

IBM(R) Host Access Toolkit Version 7



Getting Started

IBM(R) Host Access Toolkit Version 7



Getting Started

Note

Before using this information and the product it supports, read the information in “Appendix B. Notices” on page 19.

Fourth Edition (September 2002)

This edition applies to the Host Access Toolkit component of IBM® Host On-Demand Version 7.0, and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 2000, 2002. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this book.	v	Chapter 4. Migrating from a previous version of the Host Access Toolkit	9
Chapter 1. Introducing the Host Access Toolkit	1	Chapter 5. Introducing the Host Access Toolkit Application Programming Interfaces 11	
Components of the Host Access Toolkit	1	Introducing the Host Access Class Library	11
What's new in Version 7	2	Introducing the Host Access Beans	12
Where to go for more information	3	Descriptions	12
Redbooks	3	Events	14
Web sites	3	Accessibility Features	14
News groups	4	Introducing Host On-Demand J2EE Connector	15
Chapter 2. Planning for the Host Access Toolkit	5	Appendix A. Installing the iSeries Toolbox for Java	17
Supported operating systems.	5	Appendix B. Notices.	19
Disk space requirement	5	Trademarks	21
Supported browsers.	5		
Java 2 Support	6		
Chapter 3. Installing the Host Access Toolkit 7			

About this book

This book provides information necessary for you to install the Host Access Toolkit. It is intended for use by programmers and system administrators.

Information that is specific to Host On-Demand and Screen Customizer is identified by using the following icons:



This icon is used to denote information unique to Host On-Demand.



This icon is used to denote information unique to Screen Customizer.

Chapter 1. Introducing the Host Access Toolkit

The Host Access Toolkit consists of the Host Access Class Library application programming interface (API), the Host Access Beans for Java™ API, and the Host On-Demand Java 2 Enterprise Edition (J2EE) Connector API.

Programmers can use these APIs to write custom applications to either run with the IBM WebSphere Host On-Demand (hereafter referred to as Host On-Demand) product or run standalone as a custom host access application.

The Host Access Toolkit is available on a separate CD-ROM that is shipped with the Host On-Demand product. This CD-ROM contains all of the Host Access programming API's and documentation necessary for writing Host Access applets and applications.

Components of the Host Access Toolkit

The following APIs make up the Host Access Toolkit:

- The Host Access Class Library (HACL) provides a core set of classes and methods that allow the development of platform-independent applications that can access host information without the need for a graphical display. HACL represents an object-oriented abstraction of a host connection that includes reading and writing the host presentation space, enumerating the fields in the presentation space, reading the operator information area (OIA) for status information, transferring files, and performing asynchronous notification of significant events. For more information about HACL, see the following:
 - “Introducing the Host Access Class Library” on page 11
 - The *Host Access Class Library* reference
- The Host Access Beans provide emulator functions as a set of JavaBeans™ components that developers can use to rapidly develop custom applications that deliver the specific functions they want. For more information about Host Access Beans, see the following:
 - “Introducing the Host Access Beans” on page 12
 - The *Host Access Beans for Java* reference
- The Host On-Demand J2EE Connector is a Java programming interface that conforms to the J2EE Connector Specification, Version 1.0 from Sun Microsystems. Host On-Demand J2EE Connector provides a set of Resource Adapters which communicate to 3270, 5250, CICS, and VT hosts. Users can write applications or servlets by using the APIs provided by Host On-Demand J2EE Connector in WebSphere Studio Application Developer - Integrated Edition (WSAD-IE). These applications and servlets are then

deployed to an application server that supports the J2EE Connector architecture, such as WebSphere Application Server. For more information on the J2EE Connector implementation in Host On-Demand, see the following:

- “Introducing Host On-Demand J2EE Connector” on page 15
- The *Host On-Demand J2EE Connector* reference

What’s new in Version 7

Additional and improved support items of note for the toolkit are:

- Many visual beans in the Host Access Beans API now include accessibility features; see “Accessibility Features” on page 14 for a complete list. All accessibility features require installation of a Java 2 plug-in.

- Support for planes extract

Host On-Demand macros now support extraction of the following host screen information in addition to text:

- color
- field
- extended field
- DBCS
- grid

Host On-Demand stores the screen attributes as planes.

- The Screenable interface has been added to the screen specifications for Host On-Demand 3270, 5250 and Customer Information Control System (CICS) sessions. This new interface provides a standardized record for manipulating data on a screen while supporting the specialized requirements of CICS and other IBM resource adapters.

