

IBM WebSphere Business Integration
Connect Enterprise Edition 和
Advanced Edition



企业集成指南

V4.2.2

IBM WebSphere Business Integration
Connect Enterprise Edition 和
Advanced Edition



企业集成指南

V4.2.2

声明:

在使用本资料及其支持的产品之前, 请阅读第 167 页的『声明和商标』中的常规信息。

2004 年 6 月 29 日

本文档的该版本适用于 IBM WebSphere Business Integration Connect Enterprise Edition (5724-E87) 和 Advanced Edition (5724-E75), V4.2.2。

如果您对 IBM WebSphere Business Integration 文档有任何意见, 请发送电子邮件到 doc-comments@us.ibm.com。我们希望收到您的来信。

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 2003, 2004. All rights reserved. US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© Copyright International Business Machines Corporation 2003, 2004. All rights reserved.

目录

本发行版中的新增内容	vii
关于本书	ix
目标对象	ix
排版约定	ix
相关文档	ix
第 1 部分 后端集成简介	1
第 1 章 规划后端集成	3
后端集成概述	3
规划后端集成	5
使用何种业务协议?	5
您将使用哪种封装?	9
您将使用哪种消息传输?	16
如何访问您的后端应用程序?	20
消息处理	21
排队的传递	21
通信错误处理	21
重复消息	21
创建传输协议机制	22
HTTP/S 传输协议机制	22
JMS 传输协议机制	22
配置 Business Integration Connect	23
向后端系统发送文档	23
从后端系统接收文档	27
第 2 部分 与 WebSphere InterChange Server 集成	31
第 2 章 InterChange Server 集成简介	33
规划与 InterChange Server 的集成	34
Business Integration Connect 支持的 InterChange Server 版本	34
InterChange Server 支持的消息传输	34
InterChange Server 集成支持	36
为 InterChange Server 配置 Business Integration Connect	37
提供对出站文档的支持	37
提供对入站文档的支持	38
配置 InterChange Server	40
创建业务对象定义	40
创建连接器	43
=创建协作	43
部署项目	44
处理带附件的文档	44
执行转换	45
设置附件数据处理程序的环境	49
配置附件数据处理程序	50
创建与附件关联的业务对象定义	54
第 3 章 通过 HTTP 集成 InterChange Server	61
将 HTTP 传输协议用于 4.2.2 之前版本的 ICS	61

通过 HTTP 将文档发送到 4.2.2 之前版本的 ICS	61
通过 HTTP 从 4.2.2 之前版本的 ICS 接收文档	74
为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象	79
创建用于 HTTP 的 4.2.2 之前版本的 ICS 构件	84
将 HTTP 传输协议与 V4.2.2 ICS 一起使用	85
通过 HTTP 传输协议将文档传递到 V4.2.2 ICS 所需的组件	86
设置用于 V4.2.2 ICS 的 HTTP 传输环境	88
为通过 HTTP 收发文档的 V4.2.2 ICS 创建业务对象定义	91
创建用于 HTTP 的 V4.2.2 ICS 构件	97
通过 HTTP/S 发送 SOAP 文档	97
发送和接收所需的组件	98
社区参与者如何调用 Web Service	98
社区管理者如何调用 Web Service	99

第 4 章 JMS 上的 InterChange Server 集成 101

通过 JMS 传输的文档所需的组件	101
通过 JMS 传输发送文档	102
通过 JMS 传输接收文档	104
设置 JMS 传输的环境	106
配置 JMS 队列	107
配置 Adapter for JMS	107
创建 JMS 的业务对象定义	109
创建用于 JMS 的有效负载业务对象结构	109
创建 JMS 头信息	110
创建 JMS 的 ICS 构件	112
创建 JMS 连接器对象	113
绑定协作以与 Adapter for JMS 通信	113

第 3 部分 与其它后端系统集成 115

第 5 章 与 WebSphere Business Integration Message Broker 集成 117

规划与 Message Broker 的集成	118
Business Integration Connect 支持的 Message Broker 版本	118
Message Broker 支持的消息传输	118
支持 Message Broker 集成	119
为 Message Broker 配置 Business Integration Connect	119
提供对出站文档的支持	119
提供对入站文档的支持	120
配置 Message Broker	121
创建消息流	122
部署项目	122
将 HTTP 传输协议和 Message Broker 一起使用	122
通过 HTTP 传输传递文档所需的组件	122
创建用于 HTTP 传输的消息流	124
发送 SOAP 文档	125
将 JMS 传输协议和 Message Broker 一起使用	126
通过 JMS 传输文档所需的组件	126
设置 JMS 传输的环境	130
创建用于 JMS 传输的消息流	131

第 6 章 与 WebSphere Data Interchange 集成 133

简介	133
将文档发送到 WebSphere Data Interchange	133
接收来自 WebSphere Data Interchange 的文档	134
本章中使用的样本方案	135
配置您的消息交换环境	136

配置 WebSphere MQ 通信	136
配置 WebSphere Data Interchange	137
设置 JMS 环境	142
配置 Business Integration Connect Enterprise Edition	143
配置 Business Integration Connect - Express	150
配置 “我的概要文件”	150
创建 Partner One 的参与者	151
配置 Partner One 参与者	151
结束语	152
第 7 章 路由 EDI 文档	153
EDI 路由概述	153
AS 封装的特殊考虑事项	154
路由进站文档	154
路由出站文档	154
在参与者概要文件中设置标识	155
第 4 部分 附录	157
附录. 用 WebSphere MQ 配置 JMS 协议	159
配置 JMS 配置目录	159
创建 JMS 队列	160
创建 MQ 队列管理器	160
创建 MQ 通道和传输队列	160
创建 MQ JMS 本地队列	162
创建 JMS 绑定文件	163
创建 JMS 目标	164
创建 JMS 网关	165
声明和商标	167
声明	167
编程接口信息	169
商标和服务标记	169
索引	171

