

CICS Transaction Server for z/OS



因特网指南

版本 3 发行版 1

CICS Transaction Server for z/OS



因特网指南

版本 3 发行版 1

注意！

在使用本资料及其支持的产品之前，请确保阅读第 255 页的『声明』中的一般信息。

第一版（2005 年 3 月）

本版本适用于 CICS Transaction Server for z/OS 的 V3R1（程序号 5655-M15）及所有后续版本、发行版和修正版，直到在新版本中另有声明。确认您正在使用本产品级别的正确版本。

请通过您当地的 IBM 代表或 IBM 分部可订购出版物。以下地址不备有出版物。

在本出版物的背面有一页标题为“将您的意见发送给 IBM”的内容。如果希望通过邮件发送意见，请寄往以下地址：

User Technologies Department
Mail Point 095
IBM United Kingdom Laboratories
Hursley Park
WINCHESTER
Hampshire
SO21 2JN
United Kingdom

当您发送信息给 IBM 后，即授予 IBM 非专有权，IBM 可以它认为合适的任何方式使用或分发此信息，而无须对您承担任何责任。

© Copyright International Business Machines Corporation 1994, 2005. All rights reserved.

目录

前言	ix
本书内容	ix
理解本书需要了解的内容	ix
有关术语的说明	ix
更改摘要	xi
对 CICS Transaction Server for z/OS V3R1 的更改	xi
CICS Transaction Server for z/OS V2 的更改	xi

第 1 部分 CICS Web support 的概述 1

第 1 章 将 CICS 连接到 Web 3

第 2 章 因特网、HTTP 和 TCP/IP 概念 5

TCP/IP 协议	5
IP 地址	5
主机名	6
虚托管	6
端口号	7
IANA 介质类型和字符集	7
URL 的组成部分	7
HTTP 协议	8
HTTP 请求	9
HTTP 响应	10
状态码和原因短语	12
转义和未转义数据	12
HTML 表单	13
分块的传输编码	14
管道传送	14
持续连接	15
HTTP 基本认证	15

第 3 章 CICS Web support 概念和结构 17

CICS Web support 的组件	17
CICS Web support 的任务结构	20
作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理	21
作为 HTTP 客户机的 CICS 的 HTTP 请求和响应处理	24
会话令牌	27
CICS Web support 的 URL	28
CICS Web support 如何处理分块的传输编码	29
CICS Web support 如何处理管道传送	30
CICS Web support 如何处理持续连接	31
CICS Web support 的代码页转换	32
作为 HTTP 服务器的 CICS 的代码页转换	33
作为 HTTP 客户机的 CICS 的代码页转换	34
作为 HTTP 服务器的 CICS 的 HTTP/1.1 一致性	35
遵从 HTTP/1.1 的 CICS Web support 行为	36
CICS Web support 不支持 HTTP 功能	37

第 2 部分 CICS Web support 39

第 4 章 配置 CICS Web support 基本组件	41
为 CICS Web support 指定系统初始化参数	42
保留 CICS Web support 的端口	42
迁移代码页转换表 (DFHCNV) 中的条目	43
验证 CICS Web support 的操作	43
第 5 章 规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构	45
用支持 Web 的应用程序提供动态 HTTP 响应	45
用 CICS 文档模板或 HFS 文件提供静态 HTTP 响应	48
提供对通信区域应用程序的 Web 客户机访问	51
第 6 章 为作为 HTTP 服务器的 CICS 编写支持 Web 的应用程序	57
检查 HTTP 请求的请求行	58
检查消息的 HTTP 头	59
检索有关 HTTP 请求的技术和安全信息	60
检查 HTTP 请求中的表单数据	61
接收 HTTP 请求的实体主体	62
为响应写 HTTP 头	64
为 HTTP 消息产生实体主体	65
从作为 HTTP 服务器的 CICS 发送 HTTP 响应	66
使用分块的传输编码发送 HTTP 请求或响应	67
跨 HTTP 请求序列管理应用程序状态	70
第 7 章 CICS Web support 的资源定义	71
为 CICS Web support 创建 TCPIPSERVICE 资源定义	71
为 CICS Web support 创建 TRANSACTION 资源定义	74
为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义	75
为 CICS (作为 HTTP 服务器) 的 HTTP 请求的应用程序响应完成 URIMAP 定义	77
为作为 HTTP 服务器的 CICS 的 HTTP 请求的静态响应完成 URIMAP 定义	77
第 8 章 管理 CICS Web support	79
管理 CICS Web support 资源	79
管理虚托管	81
将 HTTP 请求重定向到另一个 URL	82
拒绝 HTTP 请求	82
警告标题	83
第 9 章 Web 出错程序	85
DFHWBERX, Web 出错应用程序	86
DFHWBEP, Web 出错程序	86
到 Web 出错程序的输入	88
来自 Web 出错程序的输出	88
缺省状态码和错误响应	89
第 10 章 分析器程序	91
编写分析器程序	93
分析器程序的输入	94
分析器程序的输出	95
共享分析器和转换器程序之间的数据	97
从分析器程序选择转义数据或未转义数据	97

CICS 提供的缺省分析器程序 DFHWBAAX	98
CICS 提供的样本分析器程序 DFHWBADX	99
第 11 章 转换器程序	101
写转换器	102
解码函数的输入参数	103
解码函数的输出参数	103
编码函数的输入参数	104
编码函数的输出参数	104
转换器程序: 构造 HTTP 响应	104
调用多个 CICS 应用程序	105
第 12 章 CICS Web support 的安全性	107
CICS Web support 的 CICS 系统和资源安全性	107
HTTP 客户机的认证和身份识别	108
作为 HTTP 服务器的 CICS: 认证和标识	109
作为 HTTP 客户机的 CICS: 认证和标识	110
HTTP 基本认证的密码到期管理	111
具有 CICS Web support 的 SSL	112
第 13 章 来自 CICS 应用程序的 HTTP 客户机请求	115
通过作为 HTTP 客户机的 CICS 发出 HTTP 请求	115
打开到 HTTP 服务器的连接	116
为请求写 HTTP 头	117
写 HTTP 请求	118
发送管道化序列的请求	120
接收 HTTP 响应	121
关闭到 HTTP 服务器的连接	122
通过作为 HTTP 客户机的 CICS 创建 HTTP 请求的 URIMAP 定义	123
HTTP 客户机开放出口 XWBOPEN	124
HTTP 客户机发送出口 XWBSNDO	125
第 14 章 CICS Web support 和非 HTTP 请求	127
非 HTTP 请求的资源定义	128
分析器程序和非 HTTP 请求	128
用于非 HTTP 请求的应用程序编程	129
第 15 章 CICS Web support 和 3270 显示应用程序	131
3270 应用程序的 CICS Web support 处理	132
3270 显示应用程序的 URL 路径部分	132
初始和连续请求	134
连续请求的事务标识	135
HTML 表单中的事务标识	135
3270 应用程序的 CICS Web support 中的终端控制命令	136
从 BMS 映射生成的 HTML 模板	136
从 3270 数据流生成的 HTML 页面	137
修改 DFHWBTTA 的输出	140
提供您自己的标题模板	141
提供您自己的页脚模板	142
将转换器程序与 DFHWBTTA 协同使用	142
启用可检测字段	143
使用可检测字段	143
使用 DFHWBIMG 显示图形	144

第 16 章 从 BMS 定义创建 HTML 模板	147
有关 BMS 生成的模板	147
生成定制的 HTML 模板	147
使用 DFHMSX 宏来定制	148
安装 HTML 模板	149
HTML 模板的大小限制	149
编写定制宏定义	150
处理空格	150
组合 BMS 和非 BMS 输出	150
如何选择标题部分	151
如何选择页脚部分	151
如何合并屏幕图像部分	151
DFHMDX 宏	153
使用 DFHWBOUT 宏定制模板	157
定制示例	158
第 17 章 CICSplex 中的 CICS Web support	163
将 Web 客户机的请求路由到 AOR	164
网络负载均衡	167

第 3 部分 CICS 业务逻辑接口 **169**

第 18 章 CICS 业务逻辑接口的介绍	171
如何使用 CICS 业务逻辑接口	171
处理示例	171
请求处理中的控制流	172
使用 CICS 业务逻辑接口调用程序	172
使用 CICS 业务逻辑接口运行面向终端的事务	173
请求处理中的数据流	173
转换器程序和 CICS 业务逻辑接口	174
使用 CICS 业务逻辑接口调用程序	174
面向终端的事务的请求	175
偏移方式和指针方式	178
代码页转换和 CICS 业务逻辑接口	179
配置 CICS 业务逻辑接口	179

第 4 部分 附录 **181**

附录 A. HTML 编码字符集	183
附录 B. CICS Web support 的 HTTP 头参考	185
附录 C. CICS Web support 的 HTTP 状态码参考	191
附录 D. CICS Web support 的 HTTP 方法参考	199
附录 E. 分析器程序的参考信息	203
分析器程序参数摘要	203
分析器程序的参数	204
响应和原因码	208
附录 F. 转换器程序的参考信息	209
解码	209

编码	214
附录 G. DFHWBBLI CICS 业务逻辑接口的参考信息	217
参数摘要	217
参数	218
响应	221
附录 H. Web 出错程序 DFHWBEP 的参考信息	223
参数	223
附录 I. DFHWBCLI Web 客户机接口	227
附录 J. 状态管理样本 DFH\$WBST 和 DFH\$WBSR 的参考信息	233
附录 K. CICS Web 服务器插件	235
配置 IBM HTTP Server	235
转义数据和 IBM HTTP Server	237
IBM HTTP Server 的处理示例	237
书目	239
CICS Transaction Server for z/OS 库	239
基本集合	239
仅 PDF 书籍	239
其他 CICS 书籍	241
非 CICS 书籍	241
UNIX 系统服务	241
z/OS Communications Server	242
IBM Redbooks	242
Web 上的信息	242
确定出版物是否是最新的	243
辅助功能选项	245
索引	247
声明	255
编程接口信息	256
商标	256
将您的意见发送给 IBM	257

前言

本书内容

本书说明如何设置和管理 CICS® Web support 以允许 CICS 区域作为 HTTP 服务器和 HTTP 客户机，以及如何编写与 Web 客户机和服务器交互作用的 CICS 应用程序。

理解本书需要了解的内容

本书假定您担当熟悉 CICS 的系统管理员或系统或应用程序员。

有关术语的说明

本书（通常与 CICS 文档一起）是指对作为 CICS Web support 的 HTTP 协议的支持。过去，术语 CICS Web 接口用于描述相同功能。有一些地方仍然使用旧的术语，包含消息和样本程序。

更改摘要

本书基于 CICS Internet Guide for CICS Transaction Server for z/OS® V2R3, SC34-5713-00。该版本的更改在左边空白处用垂直条标记。

对 CICS Transaction Server for z/OS V3R1 的更改

本书已更新，且大量修改为支持 HTTP/1.1、支持 HTTP 客户机请求的文档，还有对 CICS Web support 的其他改进，包括 URIMAP 定义的介绍。

在第 1 部分中，资料已完全修改，且添加了有关新功能（例如，分块的传输编码和管道传送）的信息。

第 2 部分中的大多数章节已更新和 / 或重新编排。以下章节主要包含本书中主要的新资料：

- 第 5 章，『规划您的 CICS Web support 体系结构』
- 第 6 章，『编写支持 Web 的应用程序』（从 *CICS Application Programming Guide* 返回到本书）
- 第 7 章，『CICS Web support 的资源定义』
- 第 8 章，『管理 CICS Web support』
- 第 13 章，『来自 CICS 应用程序的 HTTP 客户机请求』

第 3 部分，有关 CICS 业务逻辑接口，未作重要的更改。

添加了以下新附录：

- 附录 B，『CICS Web support 的 HTTP 头参考』
- 附录 C，『CICS Web support 的 HTTP 状态码参考』
- 附录 D，『CICS Web support 的 HTTP 方法参考』

除去了以下附录：

- 『HTML 模板管理器的参考信息』（DFHWBTL）。EXEC CICS DOCUMENT 命令提供此函数。
- 『环境变量程序的参考信息』（DFHWBENV）。EXEC CICS WEB 命令提供此函数。
- 『DFHWBPA 的参考信息』（CICS Web support 解析器程序）。EXEC CICS WEB FORMFIELD 命令提供此函数。

CICS Transaction Server for z/OS V2 的更改

以下是对 CICS Transaction Server for z/OS V2R1、R2 或 R3 进行的重要更改：

技术更改

- 添加了“CICSplex® 中的 CICS Web support”一章。
- 添加了附录“The DFHWBCLI Web Client interface”。

结构更改

- 有关使用 CICS 文档模板的信息已移到 *CICS Application Programming Guide*。

- 有关 IIOP 入站到 Java™ 应用程序的信息已移到 *Java Applications in CICS*。
- 有关安全套接字层 (SSL) 的信息已移到 *CICS RACF® Security Guide*

在 CICS Transaction Server for OS/390® V1R3 中, CICS 业务逻辑接口将它的名称从 DFHWBA1 更改为 DFHWBBLI, 并将它的参数从 **wba1_** 更改为 **wbbl_**。

第 1 部分 CICS Web support的概述

本书的这一部分概述了一些方法，通过使用这些方法，可以让各种因特网用户使用 CICS 事务处理服务。

第 1 章 将 CICS 连接到 Web

CICS 可以作为服务器与 Web 连接，从 Web 客户机接收请求；或者作为客户机对服务器发出请求。

由 CICS 应用程序提供服务的 Web 客户机请求

使用 CICS Web support

CICS Web support 允许 CICS 区域作为 HTTP 服务器。

- 使用 CICS 文档或静态文件，CICS Web support 可以对 Web 客户机提供静态响应。
- 支持 Web 的用户应用程序可以接收和分析 HTTP 请求，并提供动态应用程序生成的响应。
- CICS Web support 包括一系列 CICS 服务，这些服务支持 Web 客户机对不支持 Web 的应用程序的访问。Web 客户机可以发出请求以访问设计为与虚拟 3270 终端通信的 CICS 程序，以及设计为使用通信区域或通道从另一个 CICS 应用程序链接所至的 CICS 程序。
- CICS Web support 还支持来自客户机的非 HTTP 请求。

CICS Web support 在第 17 页的第 3 章，『CICS Web support 概念和结构』中描述。

使用 Web service

Web service 是一个软件系统，设计的目的是支持网络上协同工作的机器间的交互。它有一个用机器可处理的格式（明确地说是 Web Service 定义语言（WSDL））描述的接口。CICS Transaction Server V3 可以是 Web service 的请求者或提供者。Web service 在 *CICS Web Services Guide* 中描述。

使用 IBM® HTTP Server

IBM HTTP Server 通过外部 CICS 接口（EXCI）和 CICS 业务逻辑接口对 CICS 应用程序提供访问。要获得更多信息，请参阅第 235 页的附录 K，『CICS Web 服务器插件』和第 171 页的第 18 章，『CICS 业务逻辑接口的介绍』。

使用 CICS Transaction Gateway

CICS Transaction Gateway 提供一组 Web 服务器工具，以由 Web 客户机访问 CICS 应用程序。它们包括用于编写特定于应用程序的服务器程序（servlets）和浏览器程序（applet）的 Java 类和 Java bean，以及 IBM 为公共功能提供的代码。有一些类用于访问传统和面向对象的 CICS 应用程序。Applet 和 servlet 使用 CICS 提供的类构造 ECI（外部调用接口）和 EPI（外部演示接口）请求。（请注意，CICS Transaction Gateway for z/OS 支持 ECI，但是不支持 EPI。）要获得更多信息，请参阅 *CICS Transaction Gateway: z/OS Administration*。

要获取有关选择 Web 解决方案的指导，请参阅 IBM Redbook *Revealed! Architecting Web Access to CICS*, SG24-5466，它可以从 <http://www.redbooks.ibm.com/redbooks/pdfs/sg245466.pdf> 获取。

访问 Web 的 CICS 应用程序

CICS Web support 允许 CICS 区域作为 HTTP 客户机。CICS 中的用户应用程序可以启动对 HTTP 服务器的请求，并从 HTTP 服务器接收响应。CICS Web support 处理

| 响应用户应用程序中 EXEC CICS WEB 命令的消息。CICS Web support 在第 17 页的
| 第 3 章, 『CICS Web support 概念和结构』中描述。

第 2 章 因特网、HTTP 和 TCP/IP 概念

本部分说明超文本传输协议（HTTP）和传输控制协议 / 网际协议（TCP/IP）的关键元素。

TCP/IP 协议

TCP/IP 是用于在网络上连接计算机系统的一系列通信协议，它根据该系列中的两个协议命名：传输控制协议（TCP）和网际协议（IP）。超文本传输协议（HTTP）是 TCP/IP 系列的成员。

在许多情况下，TCP/IP 系列中的协议与开放式系统互连（OSI）模型的层一致。表 1 按照 OSI 模型显示了 TCP/IP 系列的 HTTP 和底层的层。同时还显示系统网络体系结构（SNA）层，这些层与 OSI 层大致匹配。

表 1. TCP/IP 协议系列层

层	OSI	SNA	TCP/IP
7	应用程序	应用程序	HTTP
6	演示	演示	(空)
5	会话	数据流	(空)
4	传输	传输	TCP
3	网络	路径控制	IP
2	数据链路	数据链路	子网
1	物理	物理	

网际协议（IP）

IP 是网络层协议，该协议提供 TCP 使用的无连接数据传输服务。一条链路接着一链路地传输数据；呼叫期间从来不建立端到端连接。数据传输的单位是数据报。

传输控制协议（TCP）

TCP 是传输层协议，该协议提供可靠的、全双工的、面向连接的数据传输服务。大多数因特网应用程序使用 TCP。

超文本传输协议（HTTP）

HTTP 是应用层协议，该协议用于分布式、协作式以及超媒体信息系统。HTTP 是在 Web 客户机和 Web 服务器之间使用的协议。

许多 TCP/IP 实施提供 TCP 协议（也就是传输层）的应用程序编程接口。该接口通常称为套接字接口。用于 CICS 的 TCP/IP 套接字接口是 z/OS Communications Server IP CICS 套接字接口。它随 z/OS Communications Server（而不是 CICS）一起提供，并且是 z/OS 的完整部分。它不是 CICS Web support 的一部分，并且不使用 CICS SO 域。*z/OS Communications Server: IP CICS Sockets Guide, SC31-8807*，描述了 CICS 套接字接口。

IP 地址

TCP/IP 因特网上的每台服务器或客户机都由数字 IP 地址识别。IP 代表网际协议。

标准 IP 地址类型 IPv4 (IP 版本 4) 地址是 32 位的。为了让人们方便阅读, 它通常以点分十进制表示法表示:

```
IP address in hexadecimal notation : '817EB263'x
                                     Byte 0: 81 hexadecimal = 129 decimal
                                     Byte 1: 7E hexadecimal = 126 decimal
                                     Byte 2: B2 hexadecimal = 178 decimal
                                     Byte 3: 63 hexadecimal = 99 decimal
IP address in dotted decimal notation: 129.126.178.99
```

较新的 IP 地址类型 IPv6 (IP 版本 6) 地址是 128 位的。它通常以十六进制表示法表示, 而不以点分十进制表示法表示。(CICS Web support 不支持 IPv6 地址。)

IP 地址由因特网地址管理机构 (IANA) 及其代表管理并分配给用户。

主机名

因特网上的主机或 Web 站点由主机名识别, 例如, `www.example.com`。主机名有时称为域名。主机名映射到 IP 地址, 但是主机名和 IP 地址之间没有一对一关系。

当 Web 客户机发出到主机的 HTTP 请求时, 使用主机名。发出请求的用户可能会指定服务器的 IP 地址, 而不是主机名, 但现在这在因特网上不常见。对于用户来说, 主机名比数字 IP 地址更方便。公司、组织和个人常常选择其 Web 站点的主机名, 用户能很容易地记住这些主机名。

现代 HTTP 实施中更重要的是, 在 HTTP 请求中使用主机名意味着:

- 一个主机名中的服务可以由许多服务器提供, 它们有不同的 IP 地址。
- 具有一个 IP 地址的一台服务器可以提供许多主机名中的服务。这称为**虚托管**。『虚托管』说明此过程。

主机名由称为 **DNS 服务器**或**域名服务器**的服务器映射到 IP 地址。DNS 代表域名服务。在大型网络中, 许多 DNS 服务器可以相互协作, 以提供主机名和 IP 地址之间的映射。

虚托管

HTTP 包括虚托管的概念, 虚托管的情况下单个 HTTP 服务器可代表同一 IP 地址的多个主机。

DNS 服务器可以向同一 IP 地址分配几个不同的主机名。当 HTTP 客户机向特定主机发出请求时, 它使用 DNS 服务器查找与此主机名相应的 IP 地址, 并向此 IP 地址发送请求。

在 HTTP/1.0 中, 主机名未出现在 HTTP 消息中; 解析了 IP 地址后它已丢失。这意味着如果此 IP 地址表示的服务器已拥有多个资源集, 则服务器区分哪些资源属于哪个主机时会有困难。

然而, HTTP/1.1 请求在请求 (通常在 Host 头中) 中提供主机名。主机名出现在消息中使 HTTP 服务器能将包含不同主机名的请求导向到每个主机相应的资源。HTTP 的此功能称为虚托管。CICS Web support 通过使用 URIMAP 定义提供对虚托管的支持。

端口号

在一台服务器中，多个用户进程同时使用 TCP 是有可能的。为了识别与每个进程有关的数据，将使用端口号。端口号是 16 位的，虽然实际上通常只使用端口号的一小部分子集，但允许的最大端口号是 65535。

当客户机进程第一次与服务器进程联系时，它可能使用熟知端口号来启动通信。熟知端口号由 IANA（因特网地址管理机构）通过因特网分配给特殊服务。熟知端口号的范围是 0 到 1023。表 2 中显示了某些示例：

表 2. 服务及其熟知端口号

服务	熟知端口号
文件传输协议 (FTP)	21
Telnet	23
超文本传输协议 (HTTP)	80
具有安全套接字层 (SSL) 的 HTTP	443
CORBA 因特网 ORB 间协议 (IIOP)	683
具有 SSL 的 CORBA IIOP	684

CICS 外部调用接口 (ECI) 有注册的端口号 1435。

仅使用熟知端口来建立客户机进程和服务器进程之间的通信。完成此操作后，服务器将分配临时端口号以供随后使用。临时端口号是进程开始通信时动态分配的唯一端口号。当通信结束时，释放这些端口号。

IANA 介质类型和字符集

因特网地址管理机构 (IANA) 是负责为因特网上使用的协议指定名称的国际性机构。

- IANA 媒体类型是在因特网上普遍发送的数据类型的名称。在此地址有关于它们的描述：<http://www.iana.org/assignments/media-types/>

文本介质类型（如以 `text/`，开始的类型或包含 `+xml` 的类型）由 RFC 3023 标识，可以在 <http://www.ietf.org/rfc/rfc3023.txt> 上获得它们。

- IANA 字符集是字符集注册表的名称。在此地址有关于它们的描述：<http://www.iana.org/assignments/character-sets>

CICS 不支持用于代码页转换的所有 IANA 字符集。CICS 支持的字符集在 第 183 页的附录 A，『HTML 编码字符集』中描述。

URL 的组成部分

URL（统一资源定位符）是 URI（通用资源标识）的特定类型。URL 通常在因特网上查找现有资源。当 Web 客户机向服务器发出对资源的请求时，使用 URL。

URI 和 URL 的概念由因特网协会和 IETF（因特网工程任务组织）请求评论文档 RFC 2396 统一资源标识 (URI)：一般语法定义 (<http://www.ietf.org/rfc/rfc2396.txt>)。简要说，URI 是定义为识别资源的任何一个字符串。URL 定义为按资源的位置或用户访问它的方式，而不是按资源的名称或其他属性来识别资源的那些 URI。

HTTP (HTTPS) 的 URL 通常由三或四个组成部分组成:

1. **规则。** 规则识别用于访问因特网上的资源的协议。它可以是 HTTP (不带 SSL) 或 HTTPS (带 SSL)。
2. **主机。** 主机名识别拥有资源的主机。例如, `www.example.com`。服务器在主机的名称中提供服务, 但主机和服务器之间没有一对一映射。第 6 页的『主机名』说明关于主机名的更多信息。

主机名也可以后跟**端口号**。第 7 页的『端口号』说明关于这方面的更多内容。服务的熟知端口号通常从 URL 中省略。因为多数服务器将熟知端口号用于 HTTP 和 HTTPS, 所以多数 HTTP URL 省略端口号。

3. **路径。** 路径识别主机中 Web 客户机要访问的特定资源。例如, `/software/htp/cics/index.html`。
4. **查询字符串。** 如果使用查询字符串, 则它跟随路径部分, 并且提供一串字符串, 资源使用这些字符串可以完成某些操作 (例如, 作为用于搜索的参数或用于处理的数据)。查询字符串通常是一串名称和值对, 例如, `q=blue+fish`。

URL 的规则和主机部分不定义为区分大小写, 但是路径和查询字符串是区分大小写的。通常, 整个 URL 指定为小写字母。

URL 的组成部分如下所示进行组合和定界:

`scheme://host:port/path?query`

- 规则后跟冒号和两个正斜杠。
- 如果指定端口号, 则主机名后面是号码, 并用冒号分隔。
- 路径名以单正斜杠开始。
- 如果指定查询字符串, 则在它的前面加个问号。

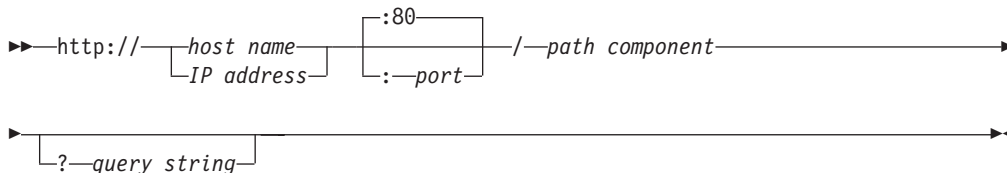


图 1. HTTP URL 语法

这是 HTTP URL 的示例:

`http://www.research.ibm.com/software/htp/cics/index.html`

如果指定了端口号, 则 URL 为:

`http://www.research.ibm.com:1030/software/htp/cics/index.html`

其他协议 (如文件传输协议 (FTP) 或 Gopher) 也使用 URL。这些协议使用的 URL 可能与 HTTP 使用的 URL 的语法不同。

HTTP 协议

HTTP 请求和响应的正确格式取决于由客户机和服务器使用的 HTTP 协议 (或 HTTP 规范) 版本。

通常在因特网上使用的 HTTP 协议版本（或“HTTP 版本”）是 HTTP/1.0（这是包含较少功能的早期协议）和 HTTP/1.1（这是包含较多功能的后期协议）。客户机和服务器可能使用不同版本的 HTTP 协议。客户机和服务器都必须在其消息的第一行声明其请求或响应的 HTTP 版本。

因特网协会和 IETF（因特网工程任务组织）请求评论文档（称为 RFC）为 HTTP 协议提供正式定义。这些文档为：

HTTP/1.0

RFC 1945, 超文本传输协议 - HTTP/1.0, 可从 <http://www.ietf.org/rfc/rfc1945.txt> 获取

HTTP/1.1

RFC 2616, 超文本传输协议 - HTTP/1.1, 可以从 <http://www.ietf.org/rfc/rfc2616.txt> 获取

RFC 声明客户机和服务器为了以下目的应该执行的操作：以适当的方式为每个版本的 HTTP 协议交换请求和响应。这些操作描述为“需求”。满足其 HTTP 协议版本需求的客户机或服务器被说成是“遵从”HTTP 规范。

HTTP 请求

HTTP 请求是客户机对位于服务器上的命名主机发出的。请求的目的是访问服务器上的资源。

要发出请求，客户机使用 URL（统一资源定位符）的组件，该 URL 包括访问资源所需的信息。第 7 页的『URL 的组成部分』说明 URL。

正确组成的 HTTP 请求包含以下元素：

1. 一个请求行。
2. 一系列 HTTP 头或头字段。
3. 消息体（需要的话）。

每个 HTTP 头后跟回车符换行（CRLF）。在 HTTP 头的最后，使用另一个 CRLF（空一行），然后开始是消息体。

请求行

请求行是请求消息中的第一行。它至少由三个项组成：

1. 一个方法。该方法是一个词的命令，告诉服务器它应该对资源采取什么操作。例如，可以要求服务器将资源发送到客户机。
2. 请求 URL 的路径部分。该路径识别服务器上的资源。
3. HTTP 版本号，显示客户机已尝试使消息遵从的 HTTP 规范。

请求行的一个示例为：

```
GET /software/htp/cics/index.html HTTP/1.1
```

在此示例中：

- 方法是 GET
- 路径是 /software/htp/cics/index.html

- HTTP 版本是 HTTP/1.1

请求行可能包含某些其他项:

- 一个查询字符串。它提供了资源可以为了某些目的而使用的一串信息。它跟随路径，前面有一个问号。
- URL 的规则和主机部分（除了路径）。资源位置以此方式指定时，它称为**绝对 URI** 格式。对于 HTTP/1.1，当请求将通过代理服务器时，使用此格式。同样对于 HTTP/1.1，如果 URL 的主机部分未包含在请求行中，则它必须包含在 Host 头的消息中。

HTTP 头

在消息上写 HTTP 头，以对接收方提供有关消息、发送方以及发送方想要与接收方通信的方式的信息。每个 HTTP 头由名称和值组成。HTTP 协议规范定义 HTTP 头的标准集合，并描述如何正确使用它们。HTTP 消息还可以包含扩展头，它们不是 HTTP/1.1 或 HTTP/1.0 规范的一部分。

客户机请求的 HTTP 头包含一些信息，服务器可以使用这些信息决定如何响应请求。例如，以下系列头可用于指定最终用户只想读取法语或德语的已请求文档，且仅当自从客户机上次获取该文档的日期和时间以来它已更改的情况下，才发送它：

```
Accept-Language: fr, de  
If-Modified-Since: Fri, 10 Dec 2004 11:22:13 GMT
```

请求消息中的 HTTP 头系列后放了一个空行（即，单个 CRLF），以将头与消息体分隔开。

消息体

任何 HTTP 消息的主体内容可以称为消息体或**实体主体**。从技术上说，实体主体是消息的实际内容。消息体包含实体主体，它可以处于其原始状态，或可以用某种方式编码以进行传输，例如，通过分成块（分块的传输编码）。为了方便起见，请求的消息体可以称为请求主体。

消息体适合某些请求方法而不适合其他请求方法。例如，具有 POST 方法的请求（它将输入数据发送到服务器）具有包含数据的消息体。具有 GET 方法的请求（它要求服务器发送资源）没有消息体。

HTTP 响应

服务器向客户机发出 HTTP 响应。响应的目的是对客户机提供它请求的资源，或通知客户机已执行它请求的操作；否则将通知客户机处理其请求时出现错误。

HTTP 响应包含:

1. 一个状态行。
2. 一系列 HTTP 头或头字段。
3. 一个消息体（通常需要）。

在一个请求消息中，每个 HTTP 头后跟回车符换行（CRLF）。在 HTTP 头的最后，使用另一个 CRLF（空一行），然后开始是消息体。

状态行

状态行是响应消息中的第一行。它由三个项组成:

1. **HTTP 版本号**, 显示服务器已尝试使消息遵从的 **HTTP** 规范。
2. **状态码**, 它是表明请求结果的三位数的号码。
3. **原因短语**, 也称为状态文本, 它是人类可以阅读的文本, 总结了状态码的含义。

响应行的一个示例为:

```
HTTP/1.1 200 OK
```

在此示例中:

- **HTTP 版本**是 HTTP/1.1
- **状态码**是 200
- **原因短语**是 OK

第 12 页的『状态码和原因短语』说明有关这些状态行元素的更多内容。

HTTP 头

服务器的响应的 **HTTP** 头包含一些信息, 客户机可以使用这些信息找出有关响应和发送响应的服务器的更多内容。该信息可以帮助客户机显示对用户的响应, 存储 (或高速缓存) 响应以供将来使用, 以及帮助客户机现在或将来对服务器发出更多请求。例如, 以下系列头告诉客户机发送响应的时间, 该响应由 CICS 发送, 并且它是 JPEG 图像:

```
Date: Thu, 09 Dec 2004 12:07:48 GMT  
Server: IBM_CICS_Transaction_Server/3.1.0(zOS)  
Content-type: image/jpeg
```

在不成功请求的情况下, 头可以用于告诉客户机它必须执行什么操作以成功完成其请求。

响应消息中的 **HTTP** 头系列后放了一个空行 (即, 单个 **CRLF**), 以将头与消息体分隔开。

消息体

为了方便起见, 响应的消息体可以称为响应主体。

消息体用于大多数响应。异常发生在服务器对使用 **HEAD** 方法 (它请求头, 但不请求响应主体) 的客户机请求做出响应的情况下, 以及服务器正在使用特定状态码的情况下。

对于对成功请求的响应, 消息体包含客户机请求的资源, 或有关客户机请求的操作状态的某些信息。对于对不成功请求的响应, 消息体可能提供有关错误原因的进一步信息, 或有关客户机为了成功完成请求需要执行的某些操作的进一步信息。

状态码和原因短语

在发送到客户机的 HTTP 响应中，三位数的状态码伴有总结代码含义的原因短语（也称为状态文本）在一起。这些项与响应的 HTTP 版本一起放在响应的第一行中，因此这一行称为状态行。

状态码按数字范围分类，这些代码的每个类都有相同的基本含义。

- 范围 100-199 分类为“信息”。
- 200-299 为“成功”。
- 300-399 为“重定向”。
- 400-499 为“客户机错误”。
- 500-599 为“服务器错误”。

将范围描述为整体时，它可能命名为“1xx”、“2xx”等。HTTP 协议规范不定义任何 600 或更大的状态码。

实际上每个范围中只有少数状态码由 HTTP/1.0 和 HTTP/1.1 规范定义。HTTP/1.1 规范包含的状态码比 HTTP/1.0 规范的状态码多。

建议使用 HTTP 规范中定义的原因短语（例如，“未找到”或“错误请求”），但这些短语是可选的。HTTP/1.1 规范声明：每个状态码的原因短语可以由本地同等短语替换。

200（确定）状态码用于常规响应，该响应提供 Web 客户机请求的完整资源。大多数其他状态码在以下情形下使用：存在阻止请求执行的错误，或者客户机需要执行某些其他操作以成功完成其请求（例如，跟随重定向 URL，或修改请求以使服务器可以接受它）。

响应的 HTTP 头和 / 或响应主体可以为客户机提供进一步指示信息和信息。HTTP 规范包括响应内容的需求和建议以及每个状态码。这些需求指定：

- 必须或可以在响应上使用的任何 HTTP 头。例如，如果使用状态码 405（不允许方法），则必须使用 Allow 头以声明允许哪些方法。
- 是否应该使用响应主体。例如，不允许消息体具有状态码 204、205 和 304。
- 如果使用了响应主体，它可以提供什么信息。例如，重定向的消息体可以提供重定向 URL 的超链接。

要获取有关状态码的含义和正确使用的完整信息，应该查询您正在遵从的 HTTP 规范。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

转义和未转义数据

为了有助于传输的正确和 HTTP 请求的解释，对于 URL 中使用某些字符存在一些限制。传输请求后，这些字符必须转换为安全格式。

在 URI 或 URL 中，一个或多个 URI 或 URL 组件上下文中有特殊用途的字符称为保留字符。例如，字符 /、?、& 和 : 用作各种组件的定界符。除了保留字符的特殊用途外，如果还出于任何其他原因使用这些字符，则机器解释器可能误释 URI 或 URL。

而且，不允许或排除在 URI 或 URL 中的任何地方使用特定字符，这是因为它们是混淆机器或人类用户的潜在原因，或是因为已知它们会导致某些机器解释器的问题。例如，URL 中不允许空格字符。

因特网协会和 IETF（因特网工程任务组织）请求评论文档 RFC 2396 统一资源标识（URI）：一般语法列出了 URI 和 URL 中保留或排除的字符。RFC 2396 可以从 <http://www.ietf.org/rfc/rfc2396.txt> 获取。

如果除了其特殊用途外，还出于任何其他原因在 URL 中需要保留字符，或者如果 URL 中需要排除的字符，则当包含 URL 组件的请求发送到服务器时，必须对这些字符进行转义。这包括以查询字符串发送的数据中的字符。

通过用格式为 %xx 的三个字符的字符串替换字符，可以对它们进行转义，其中 xx 是保留字符的 ASCII 十六进制表示法。例如，%2F 替代 / 字符。作为特例，空格字符可以由 + 替代。

请求到达服务器时，服务器可以取消转义转义字符，即，将它们从转义序列转换回原始字符。为了避免语法分析应用程序误释保留或排除字符的风险，取消转义应该仅当已语法分析 URL 和查询字符串中的信息后发生。

请求中的表单数据（不管它存在于 URL 中还是消息体中）通常与特殊转义字符一起发送，这是因为表单的缺省编码（**application/x-www-form-urlencoded**）转义保留或排除的字符。『HTML 表单』说明与此相关的更多内容。

HTML 表单

在 HTML 中，表单是由 <form> 标记定界的区域，包含文本输入框、按钮、复选框和图形用户界面的其他功能部件。Web 应用程序使用表单以允许最终用户提供要发送到服务器的数据。

在表单中，用户可与之交互作用以提供数据的元素称为**表单字段**。HTML 中对每个表单字段赋予一个名称，这会将它识别到服务器应用程序，但是对于用户不可视。

虽然一个表单的各种元素以不同样式显示给用户，但是它们都以相同的方式将信息传输给服务器应用程序：作为一系列由 & 字符分开的名称和值对。每个名称是表单字段的名称，而值是用户的操作产生的数据。例如，如果表单包含两个文本输入框以供用户输入他们的姓和名，则数据可能看上去与以下内容类似：

```
firstname=Maria&lastname=Smith
```

根据在 <form> 标记中指定哪种方法（GET 或 POST），表单数据以两种方式之一传输到服务器：

- 当方法是 GET 时，表单数据在 URL 的查询字符串中传输。
- 当方法是 POST 时，在消息体中传输表单数据。

表单数据通常用转义的特殊字符传输。第 12 页的『转义和未转义数据』说明转义的目的。

如果用 GET 方法定义表单，则因为数据作为 URL 中的查询字符串发送，所以必须总是转义保留或排除的字符。

如果用 POST 方法定义表单，则数据在消息体中发送。然而，如 HTML 2.0 规范中定义，所有表单的缺省编码类型是 **application/x-www-form-urlencoded**。（请参阅 http://www.w3.org/MarkUp/html-spec/html-spec_8.html#SEC8.2.1）当此编码用于具有 POST 方法的表单时，虽然数据在消息体中发送，但是也将转义保留或排除的字符，就如它们在 URL 中的情形一样。

如果为表单指定了备用编码类型 **multipart/form-data**（通过在 HTML `<form>` 标记上使用 ENCTYPE 属性来完成），则应该转义字段名中的非 ASCII 字符，但是字段值中的非 ASCII 字符不需要转义。该数据也出现在消息体的一系列独立部分中。较旧的应用程序可能不支持此编码。CICS 支持它。**multipart/form-data** 编码在因特网协会和 IETF 请求评论文档 RFC 1867 *HTML 中基于表单的文件上传*（<http://www.ietf.org/rfc/rfc1867.txt>）中进行了描述。

分块的传输编码

分块的传输编码也称为分块，它包括将消息体作为一系列块转换，每个分块都具有自己的块大小头。消息的结束由长度为零且带一空行的块来表明。

这个已定义过程意味着：可以在方便的段中发送应用程序生成的实体主体或大的实体主体。接收到长度为零的块后，客户机或服务器会知道分块的消息已完成。

分块的消息体后可以跟可选尾部，该尾部包含称为尾部头的补充 HTTP 头。接受尾部不需要客户机和服务器，因此补充 HTTP 头只应该提供非基本的信息，除非服务器知道客户机将接受尾部。

要使用分块的传输编码，客户机和服务器都必须使用 HTTP/1.1。分块的消息无法发送到 HTTP/1.0 客户机。HTTP/1.1 规范（RFC 2616）中定义了应用到分块的传输编码的需求和对尾部的使用。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

管道传送

管道传送涉及将多个 HTTP 请求发送到服务器而不等待响应的客户机。然后必须以接收请求的相同顺序从服务器返回响应。

确保请求是幂等的是请求方的职责。幂等性意味着：当重复所有或部分请求系列时，总是获得相同的结果。如果与服务器的连接出错时，这将确保客户机可以重试请求系列，即使它并不知道服务器是否已实施所有、部分请求或根本不实施请求。

HTTP/1.1 规范（RFC 2616）定义有关 HTTP 请求的幂等性的规则。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。简单地说，大多数请求方法都是幂等的（如果它们独立使用），这是因为每次使用方法时都会获得相同结果。（异常是 POST 方法，因为它更改服务器上的资源。）然而，在管道传送期间发送一系列请求时，该序列可能不是幂等的，特别是在资源有所更改的情况下。

如果规划管道传送请求，则检查是否可以在任何时刻终止请求序列，然后重新从头开始，而不会导致逻辑错误。如果不是这种情况，则分别发出请求并在每次请求后等待确认。

持续连接

持续连接是 Web 客户机和服务器之间的连接，可以为请求和响应的多次交换而重复使用。

在 HTTP/1.0 中，服务器的缺省操作是：它从 Web 客户机接收请求并发送响应后，关闭连接。如果 Web 客户机希望服务器保持连接打开，则它必须在请求中发送 `Connection: Keep-Alive` 头。

对于 HTTP/1.1，持续连接是缺省值。在 Web 客户机和服务器之间进行连接后，缺省情况下，服务器应该保持连接打开。仅当 Web 客户机通过发送 `Connection: close` 头请求了关闭，或达到了服务器的超时设置，或服务器遇到错误，才应该关闭连接。

因为不必为每个请求建立新的连接，所以持续连接提高了网络性能。与使用现有连接发出请求相比，建立新的连接消耗了大量其他网络资源。

HTTP 基本认证

HTTP 基本认证是简单的提问和响应机制，服务器可以通过它从客户机请求认证信息（用户标识和密码）。客户机在 `Authorization` 头中向服务器传递认证信息。认证信息是基本 64 位编码的。

对于客户机发出的请求，如果服务器需要认证信息，则服务器将发送具有以下各项的 HTTP 响应：401 状态码、表明认证错误的原因短语和 `WWW-Authenticate` 头。大多数 Web 客户机处理该响应的办法是：从最终用户请求用户标识和密码，并重新发送具有 `Authorization` 头的原始请求。客户机发出其原始请求时可以发送 `Authorization` 头，这可以由服务器接受，以避免提示和响应过程。

`WWW` 认证头包含域属性，该属性标识所请求的认证信息将要应用的资源集。每个域可使用不同的认证信息。Web 客户机可能会为每个域存储认证信息，以便最终用户不需要为每个请求重新输入信息。

注：仅当 Web 客户机和服务器之间的连接安全时，HTTP 基本认证规则才能被认为是安全的认证方法。如果该连接不安全，则该规则不提供足够的安全性以阻止未授权的用户发现和使用服务器的认证信息。如果存在密码被拦截的可能性，则基本认证应该与 SSL 结合使用，以便 SSL 加密用于保护用户标识和密码信息。

第 3 章 CICS Web support 概念和结构

CICS Web support 是允许 CICS 区域作为 HTTP 服务器和作为 HTTP 客户机的 CICS 服务集合。

作为 HTTP 服务器的 CICS

当 CICS 是 HTTP 服务器时，Web 客户机可以将 HTTP 请求发送到 CICS 并接收响应。该响应可以是 CICS 从文档模块或静态文件创建的静态响应，或者是由用户应用程序动态创建的应用程序生成的响应。

作为 HTTP 服务器的 CICS 的操作由以下各项控制：

1. 包括 TCPIP SERVICE 定义和 URIMAP 定义的系统初始化参数和资源定义，它们用于配置 CICS Web support 并指示 CICS 如何处理请求和响应。
2. 可用于分析和处理 HTTP 请求和响应的 CICS 实用程序。
3. 用于接收 HTTP 请求并提供 HTTP 响应的资料的用户编写的应用程序。它们可以是与 CICS Web support 协同使用而设计的支持 Web 的应用程序，或者是原先未设计为与 CICS Web support 协同使用的不支持 Web 的 CICS 应用程序。

作为 HTTP 服务器的 CICS Web support 的行为有条件地遵从 HTTP/1.1 规范，如 RFC 2616 中所描述。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

作为 HTTP 客户机的 CICS

当 CICS 是 HTTP 客户机时，CICS 中的用户应用程序可以启动对 HTTP 服务器的请求，并从 HTTP 服务器接收响应。

作为 HTTP 客户机的 CICS 的操作由用户编写的应用程序控制。EXEC CICS WEB 应用程序编程接口包括一些命令，应用程序可以使用这些命令构造和启动来自 CICS 的 HTTP 请求，并接收服务器发送的响应。URIMAP 资源定义可用于提供诸如 URL 或客户机证书标号的信息。

CICS Web support 和非 HTTP 消息

CICS Web support 还支持来自客户机的非 HTTP 请求。CICS Web support 的许多组件用于非 HTTP 请求和响应，如果 URIMAP 定义出现异常，则这些组件不用于非 HTTP 请求。非 HTTP 消息必须使用 TCPIP SERVICE 资源定义上的另一个协议（USER 协议），因此不对它们执行为 HTTP 消息执行的检查。

CICS Web support 的组件

CICS Web support 包括用于所有 CICS Web support 任务的某些基本组件，以及您为个别 CICS Web support 任务选择并配置的某些特定于任务的组件。

基本组件

- **TCP/IP** 支持在 CICS 中由 CICS SO（套接字）域提供，并具有 z/OS 提供的网络服务（z/OS Communications Server 以及对 DNS 服务器的访问）。
- **z/OS UNIX®** 系统服务用作 TCP/IP 支持的一部分，并且 CICS 区域需要访问这些服务。

- **安全套接字层 (SSL)** 支持用于为 CICS Web support 实施提供安全性。CICS 支持安全套接字层 (SSL) 3.0 协议, 以及传输层安全性 (TLS) 1.0 协议。(不支持 SSL 2.0)。请注意, CICS 文档中使用术语 SSL 时, 它通常指 SSL 和 TLS。CICS RACF Security Guide 中的“Support for security protocols”具有有关 SSL 和 TLS 的更多信息。
- **临时存储队列**用于存储请求行、状态行和 HTTP 头。
- **DOCCODEPAGE 系统初始化参数**指定 CICS 文档模板支持使用的缺省主机代码页。
- **LOCALCCSID 系统初始化参数**指定本地 CICS 区域的编码字符集标识 (它是被 CICS 当作应用程序缺省值的代码页)。
- **TCPIP 系统初始化参数**在启动时激活 CICS TCP/IP 服务。
- **WEBDELAY 系统初始化参数**仅在涉及 Web 3270 网桥工具的情况下才为不活动的 CICS Web 任务定义超时周期。其他 CICS Web 任务的超时由相关事务的 RTIMOUT 值处理, 或 (对于作为 HTTP 服务器的 CICS) 由 TCPIPSERVICE 定义的 SOCKETCLOSE 属性处理。
- **套接字侦听器任务 (CSOL)**通过连接 Web 连接任务检测入站 TCP/IP 连接请求, 并调用 CICS Web support。
- **Web 连接任务 (CWYN、CWYU 或别名)**从 Web 客户机接收数据并处理请求的初始处理, 这包括 URIMAP 匹配、HTTP 头的代码页转换、请求的分析和消息体的代码页转换。任务还预处理从 Web 客户机接收的分块的和管道化消息。如果交付了 (使用 URIMAP 定义) 静态响应, 则 Web 连接任务也处理此处理。

资源定义

- **TCPIPSERVICE 资源定义**用于定义为作为 HTTP 服务器的 CICS 使用的每个端口, 包括该端口上连接的安全选项以及入站请求的超时和最大大小限制。它们不用于作为 HTTP 客户机的 CICS。

注: TCPIPSERVICE 资源定义仅与 CICS 提供的 TCP/IP 服务协同使用, 而对 z/OS Communications Server IP CICS 套接字接口不起作用。CICS 的 TCP/IP 套接字接口随 z/OS Communications Server 一起提供, 它是 z/OS 的主要部分, 且不使用 CICS SO 域。

- **URIMAP 资源定义**匹配来自 Web 客户机的请求或到 HTTP 服务器的请求的 URL, 并对 CICS 提供有关如何处理请求的信息。URIMAP 定义合并并且可以替换 CICS Web support 处理函数, 这些函数已由与 TCPIPSERVICE 定义关联的分析器程序在 CICS Transaction Server for z/OS V3R1 之前提供。URIMAP 定义也可用于传递对 Web 客户机请求的静态响应, 而不涉及应用程序。
- **TRANSACTION 资源定义**用于为 HTTP 请求处理定义别名事务。CICS 为缺省别名事务 CWBA 提供资源定义。当 Web 连接任务已完成请求的初始处理时, 如果要产生应用程序生成的响应, 则别名事务将处理剩余处理阶段。这些阶段包括接收请求、执行应用程序的业务逻辑、HTTP 响应的构造和 HTTP 响应的代码页转换。

用户应用程序

- **支持 Web 的应用程序**可以为 CICS Web support 设计, 它使用 EXEC CICS WEB 和 EXEC CICS DOCUMENT 应用程序编程接口。对于作为 HTTP 服务器的 CICS, 这些程序可以接收和分析 HTTP 请求, 并对 Web 客户机提供应用程序生成的响应。对于作为 HTTP 客户机的 CICS, CICS 中的用户应用程序可以启动对服务器的 HTTP 请求, 并从服务器接收响应。

- **通信区域应用程序**设计为使用通信区域接口从另一个程序链接的程序，可以将 CICS Web support 与转换器程序协同使用来访问，以将它们的输出转换成 HTML 供传送到 Web 客户机。或者，您可以写链接到通信区域应用程序的支持 Web 的应用程序，并使用它的输出来提供 HTTP 响应。
- **3270 显示应用程序**，为了与 3270 终端通信而设计的程序，可以使用 Web 终端转换应用程序来访问。Web 终端转换应用程序创建的 HTML 输出可显示在 Web 浏览器中。

编程接口

- **EXEC CICS WEB** 应用程序编程接口解释和构造 HTTP 请求和 HTTP 响应。某些命令用于作为 HTTP 服务器的 CICS，某些命令用于作为 HTTP 客户机的 CICS，而某些命令对于两种 CICS Web support 格式都可用。
- **EXEC CICS DOCUMENT** 应用程序编程接口构造 CICS 文档以提供从 CICS 发出的响应或请求的主体。

CICS Web support 实用程序

- **分析器程序** 与 TCPIP SERVICE 定义关联。它们用于在这些情况下解释 HTTP 请求：如果 URIMAP 定义指定使用分析器程序，或如果不存在 URIMAP 定义。CICS 提供缺省分析器程序 DFHWBAAX（该程序提供基本错误处理），以及样本分析器程序 DFHWBADX（该程序支持使用 URL 格式的请求，CICS TS 3.1 前的 CICS Web support 使用此格式。）这两种分析器都可以用作您自己的分析器程序的基础。
- **转换器程序**可用于译码 HTTP 请求并将输入构造到用户应用程序。支持 Web 的应用程序通常不需要转换器程序，但不是为 CICS Web support 设计的不支持 Web 的应用程序可能需要它们。CICS 不提供转换器程序。您可以编写许多转换器程序，并选择您的 CICS 区域中的任何转换器程序来处理请求。
- **Web 出错程序**在 CICS Web support 过程中发生请求错误或异常终止时，对 Web 客户机提供错误响应。CICS 提供 Web 出错程序 DFHWBEP（该程序在大多数错误情形中使用），以及 Web 出错应用程序 DFHWBERX（当 URIMAP 匹配失败时，该程序与缺省分析器 DFHWBAAX 协同使用（且可以为其他情形指定））。Web 出错程序是用户可替代的程序，可以修改它们以定制或更改在每种错误情形下发送给 Web 客户机的错误响应。
- **Web 终端转换应用程序 DFHWBTTA**（以及它用于备用处理的别名 DFHWBTTB 和 DFHWBTTTC）可用于从为了与 3270 终端通信而设计的程序创建 HTML 输出。该程序使用 CICS 3270 网桥机制。使用 BMS 和应用程序和不使用 BMS 的应用程序都是受支持的。使用该功能不需要更改应用程序。
- **密码到期管理程序 DFHWBPW** 为连接指定了基本认证且用户密码到期后使用。该程序指导用户完成设置新密码的整个过程。您可以通过 DFHWBPW 定制或替换向用户呈现的 Web 页面。

文档构造设施

- **z/OS UNIX 系统服务 HFS** 文件可以作为对来自 Web 客户机的 HTTP 请求的响应主体。
- **文档模板**支持允许消息体从 HTML 片段构建，这些片段已在脱机状态下准备。
- **BMS 宏**从 BMS 映射集构造 HTML 文档模板。

代码页转换

CICS 提供一些工具，以将 HTTP 消息转换为适合用户应用程序的代码页或适合在因特网上使用的代码页。CICS 使用 z/OS 转换服务处理代码页转换。

在先前 CICS 发行版中必需的代码页转换表 (DFHCNV) 通常对 CICS Transaction Server for z/OS V3R1 中的 CICS Web support 不是必需的。异常情况是如果您要在先前 CICS 发行版中使用您编码的分析器程序引用 DFHCNV。在这种情况下，您必须继续提供代码页转换表或对分析器程序进行更新。第 43 页的『迁移代码页转换表 (DFHCNV) 中的条目』有关于这方面的更多信息。

CICS Web support 的任务结构

当 CICS Web support 在 CICS 区域中活动时，对于作为 HTTP 服务器的 CICS，使用不同的任务来侦听入站连接请求；接收来自套接字的数据并执行初始处理；以及包括由应用程序执行的、有关请求的工作。对于作为 HTTP 客户机的 CICS，只适用一个任务，即应用程序发出 HTTP 请求的任务。

套接字侦听器任务 (CSOL)

这是一个长期运行的 CICS 任务。CICS 系统中有一个套接字侦听器任务的实例。

该任务检测所有 CICS 定义的端口上的入站 TCP/IP 连接请求，并且调用与端口相关联的 CICS 服务。当此端口打算用于 CICS Web support (即，指定 HTTP 或 USER 作为协议) 时，Web 连接任务定义为此端口的 TCPIP SERVICE 资源定义中的事务，因此侦听器连接此任务。

Web 连接任务 (CWYN、CWYU 或别名)

当端口的 TCPIP SERVICE 定义具有协议 HTTP 时，Web 连接任务的缺省事务标识是 CWYN。当协议为 USER 时，缺省值为 CWYU。可改为使用别名，但事务总是执行程序 DFHWBXN。

当 Web 连接任务由套接字侦听器任务调用时，它要做的第一件事是发出 SOCKET RECEIVE 请求以从 Web 客户机接收数据。当已接收到某些数据时，Web 连接任务来应对 Web 客户机请求的初始处理。

- 对于 HTTP 请求 (在 HTTP 协议上)，初始处理包括 URIMAP 匹配、HTTP 头的代码页转换、请求的分析和消息体的代码页转换。该任务还预处理从 Web 客户机接收的分块的消息和管道消息。如果使用分析器程序，则该事务包含它。
- 对于非 HTTP 请求 (在 USER 协议上)，不发生初始处理。

如果静态响应交付到 HTTP 请求 (使用 URIMAP 定义)，则 Web 连接任务也处理此处理。如果需要应用程序生成的响应，则 Web 连接任务连接一个别名事务。

来自 Web 客户机的每个 HTTP 请求都有一个 Web 连接任务的实例，它处于处理的初始阶段。CICS Transaction Server for z/OS V3R1 之前，如果 Web 客户机和 CICS 有持续连接，则 CWYN 事务在持续连接的持续时间内会保留在系统中。现在，来自 Web 客户机的请求已传递到别名事务或已传递静态响应后，CWYN 事务会终止。套接字侦听器任务监控套接字，并在持续连接上为每个请求启动 CWYN 的新实例。当保留在系统中的 CWYN 事务无法连接别名事务来处理更多请求时，该行为 (称为异步接收) 避免在已达到最大任务指定数 (MXT) 的情况下可能出现死锁。

应用程序生成的响应的别名事务

当 Web 连接任务已完成请求的初始处理时，如果要产生应用程序生成的响应，Web 连接任务会连接为该请求的剩余处理阶段指定的别名事务。CICS 为缺省别名事务 CWBA 提供资源定义。在提供静态响应的情况下不使用别名事务。

别名事务处理应用程序生成的响应的处理阶段，包括接收请求、执行应用程序的业务逻辑、构造 HTTP 响应和 HTTP 响应的代码页转换。如果转换器程序用于处理请求，则别名事务也处理它。每个处于这些处理阶段中的 HTTP 请求都有一个别名事务的实例。

作为 HTTP 客户机的 CICS

对于作为 HTTP 客户机的 CICS，发出 HTTP 客户机请求的应用程序引起的所有活动都由单个任务所包含。这包括应用程序的操作、CICS 发送请求和接收请求的操作和套接字活动。如果应用程序使用 EXEC CICS LINK 命令链接到其他程序，则这些程序也包含在该任务中。该任务具有触发应用程序的事务标识。

该任务从应用程序的活动的开始到结束都保留在系统中。该任务可以涉及多个请求和响应，并且应用程序可以打开和保持到服务器的多个连接。当该任务结束时，会自动关闭所有打开的连接。

作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理

作为 HTTP 服务器的 CICS 的 HTTP 请求由对 CICS 发出请求的 Web 客户机启动。CICS 对 Web 客户机提供对它发出的请求的响应。响应可以从 URIMAP 资源定义识别的静态文档创建，或者它们可以由用户应用程序动态创建。

第 22 页的图 2 显示了 CICS Web support 为了从 Web 客户机接收请求并提供响应所执行的处理。

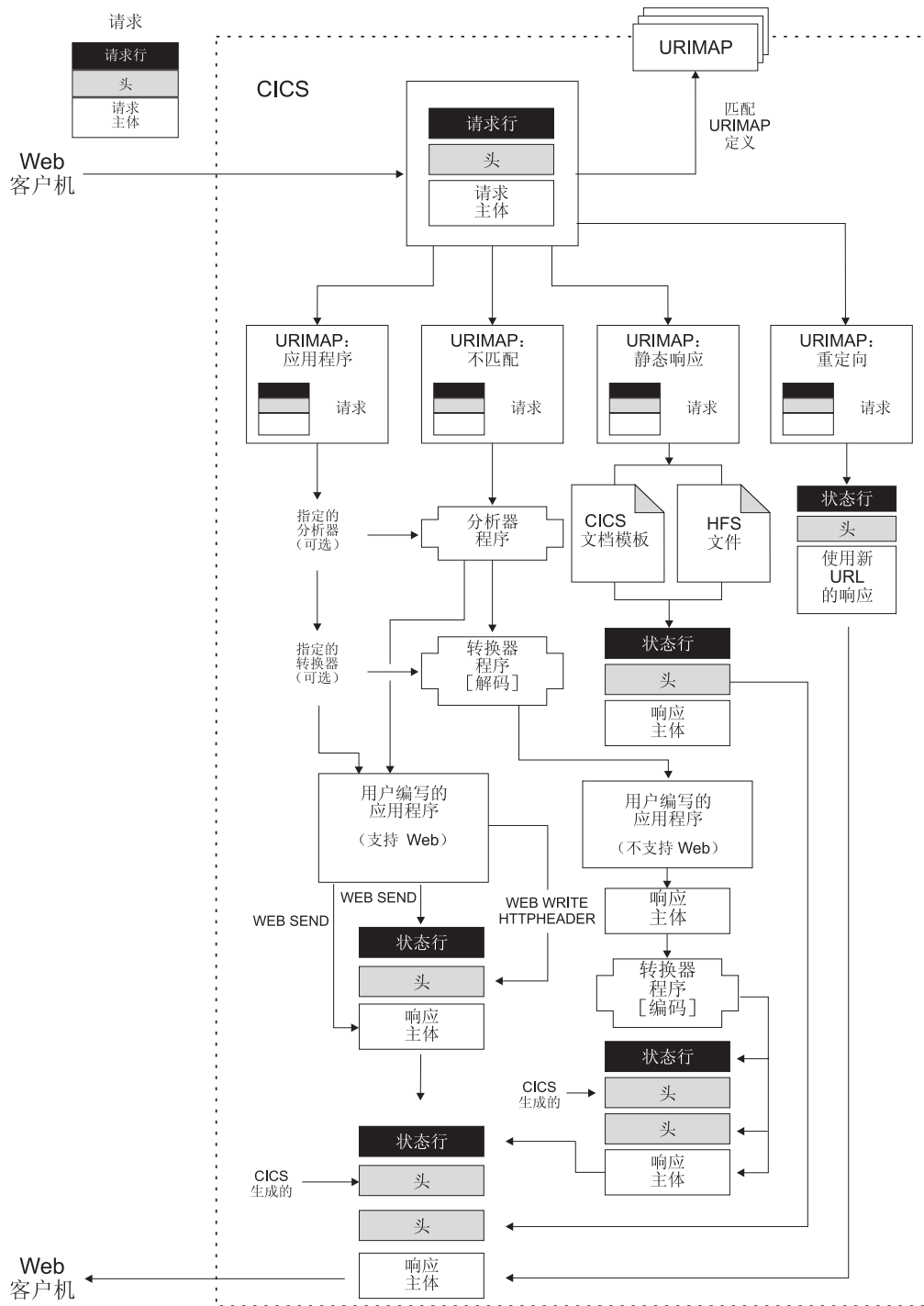


图 2. 作为 HTTP 服务器的 CICS 的处理

下面描述执行适用于作为 HTTP 服务器的 CICS 的处理:

1. **CICS 接收 TCP/IP 连接请求。** CICS 套接字域使用端口的 TCPIP SERVICE 资源定义来确定 CICS Web support 应该处理的请求。TCPIP SERVICE 定义指定应用于请求的安全属性, 指定接收请求消息的超时设置, 并且限制可为单个请求接收的最大数据量。
2. **CICS 将请求的 URL 与 URIMAP 定义匹配 (如果可用的话)。** CICS 尝试将 HTTP 请求中指定的 URL 与任何和 TCPIP SERVICE 定义相关的 URIMAP 资源定义匹

配，并应用到作为 HTTP 服务器的 CICS。如果进行了成功的匹配，则 URIMAP 定义将告诉 CICS 如何处理请求。如果未找到匹配，CICS 将继续缺省过程，该过程由分析器程序从处理阶段 5 开始。

3. 如果 URIMAP 定义指定重定向，则 CICS 将 Web 客户机重定向到指定的 URL。CICS 组成重定向消息并将它传输到 Web 客户机。这将完成对该 HTTP 请求的处理。
4. 如果 URIMAP 定义指定静态响应，则 CICS 形成并提供该响应。CICS 将文档模板或 z/OS UNIX 系统服务 HFS 文件与适当的 HTTP 头协同使用，以形成 HTTP 响应。该响应经历适当的代码页转换，然后 CICS 将响应传输到 Web 客户机。这将完成对该 HTTP 请求的处理。
5. 如果 URIMAP 定义指定了分析器程序的使用，或者没有找到匹配的 URIMAP 定义，则分析器程序可以运行。分析器程序可以动态解释请求，或者它可用于监控或审计目的。

如果还没有为请求设置 URIMAP 定义，则必须在请求处理路径中使用 TCPIP SERVICE 定义的分析器程序。如果您将具有特殊需求的、不是支持 Web 的应用程序用于代码页转换或 CICS TS V3 之前的兼容性处理，则可能也需要此分析器程序。（第 91 页的第 10 章，『分析器程序』说明这些情况。）另外，分析器程序是可选的，但要注意的是，如果未找到 URIMAP 定义，则调用分析器程序来处理该请求。

如果正在使用分析器程序，则将 HTTP 请求和 HTTP 头传递给分析器程序。分析器程序可以解释请求以确定：

- 要使用哪些 CICS 资源对请求提供服务。
 - 哪个用户标识将与请求关联。
 - 需要哪些剩余处理阶段。
6. 转换器程序可以用于对请求进行解码，并为应用程序构造输入。支持 Web 的应用程序应该接受 HTTP 请求而不进行任何解码。然而，如果要使用需要通信区域输入的不支持 Web 的应用程序来对 HTTP 请求提供服务，则可以使用转换器程序来解码请求，并构造符合您的应用程序需求的输入。可以使用 URIMAP 定义来指定转换器程序，或者可以由分析器程序来选择它。
 7. 执行应用程序以对请求提供服务。您可以使用 URIMAP 定义或使用分析器程序来指定应用程序。使用 EXEC CICS WEB 和 EXEC CICS DOCUMENT 应用程序编程接口的支持 Web 的应用程序可用于处理请求和构造。可以使用转换器程序（它将 Web 客户机的请求转换为可接受的输入并且根据程序的输出构成 HTTP 响应），或者使用调用不支持 Web 的程序并使用其输出的支持 Web 的应用程序，为 Web 启用不支持 Web 的应用程序。

在别名事务下运行的应用程序。

应用程序可以执行以下任务：

- 如果应用程序支持 Web，则它可以检查请求的 HTTP 头，从请求行抽取信息（例如，查询字符串），将请求主体接收到缓冲区中以供处理，为响应的状态行选择状态码和文本，并写响应的 HTTP 头。EXEC CICS WEB API 命令（如 WEB SEND 和 WEB WRITE HTTPHEADER）用于构造响应。

- 不管应用程序是否支持 Web，它都可以产生构成响应的主体的输出。支持 Web 的应用程序可以产生从 CICS 文档模板或从一缓冲区数据形成的实体主体。不支持 Web 的应用程序可以产生输出，该输出可以由支持 Web 的应用程序或转换器程序转换为实体主体。
8. **转换器程序可以用于编码来自应用程序的输出并构造 HTTP 响应。**如果应用程序不支持 Web，且它的输出不是发送到 Web 客户机的正确格式，则可以使用转换器程序以产生包括状态行和 HTTP 头的适当 HTTP 响应。转换器程序还可以在需要时对输出执行其他类型的处理。

转换器程序可以指定应该重复处理阶段 6（使用转换器程序解码或进行其他处理）、7（应用程序）和 8（使用转换器程序编码或进行其他处理）。因为转换器程序可以更改应用程序的名称，所以您可以使用此设施以允许多个应用程序依次处理同一请求，并提供单个响应。

9. **如果 CICS Web support 进程中发生请求错误或者异常终止，则会发送一个错误响应给 Web 客户机，该响应可以使用用户可替代的 Web 出错程序进行定制。**DFHWBEP 或 DFHWBERX 接收有关错误情况的信息，以及 CICS 计划发送给 Web 客户机的缺省 HTTP 响应（包含状态码和状态文本）。用户可替代的程序可以定制响应或构建新的响应，并且将它返回给 CICS 以进行发送。

不是在所有的错误情况中都会使用 Web 出错程序。当请求的初始处理中出现问题，且后继处理中发生异常终止或故障时会使用它们。在处理（例如用户编写的应用程序执行的）正确地处理以及发生预期的错误或重定向响应的情况下，不使用 Web 出错程序。

10. **CICS 生成某些必需的 HTTP 头并将它们添加到消息。**根据响应的 HTTP 版本生成适当的头。如果响应是 HTTP/1.1 的，则 CICS 添加 HTTP/1.1 消息需要的头。如果响应是 HTTP/1.0 版本，则如果客户机已请求持续连接，CICS 将添加 Connection: Keep-Alive 头和少数其他头。部分这些头的值由 CICS 直接生成（例如，Date 头），而其他头的值基于支持 Web 的应用程序（使用 WEB SEND 命令）或 URIMAP 定义提供的信息。这些头可以添加到支持 Web 的应用程序的输出，也可以添加到转换器程序的输出。
11. **CICS 将完整的 HTTP 响应传输到 Web 客户机。**如果 Web 客户机支持持续连接，则 CICS 保持连接打开以获取更多可能的 HTTP 请求，直到用户应用程序或 Web 客户机请求关闭或达到超时值。

在此过程期间，当消息进入和离开 CICS 环境时通常需要代码页转换，因此 CICS Web support 处理和用户编写的应用程序（它们通常使用 EBCDIC 编码）可以与 Web 客户机（它们通常使用 ASCII 编码）通信。第 32 页的『CICS Web support 的代码页转换』说明何时和如何发生这种情况。可以在 WEB SEND 命令或 WEB RECEIVE 命令中使用一些选项来指定需要的代码页转换的类型。

作为 HTTP 客户机的 CICS 的 HTTP 请求和响应处理

对于作为 HTTP 客户机的 CICS，CICS 是 Web 客户机，并且它与 HTTP 服务器通信。用户编写的应用程序通过 CICS 将请求发送到 HTTP 服务器，并接收来自它的响应。CICS 维护与服务器的持续连接。在此应用程序发出的命令中使用会话令牌来识别连接。

| 发出 HTTP 请求并接收响应的应用程序必须使用 EXEC CICS WEB API 命令以明确
| 地指示与服务器进行交互操作。可以使用支持 Web 的应用程序发出 HTTP 请求，然后
| 处理结果以将信息提供给不支持 Web 的应用程序。

| 应将启动 HTTP 请求的应用程序设计为处理 CICS 从服务器接收的对该请求的任何一
| 种响应，这包括错误响应、到另一个 URL 的重定向、嵌入式超文本链接、HTML 表
| 单、图象源或从应用程序请求操作的其他项。如果需要，CICS 可以为请求和响应执行代
| 码页转换。

| 第 26 页的图 3 显示此主题中描述的过程。
|

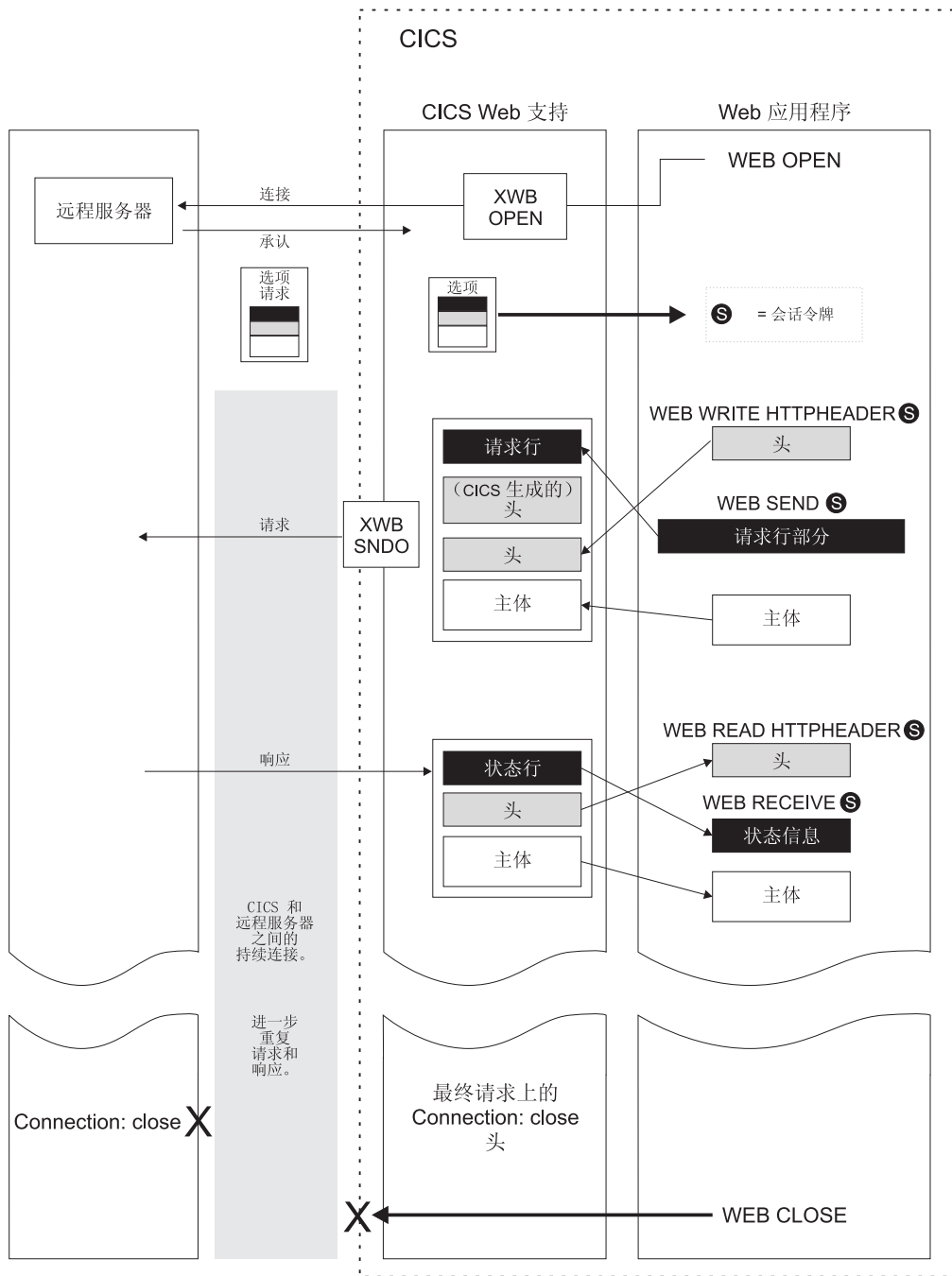


图3. 作为 HTTP 客户机的 CICS 的处理

作为 HTTP 客户机的 CICS 的处理过程如下所示:

1. 应用程序通过 **CICS** 启动与 **HTTP** 服务器的连接。应用程序通过发出 EXEC CICS WEB OPEN 命令来执行该操作。可引用您已创建的 URIMAP 资源定义为此连接指定规则和主机名，或者应用程序可以提供此信息。（*CICS Resource Definition Guide* 具有有关 URIMAP 定义的更多信息。）应用程序每次可以打开多个连接。
2. **CICS** 建立与服务器之间的连接。使用应用程序提供的信息，CICS 在套接字上打开 TCP/IP 连接并联系服务器。在此过程中，如果需要，可使用 XWBOPEN 用户出口（如果已使用 ENABLE PROGRAM 命令激活它）通过代理服务器重定向应用程序的请求，并将安全策略应用到主机的连接。当打开连接时，CICS 查询服务器的

HTTP 版本，并将此信息用于随后的通信。CICS 还将 HTTP 版本信息返回给应用程序，以供您计划写 HTTP 头或发送分块的消息时检查。当建立 TCP/IP 连接时，CICS 将会话令牌返回给应用程序以唯一地识别连接。此会话令牌在有关此连接的应用程序发出的所有剩余命令中使用。『会话令牌』说明有关会话令牌的更多信息。

3. 应用程序可为它的请求编写 **HTTP 头**。可使用 `WEB WRITE HTTPHEADER` 命令构建用户编写的 HTTP 头并存储以备发送。
4. 应用程序指定请求行的组成部分。请求方法、路径信息和查询字符串是使用 `WEB SEND` 或 `WEB CONVERSE` 命令指定的。请求的 HTTP 版本由 CICS 提供。
5. 应用程序可为它的请求产生实体主体。请求的内容在 `WEB SEND` 或 `WEB CONVERSE` 命令中指定。它可以从 CICS 文档（使用 `DOCUMENT` 接口）形成或从缓冲区的内容形成。如果服务器是 HTTP/1.1，则可将分块的传输编码用于从缓冲区的内容形成的请求主体（但不用于 CICS 文档）。
6. 应用程序启动请求的传输。当应用程序发出 `WEB SEND` 或 `WEB CONVERSE` 命令时，请求被传递到 CICS 以跨会话令牌指定的连接进行发送。
7. **CICS 生成某些必需的 HTTP 头并将它们添加到请求，然后将请求发送到服务器。** 某些头的值直接由 CICS 生成（如 `Date` 头），而其他头的值基于应用程序（使用 `WEB SEND` 命令）或 `URIMAP` 定义提供的信息。发送请求期间，如果需要，会调用 `XWBSNDO` 用户出口为个别请求应用安全策略。
8. 服务器接收请求并进行处理，而且提供响应。CICS 将响应传递给应用程序。
9. 应用程序会检查响应。可使用 `WEB READ HTTPHEADER` 命令或 `HTTP` 头浏览命令检查响应的头。`WEB RECEIVE` 或 `WEB CONVERSE` 命令接收可由应用程序处理的响应的主体（如果有），以及响应的状态码和状态文本。
10. 应用程序会启动更多的请求和响应。如果服务器支持持续连接，则通过会话令牌识别的连接为更多的请求保持打开状态。
11. 应用程序启动服务器连接的关闭。当完成所有请求和响应时，应用程序发出 `WEB CLOSE` 命令，且 CICS 关闭它的 TCP/IP 连接端。如果应用程序不发出 `WEB CLOSE` 命令，则在任务结束时关闭连接。

在此过程中，当消息进入和离开 CICS 环境时通常需要代码页转换，以便 CICS Web support 处理和用户编写的应用程序（通常使用 EBCDIC 编码）可以与 HTTP 服务器（通常使用 ASCII 编码）通信。第 32 页的『CICS Web support 的代码页转换』说明何时和如何发生这种情况。可使用 `WEB SEND`、`WEB RECEIVE` 或 `WEB CONVERSE` 命令中的选项指定需要的代码页转换的类型。对于作为 HTTP 客户机的 CICS，缺省情况是当发送和接收消息时进行代码页转换。

会话令牌

会话令牌是唯一地识别作为 HTTP 客户机的 CICS 和 HTTP 服务器之间的连接的 8 字节二进制值。每个连接使用会话令牌意味着 CICS Web support 可以通过不同的任务管理到服务器的多个连接，并也意味着应用程序可以控制多个连接。

连接以响应用户应用程序发出的 `WEB OPEN` 命令开始。成功完成 `WEB OPEN` 命令后返回会话标记，并且在与该连接相关的应用程序发出的所有 `EXEC CICS` 命令中使用该会话标记。

使用此连接，用户应用程序可向服务器发出 HTTP 客户机请求并接收来自它的响应。此连接可在多个请求和响应的交换期间持久，直到应用程序或服务器选择终止此连接。

第 31 页的

第 31 页的『CICS Web support 如何处理持续连接』中有关于 CICS Web support 如何处理持续连接以及如何终止它们的更多详细信息。

如果服务器终止连接，则应用程序无法使用此连接发送任何进一步的连接，但它可以读服务器在终止连接之前发送的响应。应用程序发出 WEB CLOSE 命令之前，会话令牌都保持有效以用于访问此数据的命令。发出 WEB CLOSE 命令之后，应用到连接的会话令牌不再有效。如果应用程序不发出 WEB CLOSE 命令，则在任务结束时关闭连接。

