IMS Web User's Guide

webu-0002-01

Note

Before using this information and the product it supports, be sure to read the information in “Notices” on page v.

IMS Web user’s guide

Contents

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

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

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

Any references in this publication 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 as your own risk.

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
W92/H3
555 Bailey Avenue
P.O. Box 49023
San Jose, CA 95161-9023
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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

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

Trademarks

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

AIX
IBM
IMS
Net.Data
OpenEdition
OS/2
OS/390
RACF
VisualAge

Microsoft and Windows NT are registered trademarks of Microsoft Corporation.

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

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

Chapter 1. Overview of IMS Web

The following topics are described in this section:

IMS Web gives you a way to reuse your existing IMS transactions in the World Wide Web environment by providing a complete end-to-end solution, including runtime libraries and a tool that generates all the code required to build a Web-based application that accesses existing IMS transactions.

IMS Web consists of the following components:

- IMS Web Development

  IMS Web Development includes:

  - IMS Web Studio: a GUI tool that allows you to:
    - Select and download MFS source files for IMS transactions
    - Generate both a CGI program and HTML file for IMS Web-driven applications from the MFS source files
    - Generate source for a Net.Data transaction DLL and macro for Net.Data-driven applications from the MFS source files
  - IMS Web library: a set of library and header files that are required to build the generated code into an executable that runs in the IMS Web Runtime environment

  IMS Web Development supports the following platforms:

  - Windows NT
  - AIX
  - OS/390 (IMS Web library only)
- IMS Web Runtime

  IMS Web Runtime provides a set of functions that are called by the IMS Web Studio generated code.

  The IMS Web Runtime component functions include:

  - Communication to IMS TCP/IP OTMA Connection (IMS TOC) via TCP/IP
  - MFS-like message formatting and platform data conversion

  The IMS Web Runtime component supports the following platforms:

  - Windows NT
  - AIX
  - OS/390

Attention: IMS Web can be used by itself or in conjunction with Net.Data. If you plan to use Net.Data to submit your IMS transactions, the IMS Web Runtime component is a prerequisite for Net.Data.

IMS Web Studio generates files in two directories under the project directory: cgi and dtw. The cgi directory contains all of the files needed for an IMS Web-driven transaction. The dtw directory contains all of the files needed for the IMS Web backend of a Net.Data-driven transaction.

The following elements are generated in the cgi subdirectory:

- C++ classes and source code for a CGI-BIN program that runs an IMS transaction from a workstation and dynamically creates an output HTML form to display the results of that transaction
- An input HTML form for submitting transaction input data and for running the transaction CGI-BIN program

The following elements are generated in the dtw subdirectory:

- C++ classes and code for a transaction DLL that runs an IMS transaction from a workstation
- Input and output HTML for the transaction packaged in a Net.Data macro

IMS Web Development component

The IMS Web Development component (see “Chapter 3. IMS Web Development” on page 9) includes:

- IMS Web Studio
- IMS Web library: These are the C++ header files and library files required to compile and link the generated C++ files for the IMS Web-driven CGI-BIN program or Net.Data-driven transaction DLL on the NT, AIX, or OS/390 platform

IMS Web Studio is a Java application that runs on JDK or JRE 1.1.4 or above and provides the following functions:

- Allows you to browse and download a set of MFS source files for a transaction
- Generates, from the downloaded files, C++, HTML, and Net.Data macro files for the transaction
- Generates .mak files to compile and link the generated C++ files to create an executable CGI-BIN program and transaction DLL

You move the CGI-BIN program and input HTML form, and/or transaction DLL and Net.Data macro, to your Web server to provide access to the IMS transaction.

IMS Web Runtime component

The IMS Web Runtime component (see “Chapter 4. IMS Web Runtime component” on page 31) consists of libraries (which are installed on the Web server and are invoked by the generated CGI-BIN program or transaction DLL), conversion tables for the IMS Web supported languages, and an executable file, HWSCMD.EXE.

The conversion tables are used to convert text data in IMS messages so that it is displayed properly on all supported platforms, in all supported languages. HWSCMD.EXE is the command line executable file that is used to start and stop tracing of activity by the IMS Web Runtime component, as well as display and terminate IMS Web conversational transactions.

For a Windows NT or OS/390 Web server, the runtime component libraries are Dynamic Link Libraries (DLLs); for an AIX Web server, the runtime component libraries are shared libraries (.a files for AIX). These libraries provide the following functions:

- Communication to and from IMS TCP/IP OTMA Connection, which accesses IMS
- Conversion of IMS message data between Web server and host platforms
- MFS-like formatting of messages

Net.Data support

Net.Data is an IBM software product that allows Web developers to build Web applications using Net.Data Web macros. Net.Data provides connectivity to a variety of data sources and applications including relational databases, flat files, registries, system services, and IMS databases through IMS transactions.

Net.Data is an application that runs on an HTTP Web server. You can use it as a CGI application or as a Web server API application. When Net.Data is used as a CGI application, it is invoked as the executable DB2WWW. When Net.Data is used as a Web server API application with NSAPI, ICAPI, or ISAPI, the server calls it as a dynamic linked library (DLL) or shared library.

Restriction: The IMS Web language environment of Net.Data is only supported when Net.Data runs as a CGI application. It is not supported with Net.Data as ICAPI.

Language environments are Net.Data’s interface to data and applications. Each language environment provides a specific interface. For example, Net.Data provides language environments to access DB2 and ODBC databases, REXX, PERL, and C applications, file systems, Web registries, Java applets, and IMS transactions through IMS Web.

The Net.Data Web macro is a file that contains a series of macro language, HTML, and language environment-specific statements. When a Web server receives a URL that refers to Net.Data, the Web server starts an instance of Net.Data and passes essential information to it, including the name of the Web macro that it needs. Net.Data reads and parses through the Web macro and interprets the statements.

When a Net.Data %FUNCTION statement is encountered in the Web macro, it loads the library of the requested language environment and passes language-specific information to the language environment for processing. The language environment processes the information and returns the results to Net.Data.

After all parsing and language environment processing are completed, all that remains is HTML text that any Web browser can interpret. The HTML text is passed back to the Web server and Net.Data terminates. The resulting HTML is passed to the Web browser.

IMS Web Studio generates the Net.Data Web macro and the source code for the associated DLL or shared library used to process an IMS transaction via IMS Web. The Net.Data Web macro includes both the input and output HTML for the IMS transaction, as well as a %FUNCTION statement to invoke the IMS Web language environment, which, in turn, dynamically loads and executes the transaction DLL or shared library.

What’s new in release 2.1.0

- Support of different transaction types:
  - Nonconversational transaction support: same as before Release 210
  - IMS-conversational transaction support: IMS transactions that use the SPA to save state data between iterations of the conversation
- Options to generate HTML to gather host information from a browser user:
  - IMS TOC hostname
  - IMS TOC port
  - Target IMS XCF member name
- Options to generate HTML to gather user information from a browser user:
  - userid
  - groupname
  - password
- Redesigned Studio tool. The IMS Web Studio is now a TaskGuide that leads you through the process of generating IMS Web code, step by step. The tool is implemented in Java™, so it can run on all platforms.
- Support for user-provided exits in IMS Web Runtime and IMS TOC to encrypt and decrypt the password.
- Documentation available in PostScript and PDF formats.

What’s new in release 2.1.1

- Error reporting in the parsing function of the IMS Web Studio has been expanded.

Reader comment form

A reader comment form is included with the hardcopy version of this book. If you would like to fill out the online version of the reader comment form, please link here or go to http://www.software.ibm.com/data/rcf/.

Chapter 2. Prerequisites

The following topics are described in this section:

- “IMS Web Development component”
- “IMS Web Runtime component”
- “Net.Data” on page 6

Following are the prerequisites for enabling IMS transactions on the World Wide Web using IMS Web. If you plan to use Net.Data to submit your IMS transactions, the IMS Web Runtime component is a prerequisite for “Net.Data” on page 6.

IMS Web Development component

The IMS Web Development component consists of the IMS Web Studio tool and the IMS Web library. The IMS Web library is a set of library and header files that are required to build the generated code into an executable that runs in the IMS Web Runtime environment.

Requirement: For OS/390, IMS Web Studio is installed on a Windows NT or AIX workstation and the IMS Web library is installed on an OS/390 machine.

The IMS Web Development component requires one of the following:

- Windows NT Version 4.0 or later, running on an Intel® processor
- AIX 4.1.4 or later
- OS/390 1.3

The machine on which the IMS Web Development component is installed must have:

- For NT, Microsoft Visual C++ 4.2 or later
- For AIX, IBM C Set++ for AIX
- For OS/390, IBM C/C++ Compiler for MVS/ESA

IMS Web Studio requires:

- The JDK or JRE 1.1.4 or later. To download the JDK or JRE, go to the following URL: www.javasoft.com/products/.

IMS Web Runtime component

The IMS Web Runtime component is installed on a Web server machine and requires:

- One of the following:
  - Windows NT Version 4.0 or later, running on an Intel processor
  - AIX 4.1.4 or later
  - OS/390 1.3
- A Web server product. IMS Web has been tested with the following Web server products:
  - IBM’s Internet Connection and Domino Go servers on Windows NT, AIX, and OS/390

- Microsoft Internet Information server on Windows NT
- Netscape FastTrack server on Windows NT

Net.Data

This section includes prerequisites for Net.Data and installation instructions.

Prerequisites

If you plan to use Net.Data as a front end for your IMS transactions, the following component is a prerequisite for Net.Data:

- IMS Web Runtime component installed on your Web server

Installation information for the IMS Web language environment

The first IMS Web language environment (DTWHWS.dll) at the IMS Web 1.2.0 level ran on a Windows NT Web server platform and supported nonconversational transactions. The IMS Web language environment at the IMS Web 2.1.1 level (DTWHWS.dll for Windows NT, dtwhws for OS/390, and dtwhwsshr.o for AIX) has enhancements for conversational processing and runs on Windows NT, OS/390, and AIX Web server platforms.