The Screenable interface is supported along with the older Streamable screen interface. Its advantages over the Streamable interface include the following:

- Screenable records fully support field attributes. Screens do not need to be re-transmitted to send field attributes, improving performance
 - The Screenable interface does not require prior knowledge of the screen’s layout. It supports both generic screen formats (where the sender does not know the screen format) and screen formats based on specific layout information.
 - Screenable records do not need to contain blank space information.
 - The Screenable interface supports a public interface for exchanging data between the screen record and the resource adapter.
- Macro enhancements
Macro functionality has been enhanced to support the following:

- Manipulating variables during macro play
- Conditional (if-else) statements
- Launching macros from within other macros (chaining)
- Launching applications from within macros.
- Displaying numbers in localized formats.
- Copy/append and paste/append functionality.
- The Host On-Demand Java APIs have been extended to support communication through HTTP and Socks proxy servers, and host printing.
- Applet improvements for the Host Access Class Library (HACL):
 - A new interface, *CustomInterface*, provides user applets with access to the Host On-Demand components *HostTerminal*, *Applet instance*, and *Frame*.
 - Parameters can now be passed to applets.

For more detailed information about what's new in each of the APIs, see the following reference publications:

- *Host Access Class Library Reference*
- *Host Access Beans for Java Reference*
- *Host On-Demand J2EE Connector Reference*

Where to go for more information

The publications, Web sites, and news groups listed in this section provide helpful information for using the Host Access Toolkit Java APIs.



You can directly link to most of the resources referenced below; however, if you are viewing the PDF version of this book, you must do so using the browser plugin capability of the Adobe Acrobat Reader

Redbooks

IBM Host Access Client Package Redbook (SG24-6182-00, ISBN 0738424242), available from <http://www.redbooks.ibm.com/abstracts/sg246182.html>.

Web sites

Developers using the Host Access Toolkit APIs can make use of a number of Web resources, including the following:

- The Host On-Demand Web site:
<http://www.software.ibm.com/network/hostondemand/>
- The Personal Communications Web Site:
<http://www.software.ibm.com/network/pcomm/>
- The Sun Microsystems home page for accessing and downloading various Java APIs and tools: <http://java.sun.com/products/>

- The Sun Microsystems J2EE connector Web site:
<http://java.sun.com/j2ee/connector/>
- The IBM Web site *Java Technology Zone*:
<http://www.ibm.com/developerworks/java/>
- The IBM Web site *IBM Developer Kit Porting Web Page*:
<http://www.ibm.com/developerworks/java/jdk/?dwzone=java>

News groups

The following news groups require access to a Usenet newsgroup server and a client news reader:

- `ibm.software.hostondemand`
- `ibm.software.pcomm`

Chapter 2. Planning for the Host Access Toolkit



For the most up-to-date version of the requirements, see the README file and the Host On-Demand Web site.

- “Supported operating systems”
- “Disk space requirement”
- “Supported browsers”
- “Java 2 Support” on page 6

Supported operating systems

The Host Access Toolkit CD-ROM provides an installation utility for the following operating systems:

- Windows[®] 95 or Windows 98
- Windows Me
- Windows NT[®] 4.0 with service pack 3 or higher
- Windows 2000
- Windows XP

Applications or applets developed using the Host Access Toolkit can be ported to other operating systems that support Java; however, installation of the toolkit is not supported on those other systems. The necessary toolkit jar files needed to run your application or applet can be packaged with your application or applet and copied to those other systems according to the terms of your licensing agreement.

Disk space requirement

A typical installation of the Host Access Toolkit requires 110 MB of disk space.

Supported browsers

The Host Access Toolkit supports the following browsers:

- Netscape Navigator 4.6, 4.7, 6.0, 6.2.1
- Netscape Navigator (OS/2) 4.61
- IBM Web Browser for OS/2 V1.2
- Microsoft Internet Explorer 4.01 with SP1, 5.0 or 5.1, 5.5, 6.0

Java 2 Support

The Host Access Beans for Java are now supplied for both JDK 1.1.8 and Java 2 (JDK 1.3.0).

The Java 2 JAR libraries can only be run in a browser that has the Java 2 plug-in enabled or launched in a Java 2 virtual machine. The Java 2 version is Swing-based and should integrate well with other Swing components. There are known issues when building applications or applets that use both AWT components and Swing components. For more information on compatibility between AWT and Swing components, see the Sun Java Web site, java.sun.com.

The Java 2 version of the Host Access Beans offers accessibility, autoIME/on the spot conversion for double-byte character set (DBCS) languages, and print screen enhancements. Pure JDK 1.1 applets and applications that use the Host Access Toolkit JDK 1.1.8 libraries are runtime compatible with the Java 2 Virtual Machine. However, applets and applications built with the Java 2 libraries are strictly incompatible with a JVM version 1.1.