最大打开客户机连接的数（每一个由一个会话令牌表示），可以在一个 CICS 区域中同时出现的是 32768 个。

CICS Web support 的 URL

在 CICS Web support 提供的资源的请求 URL 中，URL 的路径部分由您确定。在 CICS Web support 中，URIMAP 定义或分析器程序在 CICS 提供的请求 URL 和资源之间创建链接，因此 URL 不需要与 CICS 资源有任何直接关系。然而，您可以设计 URL 以提供用于处理或管理目的的信息。

设计路径部分时，请记住以下注意事项：

- 如果用户需要手工输入 URL，则该 URL 应该尽可能简单和有含义。
- CICS Web support 的 URL 可以与 CICS 资源有逻辑关系，您可能为了以下目的而设计它：
 - 对于应用程序生成的响应，将信息提供给应用程序。
 - 对于静态响应，提供路径匹配的信息，或只需使资源管理更直接（特别是在使用 HFS 文件的地方）。虽然这对于 CICS 不是必需的，但对于 URL 的路径部分，将目录路径与服务器上文件的真实位置匹配是一个标准的做法。
- 对于使用 URIMAP 定义管理的应用程序生成的响应，可以创建具有通配符的单个 URIMAP 定义，以将路径以同样方式开头的请求 URL 映射到单个资源。例如，路径以 /staffapps/ordering/ 开头的请求可以映射到特殊 CICS 应用程序。
- 对于使用分析器程序管理的应用程序生成的响应，可以设计一个 URL，该 URL 提供有关要在处理中使用的程序和事务的信息。CICS 提供的分析器程序 DFHWBADX 分析路径部分格式为 /converter/alias/program/other path information 的 URL，其中 converter 命名转换器程序（如果存在），alias 命名别名事务，program 命名用户应用程序，而 other path information 提供分析器未使用的其他信息。
- 对于应用程序生成的响应，可以设计 URL 的路径部分以对应用程序提供处理信息。例如，路径部分可用于指定应用程序提供的特殊函数。如果支持 Web 的应用程序正在对多个其他应用程序提供前端，则 URL 的路径部分可以识别应用请求的应用程序。使用 WEB EXTRACT 命令，路径部分中的信息可以由应用程序本身抽取，以确定适当的操作。
- 对于静态响应，可以创建具有通配符的单个 URIMAP 定义，以使用路径匹配机制将路径以同样方式开头的请求 URL 映射到各种静态文档。
- 请求 URL 中的查询字符串可用于选择替换的 URIMAP 定义。要在 URIMAP 匹配中使用查询字符串，必须在 URIMAP 定义的路径属性中指定完整和确切的查询字符串以及路径本身。对于应用程序生成的响应，该应用程序可以随后使用该查询字符串。然而，对于带文档模板的静态响应，用于 URIMAP 匹配的查询字符串不作为符号列表传递到文档模板。

CICS Web support 对 URL 长度有以下限制:

- 对于进站 HTTP 请求的 URL (对于作为 HTTP 服务器的 CICS), CICS 接受的长度可达 32K。(此长度至少是常用 Web 浏览器客户机支持的长度的八倍。)如果 CICS 接收的 URL 的长度超过它所能处理的长度,则它将返回 414 (请求 URI 太长) 状态码。
- 对于出站 HTTP 请求的 URL (该请求是通过作为 HTTP 客户机的 CICS 发出的), CICS 支持 URIMAP 资源定义中多达 255 个字符的路径部分。发出请求的用户应用程序可能覆盖 URIMAP 定义 (或根本就一个也不使用), 并提供更长的路径部分。检查可以由服务器处理的 URL 长度。

为作为 HTTP 服务器的 CICS 提供的资源选择 URL 时, 请注意以下对 URL 长度的其他限制:

- CICS 支持 URIMAP 资源定义中多达 255 个字符的路径部分。不要尝试使用比这个长度长的路径部分。HTTP/1.1 规范声明: 对于总长度 (由规则、主机和路径部分以及定界符组成) 超过 255 个字符的 URL, 服务器应该谨慎, 这是因为较旧的 Web 客户机和代理可能不会正确支持这些 URL。
- 如果需要使用更长的路径部分, 则通常可以这么做, 因为您不必在 URIMAP 资源定义中指定完整路径。星号 (*) 可用作路径末尾的通配符。如果满足以下条件, 则 URIMAP 定义的行为将是正确的:
 - URL 的指定部分对于该资源来说是唯一的。
 - URL 的指定部分对于该资源来说不是唯一的, 但是您正在提供静态响应, 且正在使用路径匹配机制以使 URL 完整 (其中在路径部分和提供响应的资源名中都使用通配符)。
- 如果您为了 URIMAP 匹配目的而使用查询字符串, 并在 URIMAP 定义的路径属性中指定它, 则总长度必须仍然限制为 255 个字符。(如果该行为仍然正确, 则路径的部分可以由星号替换。)如果没有为此目的使用查询字符串, 则可以接受任何长度的查询字符串, 最长为 CICS Web support 的 URL 长度的 32K 限制。
- 对于重定向 (使用 URIMAP 定义中的 LOCATION 和 REDIRECTTYPE 属性), CICS 支持多达 255 个字符的重定向 URL。此 URL 必须是完整的 URL, 包括规则、主机和路径部分以及合适的定界符。如果计划将资源用作已重定向客户机的目标, 则确保其完整 URL 符合此 255 个字符的限制。

CICS Web support 如何处理分块的传输编码

使用分块的传输编码的消息可以由 CICS 发送和接收。

作为 HTTP 服务器的 CICS 可以将分块的消息作为请求接收, 或将分块的消息作为响应发送。作为 HTTP 客户机的 CICS 可以将分块的消息作为请求发送, 或将分块的消息作为响应接收。CICS Web support 处理这些不同情况, 具体如下:

- 当作为 HTTP 服务器的 CICS 将分块的消息作为 HTTP 请求接收时, CICS Web support 会识别分块的编码。它将等待接收完所有块 (由接收到零长度的块来表明), 并组装这些块以形成完整的消息。用户应用程序可以使用 WEB RECEIVE 命令接收组装的消息体。
 - 使用 TCPIP SERVICE 资源定义上的 MAXDATALEN 选项 (该资源定义与请求到达的端口相关), 您可以限制 CICS 为单个分块消息接受的总数据量。

- 当 CICS 是 HTTP 服务器时，接收分块消息的超时值由 TCPIP SERVICE 定义的 SOCKETCLOSE 属性设置。
- 可以使用 HTTP 头命令读取分块消息的尾部头。尾部头识别作为尾部头出现的头的名称。如果正在使用请求的处理路径中的分析器程序，则请注意，尾部头不与主请求头一起传递到分析器程序。
- 当作为 HTTP 客户机的 CICS 将分块的消息作为对应用程序请求的响应接收时，在将这些块作为实体主体传递到应用程序之前，会先组装它们，而任何尾部头可以使用 HTTP 头命令读取。使用事务标识（与应用程序相关的）的事务概要文件定义的 RTIMOUT 属性，可以指定应用程序将等待多久以接收响应。
- CICS 作为 HTTP 服务器（响应）或作为 HTTP 客户机（请求）发送分块消息时，通过 WEB SEND 命令中使用的 CHUNKING(CHUNKYES) 选项，应用程序可以为每个消息块指定分块的传输编码。可以用对应用程序来说最方便的任何方式来分割消息。CICS 发送每个消息块，并且添加适当的 HTTP 头向接收方表明正在使用分块的传输编码。应用程序发出带 CHUNKING(CHUNKEND) 的 WEB SEND 以表明消息结束。然后，CICS 会发送一个空块（包含空白行）以及需要的任何尾部头以结束分块的消息。

第 67 页的『使用分块的传输编码发送 HTTP 请求或响应』说明从 CICS 发送 HTTP 消息时，用于分块的传输编码的过程。为了让接收方可以接受分块的消息，应该按此过程进行操作。

CICS Web support 如何处理管道传送

管道化的请求序列可以由 CICS 发送和接收。作为 HTTP 服务器的 CICS 可以从 Web 客户机接收管道化的请求序列，而作为 HTTP 客户机的 CICS 可以将管道化的请求序列发送到服务器。

CICS Web support 处理管道化的请求序列以及对它们的响应，具体如下：

- 当作为 HTTP 服务器的 CICS 接收管道化序列的 HTTP 请求时，将连续地处理这些请求。这是为了确保返回响应的顺序与发送请求的顺序相同。CICS 将管道化序列中的每条消息作为独立的事务对待，要么提供在 URIMAP 定义中指定的静态响应，要么将消息传递到应用程序并等待该应用程序产生响应。每个事务处理单个请求并提供响应。管道传送的消息序列中的剩余请求由 CICS 保留，直到发送了对先前请求的响应，然后才启动一个新的事务以处理每个进一步请求。
- 当作为 HTTP 客户机的 CICS 发送管道化的请求序列时，将自动启用管道传送。每个 HTTP 请求立即发送，因此应用程序可以在接收任何响应前发送多个 HTTP 请求。当管道化序列中的最后一条消息已发送后，应用程序可以开始接收响应。
- 当作为 HTTP 客户机的 CICS 接收到管道化的请求序列的 HTTP 响应时，响应返回到应用程序的顺序将是 CICS 从服务器接它们的顺序。支持管道传送的服务器以接收请求时的相同顺序提供响应。应用程序在完成发送其所有 HTTP 请求后开始接收响应。

对于作为 HTTP 客户机的 CICS，确保请求的任何管道化序列是幂等的是应用程序的职责。第 14 页的『管道传送』说明幂等性。为了应用程序逻辑的利益以及服务器的利益，如果您不确定请求序列是幂等的，则建议您发出独立的请求，并且等待对每个请求的响应，然后再发出下一个请求。

CICS Web support 如何处理持续连接

持续连接是 CICS Web support 的缺省行为。

在 CICS TS 3.1 前，连接行为是：已从 Web 客户机接收数据后，CICS 通常将关闭连接，除非 Web 客户机发送 Connection: Keep-Alive 头。

现在，在 Web 客户机和作为 HTTP 服务器的 CICS 之间进行连接后，或者在作为 HTTP 客户机的 CICS 和服务器之间进行连接后，缺省情况下，CICS 尝试保持连接打开。

当 CICS 是 HTTP 服务器时，在以下情况下，持续连接关闭：

- 处理请求的用户编写的应用程序关闭连接（通过在 WEB SEND 命令中指定 CLOSESTATUS(CLOSE) 选项）。
- Web 客户机关闭连接（由 Connection: close 头通知）。
- Web 客户机是不发送 Connection: Keep-Alive 头的 HTTP/1.0 客户机。
- 达到超时期限（表明连接已失败，或 Web 客户机已故意退出连接）。

否则，CICS 保持持续连接打开以使 Web 客户机发送更多请求。如果存在与客户机的持续连接，则通过 Web 出错程序发送错误响应后，CICS 保持连接打开。特例是 CICS 为响应选择 501（方法未实施）状态码，在这种情况下，CICS 关闭此连接。

使用 CICS Web support 的 TCPIP SERVICE 资源定义，TCPIP SERVICE 定义的 SOCKETCLOSE 属性不应该指定为零。SOCKETCLOSE 设置为零意味着作为 HTTP 服务器的 CICS 从 Web 客户机接收数据后立即关闭连接，除非有更多数据正在等待。这意味着无法维护持续连接。

当 CICS 是 HTTP 客户机时，在以下情况下，持续连接关闭：

- 服务器关闭连接（由发送 Connection: close 头的 HTTP/1.1 服务器通知或由无法发送 Connection: Keep-Alive 头的 HTTP/1.0 服务器通知）。
- 用户应用程序关闭连接（通过在 WEB SEND 或 WEB CONVERSE 命令中指定 CLOSESTATUS(CLOSE) 选项，或通过发出 WEB CLOSE 命令）。
- 任务已结束而连接尚未关闭。

如果应用程序需要测试服务器是否已请求终止连接，则可以使用 READ HTTPHEADER 命令在来自服务器的最后一条消息中查找 Connection: close 头。如果服务器请求关闭连接，但是应用程序尚未发出 WEB CLOSE 命令，则 CICS 关闭连接，但是仍将维护与连接有关的数据（包括从服务器接收的最后一条响应及其 HTTP 头）。应用程序可以继续使用该数据，直到它发出 WEB CLOSE 命令或任务结束为止。

作为 HTTP 客户机的 CICS 的 WEB CLOSE 命令不导致 CICS 通知服务器应该终止持续连接。它只是让 CICS 关闭连接。好的做法是在发出给服务器的最后一个请求上包含 Connection: close 头。使用此头意味着服务器在发送最后一条响应后可以立即关闭其持续连接，而不是等着看 CICS 是否发送更多请求，然后超时。要包含此头，在 WEB SEND 或 WEB CONVERSE 命令中指定 CLOSESTATUS(CLOSE) 选项。

如果作为 HTTP 客户机的 CICS 正在与 HTTP/1.0 服务器通信，则 CICS 自动在 HTTP 消息上发送 Connection: Keep-Alive 头。请求连接的应用程序不需要提供这些。Keep-Alive 通知服务器需要持续连接。

CICS Web support 的代码页转换

当 CICS 与 Web 客户机或服务器交换消息时，在进入和离开 CICS 环境时通常需要对这些消息中的字符数据进行代码页转换。

出于两个原因需要对这些消息中的文本进行代码页转换：

- CICS 和用于 CICS 的用户编写的应用程序通常使用 EBCDIC 编码，但是 Web 客户机和服务器通常使用 ASCII 编码。
- 在每个编码中，使用大量不同的代码页来支持本地语言。

消息的非文本内容（如图像或应用程序数据）不需要进行代码页转换。

在 CICS Transaction Server for z/OS V3R1 之前的 CICS 发行版中，CICS Web support 的代码页转换是使用代码页转换表（DFHCNV）处理的。在 CICS Transaction Server for z/OS V3R1 中，除了用于迁移目的的极少数情况，CICS Web support 不再需要代码页转换表。CICS Web support 使用 z/OS 转换服务处理代码页转换。

在 CICS Web support 中，文本代码页转换的缺省值如下：

- 缺省字符集是 ASCII Latin-1 字符集 ISO-8859-1。在 HTTP 消息中，所有请求或状态行和 HTTP 头都是 US-ASCII 字符集，它是 ISO-8859-1 较早的子集。包含文本的消息体通常是 ISO-8859-1。
- CICS 环境中数据的缺省 EBCDIC 代码页由 CICS 区域的 LOCALCCSID 系统初始化参数指定。LOCALCCSID 的缺省值是 EBCDIC Latin 字符集（代码页 037）。

有时可以标识更合适的备用代码页：

- Web 客户机或服务器可以在 Content-Type 头中为请求或响应指定字符集，它是已用于消息体的字符集。
- Web 客户机可以在请求上发送 Accept-Charset 头，以表明对于响应哪些字符集是可接受的。
- 对于非 HTTP 请求和某些较早的 HTTP 实施，传输消息时使用的字符集可能未在消息头中标识，并且您可能需要从您自己对消息源的了解来标识它。
- 如果缺省值不合适，则应用程序员需要标识他们的应用程序可用于接收消息数据的合适代码页。

CICS 不支持 IANA 命名的所有字符集。CICS 为进行代码页转换支持的 IANA 字符集在第 183 页的附录 A，『HTML 编码字符集』中列出。

在大多数情况下，消息的介质类型可以确定是否发生代码页转换。具有非文本介质类型的请求或响应主体通常不经历代码页转换。与支持 Web 的应用程序的兼容性的异常在先前发行版中编码；如果命令中使用的选项表明应用程序在 CICS Transaction Server for z/OS V3R1 前编码，则介质类型不影响代码页转换。

根据消息类型和处理路径，代码页转换信息可由 CICS 自动标识，或在 URIMAP 定义中指定，或由分析器程序指定，或在支持 Web 的应用程序发出的命令中指定。第 33 页的『作为 HTTP 服务器的 CICS 的代码页转换』说明作为 HTTP 服务器的 CICS 的过程，而第 34 页的『作为 HTTP 客户机的 CICS 的代码页转换』说明作为 HTTP 客户机的 CICS 的过程。

作为 HTTP 服务器的 CICS 的代码页转换

当作为 HTTP 服务器的 CICS 与 Web 客户机交换消息时，通常需要为消息体进行代码页转换。指定代码页转换的方法取决于您作出应用程序生成的响应还是静态响应，以及您使用支持 Web 的应用程序还是不支持 Web 的应用程序。

请求行和 HTTP 头

请求行或状态行的代码页转换、以及 HTTP 头的代码页转换按如下方式处理：

- 接收请求后不久，CICS 将请求行和 HTTP 头从 US-ASCII 字符集转换为 EBCDIC 代码页 037。请求行和头应该总是 US-ASCII。CICS 需要代码页 037 来进行内部处理。
- 支持 Web 的应用程序使用 WEB EXTRACT 和 WEB READ HTTPHEADER 命令从请求行和 HTTP 头抽取信息时，该信息以其转换的格式 EBCDIC 代码页 037 表示。
- 当 CICS 准备发出响应时，请求行和 HTTP 头可以由 CICS 生成，或由应用程序使用 WEB WRITE HTTPHEADER 命令指定。发送前，所有头和请求行从指定它们的 EBCDIC 代码页转换为 US-ASCII 字符集。

消息体：应用程序生成的响应

如果请求要从用户编写的应用程序获取动态响应，则消息体的代码页转换按如下方式处理：

- 如果支持 web 的应用程序接收请求，则在 WEB RECEIVE 命令中使用任何一个代码页转换选项指定转换的情况下，CICS 执行代码页转换。如果这些选项未出现，则不执行代码页转换。您可以提供字符集或允许 CICS 识别字符集，并在缺省代码页不适合的情况下请求代码页。
- 如果分析器程序用于请求的处理路径，则分析器程序可以为传递到通信区域中后继处理阶段的请求主体指定或抑制代码页转换。您提供所使用的字符集和应用程序代码。CICS 仍然保存请求主体的原始版本，并且使用 EXEC CICS WEB API 命令的支持 web 的应用程序访问原始主体而不是通信区域。
- 如果转换器程序需要通信区域中的请求，并且处理路径中没有分析器程序，则 CICS 转换通信区域中的请求主体，并识别字符集和转换成缺省代码页。
- 为了识别 Web 客户机用于请求主体的字符集，CICS 会检查请求头。如果请求头不提供此信息或指定的字符集不受支持，则 CICS 会假设字符集为缺省字符集，即消息体为 ISO-8859-1 字符集。如果消息体不是此字符集，且头中没有信息，则需要识别正确的字符集。
- 缺省情况下，CICS 将请求主体转换为 LOCALCCSID 系统初始化参数（它应用到整个本地 CICS 区域，且具有缺省值 037）指定的 EBCDIC 代码页。如果您的应用程序需要另一个代码页（可以是 EBCDIC 或 ASCII），则您可以指定该代码页。
- 如果支持 web 的应用程序发送响应，则在 WEB SEND 命令中使用任何一个代码页转换选项指定转换的情况下 CICS 执行代码页转换。如果这些选项未出现，则不执行代码页转换。注意，对于支持 web 的应用程序，不能将 CICS 文档和文档模板转换为 UTF-8 和 UTF-16 字符编码。如果您要发送这些字符编码的出站消息，则在 WEB SEND 命令中使用 FROM 选项指定形成消息体的数据缓冲区，而不使用 DOCTOKEN 选项指定 CICS 文档。

- 如果使用通信区域的转换器程序发送响应，则 CICS 镜像已为请求执行的代码页转换。使用分析器程序的字符集和主机代码页设置，或缺少分析器程序时使用缺省设置。如果分析器程序抑制了请求的代码页转换，则不会为响应主体执行代码页转换。

消息体：静态响应

如果请求要获取由 URIMAP 定义确定的静态响应，则消息体的代码页转换按如下方式处理：

- 对于静态响应，CICS 不检查出现在 Web 客户机的请求中的任何消息体，因此不需要代码页转换。
- 您在产生静态响应的 URIMAP 定义中为响应主体指定代码页转换。如果响应包含文本，则 URIMAP 定义需要指定所有以下几项：
 - 文本介质类型，使用 MEDIATYPE 属性。此属性没有缺省值。
 - Web 客户机的字符集，使用 CHARACTERSET 属性。（注意，不能将 CICS 文档和文档模板转换为 UTF-8 和 UTF-16 字符编码。）
 - 对响应的 CICS 文档模板或 HFS 文件进行编码所用的代码页，使用 HOSTCODEPAGE 属性。

CICS 检索 HFS 文件，或检索 CICS 文档模板并创建文档，然后执行适当的代码页转换。

作为 HTTP 客户机的 CICS 的代码页转换

当作为 HTTP 客户机的 CICS 与服务器交换消息时，通常需要为消息体进行代码页转换。打开连接时指定应用程序代码。通常字符集可以由 CICS 识别或允许使用缺省值。

请求行和 HTTP 头

请求行或状态行的代码页转换、以及 HTTP 头的代码页转换按如下方式处理：

- 当 CICS 准备发出请求时，请求行和 HTTP 头可以由 CICS 生成，或由应用程序使用 WEB WRITE HTTPHEADER 命令指定。发送前，所有头从指定它们的 EBCDIC 代码页转换为 US-ASCII 字符集。
- 接收响应后不久，CICS 将状态行和 HTTP 头从 US-ASCII 字符集转换成 EBCDIC 代码页 037。应用程序接收状态行和其他信息，并检查 EBCDIC 代码页 037（HTTP 头的转换格式）中的 HTTP 头。

消息体

消息体的代码页转换按如下方式处理：

- 由应用程序使用的 EBCDIC 代码页是在启动与服务器通信的 WEB OPEN 命令上指定的。缺省值是由 LOCALCCSID 系统初始化参数（它应用到整个本地 CICS 区域，且具有缺省值 037）指定的 EBCDIC 代码页。CICS 使用此信息在此连接上转换请求和响应的消息体。
- 对于应用程序发出的每个请求，WEB SEND 或 WEB CONVERSE 命令中的 CLIENTCONV 选项指定 CICS 是否为请求主体执行代码页转换。缺省情况是执行代码页转换。如果正在使用 WEB CONVERSE 命令，则可以为请求主体和 / 或响应主体指定代码页转换，或者都不指定。

- 如果您已为请求指定转换，则缺省情况是 CICS 将请求主体转换成 ISO-8859-1 字符集。如果您知道服务器更希望使用另一个代码页，则可以使用 WEB SEND 或 WEB CONVERSE 命令的 CHARACTERSET 选项来选择另一个代码页。
- 对于应用程序接收的每个响应，WEB RECEIVE 或 WEB CONVERSE 命令的 CLIENTCONV 选项指定 CICS 是否为响应主体执行代码页转换，以将它转换为打开连接时指定的 EBCDIC 代码页。缺省情况是执行代码页转换。CICS 检查响应头以识别服务器用于响应主体的字符集。如果响应头不提供此信息或指定的字符集不受支持，则 CICS 会假设字符集为缺省字符集，即消息体为 ISO-8859-1 字符集。

作为 HTTP 服务器的 CICS 的 HTTP/1.1 一致性

CICS Web support 现在支持 HTTP/1.1。

CICS Transaction Server for z/OS V3R1 前的 CICS 发行版支持 HTTP/1.0。现在增强 CICS Web support 以处理和提供 HTTP/1.1 规范的功能，包括分块传送编码、管道传送和持续连接。

正如因特网协会和 IETF（因特网工程任务组织）请求评论 RFC 2616 *Hypertext Transfer Protocol - HTTP/1.1* (<http://www.ietf.org/rfc/rfc2616.txt>) 中所描述，CICS Web support 有条件地遵从 HTTP/1.1 规范。

有条件地遵从 HTTP/1.1 规范意味着 CICS 满足所有“MUST”级别需求，但并不满足所有的 SHOULD 级别需求（这在 HTTP/1.1 规范中有详细的描述），在那些地方这些需求与 CICS 自身实际提供的功能相等。满足其协议的所有 MUST 或 REQUIRED 级别和所有 SHOULD 级别需求的实施称为无条件地遵从。如果实施无法满足它实施的协议的一个或多个 MUST 或 REQUIRED 级别需求，则该实施是不遵从的。

遵从 HTTP/1.1 规范的 CICS Web support 有三个方面。

- CICS Web support 执行 HTTP 服务器需要的操作。例如，CICS Web support 接收入站请求、维护持久连接、写特定 HTTP 头并传输响应。这些操作期间的 CICS Web support 的行为有条件地遵从 HTTP/1.1 规范。必要时，这表示从 CICS 的以前发行版在行为方面所作的更改。第 36 页的『遵从 HTTP/1.1 的 CICS Web support 行为』描述 CICS Web support 的行为如何遵从 HTTP/1.1，以及它在哪些方面与 CICS Transaction Server for z/OS V3R1 前的 CICS 发行版不同。
- HTTP/1.1 规范的某些部分与 CICS Web support 无关。例如，CICS Web support 不作为代理服务器，也不提供高速缓存工具。第 37 页的『CICS Web support 不支持 HTTP 功能』记录了这些区域，因此您可以知道在哪些区域中 HTTP/1.1 一致性对于 CICS Web support 且对于您的用户应用程序无关紧要。
- CICS 中的支持 Web 的用户应用程序可用于创建应用程序生成的 HTTP 响应，并指示 CICS Web support 如何对响应提供服务。如果您希望 CICS Web support 实施遵从 HTTP 协议规范（特别是 HTTP/1.1），则您的用户应用程序共同承担责任，以保证它们执行的操作的一致性。本文档中提供某些基本指导信息，但是检查您正在遵从的 HTTP 规范以获取更多详细信息很重要。第 57 页的第 6 章，『为作为 HTTP 服务器的 CICS 编写支持 Web 的应用程序』描述为 CICS Web support 编写用户应用程序的过程。

在实践方面，HTTP/1.1 规范中的不同需求级别（MUST、SHOULD 或 MAY）应该由您的应用程序处理，具体如下：

- 必须实施 MUST 级别需求以维护一致性。CICS Web support 设计为处理或协助您遵从所有 MUST 级别需求，这些需求应用到直接的活动。如果选择满足 CICS Web support 尚未处理的可选需求，则可能应用某些其他 MUST 级别需求。而且，某些 MUST 级别需求与您的应用程序在特定情况下必须不执行的操作有关。
- SHOULD 级别需求对于有条件地遵从 HTTP 规范不是必需的。不遵从 HTTP/1.1 规范中的所有 SHOULD 级别需求。然而，如果您的应用程序可以遵从 SHOULD 级别需求而没有什么不方便，则建议您这么做。
- MAY 操作对于任何 HTTP 实施是可选的（不管它的一致性级别如何）。应该将它们作为建议或推荐来对待。

遵从 HTTP/1.1 的 CICS Web support 行为

为了您的利益，CICS Web support 遵从 HTTP/1.1 规范中的许多要求。

不管您正在使用 URIMAP 定义还是分析器程序处理作为 HTTP 服务器的 CICS 的 HTTP 请求，都将应用此处描述的大多数操作，但有一些例外。如果您未在使用 URIMAP 定义，则应该注意，可能需要在您的分析器程序的行为中进行某些更改，以确保 HTTP/1.1 一致性。

CICS TS V3 的新行为

- **CICS 检查入站消息是否遵从 HTTP/1.1，并处理或拒绝不遵从此规范的消息。**在涉及 URIMAP 定义或分析器程序前，接收请求时立即进行检查。会进行各种基本验收检查，如果消息不可接受且由 CICS 自己处理问题也不合适，则可能的话会将错误响应返回给 Web 客户机。不会为 HTTP/1.0 消息执行这些基本验收检查，如果在 TCPIP SERVICE 定义中指定了 USER 协议（而不是 HTTP 协议），也不会执行基本验收检查。

注：CICS 要求在具有消息体的所有入站 HTTP/1.1 消息中都有 Content-Length 头。

如果有消息体但未提供头，或头的值不正确，则错误消息或后继消息的套接字接收会产生不可预测的结果。对于具有消息体的 HTTP/1.0 消息，Content-Length 头是可选的。

- **CICS 按照 HTTP/1.1 规则来比较 URL。**规则名和主机名比较不区分大小写，但是路径比较区分大小写。比较之前 URL 的所有组成部分都未转义。CICS 还处理请求（其中主机名在请求行中指定）中的绝对 URI 格式。注意，当使用分析器程序而不是 URIMAP 定义处理入站 HTTP 请求时，如果您需要达到这方面的一致性，则必须对分析器程序进行编码以按照 HTTP/1.1 规范中声明的规则来执行 URL 比较。（请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。）
- **CICS 在出站消息的头一行提供适合的 HTTP 版本号。**除非 CICS 知道 Web 客户机或服务器为 HTTP/1.0 级别，否则它通常将消息识别为 HTTP/1.1。在这种情况下，CICS 将消息识别为 HTTP/1.0。CICS 不支持 Web 客户机从一个 HTTP 版本升级到另一个 HTTP 版本或不同安全协议的请求。
- **在出站 HTTP/1.1 消息中，CICS 提供通常应该出现的 HTTP 头，以使消息遵从 HTTP/1.1。**CICS 还产生一些头，实现一致性并不需要它们，但是它们支持应用程序已请求的操作（例如，Expect 头）。第 185 页的附录 B，『CICS Web support 的 HTTP 头参考』描述 CICS 编写的头以及创建这些头的情况。这些一致性头提供给由支持 web 的应用程序和不支持 web 的应用程序发送的消息。如果为 TCPIP SERVICE 定义指定 USER 协议（而不是 HTTP 协议），则不提供它们。对于 HTTP/1.0 消息，只提供 Connection: Keep-Alive、Content-Length、Date 和 Server 头。

- **CICS 对入站请求和出站请求的 Expect 头执行操作。** 当作为 HTTP 服务器的 CICS 接收具有 Expect 头的请求时，它将 100-Continue 响应发送给 Web 客户机，并等待剩余请求。对于作为 HTTP 客户机的 CICS，可以使用 WEB SEND 命令中的 EXPECT 选项使 CICS 将 Expect 头发送到服务器，并在发送请求前等待 100-Continue 响应。如果服务器返回另一个响应，则 CICS 通知应用程序并取消发送。
- **CICS 处理来自 Web 客户机的 OPTIONS 请求，并做出适当的响应。** OPTIONS *（整个服务器，而不是特定资源上的查询）是唯一接受的格式。响应包含有关作为 HTTP 服务器的 CICS 的基本信息（HTTP 版本和服务器软件描述）。未涉及用户应用程序。当 CICS 是 HTTP 客户机时，它使用 OPTIONS 请求来检查服务器的 HTTP 版本，并在 WEB OPEN 命令中返回此信息。
- **CICS 处理来自 Web 客户机的 TRACE 请求，并做出适当的响应。** 当 CICS Web support 接收具有 TRACE 方法的正确格式的请求时，它返回包含请求及其原始头以及它获取的任何头（例如，via 头字段）的响应。未涉及用户应用程序。
- **CICS 接受具有分块的传输编码的入站消息，并进行汇编，且支持您使用分块的传输编码以发送出站消息。** 可通过 HTTP 头读、写和浏览命令来操纵分块消息的尾部头。这意味着应用程序可将分块的消息当作正常消息来处理。CICS 还支持从用户应用程序发送分块的消息，但您必须确保您按照正确的过程以使分块的消息被接收方所接受。第 29 页的『CICS Web support 如何处理分块的传输编码』说明这方面的 CICS Web support 行为。
- **CICS 支持入站和出站消息的管道传送。** CICS 允许您从 Web 客户机接收管道化的请求。CICS 依次将每个请求传递到应用程序，通过按接收请求的顺序来响应它们，以确保应用程序遵从 HTTP/1.1。CICS 还允许您将管道化的请求发送到服务器，但是您必须确保按正确的过程进行操作以使管道化的请求序列遵从 HTTP/1.1。第 30 页的『CICS Web support 如何处理管道传送』说明这方面的 CICS Web support 行为。
- **CICS 支持虚托管（同一 IP 地址的多个主机名）。** 对虚拟主机的支持基于您的 URIMAP 定义。第 6 页的『虚托管』说明所提供的支持。

已更改的行为，与 CICS TS V2 进行比较

- **缺省情况下，连接现在是持续的。** 对于作为 HTTP 服务器的 CICS 和作为 HTTP 客户机的 CICS 都是这种情况。如果 CICS 正在与 HTTP/1.1 级别的 Web 客户机或服务器通信，则它保持连接打开，除非 CICS 中的 Web 客户机、服务器或用户应用程序特别请求关闭或任务结束。如果 CICS 正在与 HTTP/1.0 级别的 Web 客户机或服务器通信，则当支持持续连接时，它发送 Connection: Keep-Alive 头。第 31 页的『CICS Web support 如何处理持续连接』说明这方面的 CICS Web support 行为，以及它与先前的 CICS 发行版的区别。
- **CICS 处理范围更大的错误情况和不受支持的消息** CICS Web support 是为了以下目的而设计的：针对许多错误情况，或入站消息可能导致用户应用程序出现问题的情况而做出反应。提供了更多的错误响应，这适合 HTTP/1.1 或 HTTP/1.0 客户机的需求。第 191 页的附录 C，『CICS Web support 的 HTTP 状态码参考』说明 CICS 向 Web 客户机提供响应的情况。可以修改用户可替代的 Web 出错程序以修改 CICS 提供的响应。第 85 页的第 9 章，『Web 出错程序』有更多信息。

CICS Web support 不支持 HTTP 功能

HTTP/1.1 规范为使用 HTTP 协议的各方定义各种角色。CICS Web support 执行适合于原始服务器、客户机和用户代理（尽管用户可能不涉及每一个 HTTP 客户机请求）的

许多功能。HTTP/1.1 规范还包括与代理、网关、隧道和高速缓存相关的需求，而这些需求与 CICS Web support 不相关，可以忽略它们。

- **CICS 不担当代理。** 您可以忽略与代理行为有关的所有 HTTP/1.1 规范中的需求。
- **CICS 不担当网关（另一台服务器的调解者）或隧道（HTTP 连接间的中继设备）。** 您可以忽略与网关和隧道行为有关的所有 HTTP/1.1 规范中的需求。
- **CICS 不提供高速缓存工具，也不提供对用户编写的高速缓存工具的支持。** 您可以忽略与高速缓存行为有关的所有 HTTP/1.1 规范中的需求。虽然您可以存储从服务器接收的任何信息，但是应该小心：不要将存储的信息交付到正在发出请求的用户，该请求期望从服务器接收当前信息。
- **CICS 不旨在用作 Web 浏览器。** 通过作为 HTTP 客户机的 CICS，用户应用程序可以为服务器提供的个别、已知资源发出请求，但是浏览因特网时通常不需要它们。CICS 不提供历史记录列表（列出 Web 浏览器的收藏夹或其他功能），因此可以忽略有关这些项的任何需求。

请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

第 2 部分 CICS Web support

本书的这一部分描述了 CICS Web support。

第 4 章 配置 CICS Web support 基本组件

所有 CICS Web support 任务都需要 CICS Web support 的基本组件。开始使用 CICS Web support 前，您需要配置这些组件。

第 17 页的『CICS Web support 的组件』列出了所有组件。您需要设置的基本组件是：

- TCP/IP 支持
- 对 z/OS UNIX 系统服务的访问
- SSL 支持
- 临时存储队列
- 系统初始化参数

如果要在先前的 CICS 发行版中使用编码的分析器程序来引用代码页转换表 DFHCNV，则可能需要设置某些 DFHCNV 条目。对于新的 CICS Web support 开发，不需要代码页转换表条目。

已设置这些基本组件后，您可以使用提供的样本程序验证 CICS Web support 的操作。

1. 按照《CICS 安装指南》中的『在 CICS 区域中使用 TCP/IP』中的指示信息为 CICS 区域启用 TCP/IP 支持。此过程包括设置 Communications Server 和建立通过 z/OS 对 DNS、或域名、服务器的访问。
2. 通过将 OMVS 段包括在 CICS 区域用户标识的用户概要文件中使 CICS 能访问 z/OS UNIX 系统服务，按《CICS 安装指南》中的『授权 CICS 区域用户标识访问 z/OS UNIX 系统服务』中的指示信息操作。
3. 设置 SSL 支持，按 the *CICS RACF Security Guide* 中的指示信息操作。*CICS RACF Security Guide* 说明了 SSL 提供的工具。
4. 为临时存储队列创建一个或多个 TSMODEL 定义以存储来自 Web 事务的 CICS Web support 数据。*CICS Resource Definition Guide* 将告诉您如何执行此操作。临时存储队列用于存储请求行、状态行，以及进站和出站消息的 HTTP 头。（它们不用于消息体。）为接收请求的端口在 TCPIP SERVICE 定义上指定临时存储队列的前缀。

注：不要指定 TSMODEL（它指定 POOLNAME 属性）：CICS Web support 不支持使用共享临时存储队列。而且，建议不要为 CICS Web support 使用可恢复的临时存储队列，因为这会导致锁定问题并减少吞吐量。

5. 按照第 42 页的『为 CICS Web support 指定系统初始化参数』中的步骤，指定适当的系统初始化参数。某些系统初始化参数在此阶段是可选的。
6. 尽可能根据需为 CICS Web support 保留属于 z/OS Communications Server 的端口。第 42 页的『保留 CICS Web support 的端口』具有有关此内容的更多信息。
7. **可选：**如果您在 CICS Web support 中有现有请求处理结构，而这些结构包括引用转换表（DFHCNV）中条目的分析器程序，并且您希望继续使用这些未更改的请求处理结构，则第 43 页的『迁移代码页转换表（DFHCNV）中的条目』会告诉您如何操作。对于任何其他 CICS Web support 任务，不需要代码页转换表。
8. 使用提供的样本验证 CICS Web support 的操作。第 43 页的『验证 CICS Web support 的操作』将告诉您如何执行此操作。

为 CICS Web support 指定系统初始化参数

CICS Web support 使用的系统初始化参数如下:

- DOCCODEPAGE
- LOCALCCSID
- TCPIP
- WEBDELAY

如果您正在使用安全套接字层, 则其他系统初始化参数如下:

- CRLSERVER
- ENCRYPTION
- KEYSRING
- MAXSSLTCBS
- SSLCACHE

CICS RACF Security Guide 说明如何使 SSL 与 CICS 一起运作, 包括指定这些系统初始化参数。

CICS System Definition Guide 具有有关所有系统系统初始化参数的缺省值的完整描述和信息。

- **必需:** 指定 TCPIP=YES 以激活 CICS TCP/IP 服务。缺省设置是 NO。必须指定 YES 以启用 CICS Web support。
- 使用 LOCALCCSID 系统初始化参数为本地 CICS 区域指定编码字符集标识。这是 CICS 认为是应用程序的缺省值的代码页。缺省值是 EBCDIC 代码页 037。如果未选择备用代码页转换选项, 则 CICS 将入局 HTTP 请求的数据内容传递到应用程序之前将它转换为此代码页, 并将假定应用程序已用此代码页提供 HTTP 响应。
- 如果您正在规划使用 CICS 文档模板支持, 则对 HTTP 请求提供静态响应, 或作为应用程序生成的响应的一部分, 使用 DOCCODEPAGE 系统初始化参数为文档域指定缺省主机代码页。缺省值是 EBCDIC 代码页 037。
- 如果您正在规划对 3270 显示应用程序提供 Web 客户机访问, 则使用 WEBDELAY 系统初始化参数指定合适的超时周期。两个超时周期控制:
 - 以分钟为单位的时间长度, 如果 Web 任务及其相关数据在这段时间内没有发生活动, 则将它们标记为删除。缺省值是 5 分钟。
 - 以分钟为单位的频率, 以此频率运行无用信息收集事务 CWBG 以删除标记的任务及其数据。缺省值为 60 分钟。

WEBDELAY 不应用到任何其他不涉及 3270 显示应用程序的 CICS Web support 任务。

保留 CICS Web support 的端口

建议您根据需要尽可能为 CICS Web support 保留属于 z/OS Communications Server 的端口, 并确保可能的情况下 CICS Web support 独用那些端口。

- 对于 HTTP, 熟知 (或缺省) 端口号是 80, 而对于 HTTPS, 熟知端口号是 443。您应该在可能使用熟知端口的同一 IP 地址上小心地解决与任何其他服务器的冲突。

- 对于非标准的服务器，应用程序员可以使用从 1024 到 32767 的端口号。1024 以下的端口是由 IANA 为了特殊函数构建的熟知端口号，因此需要 HTTP 端口 80 和 HTTPS 端口 443，它们不应该用于 CICS Web support。SSL 和非 SSL 请求必须使用不同的端口。
- 要保留 CICS Web support 侦听入局客户机请求的端口，您可以在 PROFILE.TCPIP 数据集中指定 PORT 语句或 CICS 作业名，如 *z/OS Communications Server: IP Configuration Reference*, SC31-8776 中所描述。
- 程序正在侦听的 TCP/IP 端口的任何请求队列的最大长度是由 PROFILE.TCPIP 数据集中的 SOMAXCONN 参数控制的。CICS 侦听 TCP/IP 端口，因此您必须协调此参数的值与为 TCPIP SERVICE 定义中 BACKLOG 参数选择的值。

迁移代码页转换表 (DFHCNV) 中的条目

在 CICS Transaction Server for z/OS V3R1 中，对于 CICS Web support，通常不需要代码页转换表。然而，如果您有现有请求处理结构，而这些结构包括引用转换表中的条目的分析器程序，并且您不希望更改分析器程序，则可继续提供 DFHCNV 条目。

在 CICS Transaction Server for z/OS V3R1 之前的 CICS 发行版中，代码页转换表 (DFHCNV) 用于定义 CICS 中使用的代码页和 Web 客户机使用的 ASCII 代码页之间的代码页转换。对于 CICS Transaction Server for z/OS V3R1 中的 CICS Web support，您不需要在代码页转换表中创建任何新条目。CICS Web support 使用 z/OS 转换服务处理代码页转换。

然而，如果您要继续使用先前 CICS 发行版中编码的分析器程序来引用 DFHCNV，则必须继续提供代码页转换表中的条目，或更改分析器程序。更改分析器程序涉及对两个新输出参数进行编码以指定客户机和服务器代码页，来代替指定 DFHCNV 条目的名称的输出参数。如果这样做，则不需要迁移 DFHCNV 条目。第 93 页的『编写分析器程序』告诉您如何用这种方式对输出参数进行编码。

注：所提供的 CICS 提供的样本分析器 DFHWBADX 指定样本代码页转换表 DFHCNVW\$ 中定义的条目。无需进行任何配置就可以使用样本转换表，但是您可能宁愿修改 DFHWBADX 以使用新的输出参数，从而提供更好的控制并避免使用样本转换表。

如果您愿意继续使用 DFHCNV，则遵循这些步骤操作：

1. 在先前 CICS 发行版中查找您要用于定义转换表的 DFHCNV 资源定义宏的源。这些宏的顺序应包括每个代码页对的 DFHCNV TYPE=ENTRY 宏。
2. 按照 *CICS Family: Communicating from CICS on System/390*[®] 中描述的过程，使用这些宏设置 DFHCNV 转换表。您需要定义、汇编和链接编辑该表。

验证 CICS Web support 的操作

提供样本应用程序 DFH\$WB1A 以帮助您测试 CICS Web support 的操作。样本应用程序构造以下 HTTP 响应：

```
DFH$WB1A on system applid successfully invoked through the CICS Web Interface
```

其中 *applid* 是正在运行 CICS Web support 的 CICS 系统的应用程序标识。

要运行样本程序:

- 安装指定 CICS 提供的样本分析器程序 DFHWBADX 的 TCPIP SERVICE 定义。
- 在 Web 浏览器上, 使用以下 URL 部分输入连接到 CICS Web support 的 URL:

Scheme

HTTP

Host 指定到 z/OS 映像的主机名

Port number

TCPIP SERVICE 定义中指定的端口号

Path /CICS/CWBA/DFH\$WB1A

第 7 页的『URL 的组成部分』具有有关如何定界 URL 的组成部分的更多信息。

DFH\$WB1C 是另一个样本应用程序。它与 DFH\$WB1A 具有相同的功能, 但它是用 C 编写的。

样本应用程序 DFH0WBCA 证明如何可以从 SSL 客户机证书抽取信息, 并使用 EXEC CICS DOCUMENT 命令将响应构造为 CICS 文档。此样本应用程序的定义在 DFH\$WBSN 资源定义组中提供。

第 5 章 规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构

开始规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构前，请阅读第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』主题，以了解可能涉及的处理阶段。

根据您希望作为 HTTP 服务器的 CICS 执行的任务不同，它的 CICS Web support 体系结构也不同。CICS Web support 的某些可配置组件对于所有任务来说都是必需的，例如，对接收入站请求的端口的 TCPIP SERVICE 定义。其他可配置组件仅对特定任务来说是必需的。

- 使用特殊设计的支持 Web 的 CICS 应用程序，您可以对 HTTP 请求提供动态、应用程序生成的响应。『用支持 Web 的应用程序提供动态 HTTP 响应』将告诉您如何执行此操作。
- 使用可用于 CICS 的文档或文件，您可以对 HTTP 请求提供静态响应。第 48 页的『用 CICS 文档模板或 HFS 文件提供静态 HTTP 响应』将告诉您如何执行此操作。
- 您可以允许 Web 客户机使用 HTTP 访问 CICS 中的现有通信区域应用程序。第 51 页的『提供对通信区域应用程序的 Web 客户机访问』将告诉您如何执行此操作。
- 您可以允许 Web 客户机使用 HTTP 访问 CICS 中的现有 3270 显示应用程序。第 131 页的第 15 章，『CICS Web support 和 3270 显示应用程序』将告诉您如何执行此操作。
- 您可以从客户机接收非 HTTP 请求，并提供应用程序生成的响应。第 127 页的第 14 章，『CICS Web support 和非 HTTP 请求』将告诉您如何执行此操作。

用支持 Web 的应用程序提供动态 HTTP 响应

您可以使用支持 Web 的应用程序对来自 Web 客户机的 HTTP 请求提供应用程序生成的响应。

必须在开始前配置 CICS Web support 的基本组件，如第 41 页的第 4 章，『配置 CICS Web support 基本组件』中所描述。

以下 CICS Web support 的特定于任务的组件用于此任务：

- TCPIP SERVICE 资源定义
- URIMAP 资源定义
- 支持 web 的应用程序，使用 EXEC CICS WEB 编程接口
- 应用程序的别名事务
- 分析器程序
- 安全工具
- Web 出错程序

第 46 页的图 4 说明此 CICS Web support 任务的体系结构元素。第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』说明进程元素如何一起运作。

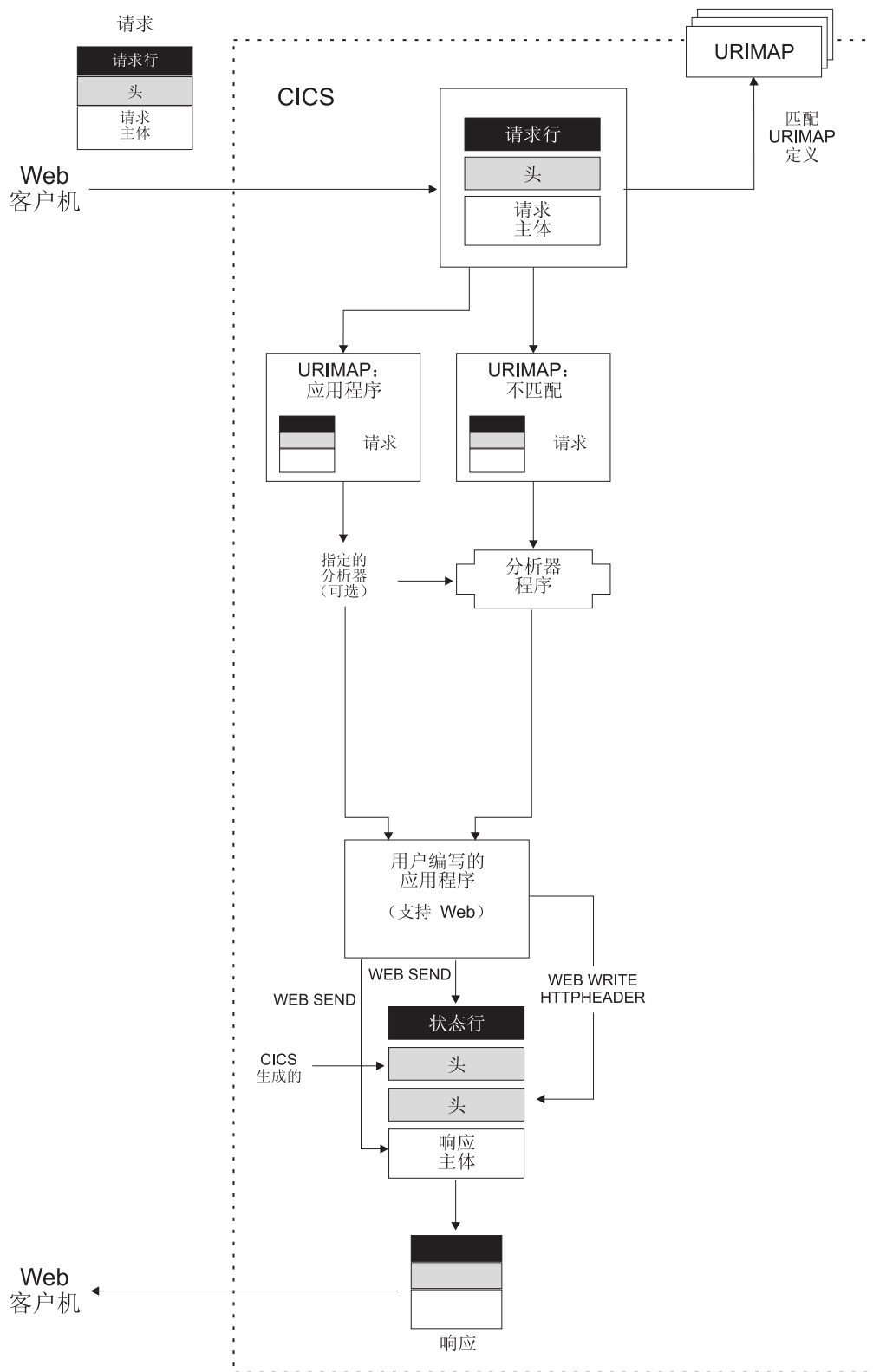


图 4. 具有支持 Web 的应用程序的动态 HTTP 响应

1. 使用，使用第 57 页的第 6 章，『为作为 HTTP 服务器的 CICS 编写支持 Web 的应用程序』中的信息，设计和编码一个或多个支持 Web 的应用程序以对 Web 客户机的每个请求提供响应。支持 Web 的应用程序可以使用 EXEC CICS WEB 和

EXEC CICS DOCUMENT 命令以接收和分析 HTTP 请求，并写响应以及对请求发送响应。每个程序处理单个请求和响应。

注：使用 EXEC CICS WEB 命令的支持 Web 的应用程序必须在接收 Web 客户机的请求的 CICS 区域中运行。然而，它们可以链接到其他 CICS 区域中的应用程序（例如，执行业务逻辑）。

2. 为此 CICS Web support 任务考虑安全问题。CICS 可以为连接实施 HTTP 基本认证，其中用户必须提供标识和密码。（注意，如果您需要带基于用户标识的访问控制的资源级安全性，则需要传递作为应用程序生成的响应的资料。资源级安全性不应用于作为静态响应传递的项。）如果您需要保护在因特网上传递的信息（包括用于基本认证的用户标识和密码），则考虑使用安全套接字层（SSL）。要获得更多信息，请参阅第 107 页的第 12 章，『CICS Web support 的安全性』。
3. 决定 Web 客户机将用于每个请求的 URL，包括规则、主机和路径部分以及任何查询字符串。第 7 页的『URL 的组成部分』说明了这些组件中的每一个以及如何对他们定界。第 28 页的『CICS Web support 的 URL』说明为 CICS Web support 选择 URL 要考虑的因素和限制。
4. 决定将用于请求的端口。第 42 页的『保留 CICS Web support 的端口』有关可由 CICS Web support 使用的端口的更多信息。对于 HTTP，缺省端口号是 80，而对于 HTTPS（具有 SSL），缺省端口号是 443。需要在请求的 URL 中明确指定不是规则的缺省值的端口号。如果您愿意，可以允许请求使用与 CICS Web support 关联的任何端口。
5. 如果接收请求的端口尚未有 TCPIPService 定义，则按第 71 页的『为 CICS Web support 创建 TCPIPService 资源定义』中的过程操作以设置一个 TCPIPService 定义。相关 TCPIPService 定义的名称在请求的 URIMAP 定义中指定。不指定 TCPIPService 定义意味着 URIMAP 定义匹配的请求可以使用存在 TCPIPService 定义的任何端口。
6. 为涉及本任务的用户应用程序选择别名事务标识。缺省别名事务是 CWBA。按第 74 页的『为 CICS Web support 创建 TRANSACTION 资源定义』中的指示信息进行操作，您可以创建您自己的别名事务。您可以使用 URIMAP 定义或分析器程序为每个 HTTP 请求指定别名事务。
7. 确定应如何使用与 TCPIPService 定义关联的分析器程序，并选择适当的程序。第 91 页的第 10 章，『分析器程序』具有关于使用分析器程序可执行的操作的更多信息。对于支持 web 的应用程序，您可从以下策略中选择：
 - a. 使用 CICS 提供的缺省分析器程序 DFHWBAAX 提供错误句柄。DFHWBAAX 适合于这种情况：使用此端口的所有请求是使用 URIMAP 定义处理的。如果找到匹配 URIMAP 定义，则 DFHWBAAX 不执行操作。如果找不到匹配，则它控制用户可替代 Web 出错程序 DFHWBERX 以产生错误响应。
 - b. 对于使用 URIMAP 定义的请求和按照 CICS TS 3.1 之前的 CICS Web support 使用的同一过程的请求，使用 CICS 提供的样本分析器程序 DFHWBADX 为它们提供基本支持。如果找到匹配 URIMAP 定义，则 DFHWBADX 不执行操作。如果找不到匹配，则它用 CICS TS 3.1 之前要求的格式分析 URL。如果分析失败，则 DFHWBADX 控制用户可替代 Web 出错程序 DFHWBEP 来产生错误响应。（如果您的 URIMAP 定义中指定的 URL 不适合 CICS TS 3.1 之前要求的格式，则您应定制 DFHWBADX 或 DFHWBEP 以便提供更有意义的响应。）
 - c. 使用您自己的分析器程序提供定制支持。这可能包括：

- 对使用 URIMAP 定义的请求的处理进行动态更改。
- 请求处理期间提供监控或审计操作。
- 支持按照 CICS TS 3.1 之前 CICS Web support 使用的同一过程的请求。
- 使用用户可替代的 Web 出错程序 DFHWBEP 和 / 或 DFHWBERX 提供错误响应。

使用 URIMAP 定义中的 ANALYZER(YES) 属性可以指定定制分析器程序，然后将它包括在请求的处理路径中。如前所述，如果从 URIMAP 定义调用 DFHWBAAX 和 DFHWBADX，则它们不执行操作。

8. 对于 HTTP 请求以及用户应用程序对它们提供的响应，决定希望如何执行代码页转换。代码页转换通常由以下操作组成：将使用 ASCII 字符集发出的 Web 客户机请求转换为 EBCDIC 代码页，以供应用程序使用；然后颠倒此过程以将应用程序的输出返回给 Web 客户机。第 32 页的『CICS Web support 的代码页转换』说明代码页转换的过程。您可以在支持 Web 的应用程序发出的 EXEC CICS WEB API 命令中指定代码页转换设置。
9. 设置 URIMAP 定义以处理每个请求。按第 75 页的『为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义』和第 77 页的『为 CICS（作为 HTTP 服务器）的 HTTP 请求的应用程序响应完成 URIMAP 定义』中的过程进行操作。按照 CICS TS 3.1 之前的 CICS Web support 使用的同一过程，可以将 HTTP 请求直接传递到分析器程序而无需使用 URIMAP 定义。然而，使用 URIMAP 定义可以更容易地管理 HTTP 请求。没有 URIMAP 定义的情况下，如果要更改 CICS 对特殊 HTTP 请求响应的方式，则需要更改分析器程序中的逻辑。使用 URIMAP 定义，您可以将这些更改作为系统管理任务动态执行。
10. 确保您的体系结构中包括的用户可替代 Web 出错程序向 Web 客户机提供适当的响应。如果 CICS Web support 进程中发生初始处理错误，或发生异常终止或故障情况，则使用 Web 出错程序。不是在所有错误情况下都使用它们。第 85 页的第 9 章，『Web 出错程序』说明使用 DFHWBEP 和 DFHWBERX 的情况，并告诉您如何定制它们提供的响应。

用 CICS 文档模板或 HFS 文件提供静态 HTTP 响应

您可以使用 CICS 文档模板或 z/OS UNIX 系统服务 HFS 文件对来自 Web 客户机的 HTTP 请求提供静态响应。

必须在开始前配置 CICS Web support 的基本组件，如第 41 页的第 4 章，『配置 CICS Web support 基本组件』中所述。

以下 CICS Web support 的特定于任务的组件用于此任务：

- TCPIPService 资源定义
- URIMAP 资源定义
- HFS 中的文件
- CICS 文档模板支持
- 安全性工具
- Web 出错程序

图 5 说明此 CICS Web support 任务的体系结构元素。第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』说明进程元素如何一起运作。

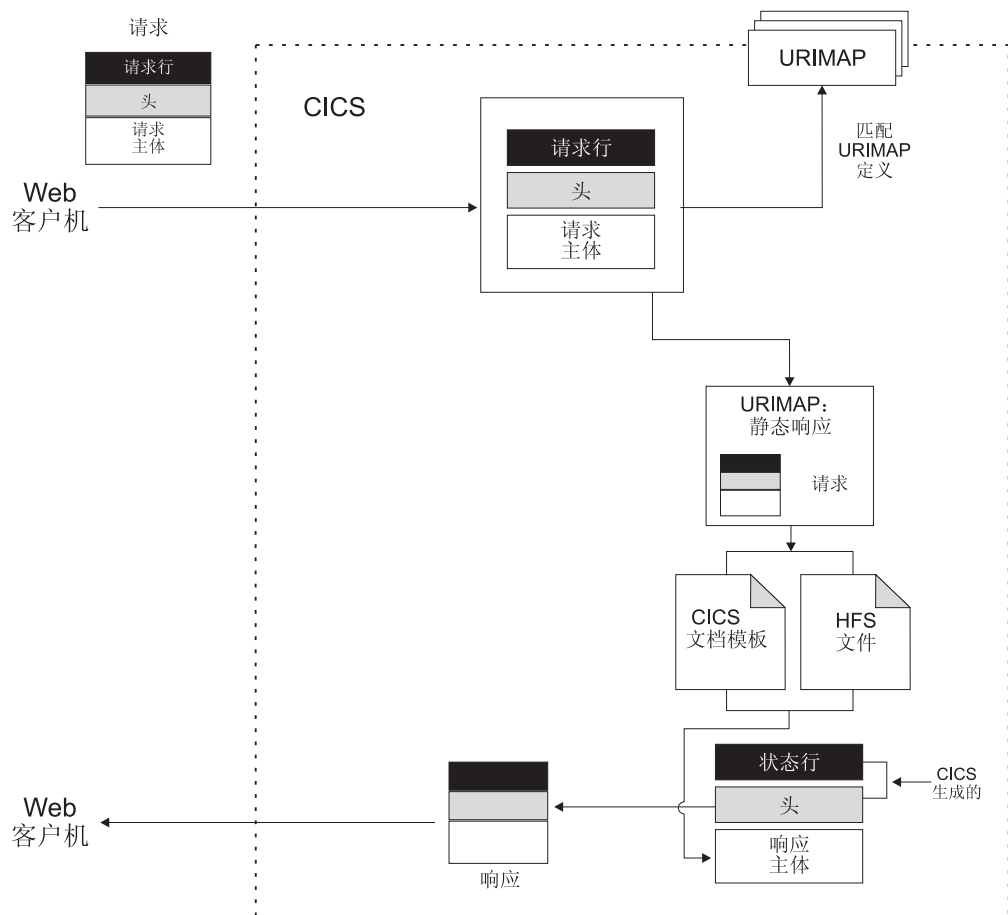


图 5. 静态 HTTP 响应

1. 为此 CICS Web support 任务考虑安全问题。CICS 可以为连接实施 HTTP 基本认证，其中用户必须提供标识和密码。（注意，如果您需要带基于用户标识的访问控制的资源级安全性，则需要传递作为应用程序生成的响应的资料。资源级安全性不应用于作为静态响应传递的项。）如果您需要保护在因特网上传递的信息（包括用于基本认证的用户标识和密码），则考虑使用安全套接字层（SSL）。要获得更多信息，请参阅第 107 页的第 12 章，『CICS Web support 的安全性』。
2. 如果要使用 z/OS UNIX 系统服务 HFS 文件以提供响应，则创建文件并将它放在您在 HFS 文件系统中选择的适当位置。标识此响应时，CICS 检索该文件并执行适当的代码页转换。
 - a. 不要在 HFS 文件中包含任何 HTTP 头或状态行信息。发送响应后，CICS 生成必需的信息。HFS 文件只提供响应主体。
 - b. 如果要使用路径匹配，则文件位置很重要，如本主题中稍后所述。如果不想使用路径匹配，则文件位置不需要与请求的 URL 有任何关系。
 - c. CICS 区域必须有访问 z/OS UNIX 的许可权，而且它必须具有访问包含文件的 HFS 目录以及文件本身的许可权。CICS 中的 Java 应用程序说明了如何授予这些许可权。

3. 如果要使用 CICS 文档模板提供响应，则创建该文档模板，并按 *CICS Application Programming Guide* 中的指示信息进行操作。文档模板是使用 DOCTEMPLATE 资源定义来定义的。文档模板可以保存在以下各项中：已分区的数据集 CICS 程序、文件、临时存储队列、瞬态数据队列、出口程序或 z/OS UNIX 系统服务 HFS 文件。标识此响应时，CICS 使用模板创建文档、检索该文档，并执行适当的代码页转换。

- a. 不要在文档模板中包含任何 HTTP 头或状态行信息。发送响应后，CICS 生成必需的信息。文档模板只提供响应主体。
- b. 组成名称值对的查询字符串可用作符号列表并替换成文档模板。（如果查询字符串已用于 URIMAP 匹配，作为 URIMAP 定义的 PATH 属性的一部分，则不能这样使用它。）使客户机发送 URL 中期望格式的查询字符串的建议方式是：发送具有 GET 方法的 HTML 表单以供用户填写。查询字符串中的任何名称都可在文档模板中作为符号编码，并且使用模板后，CICS 用每个符号替代在查询字符串中指定的值。例如，如果已获取包括名称和值对 `username=Peter` 的查询字符串，则可以通过将 `username` 编码为符号以在文档模板中使用此查询字符串：

欢迎使用财务系统，&username;。

传递到用户的结果静态响应将读到

欢迎使用财务系统，Peter。

注：文档模板中的符号是区分大小写的。将名称的大小写指定为与原始查询字符串中出现的大小写一样。

与文档模板中的符号不符合的任何名称和值对都会被忽略。

4. 识别由 HFS 文件或 CICS 文档模板提供的介质类型（数据内容的类型）。（请参阅第 7 页的『IANA 介质类型和字符集』以获取有关介质类型的更多信息。）请注意，使用 URIMAP 定义发送静态响应时，不支持使用质量指标（“q”参数）。质量指标可用于在客户机的可接受介质类型或字符集列表中选择，如 Accept 头中所指定。如果要执行这种类型的分析，则可以代之以使用应用程序生成的响应。

5. 识别 CICS 做出静态响应的代码页转换所需要的信息。只有在指定文本介质类型时才执行代码页转换。第 32 页的『CICS Web support 的代码页转换』说明代码页转换的过程。

- a. 将静态响应发送到 Web 客户机前，识别 CICS 应该将该静态响应转换到的字符集。CICS 为进行代码页转换支持的 IANA 字符集在第 183 页的附录 A，『HTML 编码字符集』中列出。
- b. 识别 IBM 代码页（EBCDIC），在该代码页中编码了提供响应主体的文档模板或 HFS 文件。

对于静态响应，此信息在请求的 URIMAP 定义中指定。

6. 决定 Web 客户机将用于每个请求的 URL，包括规则、主机和路径部分以及任何查询字符串。第 7 页的『URL 的组成部分』说明了这些组件中的每一个以及如何对他们定界。第 28 页的『CICS Web support 的 URL』说明为 CICS Web support 选择 URL 要考虑的因素和限制。

7. 决定是否要在 URIMAP 定义中使用路径匹配。如果使用，则计划您的请求 URL，并调整您的 CICS 文档模板的名称或您的 HFS 文件的位置以支持此操作。在路径匹配中，通配符在 URIMAP 定义的路径部分中使用，也在 URIMAP 定义指定的 CICS 文档模板或 HFS 文件的名称中使用。由通配符覆盖的路径部分用于选择文档模板或 HFS 文件以提供响应。

- a. 对于 CICS 文档模板，由通配符覆盖的路径部分替换为模板名的最后一部分。您可以创建名称以相同方式开头的文档模板集合，并通过单个 URIMAP 定义，使用路径以相同方式开头的请求 URL 来访问它们。
 - b. 对于 HFS 文件，由通配符覆盖的路径部分替换为 HFS 文件名的最后一部分。您可以将许多 HFS 文件存储在同一目录中，并通过单个 URIMAP 定义，使用路径以相同方式开头的请求 URL 来访问它们。请记住，因为 URIMAP 定义必须指定一种类型的数据内容 (MEDIATYPE 属性)，因此单个 URIMAP 定义只能处理产生同一类型数据内容的一组 HFS 文件。
8. 决定将用于请求的端口。第 42 页的『保留 CICS Web support 的端口』有关可由 CICS Web support 使用的端口的更多信息。对于 HTTP，缺省端口号是 80，而对于 HTTPS (具有 SSL)，缺省端口号是 443。需要在请求的 URL 中明确指定不是规则的缺省值的端口号。如果您愿意，可以允许请求使用与 CICS Web support 关联的任何端口。
 9. 如果接收请求的端口尚未有 TCPIPService 定义，则按第 71 页的『为 CICS Web support 创建 TCPIPService 资源定义』中的过程操作以设置一个 TCPIPService 定义。相关 TCPIPService 定义的名称在请求的 URIMAP 定义中指定。不指定 TCPIPService 定义意味着 URIMAP 定义匹配的请求可以使用存在 TCPIPService 定义的任何端口。
 10. 设置 URIMAP 定义以处理每个请求。按第 75 页的『为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义』和第 77 页的『为作为 HTTP 服务器的 CICS 的 HTTP 请求的静态响应完成 URIMAP 定义』中的过程进行操作。URIMAP 定义可以识别 HFS 文件或文档模板。
 11. 为此 CICS Web support 任务检查错误句柄过程。
 - a. 检查与 TCPIPService 定义关联的分析器程序行为，以了解在哪个端口上接收请求。如果为请求的 URIMAP 匹配失败，则请求被传递到分析器程序。如果此端口只用于静态响应，则 CICS 提供的缺省分析器程序 DFHWBAAX 提供合适的错误句柄。否则，分析器程序的选择可能取决于用户应用程序的需求，并且您可能需要定制它来为静态响应提供合适的错误句柄。第 91 页的第 10 章，『分析器程序』具有关于使用分析器程序可执行的操作的更多信息。
 - b. 确保您的体系结构中包括的用户可替代 Web 出错程序向 Web 客户机提供适当的响应。第 85 页的第 9 章，『Web 出错程序』说明使用 DFHWBEP 和 DFHWBERX 的情形，并告诉您如何定制它们提供的响应。

提供对通信区域应用程序的 Web 客户机访问

您可以使用 CICS Web support 以允许 Web 客户机与 CICS 应用程序交互，这些应用程序使用通信区域接口与其他程序通信。您可以写支持 Web 的应用程序，它会链接到应用程序并使用其输出提供 HTTP 响应。另外，您可以使用转换器程序将 Web 客户机的输入转换为合适的通信区域，并将应用程序的输出转换为 HTML 以提供响应。

必须在开始前配置 CICS Web support 的基本组件，如第 41 页的第 4 章，『配置 CICS Web support 基本组件』中所述。

以下 CICS Web support 的特定于任务的组件用于此任务：

- TCPIPService 资源定义
- URIMAP 资源定义
- 通信区域应用程序

- 要么是支持 Web 的应用程序，它们使用 EXEC CICS WEB 编程接口链接到通信区域应用程序并使用这些应用程序的输出。
- 或者是转换器程序，它们可以提供合适的通信区域输入，并将应用程序的输出转换成 HTML
- 包括此进程中涉及的用户应用程序的别名事务
- 分析器程序
- 安全性工具
- Web 出错程序

第 53 页的图 6 说明此 CICS Web support 任务的体系结构元素。第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』说明进程元素如何一起运作。

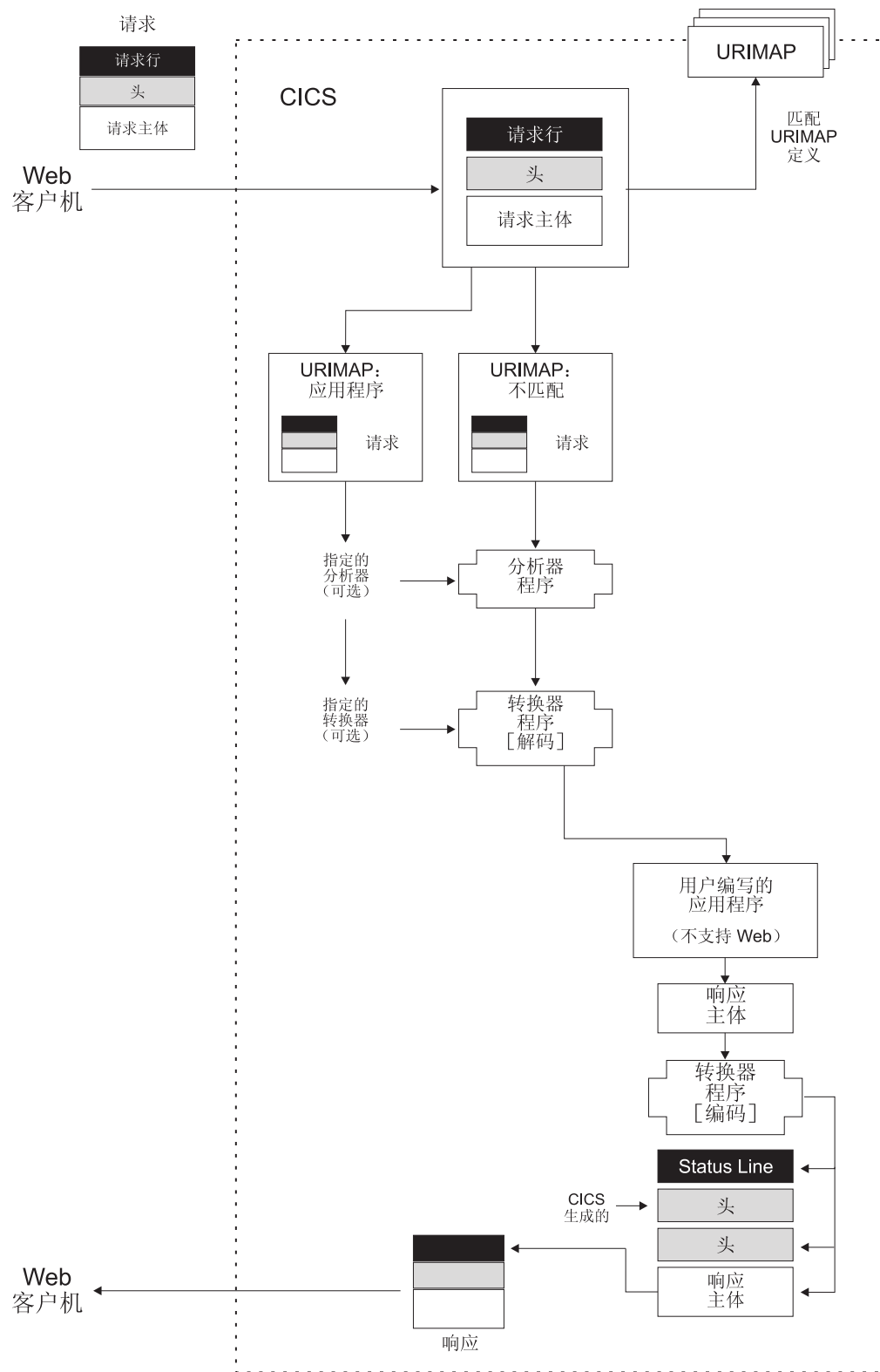


图 6. 来自通信区域应用程序的 HTTP 响应

1. 决定是否要写支持 Web 的应用程序，该应用程序处理 HTTP 请求并链接到通信区域应用程序；或者是否要写转换器程序以将 Web 客户机的输入转换为合适的通信区域，并将应用程序的输出转换为 HTML 以提供响应。

注: 对于使用转换器程序的通信区域应用程序, CICS 无法提供关于结构化响应和 HTTP 协议一致性的最完整的可能协助。要利用所有可用的 CICS Web support 工具, 建议您在可能的情况下使用支持 Web 的应用程序处理请求。

- a. 如果要使用支持 Web 的应用程序, 则按第 45 页的『用支持 Web 的应用程序提供动态 HTTP 响应』中的步骤进行操作。编写支持 Web 的应用程序的代码以链接到通信区域应用程序并使用其输出。唯一不能由支持 Web 的应用程序执行, 但能由转换器程序执行的任务是: 接收分析器程序已创建的、要传递到下一个处理阶段(在用户令牌或共享工作区中)的信息。开发新的 CICS Web support 应用程序时, 可能不需要考虑这一点。
 - b. 如果要使用转换器程序, 则按本主题中的步骤进行操作。
2. 确定应如何使用与 TCPIP SERVICE 定义关联的分析器程序, 并选择适当的程序。第 91 页的第 10 章, 『分析器程序』提供关于使用分析器程序可执行的操作的更多信息。URIMAP 定义或分析器程序可用于将请求从 Web 客户机映射到适当的转换器程序和用户编写的应用程序。对于不支持 Web 的应用程序, 即使您具有 URIMAP 定义, 在以下这些情况中您也需要使用请求的处理路径中的定制分析器程序:
- a. 如果您需要非标准代码页转换。不支持 Web 的应用程序和转换器程序没有为 HTTP 请求和响应指定代码页转换的机制。在没有分析器程序的情况下, CICS 使用 CICS 系统的缺省代码页(由 LOCALCCSID 系统初始化参数指定)转换通信区域中提供的关于输入和输出的消息体, 以及 Web 客户机的请求中指定的任何支持字符集(或 ISO-8859-1 字符集)。如果此操作不适当, 则必须在处理路径中使用分析器程序以指定替换设置。第 32 页的『CICS Web support 的代码页转换』具有有关代码页转换过程的更多信息。
 - b. 如果您需要将除标准输入之外的任何信息与转换器程序进行通信。将提供一个用户令牌, 分析器和转换器程序可以用它来交换少量信息或交换包含更多信息的共享工作区的地址。
 - c. 如果您需要可以由分析器程序执行的监控或审计操作。
 - d. 如果您需要对过程元素(如, 使用的转换器程序、处理请求的应用程序, 或者用于请求的别名事务和用户标识)进行动态更改。

如果您不需要这些功能中的任何一个功能, 则可使用 CICS 提供的缺省分析器程序 DFHWBAAX 或 CICS 提供的样本分析器程序 DFHWBADX 来提供基本错误句柄。DFHWBAAX 适合于这种情况: 使用此端口的所有请求是使用 URIMAP 定义处理的。对于使用 URIMAP 定义请求和按照 CICS TS 3.1 之前的 CICS Web support 使用的同一过程的请求, DFHWBADX 提供基本支持。

3. 使用第 101 页的第 11 章, 『转换器程序』中的信息以创建合适的转换器程序。CICS 将 Web 客户机的请求传递到通信区域中的转换器程序。(如果请求在通信区域中不适合, 则会截断它。)会调用转换器程序两次, 第一次是为**解码函数**调用, 它接收请求并将它传递到应用程序, 第二次是为**编码函数**调用, 它接收应用程序的输出和创建 HTTP 响应。由转换器程序执行的任务有:

解码函数

- 接收通信区域中的 HTTP 请求, 连同 URIMAP 定义或分析器程序提供的任何其他信息。
- 处理 HTTP 请求并将相关元素转换成应用程序的合适输入(在同一通信区域或新的通信区域中)。