Restrictions:

- The IMS Web language environment at the IMS Web 1.2.0 level (DTWHWS.dll), requires the IMS Web Runtime component at the 1.2.0 level.
- The IMS Web language environment for Windows NT at the IMS Web 2.1.1 level (DTWHWS.dll), requires the IMS Web Runtime component at the 2.1.1 level and is used with Version 1 of Net.Data for Windows NT.
- The IMS Web language environment for AIX at the IMS Web 2.1.1 level (dtwhwsshr.o), requires the IMS Web Runtime component at the 2.1.1 level and is used with Version 1 of Net.Data for AIX.
- The IMS Web language environment for OS/390 at the IMS Web 2.1.1 level (dtwhws), requires the IMS Web Runtime component at the 2.1.1 level and is used with Version 2 of Net.Data for OS/390.
- The IMS Web language environment of Net.Data is only supported when Net.Data runs as a CGI application. It is not supported with Net.Data as ICAPI.
- Net.Data currently cannot display transaction output message fields greater than 1024 single-byte characters.

If you want to use the 2.1.1 level of the IMS Web language environment before the release of Net.Data that includes it, proceed as follows:

1. IMS Web 2.1.1 is available from the Internet. Go to the IMS Web home page (http://www.software.ibm.com/data/ims/about/imsweb/) and select the Downloads icon.
2. From the IMS Web Download page, select the IMS Web 2.1.1 link.
3. Fill out the survey and click Submit after accepting the license agreement.
4. In the section “Net.Data Language Environment for IMS Web 2.1.1,” select the link to download the language environment for your platform.

For **Windows NT**:

1. Select “Download: Net.Data Language Environment for IMS Web 2.1.1 for Windows NT (DTWHWS.dll)” to download the language environment. Place it in the same directory as the other Net.Data language environments. Typically, this directory is c:\winnt\system32.
2. Check the Net.Data configuration file (db2www.ini) for the following statement and add it if it is missing:

   ```
   ENVIRONMENT  (HWS_LE)   C:\WINNT\System32\DTWHWS.DLL   ( OUT RETURN_CODE )
   ```

For **AIX**:

1. Select “Download: Net.Data Language Environment for IMS Web 2.1.1 for AIX (dtwhwsshr.o)” to download the language environment. Place it in /usr/lpp/internet/db2www/lib.
2. Add the following statement to the Net.Data configuration file (db2www.ini):

   ```
   ENVIRONMENT  (HWS_LE)   /usr/lpp/internet/db2www/lib/dtwhwsshr.o   ( OUT RETURN_CODE )
   ```

For **OS/390**:

1. Select “Download: Net.Data Language Environment for IMS Web 2.1.1 for OS/390 (dtwhws)” to download the language environment.
2. Upload the language environment (dtwhws) to the /usr/lpp/netdata/cgi-bin directory on OS/390. Be sure that you set the transfer mode to binary. Use the ftp port that gives you direct access to the HFS. For example, for Windows NT enter:

   ```
   ftp
   open CSDMEC08 3021
   bin
   cd /usr/lpp/imsweb
   put dtwhws
   bye
   ```

3. Add the following statement to the Net.Data configuration file (db2www.ini):

   ```
   ENVIRONMENT  (HWS_LE)   /usr/lpp/netdata/cgi-bin   ( OUT RETURN_CODE )
   ```

In addition to the previous installation steps, refer to “Creating request templates” on page 33 for potential extra Web server configuration tasks, especially for OS/390.

Chapter 3. IMS Web Development

This section describes the IMS Web Development process and includes the following topics:

The IMS Web Development Component consists of the IMS Web Studio tool and the IMS Web library (a set of library and header files that are required to build the generated code into an executable that runs in the IMS Web Runtime environment). The IMS Web library lets you compile and link the generated C++ project files for the NT, AIX, and OS/390 platforms.

How to install the IMS Web Development component

For Windows NT and AIX:

Requirement: JDK or JRE release 1.1.4 or later is a prerequisite for the IMS Web Development component. You can download the JDK or JRE by linking to the following URL: **www.javasoft.com/products/.**

1. Install JRE Version 1.1.4 (or higher) or JDK Version 1.1.4 (or higher). Ensure that you can run either **jre.exe** or **java.exe** from the command prompt. It is also helpful to add .; to your classpath to ensure that Java searches the current directory.
2. IMS Web 2.1.1 is available from the Internet. Go to the IMS Web home page (http://www.software.ibm.com/data/ims/about/imsweb/) and select the Downloads icon.
3. From the IMS Web download page, select the IMS Web 2.1.1 link.
4. Fill out the survey and click Submit after accepting the license agreement.
5. In the section “IMS Web,” select the link Download: IMS Web Development Component 2.1.1 (hwswd211.class). This link downloads an executable file to your workstation.

6. From the MS-DOS or AIX command line, change to the directory where the image (hwswd211.class) is located and enter:

   ```
   jre -cp . hwswd211
   ```

   Recommendation: If you are using the JDK, enter `java hwswd211`. However, we recommend that you use the JRE.

   Requirement: The class name used with jre.exe or java.exe is case sensitive.

For OS/390:

The IMS Web Development component consists of the IMS Web Studio and the IMS Web library (a set of header and library files that are required to build the generated code into an executable file that runs in the IMS Web Runtime environment). The IMS Web Studio is not available on OS/390, so it must be installed on a Windows NT or AIX workstation. The IMS Web library must be installed separately on an OS/390 machine.

Requirement: JDK or JRE release 1.1.4 or later is a prerequisite for the IMS Web Development component. You can download the JDK or JRE by linking to the following URL: **www.javasoft.com/products/.**

1. On a Windows NT or AIX workstation, install JRE Version 1.1.4 (or higher) or JDK Version 1.1.4 (or higher). Ensure that you can run either **jre.exe** or **java.exe** from the command prompt. It is also helpful to add .; to your class path to ensure that Java searches the current directory.
2. IMS Web 2.1.1 is available from the Internet. Go to the IMS Web home page (http://www.software.ibm.com/data/ims/about/imsweb/) and select the Downloads icon.
3. From the IMS Web download page, select the IMS Web 2.1.1 link.
4. Fill out the survey and click Submit after accepting the license agreement.
5. In the section “IMS Web,” select the link Download: IMS Web Studio 2.1.1 (hwswd211.class). This link downloads an executable file to your workstation.
6. From the MS-DOS command line, change to the directory where the image (hwswd211.class) is located and enter:

   ```
   jre -cp . hwswd211
   ```

   Recommendation: If you are using the JDK, enter `java hwswd211`. However, we recommend that you use the JRE.

   Requirement: The class name used with jre.exe or java.exe is case sensitive.
7. Provide the requested information to InstallShield to install the IMS Web Development component.
8. In the section “IMS Web,” select the link Download: IMS Web Runtime Component 2.1.1, for OS/390 (HWS3B211.exe). This link downloads an executable file to your workstation.

   Requirement: You must download this self-extracting file to a Windows or OS/2 platform, because the program used to unzip the file is not available on AIX.
9. From the MS-DOS command line of your Windows workstation, change to the directory where the file HWS3B211.exe is located and enter:

   ```
   HWS3B211
   ```

   Note: If HWS3B211 fails with the following error message:

```
Usage : OS2 /P /C
```

Try the following:

```
OS2 /P HWS3B211.exe /C
```

The file you downloaded, HWS3B211, will be unzipped into two OS/390 .tar files:

- HWS3D211.tar for the IMS Web library
- HWS3R211.tar for the IMS Web Runtime component

Note: Refer to “How to install the IMS Web Runtime component” on page 31 for instructions on installing the IMS Web Runtime component for OS/390.

In addition, HWS3B211 creates HTML and text versions of these installation instructions and a cumulative list of the changes in the current version of IMS Web.

10. Before beginning this step, you should work with your OS/390 security system administrator to ensure that your user ID has the appropriate authorization to install the IMS Web library on your OS/390 system. Create an imsweb subdirectory in /usr/lpp. You require update authority to the HFS path /usr/lpp/imsweb.
11. Upload HWS3D211.tar to the IMS Web directory on OS/390 (/usr/lpp/imsweb). Be sure that you set the transfer mode to binary and do not use any character conversion because the tar file has been created on an OS/390 machine running Unix System Services. Use the ftp port that gives you direct access to the HFS. For example, for Windows NT enter:

```
ftp
open CSDMEC09 3021
bin
cd /usr/lpp/imsweb
put HWS3D211.tar
bye
```

12. From /usr/lpp/imsweb directory, enter:

```
tar -xf HWS3D211.tar
```

This unpacks the IMS Web library header files and side-decks into the /usr/lpp/imsweb/include and /usr/lpp/imsweb/lib directories, respectively. Again, no character conversion should be used when you unpack HWS3D211 because the characters are already in EBCDIC format.

Attention: For first time users of the C++ compiler under OE, you will probably need to include the following statement in your $HOME/.profile file in order for the compiles specified in your project .mak files to recognize the ″.cpp″ suffix we use for the generated C++ source files in IMS Web projects:

```
export _CXX_CXXSUFFIX=cpp
```

Starting IMS Web Studio

To start IMS Web Studio, do the following:

For Windows NT:

1. Click Start.
2. Select the **Programs** option from the popup menu.

3. Select the **IMS Web Studio V2** option.

The IMS Web Studio window opens, providing a TaskGuide that you use to create a project.

You may navigate through the TaskGuide using Next/Back. Tasks are checked off as you complete them. You may also navigate through the TaskGuide with the items on the left panel that are checked off.

For AIX:

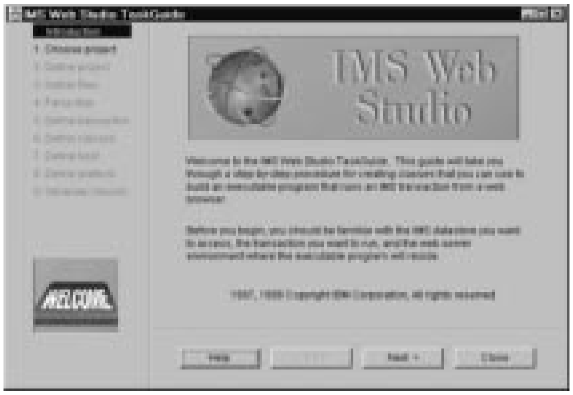
1. Enter the execution file name (hwsgui) at the command line prompt.
2. Ensure that the directory where the IMS Web Studio was installed is added to your path environment variable.

Creating a project in IMS Web Studio

A project is a container for organizing the files required to execute an IMS transaction using IMS Web.

To create a new project, perform the following actions:

1. Select **Choose Project** from the IMS Web Studio tasks on the initial TaskGuide window or click Next. The Choose project TaskGuide window displays.