本发行版中的新增内容

本节描述 IBM WebSphere Business Integration Connect 的新增功能，它们包含在本版本的企业集成指南中。在这次更新 IBM WebSphere Business Integration Connect 4.2.2 发行版时，已将下面的更改添加到本文档：

- V4.2.1 集成指南已经改名为企业集成指南。
- 文档已经进行彻底地重新组织以提供更好的可用性。特别地：
 - 已将如何配置 WebSphere Business Integration Connect 的信息已经从如何配置后端系统中分离出来，因为通常假设由不同的人员或角色来执行这两个任务。
 - 关于如何与 WebSphere InterChange Server 集成的信息已经进行了扩展，并分成几个章节，它没有包含在本书的第 2 部分。要获取与 InterChange Server 集成的介绍，请参阅第 33 页的第 2 章，『InterChange Server 集成简介』。
- WebSphere Business Integration Connect 现在可以使用 WebSphere Business Integration Adapter 获取 HTTP 以提供在 HTTP 传输协议之上提供与 V4.2.2 WebSphere InterChange Server 集成的支持。要了解更多信息，请参阅第 85 页的『将 HTTP 传输协议与 V4.2.2 ICS 一起使用』。
- 新的章节“第 117 页的第 5 章，『与 WebSphere Business Integration Message Broker 集成』”提供关于如何使 WebSphere Business Integration Connect 与 WebSphere Business Integration Message Broker 集成的消息。

关于本书

本文档介绍了后端集成接口，它是后端系统和 IBM^(R) WebSphere^(R) Business Integration Connect 用于通信的机制。接着，文档描述如何通过使用后端集成接口，使 WebSphere InterChange Server 和 WebSphere Data Interchange 与 Business Integration Connect 集成起来。

目标对象

本书是为那些负责将 Business Integration Connect Enterprise Edition 或 Advanced Edition 与后端系统集成的人员准备的。

排版约定

本文档使用了下列约定。

<code>courier</code> 字体	表示字面值，例如命令名、文件名、您输入的信息或系统打印在屏幕上的信息。
粗体	用于表示首次出现的新术语。
<i>斜体</i> , <i>斜体</i>	表示变量名或交叉引用。
蓝色轮廓线	蓝色轮廓线表示交叉引用超链接，仅当您联机查看手册时才能看到蓝色轮廓线。单击轮廓线内部可以跳转至引用目标。
{ }	在语法行中，花括号中包含一组选项，必须从这组选项中选择一个并且只能选择一个选项。
[]	在语法行中，方形方括号中包含一个可选参数。
...	在语法行中，省略号表示重复前面的参数。例如， <code>option[,...]</code> 表示您可以输入多个逗号分隔的选项。
< >	在命名约定中，尖括号中包含了某个名称的个别元素，以便将它们相互区别开，就象在 <code><server_name><connector_name>tmp.log</code> 。
/, \	在本文档中，反斜杠 (\) 用于表示目录路径。对于 UNIX 安装，用斜杠 (/) 代替反斜杠。所有 IBM WebSphere InterChange Server 产品路径名都是相对路径，相对于 IBM WebSphere InterChange Server 产品在系统上的安装目录。
<code>%text%</code> 和 <code>\$text</code>	在百分号 (%) 中的 <code>text</code> 指出 Windows <code>text</code> 系统变量或用户变量的值。在 UNIX 环境中等价的符号表示法是 <code>\$text</code> ，指出 <code>text</code> UNIX 环境变量的值。
<code>ProductDir</code>	表示安装产品的目录。

相关文档

与本产品一同提供的完整文档集包括安装、配置、管理以及使用 WebSphere Business Integration Connect Enterprise Edition 和 Advanced Edition 的综合信息。

您可以从下面的站点下载该文档，也可以直接在该站点上在线阅读：
<http://www.ibm.com/software/integration/wbiconnect/library/infocenter>

关于本产品的重要信息可以在本产品发布后所发行的技术支持 Technote 和 Flash 中找到。这些信息可以在 WebSphere Business Integration 支持 Web 站点中找到，URL 是：<http://www.ibm.com/software/integration/websphere/support>。选择您感兴趣的组件区域并浏览 Technote 和 Flash 部分。

第 1 部分 后端集成简介

第 1 章 规划后端集成

本章描述了如何规划 IBM WebSphere Business Integration Connect 和后端系统的集成。在中心社区之内，您会与社区参与者交换业务文档。交换这些文档的目的是交流信息，通常涉及处理数据和返回结果。当您从社区参与者那里接收到数据时，通常会在您公司的后端系统中处理这些数据。

本章提供了下列关于后端集成的常规信息：

- 『后端集成概述』
- 第 5 页的『规划后端集成』
- 第 21 页的『消息处理』

后端集成概述

通过 Business Integration Connect，您可以与社区参与者交换业务文档。交换这些文档的目的是交流信息，通常涉及处理数据和返回结果。当您从社区参与者那里接收到数据时，通常会在您公司的后端系统中处理这些数据。WebSphere Business Integration Connect 充当中心社区里企业之间收发信息的路由点。

为了和企业通信，要涉及以下贸易社区的组件：

- **社区管理者**是处理从社区参与者到后端系统的消息发送和接收的参与者。

社区管理者按如下方式使用 WebSphere Business Integration Connect Enterprise Edition 或 Advanced Edition：

- Business Integration Connect Enterprise Edition 充当企业的社区管理者。
- Business Integration Connect Advanced Edition 可以充当它自身小型中心社区的社区管理者和较大中心社区的社区参与者。