编码函数

- 从应用程序接收输出。
 - 如果多个应用程序要提供数据，则设置循环响应，以再次调用解码函数。第 105 页的『调用多个 CICS 应用程序』说明如何完成此操作。
 - 在通信区域中构造 HTTP 响应。
 - 将 HTTP 响应传递到 CICS 以发送给 Web 客户机。
4. 决定 Web 客户机将用于每个请求的 URL，包括规则、主机和路径部分以及任何查询字符串。第 7 页的『URL 的组成部分』说明了这些组件中的每一个以及如何对他们定界。第 28 页的『CICS Web support 的 URL』说明为 CICS Web support 选择 URL 要考虑的因素和限制。
 5. 决定将用于请求的端口。第 42 页的『保留 CICS Web support 的端口』有关可由 CICS Web support 使用的端口的更多信息。对于 HTTP，缺省端口号是 80，而对于 HTTPS（具有 SSL），缺省端口号是 443。需要在请求的 URL 中明确指定不是规则的缺省值的端口号。如果您愿意，可以允许请求使用与 CICS Web support 关联的任何端口。
 6. 为涉及本任务的用户应用程序选择别名事务标识。缺省别名事务是 CWBA。按第 74 页的『为 CICS Web support 创建 TRANSACTION 资源定义』中的指示信息进行操作，您可以创建您自己的别名事务。您可以使用 URIMAP 定义或分析器程序为每个 HTTP 请求指定别名事务。
 7. 为此 CICS Web support 任务考虑安全问题。CICS 可以为连接实施 HTTP 基本认证，其中用户必须提供标识和密码。（注意，如果您需要带基于用户标识的访问控制的资源级安全性，则需要传递作为应用程序生成的响应的资料。资源级安全性不应用于作为静态响应传递的项。）如果您需要保护在因特网上传递的信息（包括用于基本认证的用户标识和密码），则考虑使用安全套接字层（SSL）。要获得更多信息，请参阅第 107 页的第 12 章，『CICS Web support 的安全性』。
 8. 设置 URIMAP 定义以处理每个请求。按第 75 页的『为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义』和第 77 页的『为 CICS（作为 HTTP 服务器）的 HTTP 请求的应用程序响应完成 URIMAP 定义』中的过程进行操作。按照 CICS TS 3.1 之前的 CICS Web support 使用的同一过程，可以将 HTTP 请求直接传递到分析器程序而无需使用 URIMAP 定义。然而，使用 URIMAP 定义可以更容易地管理 HTTP 请求。没有 URIMAP 定义的情况下，如果要更改 CICS 对特殊 HTTP 请求响应的方式，则需要更改分析器程序中的逻辑。使用 URIMAP 定义，您可以将这些更改作为系统管理任务动态执行。
 9. 如果接收请求的端口尚未有 TCPIPService 定义，则按第 71 页的『为 CICS Web support 创建 TCPIPService 资源定义』中的过程操作以设置一个 TCPIPService 定义。相关 TCPIPService 定义的名称在请求的 URIMAP 定义中指定。不指定 TCPIPService 定义意味着 URIMAP 定义匹配的请求可以使用存在 TCPIPService 定义的任何端口。
 10. 为此 CICS Web support 任务检查错误句柄过程。
 - a. 检查与 TCPIPService 定义关联的分析器程序行为，以了解在哪个端口上接收请求。如果为请求的 URIMAP 匹配失败，则请求被传递到分析器程序。第 91 页的第 10 章，『分析器程序』提供关于使用分析器程序可执行的操作的更多信息。
 - b. 确保您的体系结构中包括的用户可替代 Web 出错程序向 Web 客户机提供适当的响应。第 85 页的第 9 章，『Web 出错程序』说明使用 DFHWBEP 和 DFHWBERX 的情况，并告诉您如何定制它们提供的响应。

第 6 章 为作为 HTTP 服务器的 CICS 编写支持 Web 的应用程序

在 CICS 中，编写支持 Web 的应用程序是使用 EXEC CICS WEB 命令通过 CICS 与 Web 客户机或服务器交互的程序。对于作为 HTTP 服务器的 CICS，这些程序可以接收和分析 HTTP 请求，并对 Web 客户机提供应用程序生成的响应。

开始为作为 HTTP 服务器的 CICS 编写支持 Web 的应用程序代码前，请阅读第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』主题，以了解会涉及到的处理阶段。

如果您希望对 Web 客户机提供的服务遵从 HTTP 协议规范，特别是 HTTP/1.1，请阅读主题第 35 页的『作为 HTTP 服务器的 CICS 的 HTTP/1.1 一致性』以获取有关 CICS 和您的应用程序实现这一目的的可采取的操作的更多信息。

对于需要应用程序生成的响应的每个 HTTP 请求，CICS 调用请求的 URIMAP 定义上指定，或者，如果使用分析器，由分析器程序指定的支持 Web 的应用程序。如果使用 URIMAP 定义指定应用程序，您可选择单个应用程序，以使用特殊 URL 为所有请求提供服务。如果您使用分析器程序代替（或除了）URIMAP 定义，它可在请求上执行分析，并确定备用应用程序。

切记：使用 EXEC CICS WEB 命令的支持 Web 的应用程序必须在接收 Web 客户机的请求的 CICS 区域中运行。然而，它们可以链接到其他 CICS 区域中的应用程序（例如，执行业务逻辑）。

对于作为 HTTP 服务器的 CICS，当应用程序已对请求发送响应且将控制返回给 CICS 后，它不等待来自 Web 客户机的更多请求。即使请求形成了逻辑序列，或正在使用持续连接发出请求，或者是管道化的请求，都处于这种情况。如果需要在不同的程序间（或同一程序的新实例间）共享信息，则可以使用 CICS 管理的资源或使用 Web 客户机发送的请求的元素来执行此操作。

您可以编码每个支持 Web 的应用程序以执行处理 HTTP 请求的以下部分或所有操作：

1. 使用 WEB EXTRACT 命令，从请求行（包括请求 URL）获取您的应用程序所需的任何信息。第 58 页的『检查 HTTP 请求的请求行』将告诉您如何执行此操作。请求行包括 HTTP 方法，它表明应用程序应该执行的操作。您还可以设计请求 URL 的路径部分以对应用程序提供处理信息。如果请求 URL 中有查询字符串，则应用程序可以将它作为整体来抽取以便进行处理。
2. 使用 HTTP 头命令读取或浏览请求的 HTTP 头。第 59 页的『检查消息的 HTTP 头』将告诉您如何执行此操作。HTTP 头中的信息可能对应用程序处理和响应请求是有用的。
3. 获取有关您的应用程序需要使用的请求的任何技术信息。您可使用 EXEC CICS 命令访问有关 TCP/IP 环境和安全选项的信息。第 60 页的『检索有关 HTTP 请求的技术和安全信息』将告诉您如何执行此操作。
4. 如果请求包含您要抽取的表单数据，则使用表单字段命令读取或浏览数据。第 61 页的『检查 HTTP 请求中的表单数据』将告诉您如何执行此操作。该数据可以在请求主体中或者作为 URL 中的查询字符串，且 CICS 可以从这些位置中的任一位置抽取该数据。

5. 如果请求具有需要使用的消息体，则使用 WEB RECEIVE 命令将它接收到缓冲区中。第 62 页的『接收 HTTP 请求的实体主体』将告诉您如何执行此操作。如果有消息体，则 CICS 不要求您再接收一个，而某些请求没有消息体。
6. 使用您已收集的信息为请求处理执行业务逻辑。您可能要包含其他应用程序以执行处理。根据从不支持 Web 的程序接收的信息，支持 Web 的应用程序可以产生到 HTTP 请求的响应。建议您将业务逻辑与演示逻辑分开。在支持 Web 的应用程序中，演示逻辑控制与 Web 客户机的交互。要获得如何区别业务和演示逻辑的建议，请参阅 *CICS Application Programming Guide*。
7. 使用 WEB WRITE HTTPHEADER 命令为响应写 HTTP 头。第 64 页的『为响应写 HTTP 头』将告诉您如何执行此操作。CICS 自动提供某些必需的头（如，Date 头）。您可以出于其他目的提供其他头。
8. 产生实体主体或消息体，它是 HTTP 响应的内容。第 65 页的『为 HTTP 消息产生实体主体』将告诉您如何执行此操作。实体主体可以由 CICS 文档（使用 EXEC CICS DOCUMENT 应用程序编程接口创建的）形成，或由应用程序提供的数据缓冲区形成。
9. 使用 WEB SEND 命令将响应发送到 Web 客户机。第 66 页的『从作为 HTTP 服务器的 CICS 发送 HTTP 响应』将告诉您如何执行此操作。您需要选择合适的状态码和原因短语，并指定实体主体。CICS 使用这些项和 HTTP 头汇编响应。如果要使用分块的传输编码，还需要按第 67 页的『使用分块的传输编码发送 HTTP 请求或响应』中的特殊指示信息进行操作。
10. 如果您希望与该 Web 客户机交换更多的请求和响应，且需要跨请求序列共享数据，则使用第 70 页的『跨 HTTP 请求序列管理应用程序状态』中的建议来实现这一目的。

当 EXEC CICS WEB 命令用于作为 HTTP 服务器的 CICS，不要使用 SESSTOKEN 选项。SESSTOKEN 选项表明命令用于作为 HTTP 客户机的 CICS。

检查 HTTP 请求的请求行

CICS 存储用于每个 HTTP 请求的请求行，以供应用程序在需要时可以访问。应用程序可以使用 WEB EXTRACT 命令来抽取请求 URL 的部分（包括路径、主机名、端口号和查询字符串）、用于请求的方法或请求的 HTTP 版本。非 HTTP 请求也可以用这种方式识别。

第 9 页的『HTTP 请求』说明请求行中的项。请求 URL 是请求行的主要元素；第 7 页的『URL 的组成部分』说明 URL 的不同部分。为了处理请求并提供适当的响应，您的应用程序可能需要检查请求行中的任何项。从请求行抽取信息的某些常见原因如下：

- 因为调用同一应用程序来处理很多不同的请求，也许作为逻辑请求序列的一部分请求，或作为与同一资源相关的不同请求。
- 要查看 HTTP 方法从应用程序请求了什么操作。第 199 页的附录 D，『CICS Web support 的 HTTP 方法参考』说明 Web 客户机可能对请求使用的不同方法，并建议适合每种情况的操作。
- 要使用 URL 的路径部分。这将识别应用请求的资源。除了用于将请求映射到处理应用程序外，URL 的路径部分还可以设计为对应用程序提供处理信息。例如，路径部分可用于指定应用程序提供的特殊函数。或者如果支持 Web 的应用程序正在对多个其他应用程序提供前端，则 URL 的路径部分可以识别应用请求的应用程序。第 28 页的『CICS Web support 的 URL』说明如何达到此目的。

- 要获取由应用程序处理的查询字符串。
- 要识别 Web 客户机的 HTTP 版本，从而应用程序可以提供适当的响应。Web 客户机使用的 HTTP 版本会影响响应的 HTTP 头、状态码和消息内容。HTTP/1.0 客户机可能不了解 HTTP/1.1 规范中描述的更多高级功能。
- 要识别非 HTTP 请求。第 127 页的第 14 章，『CICS Web support 和非 HTTP 请求』具有有关处理非 HTTP 请求的更多信息。

CICS Application Programming Reference 有 WEB EXTRACT 命令中可用的选项的完整参考信息和描述。WEB EXTRACT 命令可以让您获取以下项：

- 使用 HOST 选项以获取请求 URL 的主机部分，如请求的 Host 头字段或请求行（如果绝对 URI 格式用于请求）中所指定。
- 使用 HTTPMETHOD 选项以获取请求的 HTTP 方法（例如，GET 或 PUT）。
- 使用 HTTPVERSION 选项以识别 HTTP 版本（HTTP/1.1 或 HTTP/1.0）。
- 使用 PATH 选项以获取 URL 的路径部分。
- 使用 PORTNUMBER 选项以获取应用到 URL 的端口号。服务的熟知端口号通常从 URL 中省略。如果 URL 中不存在该端口号，则 WEB EXTRACT 命令将根据规则识别并返回它。对于 HTTP，熟知端口号是 80，而对于 HTTPS，熟知端口号是 443。
- 使用 QUERYPSTRING 选项以获取整个查询字符串。查询字符串以其转义形式返回，具有 %xx 序列表示特定字符，这些字符会妨碍正确的语法分析。请参阅第 12 页的『转义和未转义数据』以获取有关此内容的说明。另外，如果查询字符串包括作为名称和值对（例如，account=40138025）的表单数据，则可以使用 WEB READ FORMFIELD 命令以获取非转义形式的此数据。第 61 页的『检查 HTTP 请求中的表单数据』将告诉您如何执行此操作。
- 使用 REQUESTTYPE 选项以识别非 HTTP 请求。

检查消息的 HTTP 头

请求或响应消息的每个 HTTP 头由头名称和头值组成。如果需要，CICS 存储此信息以供应用程序访问。应用程序可以接收指定头的值，或浏览请求或响应的所有头的名称和值。您还可以将从头中获取的结构化日期和时间戳记字符串转换为 ABSTIME 格式。

为了处理请求或响应并构造后续消息，您的应用程序可能需要检查头中的信息。例如：

- TE 头告诉应用程序：分块的响应消息中是否允许尾部头。
- 条件头可以对应用程序提供指示信息，例如，仅当响应文档更改后才应答。

请记住，除非您知道 HTTP 请求或响应的确切格式，否则您的应用程序不应该依赖于任何特殊头的存在，这是因为 Web 客户机和服务器发送的头可能不一致。

某些 HTTP 头包含日期和时间戳记。CICS 提供 CONVERTTIME 命令，将结构化日期和时间戳记字符串的常见格式转换为 ABSTIME 格式，以供应用程序使用。

标准 HTTP 头在 HTTP/1.1 规范（RFC 2616）和 HTTP/1.0 规范（RFC 1945）中描述。第 185 页的附录 B，『CICS Web support 的 HTTP 头参考』说明 CICS Web support 中 HTTP 头的一般用法，以及 CICS Web support 为消息中接收的特定头采取的操作。

某些 HTTP 头会被 CICS 忽略，并且由用户应用程序来决定执行适当的操作以响应这些头。要获取有关每个 HTTP 头的含义和正确使用的详细指导和需求，请检查您正在遵从的 HTTP 规范。

如果消息包含任何尾部头，则您可使用 EXEC CICS WEB 命令，通过用于标准头的相同方法读取这些尾部头。消息上的尾部头指定作为尾部头发送的所有 HTTP 头的名称。

- 要检查特殊 HTTP 头的内容，请使用 WEB READ HTTPHEADER 命令。您的应用程序需要提供接收头内容的缓冲区。如果请求中不存在头，则 CICS 返回 NOTFND 条件。
- 要浏览请求或响应中的所有头：
 1. 使用 WEB STARTBROWSE HTTPHEADER 命令以开始浏览头行。
 2. 使用 WEB READNEXT HTTPHEADER 命令以检索每一行的头名称和头值。您的应用程序需要提供两个缓冲区：一个接收头的名称，而另一个接收其内容。读取所有头后，CICS 将返回 ENDFILE 条件。
 3. 当您的程序已检索所有相关头信息后，使用 WEB ENDBROWSE HTTPHEADER 命令结束浏览。
- 要转换 HTTP 头中提供的结构化日期和时间戳记字符串，请使用 WEB READ HTTPHEADER 命令将它接收到缓冲区中，然后使用 CONVERTTIME 命令处理它。您不需要识别日期和时间戳记的格式；CONVERTTIME 命令会识别在因特网上常用的三种不同日期和时间戳记格式并进行转换。它们是 RFC 1123 格式（Web 标准）、RFC 850 格式（较早的格式）和 ASCtime 格式（C 函数的输出）。使用 FORMATTIME 命令，应用程序可以将 ABSTIME 转换为其他格式。

检索有关 HTTP 请求的技术和安全信息

应用程序可以获取有关 HTTP 请求的 TCP/IP 环境的信息（包括正在使用的安全选项），以及有关 Web 客户机已提供的客户机证书的信息。

CICS 管理 Web 客户机和服务器之间的 TCP/IP 连接，应用适当的安全措施，并管理 Web 客户机的标识的认证过程。CICS 为每个连接采取的操作，由接收 Web 客户机请求的端口的 TCPIPService 定义中设置的选项确定。用户编写的应用程序可以检查此过程获取的信息，如果这可用于确定如何处理请求。例如，您可以获取发送 HTTP 请求的 Web 客户机的主机名和 IP 地址，或检查连接的安全和加密级别。

EXTRACT TCPIP 命令提供有关 TCP/IP 连接，以及有关 TCPIPService 定义中指定的安全选项的信息。EXTRACT CERTIFICATE 命令提供从任何 X.509 客户机证书获取的信息，该证书是在安全套接字层（SSL）握手期间从 Web 客户机接收的。CICS *Application Programming Reference* 提供有关这些命令的可用选项的完整参考信息和描述。

- 要获取发送 HTTP 请求的 Web 客户机的主机名和 IP 地址，请使用具有 CLIENTNAME 和 CLIENTADDR 选项的 EXTRACT TCPIP 命令。IP 地址作为二进制数字提供，或作为包含其点分十进制表示法的字符串提供。
- 要获取运行应用程序的主机系统（即，CICS 本身）的主机名和 IP 地址，可使用带有 SERVERNAME 和 SERVERADDR 选项的 EXTRACT TCPIP 命令。IP 地址再次作为二进制数字提供，或作为包含其点分十进制表示法的字符串提供。

- 要获取接收请求的端口号，可以使用具有 PORTNUMBER 选项的 EXTRACT TCPIP 命令。端口号作为二进制数字或字符串提供。也可以使用具有 PORTNUMBER 选项的 WEB EXTRACT 命令。
- 要获取与请求关联的 TCPIP SERVICE 资源定义的名称，使用具有 TCPIP SERVICE 选项的 EXTRACT TCPIP 命令。
- 要识别 TCPIP SERVICE 定义中指定的认证类型（基本认证、客户机证书认证或没有认证），使用具有 AUTHENTICATE 选项的 EXTRACT TCPIP 命令。第 109 页的『作为 HTTP 服务器的 CICS: 认证和标识』说明有关不同类型的认证的更多信息。
- 要识别是否在 TCPIP SERVICE 定义中指定了安全套接字层（SSL）支持和使用的 SSL 加密级别，则使用具有 SSLTYPE 和 PRIVACY 选项的 EXTRACT TCPIP 命令。第 112 页的『具有 CICS Web support 的 SSL』说明有关 SSL 的更多内容。
- 要从 X.509 证书（该证书是在 SSL 握手期间从 Web 客户机接收的）检索信息，请使用 EXTRACT CERTIFICATE 命令。通过依照安全管理器的数据库和您设置的证书撤回列表检查所提供的证书，CICS 已验证了该提供的证书。证书包含识别证书主题（有时称为所有者或用户）的字段，以及识别发布证书的认证中心（发布者）的字段。可以通过指定 OWNER 或 ISSUER 选项来选择您需要的信息。您还可以使用 SERIALNUM 和 USERID 选项来获取证书号以及与证书关联的 RACF 用户标识。CICS RACF Security Guide 说明了有关证书的内容以及如何使用它们的更多信息。

检查 HTTP 请求中的表单数据

表单数据是用户通过与 HTML 表单中的元素（例如，文本输入框、按钮或复选框）交互作用而提供的信息。该信息作为一系列名称和值对发送。CICS 可以搜索 HTTP 请求以检取表单字段，因此应用程序可使用 CICS 命令获取数据而无需接收和分析整个请求主体。

第 13 页的『HTML 表单』说明有关表单和表单字段的更多内容。

应用程序可以接收指定表单字段的值，或浏览请求中包含的所有表单字段的名称和值。如果需要将数据转换到不同的代码页中以供应用程序使用，则可以指定代码页转换选项。

使用 GET 方法时，Web 客户机在查询字符串中发送表单数据，而使用 POST 方法时，它在消息体中发送表单数据。CICS 可以从这两个位置中的任一位置抽取数据，所以您不需要指定使用哪个方法。作为备用方法，如果在查询字符串中发送表单数据，则可以使用 WEB EXTRACT 命令检索整个查询字符串。第 58 页的『检查 HTTP 请求的请求行』将告诉您如何执行该操作。

只有 CICS 是 HTTP 服务器时，CICS 才读取表单数据。当 CICS 是 HTTP 客户机时，该工具不可用。

- 要获取 HTML 表单的特殊字段的值，请使用 WEB READ FORMFIELD 命令。您的应用程序可以提供将接收值的缓冲区，或者可以提供 CICS 设置为值地址的指针。如果表单数据不包含具有指定名称的字段，则 CICS 返回 NOTFND 条件。表单数据返回前 CICS 不对它进行转义，%xx 序列将转换回原始字符。请参阅第 12 页的『转义和未转义数据』以获取有关此内容的说明。
- 要浏览表单数据中的所有字段:
 1. 使用 WEB STARTBROWSE FORMFIELD 命令开始浏览这些字段。

2. 使用 WEB READNEXT FORMFIELD 命令依次检索每个字段的名称和值。您的应用程序需要提供两个缓冲区：一个接收字段名，而另一个接收其内容。读取所有字段后，CICS 将返回 ENDFILE 条件。
 3. 当您的程序已检索所有相关字段后，使用 WEB ENDBROWSE FORMFIELD 命令以结束浏览。
- CICS 将接收的数据从为 Web 客户机指定的代码页转换为应用程序指定的代码页。如果需要对这些代码页进行更改，则在 WEB READ FORMFIELD 和 WEB READNEXT FORMFIELD 命令中使用 CLNTCODEPAGE 和 HOSTCODEPAGE 选项。
 1. HOSTCODEPAGE 指定应用程序使用的代码页。CICS 使用的缺省值是 EBCDIC Latin 字符集（代码页 037）。
 2. CLNTCODEPAGE 指定 Web 客户机使用的代码页。如果未指定此内容，则 CICS 从 HTTP 请求的 Content-Type 头获取此信息。如果这些头不包含字符集信息，则 CICS 采用 ISO-8859-1 字符集。

接收 HTTP 请求的实体主体

应用程序可以发出 WEB RECEIVE 命令以接收 HTTP 请求的实体主体。您只能接收实体主体的第一部分，或使用一系列 WEB RECEIVE 命令来以较小的多个部分接收整个主体。

WEB RECEIVE 命令不设置超时值。只有已从 Web 客户机成功接收完整请求后才调用用户应用程序，且该应用程序由 CICS 保存。对于作为 HTTP 服务器的 CICS，端口的 TCPIP SERVICE 定义中的 SOCKETCLOSE 属性确定 Web 客户机必须完成其请求发送所需的时间。这段时间到期时，CICS 将 408（请求超时）响应返回给 Web 客户机。

如果使用分块的传输编码发送请求消息，则 CICS 在将块传递到应用程序前，会把块汇编到单个消息中。如果发送了一系列管道化的请求，则 CICS 将每个请求作为独立的事务对待，而且在使下一次请求可供下一个用户应用程序处理前，需要来自用户应用程序的响应。

CICS Application Programming Reference 提供有关 WEB RECEIVE 命令的可用选项的完整参考信息和描述。发出 WEB RECEIVE 命令时：

1. 标识您是否需要为此请求接收实体主体。
 - a. 对于某些请求方法（如 GET 方法），实体主体不适用，而且允许您的应用程序忽略存在的任何实体主体。第 199 页的附录 D，『CICS Web support 的 HTTP 方法参考』表明适用实体主体的方法。如果存在不适用的实体主体，则如果您需要，您可仍然接收它。第 58 页的『检查 HTTP 请求的请求行』告诉您如何标识请求方法。
 - b. 对于 HTTP/1.1 请求，实体主体的存在由请求上非零的 Content-Length 头（或者如果消息已分块，由 Transfer-Encoding 头）表明。如果 Content-Length 头的值为零，或者如何 Transfer-Encoding 头和 Content-Length 头都没有提供，则不存在实体主体。第 59 页的『检查消息的 HTTP 头』告诉您如何读取消息的 HTTP 头。

- c. HTTP/1.0 请求不需要指定 Content-Length 头，但是它们可能会指定。如果您在请求上找到非零的 Content-Length 头，这表明存在实体主体。如果不存在 Content-Length 头，但是请求方法（特别是 POST 方法）表明实体主体适用，则可能存在实体主体。
2. 通过指定 INTO 选项（对于数据缓冲区）或 SET 选项（对于指针引用）和 LENGTH 选项来接收实体主体。返回时，LENGTH 选项设置为所接收的数据的长度。
3. 如果要限制从实体主体接收的数据量，则指定 MAXLENGTH 选项。
 - a. 如果要只接收第一部分实体主体，并废弃超出此长度的任何数据，则省略 NOTRUNCATE 选项。这是缺省值。
 - b. 如果要保留（而不是废弃）超出此长度的任何数据，则指定 NOTRUNCATE 选项。可以使用更多 WEB RECEIVE 命令来获取任何剩余数据。

如果已使用分块的传输编码发送数据，则 CICS 将块组装到单个消息中，然后再将它传递到应用程序，因此 MAXLENGTH 选项应用到已分块消息的整个实体主体长度，而不是应用到每个独立的块。对于一条消息，CICS 接受的数据总量由 TCPISERVICE 定义的 MAXDATALEN 属性限制。

4. 指定您要在此处设置的任何选项以执行代码页转换。
 - a. SERVERCONV 选项提供代码页转换的全部控制。使用它指定是否进行代码页转换。对于作为 HTTP 服务器的 CICS，考虑到与早期发行版中已编码的支持 Web 的应用程序的兼容性，如果未指定 SERVERCONV 但指定了另一个代码页转换选项，则会进行代码页转换。如果您要阻止代码页转换，则指定 SERVERCONV(NOSRVCONVERT) 或省略所有代码页转换选项。

注：如果您接收已打包或已压缩的实体主体，如消息的 Content-Encoding 头表明的，则确保抑制了代码页转换。CICS 不对这些消息类型进行解码，并且如果应用代码页转换，结果将不可预测。如果您无法对已打包或已压缩的实体主体进行解码，则可以通过返回 415 状态码告知 Web 客户机。

- b. 如果您希望进行代码页转换，但 CICS 无法确定 Web 客户机的字符集，则使用 CHARACTERSET 选项指定它。对于较早的 Web 客户机，请求头可能不提供该信息。在这种情况下，CICS 采用 ISO-8859-1 字符集，因此，您只需在所采用的字符集不正确的情况下指定字符集。
- c. 如果您希望进行代码页转换，但本地 CICS 区域的缺省代码页（由 LOCALCCSID 系统初始化参数指定）不适合于您的应用程序，则使用 HOSTCODEPAGE 选项指定备用主机代码页。

对于指定非文本介质类型的消息，不进行代码页转换（除非您不指定 SERVERCONV，在这种情况下，出于兼容目的，会不考虑介质类型）。注意，出于兼容目的，如果入站消息不指定介质类型，CICS 会偏离 HTTP/1.1 要求而缺省为 application/octet-stream。CICS 改为使用 text/plain 作为缺省值，以便可以为消息执行代码页转换。

5. 如果指定了 MAXLENGTH 和 NOTRUNCATE，且有更多数据要接收，则发出更多 WEB RECEIVE 命令。使用 SET 选项但不带 MAXLENGTH 选项的一条 RECEIVE 命令接收所有剩余的数据，而不管它的长度有多长。另外，您可以使用具有 NOTRUNCATE 选项的一系列 RECEIVE 命令来接收适当块中的剩余数据。持续发出 RECEIVE 命令，直到不再获取 LENGERR 响应。请记住，如果接收的长度比 MAXLENGTH 选项上请求的长度短，这并不必然地表明数据的结束；如果 CICS 需要避免返回数据末尾的部分字符，则可能发生这种情况。

为响应写 HTTP 头

对于应用程序创建的动态响应，根据用于消息的 HTTP 协议版本，CICS 自动提供基本消息所需的 HTTP 头。您可能需要将更多 HTTP 头添加到您的响应。

如果消息需要某些 HTTP 头，则 CICS 将自动创建它们。您的应用程序不需要写这些头。CICS 创建的完整头列表如下：

- Connection
- Content-Length
- Content-Type
- Date
- Expect
- Host
- Server
- TE (由 CICS 写，但是可能添加了更多实例)
- Transfer-Encoding
- User-Agent
- WWW-Authenticate

请注意，只有 CICS 是 HTTP 客户机时，这些头中的一部分才是适用的且会被创建。创建这些头的情况在第 185 页的附录 B，『CICS Web support 的 HTTP 头参考』中描述。如果您在响应中写这些头，则 CICS 不覆盖它们，但会使用您应用程序提供的版本。

CICS 在发送响应时提供的头是通常应该为基本消息所写的、以使该消息遵从相应 HTTP 协议规范的头。您可能为了诸如以下目的要将更多 HTTP 头添加到响应：

- 高速缓存和文档到期的控制（例如，Cache-Control、Expires 和 Last-Modified）
- 内容协商（例如，Accept-Ranges、Vary）
- Web 客户机的信息（例如，Title、Warning 和更多 Content 头）

如果您的应用程序正在执行复杂操作，或者如果您的响应选择某些状态码，则您所遵从的 HTTP 规范可能要求将特定的 HTTP 头用于您的消息。当将任何 HTTP 头添加到响应时，检查一些重要需求所遵从的 HTTP 规范是否应用于那些头。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

在您发出 WEB SEND 命令以发生消息前，为消息编写其他 HTTP 头。此规则的特例是您编写的头要在已分块消息上作为尾部头发送，在此情况下，应用下面提到的特殊进程。为响应写 HTTP 头时：

- 对您要添加到消息的每个头使用 WEB WRITE HTTPHEADER 命令。确保以您正在遵从的 HTTP 规范所描述的格式为每个头指定名称和值。（CICS 不验证 HTTP 头的内容，这是因为它可能要使用新的或用户定义的头。）该命令添加单个头，您可以重复该命令以添加更多头。如果您编写已经写好的头，则除了现有的头之外，CICS 还会将新头添加到请求或响应。确保只在 HTTP 规范声明可以重复头的地方执行此操作。CICS 发送请求时，会存储准备添加到该请求的头。
- 如果您使用的任何 HTTP 头可能不适合低于 HTTP/1.1 级别的 Web 客户机，则写那些头前，请检查 Web 客户机已提供给您的 HTTP 版本信息。使用 WEB EXTRACT

命令来获取此信息。为了允许您使用用户定义的（非标准）头，CICS 不会除去用户所写的不适用的头。低于 HTTP/1.1 级别的服务器无法理解某些 HTTP 头，并且会导致处理您的请求时出错。

注：CICS 对低于 HTTP/1.0 级别的服务器或 Web 客户机不提供任何特殊支持。CICS 运作时假定它们都处于 HTTP/1.0 级别，并将 HTTP/1.0 作为 HTTP 版本返回。

- 如果要产生要在某个 HTTP 头（例如，Last-Modified 头）中使用的日期和时间戳记，则可以使用 FORMATTIME 命令。该命令将当前日期和时间（ABSTIME 格式的）或应用程序产生的日期和时间转换为 RFC 1123 格式。RFC 1123 格式是 CICS 为了在 Web 上使用而创建的唯一日期和时间戳记格式。其他日期和时间戳记格式可能不被与 CICS 通信的某些 Web 客户机或服务器所接受。
- 如果正在使用分块的传输编码发送 HTTP 请求或响应，并且要将尾部头包括在分块消息的末尾，则按第 67 页的『使用分块的传输编码发送 HTTP 请求或响应』中的特殊指示信息进行操作。在发送消息的第一块前，需要写尾部头。继 WEB SEND 命令后为第一块写的所有 HTTP 头作为尾部头来对待。
- 确保您的应用程序执行用户所写的头暗示的任何操作。例如，如果已写内容协商头，则应用程序需要提供不同版本的资源。

为 HTTP 消息产生实体主体

支持 web 的应用程序可以产生从 CICS 文档或数据缓冲区形成的实体主体。

CICS 文档可以用作 HTTP 消息的实体主体。它们是使用 EXEC CICS DOCUMENT 命令创建的。它们可以由应用程序和文档模板（定义为 CICS 资源或由另一个 CICS 程序创建的文档的一部分）直接指定的数据植入。文档和文档模板可以存储以供再次使用。

您还可以指定应用程序创建的一缓冲区的数据。您可能发现此选项对于简短或简单的实体主体来说更方便，而且它是允许您使用消息的已分块转换编码的唯一选项。然而，以此方式创建的数据无法那么容易地存储以供再次使用。

1. 要创建 CICS 文档，遵循 *CICS Application Programming Guide* 中的指示信息。该文档是使用 EXEC CICS DOCUMENT 应用程序编程接口（EXEC CICS DOCUMENT CREATE, INSERT 和 SET 命令）创建的。WEB SEND 命令中的 DOCTOKEN 选项用作为完成的文档指定文档令牌。根据您在 WEB SEND 命令上指定的选项，CICS 检索文档并执行适当的代码页转换。无法从 CICS 文档形成分块的消息体。
2. 或者，组装您的应用程序中的消息体。WEB SEND 命令中的 FROM 选项用于指定数据的缓冲区。对于数据缓冲区的大小未设置最大值限制，但是您需要考虑实际上可能限制其大小的以下因素：
 - CICS 区域的 EDSA 限制。
 - 在 CICS 区域中可以同时汇编的其他消息体数。调度约束可能由应用到 CICS Web support 事务的任何事务类定义的 MAXACTIVE 设置实施。
 - 用于消息体的代码页转换类型。对于从 EBCDIC 代码页 037 转换到 ASCII 代码页 ISO-8859-1，CICS 覆盖数据的相同缓冲区，因此不使用额外的存储器。对于任何其他类型的代码页转换，CICS 需要额外的存储器以包含转换的消息体。根据使用的字符集，此额外存储区域的大小范围可以是：从与原始消息体相同的大小，到理论上最大为原始消息体大小的四倍（这不太可能）。例如，使用 FROM 选项

发送的 2MB 缓冲区的数据将需要总数至少为 4MB 的存储器。双字节字符集 (DBCS) 或多字节字符集可能需要此范围中较大的存储区域。

从作为 HTTP 服务器的 CICS 发送 HTTP 响应

使用 WEB SEND 命令使 CICS 汇编 HTTP 响应的 HTTP 头、实体主体、状态码和原因短语，执行代码页转换，并将响应传输到 Web 客户机。

发出 WEB SEND 命令前，使用 WEB WRITE HTTPHEADER 命令写响应的任何其他 HTTP 头，如第 64 页的『为响应写 HTTP 头』中所描述。还将产生消息所需的任何实体主体，如第 65 页的『为 HTTP 消息产生实体主体』中所描述。

您需要在 WEB SEND 命令中指定状态码和原因短语。第 12 页的『状态码和原因短语』说明这些是什么。第 191 页的附录 C，『CICS Web support 的 HTTP 状态码参考』提供您的应用程序可能使用的状态码的概述。要规划使用状态码并查找有关它们的更多信息，应该查询您正在遵从的 HTTP 规范。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

需要的话，响应可以分块（分块的传输编码）发送。您不能将管道化的响应发送回 Web 客户机（必须对 Web 客户机发送的每个请求发送单个响应）。

CICS Application Programming Reference 提供有关 WEB SEND 命令可用选项的完整参考信息和描述。发出该命令时：

1. 根据情形，指定 STATUSCODE 选项为响应选择适当的状态码，并指定 STATUSTEXT 和 STATUSLEN 选项以提供原因短语。CICS 不验证您选择的状态码，而确保该值有效且符合 HTTP 状态码的规则是用户应用程序的职责。根据您的选择的状态码，可能需要在发出 WEB SEND 命令前完成以下某些或所有步骤：
 - a. 检查 Web 客户机请求的 HTTP 版本，以确保可理解状态码。HTTP/1.1 规范包含的状态码比 HTTP/1.0 规范的状态码多。
 - b. 如果 HTTP 规范声明状态码应该伴有特定 HTTP 头，则使用 WRITE HTTPHEADER 命令来创建那些头。
 - c. 如果 HTTP 规范声明状态码应该伴有提供特殊信息的信息体，则创建适当的实体主体。通常如果状态码表明错误或从客户机请求进一步操作时，会出现这种情况。不允许消息体具有状态码 204、205 和 304。如果您已选择不允许消息体的状态码，并尝试使用消息体，则 CICS 向 WEB SEND 命令给出错误响应。
2. 通过执行以下操作识别响应的任何实体主体的源：为已创建的 CICS 文档指定 DOCTOKEN 选项，或为已汇编的数据的主体指定 FROM 选项。如果使用 FROM 选项，则指定 FROMLENGTH 选项给出实体主体或块（如果使用分块的传输编码）的长度。对于分块的传输编码，不能使用 DOCTOKEN 选项。
3. 使用 MEDIATYPE 选项指定响应主体的介质类型。CICS 不会针对数据内容检查规范的有效性。没有缺省值。如果您未指定此选项，则 CICS 不为响应构建 Content-Type 头。
4. 如果要立即发送消息，而不是在任务结束时发送（这是缺省值），则为 ACTION 选项指定 IMMEDIATE。如果正在使用分块的传输编码，则 IMMEDIATE 是缺省值，因此不需要进行此选择。

注：一个任务期间只能发送一个响应。这可以是使用一个 WEB SEND 命令的标准响应，或使用一序列 WEB SEND 命令的分块的响应。

5. 如果要在发送响应后关闭连接，则为 CONNECTION 选项指定 CLOSE。CICS 在响应上编写 Connection: close 头，这将通知 Web 客户机：连接已关闭，且不应该再发送请求。（对于 HTTP/1.0 级别的 Web 客户机，CICS 通过省略 Connection: Keep-Alive 头达到同样的效果。）
6. 为消息体的代码页转换指定适当的设置。
 - a. SERVERCONV 选项提供代码页转换的全部控制。使用它指定是否进行代码页转换。对于作为 HTTP 服务器的 CICS，考虑到与早期发行版中编码的支持 Web 的应用程序的兼容，如果未指定 SERVERCONV 但指定了另一个代码页转换选项，则会进行代码页转换。如果您要阻止代码页转换，则指定 SERVERCONV(NOSRVCONVERT) 或省略所有代码页转换选项。
 - b. 如果您希望进行代码页转换，但 CICS 选择的字符集不合适，则使用 CHARACTERSET 选项指定一个替换字符集。缺省情况下，CICS 使用 Web 客户机的原始请求的 Content-Type 头中指定的字符集。如果该字符集不受支持或未声明，则 CICS 改为使用 ISO-8859-1 字符集。

Web 客户机可以在 Accept-Charset 头中指定替换的可接受字符集。如果您要指定这些字符集中的一个字符集，则由您的应用程序来决定分析头（可能包括质量值以表明 Web 客户机的首选项）和选择适当的支持字符集。CICS 并不支持 IANA 命名的所有字符集。第 183 页的附录 A，『HTML 编码字符集』列出了 CICS 支持用于代码页转换的 IANA 字符集。

- c. 如果您希望进行代码页转换，并要使用 FROM 选项指定消息体，则在您应用程序的代码页不是本地 CICS 区域的缺省代码页（如 LOCALCCSID 系统初始化参数中指定的）的情况下，需要使用 HOSTCODEPAGE 选项识别您应用程序的代码页。如果您使用 CICS 文档（DOCTOKEN 选项），则 CICS 从 CICS 文档域的文档主机代码页记录中识别主机代码页。

对于指定非文本介质类型的消息，不进行代码页转换（除非您不指定 SERVERCONV，在这种情况下，出于兼容目的，会不考虑介质类型）。HTTP 头和状态行总是被 CICS 转换成 ISO-8859-1 字符集。

7. 如果您要使用分块的传输编码（或分块），则除了本主题中的基本指示信息，您还需要遵循『使用分块的传输编码发送 HTTP 请求或响应』中的某些特殊指示信息。您需要确保正确按照该主题中描述的过程进行操作，从而接收方可以接受分块的消息。分块的消息是使用具有特殊选项的 WEB SEND 命令的多个实例发送的。

使用分块的传输编码发送 HTTP 请求或响应

本主题告诉您如何为以下内容设置分块的传输编码：作为 HTTP 客户机的 CICS 发出的 HTTP 请求，或来自作为 HTTP 服务器的 CICS 的 HTTP 响应。

设置分块的传输编码前，需要规划您要发送的项的以下属性：

1. 应该在消息开头使用的 HTTP 头。CICS 提供其常见消息头，这在第 185 页的附录 B，『CICS Web support 的 HTTP 头参考』中列出。对于分块的消息，CICS 为分块的传输编码提供合适的头，包括 Transfer-Encoding: chunked 头。如果消息开头需要任何其他头，则应用程序可以在进行第一个 WEB SEND 命令前编写它们。
2. 消息末尾的应该在尾部发送的任何头。这些头称为尾部头。注意，HTTP/1.1 规范设置使用尾部头的要求，即使接收方忽略它们也应该没有关系。

3. 应该如何分割消息。可以用对应用程序来说最方便的任何方式执行此操作。例如，来自许多其他应用程序的输出可以用它所产生的方式发送，或者，可以单独读取和发表表的每一行的数据。
4. 将发送的每个数据块的长度。不要包含任何尾部头的长度。

本主题中描述的过程允许您创建正确构造的分块消息，如 HTTP/1.1 规范中定义的。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP/1.1 规范的更多信息。如果未正确构造已分块消息，则接收方可能废弃它。

第 66 页的『从作为 HTTP 服务器的 CICS 发送 HTTP 响应』是编写应用程序以发送服务器响应的主要指示信息集合。第 115 页的『通过作为 HTTP 客户机的 CICS 发出 HTTP 请求』是编写应用程序以发出客户机请求的主要指示信息集合。您可以将当前主题中的指示信息与这两个指示信息集合中的任一个结合使用。

分块的消息体无法直接从 CICS 文档形成（因此不能使用 DOCTOKEN 选项）。FROM 选项必须用于指定数据以形成分块的消息体。

已开始发送部分分块的消息后，直到发送最后一个空块并完成分块的消息后，您才能发送其他消息或接收项。

1. 开始分块的消息前，请验证 Web 客户机或服务器是否为 HTTP/1.1 版本。所有 HTTP/1.1 应用程序对于理解分块的传输编码都是必需的。分块的消息不能被发送到 HTTP/1.0 接收方。
 - a. 对于作为 HTTP 服务器的 CICS 发送的响应，使用 WEB EXTRACT 命令以检查为 Web 客户机的请求指定的 HTTP 版本。
 - b. 对于作为 HTTP 客户机的 CICS 发送的请求，服务器的 HTTP 版本在连接的 WEB OPEN 命令上返回。
2. 必要的话，多次使用 WRITE HTTPHEADER 命令以写任何 HTTP 头，这些 HTTP 头应该在消息体之前发送。不要为分块的传输编码写头；CICS 会使用应用程序提供的块长度信息自己写这些头。
3. 如果要尾部头（在消息体之后发送的头）与分块的消息包含在一起，则使用 WRITE HTTPHEADER 命令编写 Trailer 头：将您打算在尾部中发送的所有 HTTP 头的名称指定为尾部头的值。除了 Transfer-Encoding、Trailer 和 Content-Length，您可以发送任何头作为尾部头。
 - a. 对于作为 HTTP 服务器的 CICS 发送的响应，您需要确保 Web 客户机在它的请求中已发送 TE: 尾部头。该头表明客户机了解尾部头。当客户机未发送 TE: 尾部时，CICS 向 WRITE HTTPHEADER 命令返回 RESP2 值为 6 的 INVREQ 响应。或者，您可使用 READ HTTPHEADER 命令检查是否存在 TE: 尾部头。
 - b. 对于作为 HTTP 客户机的 CICS 发送的请求，可以包含尾部头而无需引用 TE 头。

在分块的发送过程期间，写尾部头本身。

4. 使用 WEB SEND 命令发送消息的第一个块。
 - a. 指定 CHUNKING(CHUNKYES) 以告诉 CICS 这是一个消息块。
 - b. 使用 FROM 选项指定来自消息体的第一块数据。
 - c. 使用 FROMLENGTH 选项指定块的长度。

- d. 对于作为 HTTP 客户机的 CICS 发出的请求，必须在 METHOD 选项中指定适当的方法。分块的传输编码与没有消息体的请求无关，因此它与 GET、HEAD、DELETE、OPTIONS 和 TRACE 方法无关，但是它可以用于 POST 和 PUT 方法。
 - e. 指定应用到分块和非分块消息的任何其他选项，如您的主要指示信息集中所提供。例如，如果此分块的消息是您要发送到此服务器或 Web 客户机的最后一条消息，则指定 CLOSESTATUS(CLOSE) 选项。
5. 必要的话，多次使用 WEB SEND 命令以发送每个剩余消息块。在每个 WEB SEND 命令中，只指定以下项：
 - a. CHUNKING(CHUNKYES)。
 - b. FROM 选项，它给定数据块。
 - c. FROMLENGTH 选项，它给定块的长度。

不要为该命令指定任何其他选项。当您发出该命令时，CICS 发送每个块。

6. **可选：**为第一个块发出 WEB SEND 命令之后，但为最后一个空块发出 WEB SEND 命令之前的任何时候（请参阅下一步），使用 WRITE HTTPHEADER 命令创建应该作为尾部头发送的更多 HTTP 头。假如尾部头已写在第一块消息上，则分块的发送过程期间写的 HTTP 头被 CICS 当作尾部头看待，且它们与最后一个空块一起发出。（如果未写尾部头，则 CICS 不允许写任何尾部头。）注意，CICS 不检查您的尾部头是否与您在第一块消息的初始 Trailer 头中指定的名称匹配。
7. 当已发送最后一个数据块时，再指定一个带 CHUNKING(CHUNKEND) 但不带 FROM 或 FROMLENGTH 选项的 WEB SEND 命令。CICS 然后生成空块并将它发送给接收方以完成分块的消息。空块与包含您所写的任何尾部头的尾部一起发送。
8. 对于作为 HTTP 服务器的 CICS，如下所示处理错误：
 - a. 如果序列中的某个 WEB SEND 命令失败，则返回错误响应并且以后的发送也将失败。应用程序应对这种情况进行相应的处理。
 - b. 如果成功发送所有块，但应用程序未发出带 CHUNKING(CHUNKEND) 的最终 WEB SEND 命令，则异常终止事务，且带有异常终止代码 AWBP。这是必然出现的情况，因为 CICS 无法保证分块的消息是完整的并且是正确的，因而无法代表应用程序发出最终的空块。

接收方应忽略不完整的分块消息并废弃它。Web 客户机会决定是否重试请求。

9. 对于作为 HTTP 客户机的 CICS，如下所示处理错误：
 - a. 如果您的应用程序在已分块 transfer-coding 过程的任何时候获知到错误，则使用 WEB CLOSE 命令停止该过程并关闭连接。服务器将不接收最终的空块，因此应忽略和废弃迄今为止您已发送的数据。您可以决定是否重试请求。
 - b. 如果您不发送最终空块或发出 WEB CLOSE 命令，则在终止对 CWBO（CICS Web support 消息的瞬时数据队列）的任务时写警告消息。服务器应使接收超时并忽略和废弃您已发送的数据。

跨 HTTP 请求序列管理应用程序状态

CICS 为 Web 客户机发出的每个请求启动新的别名事务和新的程序。这是用于以下各项的情况：管道化的请求、使用持续连接发出的请求、形成逻辑序列的请求和个别的单机请求。您需要考虑如何在请求之间管理应用程序的状态。如果需要跨请求序列，在不同的程序间或同一程序的不同实例间共享数据，则可以使用 CICS 管理的资源或使用 Web 客户机发送的请求的元素执行此操作。

为了成功完成任务需要 Web 客户机和 CICS 之间的多个请求和响应交换时，序列中的每个新步骤都由 Web 客户机启动。您可以设计 CICS 发送的响应以指导 Web 客户机和 Web 客户机的任何个人用户到下一步。例如，实体主体可以包含最终用户可用于组成下一个请求的控件（例如，链接或按钮）。然而，您不能简单地实现请求的正确序列。特殊情况下，如果发生以下情况，则规划的序列会被破坏：

- 客户机是 Web 浏览器，且最终用户输入已知 URL 以启动特殊请求，而不是选择先前响应提供的 HTML 页面中的控件。
- 通过关闭 Web 客户机或通过更改为 Web 客户机的某个替换活动，最终用户将完全放弃活动。

最终用户也可能延迟序列中任何请求的启动。

应该设计您的应用程序，使它们可以应付请求序列中的延迟或破坏。例如，如果正在跨请求序列共享数据，那么如果该请求序列未完成或过分延迟，则应该确保清除了数据。如果您的应用程序更新受保护的资源，则应该确保必须一起提交或复原的更新要在同一事务中进行。（这意味着应该设计来自 Web 客户机的单个请求以完成更新。）

应用程序的理想情形是：请求和响应的每次交换是自包含的，且完成任务的独立元素。然而，此设计并不总是可行，特别是当任务复杂时或当 Web 客户机已发送管道化的请求序列时。可能需要伪会话模型，其中应用程序的状态必须在请求间进行管理。这可以使用以下技术来安排：

- 您可以设计 Web 客户机发送的请求，以便应用程序状态或共享数据合并请求中，例如，作为 Web 客户机提交 HTML 表单时使用的请求 URL 的一部分。下一个程序可以检查请求 URL 以获取共享数据。
- 您可以使用隐藏字段在作为响应返回给 Web 客户机的 HTML 表单中存储少量的应用程序状态。当用户执行计划序列中的下一个操作时，他们发送到 CICS 的请求可以包括这些隐藏字段，下一个应用程序会找到它们并读取。
- 对于数量较大的状态，以及具有扩展生存期的状态，您可创建 CICS 管理的资源来维持应用程序的状态并传递表示该资源的令牌。CICS 提供样本状态管理程序（DFH\$WBST 和 DFH\$WBSP），它们在主存储器或临时存储队列中存储应用程序状态，并提供应用程序可用于访问该信息的令牌。令牌在伪会话中可作为 HTML 表单中的隐藏字段在程序间转换，或作为 URL 中的查询字符串在交互间转换。该技术可用于保存整个伪会话中的信息，还可用于保存最终用户和各种 CICS 应用程序间的整个扩展交互中的信息，这也许要跨几个伪会话。

第 7 章 CICS Web support 的资源定义

CICS 提供 DFHWEB 资源定义组，它包含以下 CICS Web support 资源：

- CICS Web support 任务（例如，CWBA 和 CWXN）的事务定义
- CICS Web support 实用程序，包括：
 - 缺省分析器程序 DFHWBAAX 和样本分析器程序 DFHWBADX
 - Web 出错程序 DFHWBEP 和 DFHWBERX
- 临时存储模型 DFHWEB

CICS Web support 消息的瞬时数据队列 CWBO（对于大多数消息）和 CWBW（警告头消息的独立队列）在组 DFHDCTG 中。

您需要为您要执行的每个 CICS Web support 任务创建某些其他资源定义。第 45 页的第 5 章，『规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构』具有规划指导，它指定您执行每个任务所需的资源定义。您可能需要设置的资源定义如下：

- 创建 TCPIP SERVICE 资源定义以定义每个端口，您使用这些端口接收用于 CICS Web support 的入站 HTTP 请求。此资源定义是您指定为每个端口应用的安全措施的位置，具有端口操作的技术信息。『为 CICS Web support 创建 TCPIP SERVICE 资源定义』将告诉您如何执行此操作。
- 可选：为您要用于入站 HTTP 请求处理的任何别名事务创建 TRANSACTION 资源定义。第 74 页的『为 CICS Web support 创建 TRANSACTION 资源定义』将告诉您如何执行此操作。
- 创建 URIMAP 资源定义为以下各项提供处理信息：作为 HTTP 服务器的 CICS 的每个 HTTP 请求，以及从作为 HTTP 客户机的 CICS 发出的每个 HTTP 请求。
 1. 对于作为 HTTP 服务器的 CICS 的所有 HTTP 请求，按第 75 页的『为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义』中的步骤执行，以开始定义。
 2. 对于提供了应用程序生成的响应的 HTTP 请求，按第 77 页的『为 CICS（作为 HTTP 服务器）的 HTTP 请求的应用程序响应完成 URIMAP 定义』中的步骤执行。
 3. 对于提供了静态响应的 HTTP 请求，按第 77 页的『为作为 HTTP 服务器的 CICS 的 HTTP 请求的静态响应完成 URIMAP 定义』中的步骤执行。
 4. 对于从作为 HTTP 客户机的 CICS 发出的 HTTP 请求，按第 123 页的『通过作为 HTTP 客户机的 CICS 创建 HTTP 请求的 URIMAP 定义』中的步骤执行。
- 如果您需要创建由作为 HTTP 客户机的 CICS 使用的 URIMAP 资源定义，执行第 123 页的『通过作为 HTTP 客户机的 CICS 创建 HTTP 请求的 URIMAP 定义』中的步骤。

为 CICS Web support 创建 TCPIP SERVICE 资源定义

TCPIP SERVICE 资源定义用于定义端口和 CICS 服务（包括 CICS Web support）之间的关联。为您用于 CICS Web support 的每个端口定义和安装 TCPIP SERVICE 资源定义。

CICS 系统中活动的每个 TCPIP SERVICE 定义必须指定唯一的端口号。CICS 在端口接收入站 TCP/IP 连接请求时，它将 TCPIP SERVICE 定义用于该端口以确定应该调用哪个 CICS 服务。PROTOCOL 属性用于识别服务。为标准 CICS Web support 指定 HTTP，而为使用 CICS Web support 处理的非 HTTP 请求指定 USER。（剩余协议是 IIOF 和 ECI。）

对于 CICS Web support，您通常需要为用于因特网服务的缺省或熟知端口号创建 TCPIP SERVICE 定义。对于 HTTP，缺省端口号为 80，而对于 HTTPS，缺省端口号为 443。您还可以使用非标准端口号。

每个 TCPIP SERVICE 定义只能为 Web 连接任务指定一个分析器程序和一个事务定义。如果您需要使用这些项中的多个项，则需要使用不同的 TCPIP SERVICE 定义并因而使用不同的端口。

CICS 为组 DFH\$\$SOT 中的 CICS Web support 提供样本 TCPIP SERVICE 定义：

HTTPNSSL

不具有 SSL 支持的 CICS Web TCPIP SERVICE

HTTPSSL

具有 SSL 支持的 CICS Web TCPIP SERVICE

重要：TCPIP SERVICE 资源定义是指定应用于每个端口的安全措施的地方。您可以选择是否使用 SSL，并且如果您使用 SSL，则需要选择应用的确切安全措施（例如，认证方法、通过客户机和服务器发送证书以及消息的加密）。阅读第 107 页的第 12 章，『CICS Web support 的安全性』以获取有关您可用于使您的 CICS Web support 工具安全的安全功能的更多信息。

CICS Resource Definition Guide 具有有关资源定义的不同方法的信息，以及有关您将在此过程中使用的所有 TCPIP SERVICE 资源定义属性的完整参考信息。

1. 标识您要用于 CICS Web support 的 TCP/IP 端口。建议您保留端口号以供 CICS Web support 使用。第 42 页的『保留 CICS Web support 的端口』具有有关端口用法的注意事项。
2. 使用 *CICS Resource Definition Guide* 中列出的某个方法，开始具有名称和您的一组选项的 TCPIP SERVICE 定义。当您在此端口为入站 HTTP 请求设置 URIMAP 定义时，您将需要指定 TCPIP SERVICE 定义的名称。
3. 使用 STATUS 属性以指定 CICS 是否应该在安装定义后立即开始侦听此服务。如果您指定 CLOSED，则需要可以在使用服务之前，将它设置为打开。您可以使用 CEMT 事务或 SET TCPIP SERVICE 系统编程命令将服务设置为打开或关闭。
4. 指定 PORTNUMBER 属性作为此定义适用的 TCP/IP 端口的号码。
5. 使用 IPADDRESS 属性指定 TCPIP SERVICE 将侦听入局连接的点分十进制 IP 地址。另外，为了配置多个 IP 堆栈，您可以指定 INADDR_ANY 以使 CICS 尝试绑定到定义它的每个堆栈上的端口。有关此 TCPIP SERVICE 资源定义属性的参考信息详细说明了某些其他注意事项，如果您希望多个 CICS 区域共享此 TCPIP SERVICE 定义，或者如果您希望多个 CICS 区域绑定到它指定的端口号，则这些注意事项很重要。
6. 使用 DNSGROUP 和 GRPCRITICAL 属性指定服务在综合系统域中使用的 DNS 组名，以及该服务的关键状态。此信息允许 CICS 注册到工作负载管理器，以优化

DNS 连接。 *Java Applications in CICS* 中的 “Domain Name System (DNS) connection optimization” 具有有关此区域的更多信息。

7. 使用 `PROTOCOL` 属性指定 CICS Web support 在此端口上处理请求。
 - a. 对于常规 HTTP 请求，指定 HTTP。如果您指定端口 80 或 443，则 CICS 执行此操作。此选项适用带 SSL 的 HTTP 和不带 SSL 的 HTTP。SSL 选项指定是否包含 SSL。
 - b. 为使用 CICS Web support 处理的非 HTTP 请求指定 USER。指定 USER 后，CICS Web support 工具用于处理请求，但是对于使用此协议发送和接收的消息，不执行验收检查。请求标记为非 HTTP 并直接传递到分析器程序。URIMAP 定义不用于这些请求。
8. 将 `TRANSACTION` 属性指定为 Web 连接任务的 4 个字符标识，它通常是用于 HTTP 请求的 `CWXN`，或用于非 HTTP (USER 协议) 请求的 `CWXU`。此任务处理请求的初始处理。如果您指定端口 80 或 443，则 CICS 提供 `CWXN` 作为缺省值。如果为了计算或监控目的，则您可以指定 `CWXN` 或 `CWXU` 的别名，它们都必须执行程序 `DFHWBXN`。
9. 指定 `URM` 属性作为与此 `TCPIPService` 定义关联的分析器程序的名称。对于非 HTTP (USER 协议) 请求，总是使用分析器程序。对于 HTTP 请求，如果 `URIMAP` 定义指定使用分析器程序，或者如果不存在 `URIMAP` 定义，则分析器程序用于解释该请求。必须指定分析器程序。只能为每个 `TCPIPService` 定义选择一个分析器程序，但是您可以编码它以处理任何请求。如果您打算使用 `URIMAP` 定义处理您所有的 HTTP 请求，第 91 页的第 10 章，『分析器程序』会告诉您关于您的分析器程序必须提供的基本支持的信息。第 45 页的第 5 章，『规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构』中的体系结构指导可帮助您确定是否需要将分析器程序用于任何特定的 HTTP 请求。
10. 使用 `SOCKETCLOSE` 属性指定 CICS 在发出接收套接字上的入局数据后并在关闭该套接字前应该等待多久。NO 意味着套接字保持打开状态直到接收到数据为止或直到 Web 客户机关闭它为止。为了不让套接字被速度较慢的 Web 客户机或连接中断的 Web 客户机阻塞，您应指定超时值而不指定 NO。在进行连接后 Web 连接任务发出的第一条接收命令上将忽略此超时值，并且该任务将等待 CICS 确定的一段时间 (对于 HTTP 为 30 秒) 以接收来自 Web 客户机的数据。这将阻止套接字连接一启动就关闭 (即使没有立即可用的数据)，并因此会阻止 Web 客户机上的连接复位错误。

注：对于 CICS Web support，您决不应该为 `SOCKETCLOSE` 指定零设置。`SOCKETCLOSE(0)` 意味着即使 Web 客户机请求持续连接，也无法维持它。

11. 使用 `BACKLOG` 属性指定 TCP/IP 开始拒绝来自 Web 客户机的入局请求之前可排队的连接数。缺省值为 1。
12. 使用 `MAXDATALEN` 属性指定此连接上可以接收的最长数据长度。缺省值是 32K，而最大值是 524288K。此选项有助于防止在涉及大量数据传输时受到拒绝服务的攻击。
13. 使用 `TSQPREFIX` 属性指定由 CICS Web support 创建的临时存储队列的前缀。临时存储队列用于存储请求行、状态行，以及入站和出站消息的 HTTP 头。您指定的前缀必须符合现有 `TSMODEL` 定义。CICS 提供的缺省定义是 `DFHWEB`。第 41 页的第 4 章，『配置 CICS Web support 基本组件』说明如何为此目的设置 `TSMODEL` 定义。

注: 对于 CICS Web support, 不支持共享临时存储队列, 并且建议不使用可恢复的临时存储队列。

14. 使用 SSL 属性指定是否将安全套接字层 (SSL) 用于此端口。 YES 意味着使用 SSL, 且 CICS 将服务器证书发送到 Web 客户机。 CLIENTAUTH 意味着使用 SSL, 以及除了 CICS 将服务器证书发送到 Web 客户机, 还请求 Web 客户机将客户机证书发送到 CICS。如果您指定端口号 443, 则 CICS 提供 YES 作为缺省值。第 107 页的第 12 章, 『CICS Web support 的安全性』说明如果您要使用 SSL, 需执行的操作。
15. 如果您已指定 SSL (YES) 或 SSL (CLIENTAUTH), 则使用 CERTIFICATE 属性指定 X.509 证书的标号, SSL 握手期间 CICS 将该证书用作服务器证书。如果省略此属性, 则使用 CICS 区域用户标识的密钥环中定义的缺省证书。该证书必须存储在外部安全管理器数据库中的密钥环中。第 107 页的第 12 章, 『CICS Web support 的安全性』具有关于使用这些证书的更多信息。
16. 使用 AUTHENTICATE 属性指定用于在此端口发出请求的 Web 客户机的认证级别。第 107 页的第 12 章, 『CICS Web support 的安全性』说明有关认证和身份识别的信息。
 - a. 如果发送认证或身份识别信息不需要 Web 客户机, 则指定 NO。如果客户机发送已向安全管理器注册的有效证书, 则 CICS 可以使用它。
 - b. 指定 BASIC 以使 CICS 尝试 HTTP 基本认证, 在这种情况下 CICS 从 Web 客户机请求用户标识和密码。
 - c. 指定 CERTIFICATE 以使用 SSL 客户机证书认证。Web 客户机必须发送已向安全管理器注册并与用户标识关联的有效证书。如果未接收到有效证书或证书与用户标识无关, 则拒绝连接。如果您要使用此选项, 则必须指定 SSL(CLIENTAUTH)。
 - d. 指定 AUTOREGISTER 以使用向安全管理器自动注册的 SSL 客户机证书认证。Web 客户机必须发送有效证书。如果 CICS 发现该证书尚未注册到安全管理器, 则 HTTP 基本认证用于请求用户标识和密码, 且 CICS 使用此信息来注册该证书。如果您要使用此选项, 则必须指定 SSL(CLIENTAUTH)。
 - e. 指定 AUTOMATIC 使用向安全管理器自动注册的 SSL 客户机证书认证 (关于 AUTOREGISTER 选项), 或如果未发送证书, 则使用 HTTP 基本认证 (关于 BASIC 选项)。

为 CICS Web support 创建 TRANSACTION 资源定义

TRANSACTION 资源定义用于为 CICS Web support 定义别名事务。别名事务处理 HTTP 请求的稍后处理阶段, 包括接收请求、执行应用程序的业务逻辑、HTTP 响应的构造和 HTTP 响应的代码页转换。别名事务也可用于处理非 HTTP 请求。

CICS 为缺省别名事务 CWBA 提供资源定义。您可以为了以下目的想使用备用别名事务名:

- 审计、监控或记帐
- 对安全性的资源和命令检查
- 分配启动优先级
- 分配 DB2® 资源
- 将不同的失控值指定给不同的 CICS 应用程序

- 事务类限制

您可以根据需要设置足够数量的别名事务定义。您可以使用 URIMAP 定义或分析器程序指定特殊请求需要的别名事务。

重要: 确保用于应用程序生成的响应（如 CWBA）的别名事务的优先级等于或高于与 Web 连接任务（如 CWXN 或 CWXU）关联的事务的优先级。 *CICS Performance Guide* 说明为何这很重要。

CICS Resource Definition Guide 具有此类资源定义的完整指示信息。当您正在按这些指示信息进行操作时，请记住：

- 在 CWBA 定义的基础上建立您的别名事务定义，进行您需要的任何更改，例如对优先级的更改。 CWBA 的定义如下：

```
DEFINE TRANSACTION(CWBA)  GROUP(DFHWEB)
                           PROGRAM(DFHWBA)  TWASIZE(0)
                           PROFILE(DFHICST)  STATUS(ENABLED)
                           TASKDATALOC(BELOW)  TASKDATAKEY(USER)
                           RUNAWAY(SYSTEM)  SHUTDOWN(ENABLED)
                           PRIORITY(1)  TRANCLASS(DFHTCL00)
                           DTIMOUT(NO)  INDOUBT(BACKOUT)
                           SPURGE(YES)  TPURGE(NO)
                           RESSEC(NO)  CMDSEC(NO)
```

- 您的别名事务定义必须使用 CICS 提供的别名程序 DFHWBA。 别名程序调用您已为处理请求而指定的用户应用程序。
- 您的别名事务定义必须是本地事务。

为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义

URIMAP 资源定义用于定义如何处理 HTTP 请求。对于作为 HTTP 服务器的 CICS 的任何 HTTP 请求，通过为 Web 客户机的请求指定期望的 URL 的组成部分（规则、主机和路径）和其他基本信息来启动 URIMAP 定义。

如果您还未计划如何向作为 HTTP 服务器的 CICS 的 HTTP 请求提供响应，第 45 页的『用支持 Web 的应用程序提供动态 HTTP 响应』和第 48 页的『用 CICS 文档模板或 HFS 文件提供静态 HTTP 响应』会告诉您如何操作。

CICS Resource Definition Guide 具有有关资源定义的不同方法的信息，以及有关您将在此过程中使用的所有 URIMAP 资源定义属性的完整参考信息。

1. 识别您计划作为来自 Web 客户机的 HTTP 请求而接收的 URL。此 URL 表示您计划通过 CICS 可用于 Web 客户机的资源。
2. 将 HTTP 请求的 URL 划分成它的规则、主机和路径部分。第 7 页的『URL 的组成部分』说明这些每一个组成部分以及如何对它们定界。例如，在 URL `http://www.research.ibm.com/software/http/cics/index.html` 中：
 - 规则部分是 `http`
 - 主机部分是 `www.research.ibm.com`
 - 路径部分是 `/software/http/cics/index.html`

如果您希望 URIMAP 定义与多个路径匹配，则可以在路径结束处使用星号作为通配符。例如，指定路径 `/software/http/cics/*` 会使 URIMAP 定义与路径以字符串 `/software/http/cics/` 开始的所有请求匹配。如果多个带通配符的 URIMAP 定义与 HTTP 请求匹配，则采用最特定的匹配。

3. 如果查询部分出现在 URL 中，并且您希望 URIMAP 定义只与该特定查询匹配，则可以包含该特定查询作为 PATH 规范的一部分。可以在包含星号作为通配符的路径后面使用查询字符串，但查询字符串本身不能包含星号作为通配符。必须指定完整和精确的查询字符串。对于带有 CICS 文档模板的静态响应，可使用查询字符串选择 URIMAP 定义或可将它替换到文档模板中，但是两者无法同时发生。如果在 URIMAP 定义中不包含查询字符串，则只进行路径匹配，且为了匹配会自动忽略请求中出现的任何查询字符串。
4. 使用 *CICS Resource Definition Guide* 中列出的某个方法，开始具有名称和您的一组选项的 URIMAP 定义。
5. 使用 STATUS 属性指定应在启用状态还是在禁用状态安装 URIMAP 定义。
6. 指定 SERVER (作为 HTTP 服务器的 CICS) 的 USAGE 属性。
7. 指定 SCHEME 属性作为 HTTP 请求的 URL 的规则部分。可使用 HTTP (不带 SSL) 或 HTTPS (带 SSL)。不要包含跟随规则部分的定界符 `://`。指定 HTTP 规则的 URIMAP 接受使用 HTTP 规则或更安全的 HTTPS 规则发出的 Web 客户机请求。指定 HTTPS 规则的 URIMAP 仅接受使用 HTTPS 规则发出的 Web 客户机请求。
8. 指定 TCPIPService 属性作为 TCPIPService 定义的名称，它定义与此 URIMAP 定义相关的入站端口。如果不指定该属性，则 URIMAP 定义应用到任何入站端口上的匹配 HTTP 请求。

注： 当将 HTTPS (带 SSL 的 HTTP) 作为规则的 URIMAP 定义与 Web 客户机发出的请求进行匹配时，CICS 检查此请求使用的入站端口是否使用 SSL。如果不为此端口指定 SSL，则拒绝请求，其状态码为 403 (已禁止)。当 URIMAP 定义应用到所有入站端口时，该检查会确保 Web 客户机无法使用不安全的端口来访问安全的资源。因为不会对指定 HTTP 作为规则的 URIMAP 定义实施检查，所以 Web 客户机可使用不安全的端口或安全的 (SSL) 端口访问这些资源。

9. 如果您需要区别包含不同主机名的 URL，则指定 HOST 属性作为 HTTP 请求的 URL 的主机组成部分。不要包含端口号。IPv4 地址可用作主机名，但不支持 IPv6 地址。如果您指定单个星号作为 HOST 属性，则 URIMAP 定义与入局 URL 中的任何主机名匹配。如果您不使用多个主机名或如果您不想区别它们，则使用此选项。
10. 指定 PATH 属性作为 HTTP 请求的 URL 的路径部分，如果需要，包含星号作为通配符。您可以在路径部分开始包含或省略定界符 `/` (正斜杠)；如果您省略它，CICS 会自动提供它。如果查询部分出现在 URL 中，并且您希望只将 URIMAP 定义应用于该特定查询，则包含该特定查询作为路径的一部分 (在字符串的开始包含问号)。

注： 除非也安装了更多特定的 URIMAP 定义 (在这种情况下，会进行最特定的匹配)，否则将 PATH 属性指定为 `/*` 会使 URIMAP 定义与导向到主机 (以 HOST 属性命名) 的所有请求相匹配。

11. 现在，来完成您的定义：
 - a. 对于应用程序生成的响应，按照第 77 页的『为 CICS (作为 HTTP 服务器) 的 HTTP 请求的应用程序响应完成 URIMAP 定义』中的指示信息操作。
 - b. 对于静态响应，按照第 77 页的『为作为 HTTP 服务器的 CICS 的 HTTP 请求的静态响应完成 URIMAP 定义』中的指示信息操作。

为 CICS (作为 HTTP 服务器) 的 HTTP 请求的应用程序响应完成 URIMAP 定义

在您已通过指定期望的 URL 的组成部分(规则、主机和路径)和其他基本信息启动 URIMAP 定义时,如果您要向 HTTP 请求提供应用程序生成的响应,通过提供有关处理请求和提供 HTTP 请求的一个或多个应用程序的信息来完成此定义。

如果您还未计划如何向作为 HTTP 服务器的 CICS 的 HTTP 请求提供响应,第 45 页的『用支持 Web 的应用程序提供动态 HTTP 响应』会告诉您如何操作。然后您需要如第 75 页的『为作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义』中描述的启动您的 URIMAP 定义。

当您已规划应用程序生成的响应并开始 URIMAP 定义后,按本主题中的指示信息完成该定义。*CICS Resource Definition Guide* 具有有关资源定义的不同方法的信息,以及有关您将在此过程中使用的所有 URIMAP 资源定义属性的完整参考信息。

1. 在此请求的处理路径中,如果涉及与此 URIMAP 定义相关的 TCPIP SERVICE 定义(或多个定义)关联的分析器程序,则为 ANALYZER 属性指定 YES 以激活它。如果使用分析器程序,您仍然可以指定 CONVERTER、TRANSACTION、USERID 和 PROGRAM 属性。您为这些属性指定的值用作分析器程序的输入,但它们会被此程序重设。或者,您可保留这些属性为空白并让分析器程序指定它们。
2. 如果您要使用转换器程序,则指定 CONVERTER 属性作为该程序的名称。该程序可以是 CICS 中可用的任何转换器程序;转换器程序和 TCPIP SERVICE 定义之间没有关联,这是因为分析器程序与此定义之间有关联。如果使用转换器程序,则您仍然可以指定 PROGRAM 属性。您为此属性指定的值用作转换器程序的输入。转换器程序可以更改 PROGRAM 属性以指定不同的应用程序来处理请求。
3. 指定 TRANSACTION 属性作为 CICS 中可用的别名事务的名称,CICS 可使用此别名事务运行提供响应的应用程序。缺省别名事务是 CWBA。如果指定 ANALYZER(YES),则事务名称还可以由分析器程序更改或提供。
4. 指定 USERID 属性作为附加别名事务所凭借的用户标识。您在 URIMAP 定义中指定的用户标识会被直接从 Web 客户机获取(由连接的 TCPIP SERVICE 定义的 AUTHENTICATE 属性指定)的任何用户标识覆盖。如果指定 ANALYZER(YES),则分析器程序可以更改这些用户标识中的任意一个,或提供一个。如果未通过这些方式中的任何一种指定用户标识,则缺省用户标识为 CICS 缺省用户。
5. 指定 PROGRAM 属性作为向请求提供响应的应用程序的名称。如果 URIMAP 定义中未指定分析器程序或转换器程序,则 HTTP 请求直接被传递到此应用程序。如果指定分析器程序或转换器程序,则您可以保留此属性为空白并让分析器程序或转换器程序指定它。

为作为 HTTP 服务器的 CICS 的 HTTP 请求的静态响应完成 URIMAP 定义

对于静态响应,当已通过指定期望的 URL 的组成部分(规则、主机和路径)和其他基本信息启动 URIMAP 定义时,使用文档模板或 HFS 文件,通过提供 CICS 构造请求的静态响应所需的信息来完成此定义。您可将通配符用于路径匹配,在这种情况下 CICS 采用通配符涵盖的每个 HTTP 请求的路径部分,并将它替换为模板名称或 HFS 文件路径的最后一部分。

如果您还未计划如何向 HTTP 请求提供静态响应，第 48 页的『用 CICS 文档模板或 HFS 文件提供静态 HTTP 响应』会告诉您如何这样操作。然后您需要如第 75 页的『作为 HTTP 服务器的 CICS 的任何请求启动 URIMAP 资源定义』中描述的启动您的 URIMAP 定义。

当您已规划静态响应并开始 URIMAP 定义后，按本主题中的指示信息完成该定义。*CICS Resource Definition Guide* 具有有关资源定义的不同方法的信息，以及有关您将在此过程中使用的所有 URIMAP 资源定义属性的完整参考信息。

注：资源级安全性不应用到作为静态响应传递的项。如果您需要将基于用户标识的访问控制应用到以这种方式传递的项，则需要改为传递作为应用程序生成的响应的资料。

1. 指定 `MEDIATYPE` 属性作为 CICS 提供的静态响应的数据内容。例如，`text/html` 或 `text/xml` 分别是 HTML 和 XML 数据内容的名称。（请参阅第 7 页的『IANA 介质类型和字符集』以获取有关介质类型的更多信息。）此属性没有缺省值，并且必须指定它。CICS 使用此信息创建响应的 `Content-Type` 头。
2. 如果静态响应是从文本文档（文档模板或 HFS 文件）形成的，则指定代码页转换所需的属性。在 `MEDIATYPE` 属性指定数据内容的文本类型的情况下，才进行代码页转换。
 - a. 指定 `CHARACTERSET` 属性作为将静态响应发送到 Web 客户机之前 CICS 将它转换所至的字符集。CICS 不支持 IANA 命名的所有字符集。第 183 页的附录 A，『HTML 编码字符集』列出了 CICS 支持的 IANA 字符集。此信息包含在响应的 `Content-Type` 头中。
 - b. 将 `HOSTCODEPAGE` 属性指定为编码静态文档的 IBM 代码页（EBCDIC）。
3. 如果要使用 CICS 文档模板形成静态响应，则指定 `TEMPLATENAME` 属性作为此文档模板的名称。必须使用 `DOCTEMPLATE` 资源定义来定义文档模板。如果您要使用路径匹配，则在 CICS 文档模板的名称结束处，以及 `PATH` 属性指定的路径结束处包含星号作为通配符。CICS 采用此通配符涵盖的每个 HTTP 请求的路径部分，并将它替换为模板名称的最后一部分。*CICS Resource Definition Guide* 具有这方面的示例。

当指定 `TEMPLATENAME` 属性时，如果查询字符串出现在 URL 中，则 CICS 将查询字符串的内容作为符号列表传递到指定的 CICS 文档模板。只有 URIMAP 定义的 `PATH` 属性中还未使用查询字符串时才会发生这种情况。

4. 如果您要使用 HFS 文件形成静态响应，则指定 `HFSFILE` 属性作为此文件的标准（绝对）名称或相对名称。HFS 文件可指定为以斜杠开始的绝对或标准路径，或指定为不以斜杠开始的相对路径。相对路径相对于 CICS 区域用户标识的 HOME 目录。CICS 区域必须有访问 z/OS UNIX 的许可权，而且它必须具有访问包含文件的 HFS 目录以及文件本身的许可权。*Java Applications in CICS* 说明了如何授予这些许可权。如果您要使用路径匹配，则在 HFS 文件的路径的结束处包括作为通配符的星号，并还在 `PATH` 属性指定的路径的结束处包括星号。CICS 采用此通配符涵盖的每个 HTTP 请求的路径部分，并将它替换为 HFS 文件路径的最后一部分。*CICS Resource Definition Guide* 具有这方面的示例。

注：您不能在 `HFSFILE` 规范中单独使用星号。必须指定至少一个目录结构级别。不能将查询字符串替换成 HFS 文件。

第 8 章 管理 CICS Web support

已配置 CICS 以执行各种 CICS Web support 任务，且已开始响应来自 Web 客户机的请求后，您可能需要执行某些管理活动以管理您的 CICS Web support 结构，并在资源不可用时，对请求提供适当的处理。

通过用 URIMAP 定义来管理您的 HTTP 请求，可以使管理 CICS Web support 变得容易。URIMAP 定义允许您执行以下操作：

- 如果特定的 HTTP 请求所需的资源（例如，CICS 程序）不可用，则在运行中的 CICS 系统中动态重定向或拒绝这些请求。
- 让 CICS 创建虚拟主机，可以使用 CICS 命令来管理它们。

如果没有 URIMAP 定义，您可以在 TCPIP SERVICE 资源定义级别管理 CICS Web support，这将管理特殊端口上的所有请求，但是如果在 URIMAP 资源定义级别管理的话，控制程度和详细程度就更高。

- 您可以使用 CICS 系统编程接口和 CICS 提供的事务创建、安装、更新和删除 CICS Web support 资源，包括 TCPIP SERVICE、URIMAP 和 TRANSACTION 资源定义。
『管理 CICS Web support 资源』说明您可用于管理这些资源的命令。
- 您可使用 INQUIRE HOST 命令和虚拟主机浏览命令查看 CICS 从您的 URIMAP 定义创建的虚拟主机，并使用 SET HOST 命令更改它们的状态。第 81 页的『管理虚拟托管』将告诉您如何执行此操作。
- 如果 CICS 系统中的应用程序或资源暂时不可用，并且您要通过重定向提供另一种方式，则可以重定向由 URIMAP 定义处理的 HTTP 请求，并在资源再次可用时除去重定向。如果永久更改了资源的位置或请求 URL 格式，则可以设置永久重定向。第 82 页的『将 HTTP 请求重定向到另一个 URL』将告诉您如何执行此操作。
- 如果 CICS 系统中的应用程序或资源暂时不可用，并且您无法通过重定向提供另一种方式，或者如果已永久除去应用程序或资源，则您可以通过禁用资源定义来拒绝几个不同级别（单独请求 URL、虚拟主机、端口或所有端口）的 HTTP 请求。您可使用这些方法终止全部或部分 CICS Web support 服务。第 82 页的『拒绝 HTTP 请求』将告诉您如何执行此操作。
- CICS 记录从入站消息中的警告标题到瞬时数据队列 CWBW 的信息。通常该信息供用户阅读。第 83 页的『警告标题』会告诉您有关该信息的一些内容。