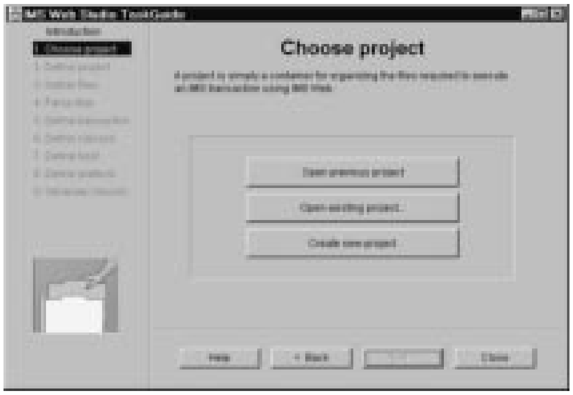


Welcome to the IMS Web Studio TaskGuide. This guide will take you through a step-by-step procedure for creating classes that you can use to build an executable program that runs an IMS transaction from a web browser.

Before you begin, you should be familiar with the IMS datastore you want to access, the transaction you want to run, and the web server environment where the executable program will reside.

1997, 1998 Copyright IBM Corporation, All rights reserved

2. Click Create new project.

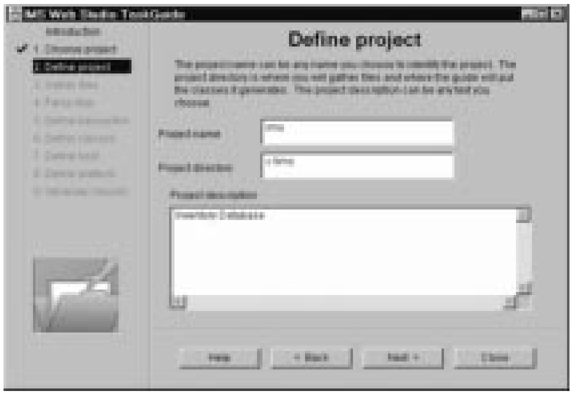


Choose project

A project is simply a container for organizing the files required to execute an IMS transaction using IMS Web.

The Define project TaskGuide window displays.

3. Specify project-related information in the Define project TaskGuide window, as in the following illustration.



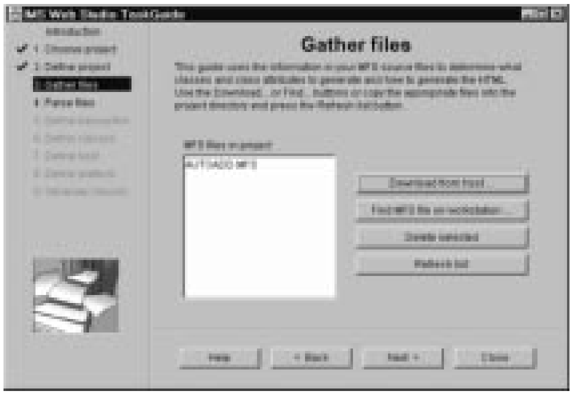
The project name is the name that is used for the generated executables and must be unique for a given Web Server. The name must also comply with the naming convention for executables for your platform and environment. Be aware of any restrictions when selecting a project name. For example, Net.Data does not allow using the hyphen (-) in the name of an executable. Selecting an invalid project name will not be considered an error by IMS Web Studio, but might result in the Web Server being unable to execute the project.

4. Click Next. The Gather files TaskGuide window displays.

Associating MFS source with IMS Web Studio project

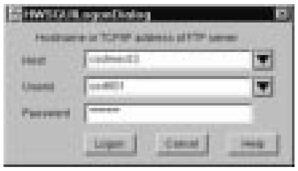
After you have created a project in IMS Web Studio, select and download the MFS source files that are associated with the transaction for which you want to generate code. To select and download these files, do the following:

1. To download files from the host, click Download from host.

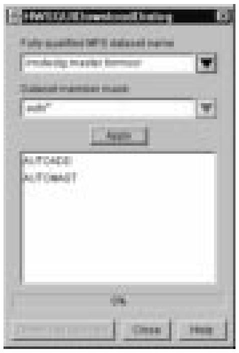


A Logon dialog box displays.

2. Enter the host name of the machine where the MFS source library is located, enter your userid and password, and click Logon.



A download dialog box displays.



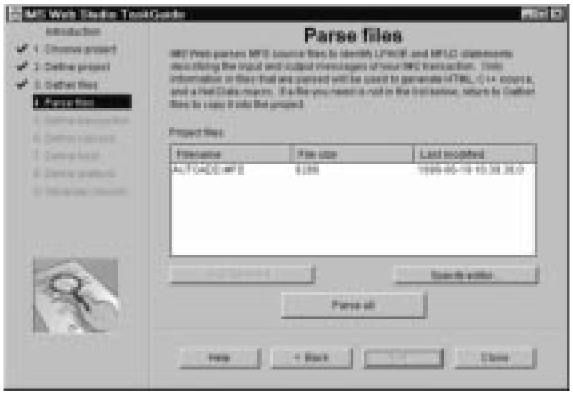
3. Enter a fully qualified data set name in the Fully qualified MFS dataset name field. Do not include quotation marks.
4. If you want to list all members beginning with cdc*, enter a mask in the Dataset member mask field as shown in the example above. Leave this field blank to show all members.
5. Click Apply. Members of your data set display in the list box.
6. Select each member you want included and click Download selected. Then, click Close on the Download window, followed by clicking Next on the Gather files TaskGuide window. The Parse files TaskGuide window displays.

   Notes:

   a. MFS file name should be in uppercase, including the file extension, .*MFS*.
   b. From the Gather files TaskGuide window, you may also click Find MFS file on workstation to move MFS files from another project directory on the workstation.
   c. From the Gather files TaskGuide window, you may also copy MFS files to your project directory (using whatever disk utility you normally use) and click Refresh list to reflect the current contents of the project directory.
   d. From the Gather files TaskGuide window, you may delete MFS files from the project directory by clicking Delete selected.

Parsing files

Once the MFS source files are downloaded, they are parsed to identify the logical page (LPAGE) and message field (MFLD) entries that describe the input and output of your IMS transaction. This parsed information is used to generate HTML, C++ source code, and a Net.Data macro. This task is mandatory and must be completed each time you use the IMS Web Studio TaskGuide.



1. Make the following selections:

   Project files
   You can edit a project file (MFS file) by selecting it in the Project files list box and then by clicking Edit selected. **Note**: You might need to specify an editor the first time you use this window.

   Edit selected
   Edit the file you selected in the Project files list box. If you have not specified a default editor using the Specify editor button, an Open file dialog box displays so that you can find the executable file that starts your editor.

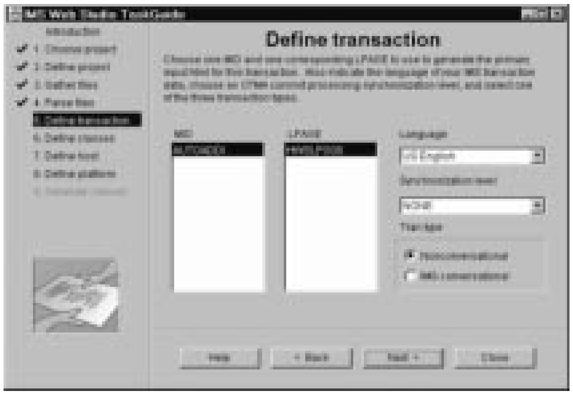
   Specify editor
   If you want to use a particular editor to edit the selected file, click Specify editor. An Open file dialog box displays. You can find the executable file that starts your editor. This editor then becomes the default editor. You need to specify an editor for each installation of IMS Web.

   Parse all
   Click Parse all when you are ready to have the project files parsed. Any syntax errors detected in the MFS source will be reported in an error dialog box. All MFS files in the project directory must be parsed successfully before you can proceed to the next tasks.
2. When you are finished, click Next.

The Define transaction TaskGuide window displays.

A note about TaskGuide items: You can navigate through the TaskGuide using the items in the list on the left of each TaskGuide window that are checked off. As you complete each task, the corresponding item in the list is checked off. Each time you reenter the tool, you must reparse the MFS source. Once the **Parse files** task is complete, it is checked off, as well as any subsequent tasks, if the TaskGuide has sufficient information to complete them without your interaction.

Defining transactions



1. Set up the transaction information needed to generate the primary input HTML.

   MID The MID list box displays the names of all the message input descriptors in the MFS source associated with the project. If multiple MIDs are specified in the MFS source set, select a MID name from the list displayed.

   LPAGE
   The LPAGE list box displays the names of all the input logical page descriptors associated with the MID that is selected in the MID list box. If multiple LPAGEs are specified in the selected MID, select an LPAGE name from the list displayed.

   The MID and LPAGE are used as the primary input page for the transaction. For nonconversational transactions, this input page is the only input page. For conversational transactions, this input page is the first input page.

   IMS Web transactions, whether conversational or nonconversational, are started from an HTML form that is generated from the single MID/LPAGE that is selected in the Define transaction TaskGuide window. This MID/LPAGE typically contains an MFLD statement for the IMS transaction code.

   For IMS conversational transactions, intermediate HTML forms are generated by combining the output fields of each MOD in the set of MFS source files parsed, and the input fields of the MID referenced by its `NXT=` keyword.

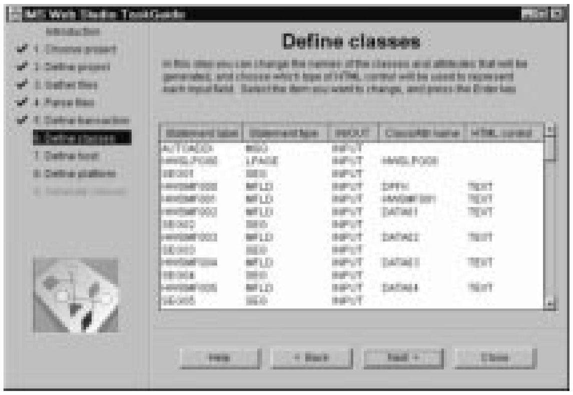
   Language
   You are shown a list of the languages that IMS Web currently supports for IMS transaction data. For these languages, IMS Web converts the data between the host platform and server platform representations. Select the language of your IMS transaction data.

   Synchronization level
   Choose an OTMA commit processing synchronization level from the drop-down list. The default is NONE.

   Tran type
   Check the appropriate box depending on whether or not your transaction is conversational.