中心管理员（Hub Admin）指派中心社区的一名参与者作为社区管理者。

- **后端系统**管理社区管理者和企业之间的消息，如下所示：
 - 可以将消息通过后端系统从企业发送至社区管理者。在这种情形下，后端系统生成文档。Business Integration Connect 检测到该文档并将其路由到相应的社区参与者。
 - 消息通过社区管理者发送到后端系统，从而实现将消息路由到企业。在这种情形下，社区参与者生成文档。Business Integration Connect 检测该文档并把它路由到后端系统，后端系统处理该文档，可能将它路由到企业内的其它目的地。

通过 Business Integration Connect 连接的后端系统来访问企业。Business Integration Connect 的所有版本都提供了连接到后端系统的能力。这些版本的区别在于所支持的传输协议不同，如下所示：

- Business Integration Connect - Express 提供基于文件的集成。
- Business Integration Connect Enterprise Edition 和 Advanced Edition 提供基于文件的集成。另外，它们提供了通过 HTTP、HTTPS 和 JMS 协议进行的集成。

除了 RosettaNet 之外，在社区参与者和 Business Integration Connect 之间交换的文档还可以采用其它各种不同的格式。文档可以是 SOAP、cXML、XML、EDI 以及二进制格式。管理员指南列出了受支持的全部文档类型及可以用来发送文档的传输协议（例如，HTTP）。

请考虑下面的示例 - 一个社区参与者向 Business Integration Connect (Enterprise Edition 或 Advanced Edition) 上相应的目标发送了一个 RosettaNet 格式的采购订单（该订单打算提供给社区管理者）。社区管理者有一个处理采购订单的后端系统并且希望接收 RosettaNet Service Content (RNSC) 格式的订单。当社区参与者和社区管理者之间的连接建立起来之后，它们达成下面的一致：

- 文档将从 RosettaNet 格式转换成 RNSC 格式。
- 路由到后端系统的文档具有后端集成封装，这意味着要将传输级头添加到该文档中以传递交换所需的信息。

然后后端系统可以处理该文档。

可在 Business Integration Connect 和社区管理者的后端系统之间交换的文档以及与文档相关的传输类型将在第 24 页的表 15 和第 28 页的表 20 中说明。

图 1 说明了 Business Integration Connect 如何使用后端集成接口与社区管理者的后端系统通信。注意，所有箭头都是双向的；也就是说，文档可以源自社区管理者的后端系统。

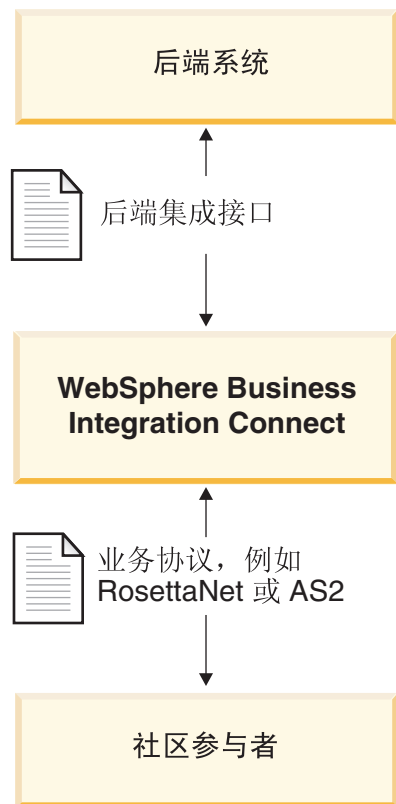


图 1. 业务协议和封装在文档流中的角色

规划后端集成

本节提供了下列信息，说明了如何利用 WebSphere Business Integration Connect 规划您的后端集成：

- 『使用何种业务协议？』
- 第 9 页的『您将使用哪种封装？』
- 第 16 页的『您将使用哪种消息传输？』
- 第 20 页的『如何访问您的后端应用程序？』

使用何种业务协议？

消息的业务协议决定了文档格式。业务协议会影响您在规划后端系统集成时必须制定的许多决策。业务协议的选择决定了必须使用的封装方法，它反过来会影响可使用的消息传输协议。

要获取业务协议的全面描述，请参阅**管理向导**。本节描述了针对以下各业务协议的集成信息：

- 『Web Service (SOAP)』
- 『cXML』
- 第 6 页的『RosettaNet』

Web Service (SOAP)

Business Integration Connect 使中心成员可以使用以下 Web Service：

- 社区参与者可以使用社区管理者提供的 Web Service。

必须向社区参与者提供 Business Integration Connect 生成的公共 WSDL。值得注意的是，社区参与者调用 Web Service 的 URL 是上载该 Web Service 时指定的公共 URL。Business Integration Connect 充当一个代理。它从参与者那接收一个 SOAP 消息并确定对应的专用 Web Service。然后它使用相同的 SOAP 消息调用专用的 Web Service（由社区管理员提供）。接着将社区管理者返回的响应返回给参与者。

- 社区管理者可以使用社区参与者提供的 Web Service。

值得注意的是，多个合作伙伴可提供同一个 Web Service 接口。Business Integration Connect 使社区管理者通过在上载 Web Service 时控制台中指定的 Web Service URL 使用这个 Web Service。另外，社区管理者必须提供 URL 参数来标识“接收方合作伙伴”。请参阅**中心配置指南**获取更详细的信息。Business Integration Connect 充当一个代理。它接收来自社区管理者的 SOAP 消息并确定对应的 Web Service 和“接收方合作伙伴”。然后它使用相同的 SOAP 消息调用合作伙伴提供的 Web Service。接着将合作伙伴返回的响应消息返回给社区管理者。

请参阅**中心配置指南**以获取更多信息，包括如何为 Web Service 设置文档流定义。

cXML

您可以在社区参与者之间收发 cXML 文档。当 Business Integration Connect 接收到来自社区参与者的 cXML 文档时，在把该文档发送到社区管理者的后端系统前，它验证该文档并进行转换（如果指定的话）。注意转换不能用来同步 cXML 消息。在同步交换时，后端系统生成一个响应，Business Integration Connect 将该响应返回给社区参与者（如果与消息相对应）。