管理 CICS Web support 资源

您可以使用 CICS 系统编程接口和 CICS 提供的事务 CEMT 和 CEDA 来创建、安装、更新和删除 CICS Web support 资源。

CICS 提供的事务 CEDA 可用于创建和安装 TCPIP SERVICE、URIMAP、TRANSACTION 和 DOCTEMPLATE 资源定义。第 71 页的第 7 章，『CICS Web support 的资源定义』具有有关为 CICS Web support 设置资源定义的更多信息。

CICS 系统编程接口包括用于 CICS Web support 管理的以下命令：

INQUIRE TCPIP

查询 CICS 系统中的 TCP/IP 支持状态。该命令返回 TCP/IP 的打开状态和 CICS 区域中的当前最大 IP 套接字数。

SET TCPIP

您可以使用此命令打开或关闭 TCP/IP 支持，无论是正常情况（允许活动任务完成）还是立即（异常终止活动任务）。关闭 TCP/IP 支持意味着拒绝所有入站和出站请求，且 CICS Web support 完全停止。您还可以使用此命令增加或减少 CICS 区域中的最大 IP 套接字数。如果您没有超级用户权限，则您可以设置的限制较低，且 CICS 将通知您它是否已利用此较低限制。

CREATE TCPIP SERVICE

为端口创建 TCPIP SERVICE 定义。

DISCARD TCPIP SERVICE

删除端口的 TCPIP SERVICE 定义。可以废弃 TCPIP SERVICE 定义前，必须关闭它（使用 SET TCPIP SERVICE 命令）。

INQUIRE TCPIP SERVICE

查询 TCPIP SERVICE 定义。象显示定义的属性一样，该查询显示定义的当前 DNS 注册状态和打开状态。还可以浏览 TCPIP SERVICE 定义。

SET TCPIP SERVICE

您可以使用此命令以更改 TCPIP SERVICE 定义的请求队列限制、数据接收限制、DNS 注册状态或分析器程序。您还可以使用此命令关闭 TCPIP SERVICE 定义。您可以选择停止侦听端口的方式：正常 - 允许活动任务完成，或者立即 - 异常终止活动任务。

CREATE URIMAP

为请求创建 URIMAP 定义。

DISCARD URIMAP

删除 URIMAP 定义。由已删除定义处理的请求可能与不太特定的 URIMAP 定义匹配，该 URIMAP 定义的路径中有通配符。否则，它们将传递到 TCPIP SERVICE 定义的分析器程序。如果要拒绝请求而不进行此可选处理，则禁用 URIMAP 定义，而不是废弃它。

INQUIRE URIMAP

查询 URIMAP 定义。象显示定义的属性一样，该查询说明：是否已在个别基础上禁用了 URIMAP，或者它是否因为虚拟主机（它是该虚拟主机的一部分）已被禁用而不可用。还可以浏览 URIMAP 定义。

SET URIMAP

您可以使用此命令以启用或禁用 URIMAP 定义。禁用 URIMAP 定义后，CICS 通过 Web 出错程序将 HTTP 503 响应（服务不可用）发送到 Web 客户机。您还可以使用此命令来设置 LOCATION 和 REDIRECTTYPE 属性，以指定重定向或结束重定向。

INQUIRE HOST

查询虚拟主机。还可以浏览虚拟主机。第 81 页的『管理虚托管』说明如何管理虚拟主机。

SET HOST

启用或禁用虚拟主机。第 81 页的『管理虚托管』说明如何管理虚拟主机。

您用于 CICS Web support 的 TRANSACTION、DOCTEMPLATE 和 PROGRAM 资源也可以使用 SPI 命令来管理。《CICS System Programming Reference》具有有关所有这些命令的完整信息。

CICS 提供的事务 CEMT 包括用于 CICS Web support 管理的以下命令：

- INQUIRE TCPIP
- SET TCPIP
- INQUIRE TCPIPSERVICE
- SET TCPIPSERVICE
- INQUIRE URIMAP
- SET URIMAP
- INQUIRE HOST
- SET HOST

您用于 CICS Web support 的 TRANSACTION 和 PROGRAM 资源也可以使用 CEMT 来管理，且您可以在 DOCTEMPLATE 资源上查询。《CICS 提供的事务》具有有关所有这些命令的完整信息，并说明如何使用 CEMT。

CICSplex SM 还可以用于管理此处列出的资源。

管理虚托管

CICS 通过 URIMAP 资源定义对象支持虚托管。

您为作为 HTTP 服务器的 CICS（URIMAP 定义中具有 USAGE(SERVER)）设置的每个 URIMAP 定义，包含 Web 客户机期望在它的请求中提供的主机名。通过将 CICS 区域中指定同一主机名和同一 TCPIPSERVICE 定义的所有 URIMAP 定义一起分组成单个数据结构，CICS 自动为您创建虚拟主机。未指定 TCPIPSERVICE 并因而应用到它们全部的 URIMAP 定义被添加到指定匹配主机名的所有数据结构，因此这些 URIMAP 定义可能是多个数据结构的一部分。然后这些 URIMAP 定义组中的每一个组形成可作为单个单元管理的虚拟主机。

您可使用以下 CICS 命令管理 CICS 已从您的 URIMAP 定义创建的虚拟主机：

- INQUIRE HOST 命令用于查询虚拟主机的状态。此命令告诉您虚拟主机的主机名、它关联的 TCPIPSERVICE 定义（或如果它与 CICS 区域中的每个 TCPIPSERVICE 定义关联）以及启用它还是禁用它。
- SET HOST 命令用于将虚拟主机的状态设置为启用或禁用。禁用虚拟主机意味着应用程序不能访问构成虚拟主机的所有 URIMAP 定义。（但要注意的是，不能废弃已用这种方式禁用的 URIMAP 定义。）当禁用虚拟主机时，CICS 向 Web 客户机返回 HTTP 503 响应（服务不可用）。
- 虚拟主机浏览命令用于浏览 CICS 系统中的虚拟主机。

统计程序 DFHOSTAT 包含一个报告，它显示 CICS 已创建的虚拟主机。

如果已删除构成虚拟主机的所有 URIMAP 定义，则 CICS 自动删除虚拟主机。如果您不希望管理 CICS 已为您创建的虚拟主机，则可以忽略它们，并在您的 URIMAP 定义的级别进行管理。

您还可以使用分析器程序处理虚拟主机。将 HTTP 请求的主机名传递给分析器程序，并且您可以对程序进行编码，以向请求提供与主机相关的响应。然而，无法使用 INQUIRE HOST、SET HOST 和虚拟主机浏览命令管理以这种方式设置的虚拟主机。

将 HTTP 请求重定向到另一个 URL

可以使用 URIMAP 定义将作为 HTTP 服务器的 CICS 的 HTTP 请求重定向到另一个 URL。

您可能希望应该总是通过将 Web 客户机重定向到另一个 URL 来提供资源。而当需要的资源不可用（例如，页面告诉请求方它们需要的应用程序已脱机）时，可能要使用重定向以提供对请求的临时响应。在任一情况下，都可以使用与请求匹配的 URIMAP 定义重定向请求，具体如下：

1. 找到要重定向的 URL 的 URIMAP 定义。
2. 使用 URIMAP 定义的 LOCATION 属性指定多达 255 个字符的 URL，匹配的 HTTP 请求被重定向到该 URL。此 URL 必须是完整的 URL，包括规则、主机和路径部分。包括所有定界符。CICS 检查 URL 是否完整并是否正确限定，但 CICS 不检查目标是否有效。
3. 使用 URIMAP 定义的 REDIRECTTYPE 属性指定临时或永久重定向。当在临时基础上重定向请求时，用于响应的 HTTP 状态码是 302（已找到）。当永久重定向请求时，用于响应的 HTTP 状态码是 301（永久移动）。CICS 编写重定向响应且无法定制重定向响应。
4. 安装更改的 URIMAP 定义。当指定 REDIRECTTYPE(TEMPORARY) 或 REDIRECTTYPE(PERMANENT) 时，URIMAP 定义中的 LOCATION 属性覆盖 URIMAP 定义中的其他任何属性，并重定向 HTTP 请求。安装 URIMAP 定义后，您可使用 SET URIMAP LOCATION 命令更改 LOCATION 属性。
5. 如果资源再次变得可用时，使用命令 SET URIMAP REDIRECTTYPE(NONE) 关闭重定向并重新安装已更改的定义。会保留 LOCATION 属性中指定的 URL，但除非您重新激活重定向，否则不使用此 URL。

拒绝 HTTP 请求

如果 CICS 系统中的应用程序或资源不可用，有时您可能需要拒绝 Web 客户机向作为 HTTP 服务器的 CICS 发出的请求。

您可以在几个不同的级别拒绝 HTTP 请求：

1. 在特定请求 URL 级别。要达到此详细程度级别，URIMAP 定义应包含请求 URL。如果您没有 URIMAP 定义，则可通过更改处理 HTTP 请求的分析程序来修改这些请求的处理，但这不太方便。
2. 在虚拟主机级别（包含特定主机名的所有请求）。对于被合并到虚拟主机中的请求，它必须包含在 URIMAP 定义中。
3. 在端口级别。端口映射到 TCPIP SERVICE 定义。例如，对于缺省 HTTP 端口 80，禁用 TCPIP SERVICE 定义使 CICS 除了接收使用 SSL 或使用非标准端口的 HTTP 请求外，不会接收任何其他 HTTP 请求。
4. 完全在所有端口级别。在 CEMT 事务或 CPSM 中，您可关闭 CICS 内部 TCP/IP 套接字支持，并因而完全关闭 CICS Web support。

一般地，如果您在越详细的级别拒绝 HTTP 请求，CICS 就会向 Web 客户机给出越合适和详尽的错误响应。例如，如果您通过禁用 URIMAP 定义或虚拟主机来拒绝 HTTP 请求，则 CICS 通过 Web 出错程序将 HTTP 503 响应（服务不可用）返回到 Web 客户机。您可以定制 Web 出错程序来修改此响应。然而，如果您通过禁用 TCPIP SERVICE 定义来拒绝 HTTP 请求，则 Web 客户机将只接收表明服务器错误的一般错误响应。

- 要拒绝到特定请求 URL 的请求：
 1. 如果您有 URL 的 URIMAP 定义，则使用第 79 页的『管理 CICS Web support 资源』中描述的一个方法禁用 URIMAP 定义。检查请求 URL 是否与路径中有通配符的不特定 URIMAP 定义不匹配。CICS 通过 Web 出错程序将 HTTP 503 响应（服务不可用）返回给 Web 客户机。您可以通过更改 Web 出错程序来定制此响应。
 2. 如果您没有 URL 的 URIMAP 定义，则可拒绝请求，方法是更改与发出请求的端口的 TCPIP SERVICE 定义关联的分析器程序。您可能希望对分析器程序进行编码以为每个 URL 提供各自的拒绝消息，或可能更喜欢提供涵盖任何不可用的 URL 的一条消息。第 91 页的第 10 章，『分析器程序』告诉您哪些操作适用于处理所拒绝的请求。
- 要拒绝到虚拟主机的请求（即，到某个主机名的所有请求），如第 81 页的『管理虚拟托管』中描述的，使用 SET HOST 命令禁用虚拟主机。CICS 通过 Web 出错程序将 HTTP 503 响应（服务不可用）返回给 Web 客户机。您可通过更改该 Web 出错程序来定制此响应。
- 要拒绝特定端口上的所有请求，使用第 79 页的『管理 CICS Web support 资源』中描述的一个方法禁用 TCPIP SERVICE 定义。您可以选择停止侦听端口的方式：正常 - 允许活动任务完成，或者，立即 - 异常终止活动任务。
- 要拒绝所有入站请求和出站请求并完全停止 CICS Web support，如第 79 页的『管理 CICS Web support 资源』中描述的，使用 CEMT 事务或 CPSM 关闭 TCP/IP。您可以选择关闭的方式：正常 - 允许活动任务完成，或者，立即 - 异常终止活动任务。

警告标题

如果警告标题出现在 HTTP 消息中，则它通常包含打算供用户阅读的信息。如果 CICS Web support 接收到带警告标题的消息，则将与此标题关联的文本写到 CWBW 瞬时数据队列。

对于作为 HTTP 服务器的 CICS，用于记录请求中警告标题的消息号是 DFHWB0750，而作为 HTTP 客户机的 CICS，对于响应的警告标题，消息号是 DFHWB0752。每个警告标题的消息包含：

- 与警告标题关联的文本。
- 服务器和客户机的 IP 地址。

消息被写到 CICS 提供的瞬时数据队列 CWBW，它对于 CSSL 是间接的。在组 DFHDCTG 中，有一个队列的样本定义。

CWBO 是通常用于 CICS Web support 的队列，而提供 CWBW 是用于单独保存警告消息。如果您接收太多的警告标题或不再有用的警告标题（如服务器总是发送一个警告以响应您重复发出的客户机请求），则可以除去 CWBW 瞬时数据队列以抑制这些记录。

第 9 章 Web 出错程序

当 CICS Web support 过程中发生请求错误或异常终止时，用户可替代的 Web 出错程序将错误响应提供给 Web 客户机。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

在以下情况下使用 Web 出错程序：

- 当 CICS Web support 在初始处理来自 Web 客户机的请求时检测到问题；例如，如果请求中缺少必需的信息，或者如果信息发送太慢，而且达到接收超时。
- 已安装的 URIMAP 定义与请求匹配，但禁用 URIMAP 定义或虚拟主机，或者，禁用或找不到静态响应的资源。
- 当 URIMAP 匹配失败，且为 TCPIP SERVICE 定义指定的分析器无法处理请求并将控制传递到 Web 出错程序时。
- URIMAP 定义和分析器程序以及转换器程序都无法确定应执行什么应用程序来服务请求。
- 分析器程序、转换器程序或用户编写的应用程序中发生异常终止。这会确保即使处理已失败也会将响应返回给 Web 客户机。

在以下情形下不使用 Web 出错程序：

- 发生套接字发送错误或接收错误。在这种情况下，关闭套接字并不向 Web 客户机发送响应。
- URIMAP 指定重定向响应。这些响应由 CICS 组成，并且不能定制。
- 用户编写的应用程序已成功完成处理并要返回表明错误的响应；例如，如果客户机指定了资源不支持的方法。这些响应由应用程序编写和发送。
- 对于涉及作为 HTTP 客户机的 CICS 的处理。向服务器发送错误响应，Web 客户机不是必需的。从服务器接收的响应由客户机应用程序处理。

如果存在与客户机的持续连接，则通过 Web 出错程序发送错误响应后，CICS 保持连接打开。特例是 CICS 为响应选择 501（方法未实施）状态码，在这种情况下，CICS 关闭此连接。

CICS 中提供了两个用户可替代 Web 出错程序：

DFHWBERX (Web 出错应用程序)

DFHWBERX 可以由分析器程序或在 URIMAP 定义中指定为应用程序以处理请求。当 CICS 提供的缺省分析器 DFHWBAAX 作为 TCPIP SERVICE 定义上的分析器程序指定，且没有找到请求的匹配 URIMAP 定义时，使用它。EXEC CICS WEB 和 DOCUMENT 应用程序编程接口可以从 DFHWBERX 获取。它使用 EXEC CICS 命令以获取有关 Web 客户机请求的信息，并创建和发送错误响应。

DFHWBEP (Web 出错程序)

当 CICS 在请求处理中检测到错误时，DFHWBEP 在需要 Web 出错程序的所有其他情形中使用。DFHWBEP 使用基于通信区域的接口，其中有关错误情形

的信息传递到通信区域中的程序，且完整 HTTP 响应作为一缓冲区的数据创建。EXEC CICS WEB 和 DOCUMENT 应用程序编程接口不能从 DFHWBEP 获取。

要获取有关编写用户可替代程序的更多信息，请参阅 *CICS Customization Guide*

DFHWBERX, Web 出错应用程序

DFHWBERX 使用 EXEC CICS WEB 和 DOCUMENT 应用程序编程接口来获取有关 Web 客户机请求的信息，并创建和发送错误响应。它称为应用程序。如果请求总是需要错误响应，则 DFHWBERX 可以由分析器程序指定，或作为 URIMAP 定义中的 PROGRAM 属性指定。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

当 CICS 提供的缺省分析器 DFHWBAAX 作为 TCPIP SERVICE 定义上的分析器程序指定，且没有找到请求匹配的 URIMAP 定义时，使用 DFHWBERX。使用 wbra_server_program 输出参数，DFHWBAAX 将 DFHWBERX 设置为应用程序以处理请求。

DFHWBERX 是用户可替代的。CICS 仅为组装器中的 DFHWBAAX 提供源代码。

DFHWBERX 不使用通信区域中提供的信息，且不接收来自 CICS 的缺省 HTTP 响应。它而是使用 EXEC CICS 命令来获取有关 Web 客户机请求的信息，并创建和发送错误响应。这意味着 CICS 可以对 DFHWBERX 提供更多有关组装响应的帮助。

DFHWBERX 提供如下错误响应：

- 如果 Web 客户机的请求是介质类型为 text/xml 的 POST 请求，则假定它为 SOAP 1.1 请求，且返回 SOAP 1.1 故障响应。
- 如果请求是介质类型为 application/soap+xml 的 POST 请求，则假定它是 SOAP 1.2 请求，且返回 SOAP 1.2 故障响应。
- 所有其他请求假定为标准 HTTP 请求，因此用 404（未找到）状态码组成并返回合适的 HTTP 响应。

在 DFHWBERX 中：

- EXEC CICS WEB EXTRACT 命令用于获取 Web 客户机请求的 URL，该请求需要错误响应。
- EXEC CICS DOCUMENT 命令用于构造消息体。
- 对于 SOAP 故障响应，EXEC CICS WEB WRITE HTTPHEADER 命令用于编写合适的 SOAP 操作头。
- EXEC CICS WEB SEND 命令用于指定合适的状态码，并将响应发送给 Web 客户机。为响应主体的代码页转换指定了 UTF-8 字符集。

DFHWBEP, Web 出错程序

DFHWBEP 接收通信区域（提供有关错误情形的信息）中的参数列表，以及 CICS 打算发送到 Web 客户机的缺省 HTTP 响应（包括状态码和状态文本）。程序可定制响应或构建一个新的，并将它返回给 CICS 用于发送。DFHWBEP 是用户可替代的，并使用基于通信区域的接口，完整 HTTP 响应在该接口中作为一缓冲区的数据创建。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

第 223 页的附录 H, 『Web 出错程序 DFHWBEP 的参考信息』具有通信区域中传递到 DFHWBEP 的所有参数的列表和技术描述。

传递到 DFHWBEP 的参数列表包含 CICS 已在缺省响应中使用的三位数 HTTP 状态码。参数列表还提供识别错误情况的信息（如错误代码、异常终止代码、消息号、响应码和原因码以及发生了错误的程序的名称）。如果定制 DFHWBEP, 则确保您正在使用合适范围的输入参数来识别应用定制响应的情形, 而不只依赖状态码。CICS Web support 可以为了各种目的使用每个状态码。任何具有您的程序不了解的状态码的 HTTP 响应都应未经更改地传递。

参数列表还包含指向缓冲区 (wbep_response_ptr) 的指针, 该缓冲区包含 CICS 为检测到的错误返回的缺省 HTTP 响应以及响应长度 (wbep_response_len)。该缓冲区包含完整的 HTTP 响应, 这包括状态行、HTTP 头和消息, 并且此长度是完整的响应消息的长度。Web 出错程序可以:

- 保持响应未经更改。
- 修改要返回的响应, 并相应地更新 wbep_response_len 中的长度。
- GETMAIN 一个新的缓冲区, 构建新的 HTTP 响应, 并且回传 wbep_response_ptr 中的新缓冲区的地址以及 wbep_response_len 中新响应的长度。

不能从 DFHWBEP 获得 EXEC CICS WEB 和 DOCUMENT 应用程序编程接口。DFHWBEP 在通信区域中创建响应, 因此完整的 HTTP 响应创建为单个数据缓冲区。

可以修改响应中的所有项, 但注意以下个别项的指导:

HTTP 版本 (HTTP/1.1 或 HTTP/1.0)

这由 CICS 根据 Web 客户机的 HTTP 版本确定。不要修改响应的这个元素。

数字状态码 (例如, 404 或 500)

修改响应的这个元素时要谨慎。第 191 页的附录 C, 『CICS Web support 的 HTTP 状态码参考』列出了 CICS Web support 使用的状态码以及使用它们的原因。HTTP/1.1 规范具有有关所有状态码和正确使用它们的需求的更多信息。如果您确定有另一种状态码比 CICS Web support 选择的 状态码更适合, 则确保您的用法遵循 HTTP/1.1 规范中的需求。特别要检查状态码是否适合 Web 客户机的 HTTP 版本。对于非 HTTP 错误, CICS 总是使用 400 状态码。

原因短语或状态文本 (例如, 未找到)

您可以修改响应的这个元素。推荐使用 HTTP/1.1 规范建议的原因短语 (例如, “未找到”或“错误请求”), 但这些短语是可选的。HTTP/1.1 规范声明: 每个状态码的原因短语可以由本地同等短语替换。

HTTP 头

传递到 DFHWBEP 的缺省响应包含 CICS 已为响应编写的头 (例如, Date 和 Server 头)。CICS 编写的头适合于消息的 HTTP 版本, 且不应该除去, 这是因为可能为了遵循 HTTP 规范而需要它们。第 185 页的附录 B, 『CICS Web support 的 HTTP 头参考』列出 CICS 可以编写的头。如果合适, 可以为响应添加更多 HTTP 头。检查 HTTP/1.1 规范是否允许在此上下文中使用头。如果已选择不同的状态码, 则 HTTP/1.1 规范可能需要某些头。

消息体

缺省消息体重复发行版中给出的状态码和原因短语。您可以修改响应的这个元素。对于许多状态码，消息体可用于向用户提供更多信息。某些状态码不能与消息体一起出现。

到 Web 出错程序的输入

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

第 223 页的附录 H, 『Web 出错程序 DFHWBEP 的参考信息』具有通信区域中传递到 DFHWBEP 的所有参数的列表和技术描述。

DFHWBEP 的输入参数是:

- 标识原始错误原因的错误代码 (wbep_error_code)。DFHWBUCC 副本列出这些代码。
- 发生错误时进行的处理类型 (服务器或管道)。
- 发生错误的程序名称。
- 与错误 (wbep_abend_code) 关联的 CICS 异常终止代码。
- 与错误关联的 CICS 消息号, 以及指向消息文本的指针 (wbep_message_number)。
- 分析器程序或转换器程序 (如果使用) 返回的响应码和原因码。
- 接收到请求的端口的 TCPIP SERVICE 资源定义名。(它标识分析器程序 (如果使用) 的名称。)
- 所使用的转换器程序的名称。
- 目标程序 (即, 目的在于向请求提供响应的应用程序) 的名称。
- CICS 要向 Web 客户机返回的 3 位 HTTP 状态码 (wbep_http_response_code)。第 191 页的附录 C, 『CICS Web support 的 HTTP 状态码参考』列出了 CICS Web support 使用的状态码以及使用它们的原因。
- 指向缓冲区的指针, 而该缓冲区包含要向 Web 客户机返回的缺省 HTTP 响应。它是一个完整的 HTTP 响应, 包括状态行、HTTP 头和消息体。
- 要向 Web 客户机返回的缺省 HTTP 响应的长度。响应缓冲区的最长长度是 32K。
- 此请求的作为 HTTP 服务器的 CICS 的点分十进制 IP 地址。
- Web 客户机的点分十进制 IP 地址。

来自 Web 出错程序的输出

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

第 223 页的附录 H, 『Web 出错程序 DFHWBEP 的参考信息』具有通信区域中传递到 DFHWBEP 的所有参数的列表和技术描述。

DFHWBEP 的输出参数是:

- 要返回到 Web 客户机的 HTTP 响应的地址。它是一个完整的 HTTP 响应, 包括状态行、HTTP 头和消息体。
- 要返回到 Web 客户机的 HTTP 响应的长度。响应的最长长度为 32K。指定整个缓冲区的长度; CICS 计算消息体的长度并提供合适的 Content-Length 头。

如果您更改消息体，则不需要计算主体长度或更改 Content-Length 头。在返回给 CICS 时，指定包含完整消息的缓冲区的长度。CICS 检查您已提供的消息体的长度，并相应地改正 Content-Length 头。

缺省 HTTP 响应传递到 EBCDIC 代码页 037 中的 DFHWBEP。CICS 假定从 Web 出错程序退出时由 WBEP_RESPONSE_PTR 寻址的 HTTP 响应格式为 EBCDIC，并对响应执行代码页转换以在它返回给客户机前将它转换为合适的 ASCII 字符集。如果分析器程序包含在处理路径中且已为代码页转换设置参数（作为独立服务器和客户机代码页参数，或作为 DFHCNV 键），则 CICS 将这些选项用于代码页转换。如果未包括分析器程序，或发生错误之前未调用分析器，则响应使用 ISO-8859-1 字符集。

缺省状态码和错误响应

由分析器或转换器程序设置的响应代码和原因码映射到缺省状态码和关联响应。如果在使用 URIMAP 定义产生静态响应时发生错误，CICS 也选择缺省状态码和相关响应。状态码和响应可由用户可替代 Web 出错程序 DFHWBEP 进行修改，此 Web 出错程序在通信区域中接收此信息。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

HTTP 协议规范定义了状态码，当请求无法成功完成时，服务器可以在 HTTP 响应中返回它们。第 191 页的附录 C，『CICS Web support 的 HTTP 状态码参考』提供有关这些状态码的信息。

要获取有关 HTTP 响应结构的更多信息，请参阅第 10 页的『HTTP 响应』。

如果分析器或转换器程序处理期间发生错误，则响应和原因码，以及其他信息传递给通信区域中的 Web 出错程序，以帮助确定相应的错误响应。对于分析器或转换器程序处理，或者使用 URIMAP 定义的静态响应产生，完整的缺省错误响应（包括状态码）传递给 Web 出错程序，并可由它进行修改。伴随着错误响应的是一条 CICS 消息和异常跟踪项。

分析器程序使用的响应代码的缺省状态码如下所示：

表 3. 分析器程序处理错误的缺省状态码

wbra_response	缺省状态码
除 URP_OK 以外的任何值	400 错误请求

转换器程序的缺省状态码如下所示：

表 4. 转换器的解码函数的缺省状态码

decode_response	解码原因	缺省状态码
URP_EXCEPTION	URP_CORRUPT_CLIENT_DATA	400 错误请求
URP_EXCEPTION	URP_SECURITY_FAILURE	403 禁止
URP_EXCEPTION	任何其他值	501 未实现
URP_INVALID	任何值	501 未实现
URP_DISASTER	任何值	501 未实现
任何其他值	任何值	500 内部服务器错误

表 5. 转换器的编码函数的缺省状态码

encode_response	编码原因	缺省状态码
任何值, 除了 URP_OK URP_OK_LOOP	任何	501 未实现

使用 URIMAP 定义产生静态响应时所发生错误的缺省状态码如下所示:

表 6. 静态响应处理错误 (使用 URIMAP 定义) 的缺省状态码

资源选项	错误	缺省状态码
TEMPLATENAME (CICS 文档模板)	无法找到指定的文档模板, 或模块引用的 CICS 资源	404 未找到
TEMPLATENAME (CICS 文档模板)	文档引用的 CICS 资源已存在, 但是被禁用	503 服务不可用
HFSFILE (HFS 中的文件)	无法找到指定的 HFS 文件	404 未找到
任何	异常错误	500 内部服务器错误

第 10 章 分析器程序

分析器程序与 TCPIP SERVICE 定义关联。它们的主要作用是在以下情况下解释 HTTP 请求：如果 URIMAP 定义指定使用分析器程序，或者如果不存在 URIMAP 定义。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

当 CICS 作为 HTTP 客户机，或者用于 Web 服务调用时，不能调用分析器程序；仅当 CICS 作为 HTTP 服务器时才调用它们。作为 HTTP 服务器的 CICS 的 CICS Web support 过程中的分析器程序角色在第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』中描述。第 45 页的第 5 章，『规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构』具有帮助您规划作为 HTTP 服务器的 CICS 的体系结构的信息。

分析器程序和 URIMAP 定义之间的关系

CICS Transaction Server for z/OS V3R1 前，作为 HTTP 服务器的 CICS 的所有 HTTP 请求由分析器程序解释。在 CICS TS V3 中，URIMAP 定义是用于控制 HTTP 请求处理的关键性工具。它们取代了分析器程序的主要功能，将请求的 URL 与处理请求的应用程序匹配，以及指定使用转换器程序和别名事务。

然而，URIMAP 定义可以为选择的 HTTP 请求调用分析器程序，以接管某些处理阶段并执行其他操作（例如，监控或审计操作）。命名为 CONVERTER（转换器程序名）、TRANSACTION（别名事务）、USERID（别名事务的用户标识）和 PROGRAM（处理请求的应用程序名）的 URIMAP 定义的属性（这些属性重新产生分析器函数）可以传递到分析器程序中，且分析器程序可以选择覆盖它们。

按 CICS Transaction Server for z/OS V3R1 前 CICS Web support 使用的同一过程操作，您可以选择不使用 URIMAP 定义就直接将 HTTP 请求传递到分析器程序。然而，没有 URIMAP 定义的情况下，如果要更改 CICS 对特殊 HTTP 请求响应的方式，则需要更改分析器程序中的逻辑。使用 URIMAP 定义，您可以将这些更改作为系统管理任务动态执行。也请注意，如果继续使用分析器程序而不是 URIMAP 定义来处理请求，且如果需要在这方面遵从 HTTP/1.1，则根据 HTTP/1.1 规范（RFC 2616）中声明的规则，必须编码分析器程序以执行 URL 比较。

注： 已发现请求的匹配 URIMAP 定义后，所提供的 CICS 提供的样本分析器程序 DFHWBADX 和 CICS 提供的缺省分析器程序 DFHWBAAX 不执行该请求的任何分析，即使 URIMAP 指定 ANALYZER(YES) 时也是如此。

使用分析器程序进行错误处理

虽然现在并不是每个 HTTP 请求的处理路径中都需要分析器程序，但是仍然必须为用于 CICS Web support 的每个 TCPIP SERVICE 资源定义指定一个分析器程序。

分析器程序的名称在资源定义的 URM 属性中指定。您可以在每个 TCPIP SERVICE 定义中指定不同的分析器，或在多个 TCPIP SERVICE 定义中指定相同的分析器。如果正在从 URIMAP 定义调用分析器程序，则无法在不同的分析器程序间进行选择；您只能选择是否使用为 TCPIP SERVICE 定义指定的分析器程序。

如果 CICS 未找到匹配的 URIMAP 定义用于此请求，则调用为 TCPIP SERVICE 定义指定的分析器程序，以处理 HTTP 请求。这可能由于用户输入请求 URL 时出错而引起，或者是因为未安装适合的 URIMAP 定义。（如果 URIMAP 定义存在，但是已禁用，则请求由 Web 出错程序而不是分析器程序处理。）

因为这个原因，最低限度，为每个 TCPIP SERVICE 定义指定的分析器程序应该包含一个过程，以处理它不识别的任何 HTTP 请求，并提供合适的错误响应。您还可以识别本应该由 URIMAP 定义处理的特定请求，并提供更相关的错误响应。错误情况下分析器程序的输出传递给 Web 出错程序，您可使用它修改 HTTP 响应。第 85 页的第 9 章，『Web 出错程序』说明如何制定这些内容。

当 TCPIP SERVICE 定义指定 PROTOCOL(HTTP) 时，CICS 提供的缺省分析器程序 DFHWBAAX 是缺省值。当端口上所有请求应该由 URIMAP 定义处理时，DFHWBAAX 提供基本错误处理。它不提供对使用 URL 格式的请求的支持，此格式是在 CICS TS 3.1 之前，CICS Web support 使用的格式。如果您需要在分析器程序中为 URIMAP 定义不处理的请求提供处理，TCPIP SERVICE 定义上指定的分析器程序应该是 CICS 提供的样本分析器程序 DFHWBADX 或自己定制的分析器程序。

将分析器程序用于某些不支持 Web 的应用程序和非 HTTP 消息

从 URIMAP 定义直接调用不支持 Web 的应用程序时，这些应用程序可能正确运行。然而，因为这些应用程序不使用 EXEC CICS WEB API 命令，所以某些应用程序可能取决于只能由分析器程序为它们提供的工具。在以下情况下，在 HTTP 请求的处理路径中使用分析器程序是基本的：

- 您正在使用不支持 Web 的应用程序和转换器程序产生响应，而它需要标记为 CICS TS V3 前的兼容性处理，这是因为 Web 客户机要求响应与它可能已接收的 CICS TS V3 前的响应一样。（例如，用户编写的客户机在使用新错误响应或其他 HTTP 头时可能遇到问题。）
- 您正在使用不支持 Web 的应用程序和转换器程序产生响应，而传递到通信区域中转换器程序的请求需要非标准代码页转换。如果分析器程序不与不支持 Web 的应用程序协同使用，且请求头表明消息体是文本，则 CICS 使用以下标准设置转换通信区域中提供的消息体：
 - 对于字符集，如果 Web 客户机的请求具有命名 CICS 支持的字符集的 Content-Type 头，则使用该字符集。如果 Web 客户机的请求没有 Content-Type 头，或命名的字符集不受支持，则使用 ISO-8859-1 字符集。
 - 对于主机代码页，CICS 将缺省代码页用于本地 CICS 区域，如 LOCALCCSID 系统初始化参数中所指定。

通信区域中产生的响应的代码页转换使用同一设置。如果这些标准设置不合适，或者如果不需要代码页转换，则分析器程序必须用于指定备用设置。

在这些情况下，可以为请求设置 URIMAP 定义，但是它必须指定分析器程序。

对于使用 TCPIP SERVICE 定义上用户定义的（USER）协议的非 HTTP 请求，总是需要分析器程序以处理请求，且不能使用 URIMAP 定义。第 127 页的第 14 章，『CICS Web support 和非 HTTP 请求』说明如何处理非 HTTP 请求。

将分析器程序用于其他处理

在可以选择是否在处理路径中使用分析器程序的情况下，您可能为了以下原因选择使用分析器程序：

- 根据请求内容，您可能要对处理路径的元素进行动态更改。HTTP 请求的每个 URL 由单个 URIMAP 定义匹配，它定义了单个处理路径。分析器程序可以解释请求内容并更改某些元素，例如，处理请求的应用程序、转换器程序的涉及，或用于请求的别名事务和用户标识。
- 您要将监控或审计操作引入进程。分析器程序是执行此操作的理想位置。
- 您正在从 CICS TS V2 迁移现有 CICS Web support 体系结构，且您的现有分析器程序提供您要在请求处理期间维护的其他功能（例如，将信息传递到转换器程序）。

『编写分析器程序』说明了分析器程序可以执行的完整范围函数。

编写分析器程序

您可以用汇编程序、C、COBOL 或 PL/I 编写分析器程序。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

分析器程序的输入和输出参数在通信区域中传递。第 203 页的附录 E，『分析器程序的参考信息』中描述了语言相关的头文件、包含文件和映射通信区域的副本。

分析器程序可以执行的完整范围的功能如下：

- 确定是否应该为请求继续处理，或者 CICS 是否应该将错误响应返回到 Web 客户机。
- 分析请求内容和已从 URIMAP 定义传递到转换器程序的任何参数，以确定需要哪些后续处理阶段，以及需要哪些 CICS 资源以执行每个阶段。（此分析期间可以使用 EXEC CICS WEB API 命令。）
- 在请求传递到应用程序之前，指定要处理请求的转换器程序名。转换器程序通常与不支持 Web 的应用程序协同使用。需要的话，提供用户令牌以使分析器程序与转换器程序通信。请求传递到一块 32K 存储器中的转换器程序，它由通信区域中的指针表明。第 101 页的第 11 章，『转换器程序』说明转换器程序的功能。
- 指定要处理请求并提供响应的用户编写的应用程序名。
- 指定处理剩余处理阶段的别名事务的事务标识。
- 指定要与别名事务关联的用户标识。
- 指定或禁止传递到存储器块中转换器程序的请求和转换器程序发送的响应的代码页转换。这不影响使用 EXEC CICS WEB API 命令的支持 Web 的应用程序；它们直接从 CICS 请求代码页转换。第 32 页的『CICS Web support 的代码页转换』说明代码页转换过程。
- 指定为迁移目的而提供的标志，该标志表明不支持 Web 的应用程序需要在哪里进行 CICS TS V3 前的兼容性处理。
- 修改请求主体。所做的任何更改在传递到存储器块中转换器程序的数据中都是可视的，但是在 EXEC CICS WEB API 命令的数据中不可视。

CICS 提供缺省分析器程序 DFHWBAAX（它在第 98 页的『CICS 提供的缺省分析器程序 DFHWBAAX』中描述）和样本分析器程序 DFHWBADX（它在第 99 页的『CICS 提

供的样本分析器程序 DFHWBADX 中描述)。如果这些分析器不符合您的需求,则需要编写您自己的分析器。您可能可以将 DFHWBADX 作为示例使用。

所有用户可替代程序对于操作 CICS Web support 的系统来说都必须是本地的。如果您不使用程序的自动安装,则必须为 CICS Web support 使用的所有用户可替代程序(包括分析器和转换器程序)定义和安装程序定义。如果使用程序的自动安装,则必须确保用正确的属性安装了用户可替代程序。请注意,必须使用 EXECKEY (CICS) 来定义分析器程序。

要获取有关编写用户可替代程序的更多信息,请参阅 *CICS Customization Guide*。

分析器程序的输入

输入参数传递到通信区域中的分析器程序,提供有关请求特性和内容的信息,以及 URIMAP 定义提供的任何输入。分析器程序可选择接受这些值并将它们作为输出参数传递,或它可根据它对请求内容的分析来动态地重设它们。

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

第 204 页的『分析器程序的参数』具有通信区域中所有参数的列表和技术描述。

输入参数包含以下各项,或指向它们的指针:

- 分析器参数列表的识别标识。
- 客户机和服务器(作为 HTTP 服务器的 CICS)的点分十进制 IP 地址。
- 请求是否为 HTTP 请求的指示符。
- 是否找到请求的匹配 URIMAP 定义的指示符。如果此指示符为正,则 URIMAP 定义可能已将其他输入参数传递到分析器程序。
- HTTP 版本。
- 请求方法。
- 为请求指定的主机名,从 Host 头或(对于绝对 URI)从请求 URL 获取。对于 HTTP/1.1 请求,必需主机名,因此此参数总是传递到分析器。对于 HTTP/1.0 请求,可能不提供主机名。
- URL 的路径部分。
- 为请求指定的任何查询字符串。
- 请求的 HTTP 头。

注: 如果已使用分块的传输编码发送请求,则尾部头不会与主请求头一起被传递到分析器程序。

- 请求主体,或所需数量的请求主体将适合一块 32K 的存储器。(这是指向包含请求的独立存储器块的指针。)

对于在使用 SSL 客户机认证的连接上接收到的 HTTP 请求,还传递以下参数:

- 从客户机证书获取的用户标识。

如果已为请求找到匹配 URIMAP 定义并且已调用分析器程序,则在以下参数已存在于 URIMAP 定义的情况下,将它们从 URIMAP 定义传递到分析器程序:

- 请求被传递到应用程序之前处理它的建议转换器程序的名称 (URIMAP 定义中的 CONVERTER 属性)。

- 处理请求并提供响应的用户编写的建议应用程序的名称（URIMAP 定义中的 PROGRAM 属性）。
- 覆盖剩余处理阶段的建议别名事务的事务标识（URIMAP 定义中的 TRANSACTION 属性）。
- 要与别名事务关联的建议用户标识（URIMAP 定义中的 USERID 属性）。如果客户机提供了用户标识，则可以覆盖它。

wbra_urimap 输入参数可用于测试是否在请求的处理路径中使用 URIMAP 定义。

注：如果您正在使用分析器程序（而不是 URIMAP 定义）来处理请求，并且您需要在在这方面遵循 HTTP/1.1，则必须按照 HTTP/1.1 规范中声明的规则编码您的分析器程序以执行 URL 比较。按照这些规则，比较规则名和主机名不区分大小写，但比较路径区分大小写。比较之前 URL 的所有组成部分都未转义。当 CICS 将 URL 与 URIMAP 定义比较时，它按照这些规则执行。

如果愿意，您还可以使用 EXEC CICS WEB API 命令检查 HTTP 请求。使用 EXEC CICS WEB 命令可以增加您的请求分析的准确性和完整性，特别当检查内容和用法容易发生较大变化的 HTTP 头时。EXEC CICS WEB 命令还简化了从请求查找和抽取查询字符串或格式字段信息的过程，这可能是后续处理的关键决定因素。

您可以使用 EXTRACT TCPIP 命令获取有关正在处理的客户机请求的以下信息：

- Web 客户机的 IP 地址
- 如 DNS 服务器所知的 Web 客户机的主机名
- Web 客户机发送它的连接请求的端口号
- 服务器的（即，作为 HTTP 服务器的 CICS 的）IP 地址
- 使用的认证类型
- 使用中的 SSL 支持级别
- 与请求关联的 TCPIP SERVICE 定义

分析器程序的输出

分析器程序在通信区域中提供输出。该输出包括响应代码，以及可用于指定更多处理阶段并用于与转换器程序共享信息的可选输出参数范围。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

第 204 页的『分析器程序的参数』具有通信区域中所有参数的列表和技术描述。

分析器程序必须在它的通信区域中提供以下输出：

- 响应码。
 - 如果分析器程序返回响应码 URP_OK，处理会继续下一步。
 - 如果分析器程序返回任何其他值，则 CICS 将错误响应返回到 Web 客户机。该响应可以用用户可替代的 Web 出错程序来修改。第 89 页的『缺省状态码和错误响应』告诉您来自分析器的返回码如何映射到 CICS 返回到 Web 客户机的状态码。

分析器程序也可能提供以下输出：

- 在请求传递到用户编写的应用程序之前，用于处理它的转换器程序的名称。
 - 如果已从 URIMAP 定义输入转换器程序名，则您可接受或重设它。

- 如果分析器表明不需要转换器程序，则请求的第一个 32K 字节传递到一块存储器中用户编写的应用程序。支持 Web 的应用程序可以忽略此内容并使用 EXEC CICS WEB API 命令读取请求。
- 处理请求并提供响应的应用程序的名称。
 - 如果已从 URIMAP 定义输入程序名，则您可接受或重设它。
 - 如果您要使用转换器程序，则转换器程序可指定或重设程序名。可以用这种方式使用转换器以在处理请求时涉及到多个程序。
- 涵盖剩余处理阶段的别名事务的事务标识。如果已从 URIMAP 定义输入事务标识，则您可接受或重设它。
- 与别名事务关联的用户标识。如果已从 URIMAP 定义输入用户标识，则您可接受或重设它。如果您未指定用户标识，则 CICS 将这样确定：
 - 如果已从 URIMAP 定义输入用户标识，则使用它。
 - 如果 HTTP 请求在客户机认证下使用 SSL，就从客户机证书中获得用户标识。
 - 在其他情况下，使用 CICS 缺省用户标识。
- 如果它在通信区域中产生，则是与包含请求的 32K 存储器块的代码页转换相关的参数，以及与响应主体的代码页转换相关的参数。

注：使用 EXEC CICS WEB API 命令的支持 Web 的应用程序不使用通信区域机制接收和发送响应，且它们直接从 CICS 请求代码页转换。

您可以用两种方式之一为包含请求的一块存储器转换指定参数：

- 作为一对参数，它指定 Web 客户机使用的字符集 (wbra_characteraset) 和适用于应用程序的主机代码页 (wbra_hostcodepage)。用这种方式指定参数意味着代码页转换表 (DFHCNV) 中的条目不是必需的。
- 作为 DFHCNV 代码页转换表中条目的键 (wbra_dfhcnv_key)。除了迁移目的，不建议使用它。

如果未指定任何这些参数，则 CICS 的缺省行为是使用第 91 页的第 10 章，『分析器程序』中描述的标准设置转换文本消息。如果您要为通信区域中的请求和响应执行代码页转换，则将 wbra_dfhcnv_key 设置为空或空的。

- 此标志表明，使用转换器程序的非 Web support 的应用程序需要 CICS TS V3 兼容性预处理 (wbra_commarea)。此标志是为了迁移目的提供的，以在特殊情况下只由不使用 EXEC CICS WEB API 命令的应用程序使用：Web 客户机需要与它从 CICS TS V3 之前版本接收到的响应相同的响应。设置此标志意味着：
 - CICS 不添加通常为 HTTP/1.1 消息插入的任何响应头。仅使用 CICS Transaction Server for z/OS V3R1 前发送到客户机的头。
 - 如果需要错误处理，CICS 发送适合于，并标记为 HTTP/1.0 响应的错误响应，无论 Web 客户机的 HTTP 版本是什么。CICS 将通常使用 HTTP/1.1 错误响应来应答 HTTP/1.1 客户机，但是这可能会使客户机误认为该应用程序通常将发送 HTTP/1.1 响应。
- 一个 8 字节用户令牌，用于在分析器程序和转换器程序间共享信息。第 97 页的『共享分析器和转换器程序之间的数据』说明了这种情况。
- 已修改的请求主体长度的值

分析器可以修改请求的内容：

- 修改过的数据长度可以短于原始数据，或者和原始数据相同。不能延长请求主体。

- 所做的任何更改在传递到转换器程序的数据中可视，但是对于 EXEC CICS WEB API 命令不可视。

共享分析器和转换器程序之间的数据

CICS 在分析器程序和转换器程序之间传递三个参数，使数据能被这些处理阶段共享：

user_data 指针

该参数包含阶段间传递的 32K 存储块的地址。在分析器的入口处，指向包含 HTTP 请求的存储块的指针。当转换器编码函数完成时，CICS Web support 使用它查找包含 HTTP 响应的存储块。

尽管您可以修改该指针所指存储器的内容，但禁止更改分析器中的指针值。

在转换器和 CICS 应用程序之间，您可以将指针不更改地从一个阶段传递到另一个阶段，或者您可以在程序中发出 GETMAIN 命令并传递新获取的存储器地址（以指针的形式）。

user_data 长度

该参数是 user_data 指针定位的存储器的长度。

用户令牌

用户令牌是一个由分析器和转换器共享的 8 字节字段。它可以包含任何您想要的信息：

- 您可以在用户令牌里直接传递少量的共享信息。
- 要传递更多的信息，您可以在某个程序中发出 GETMAIN 命令来获取共享工作区域的存储器。然后使用用户令牌传递共享存储器的地址。

您可以在每个程序中更改用户令牌的内容：例如，用户令牌在从分析器传递到转换器解码阶段可能有某种含义，而传递到编码阶段可能会有不同含义。

注：分析器和转换器程序在不同的 CICS 任务下执行。因此，如果您在分析器程序中发出 GETMAIN 命令，而存储器要在转换器程序中可见的话，您必须对 SHARED 选项进行编码。通常，CICS 不会自动释放大 SHARED 选项获取的存储器，因此当您的程序不再需要该存储器时，必须发出 FREEMAIN 命令。然而，在 HTTP 响应已发送到 Web 客户机后，CICS 将释放 user_data 指针寻址的存储器。

从分析器程序选择转义数据或未转义数据

传递给分析器程序以进行解析的 HTTP 请求使用它的转义形式。URL 中或消息体中表单数据中的保留字符或排除字符作为 %xx 序列出现，其中 xx 是保留字符的 ASCII 十六进制表示法。分析器可以用以下两种形式将一块 32K 存储器中的请求传递到后续处理阶段：转义序列仍然出现的转义形式，或转义序列转换回原始字符的未转义形式。使用 EXEC CICS WEB API 命令的支持 Web 的应用程序不使用此机制接收响应，且它们直接从 CICS 请求未转义。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

第 12 页的『转义和未转义数据』说明转义及其目的。转义和未转义仅应用到 HTTP 请求的以下元素：

- 请求行的 URL 部分，包括任何查询字符串。查询字符串可能是来自具有 GET 方法的表单的数据。

- 从具有 `POST` 方法的表单返回的表单数据以及缺省编码 `application/x-www-form-urlencoded`。此数据出现在消息体中。第 13 页的『HTML 表单』说明关于表单数据的更多内容。

如果一块 32K 存储器中的请求以未转义形式传递，则分析器可以将数据从转义形式转换成未转义形式，或由 CICS 执行该转换。

- 要以转义形式传递请求，请将分析器中的 `WBRA_UNESCAPE` 设置为 `WBRA_UNESCAPE_NOT_REQUIRED`。`WBRA_UNESCAPE_NOT_REQUIRED` 是缺省值。
- 要以未转义形式传递请求并由 CICS 执行转换，请将分析器中的 `WBRA_UNESCAPE` 设置为 `WBRA_UNESCAPE_REQUIRED`。
- 要在分析器执行转换以后，以未转义形式传递请求，请将 `WBRA_UNESCAPE` 设置为 `WBRA_UNESCAPE_NOT_REQUIRED`。

使用 `EXEC CICS WEB API` 命令的支持 Web 的应用程序不使用通信区域机制接收和发送响应，且它们直接从 CICS 请求未转义。对于使用 `EXEC CICS WEB API` 命令的支持 Web 的应用程序，当从使用 `WEB READ FORMFIELD` 命令或表单字段浏览命令从请求抽取表单数据时，CICS 执行未转义，并且数据以它的未转义形式返回。当使用 `WEB EXTRACT` 命令从请求抽取查询字符串时，数据以它的转义形式返回。

如果您正用可以通过 CICS Web support 或通过 CICS 业务逻辑接口运行的通信区域接口编写应用程序，则确保 `WBRA_UNESCAPE` 设置为 `WBRA_UNESCAPE_NOT_REQUIRED`，且将任何未转义委托给应用程序。如果不执行此操作，则应用程序通过 CICS 业务逻辑接口传递未转义数据，而通过 CICS Web support 转义数据，这可能会导致不可预测的结果。

CICS 提供的缺省分析器程序 DFHWBAAX

CICS 提供缺省分析器程序 DFHWBAAX。DFHWBAAX 为用于 CICS Web support 的 TCPIP SERVICE 资源定义提供错误处理函数。当使用一个端口的所有请求都是使用 URIMAP 定义处理时，它适合使用。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

CICS 仅为组装器中的 DFHWBAAX 提供源代码。

DFHWBAAX 是指定 PROTOCOL(HTTP) 的 TCPIP SERVICE 定义的缺省分析器程序。

在通信区域中，DFHWBAAX 将相同的输入和输出参数作为标准分析器程序接收。如前所述，它不使用大多数这些参数，并且它不对使用 URL 格式的请求提供支持，该格式是在 CICS TS 3.1 之前的 CICS Web support 使用的格式。它而是采用如下的简单操作：

- 已发现请求的匹配 URIMAP 定义后，DFHWBAAX 不执行进一步处理，即使 URIMAP 指定 `ANALYZER(YES)` 时也是如此。它使用 `wbra_urimap` 输入参数测试 URIMAP 定义是否存在，以及结果是否为正，不对请求 URL 执行任何分析就返回。这意味着为别名事务在 URIMAP 定义中指定设置，自动接受转换器程序和应用程序，并将它们用于确定后续处理阶段。
- 如果未找到匹配的 URIMAP 定义，则 DFHWBAAX 对用户可替代 Web 出错事务程序 DFHWBERX 提供控制以产生错误响应。使用 `wbra_server_program` 输出参数，

通过将 DFHWBERX 设置为应用程序以处理请求可以实现此目的。DFHWBAAX 不对通信区域进行任何其他更改。根据 Web 客户机发出的请求，接收控制时，DFHWBERX 提供 404（未找到）状态码的 HTTP 响应，或 SOAP 故障请求。

DFHWBAAX 使用标准范围的响应：**URP_OK**、**URP_EXCEPTION** 和 **URP_INVALID**。如前所述，没有为 DFHWBAAX 构建的原因值。请注意，如果响应不同于 **URP_OK**，则它表明处理中的错误，且控制传递到用户可替代的 Web 出错程序 DFHWBEP，而不是 Web 出错应用程序 DFHWBERX。

CICS 提供的样本分析器程序 DFHWBADX

CICS 提供有效的样本分析器程序 DFHWBADX。如果您需要通过分析器程序提供请求处理程序，又或者替代通过 URIMAP 定义，您可以将 DFHWBADX 用作编写您自己的分析器程序的起点。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

CICS 提供多种语言的源代码：

- DFHWBADX（汇编语言）
- DFHWBAHX（C）
- DFHWBALX（PL/I）
- DFHWBAOX（COBOL）

已发现匹配请求的 URIMAP 定义后，即使 URIMAP 指定 ANALYZER(YES)，所提供的 DFHWBADX 也不执行请求的任何分析。这意味着为别名事务在 URIMAP 定义中指定设置，自动接受转换器程序 and 应用程序，并将它们用于确定后续处理阶段。

DFHWBADX 使用 `wbra_urimap` 输入参数测试 URIMAP 定义是否存在，如果结果是存在，则不对请求 URL 执行任何分析就返回。如果编写您自己的分析器程序并希望它与 URIMAP 定义交互，则不要复制 DFHWBADX 处理的这一方面。为了以其他方式修改分析器程序的处理，您可能要测试 `wbra_urimap` 输入参数。例如，您可以测试该参数以决定是根据来自 URIMAP 定义的输入参数执行分析，还是直接对请求 URL 执行分析。

DFHWBADX 如何解释请求 URL

DFHWBADX 解释 HTTP 请求，这些请求中的 URL 路径部分有以下语法：

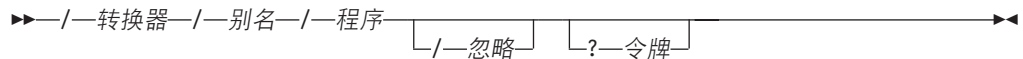


图 7. DFHWBADX 解释的路径部分语法

分析器程序处理的所有字段都转换为大写。转换之后：

converter

指定用于请求的转换器程序的名称。它最多可以有 8 个字符。

作为一个特殊情况，四位字符值 "CICS" 表示不使用转换器程序。

alias

指定用于后续请求处理的别名事务的事务标识。它最多可以有 4 个字符。

program

指定服务于请求的 CICS 应用程序的名称。它最多可以有 8 个字符。

ignored

DFHWBADX 忽略这部分路径（但是转换器程序或应用程序可能使用它）。

token

前 8 个字节指定要传递给转换器程序的用户令牌。DFHWBADX 忽略令牌的前八个字节后的数据（但转换器程序或应用程序可能使用它）。

在示例路径 `/cics/cwba/dfh$wb1a` 中：

- 未使用转换器程序。
- 别名事务是 CWBA。
- CICS 应用程序是 DFH\$WB1A。

除了原始 HTTP 请求派生的输出之外，DFHWBADX 还设置下列输出：

- 代码页转换模板为 DFHWBUD。这个模板在样本转换表 DFHCNVW\$ 中定义，它在 ASCII Latin-1 字符集（代码页 ISO 8859-1）和 EBCDIC Latin 字符集（代码页 037）之间转换数据。无需进行任何配置就可以使用样本转换表，但是请注意，可以使用输出参数 `wbra_characteraset` 和 `wbra_hostcodepage` 来代替 `wbra_dfhcnv_key` 输出参数以提供更好的控制，并避免使用转换表。
- DFHWBADX 以转义形式传递请求，并设置 `WBRA_UNESCAPE_NOT_REQUIRED`。

来自 DFHWBADX 的响应

DFHWBADX 产生的响应含义如下：

URP_OK

分析器发现请求符合缺省 HTTP 请求格式，并生成适合该别名的输出。

URP_EXCEPTION

分析器发现请求不符合缺省格式。提供的原因码如下：

- 1 资源长度小于 6。（使用 DFHWBADX 识别的 URL 格式，可能的最短资源的规范是 `/A/B/C`，要求程序 C 在事务 B 下与转换器 A 一同运行。）响应和原因是当进入请求不是 HTTP 请求时的响应和原因。
- 2 资源规范不以 `『/』` 开始。
- 3 资源规范包含一个 `『/』`，但少于 3 个。
- 4 资源规范中转换器名称的长度为 0 或大于 8。
- 5 资源规范中事务名称的长度为 0 或大于 4。
- 6 资源规范中 CICS 应用程序名称的长度为 0 或大于 8。

响应和原因代码在消息 DFHWB0723 中显示。错误响应 400（错误请求）状态码返回到 Web 客户机。这可以用用户可替代的 Web 出错程序 DFHWBEP 来修改。

URP_INVALID

`eye-catcher` 是无效的。这表示一个内部错误。

第 11 章 转换器程序

转换器程序主要与应用程序协同使用，原先没有为了与 Web 协同使用而编码这些应用程序。它们还可用于将多个应用程序的输出组合到单个 HTTP 消息中。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

当 CICS 是 HTTP 客户机时，或对于 Web service 处理，不使用转换器程序；仅当 CICS 是 HTTP 服务器时才可以调用它们。作为 HTTP 服务器的 CICS 的 CICS Web support 过程中的转换器程序角色在第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』中描述。第 45 页的第 5 章，『规划作为 HTTP 服务器的 CICS 的 CICS Web support 体系结构』具有帮助您规划作为 HTTP 服务器的 CICS 的体系结构的信息。

URIMAP 定义可以调用转换器程序以执行 HTTP 请求的相关处理。如果 CICS Web support 处理中使用了分析器程序，则分析器程序也可以调用转换器程序。转换器程序在以下情况下可以有用：

- 最初不是编码与 Web 协同使用的应用程序需要接收通信区域格式的输入，或者需要将它们的输出转换为 HTTP 响应。使用 EXEC CICS WEB 和 EXEC CICS DOCUMENT 应用程序编程接口编码的支持 Web 的应用程序不需要进行此转换。您可以使用转换器程序以执行此转换或请求内容上的其他处理。
- 当您要使多个应用程序对序列中的同一请求数据起作用，并将单个 HTTP 响应返回到 Web 客户机时。

如果从 URIMAP 定义直接调用转换器程序，则 URIMAP 定义的 PROGRAM 属性（它指定要处理请求的应用程序名）可以传递到转换器程序，且转换器程序可以选择覆盖它。

转换器程序不可以为它从分析器程序接收的请求指定代码页转换设置（其中通信区域中的指针表明包含此请求的 32K 存储块）。如果直接从 URIMAP 定义调用转换器程序，并且 Web 客户机请求的头表明该消息体是文本，则 CICS 使用以下标准设置转换存储块中提供的消息体：

- 对于字符集，如果 Web 客户机的请求具有对 CICS 支持的字符集命名的 Content-Type 头，则使用该字符集。如果 Web 客户机的请求没有 Content-Type 头，或命名的字符集不受支持，则使用 ISO-8859-1 字符集。
- 对于主机代码页，CICS 将缺省代码页用于本地 CICS 区域，如 LOCALCCSID 系统初始化参数中所指定。

如果这些标准设置不合适，或者如果不需要代码页转换，则必须在处理路径中使用分析器程序以指定备用设置。

对于发送到 Web 客户机的响应，CICS 镜像为请求执行的代码页转换。使用分析器程序的字符集和主机代码页设置，或使用缺少分析器程序时的缺省设置。如果分析器程序禁止了请求的代码页转换，则不为响应主体执行代码页转换。

转换器程序不会与分析器程序以同样的方式与 TCPIPService 定义关联。您可以使用 CICS 系统的任何本地转换器程序来处理任何 HTTP 请求。对于给定的请求，为解码和编码函数调用同一个转换器程序。

转换器程序有两个函数:

解码 HTTP 请求传递给 32K 存储块中转换器程序的解码函数, 这由通信区域中的 `user_data` 指针表明。它已被分为独立的元素, 例如, 方法、请求头和请求主体。(请注意, 如果请求太长而不适合存储块, 则其余数据将不传递到转换器程序。) 还提供应该处理请求的应用程序名, 从请求的 URIMAP 定义获取或由分析器程序设置。如果使用了分析器程序, 则它可以提供用户令牌中的其他信息。使用此信息, 解码函数将:

- 确定是否应该为请求继续处理, 或者 CICS 是否应该将错误响应返回到 Web 客户机。
- 指定要处理请求并提供响应的用户编写的应用程序名。如果名称已从 URIMAP 定义或由分析器程序输入, 则转换器程序可以接受或更改此内容。
- 构造传递到用户编写的应用程序的通信区域。通信区域包括来自 Web 客户机的请求的数据, 该请求已转换为应用程序可接受的输入格式。可以复用包含 HTTP 请求的存储器, 或者可以指定新的通信区域。

编码 使用由用户编写的应用程序返回的通信区域, 编码函数将:

- 如果多个应用程序要提供数据, 则设置循环响应, 以再次调用解码函数。解码函数更改应用程序的名称, 并在通信区域中提供合适的输入。输出将再次返回到编码函数。
- 在通信区域中构造 HTTP 响应, 该响应将被发送到 Web 客户机。

转换器程序由 CICS Web support 和 CICS 业务逻辑接口使用。CICS 业务逻辑接口中的转换器程序角色在第 172 页的『使用 CICS 业务逻辑接口调用程序』中描述。CICS 业务逻辑接口的调用者确定是否需要转换器, 以及应该调用哪个转换器程序。对于给定的请求, 为解码和编码函数调用同一个转换器程序。

要获取有关编写用户可替代程序的更多信息, 请参阅 *CICS Customization Guide*。

所有用户可替代程序对于操作 CICS Web support 的系统来说都必须是本地的。如果您不使用程序的自动安装, 则必须为 CICS Web support 使用的所有用户可替代程序(包括分析器和转换器程序)定义和安装程序定义。如果使用程序的自动安装, 必须确保用正确的属性安装了用户可替代程序。

要获取有关转换器的详细信息, 请参阅第 209 页的附录 F, 『转换器程序的参考信息』。

写转换器

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

您可以用汇编程序、C、COBOL 或 PL/I 来编写分析器。在第 209 页的附录 F, 『转换器程序的参考信息』中, 描述了和语言相关的头文件、包括文件以及副本。

转换器必须在接收请求的 TCPIP SERVICE 所在的同一 CICS 区内运行。

从转换器程序传递给用户应用程序的通信区域的大小和内容是由为 HTTP 请求调用的 CICS 应用程序的设计来支配的。对 CICS Web support 使用转换器程序时, 您可以:

- 发出 GETMAIN 命令来为通信区域获取存储器。
- 为通信区域使用在早期处理阶段(例如分析器程序)中获得的存储器。

- 为通信区域使用包含 HTTP 请求的存储器。

使用具有 CICS 业务逻辑接口的转换器程序时，某些限制可能影响您如何构造通信区域。要获得更多信息，请参阅第 178 页的『偏移方式和指针方式』。

解码函数的输入参数

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

输入参数会传递到通信区域中的解码函数。这些参数包含：

- Web 客户机的 IP 地址
- 到 HTTP 版本的指针
- 到方法的指针
- 到 URL 路径部分的指针
- 到请求头的指针
- 到实体主体的指针
- 为请求服务的 CICS 应用程序的名称（由分析器设置确定）
- 八字节的用户令牌，用于在分析器程序和转换器程序之间共享信息。请参阅第 97 页的『共享分析器和转换器程序之间的数据』。
- 用于记录为每个 HTTP 请求进入的解码函数次数的迭代计数器。计数器在第一次调用解码函数之前设置成 1，然后在以后的每次调用之前加一。
- 实体主体的地址是否可以成为 FREEMAIN 命令的目标的指示。

如果分析器程序更改了任意这些参数的值，则在转换器的通信区域中可看到已更改的值。要查看原始值，请在转换器程序中使用 EXEC CICS WEB 命令。

解码函数的输出参数

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

解码函数必须在其通信区域中提供以下输出：

- 响应码（可选地由原因码限定）。

如果解码函数返回 URP_OK 响应代码，处理将继续到下一步。

如果解码函数返回任何其他值，则拒绝 HTTP 请求。要获得有关 HTTP 服务器响应的详细信息，请参阅第 89 页的『缺省状态码和错误响应』。

- 传递到 CICS 应用程序的通信区域的地址和长度。如果未调用应用程序，则将通信区域不更改地传递给编码函数。

它也可能提供以下输出：

- 用于对请求提供服务的 CICS 应用程序的名称。如果分析器提供了名称，转换器可以更改它，或者指定不应该调用应用程序。
- 八字节的用户令牌，用于在分析器程序和转换器程序之间共享信息。请参阅第 97 页的『共享分析器和转换器程序之间的数据』。

编码函数的输入参数

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

输入参数是传递到通信区域中的编码函数。这些参数包含:

- CICS 应用程序返回的通信区域的地址和长度。如果未调用应用程序, 则从解码函数不更改地传递通信区域。
- 八字节的用户令牌, 用于在分析器程序和转换器程序之间共享信息。请参阅第 97 页的『共享分析器和转换器程序之间的数据』。
- 用于记录为每个 HTTP 请求进入的编码函数次数的迭代计数器。计数器在第一次调用编码函数之前设置成 1, 然后在以后的每次调用之前加一。

编码函数的输出参数

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

编码函数可以提供以下输出:

- 响应代码 (可选地由原因限定):
 - 如果编码函数返回 URP_OK 响应码, 则将 HTTP 响应发送给 Web 客户机。
 - 如果编码函数返回 URP_OK_LOOP 响应码, 处理将继续到解码函数。要获得更多信息, 请参阅第 105 页的『调用多个 CICS 应用程序』。
 - 如果编码函数返回任何其他值, 则拒绝 HTTP 请求。要获得有关 HTTP 服务器响应的详细信息, 请参阅第 89 页的『缺省状态码和错误响应』。
- 如果响应代码是 URP_OK, 则缓冲区的地址包含要发送到 Web 客户机的 HTTP 响应的缓冲区地址, 以及 HTTP 响应的长度。缓冲区的第一个字必须包含数据长度 (也就是说, HTTP 响应长度加 4)。
- 八字节的用户令牌, 用于在分析器程序和转换器程序之间共享信息。请参阅第 97 页的『共享分析器和转换器程序之间的数据』。

转换器程序: 构造 HTTP 响应

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

您在转换器程序中构造的 HTTP 响应的大小和内容由以下内容支配:

- 希望的在响应中包括的 HTTP 头的数量和长度。
- 希望在响应中发送的 HTML 数据的长度和内容。

以上这些内容依次由用户编写的应用程序的设计和它在其通信区域中返回的信息支配。

响应必须包含 HTTP 版本、状态码、状态文本、所需的任何 HTTP 头和消息体。响应的格式应该符合您正在遵从的 HTTP 协议规范 (HTTP/1.0 或 HTTP/1.1)。

CICS 插入适合 HTTP/1.0 或 HTTP/1.1 响应的某些 HTTP 头, 这些头在第 185 页的附录 B, 『CICS Web support 的 HTTP 头参考』中列出。如果通信区域中的消息已经包含这些头, 则 CICS 不替换它们。

对 CICS Web support 使用转换器程序时，您可以：

- 发出 GETMAIN 命令来为 HTTP 响应获取存储器。
- 为通信区域使用在早期处理阶段（例如分析器程序）中获得的存储器。
- 构造应用程序返回的通信区域中的响应。

用于响应的区域的第一个字必须包含区域长度（也就是说，HTTP 响应长度加 4）。

使用具有 CICS 业务逻辑接口的转换器程序时，某些限制可能影响您如何构造响应。要获得更多信息，请参阅第 178 页的『偏移方式和指针方式』。

调用多个 CICS 应用程序

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

有时，构造对 HTTP 响应所需的数据来自于多个应用程序。如果是这种情况，则可以根据需要重复以下序列：

- 转换器的解码函数
- 应用程序
- 转换器的编码函数

通过在编码函数中将响应设置成 URP_OK_LOOP 以完成这一操作。当 HTTP 响应完成时，将响应设置成 URP_OK。

当解码函数在第二个和随后场合被调用时，以下输入参数不可用：

- HTTP 版本
- 方法
- URL 的路径部分
- 请求头
- 实体主体

然而，可以使用 WEB EXTRACT 命令来检索相同的信息。

使用 user_data 指针和用户令牌共享解码和编码函数之间的数据 - 请参阅第 97 页的『共享分析器和转换器程序之间的数据』以获取更多信息。最后一次调用编码函数时，必须确保 user_data 指针寻址到一个有效的 HTTP 响应 - 请参阅第 104 页的『编码函数的输出参数』以获取更多信息。

第 12 章 CICS Web support 的安全性

当 CICS 连接到因特网后，为了阻止对 CICS 应用程序和数据的未授权访问，也为了阻止获取通过因特网发送的私有信息的第三方，安全措施是必需的。

您应该在 CICS Web support 体系结构的整个开发过程中考虑安全性，它应该作为 CICS Web support 应用程序和实用程序设计的一部分，为相关 CICS 工具创建资源定义时也是如此。本部分总结了可用于增强 CICS Web support 实施的安全性的措施。

CICS Web support 的 CICS 系统和资源安全性

当 CICS 是 HTTP 服务器时，必须保护 CICS 系统以不让未授权的用户访问。如果未正确地保护系统，则用户也许能访问机密数据，或阻塞系统致使拒绝对其他用户的服务。

入站端口的安全性

用于 CICS Web support 的每个端口由 TCPIP SERVICE 资源定义来定义。TCPIP SERVICE 定义指定端口的安全性选项，包括是否使用 SSL 以及从客户机请求的认证的级别。第 71 页的『为 CICS Web support 创建 TCPIP SERVICE 资源定义』说明如何创建这些定义。为保持端口的安全，记住这些要点：

- 在每个 TCPIP SERVICE 定义上指定 MAXDATALEN 属性。此选项限制 CICS 为单个请求将要接受的最大数据量，并且它有助于保护 CICS 在进行大量数据传输时免受拒绝服务的攻击。
- 无论在何种情况下，如果您希望确保您与 Web 客户机的交互保持机密并无法被第三方拦截，都可使用安全套接字层（SSL）。在传输机密数据或将权限（如用户标识和密码）传递到服务器的情况下，使用 SSL 就显得特别重要。第 112 页的『具有 CICS Web support 的 SSL』说明如何与 CICS Web support 协同使用 SSL。

如果您在一个或多个 CICS Web support 端口上执行不寻常的操作，则可使用 CICS 系统命令在不同级别（单个请求、虚拟主机、端口或整个 CICS Web support）关闭 CICS Web support，而无需关闭 CICS 系统。第 82 页的『拒绝 HTTP 请求』说明如何执行此操作。

URIMAP 资源定义指定 HTTP 或 HTTPS（带 SSL 的 HTTP）作为请求的规则。指定 HTTP 规则的 URIMAP 接受使用 HTTP 规则或更安全的 HTTPS 规则发出的 Web 客户机请求。指定 HTTPS 规则的 URIMAP 仅接受使用 HTTPS 规则发出的 Web 客户机请求。

当将 HTTPS 作为规则的 URIMAP 定义与 Web 客户机发出的请求匹配时，CICS 检查此请求使用的入站端口是否使用 SSL。如果不为此端口指定 SSL，则拒绝请求，其状态码为 403（已禁止）。当 URIMAP 定义应用到所有入站端口时，该检查会确保 Web 客户机无法使用不安全的端口来访问安全的资源。因为不会对指定 HTTP 作为规则的 URIMAP 定义实施检查，所以 Web 客户机可使用不安全的端口或安全的（SSL）端口访问这些资源。

资源级安全性

一般情况下，要保护对 CICS Web support 的访问，您应要求发出 HTTP 客户机请求的每个用户提供身份，并认证用户声明的身份。『HTTP 客户机的认证和身份识别』说明如何为作为 HTTP 服务器的 CICS 实现这一点。

当已获取 Web 客户机的已认证用户标识时，此标识可用于资源级安全性。资源级安全性不应用于作为静态响应传递的项（即，直接从 URIMAP 定义传递的 CICS 文档模板和 HFS 文件）。如果您需要将基于用户标识的访问控制应用到以这种方式传递的项，则需要将该项作为应用程序生成的响应来进行传递（即，通过使用应用程序）。

对于应用程序生成的响应，Web 客户机的用户标识应用到处理请求中使用的别名事务。别名事务由 TRANSACTION 资源定义来定义，并且 CICS 提供的缺省值是 CWBA。它们以发出请求的 Web 客户机已提供的或已指定给它们的用户标识运行。此用户标识必须具有以下权限：

- 为提供 HTTP 响应的 CICS 应用程序附加别名事务。
- 访问别名事务使用的任何 CICS 资源（如果为别名事务指定资源安全性）。
- 访问别名事务使用的任何 CICS 系统编程命令（如果为别名事务指定命令安全性）。这些系统编程命令会在产生响应的、用户编写的应用程序中使用。

CICS 系统安全

许多组件（如应用程序和资源定义）用于控制 CICS Web support。这些组件在第 17 页的『CICS Web support 的组件』中列出。

在正常的请求过程中，除非可发出 CICS 系统命令的 CICS 提供的事务是 Web support 的，否则不能直接由 Web 客户机访问 CICS 系统组件。然而，与任何其他 CICS 资源一样，不让未授权的用户修改这些组件是很重要的。例如，对端口的 TCPIP SERVICE 定义具有访问权的用户会除去 Web 客户机使用 SSL 或提供身份的要求。

在某些情况下，您可能需要设置其他权限以允许已授权的用户访问：

- 对于 URIMAP 定义，可能需要其他权限以设置 Web 客户机的用户标识。如果 CICS 区域中启用代理用户检查（将 XUSER=YES 指定为系统初始化参数），则 CICS 检查用于安装 URIMAP 定义的用户标识是否被授权为 USERID 属性指定的用户标识的代理。
- 文档模板可用于作为 HTTP 服务器的 CICS 产生的响应的主体，或作为 HTTP 客户机的 CICS 产生的请求的主体。它们由 DOCTEMPLATE 资源定义来定义。如果文档模板存储在已分区的数据集，则 CICS 区域用户标识必须具有此数据集的读权限。
- z/OS UNIX Systems Services HFS 文件可用于从作为 HTTP 服务器的 CICS 产生静态响应的主体。它们可以用它们自己的名称指定或由 DOCTEMPLATE 资源定义来定义。CICS 区域必须有访问 z/OS UNIX 的许可权，而且它必须具有访问包含文件的 HFS 目录以及文件本身的许可权。CICS 中的 *Java* 应用程序说明了如何授予这些许可权。

HTTP 客户机的认证和身份识别

客户机的认证和身份识别使服务器能保护它的资源不让未授权的用户访问。

作为 HTTP 服务器的 CICS: 认证和标识

对于作为 HTTP 服务器的 CICS, 认证规则由 TCPIP SERVICE 定义的 AUTHENTICATE 属性指定。标识可以从认证过程中获取, 或在不需要认证的情况下由 CICS 提供。

从 Web 客户机获取认证和标识是保护您的 CICS 系统不被未经授权的用户访问的一个关键步骤。

认证

CICS 支持两种认证规则, 以及与 HTTP 协议协同使用:

- **基本认证**是一种 HTTP 工具, 它允许客户机通过提供用户标识和密码将其本身认证和标识到服务器。此信息是使用基本 64 位编码来编码的, 解码也比较简单。因为这个原因, 所以仅当不可能拦截密码时, 才适合使用基本认证作为唯一的认证方式。在大多数环境中, 基本认证应该与 SSL 结合使用, 以便 SSL 加密用于保护用户标识和密码信息。第 15 页的『HTTP 基本认证』更详细地说明了基本认证。
- 使用可信第三方(或认证中心)发出的客户机证书, 并使用 SSL 加密发送, **SSL 客户机证书认证**是认证客户机的更安全方法。 *CICS RACF Security Guide* 说明了如何进行此操作。客户机证书不包含可用于 CICS 中的标识的用户标识。为了获得标识, 客户机证书可以在使用前与 RACF 或同等安全管理器中的用户标识关联, 或者当客户机发出其请求时自动关联(使用基本认证)。每次使用证书时, RACF 用户标识就会成为客户机的用户标识。 *CICS RACF Security Guide* 说明了如何设置此操作。

第 71 页的『为 CICS Web support 创建 TCPIP SERVICE 资源定义』告诉您如何为 CICS Web support 设置 TCPIP SERVICE 定义, 该定义指定这些认证规则中的某一个规则。

使用基本认证或客户机证书认证时, CICS 处理这一过程: 用户请求认证、对认证信息进行解码(必要的话)、依照安全管理器的数据库检查所提供的认证, 以及在认证不可接受的情况下拒绝请求。只有在验证和接受了认证后才会调用分析器程序或用户编写的应用程序。

对于基本认证, 如果发现用户提供的密码已到期, 则 CICS 会提示用户提供新密码, 并帮助他们重新提交其请求。CICS 提供的实用程序 DFHWBPW 用于执行此操作。您可以定制 CICS 在此过程期间显示给用户的 Web 页面上的文本。第 111 页的『HTTP 基本认证的密码到期管理』具有执行此操作所需的信息。

对于客户机证书认证, CICS 通过依照安全管理器的数据库并依照您已设置的任何证书撤回列表(可选)检查所提供的证书来验证它。用户编写的应用程序可以检查此过程获取的信息, 如果这可用于确定如何处理请求。使用 EXTRACT CERTIFICATE 命令以检索:

- 发布者或主题的专有名称的组成部分。 *CICS RACF Security Guide* 说明了专有名称。
- 与证书关联的 RACF 用户标识。

标识

标识发生在获取 Web 客户机的用户标识时。在以下情形下, 标识可以从 Web 客户机获取:

- 在基本认证期间。
- 通过用户标识与客户机证书的关联。

CICS 也可能代表 Web 客户机提供用户标识:

- 在请求的 URIMAP 定义中。
- 在应用程序生成的请求的处理路径中使用的分析器程序中。
- 作为 CICS CICS 缺省用户标识 (如果无法确定其他用户标识)。

要注意如果您代表 Web 客户机提供了用户标识, 则不存在客户机标识的认证。只有与您自己的客户机系统通信时才应执行此操作, 而该系统已经认证它的用户, 并与安全环境中的服务器通信。 *CICS RACF Security Guide* 更详细地说明了用户标识是如何确定的。

识别了客户机后, 使用 RACF 或同等安全管理器, 可以授予客户机的用户标识对 CICS 服务的访问权 (与其他任何用户标识一样)。对于应用程序生成的响应, 用户标识应用到别名事务执行的处理, 因此它需要具有与该事务有关的某些权限。第 107 页的『CICS Web support 的 CICS 系统和资源安全性』说明这些是什么。资源级安全性不应用于作为静态响应传递的项 (即, 直接从 URIMAP 定义传递的 CICS 文档模板和 HFS 文件), 并且客户机的用户标识不用于初始访问验证之前的过程。