2. After making your selections, click Next. The Define classes TaskGuide window displays.

Defining classes



Define classes

In this step you can change the names of the classes and attributes that will be generated, and choose which type of HTML control will be used to represent each input field. Select the item you want to change, and press the Enter key.

Statement label	Statement type	IN/OUT	Class/Attr name	HTML control
AUTOADD	MSG	INPUT		
HWSLPG00	LPAGE	INPUT	HWSLPG00	
SEG01	SEG	INPUT		
HWSMF000	MFLD	INPUT	DFH	TEXT
HWSMF001	MFLD	INPUT	HWSMF001	TEXT
HWSMF002	MFLD	INPUT	DATA01	TEXT
SEG02	SEG	INPUT		
HWSMF003	MFLD	INPUT	DATA02	TEXT
SEG03	SEG	INPUT		
HWSMF004	MFLD	INPUT	DATA03	TEXT
SEG04	SEG	INPUT		
HWSMF005	MFLD	INPUT	DATA04	TEXT
SEG05	SEG	INPUT		

The Define classes TaskGuide window displays the MFS statements that will be used in generating code. Each statement is represented by a single line in the list. Because some of the names in the list are used as class and attribute names in the generated code, the list is presented so you can modify the names.

1. Define the class and attribute names to be generated.

   Statement label
   If the MFS statements have a label, they appear in this column. If they do not have a label, one is generated and appears in this column.

   Statement type
   The type (MSG, LPAGE, SEG, or MFLD) of statement appears in this column.

   IN/OUT
   Input represents an input message, logical page, or message field. Output represents an output message, logical page, or message field.

   Class/Attr name
   After the MFS source files are parsed, the LPAGE and MFLD names in the source, or those that IMS Web defaults, are loaded into the Class/Attr name list box on the Define classes page.

   To edit a name, double-click the name in the Class/Attr Name list box. A dialog box displays so you can enter a new name.

   Restriction: Don’t use names beginning with HWS (HWS is reserved for IMS Web).

2. Click Next. IMS Web checks the names in the Class/Attr Name list box for duplication. Each LPAGE name and each MFLD name within each LPAGE name must be unique. If IMS Web finds a duplicate name, a message box displays, stating the nature of the problem. The second occurrence of the duplicated name is automatically selected for editing. Due to limitations of the list widget being used, the selected item cannot be automatically scrolled into view. Use the scroll bar to find the selected name.

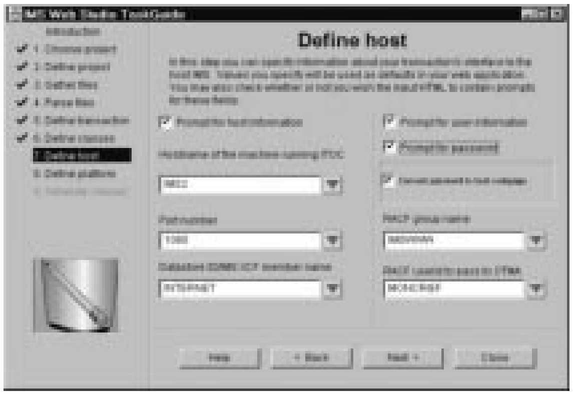
Notes:

1. The names associated with LPAGE statements will become class names in the generated code, so be sure to use only valid C++ class names. Duplicate names will not be allowed, because it does not make sense to have non-unique class names in a single application.
2. The names associated with MFLD statements become member variable (attribute) names within the class named by the preceding LPAGE statement entry in the list, so be sure to use only valid C++ class names. These names must be unique within each class or LPAGE, because it does not make sense to have two member variables with the same name in a single class.

HTML control
: This represents the type of HTML control used for each input field. Currently, we only generate text fields.

The Define host TaskGuide window displays.

Defining a host


Define host
Prompt for host information
Hostname of the machine running ITOC
Port number
Datastore ID/IMS XCF member name
Prompt for password
Help
< Back
Next >
Close

1. Enter the required information as follows:

   Prompt for host information
   If you check this box, the initial input HTML form will be generated with text entry fields (for Hostname, Port number, and Datastore ID/IMS XCF member name), and any values entered into the corresponding Define host TaskGuide window fields will be used as defaults.

   If you do not check this box, you must enter values into the Define host TaskGuide window fields and these values will be hard coded into the C++ source of the generated CGI-BIN program or transaction DLL.

   Hostname of the machine running ITOC
   The TCP/IP host name of the machine running the IMS TCP/IP OTMA Connection. Do not enter the TCP/IP address here. The host name might or might not have to be fully qualified, depending on how TCP/IP is configured for your web server.

   Port number
   The port number (maximum of 4 characters) assigned to the IMS TCP/IP OTMA Connection on the machine running the IMS TCP/IP OTMA Connection

   Datastore ID/IMS XCF member name
   The extended coupling facility member name (maximum of 8

characters). This is the Datastore ID you specified when you installed IMS TCP/IP OTMA Connection.

Prompt for user information

If you check this box, the initial input HTML form will be generated with text entry fields (for RACF group name and RACF user ID), and any values entered into the corresponding Define host TaskGuide window fields will be used as defaults.

If you do not check this box, the values in these fields will be hard coded into the C++ source of the generated CGI-BIN program or transaction DLL. The RACF group name and RACF user ID might be required, depending on the configuration of IMS TCP/IP OTMA Connection and RACF.

IMS TCP/IP OTMA Connection optionally invokes RACF with the group name, user ID, and password. If RACF security checking is successful, the UTOKEN returned by RACF is passed to IMS OTMA.

RACF group name

The RACF group name (maximum of 8 characters) to be passed to IMS OTMA. The RACF group name value entered in the HTML text entry field is converted to uppercase by IMS TCP/IP OTMA Connection before invoking RACF.

RACF userid to pass to OTMA

The RACF user ID (maximum of 8 characters) to be passed to IMS OTMA. The RACF user ID value entered in the HTML text entry field is converted to uppercase by IMS TCP/IP OTMA Connection before invoking RACF.

Prompt for password

If you check the prompt for user information check box, the prompt for password check box is enabled. Otherwise, it is disabled.

If you check the prompt for password box, the initial input HTML form will be generated with a text entry field for password. IMS TCP/IP OTMA Connection passes the password to RACF for authentication of the user ID. The password value entered in the HTML text entry field is converted to uppercase by IMS TCP/IP OTMA Connection following decryption and before invoking RACF.

If the prompt for password check box is not checked, the password field is not generated.

Convert password to host codepage

If you check this box, the resulting encrypted password is converted as text between the Web server platform and the host (IMS TCP/IP OTMA Connection) platform. This is the default. For example, if the server platform is Windows NT, the characters of the encrypted password are converted from ASCII to EBCDIC.

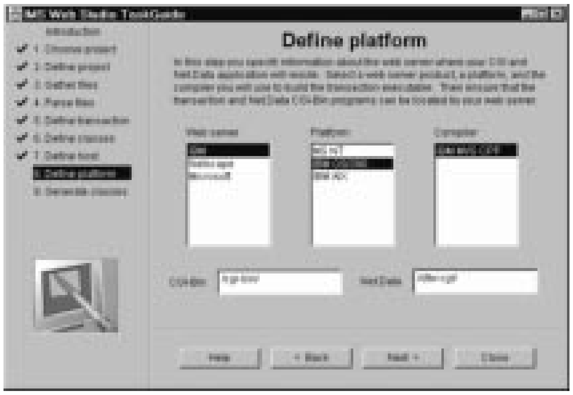
If this box is not checked, the encrypted password is not converted as text between platforms.

Restriction: Do not chose this type of encryption unless you have incorporated our sample or your own method into the IMS Web Runtime component. Otherwise, you will probably fail user authentication by RACF.

See the “Security” section in the *IMS Web Programmer’s Reference* for a detailed description of password encryption methods.

2. Click Next. The Define platform TaskGuide window displays.

Defining platforms



1. In this step, you need to specify information about the Web server where your Net.Data or CGI application will reside. Select a Web server product, a platform, and the compiler you will use to build the transaction executable. Then, ensure that the transaction and Net.Data CGI-BIN programs can be located by your Web server.

   Web server
   Select a Web server product.

   IBM Select IBM if your Web server product is in the family of Web server offerings from IBM/Lotus including:
   - IBM Domino Go Webserver
   - IBM Internet Connection Secure Server
   - IBM Internet Connection Server

   Netscape
   Select Netscape if your Web server product is in the family of Web server offerings from Netscape including:
   - Netscape Enterprise Server
   - Netscape Fastrack Server

   Microsoft
   Select Microsoft if your Web server product is Microsoft’s Internet Information Server.

   Your selection of Web server determines what is displayed in the CGI-BIN and Net.Data fields and the format of the HTML statements that are used to invoke your Web application executables.

Platform

Select the target platform for your Net.Data or CGI application. All supported platforms for the selected server family are displayed.

This information is used in generating the .mak file that is used to build the executable form of your Web application.

MS NT

Microsoft Windows NT Version 4.0 or later

IBM OS/390

OS/390 1.3

IBM AIX

AIX 4.1.4

IBM OS/2

OS/2 Warp 4

Sun Solaris

Sun Solaris 5.5.1

Restriction: For IMS Web 2.1.1, only the Windows NT, AIX, and OS/390 platforms are supported.

Compiler

Select a compiler you will use to build the CGI-BIN or Net.Data transaction DLL. This information is used to tailor the makefile for your compiler. If you use a different compiler, you might need to modify the generated makefile.

MS VC++

Microsoft Visual C++

IBM MVS CPP

IBM C/C++ Compiler for MVS/ESA

IBM CSet++

IBM C Set++ for AIX

Sun CPP

SPARCompiler C++

CGI-BIN

This is the request template that is used to locate the CGI-BIN program for your transaction. *Request template* is known as the *URL prefix* or *alias* by non-IBM Web server platforms.