The traditional JDK1.1.8 version of our APIs is most suited for applets intended for JDK1.1-based browsers or for applications launched into a Java Virtual Machine version 1.1. These JDK1.1.8 libraries (JAR and CAB) are compatible with previous versions of the Host Access Toolkit.



The latest version of the Java Virtual Machine runtime environment for your server can be downloaded from the *IBM Developer Kit Porting Web Page*: <http://www.ibm.com/developerworks/java/jdk/?dwzone=java>.

Chapter 3. Installing the Host Access Toolkit

The basic steps for installing the Host Access Toolkit on a Windows system are as follows:

1. Insert the CD-ROM into your CD-ROM drive. If AutoPlay is on, inserting the CD-ROM will automatically invoke the CD-ROM interface program.



If AutoPlay has been turned off, do the following:

- a. Select **Run** from either the **File** pull-down or the task bar.
- b. Enter
E:\setup

in the Command Line box (where *E:* is your CD-ROM drive).

2. If you wish, click on **View Documentation**, **Visit Our Web site**, or **Explore This CD** to learn more about the Host Access Toolkit.
3. When you are ready, click **Install Product**.
4. Follow the directions in the installation windows.

Chapter 4. Migrating from a previous version of the Host Access Toolkit

Previous installations of Host On-Demand do not affect installation of the Host Access Toolkit. Host Access Beans for Java and Host Access Class Library applications and applets written using a previous version of Host On-Demand will work with this version of the Host Access Toolkit.

Chapter 5. Introducing the Host Access Toolkit Application Programming Interfaces

The following sections describe each application programming interface (API) in the Host Access Toolkit:

- “Introducing the Host Access Class Library”
- “Introducing the Host Access Beans” on page 12
- “Introducing Host On-Demand J2EE Connector” on page 15

Introducing the Host Access Class Library

The Host Access Class Library (HACL) for Java provides a set of classes and methods that allow the development of platform independent applications that can access host information at the data stream level. HACL implements the core host access function in a complete class model that is independent of any graphical display and only requires a Java-enabled browser or comparable Java environment to operate. The class library represents an object-oriented abstraction of a host connection that includes reading and writing the host presentation space, enumerating the fields in the presentation space, reading the operator information area (OIA) for status information, transferring files, host printing, and performing asynchronous notification of significant events.

With HACL, application developers can write Java applets that manipulate data from the data stream presentation space (such as 3270, 5250, and Virtual Terminal) without requiring the users to have the applets residing on their machines. The presentation space represents a virtual terminal screen that contains both data and associated attributes presented by host applications. HACL Java applets can open a session to the host, wait for incoming host data, get specific strings from the imaginary screen, get associated attributes of the strings, set new string values, send data stream function keys back to the host, and wait for the next host response. After an interaction is complete, the applet can switch to other tasks or simply close the session. The entire operation can be done without ever showing host screens.



Host On-Demand’s HACL implementation provides the following additional benefits:

- It is platform independent.
- It is downloadable and executable on client workstations using standard Web and Java technology. This provides significant maintenance and resource savings.

Applications or applets written for HACL are portable between Host On-Demand and Personal Communications.



The API books describe functionality for both the Host On-Demand and Personal Communications environments. Functions specific to Personal Communications are flagged. However, these books are not meant to serve as the documentation for Personal Communications support. Please refer to the version of the books supplied with your version of Personal Communications to determine the specific functions supported in that environment.

Introducing the Host Access Beans

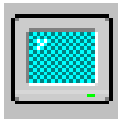
The Host Access Beans for Java provide emulator functions as a set of JavaBeans components. JavaBeans components have configurable properties, use events to communicate, and can be manipulated in visual development environments. (See the Sun JavaBeans API Specification for more information). The Host Access Beans can be used by developers to rapidly develop custom applications that deliver the specific functions they want to include in their host access applications.

Descriptions

Each bean description below is preceded by the icon used to represent the bean in most visual development environments.



Session - This non-visual bean provides methods and properties for setting up and establishing communications with the host system. The Session bean fires events that allow listeners to be notified of presentation space, operator information area (OIA), and communication changes.



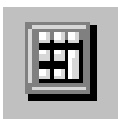
Screen - This visual bean provides the graphical interface for displaying the host data from a Session bean. The Screen bean listens to presentation space, OIA, and GUI events fired by the Session bean and interprets the events to display the main text area and the operator information area. It fires keystroke events to registered listeners, and also provides the clipboard cut, copy, and paste functions.