需要发送 cXML 文档的社区管理者的后端系统可以执行以下两个操作之一：

- 生成并发送 cXML 文档，Business Integration Connect 将该文档传递给社区参与者
- 生成并发送 XML 文档，Business Integration Connect 在将该文档发送给社区参与者之前，将其转换成 cXML

注：如果采用了 XML 文档转换，对于社区参与者的同步请求 / 响应，则会将响应异步地返回到后端系统。

请参阅**管理员指南**以获取更多信息，包括如何为 cXML 设置文档流定义。

RosettaNet

如果 RosettaNet 消息具有后端集成封装（也就是说，它们必须具有传输级头），则 Business Integration Connect 可以提供对 RosettaNet 1.1 和 2.0 的支持。这些消息必须使用 HTTP 或 JMS 传输协议。传输级头保留了元信息（它们不属于 PIP），使得 Business Integration Connect 可以正确地路由消息。

例如，假设一个应用程序希望使用 RosettaNet 通过 HTTP 向社区参与者发送消息。该应用程序提供了 RosettaNet 服务内容，并添加了传输级头。该头标识出哪个社区参与者将处理该请求、将发送哪个 PIP、PIP 版本以及其它信息。该消息能让 Business Integration Connect 向社区参与者发送正确的 PIP。

可以在**中心配置指南**中找到关于设置 RosettaNet 支持和配置 PIP 的信息。

事件通知：因为 Business Integration Connect 把应用程序与社区参与者中分离开（社区参与者是 RosettaNet 服务提供者），所以 Business Integration Connect 提供了事件通知。例如，如果 Business Integration Connect 不能向参与者发送 PIP，则**事件通知**会让 Business Integration Connect 通知应用程序。然后该应用程序可以处理故障。

事件通知消息是一个 XML 文档，它含有关于发生在 Business Integration Connect 或应用程序内的事件的消息。这些消息的结构与其它任何 Business Integration Connect 所收发消息的结构一样；也就是说它们有传输级头和有效负载。可以选择将 Business Integration Connect 配置成发送（或不发送）事件通知消息。

表 1 总结了 Business Integration Connect 可以发送到后端系统的事件通知消息。

表 1. 发送到后端系统的事件通知消息

事件条件	事件通知消息
Business Integration Connect 向社区参与者传递一个 RosettaNet 文档，并接收一个“接收应答”。	事件 100
Business Integration Connect 通过生成 0A1 消息并将其传递给社区参与者来取消 PIP。	事件 800
Business Integration Connect 收到来自社区参与者的“接收应答”异常或常规异常。	事件 900

如果将 Business Integration Connect 配置成使用“排除列表管理”来发送消息，则当它处理任何其它 PIP 时，它可以向目的地应用程序发送 0A1 消息。请参阅**管理员指南**中的“管理排除列表”。

应用程序可以向 Business Integration Connect 发送一个事件通知消息来取消 RosettaNet PIP。

事件消息结构: 事件通知消息有标准的传输级头, 并带有被设置为 XMLEvent 的 x-aux-process-type 字段。但是, 消息的有效负载有特定的结构, 如图 2 中所示的样本 XML 模式。

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace=
    "http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification"
  xmlns:evntf=
    "http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification"
  elementFormDefault="qualified">
  <!-- EventNotification version 1.0 document element -->
  <xsd:element name="EventNotification">
    <xsd:complexType>
      <xsd:all>
        <xsd:element ref="evntf:StatusCode"/>
        <xsd:element ref="evntf:StatusMessage"/>
        <xsd:element ref="evntf:EventMessageID"/>
        <xsd:element ref="evntf:BusinessObjectID"/>
        <xsd:element ref="evntf:GlobalMessageID"/>
        <xsd:element ref="evntf:Timestamp"/>
      </xsd:all>
    </xsd:complexType>
  </xsd:element>
  <!-- StatusCode element -->
  <xsd:element name="StatusCode">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="100"/>
        <xsd:enumeration value="800"/>
        <xsd:enumeration value="900"/>
        <xsd:enumeration value="901"/>
        <xsd:enumeration value="902"/>
        <xsd:enumeration value="903"/>
        <xsd:enumeration value="904"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <!-- StatusMessage element -->
  <xsd:element name="StatusMessage">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>
  </xsd:element>
```

图 2. 事件通知消息的样本 XML 模式 (1/2)

```

<!-- EventMessageID element -->
  <xsd:element name="EventMessageID">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>
  </xsd:element>
<!-- BusinessObjectID element -->
  <xsd:element name="BusinessObjectID">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>
  </xsd:element>
<!-- GlobalMessageID element -->
  <xsd:element name="GlobalMessageID">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>
  </xsd:element>
<!-- Timestamp element -->
  <xsd:element name="Timestamp">
    <xsd:simpleType>
      <xsd:restriction base="xsd:dateTime"/>
    </xsd:simpleType>
  </xsd:element>
</xsd:schema>

```

图 2. 事件通知消息的样本 XML 模式 (2/2)

表 2 描述在事件有效负载里的各个字段。

表 2. 事件通知 XML 字段

字段	描述
StatusCode	消息类型。有效值为： <ul style="list-style-type: none"> • 100 - Business Integration Connect 已经传递了该文档并接收到“接收应答”。 • 800 - 应用程序取消了 PIP。 • 900 - Business Integration Connect 接收到来自社区参与者的“接收应答”异常、常规异常或 0A1 故障 PIP。
StatusMessage	事件通知消息的字母数字描述
EventMessageID	特定事件通知消息的字母数字标识
BusinessObjectID	消息的传输级头里的 x-aux-msg-id，它受该消息通知事件的影响。因此会把原始消息的有效负载链接到该事件。
GlobalMessageID	消息的传输级头里的 x-aux-system-msg-id，它会引起该消息通知事件。
Timestamp	在事件发生时使用的 UTC 时间戳记格式： CCYY-MM-DDThh:mm:ssZ 包括小数精度的秒数 (...ss.ssssZ)。日期时间戳记必须符合表示 dateTime 的 XML 模式数据类型 (w3.org/TR/2001/REC-xmlschema-2-20010502#dateTime)

事件通知消息示例: 图 3 显示了使用 HTTP 协议发送事件通知消息的示例。