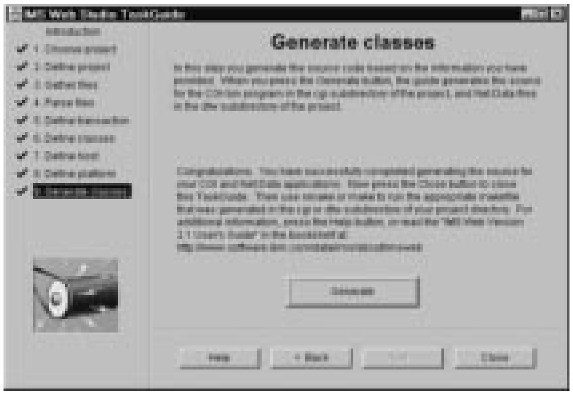
The value you enter in this field is used in the POST statement of the transaction's generated input HTML form. **Recommendation**: If you do not know this value, we strongly recommend that you accept the default. For IBM Web servers on Windows NT, AIX, and OS/390, the default is /cgi-bin/. For Netscape Web servers on Windows NT and AIX, the default is /cgi-bin/. For Microsoft's Web server on Windows NT, the default is /scripts/. Your Web server administrator should create an association between the request template value and the actual path to the CGI-BIN program if one does not already exist. IMS Web Studio changes all backward slashes to forward slashes in any value you enter.

Net.Data

This is the request template that is used to locate the Net.Data CGI-BIN program, db2www. *Request template* is known as the *URL prefix* or *alias* by non-IBM Web server platforms.

The value you enter in this field is used to form the link in the generated HTML that invokes the Net.Data macro. **Recommendation**: If you do not know this value, we strongly recommend that you accept the default. For IBM and Netscape Web servers on Windows NT and AIX, the default is /cgi-bin/. For Microsoft's Web server on Windows NT, the default is /scripts/. For IBM's Web servers on OS/390, the default is /dtw-cgi/. Your Web server administrator should create an association between the request template value and the actual path to the CGI-BIN program if one does not already exist. IMS Web Studio changes all backward slashes to forward slashes in any value you enter.

2. When you are finished, click Next. The Generate classes TaskGuide window displays.



Generate classes

In this step you generate the source code based on the information you have provided. When you press the Generate button, the guide generates the source for the CGI-bin program in the cgi subdirectory of the project, and Net.Data files in the dtw subdirectory of the project.

Congratulations. You have successfully completed generating the source for your CGI and Net.Data applications. Now press the Close button to close this TaskGuide. Then use nmake or make to run the appropriate makefile that was generated in the cgi or dtw subdirectory of your project directory. For additional information, press the Help button, or read the "IMS Web Version 2.1 User's Guide" in the bookshelf at http://www.software.ibm.com/data/ims/applications/web

Generating classes

1. Click Generate to start the code generation.
2. After the classes have finished generating, click Close.

See the following section for a description of the files that are generated.

Restriction: If you want to run the generated code on a platform other than Windows NT, AIX, or OS/390 (for example, HP-UX), you will need to modify the generated makefile before building the executable.

Directory structure and associated files

The following section outlines the directory structure and associated files for IMS Web. IMS Web Studio generates the CGI-BIN associated files and Net.Data files in two subdirectories of the IMS Web Studio project directory:

cgi For files associated with the submission of the IMS transaction using an IMS Web generated CGI-BIN program. The cgi subdirectory contains the .mak file for building the executable form of the CGI-BIN program.

dtw For files associated with the submission of the IMS transaction using Net.Data. The dtw subdirectory contains the .mak file for building the transaction DLL (the DLL or shared library that is dynamically loaded by Net.Data's IMS Web language environment).

The project directory contains the MFS source for the IMS transaction. The output from each .mak is placed in the corresponding subdirectory. The following are the files in each directory, where *proj* represents the IMS Web Studio project name.

- ***proj***

  *proj***.hws**
  The IMS Web Studio project file.

  *proj***.mfs**
  One or more MFS source files associated with the project. The name of the MFS source file *does not* have to be *proj*.

- **cgi**

  CGI*proj***.cpp**
  The C++ source code for the CGI-BIN program that submits the IMS transaction. The CGI-BIN program uses additional classes generated in this subdirectory.

  HTM*proj***.htm**
  The input HTML form used to input data for the IMS transaction.

  HTM*Parse***.cpp**
  The C++ source (implementation file) for the class that parses the input string from the generated input HTML form.

  HTM*Parse***.hpp**
  The interface file for HTM*proj*.cpp.

  *lpageIn***.cpp**
  The C++ source (implementation file) for the class that represents the input logical page of the input message. This class includes a method that is used to submit the request (input message) to IMS and return one of the generated classes for the output logical pages.

  *lpageIn***.hpp**
  The interface file for *lpageIn*.cpp.

  *lpageOut***.cpp**
  The C++ source (implementation file) for an output logical page of an output message. A class is generated for each output logical page. This class includes a method that builds a buffer of HTML that is returned to the Web client.

  *lpageOut***.hpp**
  The interface file for *lpageOut*.cpp.

  etc. A nonconversational transaction can have cpp/hpp files for multiple output logical pages. A conversational transaction can have cpp/hpp files for multiple input and output logical pages.

  CGI*proj***.mak**
  The makefile used to compile and link the source code, CGI*proj*.cpp, into an executable CGI-BIN program on the target platform.

- **dtw**

  DTW*proj***.cpp**
  The C++ source code for the DLL or shared library that submits the IMS transaction. The transaction is invoked by Net.Data’s IMS Web language environment.

  DTW*proj***.def (for Windows NT), DTW***proj***.exp (for AIX)**
  The module definition file used in building the transaction DLL.

DTW*proj***.mac**
The Net.Data Web macro that contains the input and output HTML for the IMS transaction.

DTW*proj***.htm**
A sample link that invokes Net.Data as a CGI-BIN program and references the generated Web macro and input HTML form.

*lpageIn***.cpp**
The C++ source (implementation file) for the class that represents the input logical page of the input message. This class includes a method that is used to submit the request (input message) to IMS and return one of the generated classes for the output logical pages.

*lpageIn***.hpp**
The interface file for *lpageIn*.cpp.

*lpageOut***.cpp**
The C++ source (implementation file) for an output logical page of an output message. A class is generated for each output logical page.

*lpageOut***.hpp**
The interface file for *lpageOut*.cpp.

etc.
A nonconversational transaction can have cpp/hpp files for multiple output logical pages. A conversational transaction can have cpp/hpp files for multiple input and output logical pages

DTW*proj***.mak**
The makefile used to compile and link the source code, DTW*proj*.cpp, into a DLL or shared library on the target platform.

Building an executable file

IMS Web generates two .mak files, one that has the same name as the CGI-BIN program and one that has the same name as the Net.Data macro. For example, to build the CGI-BIN executable file cgi*project_name*.exe, you run the generated .mak file cgi*project_name*.mak. To build the Net.Data transaction DLL dtw*project_name*.dll, you run the generated .mak file dtw*project_name*.mak.

In order to compile and link all the generated code for the project, you run the appropriate utility against the generated .mak file. For Windows NT, the utility is NMAKE; for AIX and OS/390, it is MAKE. You run the utility on the platform for which you generated the source. It is most convenient to use the IMS Web Studio tool on your target platform, so you do not have to move the generated source before building the CGI executables. However, this is not required. For instance, from the IMS Web Studio installed on Windows NT, you could generate code targeted to run on AIX. For OS/390, you *must* use the IMS Web Studio tool on a platform other than OS/390 and generate code targeted for OS/390.

The following macro variables are used with the generated .mak file:

- CFG

  Specify `CFG=HWSDbg` or `CFG=HWSRel` to indicate whether or not you want your executable file built with debugging information. The default is `CFG=HWSRel`.
- HWSPATH

  The value provided for HWSPATH should be the path to where the IMS Web Development Component, containing the header and library files, is installed on the machine running the make utility.

HWSPATH is required for Windows NT and AIX. It is not used when building OS/390 executables.

Examples:

For Windows NT:

nmake /F CGI*project_name***.mak CFG=HWSDbg HWSPATH=i:\IMSWeb211**

Builds the executable CGI-BIN program, CGI*project_name*.exe, with debugging information, on a Windows NT machine that has the IMS Web Development component installed in i:\IMSWeb211.

For AIX:

make -f DTW*project_name***.mak HWSPATH=/home/imsweb**

Builds the transaction shared library, DTW*project_name*shr.o, for use with Net.Data, without debugging information, on an AIX machine that has the IMS Web Development component installed in /home/imsweb.

For OS/390:

make -f CGI*project_name***.mak**

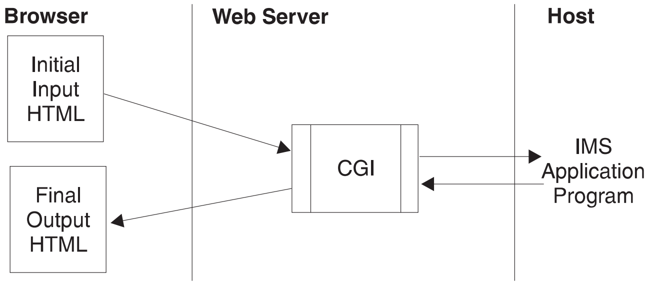
Builds the executable CGI-BIN program, CGI*project_name*, without debugging information, on an OS/390 machine that has the IMS Web header and library files installed in /usr/lpp/imsweb/include and /usr/lpp/imsweb/lib, respectively.

IMS Web transactions

IMS Web supports two types of transactions: nonconversational and IMS conversational. This section describes the two types of transactions.

Nonconversational transactions

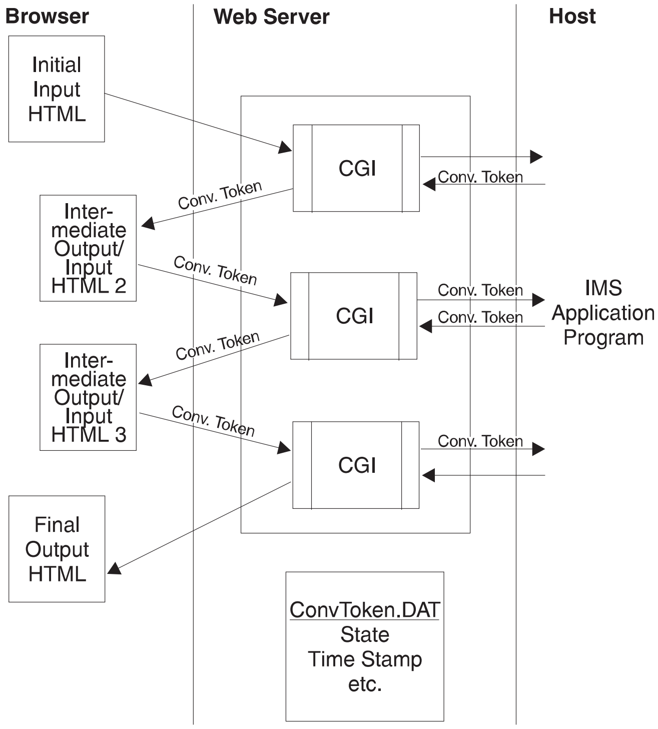
Nonconversational transactions are the simplest type of transaction supported by IMS Web. A single input to an IMS application program is followed by a single output. The following diagram illustrates how IMS Web supports this type of transaction.