作为 HTTP 客户机的 CICS: 认证和标识

通过 CICS 发出 HTTP 客户机请求时, 服务器或代理可能要求您执行基本认证、代理认证或 SSL 客户机证书认证。基本认证和代理认证由您的用户应用程序执行。使用 URIMAP 定义可以提供客户机证书。

可能要求您的客户机应用程序按以下方式认证它本身:

- **基本认证**由服务器启动, 以从您这里获取用户标识和密码。对服务器发出请求时, 服务器可能向您发送具有 401 状态码和 WWW-Authenticate 头的响应。头命名需要基本认证的域。为了接收您请求的信息, 需要重复原始请求并添加 Authorization 头, 提供允许您访问域的用户标识和密码。此信息的格式必须为 `userid:password`, 并且它必须是基本 64 位编码的。在后续情况中发出同一请求时, 您可以尝试在原始请求中发送 Authorization 头, 在这种情况下服务器可能不发出提示。请记住, 您的密码可能到期, 那时服务器将再次发出提示。
- **代理认证**由代理服务器启动。它与基本认证以相同的方式操作。对于代理认证, 响应的状态码是 407, 来自代理服务器的提示头是 Proxy-Authenticate, 而您的响应头必须是 Proxy-Authorization。您必须用相同格式和编码提供用户标识和密码信息以进行基本认证。
- **SSL 客户机证书认证**使用可信第三方 (或认证中心) 发出的客户机证书。 *CICS RACF Security Guide* 说明了如何进行此操作。当您发出 HTTPS 请求时, 服务器可能要求, 也可能不要求您提供此认证。 *CICS RACF Security Guide* 告诉您如何获取证书并将它存储在 RACF 数据库或同等外部安全性管理器的密钥环中。如果服务器请求客户机证书, 则 CICS 提供在 URIMAP 定义中指定的证书标号, 在用于连接的 WEB OPEN 命令中使用了该定义。(如果使用 URIMAP 定义却未指定证书标号, 则使用在密钥环中为 CICS 区域用户标识定义的缺省证书。)

某些服务器可能要求您提供其他类型的认证或标识。如果无法将可接受的认证或标识提供给服务器, 您的请求将被拒绝。对于基本认证或代理认证, 服务器拒绝您的请求时使用的状态码与提示的状态码 (对于服务器是 401, 或对于代理是 407) 相同。如果您响应提示, 但是随后接收到进一步响应 (它具有这些状态码中的某一个状态码), 则您使用的授权信息无效。

HTTP 基本认证的密码到期管理

当基本认证用于 HTTP 连接时，CICS Web support 检查外部安全性管理器中的用户标识和密码。如果密码已到期，则 CICS 提供的实用程序 DFHWBPW 用于提示用户选择新的密码。您可通过 DFHWBPW 定制或替换向用户显示的页面。

当应用到请求的 TCPIP SERVICE 定义是使用 AUTHENTICATE 属性的 BASIC、AUTOREGISTER 或 AUTOMATIC 选项定义的，则 DFHWBPW 仅用于密码到期管理。尽管 DFHWBPW 具有类似于转换器程序的结构，但它不是正常 CICS Web support 处理路径的一部分，因此您不需要为其他任何目的向它添加代码。当用户已选择他们的新密码时，DFHWBPW 通过将客户机重定向到原始请求的 URL 来重新启动请求提交，以使请求的完整处理路径与正常的路径一样出现。

DFHWBPW 向用户显示两个 Web 页面：

1. 密码提示页面。此页面包含两个元素：
 - a. 有关密码有效性的消息。显示给用户的初始消息说明密码已到期。如果用户尝试更改密码时有问题（例如，所提供的新密码的两个副本不匹配），则显示更多消息来说明此问题。
 - b. 用户用于更改他们的密码的 HTML 表单。
2. 确认和请求刷新页面。此页面确认已成功替换到期的密码，并提供刷新标记和 URL 链接以便可自动或手工重新发出请求。

DFHWBPW 使用三个 CICS 文档模板（DFHWBPW1、DFHWBPW2 和 DFHWBPW3）构建这些 Web 页面。这些模板的 CICS 提供的定义将它们定义为可装入的程序：即，它们的类型是 PROGRAM(DFHWBPW1) 等等。这些定义在 CICS 提供的 RDO 组 DFHWEB 中。您可以更改这些定义，方法是通过将它们复制到另一个组并使用 RDO ALTER 命令更改它们，以使模板来自不同的源。或者，您可以不更改 RDO 定义而是修改装入的程序。三个程序 DFHWBPW1、DFHWBPW2 和 DFHWBPW3 是仅汇编语言数据模块，并且它们的源在 CICS 样本库 SDFHSAMP 的相应成员中提供给您。您可以修改这些样本，并将它们重新汇编并链接编辑到您连接到 RPL 数据定义语句中的一个正常 CICS 程序库。

提示：当您在汇编语言中对“与”符号（&）进行编码时，您必须将它们输入为双“与”符号（&&）。

每个 DFHWBPW 模板的内容和功能如下所示：

DFHWBPW1

密码提示页面部分。提供页面的 HTML 页面标题，并为可能的密码有效性消息设置符号（使用用于设置符号的服务器端包含技术）。这些消息向用户传达以下信息：

消息 1

密码已到期。

消息 2

输入的用户标识无效。

消息 3

所申请的新密码的两个副本不匹配。

消息 4

输入的前一个密码（刚到期的密码）不正确。

消息 5

由于密码质量规则，外部安全性管理器不接受所申请的新密码。

消息 6

目前已撤销用户标识。

DFHWBPW 程序选择适当的符号以插入到密码提示页面的文档中。您可以定制 DFHWBPW1 以更改页眉和标题，或改变 `<body>` 标记以更改页面颜色或背景。您还可以更改消息符号的内容。

DFHWBPW2

密码提示页面部分。构建 HTML 表单，用户可在其中输入用户标识、旧（已过期）密码和申请的新密码的两个相同副本。您可以定制 DFHWBPW2 以更改用于提示用户的文本，或更改页面的布局。但您不能修改 `<form>` 标记的内容或任何一个 `<input>` 标记。如果您修改，DFHWBPW 可能不会按计划中的工作。

DFHWBPW3

确认和请求刷新页面。该文本通知用户已成功替换已到期的密码，并说明客户机机会立刻提示用户再次输入密码，然后应再次输入新密码。您可以定制文本和页面的布局。

DFHWBPW3 是为重新启动请求过程设计的。它包含 `<meta http-equiv="Refresh">` 标记，当检测到已到期的密码，此标记会在十秒后使自动重定向到用户原来请求过的页面。如果您不希望用户被自动重定向，您可以更改此标记的时间限制或完全除去它。然而，所修改的页面应总是包含指向原来请求的页面的链接。此页面的 URL 在符号 `&dfhwpw_target_url;` 中。重新启动请求过程意味着如果 Web 客户机已高速缓存旧密码，则可以立即用新密码替换它，并且还意味着 CICS Web support 处理路径不受影响。

具有 CICS Web support 的 SSL

安全套接字层（SSL）可与 HTTP 协同使用以启用加密、消息认证，以及使用证书的客户机和服务器认证。HTTPS 规则是具有 SSL 的 HTTP。将 CICS 配置为使用 SSL 后，作为 HTTP 服务的 CICS 和作为 HTTP 客户机的 CICS 都可以使用其工具。

CICS RACF Security Guide 说明了 SSL 提供的工具。*CICS RACF Security Guide* 告诉您如何让 SSL 使用 CICS。

当 CICS 是 HTTP 服务器时，可以使用 SSL 保护与 Web 客户机的交互。为了执行此操作，在 TCPIP SERVICE 定义上为 CICS 接收客户机请求的端口指定适当的安全选项。

与指定对 SSL 的使用一样，您可以要求基本认证或要求客户机证书。要对 Web 客户机提供更多帮助，可以允许客户机提供客户机证书，然后将它们自己注册到安全管理器，为 CICS 环境提供标识。您还可以允许客户机根据需要使用时注册或基本认证以提供标识。所有这些活动由 CICS 本身处理，因此如果您正在提供应用程序生成的响应，则您的应用程序不需要处理它。第 71 页的『为 CICS Web support 创建 TCPIP SERVICE 资源定义』说明如何创建包含这些安全选项的 TCPIP SERVICE 定义。

当 CICS 是 HTTP 客户机时，服务器可能需要对某些连接使用 SSL。如果是这种情况，则需要执行以下某些或所有操作：

- 将 HTTPS 用作连接规则。
- 提供您希望用于连接的密码套件列表。您可以在 URIMAP 定义中指定这些内容，您在用于连接的 WEB OPEN 命令中使用了该定义。
- 提供客户机证书。并非所有 SSL 事务都需要客户机证书，但服务器可能需要用于特殊事务的客户机证书。如果服务器请求了客户机证书，则可以在 URIMAP 定义中指定合适证书的标号，您在用于连接的 WEB OPEN 命令中使用了该定义。客户机证书必须存储在安全管理器的密钥环中。（如果使用 URIMAP 定义却未指定证书标号，则使用在密钥环中为 CICS 区域用户标识定义的缺省证书。）

第 13 章 来自 CICS 应用程序的 HTTP 客户机请求

CICS 可以担当 HTTP 客户机，并与因特网上的 HTTP 服务器通信。用户编写 的应用程序通过 CICS 将请求发送到 HTTP 服务器，并接收来自它的响应。

在 CICS Transaction Server for z/OS V3R1 中，CICS 担当 HTTP 客户机的能力完全集成到 CICS Web support 中。为了达到以下目的，您可能要使用此工具：

- 使用 HTTP 协议控制硬件或软件（例如，通常可以用此方式控制打印机）。
- 访问提供信息项（例如，股价）的 HTTP 应用程序，并检索此信息以在应用程序中使用。

注意，CICS Web support 的 HTTP 客户机工具不旨在用作浏览器。用户应用程序可以对服务器提供的个别、已知资源发出请求，但是浏览因特网时通常不需要它们。可能从服务器接收的响应范围和处理这些响应所需执行的操作应该只与以下内容有关：您预选的资源，以及可能与那些资源和您正在发出的请求类型关联的错误响应。

第 24 页的『作为 HTTP 客户机的 CICS 的 HTTP 请求和响应处理』说明作为 HTTP 客户机的 CICS 的处理结构。编写发出 HTTP 客户机请求的应用程序前，确保您了解这些请求的处理阶段，因为大多数阶段由应用程序本身启动。

当 CICS 是 HTTP 客户机时，使用多个 CICS Web support 工具：

- 您可以在 CICS 中使用 EXEC CICS WEB API 命令打开到服务器的 HTTP 连接、发出请求，并接收供应用程序处理的响应。『通过作为 HTTP 客户机的 CICS 发出 HTTP 请求』说明如何执行此操作。
- 为 CICS 发出的请求和它接收的响应执行代码页转换。第 34 页的『作为 HTTP 客户机的 CICS 的代码页转换』说明此过程。
- URIMAP 定义可用于避免在您的应用程序中指定信息，例如，URL 或证书标号。第 123 页的『通过作为 HTTP 客户机的 CICS 创建 HTTP 请求的 URIMAP 定义』将告诉您如何创建这些内容。
- 对于 HTTP 客户机请求，全局用户出口 XWBOPEN 和 XWBSNDO 使您能指定使用代理服务器，并使您能应用安全策略。请参阅第 124 页的『HTTP 客户机开放出口 XWBOPEN』和第 125 页的『HTTP 客户机发送出口 XWBSNDO』以获取有关这些出口的更多信息。

用于作为 HTTP 服务器的 CICS 的 TCPIP SERVICE 资源定义不应用到作为 HTTP 客户机的 CICS，并且您不需要创建它们以发出 HTTP 客户机请求。

通过作为 HTTP 客户机的 CICS 发出 HTTP 请求

因特网上从 CICS 到服务器进行的 HTTP 客户机请求由用户编写的应用程序启动。本主题将告诉您如何编写发出 HTTP 客户机请求的应用程序。

编写发出 HTTP 客户机请求的应用程序前，阅读有关这些请求的处理阶段，因为大多数阶段由应用程序本身启动。第 24 页的『作为 HTTP 客户机的 CICS 的 HTTP 请求和响应处理』说明应用程序需要执行什么操作，以及该过程期间 CICS 执行什么操作。

对于作为 HTTP 客户机的 CICS，应用程序向服务器发出请求，并等待响应。使用一个会话令牌来区别多个连接，应用程序就可以控制它们。

要发出 HTTP 请求和接收响应，请编写您的应用程序以按以下过程进行操作：

1. 使用 WEB OPEN 命令启动到服务器的连接。『打开到 HTTP 服务器的连接』将告诉您如何执行此操作。此步骤确保服务器可用，并生成可用于管理连接的会话令牌。
2. 使用 WEB WRITE HTTPHEADER 命令为请求写 HTTP 头。第 117 页的『为请求写 HTTP 头』将告诉您如何执行此操作。CICS 自动提供 HTTP/1.1 消息需要的头。您可以出于其他目的提供其他头。
3. 需要的话，产生实体主体或消息体，它是 HTTP 请求的内容。该过程与 CICS 作为 HTTP 服务器时是一样的，如第 65 页的『为 HTTP 消息产生实体主体』中描述的。对于许多请求方法，不使用实体主体，但是对于 POST 和 PUT 方法（它们用于对服务器提供数据），它是必需的。实体主体由 CICS 文档（使用 EXEC CICS DOCUMENT 应用程序编程接口创建）形成，或由应用程序提供的数据缓冲区形成。
4. 使用 WEB SEND 或 WEB CONVERSE 命令将请求发送到 Web 客户机。第 118 页的『写 HTTP 请求』将告诉您如何执行此操作。您需要选择合适的方法，并指定实体主体。CICS 使用这些项和 HTTP 头汇编请求。如果要使用分块的传输编码，还需要按第 67 页的『使用分块的传输编码发送 HTTP 请求或响应』中的特殊指示信息进行操作。
5. 如果要将更多请求作为管道化的序列发送，则使用第 120 页的『发送管道化序列的请求』中的指导来执行此操作。
6. 等待并接收请求，按第 121 页的『接收 HTTP 响应』中的过程进行操作：
 - a. 使用 WEB RECEIVE 命令（或您先前发出的 WEB CONVERSE 命令）接收响应的消息体。
 - b. 使用 WEB READ HTTPHEADER 命令或 HTTP 头浏览命令读取响应的头。
 - c. 根据状态码，处理服务器的响应，并执行应用程序的业务逻辑。
 - d. 如果发送了管道化序列的请求，则使用更多 WEB RECEIVE 命令从服务器接收其余响应。
7. 如果需要更多请求和响应，请重复该过程。如果服务器支持持续连接，且不关闭连接，则不需要打开一个新连接。您可以继续使用同一会话令牌。服务器关闭连接后，如果要发出更多请求，则需要再次发出 WEB OPEN 命令。
8. 完成与服务器的合作后，关闭连接。第 122 页的『关闭到 HTTP 服务器的连接』说明如何执行此操作。

打开到 HTTP 服务器的连接

在 CICS Web support 中发出 HTTP 客户机请求时，必须在发送第一次请求前打开到服务器的连接。CICS 返回表示连接的会话令牌。

通过发出 WEB OPEN 命令以启动与服务器的连接，具体如下：

1. 指定服务器的主机名、主机名长度和要使用的规则（HTTP 或 HTTPS）。如果主机的端口号不是已指定规则的缺省值，则也要指定它。您还可以在 WEB OPEN 命令中指定 URIMAP 选项，以直接从现有 URIMAP 定义使用此信息。也可以使用 SCHEME、HOST、HOSTLENGTH 和 PORTNUMBER 选项提供信息。要抽取这些

详细信息，可以使用 WEB_PARSE_URL 命令从已知 URL 抽取，或使用 WEB_EXTRACT URIMAP 命令从现有 URIMAP 定义抽取。

2. 需要的话，指定 CODEPAGE 选项将此连接的 EBCDIC 代码页更改为与本地 CICS 区域的缺省代码页（由 LOCALCCSID 系统初始化参数设置）不同的代码页。这可能是另一种本地语言的 EBCDIC 代码页。服务器返回其响应时，如果指定了转换，则 CICS 将响应主体传递到应用程序之前将它转换到此代码页中。
3. 如果您使用 HTTPS 规则，则指定适当的安全选项：
 - a. 如果您需要提供 SSL 客户机证书，则指定 CERTIFICATE 选项来执行此操作。如果您在 WEB_OPEN 命令中指定 URIMAP 选项，则可以直接使用现有 URIMAP 定义中的信息。
 - b. 使用 CIPHERS 和 NUMCIPHERS 选项指定用于连接的密码套件代码列表。如果您在 WEB_OPEN 命令中指定 URIMAP 选项，则可以接受 URIMAP 定义的设置，或指定您自己的密码套件代码列表以覆盖 URIMAP 规范。
4. WEB_OPEN 命令驱使 XWBOPEN 用户退出。需要的话，您可以创建一个用户出口程序以使到服务器的连接通过代理服务器，或者将安全策略应用到主机名。第 124 页的『HTTP 客户机开放出口 XWBOPEN』具有帮助您执行此操作的信息。

CICS 打开与服务器的连接，并将会话令牌返回给应用程序。CICS 还返回有关服务器的 HTTP 版本的信息。

保存会话令牌并将它用于与此连接有关的所有后续命令。

为请求写 HTTP 头

对于客户机 HTTP 请求，根据用于消息的 HTTP 协议版本，CICS 自动提供基本消息必需的 HTTP 头。您可能需要将更多 HTTP 头添加到您的请求。

如果消息需要某些 HTTP 头，则 CICS 将自动创建它们。CICS 创建的完整头列表如下：

- Connection
- Content-Length
- Content-Type
- Date
- Expect
- Host
- Server
- TE（由 CICS 写，但是可能添加了更多实例）
- Transfer-Encoding
- User-Agent
- WWW-Authenticate

这些头中的一部分只适合于作为 HTTP 服务器的 CICS。创建这些头的情况在第 185 页的附录 B，『CICS Web support 的 HTTP 头参考』中描述。CICS 不允许写自己的、CICS 提供的请求头版本。

CICS 为请求提供的头是通常应该为基本 HTTP/1.1 消息所写的、以使该消息遵从 HTTP/1.1 规范的头。（CICS 以提供的 HTTP/1.1 为 HTTP 版本发送您的请求。）您可能为了诸如以下目的要添加更多 HTTP 头：

- 声明服务器的首选项（例如，Accept-Encoding、Accept-Language）
- 发出条件请求（例如，If-Match、If-Modified-Since）
- 对服务器或代理提供基本认证信息（Authorization、Proxy-Authorization）

要获取与您决定用于消息的任何其他 HTTP 头相关的需求，请检查您正在遵从的 HTTP 规范。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。

在您发出 WEB SEND 命令以发生消息前，为消息编写其他 HTTP 头。此规则的特例是您编写的头要在已分块消息上作为尾部头发送，在此情况下，应用下面提到的特殊进程。为请求写 HTTP 头时：

- 对于所有命令，使用 SESSTOKEN 选项为此连接指定会话令牌。
- 对您要添加到消息的每个头使用 WEB WRITE HTTPHEADER 命令。确保以您正在遵从的 HTTP 规范所描述的格式为每个头指定名称和值。该命令添加单个头，您可以重复该命令以添加更多头。如果您编写已经为请求写好的头，则 CICS 除了现有头外，还将新的头添加到请求。确保只在 HTTP 规范声明可以重复头的地方执行此操作。CICS 发送请求时，会存储准备添加到该请求的头。
- 如果您使用的任何 HTTP 头可能不适合低于 HTTP/1.1 级别的服务器，则写那些头前，请检查 WEB OPEN 命令返回的 HTTP 版本信息。为了允许您使用用户定义的（非标准）头，CICS 不会除去用户所写的不适用的头。低于 HTTP/1.1 级别的服务器无法理解某些 HTTP 头，并且可能导致处理您的请求时出错。

注：CICS 对低于 HTTP/1.0 级别的服务器或 Web 客户机不提供任何特殊支持。CICS 运作时假定它们都处于 HTTP/1.0 级别，并将 HTTP/1.0 作为 HTTP 版本返回。

- 如果要产生要在某个 HTTP 头（例如，If-Modified-Since 头）中使用的日期和时间戳记，则可以使用 FORMATTIME 命令。该命令上的 STRINGFORMAT 选项将当前日期和时间（ABSTIME 格式的）或应用程序产生的日期和时间转换为 RFC 1123 格式。RFC 1123 格式是 CICS 为了在 Web 上使用而创建的唯一日期和时间戳记格式。其他日期和时间戳记格式可能不被与 CICS 通信的某些 Web 客户机或服务器所接受。
- 如果正在使用分块的传输编码发送 HTTP 请求，并且要将尾部头包括在分块消息的末尾，则按第 67 页的『使用分块的传输编码发送 HTTP 请求或响应』中的特殊指示信息进行操作。在发送消息的第一块前，需要写尾部头。继 WEB SEND 命令后为第一块写的所有 HTTP 头作为尾部头来对待。
- 确保您的应用程序执行用户所写的头暗示的任何操作。

写 HTTP 请求

对于作为 HTTP 客户机的 CICS，WEB SEND 命令或 WEB CONVERSE 命令可用于发出请求。WEB CONVERSE 命令组合了 WEB SEND 命令和 WEB RECEIVE 命令中可用的选项，因此您可以使用单条命令以发出请求和接收响应。

发出请求时，您需要指定 HTTP 方法。该方法告诉服务器对您的请求执行何种操作。第 199 页的附录 D，『CICS Web support 的 HTTP 方法参考』提供有关能够与 CICS Web support 协同使用的不同方法的基本指导。要获取更详细的指导（包括应用到方法使用的

任何需求)，应该查询您正在遵从的 HTTP 规范。请参阅第 8 页的『HTTP 协议』以获取有关 HTTP 规范的更多信息。CICS 以提供的 HTTP/1.1 为 HTTP 版本发送您的请求。

发出请求前，使用 WEB WRITE HTTPHEADER 命令写请求的任何额外 HTTP 头，如第 117 页的『为请求写 HTTP 头』中所描述。

需要的话，请求可以按块（分块的传输编码）发送，或者可以发送管道化的请求序列。

CICS Application Programming Reference 提供 WEB SEND 和 WEB CONVERSE 命令中可用的选项的完整参考信息和描述。发出所选的命令时：

1. 使用 SESSTOKEN 选项为此连接指定会话令牌。
2. 为请求指定 HTTP 方法（OPTIONS、GET、HEAD、POST、PUT、DELETE 或 TRACE）。第 199 页的附录 D，『CICS Web support 的 HTTP 方法参考』具有正确使用每个这些方法的指导。
3. 为应用程序需要访问的服务器上的资源指定路径信息。缺省值是任何 URIMAP 定义中提供的路径，您在此连接的 WEB OPEN 命令中引用了该定义。通过使用 URIMAP 选项指定可以从中获取路径的另一个 URIMAP 定义，可以指定备用路径。（新的 URIMAP 定义必须为当前连接指定正确的主机名。）也可以使用 PATH 和 PATHLENGTH 选项来提供路径信息。
4. 使用 QUERYSTRING 和 QUERYSTRLEN 选项为您的请求指定任何查询字符串。
5. 为 HTTP 请求指定任何实体主体及其长度。第 199 页的附录 D，『CICS Web support 的 HTTP 方法参考』具有有关哪里适合使用请求主体而哪里不适合使用请求主体的信息。
 - 对于 GET、HEAD、DELETE 和 TRACE 方法，请求主体不适合。
 - 对于 OPTIONS 方法，请求主体是允许的，但是目前 HTTP/1.1 规范不会为该主体定义任何目的。
 - 对于 POST 和 PUT 方法，必须使用请求主体。

如果需要请求主体，则主体内容可以从 CICS 文档（使用 CICS DOCUMENT 接口并指定 DOCTOKEN 选项以识别文档）形成，或者从缓冲区内容（指定 FROM 选项）形成。第 65 页的『为 HTTP 消息产生实体主体』说明如何产生此内容，并具有有关主体长度的大小限制的信息。

6. 使用 MEDIATYPE 选项为您正在提供的任何实体主体指定介质类型。对于具有 POST 和 PUT 方法的请求（它需要主体），需要指定 MEDIATYPE 选项。对于具有其他方法的请求（其中未提供主体内容），不需要 MEDIATYPE 选项。
7. 如果请求主体不需要代码页转换，则指定适当的转换选项（根据您正在使用的是 WEB SEND 命令还是 WEB CONVERSE 命令），因此 CICS 不转换请求主体。对于作为 HTTP 客户机的 CICS，缺省设置是转换请求主体，除非它具有非文本介质类型。
8. 如果需要代码页转换，并且缺省 ISO-8859-1 字符集不适合，则指定适合于服务器的字符集。大多数服务器都应接受 ISO-8859-1，除非您区别于服务器选择备用项的较早连接。
9. 如果要使用 Expect 头来测试服务器是否接受请求，则为 ACTION 选项指定 EXPECT。此设置使 CICS 将 Expect 头与请求行和请求头一起发送，并在将消息

体发送到服务器前等待 100-Continue 响应。如果接收到除 100-Continue 外的响应，则 CICS 通知应用程序并取消发送。如果等待一段时间后未接收到响应，则 CICS 无论如何也发送消息体。

10. 如果这是您要对此服务器发出的最后一次请求，则为 CLOSESTATUS 选项指定 CLOSE。CICS 写请求的 Connection: close 头，或者对于 HTTP/1.0 级别的服务器，省略 Connection: Keep-Alive 头。在发出最后一次请求时指定此选项是好行为，因为头中的信息意味着服务器在发送最后一条响应后可以立即关闭它与您的连接，而不是等着看您是否在超时前发送更多请求。仍将接收来自服务器的响应，并且该响应对应用程序可用。然而，您将无法使用此连接将任何进一步请求发送到服务器。
11. 如果要使用分块的传输编码将请求主体作为一系列块发送，则按第 67 页的『使用分块的传输编码发送 HTTP 请求或响应』中的其他指示信息进行操作。

注：以下各项不支持分块的传输编码：

- 低于 HTTP/1.1 级别的服务器。
- WEB CONVERSE 命令。
- CICS 文档 (DOCTOKEN 选项)。

12. 如果要发送管道化序列的请求，则按『发送管道化序列的请求』中的指示信息进行操作。

CICS 汇编请求行、HTTP 头和请求主体，并将请求发送到服务器。

发送管道化序列的请求

您可以发送更多请求而无需等待来自服务器的响应。这称为管道传送。使用 WEB SEND 命令来发送管道化的请求，而不能使用 WEB CONVERSE 命令（因为该命令包括等待响应）。

第 30 页的『CICS Web support 如何处理管道传送』具有有关管道传送的更多信息。如果您要管道化请求：

1. 确保与服务器有持续连接。HTTP/1.1 规范允许您尝试一次以发送管道化的序列，而无需检查连接是否持续。如果此尝试失败，则必须在重试请求前进行检查。要确定连接的特性：
 - a. 检查 WEB OPEN 命令返回的 HTTP 版本信息（或使用 WEB EXTRACT 命令以获取此信息）。
 - b. 如果已从服务器接收到先前的响应，则使用 WEB READ HTTPHEADER 命令以检查服务器是否发送 Connection:close 头或 Connection: Keep-Alive 头。

处于 HTTP/1.1 级别且不发送 Connection: close 头的服务器，以及处于 HTTP/1.0 级别且发送 Connection:Keep-Alive 头的服务器支持持续连接。

2. HTTP/1.1 规范声明：您的请求序列应该是幂等的；即，如果重复所有或部分序列，则应该获得同样的结果。第 14 页的『管道传送』具有有关幂等性的更多信息。
3. 除了管道化序列中的最后一个请求外，不要对任何其他请求指定 CLOSESTATUS(CLOSE)。

接收 HTTP 响应

WEB RECEIVE 命令或 WEB CONVERSE 命令用于从服务器接收响应。（WEB CONVERSE 命令结合了 WEB SEND 和 WEB RECEIVE 命令的功能。）WEB READ HTTPHEADER 命令或 HTTP 头浏览命令用于检查头。

应用程序为了接收响应准备等待的时间由 RTIMOUT 值表明，该值在别名事务的事务概要文件定义中指定。超时限制不应用到对响应头的读取。

当 RTIMOUT 指定的时期到期后，CICS 向应用程序返回 TIMEDOUT 响应。RTIMOUT 值为零意味着应用程序准备无限期地等待。事务概要文件定义上 RTIMOUT 的缺省设置为零，因此检查并更改该设置很重要。

使用 WEB RECEIVE 或 WEB CONVERSE 命令：

1. 使用 SESSTOKEN 选项为此连接指定会话令牌。
2. 指定用于接收服务器发送的 HTTP 状态码的数据区，以及为描述状态码而由服务器返回的任何文本。请注意，数据以其未转义的格式返回。
3. 指定数据区以接收响应主体的介质类型。
4. 通过指定 INTO 选项（对于数据缓冲区）或 SET 选项（对于指针引用）和 LENGTH 选项来接收响应主体。该数据以其转义格式返回，并转换到适合应用程序的代码页中，除非您另外请求。
5. 如果要限制从响应主体接收的数据量，则指定 MAXLENGTH 选项。如果要保留（而不是废弃）超出此长度的任何数据，则也指定 NOTRUNCATE 选项。可以使用更多 WEB RECEIVE 命令来获取任何剩余数据。如果已使用分块的传输编码发送数据，则 CICS 将块组装到单个消息中，然后再将它传递到应用程序，因此 MAXLENGTH 选项应用到已分块消息的整个实体主体长度，而不是应用到每个独立的块。
6. 如果响应主体不需要代码页转换，则指定适当的转换选项（根据您正在使用的是 WEB RECEIVE 命令还是 WEB CONVERSE 命令），因此 CICS 不转换响应主体。缺省情况是应该发生转换，且在该情况下 CICS 将服务器的响应主体转换为本地 CICS 区域的缺省代码页，或者在 WEB OPEN 命令中指定的任何备用 EBCDIC 代码页。

注：如果您接收已打包或已压缩的实体主体，如消息的 Content-Encoding 头表明的，则确保抑制了代码页转换。CICS 不对这些消息类型进行解码，并且如果应用代码页转换，结果将不可预测。如果您不希望接收已打包或已压缩的实体主体，则可以在对服务器的请求中使用 Accept-Encoding 头来指定。

发出 WEB RECEIVE 或 WEB CONVERSE 命令时，CICS 从状态行返回响应主体和信息。

7. 使用适当的 CICS 命令检查服务器的响应的 HTTP 头：
 - 如果要读取特定 HTTP 头（您知道服务器提供了该头），则使用 WEB READ HTTPHEADER 命令检查该头的内容。您的应用程序必须提供将接收头内容的缓冲区。如果请求中不存在头，则 CICS 返回 NOTFND 条件。
 - 如果要浏览响应中的所有 HTTP 头，则使用 WEB STARTBROWSE HTTPHEADER 命令开始浏览头行。使用 WEB READNEXT HTTPHEADER 命令以获取每一行的头名称和头值。您的应用程序必须提供两个缓冲区：一个将接

收头名，而另一个将接收其内容。读取所有头后，CICS 将返回 ENDFILE 条件。当您的程序已获取所有相关头信息后，使用 WEB ENDBROWSE HTTPHEADER 命令。

记得在每条 HTTP 头命令中包括会话令牌。

8. 处理服务器的响应并执行应用程序的业务逻辑。如果响应具有“常规”或参考状态码（例如，200（确定）），则可以如常处理响应。（发出 WEB RECEIVE 命令时，接收到状态码。）如果响应具有表明错误或请求更多操作的状态码，则应该执行备用处理以解决此问题。第 191 页的附录 C，『CICS Web support 的 HTTP 状态码参考』具有有关对状态码做出响应的基本指导。
9. 如果发送了管道化序列的请求，则使用更多 WEB RECEIVE 命令从服务器接收其余响应。CICS 保留响应并按 CICS 从服务器接收它们的顺序将它们返回到应用程序。处理管道化的请求的服务器将以接收请求相同顺序提供响应。

提示：正在接收对管道化的请求的响应时，如果您正在使用多个 WEB RECEIVE 命令接收过长的消息体，则请仔细地记录已发出多少 WEB RECEIVE 命令。您可能发现在单条 WEB RECEIVE 命令中接收这些响应中每一个的整个主体会更方便。

关闭到 HTTP 服务器的连接

当 CICS 是 HTTP 客户机时，CICS 和服务器之间的连接可以由服务器终止，或由 CICS 按应用程序发出的命令进行操作来终止，还可以在任务结束时终止。

为了有效管理连接关闭，建议进行以下行为：

1. 在要对服务器发出的最后一次请求上，为 WEB SEND 或 WEB CONVERSE 命令上的 CLOSESTATUS 选项指定 CLOSE。CICS 写请求的 Connection: close 头，或者对于 HTTP/1.0 级别的服务器，省略 Connection: Keep-Alive 头。在发出最后一次请求时指定此选项是好行为，因为头中的信息意味着服务器在发送最后一条响应后可以立即关闭它与您的连接，而不是等着看您是否在超时前发送更多请求。仍将接收来自服务器的响应，并且该响应对应用程序可用。然而，您将无法使用此连接将任何进一步请求发送到服务器。指定 CLOSESTATUS 选项与发出 WEB CLOSE 命令的效果范围不同。
2. 如果需要测试服务器是否已请求连接终止，则使用 WEB READ HTTPHEADER 命令在来自服务器的最后一条消息中查找 Connection: close 头。如果服务器终止连接，则应用程序无法使用该连接发送更多请求，但是它可以接收服务器在它终止连接前发送的响应。
3. 完成所有 HTTP 请求和响应后，发出 WEB CLOSE 命令，指定会话令牌。连接关闭后，应用到它的会话令牌不再有效以供使用。需要会话令牌以从服务器接收响应并读取响应的 HTTP 头，因此直到已完成与服务器和它发送的最后一条响应的所有交互后，才应该发出 WEB CLOSE 命令。如果应用程序不发出 WEB CLOSE 命令，则连接在任务结束时关闭。

通过作为 HTTP 客户机的 CICS 创建 HTTP 请求的 URIMAP 定义

此主题告诉您如何创建 URIMAP 定义，它指定 HTTP 客户机请求的 URL 的组成部分（规则、主机和路径）以及用于请求的 SSL 客户机证书（如果需要）。可以在 WEB OPEN 命令中指定 URIMAP 定义，来为连接提供规则和主机名以及缺省路径。还可以在 WEB SEND 命令中命名它，来为相关请求提供路径。或者，您可使用 WEB EXTRACT URIMAP 命令从 URIMAP 定义抽取信息并直接在发出 HTTP 客户机请求的应用程序中使用。

1. 识别您计划用于 HTTP 客户机请求的 URL 此 URL 表示您计划在服务器上访问的资源。
2. 识别客户机证书对于请求是否是必需的，并获取适当的证书标号。如果用于请求的规则是 HTTPS（带 SSL 的 HTTP），则服务器可能请求 SSL 客户机证书。如果发生这种情况，则 CICS 提供 URIMAP 定义中指定的证书标号。
3. 将请求的 URL 划分成它的规则、主机和路径部分。第 7 页的『URL 的组成部分』说明这些每一个组成部分以及如何对它们定界。如果 URL 中已显式地指定端口号，则还应使用端口号。例如，在 URL `http://www.research.ibm.com:1030/software/htp/cics/index.html` 中：

- 规则部分是 `http`
- 主机部分是 `www.research.ibm.com`
- 端口号是 `1030`
- 路径部分是 `/software/htp/cics/index.html`

跟随规则部分的定界符 `://`（冒号和两个正斜杠）不用于 URIMAP 定义。路径部分开始的定界符 `/`（正斜杠）可以包含在 PATH 规范中或省略；如果您省略它，则 CICS 会自动提供它。如果您要在请求的 URL 中提供查询字符串，则可以使用 QUERY 选项在 WEB SEND 命令中指定它。

4. 使用 *CICS Resource Definition Guide* 中列出的某个方法，开始具有名称和您的一组选项的 URIMAP 定义。
5. 使用 STATUS 属性指定应在启用状态还是在禁用状态安装 URIMAP 定义。
6. 指定 CLIENT（作为 HTTP 客户机的（CICS）的 USAGE 属性。
7. 指定 SCHEME 属性作为请求的 URL 的规则部分。可使用 HTTP（不带 SSL）或 HTTPS（带 SSL）。不要包含跟随规则部分的定界符 `://`。
8. 指定 HOST 属性作为请求的 URL 的主机部分。如果 URL 中已显式地指定端口号，则包含端口号（以冒号作为前缀）。
9. 指定 PATH 属性作为请求的 URL 的路径部分。不能在作为 HTTP 客户机的 CICS 的 URIMAP 定义中使用通配符（星号）。您可以在路径部分开始包含或省略定界符 `/`（正斜杠）；如果您省略它，CICS 会自动提供它。如果在 WEB OPEN 命令中引用 URIMAP 定义，则此路径成为该连接的 WEB SEND 命令的缺省路径。如果在 WEB SEND 命令中引用 URIMAP 定义，则此路径用于该 WEB SEND 命令，但注意，此 URIMAP 定义的主机属性必须与该连接的 WEB OPEN 命令中指定的主机匹配。不要在路径部分包含查询字符串；您可以使用 QUERY 选项在 WEB SEND 命令中指定它。
10. 可选：如果使用 SSL，则指定 CERTIFICATE 属性作为要用作此请求的 SSL 客户机证书的证书标号。
11. 可选：如果使用 SSL 或 TLS，则指定 CIPHERS 属性作为要用于此请求的密码。

HTTP 客户机开放出口 XWBOPEN

XWBOPEN 使系统管理员能指定代理服务器，而这些代理服务器应该用于作为 HTTP 客户机的 CICS 的 HTTP 请求，并使系统管理员能将安全策略应用到为那些请求指定的主机名。处理 EXEC CICS WEB OPEN 命令期间会调用 XWBOPEN，应用程序使用该命令打开与服务器的连接。

关于作为 HTTP 客户机的 CICS 的 HTTP 请求的代理服务器的使用（或其他方面），CICS 本身没有任何要求，并且 CICS 不将任何安全策略应用到那些请求。如果您的系统或组织需要这些设施，则由您负责设置它们。

EXEC CICS WEB OPEN 命令指示 CICS Web 域打开与服务器的连接。在打开连接之前调用 XWBOPEN。EXEC CICS WEB OPEN 命令的 HOST 选项指定的连接的主机名（例如，www.example.com），作为 UEPHOST 参数传递给用户出口程序以供检查。此时，用户出口程序可用于两个目的：

- **确定 HTTP 请求是否需要使用代理服务器，并返回要求的任何代理服务器的名称。** 如果需要代理服务器，则使用返回码 UERCPROX，并且将代理服务器的名称返回到 CICS Web 域（在 UEPPROXY 标识的缓冲区中）并使用它连接到服务器。如果不需要代理服务器，则使用返回码 UERCNORM。
- **将安全策略应用到主机名。** 返回码 UERCBARR 表明不允许对主机的访问。如果不允许对主机的访问，则将 INVREQ 响应返回给 WEB OPEN 命令，并且应用程序员应放弃尝试打开此连接。如果您要为个别资源以及主机资源（或不是主机资源）应用安全策略，则可以在 EXEC CICS WEB SEND 和 EXEC CICS WEB CONVERSE 命令中使用 XWBSNDO 用户出口将安全策略应用到 URL 的路径部分。

XWBOPEN 用户出口不支持使用 EXEC CICS 命令。

样本程序 DFH\$WBPI 和 DFH\$WBEX 以及关联的副本 DFH\$WBGA 显示如何在全局工作区中设置代理服务器信息或安全策略。例如，如果来自您的 CICS 系统的所有请求都应使用单个代理服务器，则您可以指定代理服务器名作为初始化参数。如果您使用很多代理服务器或者要为不同的主机名应用安全策略，则可以装入或者构建一个将主机名与相应的代理服务器相匹配的，或者将它们标记为禁止的表，然后，在处理 EXEC CICS WEB OPEN 命令期间该表可用作查找表。可以在初始化后的程序列表（PLTPI）处理期间或期望使用 EXEC CICS WEB OPEN 命令之前的任何时候运行样本程序。

出口 XWBOPEN

何时调用

处理 EXEC CICS WEB OPEN 命令期间。

特定于出口的参数

UEPHOST (CICS 提供的输入)

字段的地址，此字段包含 WEB OPEN 命令的 HOST 选项中指定的主机名。

注：当主机名保存在此字段中时，它将转换为小写字母。在匹配主机名时您的用户出口程序应考虑这一点。

UEPHOSTL (CICS 提供的输入)

包含主机名的半字长度的字段的地址。

UEPPROXY (用户出口提供的输出)

字段的地址, 该字段包含指向代理服务器名称的地址。在用户出口程序的输入中, 该参数设置为字段的地址, 该字段包含 2046 个字节区域的地址。您可以在该区域中放置代理服务器名, 并且不更改 UEPPROXY 中的地址。或者, 您可以在自己的区域中放置代理服务器名, 并用包含您自己区域的地址的字段地址替换 UEPPROXY 中的地址。

UEPPROXYL (用户出口提供的输出)

包含代理服务器名的半字长度的字段的地址。

返回码

UERCNORM

对于此 HTTP 请求, 不需要代理服务器, 且不禁主机名。

UERCPROX

对于此 HTTP 请求, 需要代理服务器。已将 UEPPROXY 设置为所需的代理服务器的名称, 并且已将 UEPPROXYL 设置为代理服务器名的长度。

UERCBARR

服务器的主机名被禁止。

UERCERR

出口处理中发生错误。

XPI 调用

可以使用所有 XPI 调用。

API 和 SPI 命令

不能使用 EXEC CICS 命令。

HTTP 客户机发送出口 XWBSNDO

XWBSNDO 使系统管理员能为作为 HTTP 客户机的 CICS 的 HTTP 请求指定安全策略。处理 EXEC CICS WEB SEND 或 EXEC CICS WEB CONVERSE 命令期间调用 XWBSNDO。主机名和路径信息被传递到出口, 并且可将安全策略应用到这些任一组件或所有这些组件。

CICS 本身不应用作为 HTTP 客户机的 CICS 的 HTTP 请求的任何安全策略; 如果您的系统或组织需要此设施, 则由您负责设置它。

WEB OPEN 命令中的 XWBOPEN 出口可用于阻碍对整个主机的访问, 并且 XWBSNDO 出口可用于执行同样的操作或阻碍对主机中特定路径的访问。如果您希望阻碍对整个主机的访问, 用 XWBOPEN 出口这样做会节省时间, 这是因为应用程序无法打开连接并因此在创建应发送的请求时不会浪费时间。提供给 XWBSNDO 出口的主机名的主要目的是使您能区分不同主机使用的相同路径。

如果分块的传输编码用于 HTTP 请求, 则 XWBSNDO 仅在分块的消息的第一个 WEB SEND 命令中调用。

XWBSNDO 用户出口不支持使用 EXEC CICS 命令。

主机作为 UEPHOST 参数传递到用户出口程序, 而路径作为 UEPPATH 参数传递。返回码 UERCNORM 表明路径是允许的, 而返回码 UERCBARR 表明该路径是不允许的。如果路径是不允许的, 则 INVREQ 响应返回到 WEB SEND 或 WEB CONVERSE 命

令，并且应用程序员应该通过使用 WEB CLOSE 命令关闭连接来处理此情况。

出口 XWBSNDO

何时调用

为作为 HTTP 客户机的 CICS 的 HTTP 请求处理 EXEC CICS WEB SEND 或 EXEC CICS WEB CONVERSE 命令期间。在 WEB SEND 命令中使用 SESSTOKEN 参数来表明客户机请求。

特定于出口的参数

UEPHOST

字段的地址，此字段包含在连接的 WEB OPEN 命令的 HOST 选项中指定的主机名。

注：当主机名保存在此字段中时，它将转换为小写字母。在匹配主机名时，您的用户出口程序应该考虑到这一点。

UEPHOSTL

包含主机名的半字长度的字段的地址。

UEPPATH

字段的地址，此字段包含 WEB SEND 命令的 PATH 选项中指定的路径。该路径是大小写混合的，如同指定它的方式一样。

UEPPATHL

包含路径的半字长度的字段的地址。

返回码

UERCNORM

允许路径。

UERCBARR

不允许路径。

XPI 调用

可以使用所有 XPI 调用。

API 和 SPI 命令

不能使用 EXEC CICS 命令。

第 14 章 CICS Web support 和非 HTTP 请求

您可以将 CICS Web support 用于作为 HTTP 服务器的 CICS 以处理来自客户机的 TCP/IP 请求，这些请求不是 HTTP 格式的。然而，对 HTTP 请求的处理和对非 HTTP 请求的处理是分开进行的。这确保 CICS 可以对 HTTP 请求和响应执行基本验收检查，并确保非 HTTP 请求不执行这些检查。这些验收检查将为非 HTTP 请求产生错误响应，且不处理请求。

必须在开始前配置 CICS Web support 的基本组件，如第 41 页的第 4 章，『配置 CICS Web support 基本组件』中所述。

CICS Web support 工具用于处理非 HTTP 请求，但是帮助您解释 HTTP 请求并构造响应的许多接口不可用。CICS TS V3 中介绍的改进（包括分块的传输编码）通常对非 HTTP 请求也不可用。持续连接不受支持。

以下 CICS Web support 的特定于任务的组件用于处理非 HTTP 请求：

- TCPIPService 资源定义。
- 分析器程序。
- 转换器程序（如果需要）。
- 用户编写的应用程序。
- 应用程序的别名事务。

URIMAP 定义不与非 HTTP 请求协同使用。

注：作为 HTTP 客户机的 CICS 无法发出非 HTTP 请求。通过 CICS Web support 发出的客户机请求使用 HTTP 协议。

1. 决定将用于请求的端口。配置 Communications Server 具有有关选择和定义可以由 CICS Web support 使用的端口的更多信息。请注意，因为每个端口只能存在一个活动的 TCPIPService 定义，所以非 HTTP 请求不能与 HTTP 请求使用同一端口。熟知端口号 80（对于 HTTP）和 443（对于 HTTPS）必须具有 HTTP 协议，因此它们不能接受非 HTTP 请求。发出非 HTTP 请求的 Web 客户机必须在 URL 中为其请求明确指定端口号。
2. 使用第 128 页的『非 HTTP 请求的资源定义』中的信息，为请求设置资源定义。非 HTTP 请求的 TCPIPService 定义必须指定 USER 协议。您还可以创建别名事务以覆盖请求处理。
3. 使用第 128 页的『分析器程序和非 HTTP 请求』中的信息，对分析器程序编码以处理每个请求。需要分析器程序以确定请求的处理。它指定应用程序和（如果使用）转换器程序以处理每个请求。它还可以指定代码页转换参数。
4. 使用第 129 页的『用于非 HTTP 请求的应用程序编程』中的信息，设计和编码一个或多个应用程序以对每个请求提供一个响应。这些程序可以使用 EXEC CICS WEB 编程接口的特定元素。它们还可以是不支持 Web 的应用程序，并产生由转换器程序编码的输出。

非 HTTP 请求的资源定义

非 HTTP 请求需要 TCPIPService 和 TRANSACTION 资源定义。非 HTTP 请求的 TCPIPService 资源定义必须指定 USER（用户定义的）协议，此协议与 CICS 提供的事务 CWXU 关联。当通过 USER 协议接收请求时，不使用 URIMAP 资源定义。

1. 为您用于非 HTTP 请求的每个端口创建带 USER 协议的 TCPIPService 资源定义。可与 USER 协议协同使用的属性与可与 HTTP 协议协同使用的属性相同。第 71 页的『为 CICS Web support 创建 TCPIPService 资源定义』将告诉您如何执行此操作。
2. 对于每个 TCPIPService 资源定义，决定是否使用 CICS 提供的事务 CWXU、CICS Web 用户定义的协议连接事务或备用事务。DFHCURDI 样本包括 CWXU 的样本定义。CWXU 执行 CICS 程序 DFHWPBXN。可能使用执行 DFHWPBXN 的备用事务，但是不使用为 TCPIPService 资源定义上的协议定义的其他缺省事务。
3. **可选：** 为您要用于请求处理的任何别名事务创建 TRANSACTION 资源定义。第 74 页的『为 CICS Web support 创建 TRANSACTION 资源定义』将告诉您如何执行此操作。

分析器程序和非 HTTP 请求

需要分析器程序以处理非 HTTP 请求。它可以构造为了通过网络传输而分割的请求，指定请求的代码页转换，并执行确定后续请求处理所需要的任何语法分析。

重建非 HTTP 请求

入局请求可以分割成多部分，以通过网络传输。对于非 HTTP 请求，CICS 不会在调用分析器程序前重建请求，而且您应该相应地编写您的分析器代码。

在分析器的入口处，user_data 指针对包含第一部分入局请求的通信区域寻址。要接收请求的下一部分，将返回码设置为 URP_EXCEPTION，并将原因码设置为 URP_RECEIVE_OUTSTANDING。CICS Web support 再次调用分析器，而 user_data 指针对消息的下一部分寻址。可以按需要多次重复这个过程直到整个请求被接收，直到最大支持长度 32767 字节。

此过程的结果对于 CICS WEB API 命令不可视。然而，重建的消息可以传递到转换器程序。

为非 HTTP 请求指定代码页转换

对于非 HTTP 请求，CICS Web support 在调用分析器程序前不对请求执行任何代码页转换。

使用代码页转换表（DFHCNV）密钥或客户机和服务器代码页输出参数，分析器可以指定非 HTTP 请求的代码页转换，就如它对 HTTP 请求所做的那样。第 93 页的『编写分析器程序』说明如何执行此操作。

另外，支持 Web 的应用程序可以在 WEB RECEIVE 命令上指定入局非 HTTP 请求的代码页转换。

请注意，不将非 HTTP 请求解析为请求行、头和主体元素。必须为整个请求执行已执行的任何代码页转换。

确定非 HTTP 请求处理

与 HTTP 请求相关的以下输入字段在非 HTTP 请求的分析器程序中未定义:

- HTTP 版本
- 方法
- 请求的路径部分
- 请求头

因此, 必须通过检查请求的内容来确定后续处理阶段。

分析器程序可以指定由转换器程序或由支持 Web 的应用程序处理的后续请求。第 93 页的『编写分析器程序』说明来自分析器程序的输入和输出, 以及它们如何用于确定请求处理。

用于非 HTTP 请求的应用程序编程

非 HTTP 请求的应用程序可以使用 EXEC CICS WEB 编程接口的特定元素。它们还可以是不支持 Web 的应用程序, 并产生由转换器程序编码的输出。

伪会话编程模型不适合非 HTTP 请求。应用程序应该设计为接收单个请求并提供单个响应。

支持 Web 的应用程序

如果要使用支持 Web 的应用程序以响应非 HTTP 请求, 则可以使用以下 CICS API 命令:

- WEB RECEIVE 命令可用于接收非 HTTP 请求。请注意, CICS 不为消息执行任何语法分析, 而且不自动组装为了通过网络传输而分割的请求。如果分析器程序组装请求, 则此结果对于 CICS WEB API 命令不可视。
- EXEC CICS DOCUMENT 命令可用于组成 CICS 文档, 以形成响应主体。
- WEB SEND 命令可用于将响应发送到非 HTTP 客户机。然而, 与 HTTP 特定操作相关的以下选项不合适:
 - STATUSCODE 和 STATUSTEXT。如果指定, 则忽略这些内容。
 - CLOSESTATUS。如果指定, 则忽略此内容。
 - CHUNKING。如果指定, 这会导致命令出错。
- WEB RETRIEVE 命令可用于检索早期 EXEC CICS WEB SEND 命令中发送的 CICS 文档。

其他 EXEC CICS WEB 命令仅与 HTTP 请求相关, 并且如果与非 HTTP 请求协同使用会导致 INVREQ 情况。

应用程序可以使用 WEB RECEIVE 命令指定非 HTTP 请求的代码页转换。

具有转换器程序的不支持 Web 的应用程序

通过不支持 Web 的应用程序, 您可以使用转换器程序将 Web 客户机的输入转换为适合应用程序的通信区域, 并将应用程序的输出转换为 HTML 以提供响应。如果为了通过网络传输而分割请求后, 分析器程序已重构该请求, 则此操作的结果会传递到转换器程序。

与 HTTP 请求相关的以下输入字段在非 HTTP 请求的转换器程序中未定义:

- HTTP 版本
- 方法
- URL 的路径部分
- 请求头

第 101 页的第 11 章,『转换器程序』具有有关编写转换器程序的更多信息。

第 15 章 CICS Web support 和 3270 显示应用程序

当 Web 客户机访问 3270 事务时，CICS 可以将输出显示为 HTML 表单。使用各种 Web 终端转换应用程序（DFHWBTTA、DFHWBTTB 或 DFHWBTTC）提供 Web 客户机对应用程序的访问，这些应用程序原先是为使用 3270 显示系统而设计的。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

可用两种方式之一从 3270 事务的输出创建 HTML 表单：

- 对于使用 BMS 的应用程序，将从 BMS 映射中生成 HTML 模板，并将其存储到模板库。您可以定制模板的生成。然而，如果您需要对生成的 HTML 进行的唯一更改可以包含在标题或页脚部分，则不需要从 BMS 映射生成模板，因为映射可以在执行时处理以生成 HTML 表单。
- 对于不使用 BMS 的应用程序，在执行生成 HTML 表单时处理 3270 出站数据流。

Web 终端转换应用程序可用于对 Web 浏览器显示 HTML 表单。

注： Web 终端转换应用程序在 HTTP/1.0 级别操作。它不完全使用 CICS Web support 中提供的工具（例如，EXEC CICS WEB API），因此不提供与 HTTP/1.1 规范的一致性。这意味着：

- 不根据 HTTP 协议规范检查来自 Web 客户机的请求和来自应用程序的响应。
- 通常或错误情形下，CICS 不提供 HTTP/1.1 响应，即使客户机为 HTTP/1.1 级别时也是如此。

所有三种不同的 Web 终端转换应用程序都支持非会话式、会话式和伪会话事务。

- DFHWBTTA 和 DFHWBTTB 执行 3270 数据流和 HTML 之间的转换，以及 BMS 映射和 HTML 生成的模板之间的转换。如果 HTML 模板为 32767 字节（32K）数据或更小，则使用 DFHWBTTA，如果 HTML 模板大于 32K，则使用 DFHWBTTB。（将 DFHWBTTB 用于较小的 HTML 模板会导致不必要的性能开销。）
- 当没有模板生成时，DFHWBTTC 执行 BMS 映射和 HTML 之间的转换。以此方法使用的 BMS 映射必须指定 TERM=3270，或者忽略 TERM 参数。DFHWBTTC 支持任何长度的 HTML 输出。如果您无须生成 HTML 模板，则使用 DFHWBTTC。

DFHWBTTB 和 DFHWBTTC 是 DFHWBTTA 的别名；在每种情况下调用同一程序（DFHWBTTA）。CICS 使用调用此程序的名称以确定需要哪个处理。

DFHWBTTA、DFHWBTTB 和 DFHWBTTC 生成符合 HTML 3.2 规范的 HTML。如果您使用不支持 HTML 3.2 的 Web 浏览器，一些功能可能不能正确工作。

您可以将指定 DFHWBTTA、DFHWBTTB 或 DFHWBTTC 的 URIMAP 定义创建为程序，将为了处理请求而调用该程序（PROGRAM 属性）。Web 客户机用于访问程序的方法是类似，但是使用 URIMAP 定义为您提供提供了可用于阻止或重定向请求的在线管理工具。当使用 URIMAP 定义时，分析器程序的使用是可选的。第 132 页的『3270 显示应用程序的 URL 路径部分』说明了使用 URIMAP 定义时如何正确指定 URL 路径。

CICS Web support 不对以下各项提供支持：分区、逻辑设备代码、磁槽式阅读器、外部格式化或其他硬件功能部件。您可以使用具有光笔支持的可检测字段。

3270 应用程序的 CICS Web support 处理

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

图 8 说明 CICS Web support 如何处理面向终端的事务。

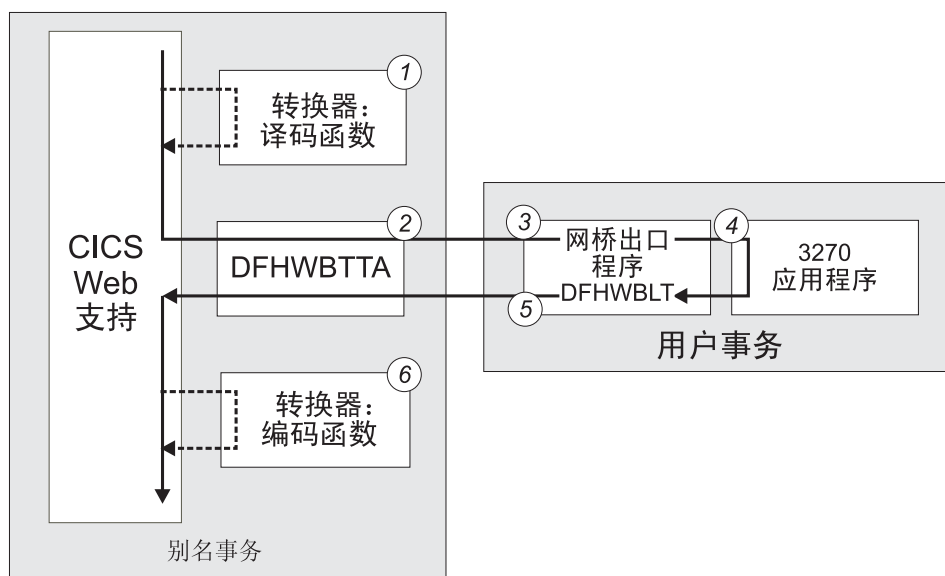


图 8. CICS Web support 如何与 3270 应用程序交互作用

图中显示的步骤如下：

1. 可选地，转换器程序构造传递到程序 DFHWBTTA 的输入。
2. DFHWBTTA 与用户事务连接，指定 DFHWBLT 作为网桥出口程序，并等待来自 DFHWBLT 的响应。用户事务在 3270 网桥环境中执行。
3. 网桥出口为用户应用程序设置 3270 环境。
4. 应用程序处理输入，并构造 3270 输出。
5. 网桥出口解释 3270 输出，并把 HTTP 响应传递到 DFHWBTTA。
6. 可选地，转换器程序修改传递到 Web 客户机的输出。

注： 当您把 CICS Web support 与 3270 应用程序协同使用时，应用程序在它自己的事务下执行，而不是在别名事务下执行。

要获取有关 3270 网桥的更多信息，请参阅 *CICS External Interfaces Guide*。

3270 显示应用程序的 URL 路径部分

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

为了从 Web 浏览器调用 CICS 3270 应用程序，您必须输入具有路径部分的 URL（该路径部分通过调用应用程序名 DFHWBTTA、DFHWBTTB 或 DFHWBTTTC 启动）以及合适的别名事务和转换器程序（如果需要）。请注意，此别名事务不应用到 3270 应用程序本身，它在它自己的事务下运行。

使用分析器程序

如果您正在使用分析器程序（如 CICS 提供的样本分析器 DFHWBADX）处理请求，则 URL 的路径部分包括应用程序（DFHWBTTA、DFHWBTTB 或 DFHWBTTC）的名称。它还包括您正在使用的任何转换器程序的名称，以及用于请求处理的别名事务（例如，缺省 CICS 提供的别名事务 CWBA）的名称。如第 99 页的『CICS 提供的样本分析器程序 DFHWBADX』中所说明，路径的这些元素由分析器程序抽取，并用于调用子序列处理阶段。

使用 URIMAP 定义

如果您正在使用 URIMAP 定义处理请求，则 URL 的路径部分在 PATH 属性中指定。使用 URIMAP 定义，URL 的路径部分不需要包含有关应用程序、转换器程序和别名事务的明确信息（虽然它仍可以包含这些内容）。使用 PROGRAM、CONVERTER 和 TRANSACTION 属性，所有这些元素都可以在 URIMAP 定义中指定。然后，这部分路径可以由您选择的任何路径所替换。为了满足 DFHWBTTA 本身的需求，请在您要在 URIMAP 定义中指定的路径末尾使用星号作为通配符。这将允许剩余路径部分更改为控制 DFHWBTTA。

使用 URIMAP 定义和分析器程序

通过在 URIMAP 定义中指定 ANALYZER(YES) 选项，您可以在请求的处理路径中使用分析器程序。分析器程序可以动态修改转换器程序、URIMAP 定义指定的别名事务标识和程序名，而 DFHWBTTA 可以看到这些更改。

提供调用应用程序所需的信息后，URL 的下一部分路径用于对 DFHWBTTA 提供控制信息。此信息包括：

- 指定是否应该使用未格式化方式的关键字。
- 您要使用的 3270 应用程序的事务标识。
- 指定事务的输入参数，它使用加号 (+) 作为定界符。

图 9 显示 DFHWBTTA 解释的路径部分语法。

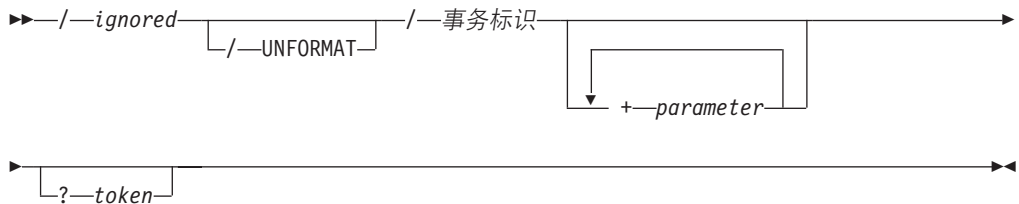


图 9. DFHWBTTA 解释的路径语法

DFHWBTTA 将 URL 的路径部分解释如下：

ignored

DFHWBTTA 将忽略路径的第一部分。这是由分析器解释或与 URIMAP 定义匹配的部分，用于提供调用应用程序所需的信息。

UNFORMAT

3270 显示可以在两种方式下操作，格式化方式和未格式化方式。如果出现这个关键字，DFHWBTTA 模拟以未格式化方式操作的 3270 显示。如果这个关键字被省略，DFHWBTTA 模拟以格式化方式操作的 3270 显示。

要获得有关 3270 显示如何以未格式化方式操作的更多信息，请参阅 *CICS Application Programming Guide*。

事务标识

在初始请求中，此信息指定要运行的 CICS 事务。在连续请求中，路径的这个元素将被忽略。

parameter

指定事务的输入参数。使用加号 (+) (而不是空格) 作为定界符以分隔事务标识和此数据，并分隔此数据的元素。

token

DFHWBTTA 将忽略它。它可以由分析器程序使用。

URL 必须总是以此格式编码。

例如，如果您正在使用 CICS 提供的分析器程序 DFHWBADX，则您可以使用以下 URL 路径发出 CEMT INQ TAS 命令：

```
/cics/cwba/dfhwbttta/CEMT+INQ+TAS
```

在此示例中：

- cics 用于表明不需要转换器程序。
- cwba 是用于请求处理的别名事务的名称。
- dfhwbttta 是应用程序名。
- CEMT+INQ+TAS 告诉 DFHWBTTA 访问 CEMT 事务并发出 INQ TAS 命令。

另外，您可以设置包含以下属性的 URIMAP 定义：

```
Path:          /terminal/*
Transaction:   CWBA
Program:       DFHWBTTA
```

启用此 URIMAP 定义后，您可以使用以下 URL 路径发出 CEMT INQ TAS 命令：

```
/terminal/CEMT+INQ+TAS
```

在此示例中：

- terminal 与 URIMAP 定义匹配，它指定别名事务和应用程序的名称。
- CEMT+INQ+TAS 被 URIMAP 定义忽略，但是它告诉 DFHWBTTA 访问 CEMT 事务并发出 INQ TAS 命令。

初始和连续请求

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

DFHWBTTA 能通过 HTTP 请求在事务中的上下文分辨出它的两种类型：

初始请求

初始请求启动 CICS 事务。用以下某种方法发送初始请求：

- 明确输入 URL。
- 选择 HTML 页面中的链接。
- 选择 HTML 表单中的按钮。在表单中输入的任何数据将被忽略。

连续请求

连续请求继续现有的 CICS 事务。用以下方式发送连续请求：

- 选择作为先前请求的响应而显示的 HTML 表单中的按钮。

连续请求使用 HTML POST 方法；表单数据在 HTML 请求的实体主体中传输。

在会话式或伪会话事务中，Web 客户机和 CICS 之间存在多个交互，其中有一个初始请求，后跟一个或多个连续请求。在更简单的事务中，仅有一个交互，其中有一个初始请求，无连续请求。

由初始请求显示并由后续请求返回的 HTML 表单中的隐藏元素 (DFH_STATE_TOKEN) 用于辨别初始请求和连续请求，并用于将连续请求与正确的事务关联起来。

连续请求的事务标识

在连续请求中，URL 是以先前请求显示的格式进行编码的。然而，在连续请求中将忽略在 URL 中编码的事务标识。反之，可以用下列方式确定事务：

- 当连续请求是对话式事务的一部分时，相同的事务继续执行。
- 当连续请求是伪对话式事务的一部分时，那么：
 - 如果先前事务是由具有 TRANSID 选项的 EXEC CICS RETURN 命令终止的，则指定的事务标识将被使用。
 - 如果先前的事务没有在它的 EXEC CICS RETURN 命令中指定事务标识，但是 AID 与事务标识关联，则使用该事务标识。
 - 如果没有在 EXEC CICS RETURN 命令中指定事务标识，并且没有与 AID 关联的事务标识，则 CICS 从 HTML 表单获取事务标识。

HTML 表单中的事务标识

从 3270 显示附加一个事务时，CICS 希望在 3270 数据流的第一个已修改字段中找到事务标识。Web 客户机发送表单数据的顺序并不总是可预料的，因此 CICS 使用表单字段名称和 3270 屏幕上相应位置之间的映射：

- 对于不使用 BMS 映射的事务，映射直接使用字段名，因为该名称反映了字段在 3270 屏幕的位置。
- 对于使用 BMS 映射的事务，字段名并不总是反映字段在 3270 屏幕的位置，所以使用间接映射。映射使用隐藏变量 DFH_NEXTTRANSID.*n*。当从 BMS 映射中创建一个 HTML 模板时，最多将创建五个变量。每个变量的值是一个输入字段的名称，按 3270 缓冲区位置的顺序排列。

当 CICS 接收 HTTP 请求时，它依次检查每个 DFH_NEXTTRANSID 字段，以确定它引用的输入字段名称，以及 HTTP 请求是否包含该字段的值。如果它包含，那是因为它最终用户修改了它，并因此假设其中包含下一个事务的事务标识。

当通过合并几个 BMS 以及非 BMS SEND 命令的输出来构造屏幕时，会出现输入字段被抑制的情况（要获得更多信息，请参阅第 151 页的『如何选择页脚部分』）。因此 CICS 可以正确地识别 3270 数据流中的事务标识，您必须确保合并的 HTML 页面中没有禁止可能包含事务标识的输入字段。它引用的输入字段名称，以及 HTTP 请求是否包含该字段的值。

3270 应用程序的 CICS Web support 中的终端控制命令

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

3270 应用程序的 CICS Web support 支持以下终端控制命令：

- SEND
- CONVERSE
- RECEIVE

它还支持最小函数 BMS 和 SEND TEXT 命令。

忽略 SEND 和 CONVERSE 命令中的 DEFRESP 选项。这可能影响应用程序恢复。

从 BMS 映射生成的 HTML 模板

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

3270 显示系统的功能与 HTML 表单有许多相似之处：

- 在两种情况下，显示区域都可以包含固定文本和用户能够输入数据的区域。
- 3270 键盘上的 AID 键与 HTML 表单上显示的按钮具有相似的功能。
- 在两种情况下，都可以检测最终用户是否修改了数据输入字段的内容。

从 BMS 映射生成的模板包含许多表示 3270 显示功能的元素：

- 映射中的保护字段作为普通 HTML 文本显示。
- 映射中的未保护字段作为文本输入元素显示。CICS 为每个元素提供一个两部分的名称，该名称最多可以长达 32 个字符：
 - 名称的第一部分长度为 11 个字符，并且具有以下格式：

```
Frrccllll_
```

其中

- *rr* 是一个二位数字，表示在 3270 屏幕上显示的字段所在的行。
- *ccc* 是一个三位数字，表示在 3270 屏幕上显示的字段所在的列。
- *llll* 是一个四位数字，表示字段的长度。
- 对于在映射中命名的 BMS 字段，第二部分包含映射中使用的名称，如果必要，可以将它截断到 21 个字符的长度。
- 对于未命名的 BMS 字段，第二部分格式为 DFH_ *nnnn*，其中 *nnnn* 是一个 4 位数。按照字段在 BMS 映射中出现的顺序为它们依次编号。

例如，假设第三个未命名的未保护字段被定位到屏幕的第 2 行、第 11 列，并且长度为 16 个字符。生成的两部分的名称为

```
F020110016_DFH0003
```

现在假设在 BMS 映射中有名称为 TOTAL_MONTHLY_PURCHASES 的相同字段。CICS 为 HTML 元素生成的名称是：

```
F020110016_TOTAL_MONTHLY_PURCHAS
```

注：字段在 3270 屏幕上显示的顺序和它们在 BMS 映射定义中编码的顺序可能不同。当相应的模板在 Web 客户机上显示时，字段是按编码顺序显示的。

- 3270 显示支持的每个辅助操作请求键都是模拟提交按钮的。按钮命名为：

- DFH_PF01 到 DFH_PF24
- DFH_PA1 到 DFH_PA3
- DFH_ENTER, DFH_CLEAR

当最终用户选择其中一个按钮时，相应的变量在 HTTP 请求中被发送。CICS 使用这个变量来确定在 3270 应用程序中模拟哪个 AID。

一个名为 DFH_PEN 的附加提交按钮与可检测字段一起使用。

- 可检测字段是用来模拟有前导复选框的文本元素的。要获得更多信息，请参阅第 143 页的『使用可检测字段』。
- 隐藏元素 (DFH_STATE_TOKEN) 用来维护应用程序在许多与 Web 客户机的交互上所看到的显示状态。
- 隐藏元素 (DFH_CURSOR) 和 JavaScript™ 函数 (dfhinqcursor()) 通过协作来向应用程序返回光标位置。
- 一系列隐藏元素 (从 DFH_NEXTTRANSID.1 到 DFH_NEXTTRANSID.n) 用于捕捉输入到 Web 客户机字段的事务标识符。

从 3270 数据流生成的 HTML 页面

将您的简短描述放在这里；用于第一段和摘要。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

对于不使用 BMS 的应用程序，CICS Web support 生成由三个部分组成的 HTML 页面：

- 标题部分
- 屏幕图像部分
- 页脚部分

标题部分

CICS Web support 生成以下标题部分：

```
<!doctype html public "-//W3C//DTD HTML 3.2//EN">
<html>
<STYLE TYPE="text/css">
<!--
    TABLE, TR, TD
    { padding: 0mm    }
    TABLE
    { width: 60%     }
-->
.BRIGHT
{font-weight: bold}
{font-family: courier}
.INPUT
{font-family: courier}
</STYLE>
<head>
<title>CICS Web support screen emulation - tranid</title>
<meta name="generator" content="CICS Transaction Server/2.2.0">
```

```

<script language="JavaScript">
<!--
  function dfhsetcursor(n)
    {for (var i=0;i<document.form3270.elements.length;i++)
      {if (document.form3270.elements[i].name == n)
        {document.form3270.elements[i].focus();
         document.form3270.DFH_CURSOR.value=n;
         break}}}
  function dfhinqcursor(n)
    {document.form3270.DFH_CURSOR.value=n}
  // -->
</script>
</head>
<body> onLoad="dfhsetcursor('&DFH_CURSORPOSN;')"

```

您可以通过提供自己的标题部分来修页面外观。要获得更多信息，请参阅第 140 页的『修改 DFHWBTTA 的输出』。

屏幕图像部分

HTML 页面的这部分是从 3270 屏幕图像的内部表示法直接生成的，其大小由与您的事务关联的 FACILITYLIKE 终端定义上的 DEFSCREEN 和 ALTSCREEN 定义决定。它包含以下元素：

普通 HTML 文本

模拟保护字段

文本输入元素

模拟未保护字段。为每个元素提供一个 11 个字符长的名称，并具有以下格式：

Frrccllll_

其中

- *rr* 是一个二位数字，表示在 3270 屏幕上显示的字段所在的行。
- *ccc* 是一个三位数字，表示在 3270 屏幕上显示的字段所在的列。
- *llll* 是一个四位数字，表示字段的长度。

例如：

- 一个在 3270 显示的第 1 行、第 1 列，并且长度为 78 字节的字段将被命名为

F010010078_

具有复选框的文本元素

模拟可检测字段。请参阅第 143 页的『使用可检测字段』以获取更多信息。

隐藏元素

名为 DFH_STATE_TOKEN 的隐藏元素用来维护应用程序在许多与 Web 客户机的交互上所看到的显示状态。

隐藏元素 (DFH_CURSOR) 和 JavaScript 函数 (dfhinqcursor()) 通过协作来向应用程序返回光标位置。

```

<!doctype html public "-//W3C//DTD HTML 3.2//EN">
<html>
<head>
<title>CICS Web support screen emulation - tranid</title>
<meta name="generator" content="CICS Transaction Server/2.1.0">
<script language="JavaScript">
<!--
  function dfhsetcursor(n)

```



```

        {for (var i=0;i<document.form3270.elements.length;i++)
            {if (document.form3270.elements[i].name == n)
                {document.form3270.elements[i].focus();
                 document.form3270.DFH_CURSOR.value=n;
                 break}}}
    function dfhinqcursor(n)
        {document.form3270.DFH_CURSOR.value=n}
    // -->
</script>
</head>
<body>
onLoad="dfhsetcursor('&DFH_CURSORPOSN;')"

```

从 3270 屏幕图像中生成的 HTML 和从用于 BMS 映射的模板中生成的 HTML 很相似。通过使用 HTML 表来实现页面上信息的水平和垂直对齐:

- HTML 表对于 3270 屏幕中包含字段开始的每个不同的列都包含一列。例如, 如果 3270 屏幕包含的字段开始于第 2、11、21 和 55 列, 则 HTML 表将包含四列。因此, 所有在 3270 屏幕上其起始位置垂直对齐的字段, 将在 HTML 页面上垂直对齐。
- HTML 表对于 3270 屏幕上包含字段开始的每一行都包含一行。因此, 所有起始位置在 3270 屏幕上水平对齐的字段将在 HTML 页面上水平对齐。3270 屏幕上不包含字段的行在 HTML 表上将不表示出来。
- 在表中, 文本以比例字体显示。

考虑包含以下字段的 3270 屏幕:

字段	行	起始列
Field_1	2	2
Field_2	3	2
Field_3	3	35
Field_4	4	2
Field_5	4	35
Field_6	9	2
Field_7	9	18
Field_8	9	35

所有字段都从 3270 屏幕的第 2、18 或 35 列开始。所以结果 HTML 表将有三列。同样, 所有字段都位于 3270 屏幕的第 2、3、4 或 9 行, 所以 HTML 表将有四行。

您可以使用转换器程序的编码函数来修改屏幕图像部分。要获得更多信息, 请参阅第 142 页的『将转换器程序与 DFHWBTTA 协同使用』。

页脚部分

CICS Web support 生成以下页脚部分。3270 显示支持的每个辅助操作请求键都是模拟提交按钮的。当最终用户选择其中一个按钮时, 相应的变量在 HTTP 请求中被发送。CICS 使用这个变量来确定在 3270 应用程序中模拟哪个 AID。一个名为 DFH_PEN 的附加提交按钮与可检测字段一起使用。

```

<input type="submit" name="DFH_PF1" value="PF1">
<input type="submit" name="DFH_PF2" value="PF2">
<input type="submit" name="DFH_PF3" value="PF3">

```

```

<input type="submit" name="DFH_PF4" value="PF4">
<input type="submit" name="DFH_PF5" value="PF5">
<input type="submit" name="DFH_PF6" value="PF6">
<input type="submit" name="DFH_PF7" value="PF7">
<input type="submit" name="DFH_PF8" value="PF8">
<input type="submit" name="DFH_PF9" value="PF9">
<input type="submit" name="DFH_PF10" value="PF10">
<input type="submit" name="DFH_PF11" value="PF11">
<input type="submit" name="DFH_PF12" value="PF12">
<br>
<input type="submit" name="DFH_PF13" value="PF13">
<input type="submit" name="DFH_PF14" value="PF14">
<input type="submit" name="DFH_PF15" value="PF15">
<input type="submit" name="DFH_PF16" value="PF16">
<input type="submit" name="DFH_PF17" value="PF17">
<input type="submit" name="DFH_PF18" value="PF18">
<input type="submit" name="DFH_PF19" value="PF19">
<input type="submit" name="DFH_PF20" value="PF20">
<input type="submit" name="DFH_PF21" value="PF21">
<input type="submit" name="DFH_PF22" value="PF22">
<input type="submit" name="DFH_PF23" value="PF23">
<input type="submit" name="DFH_PF24" value="PF24">
<br>
<input type="submit" name="DFH_PA1" value="PA1">
<input type="submit" name="DFH_PA2" value="PA2">
<input type="submit" name="DFH_PA3" value="PA3">
<input type="submit" name="DFH_CLEAR" value="Clear">
<input type="submit" name="DFH_ENTER" value="Enter">
<input type="submit" name="DFH_PEN" value="Pen">
<input type="reset" value="Reset">
</form>
</body>
</html>

```

您可以通过提供自己的页脚部分来修页面外观。要获得更多信息，请参阅『修改 DFHWBTTA 的输出』。

修改 DFHWBTTA 的输出

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

对于使用 BMS 的应用程序，您可以定制 BMS 映射所创建的 HTML 模板。要获得更多信息，请参阅第 147 页的『生成定制的 HTML 模板』。

对于非 BMS 应用程序和使用 DFHWBTTA 调用的 BMS 应用程序，可以通过提供您自己的标题和页脚部分来修改页面的外观。您不能直接更改屏幕图像部分，但是您在标题部分插入的标记将影响以后各部分的外观。