```

POST /builderURL HTTP/1.1
Content-Type: application/xml
Content-length: 250
x-aux-sender-id: 000000001
x-aux-receiver-id: 000000002
x-aux-third-party-bus-id: 000000003
x-aux-create-datetime: 2002-10-28T23:05:02Z
x-aux-protocol: XMLEvent
x-aux-protocol-version: 1.0
x-aux-process-type: XMLEvent
x-aux-process-version: 1.0
x-aux-payload-root-tag: evtntf:EventNotification
x-aux-msg-id: 98732
x-aux-system-msg-id: 12345
x-aux-production: Production
x-aux-process-instance-id: 3456
x-aux-event-status-code: 100
x-aux-transport-retry-count: 0

<?xml version="1.0" encoding="UTF-8"?>
<evtntf:EventNotification xmlns:evtntf=
"http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification">
  <evtntf:StatusCode>100</evtntf:StatusCode>
  <evtntf:StatusMessage>The message was delivered</evtntf:StatusMessage>
  <evtntf:EventMessageID>12345</evtntf:EventMessageID>
  <evtntf:BusinessObjectID>34234</evtntf:BusinessObjectID>
  <evtntf:GlobalMessageID>98732</evtntf:GlobalMessageID>
  <evtntf:Timestamp>2001-01-31T13:20:00Z</evtntf:Timestamp>
</evtntf:EventNotification>

```

图 3. 使用 HTTP 的事件通知消息的示例

您将使用哪种封装？

封装类型决定了 Business Integration Connect 向后端系统发送消息所使用的格式。

您可以使用社区控制台来建立与社区参与者之间的连接，并且指定 Business Integration Connect 和后端系统之间使用的封装。要确定使用何种封装类型，必须了解下列问题：

- 对于后端系统，使用哪种封装类型是有效的？
- 对于使用特定业务协议的消息，哪种封装类型是有效的？

要获取更多关于如何设置合作伙伴连接的信息，请参阅[中心配置指南](#)。

用于集成的有效封装类型

当使用 Business Integration Connect 进行集成时，不是所有的封装类型都是有效的。表 3 列出了 Business Integration Connect 充当社区管理者时有有效的封装类型。

表 3. 用于后端集成的相关封装类型

封装类型	描述
“无”封装	使 Business Integration Connect 向后端系统发送的消息中无任何头数据
“后端集成”封装	向消息头添加额外的属性，并且可选地，将消息内容包装到 XML 传输包络中。

注：其它封装类型（例如 AS）也可用于 Business Integration Connect。但是，为了与后端系统集成，只推荐使用“无”和“后端集成”封装类型。

“无”封装： 当封装设为“无”时，Business Integration Connect 既不在它将消息发送给后端系统时添加一个传输级头，也不期望它接收来自后端系统消息时候有一个传输级头。Business Integration Connect 仅向后端系统发送消息。文档控制路由表中的信息。

后端集成封装： 当封装设为“后端集成”时，后端系统收发的消息必须含有下面组件：

- 传输级头，它包含了关于消息的元信息
- 有效负载，它包含消息的内容
- 附件（可选）

头和有效负载是必需的部分，而附件是可选的。下面各节描述了使用“后端集成”所封装文档的每个组件。

传输级头的内容： 传输级头包含 Business Integration Connect 用来处理消息以及将消息路由到正确目的地的信息。传输级头是双向的，从而 Business Integration Connect 收发的所有消息都含有必需的字段和任何适用的可选字段。

表 4 列出了传输级头的各字段。

表 4. 传输级头字段

头字段	描述	是否需要？
x-aux-sender-id	消息发送者的标识，例如 DUNS 号。	是
x-aux-receiver-id	消息接收者的标识，例如 DUNS 号。	是
x-aux-protocol	消息内容的协议。有效值包括用于 RosettaNet 服务内容的 RNSC、XMLEvent 和二进制。对于 Business Integration Connect，该字段的值的优先级高于有效负载中的任何协议字段。	是
x-aux-protocol-version	消息内容协议的版本。	是
x-aux-process-type	要执行的过程或正被发送消息的类型。对于 RosettaNet 消息，这是 PIP 代码（例如 3A4）。对于事件消息，这是 XMLEvent，而对于二进制消息，这是二进制。对于 Business Integration Connect，该字段的值的优先级高于有效负载中的任何过程字段。	是
x-aux-process-version	过程版本。对于 RosettaNet 消息，这是 PIP 的版本号。	是
x-aux-create-datetime	使用 UTC 时间戳记格式成功发送消息的时间（CCYY-MM-DDThh:mm:ssZ）	
x-aux-msg-id	有效负载内容的标识。例如，它可以是 RosettaNet 消息的 RNPIPServiceContent 实例的标识，也可以是专有文档的标识。为了跟踪目的，这个标识将消息有效负载与消息发送者系统的某些内容相链接。	
x-aux-production	消息的路由。有效值为：Production 和 Test，在双向请求中都应填充该值。请注意，当消息是社区参与者发出的对双向 PIP 的响应时，Business Integration Connect 在请求中使用 GlobalUsageCode 并忽略传输级头中的值。	
x-aux-system-msg-id	消息的全局唯一标识（GUID），它用于重复检查。	是
x-aux-payload-root-tag	有效负载的根标记元素。例如，对于 3A4 RosettaNet 服务内容，该字段的值为 Pip3A4PurchaseOrderRequest。对于事件通知消息，该字段的值为 EventNotification。	
x-aux-process-instance-id	标识，它将多个消息业务流程中的文档链接到唯一过程实例。对于 RosettaNet，它必须对最近的 30 天内的 RosettaNet 过程是唯一的。所有交换消息作为 RosettaNet 过程实例的一部分，包括重试，都使用相同的过程实例标识。	

表 4. 传输级头字段 (续)

头字段	描述	是否需要?
x-aux-event-status-code	事件通知的状态码。请参阅第 7 页的『事件消息结构』中的 <code>Status Code</code> 字段。	
x-aux-third-party-bus-id	标识, 例如传递消息的一方的 DUNS 号。如果第三方代表社区所有者托管 Business Integration Connect, 则该标识可以不同于 <code>x-aux-sender-id</code> 和 <code>x-aux-receiver-id</code> 。	
x-aux-transport-retry-count	在这次尝试前, 试图发送消息的不成功次数。如果第一次尝试就成功地发送了消息, 则该字段值为 0。	
content-type	消息的内容类型。	
content-length	消息长度 (以字节计)。	

注: 为了与 IBM WebSphere MQ (JMS 提供程序) 兼容, JMS 协议消息的字段使用下划线代替连字符。例如, 在 JMS 消息中, 使用 `x_aux_sender_id` 字段而不是 `x-aux-sender-id` 字段。