- The transaction is started by invoking the initial input HTML form. This form is generated based on the MFS MID chosen in the IMS Web Studio.
- When the CGI-BIN program associated with the transaction receives the output from the IMS application program, it returns the final output HTML form.

IMS conversational transactions

The IMS conversational transaction allows the end user to have a continuing dialogue with an IMS application program. The IMS application program uses the SPA to save state data for subsequent iterations of the conversation. The following diagram illustrates how IMS Web supports this type of transaction.



- The conversation is started by invoking the initial input HTML form. This form is generated from an MFS MID chosen in the IMS Web Studio.
- When the CGI-BIN program associated with the transaction receives the output from the IMS application program, it saves state information about the conversation in a conversation state file, ConvTokn.DAT. In addition, it returns the Intermediate Output/Input HTML 2 form, along with conversation state data in hidden HTML fields of the form.

  The Intermediate Output/Input HTML 2 form, and all intermediate HTML forms for a conversational transaction, are generated by combining the output and input fields of each MOD and the MID referenced by its `NXT=` keyword.
- When the Intermediate Output/Input HTML 2 form triggers the CGI-BIN program again, the CGI-BIN program gets ConvTokn from the hidden HTML fields and reads the ConvTokn.DAT conversation state file to validate the current state of the conversation.
- Iterations of the conversation continue. Each execution of the CGI-BIN program updates the conversation state file with new information about the state of the conversation.
- When the CGI-BIN program receives a message from the IMS application program indicating the end of the conversation, it deletes the ConvTokn.DAT file and returns the Final Output HTML form. The Final Output HTML form is an output only HTML form, containing the contents of the final message from the IMS application program.

- Intermediate HTML forms for a conversational transaction contain an additional button that can be used to end a conversation. Selecting this button causes the CGI-BIN program to erase the conversation state file, ConvTokn.DAT, and send a message to the host to end the conversation. An indication that a message was sent to the host is displayed at the browser.

IMS Web protocol for conversations

As mentioned previously in this section, state information about a conversation is saved in hidden fields of intermediate HTML forms of the conversation. For this reason, the input data for successive iterations of a conversation must be provided in the appropriate intermediate HTML forms. You **may not** use the BACK function to select an HTML form for use as input to a conversation in progress. If you do, a protocol error message will be presented at the browser and the conversation will be terminated.

There are a number of situations where an active IMS Web conversation can be *orphaned*. An orphaned conversation is still active from the point of view of IMS Web and IMS, but it can no longer be continued or terminated using the browser that initiated it. An orphaned conversation can be terminated using the IMS Web command, `HWSCMD EXIT`.

Recommendation: Orphaned conversations can occur when a browser session is closed while an IMS Web conversation is in progress in the browser session. In addition, we **strongly** recommend that each concurrent IMS Web-initiated conversation be conducted in a separate browser session. Multiple IMS Web conversations in a single browser session involve the BACK function, and can very easily result in orphaned conversations or protocol errors.

IMS conversational transactions

This section describes how to display and terminate conversational transactions.

Displaying conversational transactions: The IMS command, `/DISPLAY CONVERSATION BUSY` is typically used to display currently active conversations. Using this command will display both conversations initiated by IMS Web and those initiated from an IMS terminal. For conversations initiated by IMS Web, `/DISPLAY CONVERSATION BUSY` output is:

TERMINAL
The OTMA TPIPE name, which is specified to the IMS Web Studio tool as *Datastore ID/IMS XCF member name*.

USER The userid associated with the IMS Web-initiated conversational transaction and specified to the IMS Web Studio tool as *RACF userid to pass to OTMA*.

nnnn An identifier of the conversation as known by IMS. This identifier represents the *host end* of the IMS Web-initiated conversation.

IMS Web-initiated conversations can be displayed using the IMS Web Runtime command, `HWSCMD DISPLAY CONV`. Using this command displays only IMS Web-initiated conversations; using this command does not display conversations initiated from an IMS terminal. The output for this command is:

Conversation token (Conv.Token)
The IMS Web identifier for the conversation. This identifier is formed by converting the second half of the ServerToken field in the OTMA message into an 8-character identifier. ServerToken is OTMA's identification of the

conversation. The first half of ServerToken is a STCK value; the second half is the STCK value converted into a readable date/time value. The conversation token is used as the filename of the conversation state file, ConvTokn.DAT.

RUName
Request Unit Name. RUName is a unique name that identifies the request the IMS Web program makes to execute the IMS transaction. IMS uses this name as the name of the LTERM that is associated with the transaction.

Next Input Lpage
This is the value of the `NXT=` keyword of the message output descriptor (MOD) that was used to format the output of the current iteration of the conversation.

Date/Time Stamp
This is the date and time the current iteration of the conversation was presented to the customer. The date and time is updated by IMS Web Runtime, for each iteration of a conversation, when an output message is received from the IMS application program. The output message is then presented to the customer in an HTML form.

Age (min.)
The length of time since the latest iteration of the conversation was presented to the customer. It is the difference between the current date and time and the date/time stamp, in minutes.

IMS Name
The XCF member name for IMS OTMA. The output of the IMS command, `/DISPLAY CONVERSATION BUSY` displays this name in the TERMINAL field for IMS Web-initiated conversations.

Host Name
The TCP/IP host name of the machine on which the IMS TCP/IP OTMA Connection is installed.

Terminating conversations: You cannot use the IMS command, `/EXIT` to terminate conversations initiated by IMS Web. IMS Web-initiated conversations are terminated either by the IMS application program, or by selecting the TERMINATE button on the IMS Web HTML form. You can also terminate IMS Web conversations by using the IMS Web Runtime command, `HWSCMD EXIT CONV`; however, reserve this method for unusual situations.

Notes for Version 1 users

- Using existing projects:
  - You can create a new project using the Version 2 Studio tool in an existing Version 1 project directory. The MFS source in the existing project will be recognized by the Version 2 TaskGuide.
  - You can create a new project in a separate directory and copy the MFS files from the existing project into the new project directory. Use the Refresh list button on the Gather files TaskGuide window to reflect the current contents of the project directory.
  - The Version 2 parser is more strict regarding MFS syntax. So, it is possible that MFS files that were parsed successfully by the Version 1 tool will fail with the Version 2 syntax checker.
- Generating on your target platform

  - In Version 1, the Studio tool was only available on Windows NT. As a result, CGI and Net.Data source generated for other platforms had to be moved from NT to the target platform before it could be built. The Version 2 Studio tool will become available on all supported target platforms as the Java Version 1.14 virtual machines become available on these platforms. This will allow you to generate the source on the same machine where it will be built, simplifying the overall process.
- Redesigned Studio tool
  - The addition of conversational transaction support, and the desire to provide the Studio tool on multiple platforms, necessitated a complete overhaul of the Studio tool. While most of the existing concepts remain the same, new features have been added to the tool. In addition, some fields have been renamed as a result of customer feedback. Please thoroughly read this user guide to understand how to use Version 2.

Java virtual machine considerations

Attention: Be sure to read this section before you change your Java virtual machine (JVM).

When you install the IMS Web Studio Version 2, a launch script is generated that you use to start the Studio tool, and an uninstall script is generated that can be used to remove the Studio tool from your workstation. These scripts contain calls that are dependent on the location of the JVM used to run the installation. If you remove or replace the JVM on your workstation, these scripts will probably no longer function. To avoid this situation, uninstall the Studio tool before changing your JVM.

If you have already changed the JVM, and are reading this because the launch and uninstall scripts no longer work, reinstall the Studio tool using the new JVM to update the launch and uninstall scripts.

Chapter 4. IMS Web Runtime component

This section describes the IMS Web Runtime component and the following topics:

- "How to install the IMS Web Runtime component"
- "Creating request templates" on page 33
- "Incorporating project files into your Web server environment" on page 34
- "IMS Web Runtime commands" on page 35
- "IMS Web Runtime data conversion" on page 39

The IMS Web Runtime component is installed on your Web server machines and enables you to run project executable files on the Windows NT, AIX, or OS/390 platforms. The Runtime component consists of:

- The IMS Runtime DLLs (or shared files for AIX) required to run an IMS Web CGI-BIN or Net.Data transaction DLL
- The conversion tables for IMS Web supported languages
- An executable file (hwscmd.exe) that is used to execute IMS Web commands on the Web server machine.

How to install the IMS Web Runtime component

For a Windows NT Web server platform:

1. IMS Web 2.1.1 is available from the Internet. Go to the IMS Web home page (http://www.software.ibm.com/data/ims/about/imsweb) and select the Downloads icon.
2. From the IMS Web Download page, select the IMS Web 2.1.1 link.
3. Fill out the survey and click Submit.
4. In the section "IMS Web," select the link Download: IMS Web Runtime Component 2.1.1, for Windows NT (HWSWR211.exe). This link downloads an executable file to your workstation.
5. Using the Save As dialog box, save the downloaded file to a temporary directory.
6. From the MS-DOS command line of your Windows NT workstation, change to the temporary directory where HWSWR211.exe is located and enter:
   ```
   HWSWR211
   ```
7. Provide the requested information to InstallShield to install the IMS Web Runtime component.

For an AIX Web server platform:

1. IMS Web 2.1.1 is available from the Internet. Go to the IMS Web home page (http://www.software.ibm.com/data/ims/about/imsweb) and select the Downloads icon.
2. From the IMS Web Download page, select the IMS Web 2.1.1 link.
3. Fill out the survey and click Submit.
4. In the section "IMS Web," select the link Download: IMS Web Runtime Component 2.1.1, for AIX (HWSXR211.img). This link downloads an image file to your workstation.
5. Ensure that you are the root user, and then perform the following actions:

a. Click the `smit` application icon to bring up the System Management Interface Tool.

b. Select the **Software Installation and Maintenance** option.

c. Select the **Install and Update Software** option.

d. Select the **Install and Update Software by Package Name** option.

e. In the Install and Update Software by Package Name dialog box that displays, specify the directory where you downloaded the image file. Select the OK push button.

f. When the multi-select list displays, select the OK push button.

g. When the "Are you sure?" dialog box displays, select the OK push button to start the installation.