Terminal - This visual bean combines the Session and Screen beans to provide a composite bean that encompasses both the communication with the host and the graphical interface for displaying the host data.



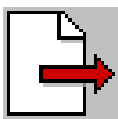
CustomTerminal - This visual bean is installed with Screen Customizer and provides a customizable graphical interface for displaying host data. Only KeyPad, KeyRemap and FileTransfer beans can be wired to this bean.



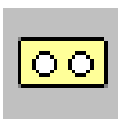
KeyPad - This visual bean provides a simple grid of buttons which invoke various host functions.



KeyRemap - This visual bean provides keyboard remap capability. Using KeyRemap, keystrokes can be mapped to alternate characters or directly to host functions.

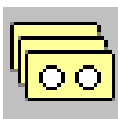


FileTransfer - This visual bean provides a toolbar interface for transferring files to and from a host.



Macro - This non-visual bean records and plays a single macro. Macro employs advanced screen recognition technology to reliably navigate host applications in any environment. Macro also provides the ability to prompt for user input and extract the following information from the screen during playback:

- text
- color
- field positions and their attributes
- extended character attributes
- DBCS attributes for each character and field attribute
- grid



MacroManager - This visual bean provides a toolbar interface for managing multiple macros. The MacroManager bean allows you to record, play, load, delete and edit multiple macros.



Color Remap - This visual bean provides a simple interface for modifying the colors displayed by the Screen or Terminal beans.



HostPrintSession - This non-visual bean extends the Session bean and provides a simple interface for creating and customizing 3270 and 5250 printer sessions.



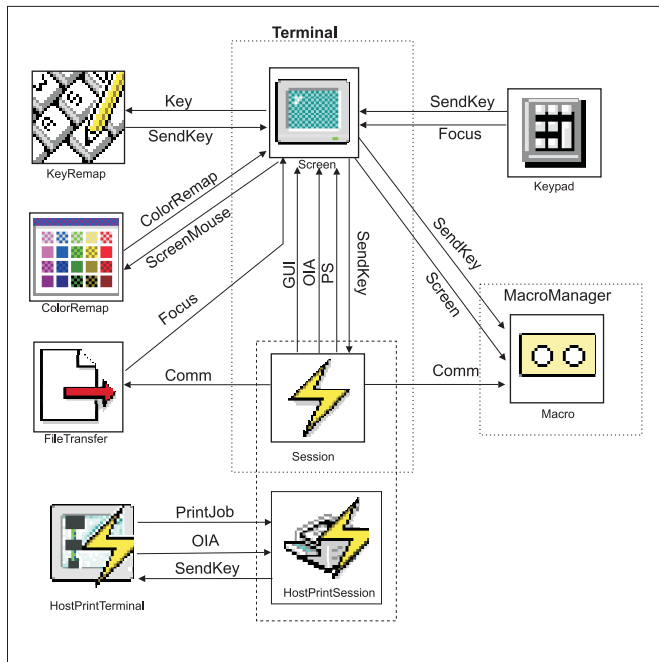
HostPrintTerminal - This visual bean includes the HostPrintSession bean and provides an interface for creating and customizing a 3270 or a 5250 printer session. At run-time, the HostPrintTerminal bean visually displays information about the status of print jobs and the connection with the host.



Converter - The Converter bean performs a codepage-to-codepage conversion. For the Arabic and Hebrew languages, Converter performs certain BIDI-specific transformations, including logical-to-visual transformations, visla-to-logical transformations, and Lam-Alef processing (Arabic only).

Events

The following diagram shows the major events that are passed between each of the Host Access Beans and the composition of the Terminal bean.



The CustomTerminal bean is wired similarly to the Terminal bean; however, only Keypad, KeyRemap, and FileTransfer beans can be wired to the CustomTerminal bean.

The HostPrintSession bean is actually an extension to the Session bean; however, it is shown separately in the diagram to indicate the events that HostPrintTerminal listens for.

Accessibility Features

Host On-Demand offers accessibility features to help users who have physical disabilities (such as restricted mobility, limited or no vision, or limited or no hearing) use host sessions successfully. Features include keyboard equivalents for all actions (mouseless operation), support for display system settings for size, font, and color for user interface controls, and descriptive text for selected graphics.

The following visual Host Access Beans for Java have accessibility features built into their graphical user interfaces:

- Screen

- Terminal
- CustomTerminal
- KeyPad
- KeyRemap
- FileTransfer
- MacroManager
- HostPrintTerminal

To use these accessibility features, you must have the Java 2 plug-in installed. Note that some beans (such as the FileTransfer bean) do not fully support all accessibility features.