要提供您自己的标题和页脚部分，定义和安装下列模板中的一个或多个，其名称在 DOCTEMPLATE 定义的 TEMPLATENAME 字段中定义：

tranHEAD

这是在事务 *tran* 输出的 HTML 页面标题中插入的模板（如果安装）。

CICSHEAD

这是要在作为没有安装相应 *tranHEAD* 模板的事务输出的 HTML 页面标题处插入的模板。

tranFOOT

这是在作为事务 *tran* 输出的 HTML 页面的页脚处插入的模板（如果安装）。如果没有安装这个模板，将使用 CICSFOOT。

CICSFOOT

这是要在作为没有安装相应 *tranFOOT* 模板的事务输出的 HTML 页面页脚处插入的模板。

要获取有关创建文档模板的更多信息，请参阅 *CICS Application Programming Guide*。

由 CICS Web support 生成的标题部分（包括 DFHWBTTTC）使用 EBCDIC Latin 字符集（代码页 037）。如果在 CICS 系统中使用不同的代码页，则必须使用您自己的代码页创建类似的标题部分：

1. 创建包含标题部分，称为 CICSHEAD 的文档模板。
2. 定义和安装模板的 DOCTEMPLATE 定义。

. 用于 CICS 生成的标题部分中的以下字符在不同代码页中有不同的表示法：

! [] { }

提供您自己的标题模板

如果您提供自己的标题模板，那么还必须提供一些 HTML 页面需要的元素。标题模板应该包含以下 HTML 元素：

- doctype 标记，例如：

```
<!doctype html public "-//W3C//DTD HTML 3.2//EN>
```

- <html> 标记

- <head> 标记

- <STYLE> 标记，它必须包含 BRIGHT 和 INPUT 类的样式表规则。例如：

```
<STYLE TYPE="text/css">
<!--
    TABLE, TR, TD
    { padding: 0mm    }
    TABLE
    { width: 60%     }
-->
.BRIGHT
{font-weight: bold}
{font-family: courier}
.INPUT
{font-family: courier}
</STYLE>
```

您可使用 TABLE 元素的 width 属性微调屏幕图像部分的外观。

- </head> 标记

- <body> 标记。您可以用这个标记指定文本颜色，或者指定用于页面背景的图像。例如：

```
<body background="/dfhwbimg/background2.gif" bgcolor=#FFFFFF"
  text=#000000" link=#00FFFF" vlink=#800080" alink=#FF0000"
  onLoad="dfhsetcursor('&DFH_CURSORPOSN;')"
```

注：此示例使用第 144 页的『使用 DFHWBIMG 显示图形』中描述的 DFHWBIMG。

- 定制页面所需的任何其他 HTML 元素，可选。

提供您自己的页脚模板

如果您提供自己的页脚模板，那么必须提供一些 HTML 页面需要的元素。页脚模板应该包含以下 HTML 元素：

- 代表任何已编程功能键或 ENTER 键的输入按钮。例如：

```
<input type="submit" name="DFH_PF1" value="Help">
<input type="submit" name="DFH_PF3" value="Quit">
<input type="submit" name="DFH_ENTER" value="Continue">
```

HTML 表单的这些表单部分由 CICS 开始。用户选择该按钮时，产生第 137 页的『从 3270 数据流生成的 HTML 页面』中所讨论的 AID 指示符，所以应该具有在此描述的名称。*value* 参数指定出现在生成的按钮上的图注。DFHWTBTA 不使用它。

- `</form>` 标记
- 定制页面所需的任何其他 HTML 元素，可选。
- 用于关闭页面的 `</body>` 标记
- `</html>` 标记

将转换器程序与 DFHWTBTA 协同使用

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

您可以使用转换器程序的解码函数修改传递到 DFHWTBTA 的请求。

- 当使用一个表示辅助操作请求键的按钮将 HTML 表单提交给客户机时，请求包含一个用来指示哪个按钮被选中的字段。可以通过修改请求来模拟另一个辅助操作请求键的影响。将字段值改为所需的辅助操作请求键，或者在通过 Web 客户机发送的字段后面插入一个新字段。
- 将 HTML 表单提交给客户机时，DFH_CURSOR 字段将包含含有光标的字段名称。可以通过修改请求来模拟另一个光标位置的影响。更改 DFH_CURSOR 字段的值以包含另一个字段名，或者在通过 Web 客户机发送的字段后面插入一个新的 DFH_CURSOR 字段。
- 可以在连续请求中更改 DFH_NEXTTRANSID.*n* 变量以选择下一个事务标识。可以插入或删除一个变量，或修改其值。有关如何使用这些字段来确定下一个事务标识的详细信息，请参阅第 135 页的『HTML 表单中的事务标识』。

不要修改 DFH_STATE_TOKEN 的值。

您可以使用转换器程序的编码函数修改 DFHWTBTA 的输出：

- 响应存在于一个缓冲区中，该缓冲区以一个用于指定缓冲区长度的 32 位无符号数开始。缓冲区的其余部分是 HTTP 响应。响应中的 HTML 对应于来自事务程序的输出 BMS 映像或 3270 数据流。
- HTTP 响应中的 HTTP 头是由 DFHWTBTA 自动生成的。由 DFHWTBTA 生成的头是：
 - Content-type: text/html
 - Content-length: <length of the entity body>
 - Pragma: no-cache
 - Connection: Keep-Alive (如果是 HTTP 1.0 持续连接的话)

如果需要任何附加的头，应该使用转换器的编码函数来将它们添加到 HTTP 响应中。

启用可检测字段

要启用 CICS Web support 3270 网桥上的可检测字段处理，必须定义允许光笔支持的网桥工具。

要这样做，按照这些步骤操作：

1. 将下列定义复制到新组。除非运行在 CICS 系统上的所有应用程序需要光笔支持，否则还要重命名这两个定义。
 - 组 DFHTERM 中，由 CICS 提供的网桥设施 CBRF。
 - 组 DFHTYPE 中，它的缺省值 TYPETERM: DFHLU2。
2. 在 TYPETERM 定义中，将“DEVICE PROPERTIES”下的 LIGHTPEN 选项更改为 YES。
3. 在 TERMINAL 定义中，将 TYPETERM 参数更改成指向新的 TYPETERM。
4. 将定义安装在 CICS 区域中。
5. 如果您已创建新的网桥工具定义，则更新将与 CICS Web support 一起运行的 3270 事务的 PROFILE 定义，以便在新的 TERMINAL/TYPETERM 定义上建模网桥工具：
 - a. 通过使用 CEDA 来标识事务使用的 PROFILE，以查看 TRANSACTION 定义的 PROFILE 参数。
 - b. 如果概要是由 CICS 提供的，则将它复制到您自己的组中并重命名。
 - c. 改变新的 PROFILE 并在 FACILITYLIKE 参数中输入新网桥设施的名称。
 - d. 改变 TRANSACTION 定义以使用新的 PROFILE 定义。

使用可检测字段

当 CICS 从 3270 数据流生成 HTML 页面时，它用复选框后面的文本输入字段模拟可检测字段。

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

要使用可检测字段，必须配置与事务关联的网桥工具。『启用可检测字段』将告诉您如何执行此操作。

可检测字段存在于：

- 将字段标识为可检测的或增强的字段属性字节。
- 并且 3270 字段的第一个字符包含有效的指示字符。这可以是“与”符号（&）、右尖括号（>）、问号（?）、空白或空。

要获取有关可检测字段的更多信息，请参阅 *CICS Application Programming Guide*。

当复选框和文本输入字段显示在 Web 客户机上时：

- 3270 字段中的指示字符将不被显示。因此，在 Web 客户机中的字段长度比在 3270 数据流中的字段长度少一个字符。
- 如果指示字符是右尖括号（>），则复选框包含选中符号（?）。否则，复选框为空。

要在 Web 客户机上使用可检测字段：

- 选中复选框以模拟设置 3270 数据流的修改数据标记（MDT）位。清除该框将关闭修改数据标记。在 HTML 页面的文本字段中输入数据不会更改修改数据标记。

- 要将数据传输到 CICS 应用程序，则选取复选框，并选择名为 DFH_PEN 的按钮。
 - 如果只选取了一个注意字段，则 CICS 应用程序只接收该字段的内容。EIBAID 字段被设置为 DFHPEN。
 - 如果选取了多个注意字段，则 CICS 应用程序接收最接近 3270 屏幕第 1 行和第 1 列的字段内容。EIBAID 字段被设置为 DFHPEN。
 - 如果未选取注意字段，则 CICS 接收所有字段的内容。EIBAID 字段被设置为 DFHENTER。

使用 DFHWBIMG 显示图形

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

CICS 提供了您可以在自己的 Web 应用程序中使用的许多图形。它们是：

CICS.GIF

CICS 徽标

MASTHEAD.GIF

具有文本“CICS Web Interface”的 CICS 徽标

BACKGROUND1.GIF

包含“CICS”字符的背景

BACKGROUND2.GIF

包含“CWI”字符的背景

TEXTURE1.JPEG

纹理背景

TEXTURE2.JPEG

纹理背景

TEXTURE3.JPEG

纹理背景

TEXTURE4.JPEG

纹理背景

TEXTURE5.JPEG

纹理背景

TEXTURE6.JPEG

纹理背景

要在 Web 浏览器中显示图形，输入的 URL（在转换到大写后）的路径应该是以下格式：

```
/DFHWBIMG/filename
```

其中 **filename** 列出的一个图形的名称。例如：

```
/dfhwibmg/Texture1.jpeg
```

要在您的输出中合并任何图形，将路径包含在相应的 HTML 标记中。例如，您可以用下面的标记包含一幅纹理背景：

```
<body background="/dfhwibmg/background1.gif" ... >
```

CICS 将路径以“/DFHQBIMG”开头的 HTTP 请求作为特殊情况处理；不调用分析器，而 DFHQBIMG 作为转换器程序运行。

CICS 在用于 CICS 提供的事务的模板中使用这些图形的一部分。

CICS 提供的图形作为 DFHQBIMG 的一部分硬编码，且不可以作为独立的文件提供；DFHQBIMG 不支持图形显示（除了指定的那些以外）。

第 16 章 从 BMS 定义创建 HTML 模板

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

本主题描述如何从 BMS 映射集定义中创建 HTML 模板。

如果希望从您不具有其源代码的现有 BMS 映射集创建 HTML 模板，可以从相应的装入模块中重新构造源代码。使用 BMS 宏生成实用程序 (DFHBMSUP)，其在 *CICS Operations and Utilities Guide*。

CICS 为安装从 BMS 映射集中创建的 HTML 模板提供了编目的过程 DFHMAPT。请参阅 *CICS Application Programming Guide* 以获取详细信息。

有关 BMS 生成的模板

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

从 BMS 映射生成的模板包含:

- 映射中的常量和输入字段
- 表示以下内容的按钮:
 - Enter 键
 - PA1、PA2 和 PA3 键
 - 程序功能键 PF1 到 PF24
 - HTML 复位
- 最多 5 个隐藏的变量，DFH_NEXTTRANSID.1 到 DFH_NEXTTRANSID.5，它们的值是映射中前 5 个字段的名称。这些变量的使用在第 131 页的第 15 章，『CICS Web support 和 3270 显示应用程序』中说明。
- 隐藏的变量 DFH_CURSOR，它的值是在映射中设置了光标的字段的名称。如果光标位于一个未命名的字段上，则 DFH_CURSOR 为零。
- JavaScript 函数 dfhsetcursor()。当 DFH_CURSOR 包含字段名时，函数将光标位置设置到此字段。
- 用于 onFocus 异常的 JavaScript 异常处理程序。这个函数调用 dfhsetcursor，并跟踪光标的移动。

生成定制的 HTML 模板

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

可以使用以下三种方法定制从 BMS 映射生成的 HTML 模板:

- 可以通过对自己的 DFHMSX 宏版本进行编码来修改从 BMS 映射生成 HTML 模板的方式。
- 可以通过在 BMS 映射定义中使用 DFHWBOUT 宏向生成的映射添加 HTML 文本
- 可以手工编辑生成的 HTML。在以下情况下这会有用:
 - 您要覆盖当程序发出 MAP SEND 命令时发生的属性动态更改。
 - 您要在 Web 3270 环境之外使用 HTML 模板。

在这两种情况中，您都将需要更改 `Frrcccllll` 变量，而这些变量是通过模板生成过程来添加的。

有关定制的模板示例，请参阅第 158 页的『定制示例』。

CICS 为以下 CICS 提供的事务（使用 BMS）提供 HTML 模板：

`CETR`

提供的模板使用 EBCDIC Latin 字符集（代码页 037）。如果您在 CICS 系统中使用不同的代码页，则必须生成这些模板的您自己的版本：CICS 生成的头部分中使用的以下字符在不同于 037 的代码页中有不同的表示法：

! [] { }

使用 `DFHMDX` 宏上的 `CODEPAGE` 参数指定代码页。

使用 `DFHMSX` 宏来定制

可以通过对自己的 `DFHMSX` 宏版本进行编码来修改从 BMS 映射生成 HTML 模板的方式。您可指定：

- 由按钮表示的 3270 键
- 显示在每个按钮上的文本或图像
- HTML 页面的标题
- 要在 HTML 页面顶部显示的报头图形
- 作为图形文件或颜色的页面背景
- 常规文本、未访问的链接、已访问的链接和活动链接的颜色
- 页面是否应该包含 HTML 复位按钮，以及显示在上面的文本
- BMS 映射中使用的颜色和用于 HTML 模板中相应文本的颜色之间的映射
- HTML 页面中应该抑制哪些 BMS 字段
- JavaScript `onLoad()` 和 `onUnload()` 异常处理程序
- 模板中的文本是以比例字体显示还是以非比例字体显示
- 生成模板时要使用的代码页，以及用于特殊字符 `[]{}` 和 `!` 的代码点
- 受保护字段应该在 HTML 页面中右对齐

注：

1. BMS 字段的 `ATTRB=BRT` 选项对于未命名、未受保护的（输入）字段不起作用。
2. 如果逻辑映射中命名的字段被清空（例如，使用 `DEL` 键）或该字段在以前的 `SEND` 命令上已经为空（空或空格），并关闭了字段的“修改数据标记（MDT）”，则不设置 `DFHBMEOF`（字段的属性字节的 3270 属性位）。

对自己的 `DFHMSX` 宏版本进行编码时，可以指定将编码的选项应用于：

- 所有映射集中的所有映射
- 特定映射集中的所有映射
- 个别映射

安装 HTML 模板

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

过程如下:

1. 复查 CICS 应用程序及其对 BMS 的使用, 了解定制是否必要。
2. 对于需要定制的 HTML 页面的应用程序, 创建一个定制宏定义, 并将它存储在与汇编程序 SYSLIB DD 语句中指定的宏库并置的库中。编写适当的 DFHWBOUT 宏调用, 然后将它们放在映射定义中适当的地方。
3. 在 DFHMSD 宏上使用 TYPE=TEMPLATE, 或者在传递给汇编程序的参数中使用 SYSPARM=TEMPLATE 汇编现有的映射定义。请注意, DFHMSD 宏上的标签用于命名为处理的映射集中每个映射产生的 HTML 模板。HTML 模板名由 DFHMSD 宏的标签组成, 后跟用字符 A-Z 和 0-9 生成的一到两个字符的后缀。当映射集中有多于 36 个映射时使用两个字符的后缀, 并且在这种情况下, 映射集名必须是六个字符或更少。在程序发出 BMS SEND 或 RECEIVE 时, 为了使网桥出口与带 BMS 映射的 HTML 模板相匹配, HTML 模板成员必须与在 SEND 和 RECEIVE 语句上使用的映射集值的名称匹配。如果使用的是定制宏, 必须向 TYPE 添加定制宏的名称。汇编程序产生 IEBUPDTE 源代码语句, 为映射集中的每个映射建立一个模板。
4. 使用 IEBUPDTE 将模板存储在模板库中。如果模板库的记录格式不是固定的块格式, 需要将它们存储在另一个分区的数据集中, 然后使用诸如 ISPF COPY 这样的命令将它们转换成模板库的记录格式。
5. 如果您希望将模板放在不是 DFHHTML DDname 中指定的一个已分区的数据集中, 则必须为您的模板定义 DOCTEMPLATE 定义, 并指定备用 DDname。备用 DDname 还必须在 CICS JCL 中指定。

要将包含模板的分区数据集分配给特定的 DDname 以从中安装模板, 可以使用 ADYN 样本事务。首先安装包含 ADYN 及其相关程序的 DFH\$UTIL 组, 然后运行 ADYN:

```
ADYN  
ALLOC DDNAME(ddname) DATASET('template-pds') STATUS(SHR)
```

其中 *ddname* 是在 DOCTEMPLATE 定义中指定的 DDname, *template-pds* 是包含要安装的模板的分区数据集的名称。要获取有关安装和使用 ADYN 的更多信息, 请参阅 *CICS Customization Guide*。

HTML 模板的大小限制

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

不存在对使用 3270 网桥运行的事务所使用模板大小的限制。然而, 存储超过 32K 的模板从小模板处理很困难, 所以要处理较大的模板, 您必需指定映射到 HTTP 请求中 DFHWBTTB 的程序名的路径。要获得更多信息, 请参阅第 132 页的『3270 显示应用程序的 URL 路径部分』。

注意, 如果符号替换显著增加了数据量, 则所需存储小于 32K 的模板可将存储扩展到大于 32K。

生成模板时，DFHWBTLG 发出的消息包含要从 DFHHTML 数据集读取的每个模板所需的存储量。使用这些消息确定您是否应该使用 DFHWBTTA 或 DFHWBTTB 的程序名。

编写定制宏定义

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

必须提供由 CICS 提供的汇编程序宏调用的完整汇编程序宏定义。必须根据汇编程序宏定义的规则编写定制宏的定义。定义中的宏调用也必须遵循汇编程序语言宏语句的规则。定制宏定义包含以下元素：

1. 一个开始定义的 MACRO 语句。
2. 宏的名称。
3. 任意数量的 DFHMDX 宏的调用。

在第 153 页的『DFHMDX 宏』中描述了 DFHMDX 的语法，在第 158 页的『定制示例』中描述了它的使用。DFHMDX 是从 DFHMSX 中调用的。

4. 一个结束定义的 MEND 语句。

处理空格

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

定制宏定义时，必须考虑对于空白的 HTML 规范。对于 3270 终端，空格（EBCDIC X'40'）和空（EBCDIC X'00'）可用于格式化屏幕数据位置。将这样的数据流转换成 HTML 时，客户机对它的解释所生成的输出与在 3270 终端上找到的不一样。

如果空格字符串后紧跟着一个开始标记，客户机就忽略它，并将所有后续连续空格序列解释成一个空格。要强制呈现所有空格，您可使用 `<pre>` 和 `</pre>` 标记。

没有指定空字符的处理，客户机处理它们的方式也不一致。它们可能显示，也可能不显示。

组合 BMS 和非 BMS 输出

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

事务可以发出一系列 BMS 和非 BMS 命令以构建 3270 显示屏幕的内容。本主题解释了如何组合所有命令的输出以构造显示在 Web 浏览器上的 HTML 页面。这些步骤为：

1. 当发出 BMS 或非 BMS SEND 命令时，生成一个 HTML 页面（包含一个标题部分、一个屏幕图像部分和一个页脚部分），但不发送给 Web 客户机。
2. 当事务发出 RECEIVE 命令或终止时：
 - 从以前生成的标题部分中选择一个。
 - 通过将所有以前生成的屏幕图像部分合并起来，创建一个新的屏幕图像部分。
 - 从以前生成的页脚部分中选择一个。
3. 结果 HTML 页面发送给 Web 客户机。

如何选择标题部分

注：在此描述中：

- 从 BMS 映射中生成的 HTML 页面的起始位置是映射左上角的行和列。
- 从非 BMS 命令中生成的 HTML 页面的起始位置是屏幕的左上角（第 1 行，第 1 列）。

根据 HTML 页面的起始位置和创建顺序，从它们中选择标题部分：

1. 选择起始位置最接近屏幕第一行的页面。
2. 如果在选择过程中留有多个页面，则再次比较其余页面的起始位置。这次，选择起始位置最接近屏幕第一列的页面。
3. 最后，如果还留有多个页面，则选择最早生成的页面。

剩下的那个选择页面中的标题部分用在发送给 Web 客户机的 HTML 页面中。

如何选择页脚部分

注：在此描述中：

- 从 BMS 映射中生成的 HTML 页面的结束位置是映射右下角的行和列。
- 从非 BMS 命令中生成的 HTML 页面的结束位置是屏幕的右下角。

根据 HTML 页面的结束位置和创建顺序，从它们中选择页脚部分：

1. 选择结束位置最接近屏幕最后一行的页面。
2. 如果在选择过程中留有多个页面，则再次比较其余页面的结束位置。这次，选择结束位置最接近屏幕最后一列的页面。
3. 最后，如果还留有多个页面，则选择最后生成的页面。

剩下的那个选择页面的页脚部分用在发送给 Web 客户机的 HTML 页面中。

如何合并屏幕图像部分

合并了因一系列 BMS 和非 BMS SEND 命令而创建的屏幕图像部分后，就会创建一个新的屏幕图像部分；它尽可能地包含用于构造它的所有屏幕图像部分的所有字段。然而，如果来自两个或多个构成屏幕图像的字段被完全或部分重叠的话，就不可能发生这种情况，可能修改或完全抑制某些重叠字段：

- 字段重叠时，与早期 BMS 或非 BMS SEND 命令相关的字段会被修改或抑制而保留以后的命令。
- 如果输入字段被部分或完全重叠，就废弃整个输入字段，并且不会在最终的 HTML 中出现。
- 如果输入字段与一些常规文本部分重叠，那么无论在 3270 设备上输入字段结尾之后是否还有更多文本可见，到输入字段开始为止的任何可见文本在最终 HTML 中都是可见的，并废弃其余数据。
- 如果表单元格包含水平规则标记（<hr>），则重叠单元格内容将产生不可预测的结果。

表 7 中总结了这些规则。

表 7. 合并的屏幕图像部分中的重叠字段

早期 SEND 产生的字段	稍后 SEND 产生的字段	结果
输入（未保护）	输入（未保护）或文本（受保护）	完全抑制早期字段
文本（受保护）	输入（未保护）	基于 3270 屏幕中的字符位置： <ul style="list-style-type: none"> • 保留输入字段左侧的受保护字符。 • 抑制由输入字段覆盖的受保护字符。 • 抑制输入字段右侧的受保护字符。
文本（受保护）	文本（受保护）	根据 3270 屏幕中的字符位置，稍后 SEND 命令产生的字符将覆盖早期 SEND 命令产生的字符。

在应用程序中使用从 BMS 映射中创建的 HTML 模板之前可以编辑它们。然而，屏幕图像部分合并所依据的算法需要部分中的 HTML 具有特定的结构。因此，当编辑模板中的屏幕图像部分，并且该部分将与其他部分合并时，您必须遵循一些准则：

- 每个 HTML 页面都包含两个分别带有字符串 DFHROW 和 DFHCOL 的注释。跟随在这些字符串后的值在合并过程中非常重要，因为它们用于计算 3270 屏幕上每个字段的位置。如果这些注释被修改或删除，将不合并屏幕图像部分，但出现在最终 HTML 页面上，一个附加在另一个的下面。
- 表单元格的结束标记（</td>）表行的结束标记（</tr>）都是可选的。
- 表单元格必须包含带有或不带有额外属性标记的一段常规文本，或者必须包含一个输入字段。另外，如果文本和输入字段一个接一个，并且它们之间没有额外的标记的话，它们可以在同一个表单元格中包含混合的文本和输入字段。
- 空的表单元格在开始标记和结束标记之间可能不包含空值（X'00'）或空格。换句话说，必须将空单元编码为 <td></td>。
- 可以使用以下一对或多对标记将一段文本或一个输入字段部分括起：

强调 ...
加强 ...
字体 ...
下划线 <u> ... </u>
闪烁 <blink> ... </blink>

如果使用这些标记，必须确保每个标记都有相应的结束标记。还必须确保正确嵌套了开始和结束标记。例如，<u> ... </u> 嵌套是正确的，但 <u> ... </u> 嵌套得不正确。

- 如果将其他标记插入表单元格，则在同一个表单元格中，它们必须在文本或输入字段之前或之后出现，不能同时出现在文本或输入字段之前和之后。

- 允许在表单元格中出现 HTML 注释，它可以出现在一段常规文本或输入字段之前、之后或两侧。
- 如果单元格包含注释，它还必须包含一段常规文本或一个输入字段。

DFHMDX 宏

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

DFHMDX 宏是从 DFHMSX 中调用的。图 10 中显示了它的语法。

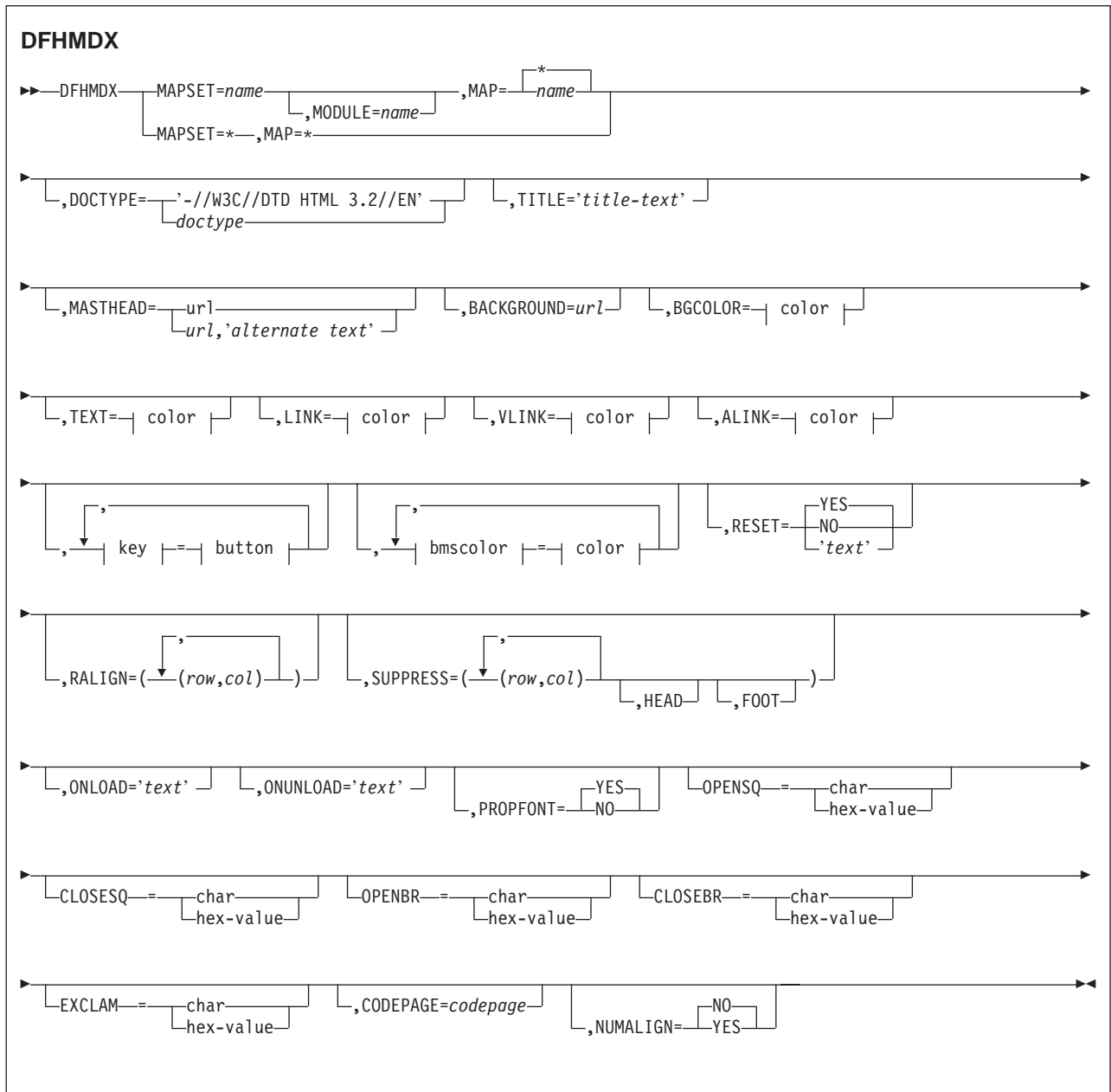


图 10. DFHMDX 的语法

这个宏的关键字参数可以以任何顺序出现。

MAPSET

指定包含其他选项所引用的映射的映射集的名称。如果指定星号，选项对于所有以后的映射集来说就成为缺省值。

MODULE

指定将映射集链接编辑到的装入模块的名称。如果未指定 **MAPSET=***，则只能使用这个参数。通过添加单一字符后缀，指定的名称（只能是七个字符）用于构造模板的名称。缺省值是映射集的名称。

MAP 指定选项引用的映射的名称，该映射位于 **MAPSET** 中指定的映射集里。如果指定星号，选项对于所有以后的映射来说就成为缺省值。

DOCTYPE

指定希望出现在 **HTML** 模板中的 `<!doctype>` 标记的 **DTD** 公共标识部分。缺省值是 `-//W3C//DTD HTML 3.2//EN`，它指定 **HTML 3.2**。在某些 **HTML** 标记中，颜色支持需要级别 3.2。

TITLE 指定要用作 **HTML** 标题以及第一个 `<h1>` 标记内容的标题。

MASTHEAD

指定在页标题上出现的报头图形的 **URL**，它在第一个 `L<h1>` 标记之前。如果提供备用文本，那么如果它无法装入指定的图形，客户机将使用文本。

BACKGROUND

指定用于页面背景的图形文件的 **URL**。

BGCOLOR

指定页面背景的颜色。

TEXT 指定常规文本的颜色。

LINK 指定页面上未访问超文本链接的颜色。

VLINK 指定页面上已访问超文本链接的颜色。

ALINK

指定页面上已激活的超文本链接的颜色。

PF1-PF24

指定为相应 3270 程序功能键分配给仿真按钮的名称或图像。

PA1-PA3

指定为相应 3270 程序辅助操作请求键分配给仿真按钮的名称或图像。

CLEAR

指定为 3270 **Clear** 键分配给仿真按钮的名称或图像。

ENTER

指定为 3270 **Enter** 键分配给仿真按钮的名称或图像。

PEN 指定为笔型选择分配给仿真按钮的名称或图像。

BLUE 指定要出现在已在 **BMS** 映射中指定了蓝色的 **HTML** 页面上的颜色。缺省值是 `#0000FF`。

限制: **DFHMDX** 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

GREEN

指定要出现在已在 **BMS** 映射中指定了绿色的 **HTML** 页面上的颜色。缺省值是 `#008000`。

限制: DFHMDX 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

NEUTRAL

指定要出现在已在 BMS 映射中指定了中间色的 HTML 页面上的颜色。缺省值是 #000000。

限制: DFHMDX 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

PINK 指定要出现在已在 BMS 映射中指定了粉色的 HTML 页面上的颜色。缺省值是 #FF00FF。

限制: DFHMDX 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

RED 指定要出现在已在 BMS 映射中指定了红色的 HTML 页面上的颜色。缺省值是 #FF0000。

限制: DFHMDX 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

TURQUOISE

指定要出现在已在 BMP 映射中指定了青绿色的 HTML 页面上的颜色。缺省值是 #00FFFF。

限制: DFHMDX 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

YELLOW

指定要出现在已在 BMS 映射中指定了黄色的 HTML 页面上的颜色。缺省值是 #FFFF00。

限制: DFHMDX 将只覆盖未命名字段的颜色; 它保留命名字段未更改。

RESET

指定是否支持 HTML 复位功能。指定 YES 可以获得一个带有缺省图注 Reset 的缺省复位按钮。指定 NO 不会获得复位按钮。使用自己的图注为复位按钮指定自己的文本。

RALIGN

指定 BMS 映射字段中的数据将在 HTML 页面中向右对齐。指定的值 *rr* 和 *cc* 必须与 DFHMDF 宏的 POS=(*rr,cc*) 规范对应, 以使字段向右对齐。每一对都必须用圆括号括起, 各对的完整列表也必须用圆括号括起。如果希望以特定列结束的每个限定字段都向右对齐, 指定结束列号, 然后为行规范输入星号。通过将字段的开始列号添加到 LENGTH 中来计算字段的结束列号, 如 DFHMDF 宏中定义的那样。只有字段是在受保护、未命名并在 DFHMDF 宏中使用 INITIAL、XINIT 或 GINIT 值初始化的情况下才能右对齐。如果使用 MAP=* 或 MAPSET=* 指定 RALIGN 参数, 则忽略它。

如果希望在宏定义中指定超过汇编程序对字符串 256 个字符限制的列表, 则使用相同的 MAPSET 和 MAP 值对额外的 DFHMDX 宏进行编码, 并将更多的值放入 RALIGN 参数中。

SUPPRESS

指定不出现在 HTML 页面中的 BMS 映射字段。为要抑制的字段的起始位置指定任何号码的行列对。指定的值 *rr* 和 *cc* 必须与 DFHMDF 宏的 POS=(*rr,cc*) 规范对应, 以抑制字段。每一对都必须用圆括号括起, 各对的完整列表也必须用圆括号括起。如果希望抑制一行中的所有字段, 指定行号, 然后为列规范输入星号。如果使用 MAP=* 或 MAPSET=* 指定 SUPPRESS 参数, 则忽略它。

使用关键字 HEAD 抑制模板中的标题信息。使用关键字 FOOT 抑制模板中的页脚信息。

如果希望在宏定义中指定超过汇编程序对字符串 256 个字符限制的列表，请使用相同的 MAPSET 和 MAP 值对额外的 DFHMDX 宏进行编码，并将更多值放入 SUPPRESS 参数中。

ONLOAD

指定要用于替换 HTML 页面的标准 onLoad 异常处理程序的 JavaScript 文本。文本不能包含双引号 (")，必须按照常规的汇编语言约定将单引号 (') 加倍 (``)。如果使用这个参数，则将光标的设置抑制到由 DFH_CURSOR 标出的字段，DFH_CURSOR 是由标准 onLoad 异常处理程序提供的。可以使用函数 dfhsetcursor 来设置光标位置。

ONUNLOAD

指定要用作 HTML 页面的 onUnload 异常处理程序的 JavaScript 文本。文本不能包含双引号 (")，必须按照常规的汇编语言约定将单引号 (') 加倍 (``)。

PROPFONT

指定字体。如果指定 YES，模板指定文本将以比例字体出现，并将连续空格减少到一个空格。如果指定 NO，模板指定文本以固定间距字体指定，并保留连续空格。

OPENSQ

用于显示左方括号的十六进制值或字符。缺省值是 X'BA' (代码页 37)。

CLOSESQ

用于显示右方括号的十六进制值或字符。缺省值是 X'BB' (代码页 37)。

OPENBR

用于显示左大括号的十六进制值或字符。缺省值是 X'C0' (代码页 37)。

CLOSEBR

用于显示右大括号的十六进制值或字符。缺省值是 X'D0' (代码页 37)。

EXCLAM

用于显示感叹号的十六进制值或字符。缺省值是 X'5A' (代码页 37)。

CODEPAGE

指定用于编码模板生成进程所生成的所有文本的 IBM 代码页号。这个代码页必须与 CICS 使用模板时使用的代码页匹配，使用的代码页在 EXEC CICS DOCUMENT 命令的 HOSTCODEPAGE 选项或分析器程序选择的 DFHCNV 宏的 SRVERCP 选项中。CICS 支持的 IBM 主机代码页在 *CICS Family: Communicating from CICS on System/390* 中描述。缺省代码页是 037。

NUMALIGN

指定在 HTML 模板中的表单元中如何对齐 DFHMDF 宏中明确定义为数字的字段：

NO 指定在表单元中不右对齐的数字字段。这是缺省值。

YES 指定在表单元中右对齐的数字字段：

- 对于保护字段，生成的 HTML 文本在单元中右对齐。如果文本包含多个尾部空格，则可能不保存它们：某些客户机将使用单个空格替换它们。

注: RALIGN 参数保存尾部多个空格; NUMALIGN 参数不保存。如果两个参数都应用于一个字段(即, 如果数字字段由 RALIGN 参数标识, 而且指定了 NUMALIGN=YES), 则不保存尾部的多个空格。

- 对于未保护字段, HTML 文本输入元素(但不是元素中的文本)在单元中右对齐。

color 可以是显式规范 #rrggb, 其中 rr、gg 和 bb 是 2 位十六进制数字, 它们提供了在所请求颜色的红色、绿色和蓝色的强度, 它也可以是以下颜色名称中的任何一个: AQUA、BLACK、BLUE、FUCHSIA、GRAY、GREEN、LIME、MAROON、NAVY、OLIVE、PURPLE、RED、SILVER、TEAL、WHITE 和 YELLOW。

key 可以是 PF1 到 PF24、PA1 到 PA3、CLEAR、ENTER 和 PEN 中的任何一个。

button 可以是 (IMAGE,url), 其中 url 指定要用于按钮的图象的 URL, 或者是"text", 其中 text 是要放在按钮上的文本, 如果按钮不出现, 则为 NO。

bmscolor 可以是 BLUE、GREEN、NEUTRAL、PINK、RED、TURQUOISE 和 YELLOW 中的任何一个。

使用 DFHWBOUT 宏定制模板

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

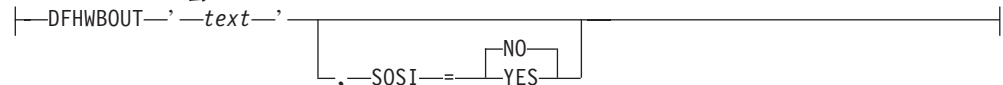
DFHWBOUT 宏用于向从 BMS 映射生成的 HTML 页面添加文本。文本仅作为 HTML 页面的一部分出现。如果在映射中, 宏在 DFHMDF 第一次出现之前使用, 则将文本放在 HTML 页面的 <head> 部分。如果宏用在映射中的其他地方, 则文本紧跟着前面的 DFHMDF 宏生成的文本放置。

当应用程序使用多个 BMS 映射构建屏幕显示时, 不要使用 DFHWBOUT 宏。

DFHWBOUT



DFHWBOUT 宏



这个宏的参数如下:

text 要插入 HTML 页面中的文本。

SOSI 文本是否包含由 shift-out (X'0E') 和 shift-in (X'0F') 定界的 DBCS 字符。缺省值是 SOSI=NO。

在使用 DFHWBOUT 宏时, 要知道插入的 HTML 文本可能影响从 BMS 映射字段中生成的页面布局。可能需要调整插入的文本来确保正确的页面布局。

定制示例

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

下面的样本显示了定制宏定义。通过为映射集名和映射名指定 *，DFHMDX 的第一个调用为应用到 DFHMDX 后面的调用设置缺省值。稍后的调用对映射集中特定的映射重设或添加参数。接续字符位于第 72 列，接下去的文本在第 16 列继续。

```
MACRO
DFHMSX
DFHMDX MAPSET=*,MAP=*,
PF1='Help',PF3='Exit',PF4='Save',PF9='Messages' *
DFHMDX MAPSET=DFHWB0,MAP=*,
TITLE='CICS Web Interface', *
PF3='Messages' *
DFHMDX MAPSET=DFHWB0,MAP=DFHWB02, *
TITLE='CICS Web Interface Enable', *
PF3='Save'
MEND
```

当 CICS 创建模板时，对于每个 BMS 映射定义，它调用 DFHMSX 定制宏。每个 DFHMDX 宏按顺序处理，如果合适，存储参数值。为特定映射或映射集指定重复参数的情况下，新值只替换此映射或映射集的先前值。

- 此示例中的第一个 DFHMDX 宏指定 MAPSET=*,MAP=* 和 PF3='Exit'。PF3 的这个值应用于每个映射集和在后继 DFHMDX 宏中未指定不同值的映射。
- 第二个 DFHMDX 宏指定 MAPSET=DFHWB0,MAP=* 和 PF3='Messages'。PF3 的这个值应用于在后继 DFHMDX 宏中未指定不同值的映射集 DFHWB0 中的每个映射。
- 第三个 DFHMDX 宏指定 MAPSET=DFHWB0,MAP=DFHWB02 和 PF3='Save'。此值仅应用于映射集 DFHWB0 中的映射 DFHWB02。

从 BMS 映射中生成的缺省模板包含表示以下所有键的按钮：

- Enter 键
- PA1、PA2 和 PA3 键
- 程序功能键 PF1 到 PF24
- HTML 复位

然而，如果使用 DFHMDX 宏指定模板中希望存在的按钮，那么只有指定的按钮才包含在模板中。例如，如果您指定

```
DFHMDX MAPSET=*,MAP=*,PF3='Exit',ENTER='Continue'
```

则模板将只包含 PF3 和 Enter 键的按钮。

这里有进一步的示例显示可以如何定制从 BMS 映射生成的 HTML 模板。

- 支持应用程序使用不是标准输出中的键。

可以按如下方式向映射 AD001 添加按钮：

```
DFHMDX MAP=AD001,PF18='Resubmit'
```

Web 客户机显示带有图注“Resubmit”的按钮。如果用户单击该按钮，就作为 PF18 向应用程序报告。

- 抑制 HTML Reset 功能。

可以按如下方式抑制映射 AD001 的 Reset 功能：

```
DFHMDX MAP=AD001,RESET=NO
```

Web 客户机显示不包含 Reset 按钮的页面。

- 更改按钮的外观或与它们关联的文本。

可以按如下方式更改 PF1 按钮上的图注:

```
DFHMDX PF1='Help'
```

Web 客户机显示具有图注“Help”的按钮。如果用户单击该按钮,它就作为 PF1 提供给应用程序。

- 为 HTML 页面提供 HTML 标题。

可以按如下方式向显示的映射添加标题:

```
DFHMDX MAP=DFHWB01,TITLE='CICS Web Interface'
```

Web 客户机显示“CICS Web 接口”作为页面标题。

- 为 HTML 页面提供报头图形。

为将具有头的映射编写 DFHMDX 宏。例如:

```
DFHMDX MASTHEAD=(/dfhwbimg/masthead.gif,'CWI')
```

Web 客户机使用指定的头,如果找不到图形文件,则显示“CWI”作为头。

- 更改背景的颜色,或指定特殊背景。

为将具有特殊背景的映射编写 DFHMDX 宏。例如:

```
DFHMDX MAP=AD001,BACKGROUND=/dfhwbimg/texture4.jpeg
```

Web 客户机使用指定的文件作为页面背景。

要更改背景的颜色,使用 BGCOLOR 参数。

- 修改 BMS 颜色。

要修改 BMS 颜色,编写类似于以下内容的 DFHMDX 宏:

```
DFHMDX MAP=AD001,BLUE=AQUA,YELLOW=#FF8000
```

Web 客户机以 HTML 浅绿色(与 BMS 青绿色相同)显示 BMS 蓝色文本,以亮橙色显示 BMS 黄色文本。

- 抑制 BMS 映射的某些部分。

可以按如下方式抑制映射中的字段:

```
DFHMDX MAP=AD001,SUPPRESS=((5,2),(6,2),(7,*))
```

显示的页面不包含映射中第 5 行第 2 列的字段,也不包含第 6 行第 2 列的字段,也不包含第 7 行中的任何字段。

- 添加 Web 客户机控制函数。

如果希望在装入页面时调用 JavaScript 函数,在定制宏中使用 DFHMDX 宏的 ONLOAD 参数。例如,如果对以下内容进行编码:

```
DFHMDX MAP=AD001,ONLOAD='jset(''CWI is wonderful'',''Hello there!'')
```

则在装入页面时使用给定参数调用 JavaScript 函数 `jset()`。

要完成这种定制，必须使用 `DFHWBOUT` 宏将 `jset` 函数的定义添加到 HTML 页面的头中。必须将宏调用放在 `BMS` 映射定义中第一个 `DFHMDF` 宏之前。下面是一个样本：

```
DFHWBOUT '<script language="JavaScript">'
DFHWBOUT 'function jset(msg,wng)'
DFHWBOUT '    {window.status = msg; alert(wng)}'
DFHWBOUT '</script>'
```

装入页面时，窗口底部的状态区包含消息“CWI is wonderful”，并且打开一个包含消息“Hello there!”的警告窗口。

- 添加只出现在 **HTML** 页面上但不是 **BMS** 映射一部分的文本。

将 `DFHWBOUT` 宏放在 `BMS` 映射定义中希望文本出现的地方。例如：

```
DFHWBOUT '<p>This text illustrates the use of the DFHWBOUT macro,'
DFHWBOUT 'which can be used to output text that should only appear'
DFHWBOUT 'in HTML templates, and will never appear in the'
DFHWBOUT 'corresponding BMS map.'
```

将在 `HTML` 模板中产生以下几行：

```
<p>This text illustrates the use of the DFHWBOUT macro,
which can be used to output text that should only appear
in HTML templates, and will never appear in the
corresponding BMS map.
```

- 向 **HTML** 页面添加 **HTML** 头信息。

将 `DFHWBOUT` 宏放在 `BMS` 映射定义中出现的第一个 `DFHMDF` 之前。例如：

```
DFHWBOUT '<meta name="author" content="E Phillips Oppenheim">'
DFHWBOUT '<meta name="owner" content="epoppenh@xxxxxxx.yyy.co*
m">'
DFHWBOUT '<meta name="review" content="19980101">'
DFHWBOUT '<meta http-equiv="Last-Modified" content="&WBDATE&W*
BTIME GMT">'
```

将在 `HTML` 模板头部分中产生以下几行：

```
<meta name="author" content="E Phillips Oppenheim">
<meta name="owner" content="epoppenh@xxxxxxx.yyy.com">
<meta name="review" content="19980101">
<meta http-equiv="Last-Modified" content="23-Dec-1997 12:06:46 GMT">
```

`DFHMSD` 将 `&WBDATE` 和 `&WBTIME` 的值设置成汇编宏时的时间和日期。

- 在 **JavaScript** 和 **HTML** 中使用特定于国家或地区的字符。

用于产生模板的缺省美式英语代码页 37 可以修改为其他代码页。例如：

```
DFHMDX OPENSQ=[,CLOSESQ=],OPENBR={,CLOSEBR=},EXCLAM=!
```

这指定需要的替换。字符必须在代码页与 `DFHCNV` 调用上的 `SERVERCP` 相应的终端上输入。

- 使字段在 **HTML** 页面中向右对齐。

可以按如下方式将字段中的数据向右对齐：

```
DFHMDX MAPSET=MAPSETA,MAP=AD001,RALIGN=((3,5),(*,15),(*,3),(6,7),(*,83))
```

在该示例中，下面所有字段中的数据都向右对齐

```
DFHMDF POS=(3,5),LENGTH=4,INITIAL='TEXT',ATTRB=PROT
DFHMDF POS=(5,80),LENGTH=3,INITIAL='123',ATTRB=PROT
DFHMDF POS=(2,10),LENGTH=5,INITIAL=' EXT',ATTRB=ASKIP
DFHMDF POS=(4,8),LENGTH=7,INITIAL='INITEX ',ATTRB=PROT
DFHMDF POS=(1,1),LENGTH=2,XINIT='C1C2',ATTRB=ASKIP
DFHMDF POS=(6,7),LENGTH=4,XINIT='0E44850F',ATTRB=PROT,SOSI=YES
DFHMDF POS=(2,9),LENGTH=6,XINIT='0E448544830F',SOSI=YES,ATTRB=PROT
DFHMDF POS=(2,9),LENGTH=6,XINIT='448544834040',PS=8,ATTRB=PROT
```

- **使数字字段右对齐**

您可以使带有 NUMERIC 属性的所有字段在其 html 表单元中右对齐，如下所示：

```
DFHMDX MAPSET=MAPSETA,MAP=AD001,NUMALIGN=YES
```


第 17 章 CICSplex 中的 CICS Web support

您可以单独或组合使用以下方法，在 CICSplex 中分布使用 CICS Web support 的应用程序：

- 您可使用网络负载均衡将来自 Web 客户机的请求分布到多个 CICS 区域。
- CICS Web support 和业务应用程序可以在同一 CICS 区域中执行。
- CICS Web support 可以在路由器区域中执行，而业务应用程序可以在一个或多个应用程序所属区域（AOR）中执行。然而，您无法在 AOR 中使用 EXEC CICS WEB API，因此支持 Web 的应用程序不能在 AOR 中执行。您可以在 AOR 中使用 EXEC CICS DOCUMENT API，但是您必须提供您自己的机制，将 HTML 输出传输回路由器区域。要获得更多信息，请参阅第 164 页的『将 Web 客户机的请求路由到 AOR』。

图 11 描述这些配置。

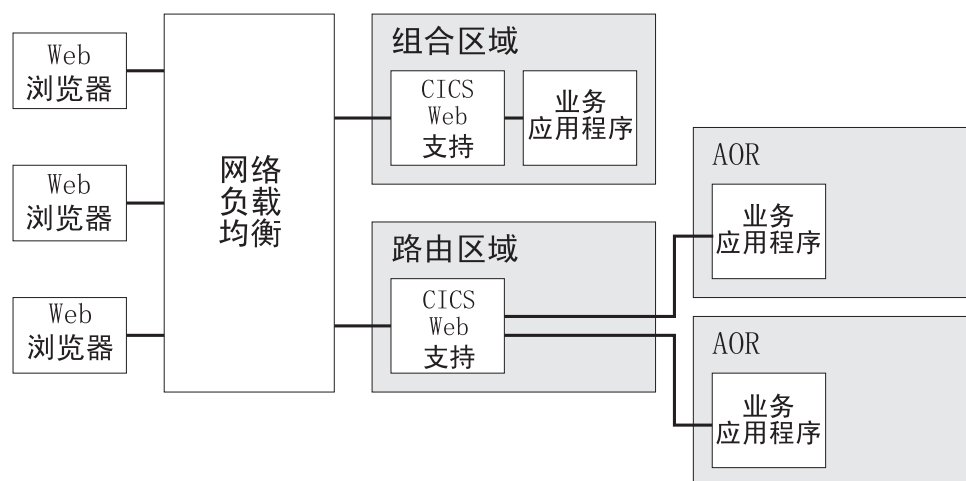


图 11. CICSplex 中的 CICS Web support 配置

您可以用相同的方式分布使用 CICS 业务逻辑接口的请求。这在第 164 页的图 12 中有说明。

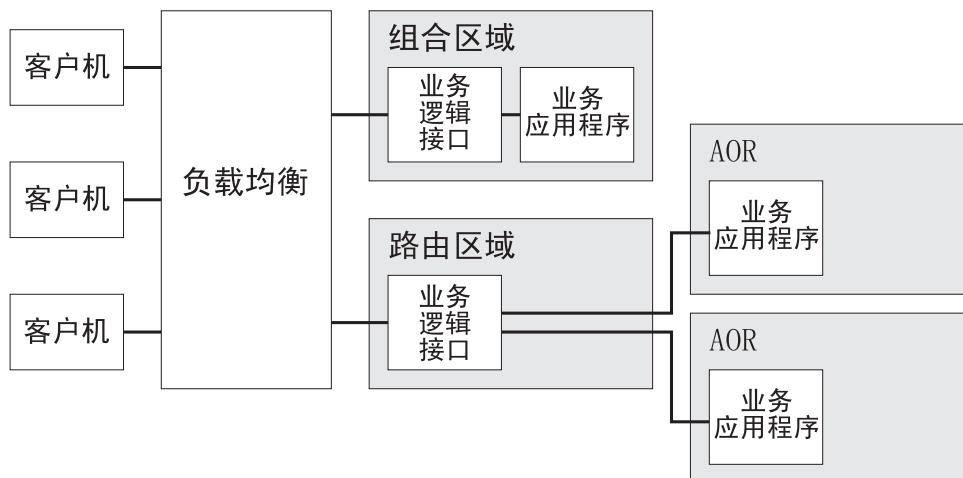


图 12. CICSplex 中的 CICS 业务逻辑接口配置

当您计划以此方法分布应用程序时，您应该考虑应用程序各部分之间存在的任何亲缘关系。要获取有关亲缘关系的更多信息，请参阅 *CICS Interdependency Analyzer for z/OS User's Guide and Reference*。

您还应该考虑如何在请求之间管理应用程序的状态。第 70 页的『跨 HTTP 请求序列管理应用程序状态』讨论涉及任何 CICS Web support 应用程序的注意事项，这些应用程序使用伪会话模型。在以下情况下，可能存在其他注意事项：

- 动态路由选择用于选择执行业务应用程序的 AOR。
- 工作负载和连接平衡用于选择路由器区域（并间接地选择 AOR）。

CICS 提供您可用于管理应用程序状态的样本状态管理程序 (DFH\$WBSR)。DFH\$WBSR 使通过资源共享应用程序状态变得容易，这些资源可以由多个 CICS 区域共享。它在第 233 页的附录 J，『状态管理样本 DFH\$WBST 和 DFH\$WBSR 的参考信息』中描述。（另一个样本 DFH\$WBST 创建亲缘关系，因此它不适合在 CICSplex 中使用。）

要获取有关在 CICSplex 中配置 CICS Web support 和 CICS 业务逻辑接口的指导，请参阅 *Workload Management for Web Access to CICS*。

将 Web 客户机的请求路由到 AOR

您无法在 AOR 中使用 EXEC CICS WEB API；您只能在路由器区域中使用它。如果要使用支持 Web 的应用程序响应请求，则一个解决方案是在支持 Web 的应用程序（它在路由器区域中执行）编码您的演示逻辑，并将您的业务逻辑（它在 AOR 中执行）编码为完全独立于演示。支持 Web 的应用程序命名为处理请求的程序，并且它必须管理它自己与执行业务逻辑的应用程序的通信。第 21 页的『作为 HTTP 服务器的 CICS 的 HTTP 请求和响应处理』说明支持 Web 的应用程序的处理阶段。

对于不支持 Web 的应用程序，路由器区域中的转换器程序可用于从 AOR 中的应用程序提供的信息产生 HTTP 响应。第 165 页的图 13 显示在应用程序所属区域中执行不支持 Web 的应用程序时，与来自 Web 客户机的请求关联的处理阶段和任务结构。

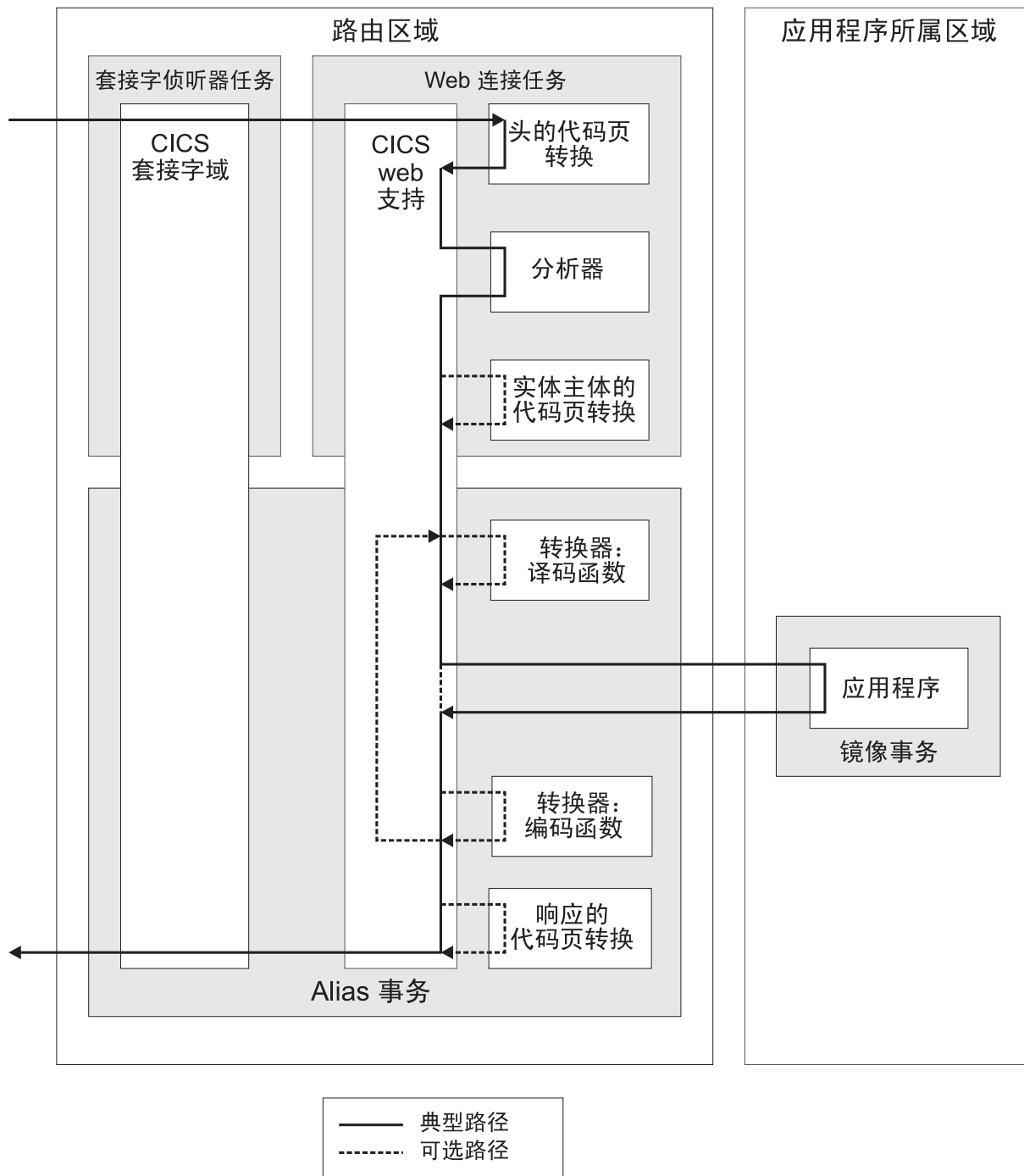


图 13. CICS Web support 如何将不支持 Web 的应用程序请求路由到 AOR

CICS 业务逻辑接口的相应阶段在第 166 页的图 14 中显示。

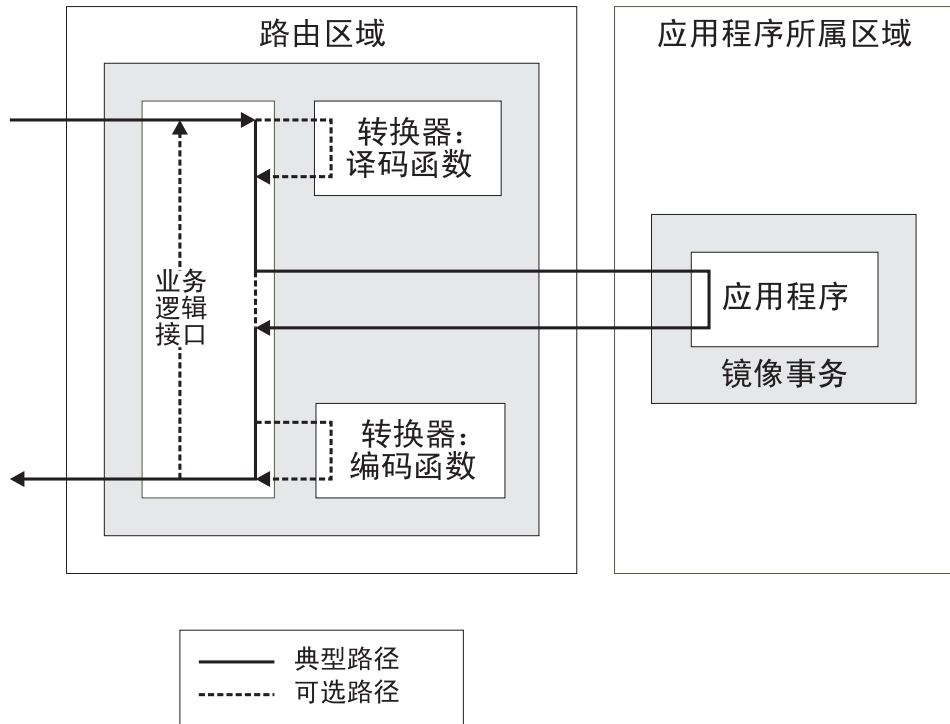


图 14. CICS 业务逻辑接口如何将应用程序请求路由到 AOR

CICS 使用分布式程序链接 (DPL) 调用 AOR 中的应用程序; 应用程序在镜像任务下执行。要获取有关 DPL 的信息, 请参阅 *CICS Intercommunication Guide*。

要在 AOR 中执行业务应用程序:

- 您必须指定 REMOTESYSTEM 属性, 或在应用程序的 PROGRAM 定义上指定 DYNAMIC(YES)。如果您指定 DYNAMIC(YES), 动态路由选择程序确定执行应用程序的位置。
- 其他资源定义 (用于分析器程序、支持 Web 的应用程序或转换器程序以及别名事务) 必须指定它们在路由器区域中执行。
- 您必须定义路由器区域和 AOR 之间的 MRO 或 APPC 连接。

如果 AOR 中执行的应用程序设计为完全独立于演示, 则它将输出返回到支持 Web 的应用程序或转换器程序, 然后它构造 HTML 输出。另外, 如果您正在使用转换器程序, 则可能要在 AOR 中使用 EXEC CICS DOCUMENT API 以构造 HTML 输出。转换器程序可以将此输出转换到包含完整 HTTP 响应的通信区域, 以由 CICS Web support 发送。

您必须提供您自己的机制, 以将应用程序的输出传回路由器区域。输出可以在通信区域中传输。另外, 您还可以使用某些其他机制 (例如, 临时存储队列), 并传输以该机制表示数据的令牌。路由器区域中的程序可以使用令牌获取输出, 然后处理它并将它传递到 Web 客户机。CICS 提供您可用于执行此操作的样本状态管理程序 (DFH\$WBSR)。它在第 233 页的附录 J, 『状态管理样本 DFH\$WBST 和 DFH\$WBSR 的参考信息』中描述 (其他样本 DFH\$WBST 创建亲缘关系, 所以不适合于在 CICSplex 中使用)。

网络负载均衡

要避免从属于单个 CICS 路由器区域，您应该考虑使用多个路由器区域从网络共享工作负载。有几项您可用于均衡路由器区域之间工作负载的技术：

综合系统分发器

综合系统分发器是 z/OS Communications Server 的功能部件，其用于平衡跨多个 IP 堆栈的 IP 分组。要获取有关综合系统分发器的更多信息，请参阅 *z/OS Communications Server: IP Configuration Guide*, SC31-8775。

虚拟 IP 寻址 (VIPA)

动态 VIPA 是 z/OS 通信服务器的功能部件，其提供围绕失败网络适配器的非破坏性路由。要获取有关 VIPA 的更多信息，请参阅 *z/OS Communications Server: IP Configuration Guide*, SC31-8775。

DNS 方法

DNS 连接优化平衡 z/OS 综合系统 IP 域中的 IP 连接，这基于来自 MVS™ WLM 有关已注册应用程序运行状况的反馈。为了供 CICS 使用，它仍受支持。要获取有关 DNS/WLM 支持的更多信息，请参阅 *z/OS Communications Server: IP Migration*, GC31-8773。

端口共享

TCP/IP 端口共享提供一种简单的方法，它可以将 HTTP 请求遍布在一组在同一 z/OS 映像中运行的 CICS 路由器区域上。不同区域中的 CICS TCPIP SERVICE 配置为在同一端口上侦听，而 TCP/IP 由 SHAREPORT 选项配置。然后，TCP/IP 堆栈平衡跨侦听器的连接请求。要获取有关 TCP/IP 端口共享的更多信息，请参阅 *z/OS Communications Server: IP Configuration Reference*, SG24-5466。

第 3 部分 CICS 业务逻辑接口

本书的这一部分包含有关 CICS 业务逻辑接口的信息。

第 18 章 CICS 业务逻辑接口的介绍

CICS 业务逻辑接口使链接到支持 Web 的业务应用程序成为可能，而不是通过 CICS HTTP 侦听器调用它。例如，z/OS 上运行的 Web 服务器可以使用外部 CICS 接口（EXCI）链接到使用 CICS 业务逻辑接口的应用程序。在此方式中，Web 客户机可以通过中介 Web 服务器与 CICS 应用程序通信，而不是直接连接到 CICS。

第 217 页的附录 G，『DFHWBBLI CICS 业务逻辑接口的参考信息』具有该接口的参考信息。

如何使用 CICS 业务逻辑接口

您可以在可链接到 CICS 应用程序的任何环境中调用 CICS 业务逻辑接口。例如：

- 您可以从 CICS 应用程序发出 LINK 命令。
- 您可以使用外部 CICS 接口（EXCI）。
- 可以从客户机上使用外部调用接口（ECI）。
- 您可以从 ONC RPC 客户机使用 CICS ONC RPC 支持。

CICS 业务逻辑接口由以下项使用：

- CICS Web 服务器插件。该插件使用外部 CICS 接口调用 CICS 业务逻辑接口。

处理示例

图 15 显示了 CICS 业务逻辑接口如何处理来自使用 EXCI 的 MVS 应用程序的请求。

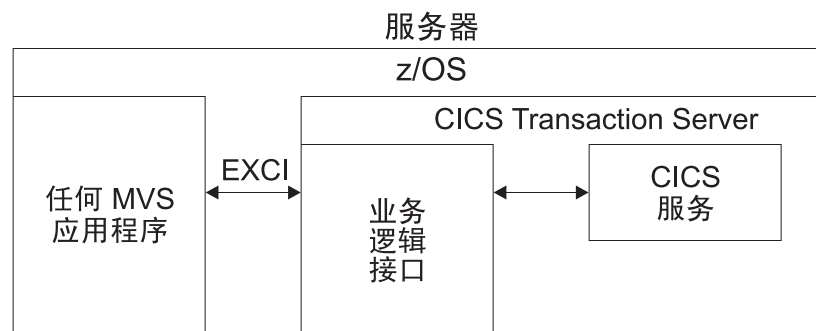


图 15. 处理 EXCI 的请求

1. MVS 应用程序构造包含 CICS 业务逻辑接口的参数的通信区域。
2. MVS 应用程序使用 EXCI 调用 CICS 业务逻辑接口。
3. CICS 业务逻辑接口调用请求的服务，并在通信区域中返回任何输出。

第 172 页的图 16 显示了 CICS 业务逻辑接口如何处理来自使用 ECI 的 CICS 客户机的请求。

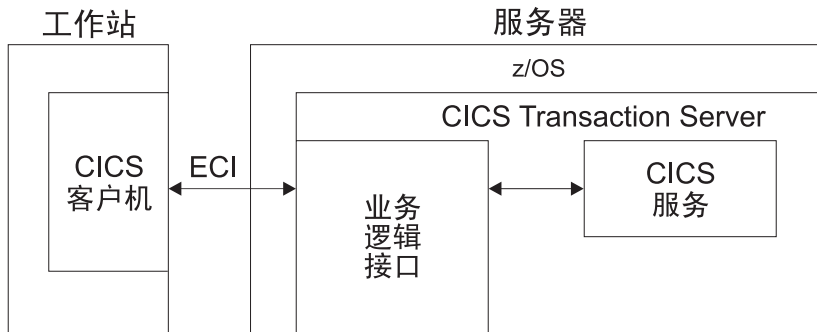


图 16. 处理 ECI 请求

1. 在工作站环境中运行的客户机构造包含 CICS 业务逻辑接口的参数的通信区域。
2. 客户机使用 ECI 调用 CICS 业务逻辑接口。
3. CICS 业务逻辑接口调用请求的服务，并在通信区域中返回任何输出。

ECI 与 SNA 协议或 TCP62 协同操作，它允许通过 TCP/IP 建立 SNA 连接（要获取更多信息，请参阅 *CICS Family: Client/Server Programming*）。

请求处理中的控制流

要决定有关您将使用的工具以及如何定制它们，需要理解 CICS 业务逻辑接口的组件是如何交互作用的。

使用 CICS 业务逻辑接口调用程序

图 17 显示了通过 CICS 业务逻辑接口到达程序的控制流。通过到 PROGRAM DFHWBBLI 的 LINK 命令来访问 CICS 业务逻辑接口。

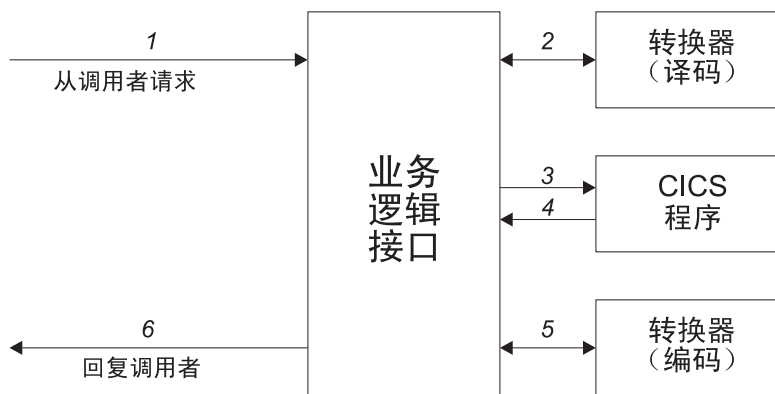


图 17. 用 CICS 业务逻辑接口 - 控制流调用程序

1. 到达 CICS 业务逻辑接口的请求。
2. 如果调用者请求转换器，则 CICS 业务逻辑接口调用它，并请求解码函数。译码为 CICS 应用程序设置通信区域。
3. CICS 业务逻辑接口调用调用者指定的 CICS 应用程序。传递到应用程序的通信区域是由译码设置的一个区域。如果 CICS 业务逻辑接口的调用者表明不需要转换器，则请求的第一个 32K 字节传递到它的通信区域中的 CICS 应用程序。

4. CICS 应用程序处理请求，并在通信区域中返回输出。
5. 如果调用者请求转换器，则 CICS 业务逻辑接口调用转换器的**编码函数**，该函数使用通信区域来准备响应。如果没有调用转换器程序，则 CICS 业务逻辑接口假定 CICS 应用程序已将需要的响应放在通信区域中。
6. CICS 业务逻辑接口将应答发送回调用者。

使用 CICS 业务逻辑接口运行面向终端的事务

图 18 显示了控制流，该控制流通过 CICS 业务逻辑接口到达面向终端的事务的请求。请注意，业务逻辑接口在 CICS 镜像事务下运行，而不是在 Web CICS 事务下运行。处理的第一部分与调用程序相同，但是如果您要运行事务，则必须在 `wbbl_server_program_name` 中将 DFHWBTTA 指定为要调用的 CICS 应用程序。

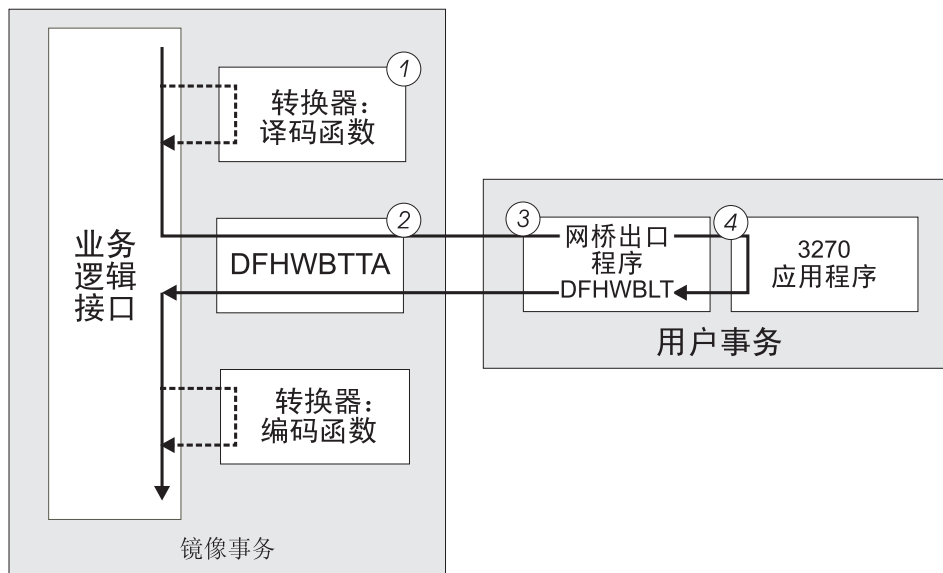


图 18. 用 CICS 业务逻辑接口 - 控制流运行事务

1. 如果调用者请求转换器，则 CICS 业务逻辑接口调用它，并请求**解码函数**。**译码**为 DFHWBTTA 设置通信区域。
2. CICS 业务逻辑接口调用 DFHWBTTA。传递到 DFHWBTTA 的通信区域是由**译码**设置的一个区域。如果没有调用转换器程序，则通信区域包含整个请求。
3. DFHWBTTA 从 HTTP 请求抽取面向终端的事务的事务标识，并启动运行 CICS Web 网桥出口的事务。
4. 当程序尝试写到它的主要功能时，CICS Web 网桥出口拦截这些数据。该出口构造返回到 CICS 业务逻辑接口的 HTML 响应。如果调用者请求转换器，则 CICS 业务逻辑接口调用转换器的**编码函数**，该函数使用通信区域来准备响应。如果没有调用转换器程序，则 CICS 业务逻辑接口假定通信区域包含需要的响应。

请求处理中的数据流

要决定有关您将使用的工具以及如何定制它们，需要理解数据是如何在 CICS 业务逻辑接口中传递的。

转换器程序和 CICS 业务逻辑接口

CICS 系统中可以存在多个转换器程序，以支持 CICS 业务逻辑接口的操作。第 172 页的图 17 和 第 173 页的图 18 中说明了转换器在 CICS 业务逻辑接口中的位置。每个转换器必须提供两个函数：

- **解码**在调用 CICS 应用程序前使用。它可以：
 - 以应用程序需要的格式，使用入局请求中的数据来构建通信区域。
 - 提供应用程序通信区域中输入和输出数据的长度。
 - 执行有关请求的管理任务。
- **编码**在已调用 CICS 应用程序后使用。它可以：
 - 使用应用程序中的数据来构建响应。
 - 执行有关响应的管理任务。

Notes:

- 如果改变了 `DECODE_DATA_PTR` 或 `ENCODE_DATA_PTR` 来寻址另一个存储位置，那么转换器程序负责将原来的存储器清空。
- CICS 业务逻辑接口的调用者负责释放 `ENCODE_DATA_PTR` 寻址的缓冲区（即，字段 `WBBL_OUTDATA_PTR` 中返回的地址减 4）。
- 如果转换器异常终止，则 CICS 将尝试释放 `DECODE_DATA_PTR` 和 `ENCODE_DATA_PTR` 寻址的存储器。因此，您应该确保这些指针从不包含已释放的存储地址。

使用 CICS 业务逻辑接口调用程序

第 175 页的图 19 说明了数据流通过 CICS 业务逻辑接口到达程序，然后返回请求者的过程。

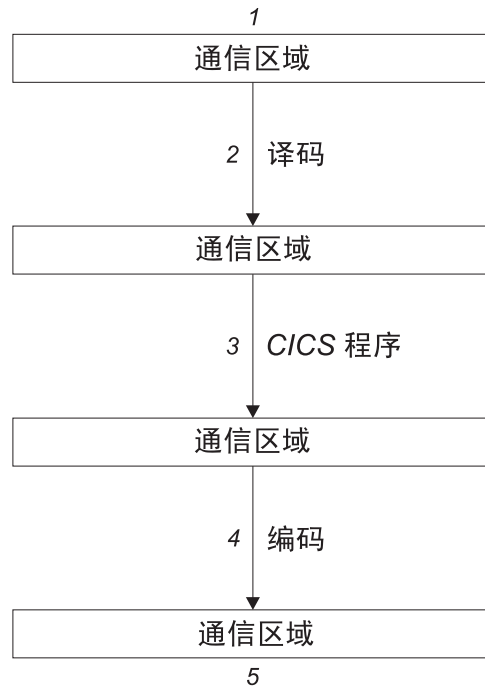


图 19. 用 CICS 业务逻辑接口 - 数据流调用程序

1. CICS 业务逻辑接口的调用者提供包含要处理的请求的通信区域。该通信区域的内容必须是后续进程可以接受的代码页。通常这意味着它们必须是 EBCDIC。
2. 如果调用者请求转换器，则转换器的译码函数为 CICS 应用程序构造通信区域。
3. CICS 应用程序更新通信区域。
4. 如果调用者请求转换器，则转换器的编码函数构造要返回给调用者的通信区域。
5. CICS 业务逻辑接口返回到它的调用者，现在该调用者就可以使用通信区域的内容了。

面向终端的事务的请求

第 176 页的图 20 显示了启动一个面向终端事务的请求的数据流。

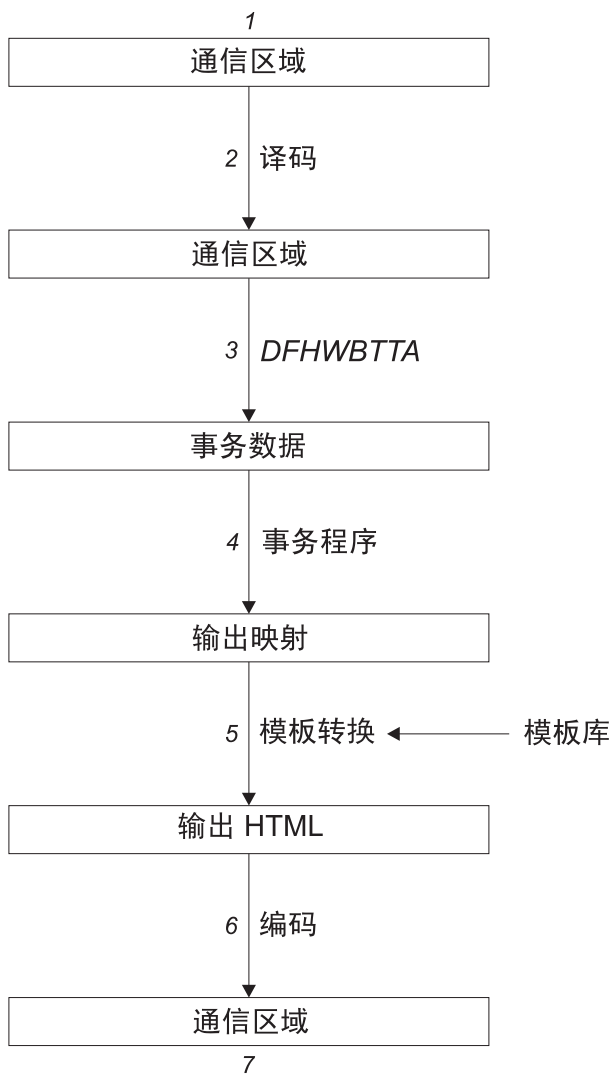


图 20. 启动面向终端的事务 - 数据流

此图显示了经过 CICS 业务逻辑接口的 3270 BMS 应用程序的数据流。

1. CICS 业务逻辑接口的调用者提供包含要处理的请求的通信区域。该通信区域的内容必须是后续进程可接受的代码页，并且 DFHWTBTA 需要 EBCDIC。
2. 如果需要，可以使用转换器的**解码**函数来修改请求。
3. 由于这是对话或者伪会话的第一个事务，因此请求包括了事务标识，还可能包括事务程序可用的数据。DFHWTBTA 抽取数据，以使它可通过 RECEIVE 命令用于事务程序。
4. 事务程序使用 RECEIVE 命令接收数据。然后它构造输出映射，并使用 SEND MAP 命令将它发送给请求者。
5. 映射及其数据内容转换为 HTML 格式。这个转换使用 DOCTEMPLATE 定义中定义的模板。
6. 如果需要，可以使用转换器的**编码**函数来修改响应。
7. CICS 业务逻辑接口返回到它的调用者，现在该调用者就可以使用通信区域的内容了。