表 4 概述了传输级头信息。下列各节提供了针对特定业务协议的传输级头信息:

- 『传输级头和 RosettaNet 消息』
- 第 12 页的『传输级头和 AS2 消息』
- 第 13 页的『传输级头和 AS1 消息』

传输级头和 RosettaNet 消息: 表 5 描述了 Business Integration Connect 从 RosettaNet 消息获取传输级头字段值的位置。

表 5. 传输级头字段和 RosettaNet 内容

头字段	值的来源: RosettaNet 2.0	值的来源: RosettaNet 1.1
x-aux-sender-id	<pre><(DeliveryHeader)> <messageSenderIdentification> <PartnerIdentification> <GlobalBusinessIdentifier></pre>	<pre><ServiceHeader> <ProcessControl> <TransactionControl> <ActionControl> 或 <SignalControl> <PartnerRouter> <fromPartner> <PartnerDescription> <BusinessDescription> <GlobalBusinessIdentifier></pre>
x-aux-receiver-id	<pre><(DeliveryHeader)> <messageReceiverIdentification> <PartnerIdentification> <GlobalBusinessIdentifier></pre>	<pre><ServiceHeader> <ProcessControl> <TransactionControl> <ActionControl> 或 <SignalControl> <PartnerRouter> <toPartner> <PartnerDescription> <BusinessDescription> <GlobalBusinessIdentifier></pre>
x-aux-protocol	设置用于 RosettaNet 的值: RNSC	和 RosettaNet 2.0 的一样
x-aux-protocol-version	设置值: 1.0	和 RosettaNet 2.0 的一样
x-aux-process-type	源 XPath 为:	源 XPath 为:
	<code>/ServiceHeader/ProcessControl/pipCode/GlobalProcessIndicatorCode</code>	<code>/ServiceHeader/ProcessControl/ProcessIdentity/GlobalProcessIndicatorCode</code>
x-aux-process-version	源 XPath 为:	源 XPath 为:
	<code>/ServiceHeader/ProcessControl/pipVersion/VersionIdentifier</code>	<code>/ServiceHeader/ProcessControl/ProcessIdentity/VersionIdentifier</code>
	各 PIP 的版本标识值使用其 PIP 规范里。	各 PIP 的版本标识值使用其 PIP 规范里。

表 5. 传输级头字段和 RosettaNet 内容 (续)

头字段	值的来源: RosettaNet 2.0	值的来源: RosettaNet 1.1
x-aux-payload-root-tag	PIP 名称, 例如 Pip3A4PurchaseOrderRequest	和 RosettaNet 2.0 的一样
x-aux-process-instance-id	对于应用程序启动的过程, 该值是这个过程实例的标识。对于不是通过工作流的社区参与者启动的过程, 该值是初始 RosettaNet 请求中的过程标识: <ServiceHeader> <ProcessControl> <pipInstanceId> <InstanceIdentifier>	<ServiceHeader> <ProcessControl> <ProcessIdentity> <InstanceIdentifier>
x-aux-msg-id	<(RNPIPServiceContent)> <thisDocumentIdentifier> <ProprietaryDocumentIdentifier>	和 RosettaNet 2.0 的一样
x-aux-production	<ServiceHeader> <ProcessIndicator> <GlobalUsageCode>	<Preamble> <GlobalUsageCode>

传输级头和 AS2 消息: 表 6 描述了 Business Integration Connect 从 AS2 消息获取传输级头字段值的位置。

注: 这些值是区分大小写的

表 6. 来自 AS2 内容的传输级头字段

头字段	值的来源
x-aux-sender-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 在后端集成消息的 x-aux-sender-id 字段中设置了 AS2 消息的 AS2-From 头字段 (后端集成消息将发送给社区管理者)。当 AS2 消息发送给社区参与者时, 入站的后端集成消息的 x-aux-sender-id 字段被用作 AS2 消息的 AS2-From 头值。
x-aux-receiver-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 在后端集成消息的 x-aux-receiver-id 字段中设置了 AS2 消息的 AS2-To 头字段 (后端集成消息将发送给社区管理者)。当 AS2 消息发送给社区参与者时, 入站的后端集成消息的 x-aux-receiver-id 字段被用作 AS2 消息的 AS2-To 头值。
x-aux-protocol	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 在后端集成消息的 x-aux-protocol 字段中设置了参与者连接的 ToProtocol (后端集成消息将发送给社区管理者)。当 AS2 消息发送给社区参与者时, 入站后端集成消息的 x-aux-protocol 字段被用作确定参与者连接的 FromProtocol。
x-aux-protocol-version	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 在后端集成消息的 x-aux-protocol-version 字段里设置了参与者连接的 ToProtocolVersion (后端集成消息将发送给社区管理者)。当 AS2 消息发送给社区参与者时, 入站的后端集成消息的 x-aux-protocol-version 字段被用作参与者连接的 FromProtocolVersion。
x-aux-process-type	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 在后端集成消息的 x-aux-process-type 字段里设置了参与者连接的 ToProcessCode (后端集成消息将发送给社区管理者)。当 AS2 消息发送给社区参与者时, 入站的后端集成消息的 x-aux-process-type 字段将被用作参与者连接的 FromProcessCode。
x-aux-process-version	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 在后端集成消息的 x-aux-process-version 字段里设置了参与者连接的 ToProcessVersion (后端集成消息将发送给社区管理者)。当 AS2 消息发送给社区参与者时, 入站的后端集成消息的 x-aux-process-version 字段被用作参与者连接的 FromProcessVersion。

表 6. 来自 AS2 内容的传输级头字段 (续)

头字段	值的来源
x-aux-payload-root-tag	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 向 AS 消息时, 仅对于定制的 XML 协议, 从消息中解析出的 XPATH 中指定的根标记, 并在 x-aux-payload-root-tag 字段中使用该根标记。当 AS2 消息发送给社区参与者时, 不需要在入站的后端集成消息中设置该字段。