Introducing Host On-Demand J2EE Connector

Host On-Demand J2EE Connector is based on version 1.0 of the Sun J2EE Connector Architecture Specification. Before the J2EE Connector architecture was introduced, there was no standard architecture for integrating heterogeneous Enterprise Information Systems (EIS). Host On-Demand users had to use Host Access Class Library (HACL) to access hosts; other vendors provide specific architectures for this purpose. J2EE Connector architecture provides a standard set of services allowing developers to quickly connect and integrate their applications with virtually any backend EIS, and to any application servers conforming to the J2EE Connector architecture.

The Host On-Demand J2EE Connector provides access to 3270, 5250, Customer Information and Control System (CICS), and Virtual Terminal (VT) hosts from the Internet. The Host On-Demand J2EE Connector is a Java programming interface which conforms to the J2EE Connector Specification version 1.0 from Sun Microsystems, and the Common Client Interface (CCI) infrastructure. This interface is intended for use with these products to develop applications in a consistent manner with other connectors.

An application or servlet uses the Host On-Demand J2EE Connector classes for host access over TCP/IP using standard telnet protocols: TN3270, TN5250, CICS or VT emulation. For 3270 and 5250 hosts, a telnet server gateway (such as IBM's Communications Server family of products) is used to translate the incoming telnet stream into an SNA datastream. Host On-Demand supports encryption of the datastream for telnet servers which support Secure Sockets Layer (SSL).

The Host On-Demand J2EE Connector provides a set of Resource Adapters which communicate to 3270, 5250, CICS, and VT hosts. These Resource Adapters (.RAR files) are deployed to a conforming application server; users then write applications or servlets using the application programming

interfaces (APIs) provided by Host On-Demand J2EE Connector in WebSphere Studio Application Developer - Integrated Edition (WSAD-IE). WSAD-IE and its tools are used to write and test user applications and servlets. These applications and servlets are then deployed to an application server that supports the J2EE Connector architecture, such as WebSphere Application Server.

J2EE Connector is just one of the programming environments that Host Access Toolkit users can use to develop customized applications, applets, and servlets. In other environments, where the customer needs the complete functionality of Host On-Demand, the HACL and JavaBeans interface should be used.

For more information on the J2EE Connector architecture, see the Sun J2EE Connector Architecture Web site at <http://java.sun.com/j2ee/connector/>.

Appendix A. Installing the iSeries Toolbox for Java

The iSeries Toolbox for Java is shipped on the Host Access Toolkit CD-ROM. It consists of a set of Java classes that enable you to write client/server applications and applets that work with data residing on your iSeries system. You can also run these applications on the OS/400 Java Virtual Machine (JVM).

The Toolbox uses iSeries servers as access points to the system. Each server runs as a separate job on the iSeries, and each job sends and receives datastreams on a socket connection.

The access classes provide low-level access to the following iSeries resources:

- Databases via a JDBC driver or record-level access
- Integrated File System
- Programs
- Commands
- Data queues
- Print
- Digital certificates
- Jobs
- Message queues
- Users and groups
- User spaces

Graphical programming interfaces are available for:

- Databases (both JDBC and record-level access)
- Command call
- Data queues
- Integrated
- File system
- Jobs
- Message queues
- Print
- Program call
- Users and groups

The following files are located on the Host Access Toolkit CD, **not** the OS/400 CD:

- jt400_all.zip contains jt400.zip, jt400.jar, utilities files, and help and message files
- jt400_doc_en.zip contains the Programmer's Guide in English
- jt400_doc_ja.zip contains the Programmer's Guide in Japanese
- jt400_doc_ko.zip contains the Programmer's Guide in Korean
- jt400_doc_zh.zip contains the Programmer's Guide in Simplified Chinese (PRC)
- jt400_doc_es.zip contains the Programmer's Guide in Spanish
- jt400_doc_zh_TW.zip contains the Programmer's Guide in Traditional Chinese.

To install the iSeries Toolbox for Java on your workstation, unzip the appropriate files. For example, if you want to install the code and the English version of the Programmers Guide, unzip jt400_all.zip and jt400_doc_en.zip .



You must use a utility that supports long filenames.

For additional information on the toolbox, see <http://www.as400.ibm.com/toolbox>.

Appendix B. 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.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

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.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Department T01
Building B062
P.O. Box 12195
Research Triangle Park, NC 27709-2195
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

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.

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.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

- IBM

Lotus[®] and Domino[™] are trademarks or registered trademarks of Lotus Development Corporation in the United States, or other countries, or both.

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

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

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



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.