图 21 显示了继续面向终端事务请求的数据流。

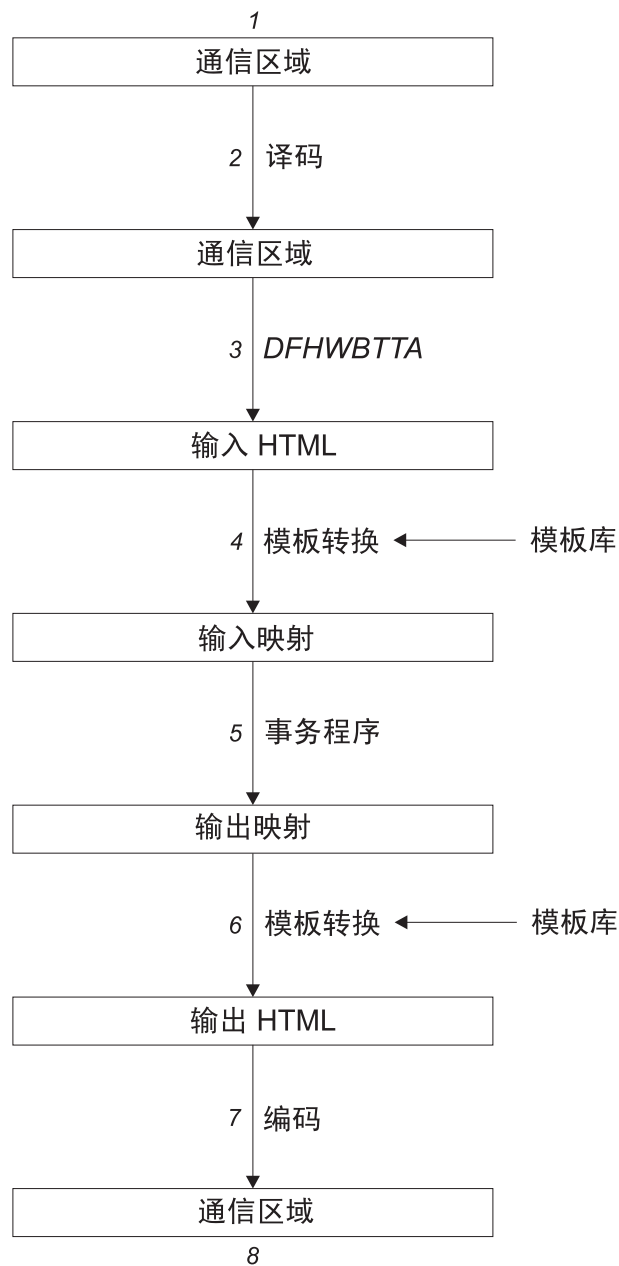


图 21. 继续面向终端的事务 - 数据流

此图显示了当 CICS 业务逻辑接口处理请求时的数据流。

1. CICS 业务逻辑接口的调用者提供包含要处理的请求的通信区域。该通信区域的内容必须是后续进程可以接受的代码页。通常这意味着它们必须是 EBCDIC。
2. 转换器的译码函数为 DFHWBTTA 构造通信区域。
3. 由于这不是对话或者伪会话的第一个事务，因此请求包括了 HTML 格式，它与事务程序希望接收的映射相对应。DFHWBTTA 抽取表单数据，以使它可通过 RECEIVE MAP 命令用于事务程序。
4. 进入表单的输入数据转换成一个 BMS 映射。这个转换使用 DOCTEMPLATE 定义中的模板。

5. 事务程序使用 RECEIVE MAP 命令接收数据。然后它构造输出映射，并使用 SEND MAP 命令将它发送给请求者。
6. 映射及其数据内容转换为 HTML 格式。这个转换使用 DOCTEMPLATE 定义中的模板。
7. 转换器的编码函数使用转换器进程的 HTML 输出来构造要返回给调用者的通信区域。
8. CICS 业务逻辑接口返回到它的调用者，现在该调用者就可以使用通信区域的内容了。

偏移方式和指针方式

可以用两种方式调用 CICS 业务逻辑接口：

偏移方式

在偏移方式中，有单个存储区域（在图 22 中为存储区 1），它包含 DFHWBBLI 通信区域和 CICS 应用程序区域。DFHWBBLI 通信区域中的字段 `wbbl_indata_offset` 包含了以存储区域开始为起点的应用程序通信区域的偏移量。这个存储区域可以存储的最大数据量为 32K 字节。

在偏移方式中，转换器程序不能更改 `DECODE_DATA_PTR` 或 `ENCODE_DATA_PTR` 的值。

指针方式

在指针方式中，有两个独立的存储区域：一个区域（图 22 中的存储区域 1）包含 DFHWBBLI 的通信区域，而另一个区域（存储区域 2）包含 CICS 应用程序的区域。DFHWBBLI 通信区域中的字段 `wbbl_indata_ptr` 包含了应用程序通信区域的地址。

在指针方式中，转换器程序可以更改 `DECODE_DATA_PTR` 或 `ENCODE_DATA_PTR` 的值。

图 22 中说明了这两种方式。

位移方式



指针方式



图 22. CICS 业务逻辑接口中的偏移方式和指针方式

调用 CICS 业务逻辑接口时，必须指定方式：

- `wbbl_mode` 设置为“O”时，表示偏移方式
- `wbbl_mode` 设置为“P”时，表示指针方式

在转换器程序中，可以测试 `decode_volatile` 或 `encode_volatile` 来确定表示的方式：

- 0 表示偏移方式
- 1 表示指针方式

当调用 CICS 业务逻辑接口时，来自以下任何源的所有请求都使用偏移方式：

- 使用 IBM HTTP Server 的 Web 客户机。
- 使用本地网关功能的 Java 应用程序。
- DCE RPC 客户机。
- 使用 CICS Transaction Gateway 的 Web 客户机。

代码页转换和 CICS 业务逻辑接口

CICS 业务逻辑接口不执行代码页转换；您传递到业务应用程序的数据以及返回的数据都在应用程序接口使用的代码页中。然而，EXEC CICS WEB 应用程序编程命令允许您指定客户机代码页，而且此数据在应用程序编程接口中自己转换。因此，当您使用这些命令时，代码页转换是在应用程序和 CICS 业务逻辑接口之间执行的。通过接口传递的数据在 EXEC CICS WEB 命令的 CHARACTERSET 选项（或其同义词 CLNTCODEPAGE）中指定的。

配置 CICS 业务逻辑接口

必须设置 WEBDELAY 系统初始化参数，如第 42 页的『为 CICS Web support 指定系统初始化参数』中所描述。

如果没有使用程序的自动安装，必须定义 CICS 业务逻辑接口的调用者使用的所有用户可替代程序（转换器）。如果使用了程序的自动安装，就不需要定义转换器。所有的转换器对于正在操作 CICS 业务逻辑接口的系统必须是本地的。

第 4 部分 附录

附录 A. HTML 编码字符集

本参考列出了支持的 IANA 注册字符集名称（在 HTTP 头中指定为 charset=），以及 IBM CCSID 等价值。

所有这些值对于以下命令中的代码页转换选项都是有效的：

- WEB RECEIVE（客户机）
- WEB RECEIVE（服务器）
- WEB SEND（客户机）
- WEB SEND（服务器）
- WEB CONVERSE
- DOCUMENT RETRIEVE
- WEB READ FORMFIELD
- WEB STARTBROWSE FORMFIELD

表 8. 编码字符集

语言	编码字符集	IANA 字符集	IBM CCSID
阿尔巴尼亚语	ISO/IEC 8859-1	iso-8859-1	819
阿拉伯语	ISO/IEC 8859-6	iso-8859-6	1089
保加利亚语	Windows [®] 1251	windows-1251	1251
白俄罗斯语	Windows 1251	windows-1251	1251
加泰罗尼亚语	ISO/IEC 8859-1	iso-8859-1	819
简体中文	GB	gb2312	1381 或 5477
繁体中文	Big 5	big5	950
克罗地亚语	ISO/IEC 8859-2	iso-8859-2	912
捷克语	ISO/IEC 8859-2	iso-8859-2	912
丹麦语	ISO/IEC 8859-1	iso-8859-1	819
荷兰语	ISO/IEC 8859-1	iso-8859-1	819
英语	ISO/IEC 8859-1	iso-8859-1	819
爱沙尼亚语	ISO/IEC 8859-1	iso-8859-1	819
芬兰语	ISO/IEC 8859-1	iso-8859-1	819
法语	ISO/IEC 8859-1	iso-8859-1	819
德语	ISO/IEC 8859-1	iso-8859-1	819
希腊语	ISO/IEC 8859-7	iso-8859-7	813
希伯来语	ISO/IEC 8859-8	iso-8859-8	916
匈牙利语	ISO/IEC 8859-2	iso-8859-2	912
意大利语	ISO/IEC 8859-1	iso-8859-1	819
日语	Shift JIS	x-sjis 或 shift-jis	943（943 的子集 932 也是有效的）
	EUC Japanese	euc-jp	5050（EUC）
韩国语	EUC Korean	euc-kr	970（for AIX [®] 或 Unix）

表 8. 编码字符集 (续)

语言	编码字符集	IANA 字符集	IBM CCSID
拉托维亚语	Windows 1257	windows-1257	1257
立陶宛语	Windows 1257	windows-1257	1257
马其顿语	Windows 1257	windows-1257	1251
挪威语	ISO/IEC 8859-1	iso-8859-1	819
波兰语	ISO/IEC 8859-2	iso-8859-2	912
葡萄牙语	ISO/IEC 8859-1	iso-8859-1	819
罗马尼亚语	ISO/IEC 8859-2	iso-8859-2	912
俄语	Windows 1251	windows-1251	1251
塞尔维亚语 (西里尔语)	Windows 1251	windows-1251	1251
塞尔维亚语 (拉丁 2)	Windows 1250	windows-1250	1250
斯洛伐克语	ISO/IEC 8859-2	iso-8859-2	912
斯洛文尼亚语	ISO/IEC 8859-2	iso-8859-2	912
西班牙语	ISO/IEC 8859-1	iso-8859-1	819
瑞典语	ISO/IEC 8859-1	iso-8859-1	819
土耳其语	ISO/IEC 8859-9	iso-8859-9	920
乌克兰语	Windows 1251	windows-1251	1251
Unicode	UCS-2	iso-10646-ucs-2	1200 (不断增加的) 或 13488 (不变的)
Unicode	UTF-16	utf-16	1200
Unicode	UTF-16 大尾数法	utf-16be	1201
Unicode	UTF-16 小尾数法	utf-16le	1202
Unicode	UTF-8	utf-8	1208

附录 B. CICS Web support 的 HTTP 头参考

在 CICS Web support 中，从 CICS 发出消息后，某些 HTTP 头由 CICS 自动提供，而某些 HTTP 头可以由用户添加消息发送到 CICS 后，CICS 将对某些 HTTP 头执行响应操作，而用户应用程序可以对其他头执行响应操作。本参考描述 CICS Web support 如何处理 HTTP 头。

标准 HTTP 头在 HTTP/1.1 规范 (RFC 2616) 和 HTTP/1.0 规范 (RFC 1945) 中描述。存在很多可能的 HTTP 头，包括扩展头，它们不是 HTTP 协议规范的一部分。要获取更完整的列表，应该查询您所遵从的 HTTP 规范。第 8 页的『HTTP 协议』有更多有关 HTTP 规范的信息。

本主题说明 CICS Web support 中 HTTP 头的一般用法，以及 CICS Web support 为特定头执行的操作。要获取有关您应该如何使用 HTTP 头的详细指导和需求（例如，头值的正确格式和应该使用每个头的上下文），请检查您正在遵从的 HTTP 规范。

CICS 所接收的消息的 HTTP 头

- CICS 接收 HTTP 请求或响应后，某些 HTTP 头用于确定 CICS Web support 执行的操作。第 186 页的表 9 说明了 CICS 为 HTTP 请求上的头执行的操作。第 187 页的表 10 说明了 CICS 为 HTTP 响应上的头执行的操作。CICS 不使用其他头，而对它们执行何种适当的响应操作则由用户应用程序决定。
- 使用 WEB READ HTTPHEADER 命令和 HTTP 头浏览命令，为消息接收的所有头（不管它们是否已由 CICS 使用）供用户应用程序进行检查。CICS 不提醒用户应用程序消息上存在任何特殊头。忽略应用程序不需要或不理解的任何头。
- CICS 已经处理 HTTP/1.1 规范中的 MUST 级别需求，这些需求与服务器或客户机在接收消息时必须执行的操作有关。因为这个原因，所以你可以接收并使用请求或响应，而无需检查头。然而，您将可能需要检查头以获取一些信息，这些信息与您将来与 Web 客户机或服务器通信所执行的操作有关。

从 CICS 发出的消息上的 HTTP 头

- 在从 CICS 发出的 HTTP/1.1 版本的 HTTP 请求或响应上，CICS 自动提供通常应该为基本消息所写的、以使该消息遵从 HTTP/1.1 规范的密钥头。在版本为 HTTP/1.0 的 HTTP 响应上，CICS 自动提供较少量的头。这些头中的某些头由 CICS 为每条消息生成，而某些是由于您在用户应用程序中的 WEB SEND 命令中指定的选项而产生。第 188 页的表 11 和第 188 页的表 12 列出为每个 HTTP 版本所写的头，以及头的源代码。

如果用户应用程序写 CICS 也会生成的头，则 CICS 将根据以下情形处理它：

- 对于作为 HTTP 服务器的 CICS，如果该头适合于响应，则 CICS 不覆盖它，但允许使用应用程序的版本。
- 对于作为 HTTP 客户机的 CICS，如果该头适合于请求，则 CICS 不允许应用程序写它，并将错误响应返回给 WEB WRITE HTTPHEADER 命令。异常是 TE 头；应用程序可以添加此头的更多实例。
- 如果该头通常不适合于消息类型（请求或响应），则 CICS 允许它，如所有用户定义的头一样。如果您的消息遵从您正在遵从的 HTTP 规范，则不应该发生此情形。

- 用户应用程序可以使用 WEB WRITE HTTPHEADER 命令将更多 HTTP 头添加到请求或响应。CICS 允许并传递任何其他 HTTP 头。注意，对于作为 HTTP 服务器的 CICS，如果要提供具有 CICS 文档或 HFS 文件的静态响应，则头只能添加到 CICS 自动提供的响应。
- CICS 不检查用户所写的头的名称或值。您应该确保您的应用程序正在以符合您正在遵从的 HTTP 规范的方式提供内容正确且格式正确的信息。如果您的应用程序正在执行复杂的操作，则请特别小心地检查 HTTP 规范以获取可应用的需求。可能存在一些重要的 (MUST 或 SHOULD 级别) 需求以提供描述这些操作的特定头。例如，如果您正在执行以下操作，则需要特殊 HTTP 头：
 - 正在使用文档或实体标记的修改日期响应或发出条件请求。
 - 根据 Web 客户机的客户机功能或本地语言需求变化响应的内容。
 - 提供响应或发出请求，它涉及某范围的文档，而不是完整的文档。
 - 为响应提供高速缓存控制信息。

在您的响应上使用特定状态码可能还需要特殊 HTTP 头。例如，如果使用状态码 405 (不允许方法)，则必须使用 Allow 头以声明允许哪些方法。第 191 页的附录 C，『CICS Web support 的 HTTP 状态码参考』具有有关使用状态码的更多信息。

Upgrade 头

- 注意，作为 CICS Web support 中的特例，不支持协议升级。这意味着：
 - 对于作为 HTTP 服务器的 CICS，应用程序不可能执行任何操作以响应 Web 客户机发送的 Upgrade 头。
 - 对于作为 HTTP 客户机的 CICS，不能在请求上写 Upgrade 头。
 连接期间，CICS 不支持 HTTP 版本中的交换，且安全层的升级也不受支持。

作为 HTTP 服务器的 CICS: CICS 接收 HTTP 请求时执行操作的头

表 9 说明 CICS 为从 Web 客户机接收的请求上的特定头而执行的操作。

表 9. 作为 HTTP 服务器的 CICS: 对 HTTP 请求头的 CICS 操作。

从 Web 客户机接收的头	CICS 执行的操作，其中的响应将由用户应用程序处理	CICS 执行的操作，其中的响应将由静态文档提供
Authorization	将提供的用户标识和密码传递到 RACF 进行验证，并在这些内容无效的情况下拒绝请求。	关于应用程序生成的响应。
Connection	执行 Web 客户机的请求以使连接在发送响应后关闭。	关于应用程序生成的响应。
Content-Length	CICS 要求在具有消息体的所有入站 HTTP/1.1 消息中都有 Content-Length 头。如果有消息体但未提供头，或头的值不正确，则错误消息或后继消息的套接字接收会产生不可预测的结果。对于具有消息体的 HTTP/1.0 消息，Content-Length 头是可选的。	虽然消息体不用于静态响应的处理，但必须仍然从套接字接收静态响应，因此相同的要求也应用于应用程序生成的响应。
Content-Type	对头进行语法分析，为代码页转换识别介质类型和字符集。	对头进行语法分析，为响应的代码页转换识别字符集。
Expect	将 100-Continue 响应发送到 Web 客户机并等待剩余请求。	关于应用程序生成的响应。

表 9. 作为 HTTP 服务器的 CICS: 对 HTTP 请求头的 CICS 操作。(续)

从 Web 客户机接收的头	CICS 执行的操作, 其中的响应将由用户应用程序处理	CICS 执行的操作, 其中的响应将由静态文档提供
Host	如果此头不存在且客户机版本是 HTTP/1.1, 则将 400 (错误请求) 响应发送到 Web 客户机。	关于应用程序生成的响应。
If-Unmodified-Since	如果头存在, 则总是将 412 (预备条件失败) 响应发送到 Web 客户机, 表明自指定的时间以来已修改响应。(这意味着用户应用程序不必检查此头是否存在。)	文档模板: 关于应用程序生成的响应。 HFS 文件: 按照修改日期。
Trailer	通过 WEB READ HTTPHEADER 命令使独立的尾部头对应用程序可用。	分块的消息不适合静态响应。
Transfer-Encoding	对于“分块”, 接收所有块并组装到单个消息中以传递到应用程序。对于除“分块”外的任何内容, 将 501 (未实施) 响应发送到 Web 客户机。Transfer-Encoding 保留在消息上, 但是它仅供参考。	分块的消息不适合静态响应。
Warning	将警告文本写入 TS 队列 CWBW。如果使用的字符多于 128 个, 则会截断警告文本。	关于应用程序生成的响应。

作为 HTTP 客户机的 CICS: CICS 接收 HTTP 响应时执行操作的头

表 10 说明 CICS 为从服务器接收的响应上的特定头而执行的操作。

表 10. 作为 HTTP 客户机的 CICS: 对 HTTP 响应头的 CICS 操作。

从服务器接收的头	CICS 执行的操作
Connection	执行服务器的请求以使连接在接收响应后关闭。
Content-Length	CICS 要求在具有消息体的所有入站 HTTP/1.1 消息中都有 Content-Length 头。如果有消息体但未提供头, 或头的值不正确, 则错误消息或后继消息的套接字接收会产生不可预测的结果。对于具有消息体的 HTTP/1.0 消息, Content-Length 头是可选的。
Content-Type	对头进行语法分析, 为代码页转换识别介质类型和字符集。
Trailer	通过 WEB READ HTTPHEADER 命令使尾部头对应用程序可用。
Transfer-Encoding	对于“分块”, 接收所有块并组装到单个消息中以传递到应用程序。对于除“分块”外的任何内容, 将 501 (未实施) 响应发送到 Web 客户机。Transfer-Encoding 保留在消息上, 但是它仅供参考。
Warning	将警告文本写入 TS 队列 CWBW。如果使用的字符多于 128 个, 则会截断警告文本。

作为 HTTP 服务器的 CICS: CICS 为 HTTP 响应写的头

第 188 页的表 11 显示了响应来自 Web 客户机的请求时 CICS 所写的头、这些头使用的 HTTP 版本以及 CICS 在头中提供的信息源。

表 11. 作为 HTTP 服务器的 CICS: CICS 为 HTTP 响应写的头

CICS 写的头	HTTP 版本	用户应用程序处理响应的源代码	静态文档提供响应的源代码
Connection	1.0 和 1.1	WEB SEND 命令中的 CLOSESTATUS 选项。如果未指定 close, 且客户机版本是 HTTP/1.0, 则发送 Keep-Alive。如果指定了 close, 则发送 Connection: close, 或如果是 HTTP/1.0 客户机, 则省略 Keep-Alive。	在静态响应中发送 Keep-Alive。
Content-Length (除非使用了分块的传输编码)	1.0 和 1.1	其中响应主体是一缓冲区数据, 长度从 WEB SEND 命令的 FROMLENGTH 选项获取。(CICS 检查您指定的长度。如果长度不正确, 则 CICS 发送响应但省略 Connection: Keep-Alive 头。)其中响应主体是 CICS 文档, 长度由 CICS 计算。	由 CICS 计算。
Content-Type	1.0 和 1.1	WEB SEND 命令中的 MEDIATYPE 选项和响应主体的字符集。(只有在指定 MEDIATYPE 选项时才创建头。)	请求的 URIMAP 资源定义的 MEDIATYPE 属性和响应主体的字符集。
Date	1.0 和 1.1	CICS 生成的当前日期和时间。	CICS 生成的当前日期和时间。
Last-Modified (仅用于静态 HFS 文件)	1.0 和 1.1	不为动态响应提供。应用程序应该在可行之处产生此内容。	对于 HFS 文件: 文件的修改日期。对于文档模板: 不提供。
Server	1.0 和 1.1	预先设置为“IBM_CICS_Transaction_Server/ 3.1.0 (zOS)”。	预先设置为“IBM_CICS_Transaction_Server/ 3.1.0 (zOS)”。
Transfer-Encoding	仅 1.1	WEB SEND 命令上的 CHUNKING 选项。	未使用。
WWW-Authenticate	1.0 和 1.1	TCPIP SERVICE 资源定义的 AUTHENTICATE 属性。	TCPIP SERVICE 资源定义的 AUTHENTICATE 属性。

作为 HTTP 客户机的 CICS: CICS 为 HTTP 请求写的头

表 12显示了应用程序将客户机请求发送到服务器时 CICS 所写的头、这些头使用的 HTTP 版本以及 CICS 在头中提供的信息源。

表 12. 作为 HTTP 客户机的 CICS: CICS 为 HTTP 请求写的头

CICS 写的头	HTTP 版本	源代码
Connection	1.0 和 1.1	WEB SEND 命令中的 CLOSESTATUS 选项。根据服务器的 HTTP 版本选择头值。
Content-Length (除非使用了分块的传输编码)	1.0 和 1.1	WEB SEND 命令中的 FROMLENGTH 选项。(CICS 检查您指定的长度。如果长度不正确, 则 CICS 发送响应但省略 Connection: Keep-Alive 头。)
Content-Type	1.0 和 1.1	WEB SEND 命令中的 MEDIATYPE 选项和响应主体的字符集。(只有在指定 MEDIATYPE 选项时才创建头。)
Date	1.0 和 1.1	CICS 生成的当前日期和时间, 格式为带 GMT 时间的 RFC 1123。
Expect	仅 1.1	WEB SEND 命令中的 ACTION(EXPECT) 选项。仅当您的请求有消息体时必须使用此选项。CICS 不将头发送到 HTTP/1.0 服务器。
Host	1.0 和 1.1	WEB OPEN 命令中的 HOST 选项。

表 12. 作为 HTTP 客户机的 CICS: CICS 为 HTTP 请求写的头 (续)

CICS 写的头	HTTP 版本	源代码
TE	仅 1.1	发送到 HTTP/1.1 服务器时总是由 CICS 添加，以声明接受分块的消息和尾部。（HTTP/1.0 服务器不发送分块的消息。）应用程序可以添加更多 TE 头。
Transfer-Encoding	仅 1.1	序列中发送分块消息的第一个 WEB SEND 命令（该命令中的 CHUNKING 选项表明已分块的 transfer-coding）。Transfer-Encoding 头仅写在第一块消息上。
User-Agent	1.0 和 1.1	预先设置为“IBM_CICS_ Transaction_Server/ 3.1.0 (zOS)”。

附录 C. CICS Web support 的 HTTP 状态码参考

HTTP 状态码由服务器提供给客户机以说明客户机请求的结果。当 CICS 是 HTTP 服务器时，根据不同的情况，CICS Web support 或用户应用程序为每个响应选择适当的状态码。当 CICS 是 HTTP 客户机时，从服务器接收的大多数状态码传递到用户应用程序以供处理。

第 12 页的『状态码和原因短语』说明状态码如何用于 HTTP 响应。

要获取有关状态码的含义和正确使用的完整信息，应该查询您所遵循的 HTTP 规范。第 8 页的『HTTP 协议』有更多有关 HTTP 规范的信息。

此主题提供与 CICS Web support 相关的 HTTP/1.1 状态码的简短摘要。当选择通过 Web 出错程序或直接从用户应用程序发送状态码时，检查您所遵循的 HTTP 规范是很重要的。HTTP 规范提供有关应如何使用状态码的详细指导和要求（如，响应主体的内容应该是什么以及应包括什么 HTTP 头）。

CICS 发送的响应的状态码（当 CICS 是 HTTP 服务器时）

- CICS Web support 在以下情况下生成对 Web 客户机的响应：
 - 当 CICS Web support 在初始处理来自 Web 客户机的请求时检测到问题；例如，如果请求中缺少必需的信息，或者如果信息发送太慢，而且达到接收超时。
 - 已安装的 URIMAP 定义与请求匹配，但禁用 URIMAP 定义或虚拟主机，或者，禁用或找不到静态响应的资源。
 - URIMAP 匹配失败，并且为 TCPIP SERVICE 定义指定的分析器不能处理请求和传递对 Web 出错程序的控制。
 - URIMAP 定义和分析器程序以及转换器程序都无法确定应执行什么应用程序来服务请求。
 - 分析器程序、转换器程序或用户编写的应用程序中发生异常终止。这会确保即使处理已失败也会将响应返回给 Web 客户机。
 - URIMAP 指定重定向响应。

在这些情况中，CICS 选择适当的状态码并创建缺省出错响应。第 192 页的表 13 描述 CICS 为这些目的而使用的状态码。注意，在以下这种情况下，CICS 不生成响应：用户编写的应用程序已成功完成处理并要返回表明错误的响应；例如，客户机指定了资源不支持的方法。在这种情况下，用户编写的应用程序创建响应。

- 对于状态码为 4xx 和 5xx 的 CICS 所生成的响应，通过编辑用户可替代的 Web 出错程序 DFHWBEP 和 DFHWBERX，可以修改发送给 Web 客户机的响应。如果 CICS 生成的响应涉及 1xx、2xx 和 3xx 状态码，则不能修改。Web 出错程序可以更改响应的状态码、原因短语、HTTP 头和消息体。当修改 Web 出错程序时，确保按照您所遵循的 HTTP 规范中的要求选择状态码和响应内容。第 85 页的第 9 章，『Web 出错程序』说明如何编辑 Web 出错程序。
- 对客户机的请求作出响应的用户应用程序需要为响应选择合适的状态码。状态码可将以下消息传送到 Web 客户机：
 - 请求已按照预期的完成。
 - 发生阻碍完成请求的错误。

- 客户机需要执行某些操作以成功完成它的请求。这包括遵循重定向 URL，或修正请求以使它可被服务器接受。

状态码影响响应的其他内容，即，消息体和 HTTP 头。第 66 页的『从作为 HTTP 服务器的 CICS 发送 HTTP 响应』告诉您如何汇编和发送响应，包括状态码和原因短语。

CICS 接收的响应的状态码（当 CICS 是 HTTP 客户机时）

- 当 CICS 是 HTTP 客户机时，CICS Web support 将带有大部分状态码的响应直接传递到用户应用程序以进行处理。少量的状态码由 CICS 处理并且不返回到应用程序。如果状态码传递到应用程序，则表明 CICS 未采取任何操作来响应此代码，并且由应用程序负责检查此代码并采取相应的操作。
- 您应适当地设计您的用户应用程序，当它接收带有表明错误的状态码的消息时，能作出合适的反应。特别地，在以下情况下您应总是检查状态码：
 - 如果您打算现在或将来连接期间向服务器发出相同的请求。
 - 如果您打算使用此连接向服务器发出进一步请求。
 - 如果您的应用程序正在执行任何进一步的处理，而此处理依赖响应中接收的信息。

检查您所遵循的 HTTP 规范以获取有关什么操作较合适的指导。HTTP/1.1 规范不包含 MUST 级别要求，它们要求在接收状态码时，应用程序需进一步操作，但包含某些 SHOULD 要求（如要求遵循重定向）。

作为 HTTP 服务器的 CICS: CICS 向 Web 客户机提供的状态码。

表 13 显示 CICS 向 Web 客户机的请求提供响应时使用的状态码。可以通过修改 Web 出错程序来定制这些响应中的一部分。用户应用程序还可使用此处列出的许多状态码。

某些状态码仅适用于 HTTP/1.1 客户机。CICS 不将这些状态码返回给 HTTP/1.0 客户机。

表 13. 作为 HTTP 服务器的 CICS: 发送给 Web 客户机的、CICS 生成的响应的状态码

所提供的状态码和原因短语	发送到 HTTP/1.0 客户机了吗?	提供此响应的情况	可以在 Web 出错程序中修改吗?
100 继续	否	Web 客户机已发送 Expect 头。	否
200 正常	是	正常响应的传递	否
301 已永久移开	是	URIMAP 定义指定重定向，具有属性 REDIRECTTYPE (PERMANENT)。	否
302 已找到	是	URIMAP 定义指定重定向，具有属性 REDIRECTTYPE (TEMPORARY)。	否
400 错误请求	是	请求中的语法错误（如错误地指定请求行，请求未完成）。或者，未提供 Host 头（仅 HTTP/1.1）。	是
401 基本认证错误	是	基本认证需要用户标识和密码。这由端口的 TCPIPService 定义的安全设置确定。	是
403 已禁止（某些情况：客户机认证错误）	是	基本认证未成功。或者，客户机证书有问题。	是

表 13. 作为 HTTP 服务器的 CICS: 发送给 Web 客户机的、CICS 生成的响应的状态码 (续)

所提供的状态码和原因短语	发送到 HTTP/1.0 客户机了吗?	提供此响应的情况	可以在 Web 出错程序中修改吗?
404 未找到 (某些情况: 未找到程序, 未找到文件)	是	未找到指定对请求作出响应的程序。或者, 未找到 URIMAP 定义中为提供静态响应而指定的资源。或者, 未找到图像文件。	是
408 请求超时	否	已超过请求的接收超时。这由端口的 TCPIP SERVICE 定义中的 SOCKETCLOSE 属性确定。	是
412 预备条件失败	否	请求中已使用 If-Unmodified-Since 头。	是
417 未达到期望的	否	接收到没有值“100-continue”的 Expect 头。	否
500 内部服务器错误	是	涉及处理请求和提供响应的某个程序中的异常终止。	是
501 方法未实施	是	对于此 HTTP 版本, 方法不受 CICS 的支持。(包括受支持但不是关于客户机请求方面的方法, 如引用特定资源的 OPTIONS 请求。)或者, 请求的介质类型是“multipart/byteranges”, 它不受支持。或者, 请求的转换编码不同于“分块的”。(注: 连接由 CICS 关闭。)	是
503 服务不可用	是	存在匹配 URIMAP 定义, 但禁用它, 或禁用它为其中一部分的虚拟主机。或者, 禁用 URIMAP 定义中为提供静态响应而指定的资源。	是
505 不支持的版本	否	HTTP 版本比 1.1 新, 并且方法没有被 CICS 支持的最新版本识别。	是

作为 HTTP 服务器的 CICS: 用户应用程序中的状态码

表 14 显示每个状态码, 描述它与用户应用程序的关联, 并根据 HTTP/1.1 规范中的推荐建议合适的操作。

记住, CICS 不采取可能由这些状态码暗示的任何特定操作, 并且 CICS 通常不根据消息的内容检查它们的有效性。您应该确保状态码正确并且您已采取任何必要的操作。确保检查所遵循的 HTTP 规范以获取更多信息和适用于每个状态码的要求。

表 14. 作为 HTTP 服务器的 CICS: 发送给 Web 客户机的、用户编写的响应的状态码

状态码和通常的原因短语	适用于 HTTP/1.0 客户机吗?	您可能提供此响应的情况	对消息体和 HTTP 头的影响 (状态码适用于用户应用程序的情况)。请参阅 HTTP 规范以获取更多信息。
100 继续	否	不要使用。CICS 处理需要的请求并自己发送 100-Continue 响应。	
101 转换协议	否	不要使用。CICS 不支持 HTTP 版本或安全协议的升级。	
200 正常	是	您已完成请求。正常的响应。	提供正常响应体。
201 已创建	是	您已创建新的资源。(如果还未创建资源则使用 202 已接受)。	需要消息体内容和一个或多个头。
202 已接受	是	您已接受请求但还未处理它, 并且不保证要处理它。	需要消息体内容。

表 14. 作为 HTTP 服务器的 CICS: 发送给 Web 客户机的、用户编写的响应的状态码 (续)

状态码和通常的原因短语	适用于 HTTP/1.0 客户机吗?	您可能提供此响应的情况	对消息体和 HTTP 头的影响 (状态码适用于用户应用程序的情况)。请参阅 HTTP 规范以获取更多信息。
203 非权威信息。	否	不要使用。您提供的头将给出权威信息。	
204 无内容	是	您未发送消息体, 也许因为您只需发送已更新的头。	不允许消息体。
205 重置内容	否	您希望客户机清除发出请求的表单。	不允许消息体。
206 部分内容	否	您支持字节范围请求, 并且此响应满足了请求。	正常响应体。需要一个或多个头。
300 多个选项	是	您能提供资源的多个版本 (例如, 不同语言的文档)。	需要消息体内容和一个或多个头。
301 已永久移开	是	建议不要由用户应用程序发出。可使用 URIMAP 定义中的 LOCATION 和 REDIRECTTYPE 属性管理重定向, 以使 CICS 生成正确的响应而无需调用应用程序。REDIRECTTYPE (PERMANENT) 选择此状态码。	
302 已找到	是	建议不要由用户应用程序发出。当将 URIMAP 定义用于重定向时, REDIRECTTYPE (TEMPORARY) 选择此状态码。	
303 参阅其他内容	否	您希望客户机为另一个给出响应 (特别是有关 POST 请求的结果的响应) 的资源发出 GET 请求。	需要消息体内容和一个或多个头。
304 未修改	是	客户机发出有条件的请求并且您提供的资源未更改。注意, 由应用程序动态构建的响应可能要在每个请求上修改。对于不更改的资源, 考虑使用 URIMAP 定义传递静态响应。	不允许消息体。需要一个或多个头。
305 使用代理	否	您希望客户机通过命名代理来发出它的请求。	需要一个或多个头。
307 临时重定向	否	建议不要由用户应用程序发出。对于 URIMAP 重定向, CICS 使用 302 状态码而不是此状态码。	
400 错误请求	是	客户机的请求包含语法错误或类似的问题, 并且您无法处理它。	需要消息体内容。
401 未授权	是	不要使用。当在 TCPIP SERVICE 定义的安全设置中指定时, 由 CICS 处理基本认证过程。	
403 禁止	是	您拒绝客户机的请求。	需要消息体内容。
404 未找到	是	您没有响应此请求的资源; 或者, 您要拒绝请求而没有说明; 或者, 没有其他相关的状态码。	需要消息体内容。
405 不允许的方法	否	客户机使用了此资源不支持的方法。	需要消息体内容和一个或多个头。

表 14. 作为 HTTP 服务器的 CICS: 发送给 Web 客户机的、用户编写的响应的状态码 (续)

状态码和通常的原因短语	适用于 HTTP/1.0 客户机吗?	您可能提供此响应的情况	对消息体和 HTTP 头的影响 (状态码适用于用户应用程序的情况)。请参阅 HTTP 规范以获取更多信息。
406 不可接受	否	客户机使用 Accept 头发出了有条件的请求, 但您没有满足它们的标准资源版本。注意, 作为使用此状态码的替换方法, 您可发送不满足条件的响应。	需要消息体内容。
407 需要代理认证	否	不要使用。CICS 不担当代理服务器。	
408 请求超时	否	建议不要由用户应用程序发出。应使用 TCPIP SERVICE 定义中的 SOCKETCLOSE 属性指定超时以由 CICS Web support 处理。	
409 冲突	否	已更改资源并且无法将客户机的请求应用到资源, 因为它现在处于停止状态。	需要消息体内容。
410 不可用	否	资源永久地不可用。	需要消息体内容。
411 需要长度	否	不要使用。CICS 需要 HTTP/1.1 请求为成功的套接字接收指定 Content-Length 头。	
412 预备条件失败	否	客户机发出了有条件的请求且条件未满足。	需要消息体内容。
413 请求实体太大	否	建议不要由用户应用程序发出。应使用 TCPIP SERVICE 定义上的 MAXDATALEN 属性指定请求大小限制以由 CICS Web support 处理。	
414 请求 URI 太长	否	客户机的请求 URL 太长而使应用程序无法处理。	需要消息体内容。
415 不支持的介质类型	否	客户机发送的消息体具有您不支持的介质类型或内容编码。	需要消息体内容。
416 无法满足请求的范围	否	客户机使用 Range 头字段 (而不是 If-Range 头字段) 发出请求, 并且尽管您支持字节范围, 此范围也不出现在资源中。	需要消息体内容和一个或多个头。
417 未达到期望的	否	不要使用。CICS 处理期望的请求。	
500 内部服务器错误	是	由于应用程序或系统错误, 您无法处理请求。	需要消息体内容。
501 未实施	是	不支持客户机的请求的方法。在客户机为 HTTP/1.0 或您在使用 USER 协议的情况下才应发出此状态码。对于 HTTP 协议, 初始处理期间, CICS 拒绝任何具有无法识别的方法的请求。如果方法被识别但不适用于这些资源, 则“405 不允许的方法”应该用于 HTTP/1.1 客户机。	需要消息体内容。
502 错误网关	是	不要使用。CICS 不担当代理或网关。	
503 服务不可用	是	除非用户应用程序需要访问另一个临时不可用的应用程序或系统, 否则它可能与使用此状态码的情况无关。	需要消息体内容和一个或多个头。
504 网关超时	否	不要使用。CICS 不担当代理或网关。	

表 14. 作为 HTTP 服务器的 CICS: 发送给 Web 客户机的、用户编写的响应的状态码 (续)

状态码和通常的原因短语	适用于 HTTP/1.0 客户机吗?	您可能提供此响应的情况	对消息体和 HTTP 头的影响 (状态码适用于用户应用程序的情况)。请参阅 HTTP 规范以获取更多信息。
505 不支持的 HTTP 版本	否	不要使用。CICS 将响应的 HTTP 版本与客户机的请求的 HTTP 版本相匹配。	

作为 HTTP 客户机的 CICS: 处理有关服务器的响应的状态码

表 15 显示您可能在来自服务器的响应上接收的状态码，并根据 HTTP/1.1 规范中的推荐建议合适的操作。WEB RECEIVE 命令返回状态码和状态文本。记住，服务器可能已更改 HTTP 规范中建议的原因短语的文本。

确保检查所遵循的 HTTP 规范以获取更多信息和适用于每个状态码的要求。

表 15. 作为 HTTP 客户机的 CICS: 处理有关响应的状态码

状态码和可能的原因短语	服务器为何要发送此状态码?	建议的用户应用程序操作
100 继续	在 WEB SEND 命令中使用了 ACTION(EXPECT) 选项，并且服务器接受全部消息发送。	CICS 通过发送消息体处理此响应。用户应用程序将不接收此状态码。
101 转换协议	不应使用。CICS Web support 不支持协议升级。	用户应用程序不应接收此状态码。
200 正常	请求成功。正常的响应。	按计划继续处理响应。
201 已创建	您已请求创建资源并且已完成创建。	按计划继续处理响应。
202 已接受	服务器接受您的请求但还未执行处理。	按计划继续处理响应，但注意，不一定已提交您所作的任何更改，并且可能从不提交。
203 非权威信息。	与消息体相关的头与服务器上的头不完全匹配。	按计划继续处理响应。
204 无内容	没有响应的消息体。	按计划继续处理响应，但注意，没有要接收的主体。
205 重置内容	服务器希望您清除导致发送请求的表单。	清除任何您用于发出请求的表单字段。
206 部分内容	您使用 Range 头字段发出请求并且它已成功。	按计划继续处理响应。
300 多个选项	资源的不同版本可用。	从提供的信息中选择您首选的版本，并发出新的请求。可能有 Location 头，它包含服务器的首选项的 URL。
301 已永久移开	已将资源永久移动到新的位置。	对服务器提供的 URL 发出新请求（可能在 Location 头中），并将此 URL 用于将来所有的请求。
302 已找到	已将资源临时移动到新的位置。	对服务器提供的 URL 发出新请求（可能在 Location 头中），但不将此 URL 用于将来的请求。
303 参阅其他内容	服务器希望您为另一个给出响应（特别是有关 POST 请求的结果的响应）的资源发出 GET 请求。	使用 GET 方法对服务器提供的 URL（可能在 Location 头中）发出新请求。
304 未修改	您发出了有条件的请求并且资源未更改。	参阅响应的现有存储版本以获取信息，但不要将它作为当前信息显示给用户，这是因为 CICS 不支持高速缓存。

表 15. 作为 HTTP 客户机的 CICS: 处理有关响应的状态码 (续)

状态码和可能的原因短语	服务器为何要发送此状态码?	建议的用户应用程序操作
305 使用代理	服务器希望您将指定的代理用于您的请求。	使用服务器提供的 URL (在 Location 头) 发出新请求。
307 临时重定向	关于“302 已找到”。	关于“302 已找到”。
400 错误请求	您的请求的语法不正确。	检查请求, 进行更改并再试一次。
401 未授权	服务器需要权限; 或者, 已拒绝您提供的权限。	请参阅第 110 页的『作为 HTTP 客户机的 CICS: 认证和标识』。
403 禁止	服务器拒绝您的请求。	不要重复请求。消息体可能包含有关为何拒绝请求的信息。
404 未找到	服务器未找到请求的 URL。	检查是否如您希望的指定请求。因为这种情况可能是暂时的, 所以考虑稍后再试一次。
405 不允许的方法	您指定了此资源不支持的方法。	读响应中的 Allow 头以获取受支持的方法的列表, 并在需要时使用这些方法中的某个方法发出新请求。
406 不可接受	您使用 Accept 头发出了请求, 但服务器没有满足您的标准的资源版本。	检查消息体以获取有关服务器具有的资源的信息, 并在需要时对其中某个资源发出新请求。
407 需要代理认证	代理服务器需要权限; 或者, 已拒绝您提供的权限。	请参阅第 110 页的『作为 HTTP 客户机的 CICS: 认证和标识』。
408 请求超时	服务器将不再等待您来完成您的请求。	需要时重复请求。检查您的应用程序是否未花较长的时间来汇编和发送消息。
409 冲突	已更改资源并且无法将您的请求应用到资源, 因为它现在处于停止状态。	检查消息体以获取有关冲突原因的信息, 并在需要时根据此信息发出新请求。
410 不可用	资源永久地不可用。	将来不要重复请求。
411 需要长度	服务器要求您提供 Content-Length 头。	除非您使用 TCPIP SERVICE 定义上的 USER 协议, 否则 CICS 通常提供该头。如果是这种情况, 则自己写头并发出新请求。
412 预备条件失败	您发出了有条件的请求且条件未满足。	按计划继续处理, 注意, 未应用您的请求中指定的任何操作。
413 请求实体太大	您的消息体太大使得服务器无法处理。	读 Retry-After 头以查看这种情况是否是暂时的。您可以等待, 或缩短消息体的长度并再试一次。您可能需要打开新连接。
414 请求 URI 太长	您的请求 URL 太长使得服务器无法处理。	检查请求并再试一次, 或放弃请求。
415 不支持的介质类型	您发送了一个消息体, 而它具有服务器 (对于此资源) 不支持的介质类型或内容编码。	检查您已指定的介质类型, 并且如果出错, 则改正并重复请求。
416 无法满足请求的范围	您使用 Range 头字段发出请求, 但此范围未出现在资源中。	读 Content-Range 头以查看资源的实际长度, 并在需要时用适当的字节范围重复请求。
417 未达到期望的	在 WEB SEND 命令中使用了 ACTION(EXPECT) 选项, 但服务器不接受全部消息发送。	您可以在不带 ACTION (EXPECT) 选项的情况下重复同一请求, 但这可能会再次失败。检查是否正确指定请求, 并且如果出错, 则改正并重复请求。
500 内部服务器错误	由于意外的错误, 服务器无法处理请求。	因为这种情况可能是暂时的, 所以考虑稍后再尝试请求一次。
501 未实施	服务器不支持此请求方法。	不要重复请求。

表 15. 作为 HTTP 客户机的 CICS: 处理有关响应的状态码 (续)

状态码和可能的原因短语	服务器为何要发送此状态码?	建议的用户应用程序操作
502 错误网关	您的请求已通过代理或网关, 而它已接收到来自另一个服务器的无效响应。	因为这种情况可能是暂时的, 所以考虑稍后再尝试请求一次, 或许可能的话, 避开此代理或网关。
503 服务不可用	服务器暂时无法处理请求。	读 <code>Retry-After</code> 头以查看这种情况是否是暂时的, 如果是, 则过一会儿再试一次。
504 网关超时	您的请求已通过代理或网关, 而它未接收到来自另一个服务器的及时响应。	需要时重复请求, 或许可能的话, 避开此代理或网关。
505 不支持的 HTTP 版本	不应使用。CICS Web support 发送版本为 HTTP/1.1 的客户机请求。	用户应用程序不应接收此状态码。

附录 D. CICS Web support 的 HTTP 方法参考

HTTP 请求包含方法，它是个关键字，说明了客户机希望服务器对请求中包含的资料执行的操作。CICS Web support 实施 HTTP/1.1 规范定义的所有标准请求方法和早期 CICS 发行版中接受的某些其他方法。

要获取对方法的正确使用和对它们的正确响应操作的详细指导，并且要获取有关可应用需求的信息，请总是查询您正在遵从的 HTTP 规范。

- 对于从 HTTP/1.1 Web 客户机（当 CICS 是 HTTP 服务器时）接收的请求，会接受 HTTP/1.1 规范定义的标准方法。这些方法是 GET、HEAD、POST、PUT、TRACE、OPTIONS 和 DELETE。
- 对于从 HTTP/1.0 以及更低版本的 Web 客户机（当 CICS 是 HTTP 服务器时）接收的请求，会接受 HTTP/1.0 规范定义的方法和某些其他方法：
 - 由 HTTP/1.0 规范定义的方法是 GET、HEAD 和 POST。
 - 入站到 CICS 的 HTTP/1.0 请求上接受的其他方法是 PUT、DELETE、LINK、UNLINK 和 REQUEUE。
- 对于作为 HTTP 客户机的 CICS 发出的请求：
 - 可以使用 HTTP/1.1 规范定义的标准方法。这些方法是 GET、HEAD、POST、PUT、TRACE、OPTIONS 和 DELETE。
 - LINK、UNLINK 和 REQUEUE 方法不支持此目的。
 - 提供的请求版本总是 HTTP/1.1。
 - 某些 HTTP/1.0 服务器可以接受不在 HTTP/1.0 规范中定义的方法。HTTP/1.0 服务器应该对它无法接受的方法返回状态码“501 未实施”。
- 消息体适合某些请求方法而不适合其他请求方法。
 - 对于作为 HTTP 服务器的 CICS，应该知道某些客户机（特别是用户所写的客户机）可能对不适合消息体的方法发送消息体，您可以选择处理或忽略这一情况。
 - 对于作为 HTTP 客户机的 CICS，CICS 禁止对不适合消息体的方法发送消息体，而要求对适合的方法发送消息体。
- 当 CICS 是 HTTP 服务器时，对于从 Web 客户机接收的请求，CICS Web support 根据方法和客户机的 HTTP 版本，采取一些操作对方法做出响应。
 - 具有最多方法的请求直接传递到应用程序以供处理。
 - CICS 为 OPTIONS 和 TRACE 方法自动返回合适的响应，而不调用用户应用程序。
 - 如果方法未在请求的 HTTP 版本实施，则 CICS 对 Web 客户机返回一个错误响应，而不调用用户应用程序。
- 除了 HTTP 规范中定义的标准请求方法外，某些服务器可能还实施非标准请求方法（称为扩展方法）。
 - 对于作为 HTTP 服务器的 CICS，CICS Web support 不接受 HTTP 协议上有非标准方法的请求。（在 CICS Transaction Server for z/OS V3R1 前，这些请求作为非 HTTP 接受和处理。）如果需要接收具有非标准方法的请求，则可以用用户定义的协议（TCPIPService 定义上的 USER 选项）执行此操作，其中不执行 HTTP 验收检查。

- 对于作为 HTTP 客户机的 CICS，不能在 EXEC CICS WEB API 命令中使用非标准方法。

本参考中的表列出了可能使用每种方法的情况。要获取有关本参考中提及的方法的更详细指导，请查询您正在遵从的 HTTP 规范。

作为 HTTP 服务器的 CICS: 从 Web 客户机接收的处理请求方法

表 16 显示 CICS 对请求方法执行的操作，以及对用户应用程序建议的操作。要获取详细指导和任何相关需求，检查您正在遵从的 HTTP 规范很重要。

表 16. 作为 HTTP 服务器的 CICS: 从 Web 客户机接收的请求方法

方法	HTTP/1.0 客户机的 CICS 操作	HTTP/1.1 客户机的 CICS 操作	消息体是否适合请求?	用户应用程序的适合操作
GET (资源的请求)	接受。请求传递到应用程序。	接受。请求传递到应用程序。	否	将资源发送到 Web 客户机，或发送说明为何无法执行此操作的错误响应。
HEAD (响应头的请求)	接受。请求传递到应用程序。	接受。请求传递到应用程序。	否	将资源发送到 Web 客户机，就好象对同一资源的 GET 请求做出响应一样。CICS 除去响应主体，只保留头。
POST (发送输入数据)	接受。请求传递到应用程序。	接受。请求传递到应用程序。	是	对方法的支持是可选的。抽出数据 (可能是表单字段)，处理它并将响应发送到 Web 客户机。也可用于更改或创建资源，在这种情况下，如同对 PUT 请求进行处理一样。
PUT (发送新的项)	接受。请求传递到应用程序。	接受。请求传递到应用程序。	是	对方法的支持是可选的。如果请求有效，则按需使用消息内容创建具有指定 URL 的资源，或用消息内容替换现有资源。将应答发送到 Web 客户机。HTTP/1.1 规范对正确的操作有详细需求。 提示: 此请求类型可能不适合于您的 CICS Web support 实施。需要的话，它可以通过为指定的 URL 创建 URIMAP 定义并存储要作为静态响应提供的资源来实现。
TRACE (请参阅请求的路径和最终状态)	拒绝，具有状态码“501 未实施”。未调用用户应用程序。	接受。CICS 响应。未调用用户应用程序。	否	未传递到用户应用程序。CICS 返回的响应包含具有原始头和它需要的任何头 (例如，via 头字段) 的请求。
OPTIONS (对于服务器相关信息的请求)	拒绝，具有状态码“501 未实施”。未调用用户应用程序。	接受 OPTIONS * (拒绝其他，具有 501)。CICS 响应。未调用用户应用程序。	当前不打算为它定义。	未传递到用户应用程序。CICS 返回具有基本信息 (HTTP 版本和服务器软件描述) 的响应。
DELETE (删除资源)	接受。请求传递到应用程序。	接受。请求传递到应用程序。	否	对方法的支持是可选的。如果请求有效，则删除现有资源，并将应答发送到 Web 客户机。
LINK、UNLINK、REQUEUE	接受。请求传递到应用程序。	拒绝，具有状态码“501 未实施”。未调用用户应用程序。	未定义	未在标准 HTTP 规范中描述。以您在先前 CICS 发行版中的相同方式处理。不要在新的 CICS Web support 应用程序中使用。

作为 HTTP 客户机的 CICS: 使用对服务器请求的方法。

表 17 列出了 CICS API 对于 HTTP 客户机请求支持的请求方法，并总结了这些方法的正确使用。要获取正确使用每种方法的指导，以及应用到使用该方法的 HTTP 客户机的任何需求，请检查您正在遵从的 HTTP 规范。

表 17. 作为 HTTP 客户机的 CICS: 发送到服务器的请求方法。

方法	是否发送到 HTTP/1.0 服务器?	是否发送到 HTTP/1.1 服务器?	请求中是否有消息体?	目的
GET (资源的请求)	是	是	否	从服务器获取资源。
HEAD (响应头的请求)	是	是	否	从服务器获取资源的头。允许您检查资源的特征、状态或大小，而不必获取整个主体。
POST (发送输入数据)	是	是	是	将数据发送到服务器。例如，表单数据可能以这种方式发送。支持此方法不需要服务器。
PUT (发送新的项)	可能不受服务器支持。	是	是	创建或修改服务器上的资源。您的请求的 URL 是资源在服务器上具有的 URL。请求可用于更新现有项或新建一个项。支持此方法不需要服务器。
TRACE (请参阅请求的路径和最终状态)	可能不受服务器支持。	是	否	获取显示请求最终状态的响应及其带到服务器的路径 (在 via 头字段中显示)。您可以查看正在使用什么代理服务器以处理您的请求。支持此方法不需要服务器。
OPTIONS (对于服务器相关信息的请求)	可能不受服务器支持。	是	允许，但是当不打算为它定义。	获取有关服务器的信息。通过将 * (星号) 指定为请求路径，将请求应用到整个服务器，或指定完整请求路径以获取有关该资源的信息。支持此方法不需要服务器。 提示: 发出 WEB OPEN 命令时，CICS 自动执行此请求，而服务器的 HTTP 版本将在 WEB OPEN 命令中返回。
DELETE (删除资源)	可能不受服务器支持。	是	否	删除服务器上的资源。请求 URL 是要删除的项的 URL。支持此方法不需要服务器。
通常是 LINK、UNLINK、REQUEUE 和扩展方法	不允许。返回的 INVREQ 响应和未发送的请求。	不允许。返回的 INVREQ 响应和未发送的请求。	未定义	对于作为 HTTP 客户机的 CICS，在 WEB API 上不可用。

附录 E. 分析器程序的参考信息

本部分提供分析器程序的参考信息，包括输入和输出参数，以及响应和原因码。

分析器程序参数摘要

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

分析器程序的参数和常量名（已转换成支持的不同编程语言的适当格式）在作为 CICS 的一部分提供的文件中定义。

语言	参数文件	常量文件
汇编语言	DFHWBTDD	DFHWBUCD
C	DFHWBTDH	DFHWBUCH
COBOL	DFHWBTDO	DFHWBUCO
PL/I	DFHWBTDL	DFHWBUCL

这些文件给出了有关通信区域中字段数据类型的特定于语言的信息。如果使用这些文件，则必须在“转换程序”步骤中指定 XOPTS(NOLINKAGE)；否则会导致编译失败。

下表中，以缩写形式给出了参数的名称：表中每个名称都必须以 **wbra_** 为前缀给出参数的名称。

表 18. 分析器程序的参数

输入 wbra_	输入输出 wbra_	输出 wbra_
client_ip_address	alias_tranid	application_style
content_length	converter_program	alias_termid
eyecatcher	server_program	characterset
function	user_data_length	commarea
hostname_length	userid	dfhcnv_key
hostname_ptr		hostcodepage
http_version_length		reason
http_version_ptr		response
method_ptr		unescape
method_length		user_token
querystring_length		
querystring_ptr		
request_header_length		
request_header_ptr		
request_type		
resource_escaped_ptr		
resource_length		
resource_ptr		
server_ip_address		
urimap		
user_data_ptr		

分析器程序的参数

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

wbra_alias_tranid

(输入和输出)

长度为 4 的字符串。用于覆盖此请求的剩余处理的别名事务的事务标识。如果涉及 URIMAP 定义, 则它包含 TRANSACTION 属性的值。如果不设置该字段, 或将其设置为空白, 则使用 CWBA。

wbra_alias_termid

(仅输出)

长度为 4 的字符串。用于别名事务的 START 请求的终端标识, 该别名事务将覆盖此请求的剩余处理。

wbra_charset

(仅输出)

客户机用于请求的实体主体的 IANA 字符集的名称。此信息用于请求和响应的实体主体的代码页转换。如果请求不是 HTTP 请求, 则此字符集用于转换整个请求和响应。还必须提供 wbra_hostcodepage。

wbra_client_ip_address

(仅输入)

客户机的 32 位 IP 地址。

wbra_commarea

(仅输出)

用于表明响应需要 CICS TS V3 之前版本的兼容性处理的标志, 该响应使用不支持 Web 的应用程序和转换器程序。此标志意味着 Web 客户机接收的响应等同于它在 CICS TS V3 之前版本接收的响应。

wbra_content_length

(仅输入)

如所接收的数据中 Content-Length HTTP 头指定的, 实体主体长度的 32 位二进制表示。

wbra_converter_program

(输入和输出)

长度为 8 的字符串。用于处理请求的转换器程序的名称。如果涉及 URIMAP 定义, 则它包含 CONVERTER 属性的值。如果未在输出上设置此字段, 则不调用转换器程序。

wbra_dfhcnv_key

(仅输出)

长度为 8 的字符串。DFHCNV 表中转换模板的名称, 该模板用于请求和响应的实体主体的代码页转换。如果请求不是 HTTP 请求, 则此模板用于转换整个请求和响应。

CICS 将此字段初始化为较大的值。如果使用此字段指定转换模板, 则必须在 DFHCNV 表中定义所选的名称, 如第 43 页的『迁移代码页转换表 (DFHCNV) 中的条目』中所描述。另外, 您可以设置 **wbra_hostcodepage** 和

wbra_characterstet 字段以指定用于代码页转换的代码页对。如果将 **wbra_dfhcnv_key** 设置为空或空的，且不设置 **wbra_hostcodepage** 和 **wbra_characterstet**，则禁止代码页转换。

wbra_eyecatcher

(仅输入)

长度为 8 的字符串。它的值为“>analyze”。

wbra_function

(仅输入)

表明正在调用分析器程序的代码。其值为 1。

wbra_hostcodepage

(仅输出)

适合正在处理请求的应用程序的主机代码页 (IBM EBCDIC 代码页) 的名称。此信息用于请求和响应的实体主体的代码页转换。如果请求不是 HTTP 请求，则此代码页用于转换整个请求和响应。还必须提供 **wbra_characterstet**。

wbra_hostname_length

(仅输入)

HTTP 请求中指定的主机名的长度 (以字节为单位)。如果未指定主机名，则表示没有定义该值。

wbra_hostname_ptr

(仅输入)

指向主机名的指针，该主机名在客户机发送的 HTTP 请求中指定。如果将绝对 URI 用于请求，则主机名从该 URI 获取。否则主机名在请求的 Host 头中指定。对于 HTTP/1.1 请求，必需主机名，因此此参数总是传递到分析器。对于 HTTP/1.0 请求，可能不提供主机名，在这种情况下不定义值。

wbra_http_version_length

(仅输入)

对于 HTTP 请求，标识客户机请求的 HTTP 版本的字符串长度 (以字节为单位)。如果请求不是 HTTP 请求，其值为 0。

wbra_http_version_ptr

(仅输入)

对于 HTTP 请求，指向标识客户机请求的 HTTP 版本的字符串的指针。如果请求不是 HTTP 请求，则不定义该值。

wbra_method_length

(仅输入)

对于 HTTP 请求，标识 HTTP 请求中所指定的方法的字符串的字节长度。如果请求不是 HTTP 请求，其值为 0。

wbra_method_ptr

(仅输入)

对于 HTTP 请求，指向 HTTP 请求中所指定的方法的指针。如果请求不是 HTTP 请求，则不定义该值。

wbra_querystring_length

(仅输入)

HTTP 请求上指定的查询字符串的长度 (以字节为单位)。如果未发送查询字符串, 则表示没有定义该值。

wbra_querystring_ptr

(仅输入)

指向查询字符串的指针, 该查询字符串在客户机发送的 HTTP 请求中指定。如果未发送查询字符串, 则表示没有定义该值。

wbra_reason

(仅输出)

分析器程序返回的原因码。请参阅第 208 页的『响应和原因码』。

wbra_request_header_length

(仅输入)

对于 HTTP 请求, HTTP 请求中第一个 HTTP 头的长度。如果请求不是 HTTP 请求, 其值为 0。

wbra_request_header_ptr

(仅输入)

对于 HTTP 请求, 指向 HTTP 请求中第一个 HTTP 头的指针。在请求缓冲区域中, 其他 HTTP 头紧跟该头。如果请求不是 HTTP 请求, 则不定义该值。

wbra_request_type

(仅输入)

如果这是 HTTP 请求, 则值是 WBRA_REQUEST_HTTP。如果请求不是 HTTP 请求, 则该值为 WBRA_REQUEST_NON_HTTP。

wbra_resource_escaped_ptr

(仅输入)

对于 HTTP 请求, 指向尚未转义 (即, 仍处于其转义形式) 的请求的 HTTP 头副本的指针。

wbra_resource_length

(仅输入)

对于 HTTP 请求, URL 的路径部分的长度 (以字节为单位)。如果请求不是 HTTP 请求, 其值为 0。

wbra_resource_ptr

(仅输入)

对于 HTTP 请求, 指向 URL 的路径部分的指针。如果涉及 URIMAP 定义, 则它包含 PATH 属性的值。如果请求不是 HTTP 请求, 则不定义该值。

wbra_response

(仅输出)

分析器程序产生的响应值。请参阅第 208 页的『响应和原因码』。

wbra_server_ip_address

(仅输入)

作为 HTTP 服务器的 CICS 使用的 32 位 IP 地址。

wbra_server_program

(输入和输出)

长度为 8 的字符串。用于处理请求的 CICS 应用程序的名称。如果涉及 URIMAP 定义，则它包含 PROGRAM 属性的值。程序名传递到 **wbra_converter_program** 中指定的任何转换器程序。如果没有设置该字段，则所传的值为 null。程序名必须在此处或由转换器程序设置，否则将不调用 CICS 应用程序。

wbra_unescape

(仅输出)

- 要指定以其未转义形式传递到 CICS 应用程序的数据，请将此参数设置为 **WBRA_UNESCAPE_REQUIRED**
- 要指定以转义形式将数据传递给 CICS 应用程序，请将该参数设置为 **WBRA_UNESCAPE_NOT_REQUIRED**。这是缺省值。

如果分析器已经将数据转换为转义形式，也应该将参数设置为 **WBRA_UNESCAPE_NOT_REQUIRED**。

wbra_urimap

(仅输入)

请求的处理路径中涉及的任何匹配 URIMAP 定义的名称。如果此字段非空，则 CICS 提供的缺省分析器 DFHWBADX 返回，而不处理 URL 的路径部分。

wbra_user_data_length

(输入和输出)

15 位整数，表示 HTTP 请求中实体主体的长度。如果请求是非 HTTP 请求，其值为请求的长度。传递给分析器的长度包括任何可能定界实体主体结尾的尾部回车换行符 (CRLF)。如果分析器减少了实体主体的长度，则空字符 'X'00' 将替换新的冗余字节。修改的值传递到字段 **wbbl_user_data_length** 中的 CICS 业务逻辑接口，并传递到字段 **decode_user_data_length** 中的转换器程序。

wbra_user_data_ptr

(仅输入)

对于 HTTP 请求，指向 HTTP 请求中实体主体的指针。如果请求不是 HTTP 请求，该指针指向该请求。

wbra_user_token

(仅输出)

作为 **decode_user_token** 传递到转换器程序的 64 位令牌。如果没有设置该字段，所传的空。如果没有此请求的转换器程序，则忽略该值。

wbra_userid

(输入和输出)

长度为 8 的字符串。输入时，它是由客户机提供的用户标识 (使用基本认证或客户机证书认证)，或者如果涉及 URIMAP 定义，则是 USERID 属性 (如果指定) 的值。输出时，它是用于别名事务的用户标识，它可以是提供的用户标识或分析器程序选择的用户标识。如果输出时此字段是空的或为空，则使用 CICS 缺省用户标识。

响应和原因码

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

分析器程序必须在 **wbra_response** 中返回以下某个值：

符号值	数值	说明
URP_OK	0	启动别名事务。
URP_EXCEPTION	4	未启动别名事务。Web 附加处理写异常跟踪项（跟踪点 4510），并发出消息（DFHWB0523）。 如果请求是 HTTP 请求，则将错误响应发送到 Web 客户机。缺省状态码是 400（错误请求），可以使用用户可替换 Web 出错程序 DFHWBEP 来配置它。 如果请求不是 HTTP 请求，则不发送响应，且关闭套接字。
URP_INVALID	8	未启动别名事务。服务器控制器写异常跟踪项（跟踪点 4510），并发出消息（DFHWB0523）。 如果请求是 HTTP 请求，则将错误响应发送到 Web 客户机。缺省状态码是 400（错误请求），可以使用用户可替换 Web 出错程序 DFHWBEP 来配置它。 如果请求不是 HTTP 请求，则不发送响应，且关闭套接字。
URP_DISASTER	12	未启动别名事务。CICS 写异常跟踪项（跟踪点 4510），并发出消息（DFHWB0523）。 如果请求是 HTTP 请求，则将错误响应发送到 Web 客户机。缺省状态码是 400（错误请求），可以使用用户可替换 Web 出错程序 DFHWBEP 来配置它。 如果请求不是 HTTP 请求，则不发送响应，且关闭套接字。

如果在 **wbra_response** 中返回任何其他值，则服务器控制器写异常跟踪项（跟踪点 4510），并且发出消息（DFHWB0523）。如果请求是 HTTP 请求，则将具有状态码 400（错误请求）的消息发送给 Web 客户机。如果请求不是 HTTP 请求，则不发送响应，且关闭套接字。

在出错情况下，可以在 **wbra_reason** 中提供 32 位的原因码来提供进一步的信息。CICS Web support 不对分析器程序返回的原因码执行任何操作，但是用户可替换 Web 出错程序 DFHWBEP 可使用它决定如何修改缺省响应。原因码在任何调用分析器程序所导致的跟踪项中输出，并在消息 DFHWB0523 中。

附录 F. 转换器程序的参考信息

本部分提供:

- 转换器程序的**解码**函数的参考信息
- 转换器程序的**编码**函数的参考信息

传递到转换器程序的通信区域中的参数和常量名, 转换为不同编程语言支持的适当格式, 并在作为 CICS Web support 的一部分提供的文件中定义。下表列出了各种语言的文件。

语言	参数文件	常量文件
汇编语言	DFHWBCDD	DFHWBUCD
C	DFHWBCDH	DFHWBUCH
COBOL	DFHWBCDO	DFHWBUCO
PL/I	DFHWBCDL	DFHWBUCL

这些文件给出了有关通信区域中字段数据类型的特定于语言的信息。如果使用这些文件, 则必须在“转换程序”步骤中指定 XOPTS(NOLINKAGE); 否则会导致编译失败。

解码

参数摘要

下表中, 参数的名称以缩写的格式给出: 表中每个名称都必须以 **decode_** 作为前缀来给出参数名。

表 19. 解码的参数

输入 decode_	输入输出 decode_	输出 decode_
client_address	data_ptr	output_data_len
client_address_string	input_data_len	reason
eyecatcher	server_program	response
entry_count	user_token	
function		
http_version_length		
http_version_ptr		
method_length		
method_ptr		
request_header_length		
request_header_ptr		
resource_length		
resource_ptr		
user_data_length		
user_data_ptr		
volatile		

函数

如果分析器或 CICS 业务逻辑接口的调用者为请求指定了转换器名称，则在调用服务于该请求的 CICS 应用程序之前先调用**解码**。

参数

decode_client_address

(仅输入)

客户机的 32 位 IP 地址。

decode_client_address_string

(仅输入)

以点十进制格式表示的客户机的 IP 地址。

decode_data_ptr

(输入和输出)

输入时，它是指向来自客户机请求（由分析器修改）的指针，或者，如果这个调用是**编码**转换器函数返回的循环，则是指向 **encode_data_ptr** 的响应数据的指针。

输出时，它是指向要传递到 CICS 应用程序的通信区域的指针。当 **decode_volatile** 的值为 **0** 时，不要修改这个参数。

decode_entry_count

(仅输入)

一个计数，记录为当前 Web 请求输入的**解码**转换器的次数。

decode_eyecatcher

(仅输入)

一个长度为 8 的字符串。它的**解码**的值为“>decode”。

decode_function

(仅输入)

一个半字长代码，被设置为常数值 **URP_DECODE**，表示**解码**正在被调用。

decode_http_version_length

(仅输入)

标识客户机支持的 HTTP 版本的字符串的长度（以字节为单位）。如果请求不是 HTTP 请求，或者 **decode_entry_count** 大于 1，则值为 0。

decode_http_version_ptr

(仅输入)

一个指针，指向标识客户机支持的 HTTP 版本的字符串。如果分析器修改了请求的这一部分，则更改在此可见。如果 **decode_http_version_length** 为 0，则表示没有定义该值。

decode_input_data_len

(输入和输出)

在输入时，这是由 **decode_data_ptr** 所指向的请求数据的长度（以字节为单位）。

用于 CICS 应用程序中 LINK 命令的 DATALENGTH 选项的值。如果输出没有设置，则缺省值为 32K。

decode_method_length

(仅输入)

HTTP 请求中指定的方法的长度 (以字节为单位)。如果请求不是 HTTP 请求，或者 **decode_entry_count** 大于 1，则值为 0。

decode_method_ptr

(仅输入)

指向 HTTP 请求中指定的方法的指针。如果分析器修改了请求的这一部分，则更改在此可见。如果 **decode_method_length** 为 0，则表示没有定义该值。

decode_output_data_len

(仅输出)

用于 CICS 应用程序中 LINK 命令的 LENGTH 选项的值。如果输出没有设置，则缺省值为 32K。

decode_reason

(仅输出)

原因码 - 请参阅第 212 页的『响应和原因码』。

decode_request_header_length

(仅输入)

HTTP 请求中第一个 HTTP 头的长度。如果请求不是 HTTP 请求，或者 **decode_entry_count** 大于 1，则值为 0。

decode_request_header_ptr

(仅输入)

指向 HTTP 请求中第一个 HTTP 头的指针。如果分析器修改了请求的这一部分，则更改在此可见。如果 **decode_request_header_length** 为 0，则表示没有定义该值。

decode_resource_length

(仅输入)

HTTP 请求中 URL 的路径部分长度 (以字节为单位)。如果请求不是 HTTP 请求，或者 **decode_entry_count** 大于 1，则值为 0。

decode_resource_ptr

(仅输入)

指向 HTTP 请求中 URL 的路径部分的指针。如果分析器修改了请求的这一部分，则更改在此可见。如果 **decode_resource_length** 为 0，则表示没有定义该值。

decode_response

(仅输出)

响应 - 请参阅第 212 页的『响应和原因码』。

decode_server_program

(输入和输出)

长度为 8 的字符串。输入时，它是 **wbra_server_program** 中的分析器提供的值，或者是由 CICS 业务逻辑接口的调用者提供的值。输出时，它是对请求提供服务的 CICS 应用程序的名称。应用程序的名称必须在此处或在分析器中设置，否则将不会调用任何应用程序。

decode_user_data_length

(仅输入)

HTTP 请求的实体主体的长度 (以字节为单位)。如果分析器修改了该值，则其在此可见。如果请求中没有实体主体，则长度为 0。如果请求不是 HTTP 请求，则该值表示请求的长度。如果 **decode_entry_count** 大于 1，则该值为零。

decode_user_data_ptr

(仅输入)

指向 HTTP 请求的任意一个实体主体的指针。如果分析器修改了请求的这一部分，则更改在此可见。如果请求中没有实体主体，则指针为 0。如果请求不是 HTTP 请求，则该指针与 **decode_data_ptr** 值相同。如果 **decode_entry_count** 大于 1，则表示没有定义该值。

decode_user_token

(输入和输出)

一个 64 位的令牌。输入时，它是由作为 **wbra_user_token** 的分析器提供的用户令牌，或者是 CICS 业务逻辑接口的调用者提供的用户令牌。输出时，令牌作为 **encode_user_token** 传到解码。

decode_version

(输入)

一个单字符的参数列表版本标识，随参数列表布局的改变而改变。它的值可以是表明 CICS TS 1.3 之前版本的参数列表的二进制零 (X'00')，或表明 CICS TS 1.3 版本参数列表的字符零 (X'F0')。

decode_volatile

(输入)

一个单字符的代码，表示 **decode_data_ptr** 指向的数据区域能否被替换。可能的值是：

- 0** 该区域是另一个通信区域的一部分，且不能被替换。
- 1** **decode_data_ptr** 指向的存储器可以被释放并被不同大小的工作区替换。

响应和原因码

您必须在 **decode_response** 返回下列值之一：

符号值	数值	说明
URP_OK	0	处理继续进行。如果请求了 CICS 应用程序，则执行它。否则，继续处理解码器的编码函数。

符号值	数值	说明
URP_EXCEPTION	4	<p>采取的操作取决于原因码:</p> <p>CICS Web support</p> <p>1 (URP_SECURITY_FAILURE) CICS 写异常跟踪条目 (跟踪点 455A), 并发出消息 (DFHWB0121)。如果请求是 HTTP 请求, 则状态码 403 发送到 Web 客户机。如果请求不是 HTTP 请求, 则不发送响应, 并且关闭 Communications Server 套接字。</p> <p>2 (URP_CORRUPT_CLIENT_DATA) CICS 写异常跟踪条目 (跟踪点 4559), 并发出消息 (DFHWB0121)。如果请求是 HTTP 请求, 则将状态码 400 发送到 Web 客户机。如果请求不是 HTTP 请求, 则不发送任何响应, 并且 MVS 套接字的 TCP/IP 关闭。</p> <p>任何其他值 CICS 写异常跟踪条目 (跟踪点 455B), 并发出消息 (DFHWB0121)。如果请求是 HTTP 请求, 则将状态码 501 发送到 Web 客户机。如果请求不是 HTTP 请求, 则不发送响应, 并且关闭 Communications Server 套接字。</p> <p>CICS 业务逻辑接口</p> <p>2 (URP_CORRUPT_CLIENT_DATA) CICS 业务逻辑接口写异常跟踪条目 (跟踪点 4556), 发出消息 (DFHWB0120), 并将响应 400 返回给它的调用者。</p> <p>任何其他值 CICS 写异常跟踪条目 (跟踪点 455B), 发出消息 (DFHWB0121), 并将响应 501 返回给它的调用者。</p> <p>CICS 应用程序和解码器的编码函数都不执行。</p>
URP_INVALID	8	<p>CICS 应用程序和解码器的编码函数都不执行。</p> <ul style="list-style-type: none"> • CICS Web support • CICS 写异常跟踪条目 (跟踪点 455C), 并发出消息 (DFHWB0121)。如果请求是 HTTP 请求, 则将状态码 501 发送到 Web 客户机。如果请求不是 HTTP 请求, 则不发送响应, 并且关闭 Communications Server 套接字。 • CICS 业务逻辑接口 • CICS 写异常跟踪条目 (跟踪点 455C), 发出消息 (DFHWB0121), 并将响应 501 返回给它的调用者。

符号值	数值	说明
URP_DISASTER	12	<p>CICS 应用程序和解码器的编码函数都不执行。</p> <ul style="list-style-type: none"> • CICS Web support • CICS 写异常跟踪条目（跟踪点 455D），并发送消息（DFHWB0121）。如果请求是 HTTP 请求，则将状态码 501 发送到 Web 客户机。如果请求不是 HTTP 请求，则不发送响应，并且关闭 Communications Server 套接字。 • CICS 业务逻辑接口 • CICS 写异常跟踪条目（跟踪点 455D），发送消息（DFHWB0121），并将响应 501 返回给它的调用者。
任何其他值		<p>CICS 应用程序和解码器的编码函数都不执行。</p> <ul style="list-style-type: none"> • CICS Web support • CICS 写异常跟踪条目（跟踪点 455E），并发送消息（DFHWB0121）。如果请求是 HTTP 请求，则将状态码 500 发送到 Web 客户机。如果请求不是 HTTP 请求，则不发送响应，并且关闭 Communications Server 套接字。 • CICS 业务逻辑接口 • CICS 写异常跟踪条目（跟踪点 455E），发送消息（DFHWB0121），并将响应 501 返回给它的调用者。

您可以在 **decode_reason** 中提供一个 32 位原因码以便在出错的情况下提供更多信息。CICS Web support 和 CICS 业务逻辑接口都不对解码返回的原因码执行任何操作，以上在 URP_EXCEPTION 中提到的原因码除外。原因码在解码调用产生的所有跟踪项中输出。

编码

参数摘要

下表中，参数的名称以缩写的格式给出：表中每个名称都必须以 **encode_** 作为前缀来给出参数名。

表 20. 编码的参数

输入 encode_	输入输出 encode_	输出 encode_
eyecatcher entry_count function input_data_len user_token	data_ptr	reason response