x-aux-process-instance-id	对于 AS2, 不使用该字段。
x-aux-msg-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 仅对于定制的 XML 协议, 从消息中解析出的 XPATH 中指定的文档标识, 并在 x-aux-payload-root-tag 字段中使用该文档标识。当 AS2 消息发送给社区参与者时, 不需要在入站的后端集成消息中设置该字段。
x-aux-system-msg-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS2 消息时, 将该字段设为内部生成的该消息唯一标识。当 AS2 消息发送给社区参与者时, 不需要在入站的后端集成消息中设置该字段。
x-aux-production	对于 AS2, 不使用该字段。

传输级头和 AS1 消息: 表 7 描述了 Business Integration Connect 获取 AS1 消息中传输级头字段值的位置。

注: 这些值是区分大小写的

表 7. 来自 AS1 内容的传输级头字段

头字段	值的来源
x-aux-sender-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 在后端集成消息的 x-aux-sender-id 字段中设置 AS1 消息 “Subject: ToID;FromID” 头字段中的 FromID (后端集成消息将发送给社区管理者)。当 AS1 消息发送给社区参与者时, 入站后端集成消息的 x-aux-sender-id 字段被用作 AS1 “Subject: ToID;FromID” 头值中的 FromID。
x-aux-receiver-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 在后端集成消息的 x-aux-receiver-id 中设置 AS1 消息 “Subject: ToID;FromID” 头字段中的 ToID (后端集成消息将发送给社区管理者)。当 AS1 消息发送给社区参与者时, 入站后端集成消息的 x-aux-receiver-id 字段被用作 AS1 消息 “Subject: ToID;FromID” 头值中的 FromID。
x-aux-protocol	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 在后端集成消息的 x-aux-protocol 字段设置参与者连接的 ToProtocol (后端集成消息将发送给社区管理者)。当 AS1 消息发送给社区参与者时, 入站后端集成消息的 x-aux-protocol 字段被用作参与者连接的 FromProtocol。
x-aux-protocol-version	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 在后端集成消息的 x-aux-protocol-version 字段里设置参与者连接的 ToProtocolVersion (后端集成消息将发送给社区管理者)。当 AS1 消息发送给社区参与者时, 入站后端集成消息的 x-aux-protocol-version 字段被用作参与者连接的 FromProtocolVersion。
x-aux-process-type	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 在后端集成消息的 x-aux-process-type 字段里设置参与者连接的 ToProcessCode (后端集成消息将发送给社区管理者)。当 AS1 消息发送给社区参与者时, 入站后端集成消息的 x-aux-process-type 字段被用作参与者连接的 FromProcessCode。
x-aux-process-version	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 在后端集成消息的 x-aux-process-version 字段里设置参与者连接的 ToProcessVersion (后端集成消息将发送给社区管理者)。当 AS1 消息发送给社区参与者时, 入站后端集成消息的 x-aux-process-version 字段被用作参与者连接的 FromProcessVersion。

表 7. 来自 AS1 内容的传输级头字段 (续)

头字段	值的来源
x-aux-payload-root-tag	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 仅对于定制的 XML 协议, 从消息中解析出在 XPATH 中指定的根标记, 并在 x-aux-payload-root-tag 字段中设置该根标记。当 AS1 消息发送给社区参与者时, 不需要在入站后端集成消息中设置该字段。
x-aux-process-instance-id	对于 S21, 不使用该字段。
x-aux-msg-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 仅对于定制的 XML 协议, 从消息中解析出在 XPATH 中指定的文档标识, 并在 x-aux-payload-root-tag 字段中使用该文档标识。当 AS1 消息发送给社区参与者时, 不需要在入站后端集成消息中设置该字段。
x-aux-system-msg-id	当社区参与者向 Business Integration Connect Enterprise Edition 或 Advanced Edition 发送 AS1 消息时, 将该字段设为内部生成的该消息唯一标识。当 AS1 消息发送给社区参与者时, 不需要在入站后端集成消息中设置该字段。
x-aux-production	对于 S21, 不使用该字段。

有效负载: 消息的有效负载包含消息的实际内容。有效负载的位置取决于发送消息的传输协议, 如表 8 所示。

表 8. 有效负载的位置

传输协议	有效负载的位置
HTTP 协议消息	在 HTTP post 的正文中
JMS 协议消息	在 JMS 消息的正文中
RosettaNet 消息	来自 PIP 的服务内容
通过 AS2 发送 EDI	EDI 消息
	有效负载不和 XML 包络合并除非消息还携带了一个或多个附件。要获取有关用于封装附件的 XML 包络和标记的信息, 请参阅第 15 页的『附件』。

有效负载可以用 Base64 编码, 并且可以作为 XML 格式**传输包络**格式出现在下面任何一种情况下:

- 如果文档包含附件

包含附件的文档必须包装到 XML 传输包络中。要获取更多关于如何格式化附件的信息, 请参阅第 15 页的『附件』。

- 如果您将后端集成封装的包络标志设置为“是”

要将文档包装到 XML 传输包络而忽略它是否包含附件, 从概要文件的 B2B 功能屏幕将后端集成包络标志设为“是”。例如, 要在中心操作者的概要文件中设置这个标志, 选择:

概要文件>中心操作者> B2B 功能

对后端集成单击“编辑”以查看包络标志。

这个 XML 传输包络将文档包装到 <transport-envelope> 根标记中。在这个根标记中，有一个包含文档有效负载的 <payload> 标记。如果存在任何附件，它们都包含在各自的 <attachment> 标记中。要获取更多关于这些标记结构的信息，请参阅『附件』。

Business Integration Connect 包括下列 W3C XML 模式文件，该文件描述了后端集成 XML 传输包络的结构：

wbipackaging_v1.0_ns.xsd

该模式文件位于在安装介质上的如下目录：

B2BIntegrate\packagingSchemas

根据该模式文件，可以使用任何 XML 编辑工具来验证您的后端集成 XML 以确保在向文档管理器发送该文档前它是有效的。

附件： 如果业务协议允许有附件，则每个文档可以含有一个或多个附件。如果文档含有附件，则文档必须包装在一个 XML 传输包络中，与第 14 页的『有效负载』中描述的一样。表 9 描述了 payload 和 attachment 标记中的 XML 属性。

表 9. payload 和 attachment 标记的 XML 属性

XML 属性	描述	是否需要？
Content-Type	标识 MIME 类型 / 子类型，例 是 如 text/xml 或 image/gif。	
Encoding	标识编码。因为附件和有效负 否 载必须是用 Base64 编码的，所 以该属性的唯一有效值是 “Base64”。	

图 4 展示了包含有效负载和一个附件的 XML 传输包络中文档的示例。

注： 本示例中的名称空间是必需的：

