Portal Server environment.

**Host connection tracing**

In the Host Publisher Server Administration Console, select **Problem Determination -> Set Trace Options** to get to the Host Connection Tracing pane. In this section you can optionally select two options:

- User Tracing for Host On-Demand Macros
- Display Terminal

Select the **Display Terminal** check box to monitor the 3270 traffic and click the **Save** button as shown in Figure 3-50.
Next, you have to make sure that the IBM WebSphere AdminServer 4.0 properties are set correctly so you can see the display terminal. In the Control Panel in your system, select the **Services** option and find IBM WebSphere AdminServer, select the **Properties** option, and in the Log On tab select the option **Allow service to interact with desktop** as illustrated in Figure 3-51.

With this trace option, you are now ready to run your Host Publisher application and monitor the behavior of your developed macros in the Host Connection Terminal in your system.
**Note:** When using the Host Publisher display terminal trace option, a new trace window will be created every time a new host connection is established.

Figure 3-52 shows the interaction between an application and the Host Connection Terminal. This option is very useful if you have problems when running your Host Publisher applications through the Portal Server.

**Credit Account Demonstration**

**PLEASE ENTER ACCOUNT OR SURNAME**

Account Number: 
Surname: 
First Name: 
Address: 

![Figure 3-52 Host Publisher application with host connection tracing](image)

**Trace files**

When traces are selected for problem determination in your Host Publisher applications, you can use the Host Publisher Server Administration Console and select **Problem Determination -> Set Trace Options -> Trace File Name**. In this pane you can provide a name and a location for a trace file. Figure 3-53 shows this option.
Figure 3-53  Trace File Name option

Example 3-3 shows a sample trace file for the application presented in this chapter. It illustrates the actions in Host Publisher Server when playing a macro when executing an Integration Object.

**Example 3-3  Sample trace file**

```
14:30:55.634 HOD  EventThread_1
  HodConn:m23cacav_HostPubServer#1: 1  Entering text: 1
14:30:55.764 HOD  EventThread_1
  HodConn:m23cacav_HostPubServer#1: 1  Executing Action: 1 on screen: Screen8.1
14:30:55.764 HOD  EventThread_1
  HodConn:m23cacav_HostPubServer#1: 1  Entering text: [enter]
14:30:55.974 HOD  EventThread_1
  HodConn:m23cacav_HostPubServer#1: 1  Registering screen for recognition: Screen9.1
14:30:55.974 HOD  EventThread_1
  HodConn:m23cacav_HostPubServer#1: 1  Executing Action: 0 on screen: Screen9.1
14:30:55.974 HOD  EventThread_1
  HodConn:m23cacav_HostPubServer#1: 1  Recognized screen: Screen9.1
14:31:01.502 HOD  Servlet.Engine.Transports:10
  HodConn:m23cacav_HostPubServer#1: 2  Playing Macro: CicsTran2
14:31:01.512 HOD  Macro_PlayThread
  HodConn:m23cacav_HostPubServer#1: 2  <<< REGISTERING START SCREEN: Screen13.1 >>>
14:31:01.512 HOD  Macro_PlayThread
  HodConn:m23cacav_HostPubServer#1: 2  Registering screen for recognition: Screen13.1
14:31:01.512 HOD  Thread-35
  HodConn:m23cacav_HostPubServer#1: 2  ---Screen: Screen13.1 Matched Description ---
  <description uselogic="1 and 2" >
    <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
    <string value="CREDIT ACCOUNT DEMONSTRATION" row="1" col="1" erow="-1" ecol="-1" casesense="true" optional="false" invertmatch="false" />
  </description>
14:31:01.512 HOD  EventThread_4
  HodConn:m23cacav_HostPubServer#1: 2  Executing Action: 0 on screen: Screen13.1
14:31:01.512 HOD  EventThread_4
  HodConn:m23cacav_HostPubServer#1: 2  Entering text: 10200
```

Chapter 3. Installing and using the WebSphere Host Publisher portlet  95
HodConn:m23cacav_HostPubServer#1: 2 Executing Action: 1 on screen: Screen13.1
14:31:01.712 MOD EventThread_4
HodConn:m23cacav_HostPubServer#1: 2 Executing Action: 2 on screen: Screen13.1
Related publications

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

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 98.

- Building Integration Objects with IBM SecureWay Host Publisher Version 2.1, SG24-5385
- New Advanced Functions in IBM WebSphere Host Publisher V2.2, SG24-5964
- WebSphere Application Servers: Standard and Advanced Editions, SG24-5460
- IBM Host Access Client Package, SG24-6182
- A Comprehensive Guide to IBM WebSphere Host Publisher Version 3.5, SG24-6281
- Access Integration Pattern using IBM WebSphere Portal Server, SG24-6267
- WebSphere V3.5 Handbook, SG24-6161
- WebSphere Scalability: WLM and Clustering Using WebSphere Application Server Advanced Edition, SG24-6153
- IBM WebSphere and VisualAge for Java Database Integration with DB2, Oracle, and SQL Server, SG24-5471
- IBM WebSphere V4.0 Advanced Edition Handbook, SG24-6176

Referenced Web sites

Find frequently asked questions (FAQs), white papers, and additional information at the products Web site:

- IBM Host Publisher on the Internet:
- IBM WebSphere Host Integration Solution:
  http://www.ibm.com/software/webservers/hostintegration
- IBM WebSphere Portal Server for multiplatforms:
- IBM WebSphere Host On-Demand:

These Web sites are also relevant as further information sources:

- The XML zone is a resource for developers on the use of XML and other open standards:
  http://www.ibm.com/developer/xml
- DeveloperWorks Java pages:
http://www.ibm.com/developer/java/

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.
Integrating Host Applications with e-business Portals

Access legacy application content from personalized portals

This Redpaper will help you integrate existing host enterprise applications so that they can be accessed from portlets using the IBM WebSphere Portal Server product. Host application portlet support is provided by the current releases of IBM WebSphere Host On-Demand and IBM WebSphere Host Publisher products.

Extend your Host Publisher applications as new portal content

In this Redpaper, you will find step-by-step examples showing ways to rapidly integrate your host enterprise applications, such as database, 3270, 5250 and VT applications, into an IBM WebSphere Portal Server environment by implementing new and enhanced capabilities incorporated in the current releases of IBM WebSphere Host On-Demand and IBM WebSphere Host Publisher. You will find numerous scenarios describing recommended ways to integrate your legacy applications using the portlet support and sample portlets provided by these products.

Deploy and use Host On-Demand portlets

A basic knowledge of Java technologies such as servlets, JavaBeans, JavaServer Pages (JSPs), as well as HTML and XML markup languages and the terminology used in Web publishing, is assumed.

For more information: ibm.com/redbooks