函数

如果分析器或 CICS 业务逻辑接口的调用者为请求指定了转换器程序名，则**编码**在 CICS 应用程序结束后调用。它从通信区域的内容构造响应。

参数

encode_data_ptr

（输入和输出）

输入时，指向通信区域的指针由 CICS 应用程序返回。如果没有调用应用程序，则它是指向**译码**创建的通信区域的指针。

输出时，它是指向包含送往客户机的响应的缓冲区的指针。必须确保指针指向有效位置，否则结果将不可预测。缓冲区域必须是双字对齐。头 4 个字节必须是 32 位的无符号数，用来指定缓冲区域的长度。（在 COBOL 中，将其指定为 PIC 9(8) COMP。）缓冲区域的余下部分则为响应。从 CICS 业务逻辑接口（它以位移方式调用）调用转换器时，不要将此字段用作输出。

encode_entry_count

（仅输入）

一个计数，记录为当前 Web 请求输入**编码**转换器的次数。

encode_eyecatcher

（仅输入）

一个长度为 8 的字符串。它的**编码**的值为“>encode”。

encode_function

（仅输入）

一个半字长代码，被设置为常数值 **URP_ENCODE**，表示正在调用**编码**。

encode_input_data_len

（仅输入）

通信区域的长度由**译码**在 **decode_output_data_len** 中指定。

编码原因

（仅输出）

原因码 - 请参阅第 216 页的『响应和原因码』。

encode_response

（仅输出）

响应 - 请参阅第 216 页的『响应和原因码』。

encode_user_token

（仅输入）

解码作为 **decode_user_token** 输出的 64 位令牌。

encode_version

（输入）

一个单字符的参数列表版本标识，随参数列表布局的改变而改变。它的值可以是表明 CICS TS 1.3 之前版本的参数列表的二进制零（X'00'），或表明 CICS TS 1.3 版本参数列表的字符零（X'F0'）。

encode_volatile

(输入)

一个单字符代码，表示 **encode_data_ptr** 所指的数据区域能否被替换。可能的值是：

- 0** 该区域是另一个通信区域的一部分，且不能被替换。
- 1** **encode_data_ptr** 指向的存储器可以被释放并被不同大小的工作区域替换。

响应和原因码

必须在 **encode_response** 中返回下列值之一：

符号值	数值	说明
URP_OK	0	缓冲区域中由 encode_data_ptr 指示的响应送往客户机。
URP_DISASTER	12	CICS Web support <ul style="list-style-type: none">• CICS 写异常跟踪条目（跟踪点 455D），并发出消息（DFHWB0122）。如果请求是 HTTP 请求，则状态码 501 发送到 Web 客户机。如果请求不是 HTTP 请求，则不发送响应，并且关闭 Communications Server 套接字。 CICS 业务逻辑接口 <ul style="list-style-type: none">• CICS 写异常跟踪条目（跟踪点 455D），发出消息（DFHWB0122），并将响应 501 返回给它的调用者。
URP_OK_LOOP	16	CICS Web 接口返回到 解码 函数的开头。存储在 encode_user_token 中的值被复制到 decode_user_token 为解码转换器函数所用。
任何其他值		CICS Web support <ul style="list-style-type: none">• CICS 写异常跟踪条目（跟踪点 455E），并发出消息（DFHWB0122）。如果请求是 HTTP 请求，则状态码 501 发送到 Web 客户机。如果请求不是 HTTP 请求，则不发送响应，并且关闭 Communications Server 套接字。 CICS 业务逻辑接口 <ul style="list-style-type: none">• CICS 写异常跟踪条目（跟踪点 455E），发出消息（DFHWB0122），并将响应 501 返回给它的调用者。

您需要在 **encode_reason** 中提供一个 32 位的原因码以便在出错情况下提供更多信息。CICS Web support 和 CICS 业务逻辑接口都不对**编码**返回的原因码执行任何操作。原因码在**编码**调用产生的所有跟踪项中输出。

附录 G. DFHWBBLI CICS 业务逻辑接口的参考信息

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

业务逻辑接口允许调用者指定在 CICS 应用程序之前和之后要执行的显示逻辑。它有两种操作方式：

- 指针方式：**解码**的输入数据位于同业务逻辑接口通信区域分开分配的存储器中。通信区域包含指向**解码**的输入数据的指针（**wbbl_data_ptr**）。当对业务逻辑接口的调用结束时，来自**编码**的输出位于与业务逻辑接口的通信区域分开分配的存储器中，并且通信区域包含指向来自**编码**的输出的指针（**wbbl_outdata_ptr**）。
- 位移方式：**解码**的输入数据是业务逻辑接口通信区域的一部分。通信区域包含了解码的输入数据的偏移量（**wbbl_data_offset**）。当调用业务逻辑接口结束时，**编码**的输出是业务逻辑接口通信区域的一部分，并且通信区域包含来自**解码**的输出的偏移量（**wbbl_outdata_offset**）。

业务逻辑接口的调用者使用 **wbbl_mode** 来指示要使用的操作方式。

有关为业务逻辑接口编写转换器的信息，请参阅第 102 页的『写转换器』。

参数摘要

警告： 此主题包含产品敏感的编程接口和相关的指导信息。

参数和常量名（已转换成支持的不同编程语言的适当格式）在作为 CICS Web support 的一部分的文件中定义。各种语言的文件如下所示：

语言	文件
汇编语言	DFHWBBLD
C	DFHWBBLH
COBOL	DFHWBBLO
PL/I	DFHWBBLI

这些文件给出了有关通信区域中字段数据类型的特定于语言的信息。

下表中，以缩写形式给出了参数的名称：表中的每个名称都必须以 **wbbl_** 为前缀给出参数的名称。

表 21. 业务逻辑接口参数

输入 wbbi_	输入输出 wbbi_	输出 wbbi_
client_address client_address_length client_address_string converter_program_name eyecatcher header_length header_offset http_version_length http_version_offset indata_length indata_offset indata_ptr length method_length method_offset mode prolog_size resource_length resource_offset server_program_name ssl_keysize status_size user_token user_data_length vector_size version	(无)	outdata_length outdata_offset outdata_ptr

通过称为 DHWBBLI 的兼容接口来支持为使用 CICS 业务逻辑接口的先前版本 (DFHWBA1) 而编写的程序。

参数

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

在将输入插入通信区域中前，必须将它清为二进制零。

wbbi_eyecatcher

(仅输入)

必须设置为字符串 **>DFHWBBLIPARMS** 的 14 个字符的字段。

wbbi_client_address

(仅输入)

必须设置为客户机的二进制 IP 地址的全字 32 位 (如果已知) 的字段。

wbbi_client_address_length

(仅输入)

必须设置为 **wbbi_client_address_string** 的长度的 1 个字节的二进制字段。

wbbl_client_address_string

(仅输入)

一个最多十五个字符的字符串，该字符串是点分十进制表示 `wbbl_client_address`，并在右端填充二进制零。

wbbl_converter_program_name

(仅输入)

用于转换器 `DECODE` 和 `ENCODE` 函数的程序的 8 个字符的名称。

wbbl_header_length

(仅输入)

必须包含与该请求相关联的 HTTP 头长度的全字二进制数。

wbbl_header_offset

(仅输入)

必须包含与该请求相关联的 HTTP 头的偏移量（从请求数据的开始部分起算）的全字二进制数。

wbbl_http_version_length

(仅输入)

必须包含用于处理请求的 HTTP 协议的版本长度的全字二进制数。

wbbl_http_version_offset

(仅输入)

必须包含用于处理请求的 HTTP 协议的版本的偏移量的全字二进制数。

wbbl_indata_length

(仅输入)

必须设置为由 `wbbl_indata_ptr` 或 `wbbl_indata_offset` 定位的数据的长度的全字二进制数。如果分析器修改该值，则它在这里是可见的。如果请求不是 HTTP 请求，请不要设置该字段。

wbbl_indata_offset

(仅输入)

如果 `wbbl_mode` 是“O”，那么该字段是传输给应用程序的 HTTP 请求数据的偏移量（从参数列表的开始部分起）。

wbbl_indata_ptr

(仅输入)

如果 `wbbl_mode` 是“P”，那么它就是传输给应用程序的 HTTP 请求数据的地址。

wbbl_length

(仅输入)

必须设置成为 BLI 参数列表的总长度的半字二进制数。

wbbl_method_length

(仅输入)

必须包含用于处理请求的 HTTP 方法的长度的全字二进制数。该方法应该是下列之一：`GET`、`POST`、`HEAD`、`PUT`、`DELETE`、`LINK`、`UNLINK` 或 `REQUEUE`。

wbbl_method_offset

(仅输入)

必须包含用于处理请求的 HTTP 方法的偏移量 (从请求数据的开始部分起算) 的全字二进制数。该方法应该是下列之一:

GET、POST、HEAD、PUT、DELETE、LINK、UNLINK 或 REQUEUE。

wbbl_mode

(仅输入)

指示 wbbl_indata 和 wbbl_outdata 寻址方式的单个字符。必须将其设置为“P”以指示这些值都是指针, 或设置为“O”以指示这些值是从参数列表的开始部分起算的偏移量。

wbbl_outdata_length

(仅输入)

全字二进制字段, DFHWBBLI 在该字段中返回由 wbbl_outdata_ptr 或 wbbl_outdata_offset 定位的响应数据的长度。

wbbl_outdata_offset

(仅输入)

如果 wbbl_mode 是“O”, 那么它是全字, DFHWBBLI 在其中返回来自应用程序的响应数据的偏移量 (从参数列表的开始部分起)。该地址不一定要和 wbbl_indata_offset 相同。

wbbl_outdata_ptr

(仅输入)

如果 wbbl_mode 是“P”, 那么它是全字地址, 在该地址中, DFHWBBLI 返回来自应用程序的响应数据的地址。该地址不一定要和 wbbl_indata_ptr 相同。

wbbl_prolog_size

(仅输入)

必须设置为 56 (即, wbbl_prolog 子结构的长度) 的半字二进制数。

wbbl_resource_length

(仅输入)

一个全字二进制数, 它必须包含被请求的 URI 资源 (即, URL 中第一个“/”字符开始的 URL 非网络部分) 的长度。

wbbl_resource_offset

(仅输入)

一个全字二进制数, 它必须包含正请求的 URI 资源 (即, URL 中第一个“/”字符开始的 URL 非网络部分) 的偏移量 (从请求数据的开始部分起算)。

wbbl_response

(仅输入)

DFHWBBLI 在其中返回响应代码的全字二进制字段。

wbbl_server_program_name

(仅输入)

用于处理请求并产生响应的 CICS 应用程序的名称 (8 个字符)。

wbbl_ssl_keysize

(仅输入)

加密密钥的大小，如果使用安全套接字层，则该大小在 SSL 握手期间协商确定。如果不使用 SSL，则其值为零。

wbbl_status_size

(仅输入)

必须设置为 `wbbl_status` 子结构的长度的 1 个字节的二进制字段。

wbbl_user_data_length

(仅输入)

必须设置为实体主体长度的全字二进制数。如果分析器修改该值，则它在这里是可见的。如果请求不是 HTTP 请求，请不要设置该字段。

wbbl_user_token

(仅输入)

一个八字符字段，DFHWBBLI 的调用者使用该字段可以传输标识客户机当前对话状态的数据。通常将它设置为 URL 的 **query-string** 部分（即，问号（?）以后的任何数据）的前 8 个字符。

wbbl_vector_size

(仅输入)

必须设置为 64（即，`wbbl_vector` 子结构的长度）的半字二进制数。

wbbl_version

(仅输入)

表明当前使用的 BLI 参数列表的版本的半字二进制数。应该使用常数值 `wbbl_current_version` 来设置该值。

响应

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

wbbl_response 中返回下列值之一。这些值同将要发送给 HTTP 客户机的 HTTP 响应相一致。

400 某个转换器函数返回了带有 `URP_CORRUPT_CLIENT_DATA` 原因码的 `URP_EXCEPTION` 响应。业务逻辑接口写异常跟踪项（跟踪点 4556），并发出消息（DFHWB0120）。

403 对 **wbbl_server_program_name** 中指定的程序的 `LINK` 命令接收到 `NOTAUTH` 响应。业务逻辑接口写异常跟踪项（跟踪点 4556），并发出消息（DFHWB0120）。

404 对 **wbbl_server_program_name** 中指定的程序的 `LINK` 命令接收到 `PGMIDERR` 响应。业务逻辑接口写异常跟踪项（跟踪点 4556），并发出消息（DFHWB0120）。

500 将发生以下某种情况:

- 业务逻辑接口检测到异常结束。根据异常结束的程序发出相应的消息。对于在 **wbbl_server_program_name** 中指定的程序，消息为 DFHWB0125。对

于转换器的**编码**函数，消息为 DFHWB0126。对于转换器的**解码**函数，消息为 DFHWB0127。对于任何其他程序，消息为 DFHWB0128。任何情况下，都会写入异常跟踪项（跟踪点 4557）。

- 对 **wbbl_server_program_name** 中指定的程序的 LINK 命令接收到 INVREQ 或 LENGERR 或意外响应。业务逻辑接口写异常跟踪项（跟踪点 4556），并发出消息（DFHWB0120）。

501 将发生以下某种情况:

- **解码**返回了 URP_EXCEPTION 响应，并带有未定义的原因码。业务逻辑接口写异常跟踪项（跟踪点 455B），并发出消息（DFHWB0121）。
- **解码**返回了 URP_INVALID 响应。业务逻辑接口写异常跟踪项（跟踪点 455C），并发出消息（DFHWB0121）。
- **解码**返回了 URP_DISASTER 响应。业务逻辑接口写异常跟踪项（跟踪点 455D），并发出消息（DFHWB0121）。
- **解码**返回了未定义响应。业务逻辑接口写异常跟踪项（跟踪点 455E），并发出消息（DFHWB0121）。
- **编码**返回了 URP_EXCEPTION 响应，并带有未定义的原因码业务逻辑接口写异常跟踪项（跟踪点 455B），并发出消息（DFHWB0122）。
- **编码**返回了 URP_INVALID 响应。业务逻辑接口写异常跟踪项（跟踪点 455C），并发出消息（DFHWB0122）。
- **编码**返回了 URP_DISASTER 响应。业务逻辑接口写异常跟踪项（跟踪点 455D），并发出消息（DFHWB0122）。
- **编码**返回了未定义响应。业务逻辑接口写异常跟踪项（跟踪点 455E），并发出消息（DFHWB0122）。

503 将发生以下某种情况:

- 对 **wbbl_server_program_name** 中指定的程序的 LINK 命令接收到 TERMERR 响应。业务逻辑接口写异常跟踪项（跟踪点 4555），并发出消息（DFHWB0120）。
- 对 **wbbl_server_program_name** 中指定的程序的 LINK 命令接收到 SYSIDERR 或 ROLLEDBACK 响应。业务逻辑接口写异常跟踪项（跟踪点 4556），并发出消息（DFHWB0120）。

附录 H. Web 出错程序 DFHWBEP 的参考信息

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

下表中列出了通信区域中的参数和常量名称，这些参数和常量传递到 Web 出错程序 DFHWBEP，并转换成所支持的编程语言的适当格式。

语言	参数文件
汇编程序	DFHWBEPD
C	DFHWBEPH
COBOL	DFHWBEPO
PL/I	DFHWBEPL

参数

警告: 此主题包含产品敏感的编程接口和相关的指导信息。

除了 **wbep_response_ptr** 和 **wbep_response_len** 可以输入和输出,所有 DFHWBEP 参数都是仅输入的; 而 **wbep_suppress_abend** 是仅输出的。

wbep_abend_code

(仅输入)

与此异常关联的 8 个字符异常终止代码。

wbep_activity

(仅输入)

发生错误时进行的处理类型。0 表明服务器处理，而 2 表明管道处理。

wbep_analyzer_reason

(仅输入)

分析器程序返回的原因码 - 如果调用了分析器程序。

wbep_analyzer_response

(仅输入)

分析器程序返回的响应码 - 如果调用了分析器程序。

wbep_client_address

(仅输入)

客户机的 15 个字符的点分十进制 IP 地址。

wbep_client_address_len

(仅输入)

WBEP_CLIENT_ADDRESS 中包含的点分十进制 IP 地址长度。

wbep_converter_program

(仅输入)

用于失败请求的转换器程序名称 - 如果使用了转换器程序。

wbep_converter_reason

(仅输入)

转换器返回的原因码 (如果调用了转换器)。

wbep_converter_response

(仅输入)

转换器返回的响应码 (如果调用了转换器)。

wbep_error_code

(仅输入)

错误代码, 标识检测到的错误。

wbep_eyecatcher

(仅输入)

一个字符字段, 包含了一个有助于诊断的 eyecatcher。DFHWBA 在调用 Web 出错程序以前将它设置成 >wbepca。

wbep_failing_program

(仅输入)

包含发生错误的程序名的八字符字段。

wbep_http_response_code

(仅输入)

CICS 为此错误返回的缺省 HTTP 状态码。可以通过操纵 WBEP_RESPONSE_PTR 所指向的缓冲区中的响应来更改该状态码。

wbep_length

(仅输入)

DFHWBEP 副本的长度。

wbep_message_len

(仅输入)

WBEP_MESSAGE_PTR 定位的消息长度。

wbep_message_number

(仅输入)

与该错误关联的 CICS WB 消息的全字数。

wbep_message_ptr

(仅输入)

指向与此异常关联的 CICS 消息文本的指针。

wbep_response_len

(输入和输出)

与这个异常关联的响应消息文本的全字长度。CICS 仅为 HTTP 请求构造响应; 对于非 HTTP 请求, 此字段为零。它是一个完整的 HTTP 响应, 包括状态行、头和消息体。

wbep_response_ptr

(输入和输出)

指向与这个异常关联的响应消息文本的指针, 仅用于 HTTP 请求。

wbep_server_address

(仅输入)

服务器的 15 个字符的点分十进制 IP 地址。

wbep_server_address_len

(仅输入)

WBEP_SERVER_ADDRESS 中包含的点分十进制 IP 地址长度。

wbep_target_program

(仅输入)

与 Web 请求关联的目标用户应用程序。

wbep_tcpipservice_name

(仅输入)

接收到请求的端口的 TCPIPService 定义名。

wbep_version

(仅输入)

CICS 传递的 DFHWBEP 版本。

wbep_suppress_abend

(仅输出)

一位标志，当设置为 on 时，其抑制异常结束 AWBM。

附录 I. DFHWBCLI Web 客户机接口

DFHWBCLI 是 CICS 提供的实用程序，您可以通过 EXEC CICS LINK 调用它，以提供 Web 客户机服务或出站 HTTP。为了迁移目的在 CICS Transaction Server for z/OS V3R1 中是支持它的。

出于兼容性原因保留了 DFHWBCLI Web 客户机接口的函数。为了获得增强的功能，您可以迁移使用 DFHWBCLI 接口的 HTTP 客户机应用程序，以为客户机请求使用 EXEC CICS WEB API 命令（具有 SESSTOKEN 选项）。注意，一个重要的区别在于，在 EXEC CICS WEB API 中，代理服务器的使用由 WEB OPEN 命令上的用户出口（XWBOPEN）指定，而代理服务器的 URL 由该用户出口提供。第 115 页的『通过作为 HTTP 客户机的 CICS 发出 HTTP 请求』描述了现在如何发出 HTTP 客户机请求。

如果要使用 DFHWBCLI，则必须设置 CICS 以使用名称服务器。请参阅第 41 页的第 4 章，『配置 CICS Web support 基本组件』。

要使用 DFHWBCLI，您必须使用包含下列副本映射其内容的参数列表的通信区域链接到它：

- DFHWBCLD for Assembler
- DFHWBCLO for Cobol
- DFHWBCLL for PL/I
- DFHWBCLH for C

这些参数具有以下含义：

WBCLI_VERSION_NO

一字节二进制数，其指定此参数列表的版本号。它应该设置为符号常量 WBCLI_VERSION_CURRENT 指定的值。

WBCLI_FUNCTION

一字节二进制数，其指定您要 DFHWBCLI 执行的函数。它应该设置为下列值之一：

0 (WBCLI_FUNCTION_CONVERSE)

发送 HTTP 请求到目标服务器，并接收相应的响应

1 (WBCLI_FUNCTION_SEND)

发送 HTTP 请求到目标服务器，并返回控制，而无须等待响应

2 (WBCLI_FUNCTION_RECEIVE)

等待并接收对前一个 SEND 函数所发送 HTTP 请求的响应

3 (WBCLI_FUNCTION_INQUIRE_PROXY)

请求 INITPARM=(DFHWBCLI,'PROXY=http://....') 系统初始化参数中所指定代理服务器的名称

4 (WBCLI_FUNCTION_CLOSE)

关闭先前由 SEND 函数建立的连接，但是无须等待 HTTP 响应

WBCLI_METHOD

一字节二进制数，其指定要在 HTTP 请求中指定的 HTTP 方法。它应该设置为下列值之一：

1 (WBCLI_METHOD_GET)

2 (WBCLI_METHOD_POST)

WBCLI_FLAGS

一字节二进制位串，其可用于指定与 HTTP 请求及其所期待响应关联的选项。二进制位串中的二进制位必须设置为下列值：

1... (WBCLI_OFFSET_MODE)

参数列表中的指针值指定为从参数列表开始处的位移值。这暗示这种指针的所有目标都包含在通信区域中。

.1.. (WBCLI_DOCUMENT)

HTTP 请求主体是由 DOCUMENT CREATE 命令创建的 CICS 文档，并且由 WBCLI_REQUEST_DOCTOKEN 中的文档令牌指定

..1. (WBCLI_USE_PROXY)

HTTP 请求通过在 WBCLI_PROXY_URL_PTR 中指定其 URL 的代理服务器发送

...1 (WBCLI_SET_RESP_BUFFER)

CICS 要获取合适大小的缓冲区，以包含 HTTP 响应主体，并在 WBCLI_REQUEST_BODY_PTR 中返回其地址

注：此地址不构成位移，无论 WBCLI_OFFSET_MODE 的设置是怎样的

.... ..1. (WBCLI_NATIVE_REQUEST_BODY)

应用程序将以其本机格式提供 HTTP 请求主体，而 CICS 不需要将它从 EBCDIC 转换为 ASCII

.... ...1 (WBCLI_NATIVE_RESPONSE_BODY)

应用程序将以其本机格式处理 HTTP 响应主体，而 CICS 不需要将它从 ASCII 转换为 EBCDIC

WBCLI_RESPONSE

半字二进制数，其设置为下列值之一，以表明函数的结果：

0 (WBCLI_RESPONSE_OK)

4 (WBCLI_RESPONSE_EXCEPTION)

8 (WBCLI_RESPONSE_DISASTER)

WBCLI_REASON

半字二进制数，其设置为下列值之一，以限定响应代码：

1 (WBCLI_REASON_INVALID_URL)

WBCLI_URL_PTR 找到的 URL 的格式无效，或者名称服务器无法解析主机位置

2 (WBCLI_REASON_INVALID_HEADER)

WBCLI_HEADER_PTR 所找到列表中的一个 HTTP 头的格式不正确

3 (WBCLI_REASON_INVALID_DOCUMENT)

WBCLI_REQUEST_DOCTOKEN 中指定的文档令牌未找到有效 CICS 文档

4 (WBCLI_REASON_GETMAIN_ERROR)

当 DFHWBCLI 尝试为其内部工作区之一获取存储器时发生错误

5 (WBCLI_REASON_PROXY_ERROR)

WBCLI_PROXY_URL_PTR 找到的代理服务器无法找到，或返回错误响应

6 (WBCLI_REASON_SOCKET_ERROR)

执行套接字操作时返回意外响应

7 (WBCLI_REASON_HTTP_ERROR)

服务器返回意外 HTTP 响应

8 (WBCLI_REASON_TRANSLATE_ERROR)

当 CICS 在主机代码页和服务器代码页之间转换数据时，返回一条错误。这可能是因为 CICS 不支持必需的转换

9 (WBCLI_REASON_TRUNCATED)

WBCLI_RESPONSE_BODY_LEN 中所指定用户提供响应缓冲区的长度不足，无法包含服务器返回的响应。超出此长度的数据已废弃。

WBCLI_SESSION_TOKEN

不透明的八字节二进制标记，其表示所建立与 HTTP 服务器的连接。它由 SEND 函数设置，而且对于 RECEIVE 和 CLOSE 函数是必需的。其他函数不使用它。

WBCLI_URL_PTR

EBCDIC 字符串的地址，其包含目标 HTTP 服务器的 URL（统一资源定位符）。URL 必须是标准的：即，它必须以'http://'或'https://'开头。

WBCLI_URL_LEN

全字二进制数，其包含 WBCLI_URL_PTR 所找到 URL 的长度。

WBCLI_PROXY_URL_PTR

EBCDIC 字符串的地址，其包含访问防火墙外远程站点所需的代理服务器的 URL（统一资源定位符）。URL 必须是标准的：即，它必须以'http://'开头。要使用代理，您还必须设置 WBCLI_USE_PROXY 标志。

WBCLI_PROXY_URL_LEN

全字二进制数，其包含 WBCLI_PROXY_URL_PTR 所找到 URL 的长度。

WBCLI_HEADER_PTR

要与 HTTP 请求一起发送的 HTTP 头列表的地址。头必须在 EBCDIC 中已编码，以下列格式表示：

```
headername: headervalue$headername: headervalue$ ...
```

其中

headername

是头的名称

headervalue

是头的值

分隔这两个的冒号（:）和空格应该如所示出现；此处显示的“\$”应该由下列一个或多个定界符替代：

回车符（X'0D'）

换行（X'25'）

新行（X'15'）

字段分隔符（X'1E'）

注：发送头时不使用这些定界符：CICS 使用结构上正确的 HTTP 定界符。

您可以按需要在列表中编码足够的头。然而，您不能包含以下头，因为 CICS 将提供它们：

Host

User-Agent
Content-Length
Content-Type

在列表最后一个头后，您无须提供定界符。

WBCLI_HEADER_LEN

全字二进制数，其包含 WBCLI_HEARER_PTR 所找到头列表的长度。

WBCLI_REQUEST_DOCTOKEN

一个 16 字节二进制文档令牌，由 DOCUMENT CREATE 命令创建，它表示用作 HTTP 请求主体的 CICS 文档。您必须通过设置 WBCLI_DOCUMENT 标志表明您正在使用此标记。

WBCLI_REQUEST_BODY_PTR

EBCDIC 字符串的地址，其包含 HTTP 请求主体的整个内容。在未设置 WBCLI_DOCUMENT 标志时使用此参数。

WBCLI_REQUEST_BODY_LEN

全字二进制数，其包含 WBCLI_REQUEST_BODY_PTR 所找到请求主体的长度。

WBCLI_RESPONSE_BODY_PTR

缓冲区的地址，DFHWBCLI 在其中返回来自服务器的 HTTP 响应主体。

- 如果未设置标志 WBCLI_SET_RESP_BUFFER，此地址和 WBCLI_RESPONSE_BODY_LEN 必须由调用者设置。如果此缓冲区不够大，以致无法包含响应主体，则主体被截断。
- 如果设置了标志 WBCLI_SET_RESP_BUFFER，此地址和 WBCLI_RESPONSE_BODY_LEN 被忽略。CICS 获取大小足够包含整个响应的新区缓冲区，并且它的地址也在此字段中返回。此地址不会转换到位移中，无论 WBCLI_OFFSET_MODE 标志的值是什么。

通常，当调用 DFHWBCLI 的任务结束时，CICS 释放此地址的存储器。或者，您可以通过在您的应用程序中发出 EXEC CICS FREEMAIN 命令来较早地释放存储器。在较长时间运行的任务中重复调用 DFHWBCLI 时，建议您这样做，以防止 CICS 变得存储器不够。

WBCLI_RESPONSE_BODY_LEN

全字二进制数，它包含 WBCLI_RESPONSE_BODY_PTR 所定位的响应缓冲区的长度。

- 在输入时，如果未设置 WBCLI_SET_RESP_BUFFER，使用此参数指定用户所提供缓冲区的长度。
- 在输出时，它包含所返回响应的实际长度。

WBCLI_MEDIATYPE

40 字节 EBCDIC 空填充字符串，其包含 HTTP 主体的 IANA 媒体类型（也称为 MIME 类型）。

- 在输入时，使用此参数指定 HTTP 请求主体的媒体类型。此媒体类型将在 HTTP Content-Type 头中发送
- 在输出时，它将包含 HTTP 响应主体的媒体类型，同在 HTTP Content-Type 头中接收到的一样。

WBCLI_CHARSET

40 字节 EBCDIC 字符串，其包含 HTTP 主体的 IANA 字符集。

- 在输入时，如果未设置 `WBCLI_NATIVE_REQUEST_BODY`，则使用此参数指定您要 CICS 将 HTTP 请求主体转换到的字符集名称。您指定的字符集用于限定 HTTP Content-Type 头中的媒体类型。如果您未指定值，假设为缺省值 `iso-8859-1`。
- 在输出时，它将包含 HTTP 响应主体的字符集，同在 HTTP Content-Type 头中接收到的一样。此字符集用于转换 HTTP 响应主体（除非设置了 `WBCLI_NATIVE_RESPONSE_BODY`）。

WBCLI_HOST_CODEPAGE

10 字符 EBCDIC 空填充字符串，其包含您的应用程序所使用 EBCDIC 代码页的名称。它与 `WBCLI_CHARSET` 组合在一起使用，以确定要在 HTTP 文档主体上执行哪种转换（除非 `WBCLI_NATIVE_REQUEST_BODY` 或 `WBCLI_NATIVE_RESPONSE_BODY` 标志阻止转换）。如果省略它，则 CICS 使用代码页 037。

WBCLI_HTTP_STATUS_CODE

三位数字 EBCDIC 字符串，在其中返回 HTTP 状态代码。此代码表明 HTTP 请求是否成功。值 200 通常表明成功，其他任何值表明某些类型的错误。

附录 J. 状态管理样本 DFH\$WBST 和 DFH\$WBSR 的参考信息

两个状态管理样本程序 DFH\$WBST 和 DFH\$WBSR 与 CICS Web support 一起提供。它们允许一个事务保存数据，以便将来通过这个事务本身或其他事务来检索。所保存的数据由第一个事务的状态管理程序所创建的令牌来访问。第一个事务必须把令牌传递给要检索数据的事务。DFH\$WBST 使用 GETMAIN 命令为保存的数据分配存储器。DFH\$WBSR 把数据保存到临时存储器队列，每个队列对应一个令牌，所以，使用合适的临时存储器表定义，就可以从多个 CICS 系统访问数据。本节的剩余部分均衡地应用于两个程序。

状态管理程序提供下列操作：

- 创建新令牌。
- 存储信息并将它与先前创建的令牌关联。
- 检索先前与令牌关联的信息。
- 销毁与令牌关联的信息并使令牌无效。

DFH\$WBST 还除去到期的信息和令牌。您可以周期性地运行这个程序以便清除期满的状态数据：

- 将程序作为 CWBT 事务运行，以便清除所有一小时内未更新的状态数据。
- 将程序作为 CWBP 事务运行，以便清除所有状态数据。

268 字节的通信区域布局如下表所示。在为您所需要的函数设置输入之前，必须将通信区域清空为二进制零。

表 22. 用于状态管理程序的参数

Offset	Length	Type	Value	Notes
0	4	C		Eyecatcher
4	1	C	'C' 'R' 'S' 'D'	Create Retrieve Store Destroy 这是函数码。它是每次调用所需的输入。
5	1	X		返回码。这是来自每次调用的输出。
6	2	X		保留。
8	4	F		令牌。这是来自 Create 调用的输出，以及到每个其他调用的输入。
12	256	C		用户数据。这是到 Create 和 Store 调用的输入，以及来自 Retrieve 调用的输出。它不用于其他调用。

返回码如下：

- 0** 执行了所请求的函数。
 - 如果是 Create 函数，新令牌在偏移量为 8 时可用。
 - 如果是 Retrieve 函数，与输入令牌关联偏移量为 8 的实体主体现在处于偏移量为 12 的实体主体区域。

- 如果是 Store 函数，偏移量为 12 的输入实体主体现在与偏移量为 8 的输入令牌相关联的。先前与该令牌相关联的任何实体主体将被覆盖。
 - 如果是 Destroy 函数，与输入令牌关联偏移量为 8 的数据将被废弃，并且令牌将不再有效。
- 2 偏移量为 4 的函数码是无效的。校正设置通信区域的程序。
 - 3 函数是 Create，但是 GETMAIN 命令给出错误响应。
 - 4 函数是 Retrieve、Store 或者 Destroy，但是偏移量为 8 的输入令牌找不到。原因是输入令牌不是 Create 返回的令牌，或者它已经期满。
 - 5 将内部数据写到临时存储队列时，WRITEQ TS 命令给出错误响应。
 - 7 ASKTIME 命令给出错误响应。
 - 8 从临时存储队列读取内部数据时，READQ TS 给出错误响应。
 - 9 超时处理期间，ASKTIME 命令给出错误响应。
 - 11 函数是 Create，但是 WRITEQ TS 命令给出错误响应。这个返回码只能由 DFH\$WBSR 产生。
 - 12 函数是 Retrieve，但是 READQ TS 命令给出错误响应。这个返回码只能由 DFH\$WBSR 产生。
 - 13 函数是 Store，但是 WRITEQ TS 命令给出错误响应。这个返回码只能由 DFH\$WBSR 产生。
 - 14 函数是 Destroy，但是 DELETEQ TS 命令给出错误响应。这个返回码只能由 DFH\$WBSR 产生。

附录 K. CICS Web 服务器插件

注：出于兼容性原因保留了 CICS WebServer 插件的函数。建议您在新的应用程序中使用 CICS Transaction Gateway。

提供的这个插件支持这样一个传递机制：使用 CICS 业务逻辑接口从 IBM HTTP Server 经外部 CICS 接口 (EXCI) 到 CICS Web support 中。

注：可流经这个接口的最大数据量为 32K 字节。

配置 IBM HTTP Server

注：出于兼容性原因保留了 CICS WebServer 插件的函数。建议您在新的应用程序中使用 CICS Transaction Gateway。

如果 IBM HTTP Server 要使用 CICS 业务逻辑接口来提供其服务的话，则必须更改其中的配置信息。*z/OS HTTP Server Planning, Installing, and Using*, SC34-4826 提供有关配置语句的详细信息。

您可使用以下过程：

1. 必须如下设置 CICS：

- 用 ISC=YES 初始化 CICS 区域。
- 安装 RDO 组 DFHWEB
- 为 EXCI 定义一个常规连接（例如，通过安装样本组 DFH\$EXCI）。
- 确保 IRC 是开的。

2. 将 CICSTS31.CICS.SDFHDLL1 装入库和 CICSTS31.CICS.SDFHEXCI 定义到 RACF 程序控制。RACF 程序控制表明包含该库的卷的卷序列号，且不允许使用另一个卷。如果您稍后将装入库或 CICSTS31.CICS.SDFHEXCI 库移动到另一个卷，则必须将它重新定义到 RACF 程序控制。

3. 将 CICSTS31.CICS.SDFHDLL1 数据集和 CICSTS31.CICS.SDFHEXCI 库添加到 IBM HTTP Server 的 JCL 中的 STEPLIB 并置。

4. 在包含 IBM HTTP Server 的 httpd.conf 文件的目录中使用以下命令：

```
ln -e DFHWBAPI dfhwbapi.so
```

在 STEPLIB 并置中使用此命令时，它建立从 IBM HTTP Server 的主目录到 CICSTS31.CICS.SDFHDLL1 库中成员 DFHWBAPI 中的 DLL dfhwbapi.so 的链接。

5. 在 httpd.conf 文件中添加一个或多个服务伪指令。服务伪指令将最终用户输入的 URL 映射到将满足请求的 CICS 资源。DFHWBAPI 的服务伪指令具有以下格式：

```
Service /sourceurl/* /home/dfhwbapi.so:DFHService/targeturl/*
```

其中的值为：

home 是包含 IBM HTTP Server 的 httpd.conf 文件的目录。

sourceurl

是一个字符串，它选择 DFHWBAPI 要处理的入局 URL。紧跟其后的星号

是一个通配符字符串，表示进入的 URL 的其余字符。*sourceurl* 可以是任何格式，所以象 *applid* 和 *transaction* 这样的细节可以对最终用户隐藏。

targeturl

targeturl 是一串字符串，DFHWBAPI 将使用它确定哪些 CICS 资源会满足用户请求。在替代了通配符后，**targeturl** 的格式必须为：

```
/applid/converter/tran/program/filename
```

其中的值为：

applid 目标 CICS 区域的应用程序标识

converter

在 CICS 区域中使用的转换器程序的名称，或者如果没有要使用的转换器，则名称为 CICS。

tran 要在 CICS 区域中执行的事务。因为该事务是 EXCI 请求的目标事务，因此它不应该是 Web 别名事务 CWBA，而应该是镜像事务，如 CSM3。该事务接收 *targeturl/** 而不是 *sourceurl/**，以此作为入局 URL。

program

要在 CICS 区域中执行的程序的名称。

filename

是由程序检测的更多信息。

如果 DFHWBAPI 用于访问 3270 应用程序，则 CICS 生成在 Web 客户机上显示的 HTML 表单。CICS 插入到 HTML 表单中的 URL 与先前请求中使用的 *targeturl* 匹配。要处理这种情况，除了上述几点外，还必须提供以下格式的服务伪指令：

```
Service /targeturl/* /home/dfhwbapi.so:DFHService
```

这里，*targeturl* 在未经更改下传递给 DFHWBAPI。

6. CICS 提供的事务的某些 CICS 提供的模板定义包含对图形文件的引用，其格式为：

```
/dfhwbimg/filename
```

其中 DFHWBIMG 是 CICS 提供的、CICS Web 网桥使用的专用转换器程序。如果希望这些图形文件正确显示，应包含如下所示的伪指令：

```
Service /dfhwbimg/* /home/dfhwbapi.so:DFHService/applid/DFHWBIMG/CSM3/*
```

其中 *applid* 指定将提供图形文件的 CICS 系统（这个系统可能与执行网桥工作的 CICS 系统不同）。

如果正在使用 CICS Web support 和 CICS 业务逻辑接口访问 CICS Web 应用程序，则必须为两者指定相同的主机代码页。对于 CICS，缺省主机代码页是 IBM-037，但是对于 IBM HTTP Server，它是 IBM-1047。您可以使用 DefaultFsCp 配置伪指令来更改 IBM HTTP Server 的缺省代码页。例如：

```
DefaultFsCp IBM-1047
```

要更改 CICS 使用的缺省代码页，在 DOCCODEPAGE 系统初始化参数中指定它（例如，DOCCODEPAGE=1047）。使用这个缺省值所引用的文档和文档碎片必须以指定的

代码页进行编码。特别是，如果使用 BMS 映射定义生成的文档模板，应该使用一个模板定制宏来更改生成模板的代码页。请使用 DFHMDX 宏的 CODEPAGE 参数来指定。例如：

```
DFHMDX MAPSET=*,MAP=*,CODEPAGE=1047
```

要获得有关定制 BMS 映射定义生成的模板的更多信息，请参阅第 147 页的第 16 章，『从 BMS 定义创建 HTML 模板』。

转义数据和 IBM HTTP Server

如果使用 IBM HTTP Server 和 CICS 业务逻辑接口访问同一 CICS 应用程序，则必须确保在这两种情况下对转义数据进行一致的处理。

IBM HTTP Server 将数据以其未转义形式传递到 CICS 应用程序；因此，必须确保 CICS Web support 执行相同的操作。

要获得更多信息，请参阅第 97 页的『从分析器程序选择转义数据或未转义数据』

IBM HTTP Server 的处理示例

图 23 显示了 CICS Web support 如何处理来自连接到 IBM HTTP Server 的 Web 客户机的请求。

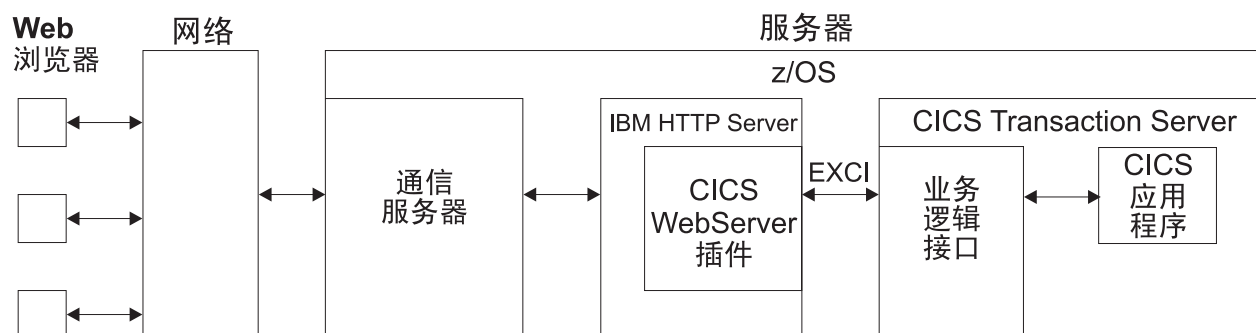


图 23. 处理来自 IBM HTTP Server 的请求

1. Web 客户机构造通过网络传递到 Communications Server 的 HTTP 请求。
2. Communications Server 将请求中继至 IBM HTTP Server。
3. IBM HTTP Server 调用 CICS Web 服务器插件。
4. CICS Web 服务器插件为 CICS 业务逻辑接口构造一个请求，并通过使用外部 CICS 接口（EXCI）将该请求传递到 CICS。
5. CICS 业务逻辑接口调用请求的 CICS 应用程序，并在通信区域中返回任何输出。

CICS Transaction Server for z/OS 库

CICS Transaction Server for z/OS的出版信息以下列格式表达:

CICS Transaction Server for z/OS 信息中心

CICS Transaction Server for z/OS 信息中心是 CICS Transaction Server 用户信息的主要来源。信息中心包含:

- HTML 格式表示的 CICS Transaction Server 信息。
- 作为 Adobe 可移植文档格式 (PDF) 文件提供的许可和未许可 CICS Transaction Server 书籍。您可使用这些文件打印书籍的硬拷贝。要获得更多信息, 请参阅『仅 PDF 书籍』。
- HTML 格式和 PDF 文件表示的相关产品信息。

在 CD-ROM 上, CICS 信息中心的一个副本与产品一起自动提供。通过指定 Information Center 功能部件号 7014, 可以订购更多的副本, 而无需支付额外的费用。

只有产品的被许可方才能使用许可文档。仅包含未许可信息的信息中心版本可通过出版物订购系统 (订单号码 SK3T-6945) 获得。

基本硬拷贝书籍

下列必备出版物以硬拷贝格式与产品一起自动提供。要获得更多信息, 请参阅『基本集合』。

基本集合

基本集合包括下列硬拷贝书籍, 这些书籍在您订购 CICS Transaction Server for z/OS V3R1 时自动提供:

Memo to Licensees, GI10-2559
CICS Transaction Server for z/OS Program Directory, GI10-2586
CICS Transaction Server for z/OS 发行指南, G151-0180
CICS Transaction Server for z/OS 安装指南, G151-0048
CICS Transaction Server for z/OS Licensed Program Specification, G151-0174

您可使用上述提供的订单号码订购基本集合中下列书籍的未来副本:

CICS Transaction Server for z/OS 发行指南
CICS Transaction Server for z/OS 安装指南
CICS Transaction Server for z/OS Licensed Program Specification

仅 PDF 书籍

下列书籍在 CICS 信息中心中作为 Adobe 可移植文档格式 (PDF) 文件提供:

CICS Transaction Server for z/OS 的 CICS 书籍

通用

CICS Transaction Server for z/OS Program Directory, GI10-2586
CICS Transaction Server for z/OS 发行指南, G151-0180
CICS Transaction Server for z/OS Migration from CICS TS Version 2.3, GC34-6425

CICS Transaction Server for z/OS Migration from CICS TS Version 1.3, GC34-6423

CICS Transaction Server for z/OS Migration from CICS TS Version 2.2, GC34-6424

CICS Transaction Server for z/OS 安装指南, G151-0048

管理

CICS System Definition Guide, SC34-6428

CICS Customization Guide, SC34-6429

CICS Resource Definition Guide, SC34-6430

CICS Operations and Utilities Guide, SC34-6431

CICS Supplied Transactions, SC34-6432

编程

CICS Application Programming Guide, SC34-6433

CICS Application Programming Reference, SC34-6434

CICS System Programming Reference, SC34-6435

CICS Front End Programming Interface User's Guide, SC34-6436

CICS C++ OO Class Libraries, SC34-6437

CICS Distributed Transaction Programming Guide, SC34-6438

CICS Business Transaction Services, SC34-6439

Java Applications in CICS, SC34-6440

JCICS Class Reference, SC34-6001

诊断

CICS Problem Determination Guide, SC34-6441

CICS Messages and Codes, GC34-6442

CICS Diagnosis Reference, LY33-6110

CICS Data Areas, LY33-6107

CICS Trace Entries, SC34-6443

CICS Supplementary Data Areas, LY33-6108

Communication

CICS Intercommunication Guide, SC34-6448

CICS External Interfaces Guide, SC34-6449

CICS 因特网指南, S151-0049

特殊主题

CICS Recovery and Restart Guide, SC34-6451

CICS Performance Guide, SC34-6452

CICS IMS Database Control Guide, SC34-6453

CICS RACF Security Guide, SC34-6454

CICS Shared Data Tables Guide, SC34-6455

CICS DB2 Guide, SC34-6457

CICS Debugging Tools Interfaces Reference, LY33-6109

CICS Transaction Server for z/OS 的 CICSplex SM 书籍

通用

CICSplex SM Concepts and Planning, SC34-6459

CICSplex SM User Interface Guide, SC34-6460

CICSplex SM Web User Interface Guide, SC34-6461

管理

CICSplex SM Administration, SC34-6462

CICSplex SM Operations Views Reference, SC34-6463
CICSplex SM Monitor Views Reference, SC34-6464
CICSplex SM Managing Workloads, SC34-6465
CICSplex SM Managing Resource Usage, SC34-6466
CICSplex SM Managing Business Applications, SC34-6467

编程

CICSplex SM Application Programming Guide, SC34-6468
CICSplex SM Application Programming Reference, SC34-6469

诊断

CICSplex SM Resource Tables Reference, SC34-6470
CICSplex SM Messages and Codes, GC34-6471
CICSplex SM Problem Determination, GC34-6472

CICS 系列书籍

Communication

CICS Family: Interproduct Communication, SC34-6473
CICS Family: Communicating from CICS on System/390, SC34-6474

许可出版物

下列许可出版物不包含在未许可版本信息中心中:

CICS Diagnosis Reference, LY33-6110
CICS Data Areas, LY33-6107
CICS Supplementary Data Areas, LY33-6108
CICS Debugging Tools Interfaces Reference, LY33-6109

其他 CICS 书籍

下列出版物包含有关 CICS 的进一步信息，但不作为 CICS Transaction Server for z/OS V3R1 的一部分提供。

<i>Designing and Programming CICS Applications</i>	SR23-9692
<i>CICS Application Migration Aid Guide</i>	SC33-0768
<i>CICS Family: API Structure</i>	SC33-1007
<i>CICS Family: Client/Server Programming</i>	SC33-1435
<i>CICS Transaction Gateway for z/OS Administration</i>	SC34-5528
<i>CICS Family: General Information</i>	GC33-0155
<i>CICS 4.1 Sample Applications Guide</i>	SC33-1173
<i>CICS/ESA 3.3 XRF Guide</i>	SC33-0661

非 CICS 书籍

UNIX 系统服务

- *z/OS UNIX System Services User's Guide*, SA22-7801
- *z/OS UNIX System Services Command Reference*, SA22-7802
- *z/OS UNIX System Services Programming Tools*, SA22-7805
- *z/OS UNIX System Services Messages and Codes*, SA22-7807

- *z/OS UNIX System Services Programming: Assembler Callable Services Reference*, SA22-7803
- *z/OS UNIX System Services File System Interface Reference*, SA22-7808
- *z/OS Using REXX and z/OS UNIX System Services*, SA22-7806
- *z/OS UNIX System Services Parallel Environment: MPI Programming and Subroutine Reference*, SA22-7812

z/OS Communications Server

- *z/OS Communications Server: IP Configuration Reference*, SC31-8776
- *z/OS Communications Server: IP Configuration Guide*, SC31-8775
- *z/OS Communications Server: IP Migration*, GC31-8773
- *z/OS Communications Server: IP CICS Sockets Guide*, SC31-8807
- *z/OS Communications Server: IP Application Programming Interface Guide*, SC31-8788
- *z/OS Communications Server: IP Programmer's Reference*, SC31-8787
- *z/OS Communications Server: IP User's Guide and Commands*, SC31-8780
- *z/OS Communications Server: Quick Reference*, SX75-0124
- *z/OS Communications Server: IP Diagnosis*, GC31-8782

IBM Redbooks

IBM Redbooks 由 IBM 国际技术支持组织开发和发布。他们研究现实客户方案的集成、实现和操作。

下列 IBM Redbooks 包含与此出版物中的材料相关的信息:

- *Accessing CICS Business Applications from the World Wide Web*, SG24-4547
- *How to Secure the Internet Connection Server for MVS/ESA*, SG324-4803
- *Revealed! Architecting Web Access to CICS*, SG24-5466
- *Workload Management for Web Access to CICS*, SG24-6118

Web 上的信息

本书中提供了一些 URL，并且说明无法保证它们的性能。

RFC (请求评论)

- RFC 1945, 超文本传输协议 - *HTTP/1.0* (*HTTP/1.0 规范*): <http://www.ietf.org/rfc/rfc1945.txt>
- RFC 1867, *HTML* 中基于表单的文件上载: <http://www.ietf.org/rfc/rfc1867.txt>
- RFC 2616, 超文本传输协议 - *HTTP/1.1* (*HTTP/1.1 规范*): <http://www.ietf.org/rfc/rfc2616.txt>
- RFC 2396, 统一资源标识 (*URI*): 一般语法: <http://www.ietf.org/rfc/rfc2396.txt>
- RFC 3023, *XML* 介质类型: <http://www.ietf.org/rfc/rfc3023.txt>

RFC 文档由因特网协会和因特网工程任务组织 (IETF) 发布。

IANA (因特网地址管理机构)

- IANA 注册的介质类型: <http://www.iana.org/assignments/media-types/>

- IANA 注册的字符集名称: <http://www.iana.org/assignments/character-sets>
- IANA 注册的端口号: <http://www.iana.org/assignments/port-numbers>

万维网联合组织 (W3C)

- W3C 主页: <http://www.w3.org/>
- W3C HTTP 概述: <http://www.w3.org/Protocols/Overview.html>
- W3C HTML 主页: <http://www.w3.org/MarkUp/>

IBM

- IBM developerWorks®: <http://www.ibm.com/developerworks/>

确定出版物是否是最新的

IBM 定期使用新的和已更改的信息更新其出版物。当首次出版时, 出版物的硬拷贝和 BookManager®软拷贝版本通常是同步的。但是, 由于印刷和发行硬拷贝书籍需要一定的时间, 因此 BookManager 版本中很有可能包含了出版该书之前进行的最后一次更改。

后继更新很可能先在软拷贝中找到, 而后才可在硬拷贝中找到。这就意味着, 从可发行的任何时间起, 软拷贝版本都将被认为是最新的。

对于 CICSTransaction Server 书籍, 这些软拷贝的更新定期出现在事务处理和数据集合包 CD-ROM, SK2T-0730-xx 中。每个重新发布的集合包都用一个更新的订单号后缀 (-xx 部分) 表示。例如, 集合包 SK2T-0730-06 比 SK2T-0730-05 要新。集合包在其封面上也清晰地标有日期。

对软拷贝的更新将由修订代码 (通常是一个 # 字符) 清晰地标记在所更改内容的左侧。

辅助功能选项

辅助功能选项功能部件帮助生理残障（例如活动受限或视力不好）用户成功地使用软件产品。

您可以使用下列某个方式执行安装、运行和维护 CICS 系统所需的大多数任务：

- 使用 3270 仿真器登录到 CICS
- 使用 3270 仿真器登录到 TSO
- 使用 3270 仿真器作为 MVS 系统控制台

IBM Personal Communications 为残障人士提供具有辅助功能选项功能部件的 3270 仿真。您可以使用这个产品来提供您在 CICS 系统中需要的辅助功能选项功能部件。

如果使用“Web 终端转换应用程序 (DFHWBTTA)”来访问 CICS 提供的事务，则辅助功能选项功能部件是由 Web 浏览器提供的。

索引

[A]

- 安全套接字层 (SSL) 17
 - 抽取信息 60
- 安全性 107
 - 标识 109, 110
 - 代理认证 110
 - 基本认证 109, 110, 111
 - 密码到期管理 111
 - 认证 107, 109, 110
 - 入站端口 107
 - 文档模板 107
 - 资源层次 107
 - CICS 系统 107
 - HFS 文件 107
 - SSL 112
 - z/OS UNIX System Services 107

[B]

- 表单 13
- 表单数据
 - 检查 61
- 表单字段 13, 61
- 别名事务 20
 - 在 URIMAP 资源定义中 77
- 不支持 Web 的应用程序 21, 51
 - 对于非 HTTP 消息 127, 129
 - 分析器程序 91

[C]

- 查询字符串 7, 9
 - 表单数据 61
 - 从请求行抽出 58
 - 来自 CICS 的客户机请求中 116, 118
 - 在 CICS Web support 中使用 78
 - 在 URIMAP 资源定义中 (作为服务器的 CICS) 75
 - CICS Web support 中使用 28, 48
- 超文本传输协议 (HTTP) 5
- 持续连接 15, 31
 - 作为 HTTP 客户机的 CICS 27
- 重定向 79, 82
- 处理示例
 - ECI 请求 171
 - EXCI 请求 171
- 传输层安全性 (TLS) 17
- 传输控制协议 (TCP) 5
- 错误处理 89

错误处理 (续)

- 分析器程序的角色 91
- 状态码参考 191
- Web 出错程序的角色 85

[D]

- 代理认证 110
- 代码页转换 32
 - 对于非 HTTP 消息 128
 - 对于使用分析器程序的不支持 Web 的应用程序 91
 - 由分析器程序指定 95
 - 在 DFHWBADX 中, 样本分析器程序 99
 - 支持的字符集 183
 - 作为 HTTP 服务器的 CICS 33, 34
 - 作为 HTTP 客户机的 CICS 25
- 代码页转换表 DFHCNV 17, 41
 - 迁移条目 43
- 端口号 7, 45
 - 安全性 107
 - 对于非 HTTP 消息 127
 - 来自 CICS 的客户机请求中 116
 - 临时 7
 - 熟知 7, 42
 - 为 CICS Web support 保留 41, 42
 - 在 URIMAP 资源定义中 (作为服务器的 CICS) 75
 - 在 URIMAP 资源定义中 (作为客户机的 CICS) 123
- 多部分 / 表单数据 13

[F]

- 方法 9, 10, 199
 - 扩展方法 199
 - 来自 CICS 的客户机请求中 118
- 非 HTTP 消息 127
 - 分析器程序 128
 - 概述 17
 - 应用程序 129
 - 支持 127
 - 资源定义 128
- 分块 14, 29
 - 分块的传输编码 14, 29
 - 方法 67
- 分析器程序 19, 21, 45, 51, 91
 - 编写 93
 - 参考信息 203
 - 参数 203, 204
 - 对于非 HTTP 消息 127, 128

分析器程序 (续)

- 分块的传输编码 29
- 函数 93
- 输出 95
- 输入 94
- 响应和原因码 208
- 与转换器程序共享数据 97
- 与 URIMAP 资源定义的关系 91, 94
- 在 URIMAP 资源定义中 77
- 转义和未转义数据 97
- CICS 所提供
 - DFHWBAAX 98
 - DFHWBADX 99
- DFHCNV, 代码页转换表 43

[G]

- 管道传送 14, 30
 - 发出管道化的请求 120
- 规则 7
 - 来自 CICS 的客户机请求中 116

[H]

- 会话令牌 25, 27, 115, 117
 - 获取 116

[J]

- 基本认证 15, 109, 110, 111
- 基本 64 位编码 15
- 介质类型 7, 66, 118, 121
 - 在静态响应的 URIMAP 资源定义中 78
- 警告头 79, 83
- 静态 HTTP 响应
 - 处理 21
 - 错误处理 89
 - 方法 48
 - URIMAP 资源定义 78
- 绝对 URI 9

[K]

- 客户机代码页 183
- 客户机证书 60, 109, 110
- 客户机 HTTP 请求
 - 分块的传输编码 67
 - 概述 115
 - URIMAP 资源定义 123
- 控制流
 - 面向终端的事务 173

控制流 (续)

- 业务逻辑接口 172
- 扩展方法 199

[L]

- 连接
 - 持续 15, 31
 - 关闭 (作为服务器) 66
 - 关闭 (作为客户机) 118, 122
 - 关闭 (作为客户机, 对于管道化的请求) 120
- 临时存储队列 17
 - 启用支持 41
- 路径 7, 9
 - 长度限制 28
 - 从请求行抽出 58
 - 来自 CICS 的客户机请求中 116, 118
 - 由 DFHWBADX 解释, 样本分析器程序 99
 - 在 URIMAP 资源定义中 (作为服务器的 CICS) 75
 - 在 URIMAP 资源定义中 (作为客户机的 CICS) 123
 - CICS Web support 中使用 28
- 路径匹配
 - 在 URIMAP 资源定义中 78

[M]

- 幂等性 14, 30, 120
- 密码到期管理程序 DFHWBPW 19, 109, 111

[Q]

- 亲缘关系
 - CICSplex 中的 164
- 请求
 - 产生 118
 - 管道传送 120
 - 接收 62
- 请求行 9
 - 检查 58
- 请求主体 9
 - 产生 118
 - 接收 62
 - 相应方法 199

[R]

- 认证 15, 109, 110, 111
- 日期和时间戳记
 - 以 RFC 1123 格式产生 64, 117
 - 转换为 ABSTIME 59

[S]

- 升级头 185
- 时间戳记
 - 以 RFC 1123 格式产生 64, 117
 - 转换为 ABSTIME 59
- 熟知端口号 7, 42
- 数据流
 - 面向终端事务 175, 177
 - 业务逻辑接口 174

[T]

- 套接字接口 5
- 套接字侦听器任务 (CSOL) 17, 20
- 通配符
 - 在 URIMAP 资源定义中 78
- 通信区域应用程序 51
 - CICS Web support 中的角色 19

[W]

- 外部调用接口 (ECI) 171
- 外部 CICS 接口 (EXCI) 171
- 网际协议 (IP) 5
- 尾部 14
- 尾部头 14, 29, 67
 - 方法 67
- 未转义 12
 - 表单数据 13
 - 在分析器程序中 97
- 文档 19, 65, 118
- 文档模板 19, 48, 65
 - 安全性 107
 - 在 URIMAP 资源定义中 78
- CICSFOOT 141
- CICSHEAD 140

[X]

- 系统编程命令 79
- 系统初始化参数 42
 - 概述 17
 - DOCCODEPAGE 42
 - LOCALCCSID 42
 - TCPIP 42
 - WEBDELAY 42
- 响应
 - 产生 66
 - 接收 121
- 响应和原因码
 - 分析器程序 208

- 响应和原因码 (续)
 - 业务逻辑接口 221
 - 转换器程序
 - 编码函数 216
 - 解码函数 212
- 响应主体 10
 - 产生 65
 - 接收 121
- 消息体 9, 10
 - 产生 65
 - 打包或压缩 121
 - 接收 62
 - 相应方法 199
- 虚托管 6, 79, 81
 - INQUIRE HOST 命令 79
 - SET HOST 命令 79
- 需要头 185

[Y]

- 样本应用程序
 - DFH0WBCA 43
 - DFH\$WB1A 43
 - DFH\$WB1C 43
- 业务逻辑接口
 - 参考信息 217
 - 事务的控制流 173
 - 响应 221
 - 一个程序的控制流 172
 - 一个事务的数据流 175, 177
- 业务逻辑 接口
 - 一个程序的数据流 174
- 异步接收 20
- 因特网地址 6
- 因特网地址管理机构 (IANA) 7
- 应用程序生成的 HTTP 响应
 - 编写应用程序 57
 - 处理 21
 - 方法 45
 - 分块的传输编码 67
 - URIMAP 资源定义 75, 77
- 应用程序状态 70, 233
- 用户标识
 - 安全性 107
 - 在 URIMAP 资源定义中 77
- 域名 6
- 域名服务器 6
- 域名服务 (DNS) 6
- 域, 用于认证 15
- 原因短语 10, 12, 66, 121, 191

[Z]

- 支持 Web 的应用程序 21, 45, 57
 - 编写 57
 - 编写 HTTP 头 64
 - 产生实体主体 65
 - 定义 18
 - 对于非 HTTP 消息 127, 129
 - 发送响应 66
 - 分块的传输编码 67
 - 检查表单数据 61
 - 检查请求行 58
 - 检查 HTTP 头 59
 - 检索安全信息 60
 - 检索 TCP/IP 信息 60
 - 接收实体主体 62
 - 伪会话模型 70
 - 应用程序状态 70
 - HTTP/1.1 支持 35, 36, 38
 - URIMAP 资源定义 75, 77
- 主机名 6, 7
 - 来自 CICS 的客户机请求中 116
 - 在 URIMAP 资源定义中 (作为服务器的 CICS) 75
 - 在 URIMAP 资源定义中 (作为客户机的 CICS) 123
- 主体实体 9, 10
 - 产生 65
 - 打包或压缩 121
 - 接收 62
 - 相应方法 199
- 转换器程序 19, 21, 51, 101, 102
 - 编码函数 102
 - 参考信息 214
 - 输出参数 104
 - 输入参数 104
 - 编写 102
 - 参考信息 209
 - 调用多个应用程序 105
 - 对于非 HTTP 消息 129
 - 构造响应 104
 - 解码函数 102
 - 参考信息 209
 - 概述 101
 - 输出参数 103
 - 输入参数 103
 - 与分析器程序共享数据 97
 - 与 URIMAP 资源定义的关系 101
 - 在 URIMAP 资源定义中 77
- 转换器程序的编码函数 102
 - 参考信息 214
 - 输出参数 104
 - 输入参数 104
- 转换器程序的解码函数 102
 - 参考信息 209
 - 概述 101
 - 输出参数 103
 - 输入参数 103
- 转义 12
 - 表单数据 13
 - 在分析器程序中 97
- 状态管理程序, DFH\$WBST 和 DFH\$WBSR 70, 233
- 状态行 12
- 状态码 10, 12, 66, 121, 191
 - 错误响应的缺省值 89
- 状态文本 12
- 资源定义 71
 - 分析器程序 94
 - 转换器程序 102
 - TCPIPSERVICE 72
 - TRANSACTION 74
 - URIMAP (服务器) 75
 - URIMAP (服务器, 静态响应) 78
 - URIMAP (服务器, 应用程序生成的响应) 77
 - URIMAP (客户机) 123
- 资源定义组
 - DFHDCTG 71
 - DFHWEB 71
 - DFH\$SOT 72, 74
- 字符集 7, 32, 183
- 作为 HTTP 服务器的 CICS
 - 安全性 107, 109, 111
 - 编写支持 web 的应用程序 57
 - 处理 21
 - 代码页转换 33
 - 概述 17
 - 规划 45, 48
 - 静态响应 48
 - 任务结构 20
 - 应用程序生成的响应 45
 - 资源定义 71
 - HTTP/1.1 支持 35, 36, 38
 - URIMAP 资源定义 75
 - 静态响应 78
 - 应用程序生成的响应 77
- 作为 HTTP 客户机的 CICS
 - 安全性 110
 - 编写 HTTP 头 117
 - 处理 25
 - 打开的连接 116
 - 代码页转换 34
 - 发出请求 115
 - 概述 17, 115
 - 关闭连接 122
 - 管道化的请求 30, 120

作为 HTTP 客户机的 CICS (续)

- 规划 45
- 接收响应 121
- 任务结构 20
- 写请求 118
- SSL 112
- URIMAP 资源定义 123

[数字]

3270 显示应用程序 19, 131

A

- ADYN 事务 149
- application/x-www-form-urlencoded 12, 13

B

- BLI (业务逻辑接口)
 - 事务的控制流 173
 - 响应 221
 - 一个程序的控制流 172
 - 一个程序的数据流 174
 - 一个事务的数据流 175, 177

C

- CCSID 183
- CICS 套接字接口 5
- CICS Transaction Gateway 3
- CICS Web 服务器插件 235
- CICS Web support 17, 21, 123
 - 安全性 107
 - 持续连接 31
 - 错误处理 85, 191
 - 代码页转换 32
 - 分块的传输编码 29, 67
 - 服务器 HTTP 处理 21
 - 管道传送 30
 - 管理 79
 - 规划 45
 - 静态 HTTP 响应 48, 78
 - 客户机 HTTP 处理 25
 - 客户机 HTTP 请求 115
 - 配置基本组件 41
 - 任务结构 20
 - 虚托管 81
 - 验证操作 41, 43
 - 应用程序生成的 HTTP 响应 45
 - 用户出口 XWBOPEN, XWBSNDO 124, 125

CICS Web support (续)

- 资源定义 71, 127, 128
- 组件 17
- 作为 HTTP 客户机的 CICS 115
- URIMAP 资源定义
 - 服务器 HTTP 处理 77, 78
- CICSFOOT (文档模板) 141
- CICSHEAD (文档模板) 140
- CONVERTTIME 命令 59
- CREATE TCPIP SERVICE 命令 79
- CREATE URIMAP 命令 79
- CSOL, 套接字侦听器任务 17, 20
- CWBA 20, 74
- CWBO, 瞬时数据队列 71, 83
- CWBW, 瞬时数据队列 71, 83
- CWXN, Web 连接任务 17, 20, 74
- CWXU, Web 连接任务 17, 20, 74, 128

D

- DefaultFsCp (配置伪指令) 236
- DFH0WBCA, 样本应用程序 43
- DFHCNV, 代码页转换表 17, 41
 - 迁移条目 43
- DFHDCTG 资源定义组 71
- DFHMDX 宏 153
- DFHWBA (别名程序) 74
- DFHWBAAX, 缺省分析器程序 98
 - 服务器 HTTP 处理 21
 - 概述 91
 - 行为 98
 - 通信区域应用程序 51
 - 支持 Web 的应用程序 45
- DFHWBADX, 样本分析器程序 99
 - 服务器 HTTP 处理 21
 - 概述 91
 - 请求 URL 格式 99
 - 通信区域应用程序 51
 - 响应 99
 - 支持 Web 的应用程序 45
 - DFHCNV, 代码页转换表 43
- DFHWBBLI, CICS 业务逻辑接口 217
- DFHWBCLI Web 客户机接口 227
- DFHWBEP, Web 出错程序
 - 参考信息 223
 - 服务器 HTTP 处理 21
 - 概述 85
 - 行为 87
 - 缺省响应 89
 - 输出 88, 223
 - 输入 88, 223
 - CICS Web support 中的角色 19

- DFHWBERX, Web 出错应用程序
 - 服务器 HTTP 处理 21
 - 概述 85
 - 行为 86
 - CICS Web support 中的角色 19
- DFHWBPW, 密码到期管理程序 19, 109, 111
- DFHWBTTA (Web 终端转换应用程序) 19, 131
- DFHWBTTB 19, 131
- DFHWBTTC 19, 131
- DFHWEB 资源定义组 71
- DFH\$SOT 资源定义组 72, 74
- DFH\$WB1A, 样本应用程序 43
- DFH\$WB1C, 样本应用程序 43
- DFH\$WBSR, 样本状态管理程序 70, 233
- DFH\$WBST, 样本状态管理程序 70, 233
- DISCARD TCPIP SERVICE 命令 79
- DISCARD URIMAP 命令 79
- DNS 服务器 6
- DOCCODEPAGE (系统初始化参数) 42

E

- ECI (外部调用接口) 171
- ECI 请求
 - 处理示例 171
- EXCI (外部 CICS 接口) 171
- EXCI 请求
 - 处理示例 171
- Expect 头 118
- EXTRACT CERTIFICATE 命令 60, 109
- EXTRACT TCPIP 命令 60
 - 在分析器程序中使用 94

F

- FORMATTIME 命令 64, 117

H

- HFS 文件 48
 - 安全性 107
 - 在 URIMAP 资源定义中 78
- HTML 表单 13
- HTML 模板 147
- HTTP 版本 9, 10
- HTTP 规范 9
- HTTP 基本认证 15
- HTTP 客户机发送出口 XWBSNDO 118
- HTTP 客户机开放出口 XWBOPEN 116, 117, 124
- HTTP 客户机请求 115
 - 安全性 110
 - 编写 HTTP 头 117

- HTTP 客户机请求 (续)
 - 打开的连接 116
 - 关闭连接 122
 - 接收响应 121
 - 写请求 118, 120
 - URIMAP 资源定义 123
- HTTP 请求 9
 - 产生 118
 - 重定向 79, 82
 - 方法 199
 - 管道传送 120
 - 接收 62
 - 拒绝 79, 82
- HTTP 请求和响应处理
 - 持续连接 31
 - 分块的传输编码 29, 67
 - 管道传送 30
 - 资源定义 71
 - 作为 HTTP 服务器的 CICS 21
 - 作为 HTTP 客户机的 CICS 25, 115
- HTTP 头 9, 10, 185
 - 检查 59, 121
 - 警告头 83
 - 来自 CICS 的客户机请求中的 Expect 头 118
 - 已分块消息中的尾部头 67
 - 应用程序编写, 作为服务器 64, 185
 - 应用程序编写, 作为客户机 117, 185
 - CICS 提供, 作为服务器 64, 185
 - CICS 提供, 作为客户机 117, 185
 - CICS Web support 的完整列表 185
- HTTP 响应 10, 12
 - 产生 66
 - 接收 121
- HTTP 协议规范 9
- HTTP 状态码 10, 12, 66, 121, 191
 - 错误响应的缺省值 89
- HTTP/1.1 支持 35, 36, 38

I

- IANA 7, 32
- IANA 字符集 7, 183
- IBM HTTP Server 3, 235
 - 处理示例 237
 - 配置 235
- INQUIRE HOST 命令 79
- INQUIRE TCPIP 命令 79
- INQUIRE TCPIP SERVICE 命令 79
- INQUIRE URIMAP 命令 79
- IP 地址 6
- IPv4 地址和 IPv6 地址 6
- ISO-8859-1 字符集 32, 183

L

LINK 命令 171
LOCALCCSID (系统初始化参数) 32, 42

P

PORT 语句 42
PROFILE.TCPIP 数据集 42
PROGRAM 定义
 分析器程序 94
 转换器 102

R

RTIMOUT 设置 121

S

SET HOST 命令 79
SET TCPIP 命令 79
SET TCPIPSERVICE 命令 79
SET URIMAP 命令 79
SIT 参数 42
 概述 17
 DOCCODEPAGE 42
 LOCALCCSID 42
 TCPIP 42
 WEBDELAY 42
SOCKETCLOSE 72, 74
SOMAXCONN 参数 42
SSL 17, 112
 抽取信息 60
 对于作为 HTTP 客户机的 CICS 112
 客户机证书认证 109, 110
 URIMAP 资源定义 107

T

TCP62
 与 ECI 客户机 172
TCPIP (系统初始化参数) 42
TCPIPSERVICE 资源定义
 安全性 107
 持续连接 31
 定义 72
 对于非 HTTP 消息 127, 128
 分析器程序的角色 (URM) 91
 服务器 HTTP 处理 21
 禁用 82
 静态 HTTP 响应 48
 虚托管 81

TCPIPSERVICE 资源定义 (续)

 样本 72
 应用程序生成的 HTTP 响应 45
 用于管理 79
 在 URIMAP 资源定义中命名 75
 CICS Web support 中的角色 17, 71
 CREATE TCPIPSERVICE 命令 79
 DISCARD TCPIPSERVICE 命令 79
 INQUIRE TCPIPSERVICE 命令 79
 SET TCPIPSERVICE 命令 79

TCP/IP 5, 17

 抽取信息 60
 启用支持 41

TLS 17

TRANSACTION 事务资源

 定义 74
 对于非 HTTP 消息 128
 缺省值, CWBA 74
 CICS Web support 中的角色 17, 71

U

URI (通用资源标识) 7

 保留和排除的字符 12
 绝对 URI 9

URIMAP 资源定义

 重定向 82
 对于作为 HTTP 服务器的 CICS 21, 75, 77, 78
 对于作为 HTTP 客户机的 CICS 25, 115, 123
 禁用 82
 静态 HTTP 响应 48
 虚托管 81
 应用程序生成的 HTTP 响应 45
 用于管理 79
 与分析器程序的关系 91
 在打开的连接中使用 116
 在发送的请求中使用 118
 CICS Web support 中的角色 17, 71
 CREATE URIMAP 命令 79
 DISCARD URIMAP 命令 79
 INQUIRE URIMAP 命令 79
 SET URIMAP 命令 79

URL (统一资源定位符) 7, 9

 保留和排除的字符 12
 长度限制 28
 对于 CICS Web support 28
 来自 CICS 的客户机请求中 116, 118
 在 URIMAP 资源定义中 (作为服务器的 CICS) 75
 在 URIMAP 资源定义中 (作为客户机的 CICS) 123

URL 中的 %xx 序列 12

W

Web 出错程序

参考信息 223

服务器 HTTP 处理 21

概述 85

缺省响应 89

通信区域输出 88, 223

通信区域输入 88, 223

应用程序生成的 HTTP 响应 45

状态码参考 191

CICS Web support 中的角色 19

DFHWBEP, Web 出错程序 87

DFHWBERX, Web 出错应用程序 86

Web 连接任务, CWXN 和 CWXU 17, 20, 74

Web 终端转换应用程序 (DFHWBTTA) 19, 131

WEB CLOSE 命令 122

WEB CONVERSE 命令 118, 121

WEB ENDBROWSE FORMFIELD 命令 61

WEB ENDBROWSE HTTPHEADER 命令 59

WEB EXTRACT 命令 58, 64

WEB OPEN 命令 116

WEB READ FORMFIELD 命令 61

WEB READ HTTPHEADER 命令 59

WEB READNEXT FORMFIELD 命令 61

WEB READNEXT HTTPHEADER 命令 59

WEB RECEIVE 命令 62, 121

对于非 HTTP 消息 129

WEB SEND 命令 65, 66, 67, 118, 120

对于非 HTTP 消息 129

Web service 3

WEB STARTBROWSE FORMFIELD 命令 61

WEB STARTBROWSE HTTPHEADER 命令 59

WEB WRITE HTTPHEADER 命令 64, 67, 117

WEBDELAY (系统初始化参数) 42

X

XWBOPEN 用户出口 25, 116, 117, 124

XWBSNDO 用户出口 25, 118, 125

X.509 证书 60

Z

z/OS Communications Server 5, 17

保留端口 42

启用支持 41

z/OS UNIX 系统服务

启用支持 41

z/OS UNIX System Services 17

z/OS UNIX Systems Services

安全性 107

声明

本信息是为在美国提供的产品和服务编写的。IBM 可能在其他国家或地区不提供本文中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档中所描述内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可证。您可以用书面方式将许可证查询寄往：

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

有关双字节（DBCS）信息的许可证查询，请与您所在国家或地区域的 IBM 知识产权部门联系，或用书面方式将查询寄往：

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

本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区： 国际商业机器公司以“按现状”的基础提供本出版物，不附有任何形式的（无论是明示的，还是默示的）保证，包括（但不限于）对非侵权性、适销性和适用于某特定用途的默示保证。某些国家或地区在某些交易中不允许免除明示或默示的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和 / 或程序进行改进和 / 或更改，而不另行通知。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：(i)允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及 (ii)允许对已经交换的信息进行相互使用，请与下列地址联系 IBM United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN。只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际程序许可证协议或任何同等协议中的条款提供。

编程接口信息

本书是为了帮助您使用 CICS Transaction Server for z/OS的 CICS Web support组件。它编录了 CICS 提供的“通用编程接口”和“相关指导信息”。

“通用编程接口”允许客户编写获得 CICS 服务的程序。

本书还编录了 CICS 提供的“产品敏感的编程接口”和“相关的指导信息”，以及“诊断、修改或调整的信息”。

“产品敏感的编程接口”允许客户安装以执行任务（如 CICS 的诊断、修改、监控、修理、裁剪或调整）。这类接口的使用依赖于 IBM 软件产品的具体设计或实现。“产品敏感的编程接口”应该只用于这些特殊目的。正因为这种依赖性（依赖于具体的设计和实现），所以可能需要更改针对这种接口编写的程序，以便能与产品新的发行版或新版本一起运行，或作为服务的结果。

“产品敏感的编程接口”和“相关的指导信息”出现时，由各章节的介绍性语句标识。

提供“诊断、修改或调整信息”，是为了帮助您诊断 CICS 系统中的问题。

注：不要将“诊断、修改或调整信息”作为编程接口使用。

“诊断、修改或调整信息”出现时，由各章节的介绍性语句标识。

商标

以下术语是 International Business Machines Corporation 在美国和 / 或其他国家或地区的商标或注册商标：

AIX	developerWorks	RACF
BookManager	IBM	Redbooks™
CICS	MQSeries®	System/390
CICSplex	MVS	z/OS
DB2	OS/390	

Java、JavaScript 和所有基于 Java 的商标和徽标是 Sun Microsystems, Inc. 在美国和其他国家或地区的商标或注册商标。

Microsoft®、Windows 和 Windows NT® 是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

UNIX 是 X/Open Company Limited 在美国和其他国家专用许可的注册商标。

其他公司、产品和服务名称是其他公司的商标或服务标记。

将您的意见发送给 IBM

如果您对本书中的任何内容有特殊欣赏或者非常不赞成的地方，请使用下面列出的一种方法将您的意见发送给 IBM。

请对本书中您认为是特定的错误或疏漏的部分，以及本书的准确性、组织、主旨或完整性自由发表您的意见。

请仅对本书中的信息以及信息表达的方式提出意见。

若要提出问题、对 IBM 产品或系统功能提出意见，或是索取其他出版物，请与 IBM 代表或 IBM 授权的代理商联系。

当您发送意见至 IBM 后，即授予 IBM 非专有权，IBM 可以它认为合适的任何方式使用或分发此意见，而无须对您承担任何责任。

可以按下列方式之一将您的意见发送给 IBM:

- 邮寄到下列地址:

User Technologies Department (MP095)
IBM United Kingdom Laboratories
Hursley Park
Hampshire
Hampshire
SO21 2JN
United Kingdom

- 传真至:

- 英国以外的地区，在拨通国际访问码之后使用 44-1962-816151
- 英国以内的地区，使用 01962-816151

- 以电子方式发送，请使用适当的网络标识:

- IBMLink™: HURSLEY(IDRCF)
- 因特网: idrcf@hursley.ibm.com

无论您使用哪种方法，都请确保包含了以下信息:

- 出版物标题和订单号
- 您的意见所针对的主题
- 您的姓名和地址 / 电话号码 / 传真号码 / 网络标识。



程序号: 5655-M15

S151-0049-00



Spine information:



CICS Transaction Server for
z/OS

因特网指南

版本 3
发行版 1