Note: These instructions apply to using the `smit` GUI tool (System Management Interface). You can use the command-line interface by typing `smit` at the console.

Attention: If you run uninstall on the IMS Web Runtime component on AIX, all files in the directory tree will be removed, even files that were not put there by the install process. If you decide to place your Net.Data transaction programs (shared library files) under the IMS Web directory, please be aware that they will be removed along with other files in that directory tree.

For an OS/390 Web server platform:

See "Considerations when using ICS for OS/390" on page 41.

1. Before beginning this procedure, you should work with your OS/390 security system administrator to ensure that your user ID has the appropriate authorization to install the IMS Web Runtime DLLs and the HWSCMD utility on your OS/390 system. Create an imsweb subdirectory in /usr/lpp, if you have not previously done so. You require update authority to the HFS path for IMS Web (/usr/lpp/imsweb) and for the Web server (/usr/lpp/internet).
2. IMS Web 2.1.1 is available from the Internet. Go to the IMS Web home page (http://www.software.ibm.com/data/ims/about/imsweb) and select the Downloads icon.
3. From the IMS Web Download page, select the IMS Web 2.1.1 link.
4. Fill out the survey and click Submit.
5. In the section "IMS Web," select the link Download: IMS Web Web Runtime Component 2.1.1, for OS/390 (HWS3B211.exe). This link downloads an executable file to your workstation. We recommend creating a new directory for the download.

   Note: You *must* download this self-extracting file to a Windows or OS/2 platform because the program used to unzip the file is not available on AIX.
6. From the MS-DOS command line of your Windows workstation, change to the directory where the file HWS3B211.exe is located and enter:

   ```
   HWS3B211
   ```

   Note: If HWS3B211 fails with the following error message:

   ```
   Usage : OS2 /P /C
   ```

   Try the following:

   ```
   OS2 /P HWS3B211.exe /C
   ```

The file you downloaded, HWS3B211, will be unzipped into two OS/390 .tar files:

- HWS3R211.tar for the IMS Web Runtime component
- HWS3D211.tar for the IMS Web Library

  Note: Refer to “How to install the IMS Web Development component” on page 9 for instructions on installing the IMS Web library for OS/390.

In addition, HWS3B211 creates HTML and text versions of these installation instructions and a cumulative list of the changes in the current version of IMS Web.

7. Upload HWS3R211.tar to the IMS Web directory on OS/390 (/usr/lpp/imsweb). Be sure that you set the transfer mode to binary and do not use any character conversion because the tar file has been created on an OS/390 machine running Unix System Services. Use the ftp port that gives you direct access to the HFS. For example, for Windows NT enter:

   ```
   ftp
   open CSDMEC08 3021
   bin
   cd /usr/lpp/imsweb
   put HWS3R211.tar
   bye
   ```

8. From the /usr/lpp/imsweb directory enter:

   ```
   tar -xf HWS3R211.tar
   ```

   This unpacks the IMS Web Runtime DLLs into the /usr/lpp/imsweb/lib directory, and unpacks the hwscmd executable file into the /usr/lpp/imsweb/bin directory. In addition, the conversion tables for the IMS Web supported languages are unpacked into the /usr/lpp/imsweb directory. Again, no character conversion should be used when you unpack HWS3R211 because the characters are already in EBCDIC format.

Creating request templates

A request template (known as an URL prefix or alias for non-IBM Web server products) is a rule your Web server uses when mapping URL requests to a file. Two of the files generated by the IMS Web Studio contain URLs:

- For CGI-BIN applications, the HTML form used to input transaction data contains a POST statement.
- For Net.Data applications, the HTML used to invoke the Net.Data macro contains a link.

By default, IMS Web Studio uses the following request templates for the URL in CGI-BIN applications:

- /cgi-bin/

  For IBM Web servers on OS/390, AIX, and Windows NT and for Netscape Web servers on AIX and Window NT
- /scripts/

  For Microsoft’s Web server on Windows NT

By default, IMS Web Studio uses the following request templates for the URL in Net.Data applications:

- /cgi-bin/

For IBM and Netscape Web servers on AIX and Windows NT

- /dtw-cgi/

  For IBM's Web servers on OS/390
- /scripts/

  For Microsoft's Web server on Windows NT

If your Web server administrator has created request templates specific to your Web server's environment, you can override the default values in the IMS Web Studio TaskGuide Define platform window fields, **CGI-Bin** and **Net.Data**, so that your request templates are used in the generated code.

Incorporating project files into your Web server environment

The IMS Web Studio generates C++ source code and associated files in your IMS Web project. The code is generated for a particular target Web Server product and platform. After you create an executable file for the target, you must incorporate it, and one or more associated files, into your Web server environment as follows:

For IMS Web only

1. Copy the CGI-Bin executable and the generated HTML file to your Web server.
   - Place the CGI-Bin executable file in the Web server's CGI-Bin directory. This directory is the one in which your Web server looks for CGI-Bin programs.

     For example, in the case of IBM's Web server for Windows NT, copy CGI*proj*.exe to your Web server's CGI-Bin directory.

     For example, in the case of IBM's Web server for AIX, copy CGI*proj* to /usr/lpp/internet/server_root/cgi-bin.
   - Place the HTML file in your Web server's HTML directory.

     For example, in the case of IBM's Web server for Windows NT, copy HTM*proj*.htm to your Web server's HTML directory.

     For example, in the case of IBM's Web server for AIX, copy HTM*proj*.htm to /usr/lpp/internet/server_root/Docs.

   Note: Your Web server administrator might have created a request template (URL prefix or alias for non-IBM Web server products) for the CGI-BIN and/or HTML directories.
2. Optionally, you may add a link to the input HTML form in the appropriate HTML file(s) in your Web server HTML file tree. You can then use the link to display the input HTML form for the transaction on a Web browser.
   - Use this link or enter the URL directly in your browser to display the HTML form on your Web server.
   - Fill in the transaction input and select the SUBMIT push button on the form to submit the transaction to IMS and receive its output at the Web browser.

For Net.Data

1. Copy the transaction DLL or shared library, Net.Data macro, and (optionally) the HTML link file to your Web server.
   - Place the transaction DLL or shared library in a directory from which the operating system retrieves DLLs or shared libraries.

For example, in the case of Windows NT, copy DTW*proj*.dll to C:\WINNT\SYSTEM32. You could also update the path environment variable to point to a directory of your choice, then place the DLL in that directory.

In the case of AIX, copy DTW*proj*shr.o to /usr/lpp/internet/imsweb/lib.

- Place the Net.Data macro in a directory from which Net.Data retrieves Web macros. (See the MACRO_PATH statement in “Configuring Net.Data” in the *Net.Data Administration and Programming Guide*.)

  For example, in the case of Windows NT, copy DTW*proj*.mac to the Macro directory.

  In the case of AIX, copy DTW*proj*.mac to /usr/lpp/internet/db2www/macro.
- Optionally, place the HTML link file in your Web server’s HTML directory.

  For example, in the case of IBM’s Web server for Windows NT, copy DTW*proj*.htm to your Web server’s HTML directory.

  For example, in the case of IBM’s Web server for AIX, copy DTW*proj*.htm to /usr/lpp/internet/server_root/Docs.

Note: Your Web server administrator might have created a request template (URL prefix or alias for non-IBM Web server products) for the HTML directory.

2. Optionally, you may use the link in the sample file, DTW*proj*.htm, to modify an HTML file in your Web server’s HTML tree. You can then use the link to invoke Net.Data and display the input HTML form for the transaction on a Web browser.
   - Use this link or enter the URL directly in your browser to display the HTML form on your workstation.
   - Fill in the transaction input and select the SUBMIT push button on the form in order to submit the transaction to IMS and receive its output at the Web browser.

IMS Web Runtime commands

IMS Web provides a number of commands to help you control IMS Web conversations as well as diagnose problems in the Runtime component. The command processor executable, HWSCMD, is invoked with appropriate parameters, from any command prompt on the Web server. The parameters you specify to the command processor allow you to:

- Display currently active IMS Web conversations
- Terminate IMS Web conversations
- Trace the execution flow of the Runtime component

Note: The syntax notation used for IMS commands is also used for IMS Web Runtime commands.

HWSCMD TRACE command

This command is used for tracing the execution of a CGI-BIN program or transaction DLL through the IMS Web Runtime component. This trace is primarily for diagnostic use by IBM service personnel.

A temporary trace file, *program_name*.tmp, records sequential executions of a CGI-BIN program or transaction DLL. *Program_name* is the name of the CGI-BIN program or Net.Data transaction DLL during tracing. When the trace is turned off, a final entry is made in the .tmp file and the file is renamed to *program_name*.log. These files can be found in the trace subdirectory of the IMS Web Runtime installation directory for Windows NT and OS/2, in /var/tmp/imsweb/trace/ for AIX and OS/390, and in /opt/imsweb/trace/ for Sun.

The IMS Web Runtime component trace is intended primarily for diagnosis. For this reason, only one instance of a particular CGI-BIN program should be traced at a time. Trace data consists of the following:

- An entry that indicates the start of tracing for the CGI-BIN program or transaction DLL
- Date/time-stamped entries that represent the flow through the IMS Web Runtime component for each sequential execution of the CGI-BIN program or transaction DLL
- An entry that indicates the end of tracing for the CGI-BIN program or transaction DLL

Trace data is appended to an existing *program_name*.tmp file. If no *program_name*.tmp file exists, a new file is created containing the trace data.

Command syntax

```
►►──────HWSCMD──TRACE──┬───────┬──prog_name ─────►◄
                       ├─ON────┤
                       └─OFF───┘
```

HWSCMD Is the name of the command processor executable.

Note: For AIX, Sun Solaris, and OS/390 OpenEdition Web server platforms, the command processor executable (hwscmd) must be entered in lower case.

Parameters

TRACE Specifies the trace function of hwscmd.

ON Starts the IMS Web Runtime component trace for the specified CGI-BIN program or Net.Data transaction DLL. The trace remains on for all future executions of the CGI-BIN program or Net.Data transaction DLL, until it is turned off.

OFF Ends the IMS Web Runtime component trace for the specified CGI-BIN program or Net.Data transaction DLL. Ending the trace makes the trace log file available.

program_name
Specifies the name of the CGI-BIN program or Net.Data transaction DLL being traced. *program_name* is generated by IMS Web Studio.

Requirement: For AIX, Sun, and OS/390 platforms, you must enter the `program_name` in the same case as was generated by IMS Web Studio.

Attention: Before you can run `hwscmd trace on`, you must have obtained full access permissions for the directory where the temporary trace file and the trace log file are stored. The CGI-BIN program or Net.Data transaction DLL must also possess write permission for the directory.

HWSCMD DISPLAY CONV command

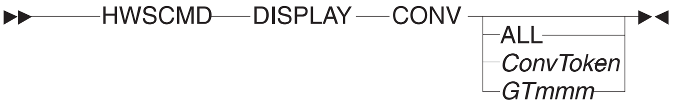
This command displays the information in one or more conversation state files on the Web server. See "IMS Web transactions" on page 25 for a description of IMS Web's support of conversational processing and a description of conversation state files. These files represent conversations initiated by IMS Web from the Web server, and are located:

- In the hwsdata subdirectory of the IMS Web Runtime installation directory for Windows NT and OS/2
- In the /var/tmp/imsweb/hwsdata subdirectory for AIX and OS/390
- In the /opt/imsweb/hwsdata subdirectory for Sun Solaris

`HWSCMD DISPLAY CONV` can be used to display:

- All active IMS Web conversations, or
- A specific IMS Web conversation, or
- All active IMS Web conversations whose last iteration occurred more than *mmm* minutes ago.

Command syntax



HWSCMD Is the name of the command processor executable.

Note: For AIX, Sun Solaris, and OS/390 OpenEdition Web server platforms, the command processor executable (hwscmd) must be entered in lower case.

Parameters

DISPLAY
Specifies the display function of hwscmd.

CONV Is used to display IMS Web conversations. CONV is required.

ALL Is used to display all active IMS Web conversations.

ConvToken
Is used to display a specific IMS Web conversation.

GTmmm Is used to display IMS Web conversations whose last iteration occurred more than *mmm* minutes ago, where *mmm* must be in the range of 1 to 999.

HWSCMD DISPLAY command sample output

The following example shows output from the HWSCMD DISPLAY CONV ALL command entered on the Web server when two IMS Web conversations are in progress on the browser.

```
ConvToken  RUName   Next Input LPage    Date/Time Stamp    Age  IMS Name   Host Name
                                                         (minute)
===================================================================================
1I09276J   HWSGJAER HWSLPG01            02/05/98 09:10:09   8   SOCKEYE    CSDMEC13
1I09184Z   HWSGJ8F  HWSLPG01            02/05/98 09:08:24  10   SOCKEYE    CSDMEC13


Processing completed for HWSCMD DISPLAY CONV ALL 02/05/98 09:18:04
```

where:

- ConvToken is the IMS Web identifier for the conversation. This identifier is formed by converting the second half of the ServerToken field in the OTMA message into an 8–character identifier. ServerToken is OTMA's identification of the conversation. The first half of ServerToken is a STCK value; the second half of ServerToken is the STCK value converted into a readable date and time value. A conversation token is used as the filename of the conversation state file (ConvToken.DAT).
- RUName is Request Unit name. RUName is a unique name that identifies the request the IMS Web application makes to execute the IMS transaction. IMS uses this name as the name of the LTERM that is associated with the transaction.
- Next Input LPage is the value of the NXT= keyword of the message output descriptor (MOD) that was used to format the output of the current iteration of the conversation.
- Date/Time Stamp is the date and time the current iteration of the IMS Web conversation was presented to the end user. Date/time is updated by IMS Web Runtime, for each iteration of a conversation, when an output message is received from the IMS application program. The output message is then presented to the end user in an HTML page.
- Age (minute) is the length of time since the latest iteration of the conversation was presented to the end user. It is the difference between the current date and time and the date and time stamp, in minutes.
- IMS Name is the XCF member name for IMS OTMA. The output of the IMS command /DISPLAY CONVERSATION BUSY displays this name in the TERMINAL field for IMS Web initiated conversations.
- Host Name is the TCP/IP host name of the machine on which the IMS TCP/IP OTMA Connection is installed.

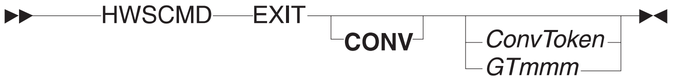
The following example shows output from the IMS /DIS CONV BUSY command when the same two IMS Web conversations are in progress. An additional conversation, initiated from an IMS terminal, is also in progress. This conversation is known to IMS, but not IMS Web.

```
00- 09.12.43  JOB00123  DFS000I  TERMINAL        USER      ID   STATUS - SYS3
  - 09.12.43  JOB00123  DFS000I  SOCKEYE         GOFISHIN  0001 ACTIVE - SYS3
  - 09.12.43  JOB00123  DFS000I  SOCKEYE         GOFISHIN  0002 ACTIVE - SYS3
  - 09.12.43  JOB00123  DFS000I  T1303           USRT002   0003 ACTIVE - SYS3
  - 09.12.43  JOB00123  DFS000I  *98036/091243*  SYS3
```

HWSCMD EXIT command

The HWSCMD EXIT command is used to terminate:

- A specific IMS Web conversation, or
- All IMS Web conversations whose last iteration occurred more than mmm minutes ago



Command syntax

HWSCMD Is the name of the command processor executable.

Note: For AIX, Sun Solaris, and OS/390 OpenEdition Web server platforms, the command processor executable (hwscmd) must be entered in lower case.

Parameters

EXIT Specifies the exit function of hwscmd.

CONV Is used to terminate IMS Web conversations. CONV is optional.

ConvToken
Is used to terminate a specific IMS Web conversation.

GTmmm Is used to terminate IMS Web conversations whose last iteration occurred more than *mmm* minutes ago, where *mmm* must be in the range of 1 to 999.

Normally, an IMS Web conversation is terminated by either the IMS application program or when the end user selects the TERMINATE button on an intermediate HTML form. However, if a browser session is closed while an IMS Web conversation is in progress, the conversation is still active and must be terminated using the HWSCMD EXIT CONV command. The HWSCMD EXIT CONV command may also be entered from the Web server while an IMS Web conversation is in progress on the browser. In this case, a protocol violation message is issued when the end user next selects the SUBMIT button.

The HWSCMD EXIT CONV command uses information in the conversation state file, convToken.DAT, to build a message and send it to IMS to end the conversation. HWSCMD EXIT CONV then deletes the conversation state file.

IMS Web Runtime data conversion

When data moves between products on different platforms, it might be converted from one representation to another. The representation of data on a particular platform is associated with a code page that reflects both the language of the data (e.g., U.S. English, Latin1, etc.) as well as the platform where the code page is used.

The data that you enter on a browser in the input HTML form generated by IMS Web Studio is converted by your Web server from the code page that is used by the browser to the code page that is used by the Web server. It is then converted by the IMS Web Runtime component from the Web server code page (see Note 1 on page 41 on page 41) to the host code page used by IMS for data. After the transaction runs on IMS and a response is sent back, the code page conversion process is followed in reverse so that data in the response is eventually converted to the correct code page for the originating browser. This enables the response to be displayed correctly in the output HTML.

The code page conversion tables used by the IMS Web Runtime component are packaged with the component. The following tables list the code pages used by the IMS Web Runtime component, based on Web server platform and language.

Web Server Platform NT, AIX, or Solaris

Language	Web Server Code Page	Host Code Page
U.S.English	IBM-819	IBM-37
Latin1	IBM-819	IBM-500
Latin2	IBM-912	IBM-870
Hebrew	IBM-916	IBM-424
Turkish	IBM-920	IBM-1026
German	IBM-819	IBM-273
Demark-Norway	IBM-819	IBM-277
Finland-Sweden	IBM-819	IBM-278
Italian	IBM-819	IBM-280
Spain-Latin America	IBM-819	IBM-284
UK English	IBM-819	IBM-285
French	IBM-819	IBM-297
Icelandic	IBM-819	IBM-871

Web Server Platform OS/390

Language	Web Server Code Page	Host Code Page
U.S.English	IBM-1047	IBM-37
Latin1	IBM-1047	IBM-500
Latin2	IBM-870	IBM-870
Hebrew	IBM-424	IBM-424
Turkish	IBM-1026	IBM-1026
German	IBM-1047	IBM-273
Denmark-Norway	IBM-1047	IBM-277
Finland-Sweden	IBM-1047	IBM-278
Italian	IBM-1047	IBM-280
Spain-Latin America	IBM-1047	IBM-284
UK English	IBM-1047	IBM-285
French	IBM-1047	IBM-297
Icelandic	IBM-1047	IBm-871

Considerations when using ICS for OS/390

When using the IMS Web Runtime component with ICS for OS/390, you must set the FsCp parameter in the ICS configuration parameters to the value for the Web server code page used by IMS Web in order to guarantee that all data will be converted correctly. See Note 2. For example:

Language	FsCP	NetCp
U.S.English	IBM-1047	IBM-819
Latin1	IBM-1047	IBM-819
Latin2	IBM-870	IBM-912
Hebrew	IBM-424	IBM-916
Turkish	IBM-1026	IBM-920
German	IBM-1047	IBM-819
Denmark-Norway	IBM-1047	IBM-819
Finland-Swedan	IBM-1047	IBM-819
Italian	IBM-1047	IBM-819
Spain-Latin America	IBM-1047	IBM-819
UK English	IBM-1047	IBM-819
French	IBM-1047	IBM-819
Icelandic	IBM-1047	IBM-819

Notes:

1. This assumes that you have followed the recommendations in “Considerations when using ICS for OS/390”.
2. Some or all of your data may be converted correctly even if the FsCp setting in ICS for OS/390 does not match the Web server code page used by IMS Web (see “Web Server Platform OS/390” on page 40) since some of the conversion tables are identical except for a small number of characters.

   Recommendation: However, it is **strongly** recommended that you set the FsCp in your Web server’s configuration parameters to match IMS Web’s Web server code page in order to guarantee that all data will be converted correctly.

Readers' Comments — We'd Like to Hear from You

IMS Web User's Guide

Publication No. webu-0002-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	□	□	□	□	□

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	□	□	□	□	□
Complete	□	□	□	□	□
Easy to find	□	□	□	□	□
Easy to understand	□	□	□	□	□
Well organized	□	□	□	□	□
Applicable to your tasks	□	□	□	□	□

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? □ Yes □ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.

Readers' Comments — We'd Like to Hear from You

webu-0002-01

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department BWE/H3
P.O. Box 49023
San Jose, CA 95161-9945

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line