```

xmlns="http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging"

<?xml version="1.0" encoding="utf-8"?>
<transport-envelope
  xmlns="http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging">
  <payload encoding="base64" contentType="application/xml">
    ...base64 encoded XML message...
  </payload>

  <attachment encoding="base64" Content-Type="text/xml">
    ...base64 encoded XML attachment...
  </attachment>
</transport-envelope>

```

图 4. 用于有效负载和一个附件的样本 XML 传输包络

注： 为了使用 WebSphere InterChange Server 处理包装在 XML 传输包络中的文档，Business Integration Connect 提供了附件数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

您的文档使用哪种封装类型？

某些业务协议中的文档只能使用某些类型的封装。例如，只有将封装方式指定为“后端集成”时才可以处理 RosettaNet。请参阅第 24 页的表 15 和第 28 页的表 20 以获取关于文档类型与封装类型关联信息的完整列表。

HTTP 上的后端集成封装的示例

图 5 显示了一个使用 HTTP 传输协议把消息从 Business Integration Connect 发送到应用程序的示例。请注意，该消息不包含附件。

```
POST /sample/receive HTTP/1.1
Host: sample.COM
Content-Type: application/xml
Content-Length: nnn
x-aux-sender-id: 000000001
x-aux-receiver-id: 000000002
x-aux-third-party-bus-id: 000000003
x-aux-create-datetime: 2002-10-28T23:05:02Z
x-aux-protocol: RNSC
x-aux-protocol-version: 1.0
x-aux-process-type: 3A4
x-aux-process-version: V02.00
x-aux-payload-root-tag: Pip3A4PurchaseOrderRequest
x-aux-msg-id: 1021358129419
x-aux-system-msg-id: 2
x-aux-production: Production
x-aux-process-instance-id: 123456
x-aux-transport-retry-count: 0

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Pip3A4PurchaseOrderRequest SYSTEM
    "3A4PurchaseOrderRequestMessageGuideline_v1_2.dtd">
<Pip3A4PurchaseOrderRequest>
  <PurchaseOrder>
    ...
  </PurchaseOrder>
  ...
  <thisDocumentIdentifier>
    <ProprietaryDocumentIdentifier>1021358129419
    </ProprietaryDocumentIdentifier>
  </thisDocumentIdentifier>
  <GlobalDocumentFunctionCode>Request</GlobalDocumentFunctionCode>
</Pip3A4PurchaseOrderRequest>
```

图 5. 使用 HTTP 传输协议的样本消息

您将使用哪种消息传输？

当后端系统和 WebSphere Business Integration Connect 相互发送消息时，双方必须使用相同的消息传输协议。消息传输协议定义了发送消息时所用的通信协议。

Business Integration Connect 通过它的后端集成接口与后端系统进行通信。表 10 列出了该后端集成接口支持的传输协议。

表 10. Business Integration Connect 支持的传输协议

传输协议	要获取更多信息
HTTP 或 HTTPS 文件系统文件	『 HTTP 传输协议 』 第 19 页的 『 Enterprise Edition 和 Advanced Edition 的文件系统协议 』
JMS	第 18 页的 『 JMS 协议 』

请参阅第 24 页的表 15 和第 28 页的表 20 以获取关于哪些传输协议可用于消息内容和后端集成封装特定组合的信息。

HTTP 传输协议

为了使用 HTTP 协议发送消息，Business Integration Connect 使用了 HTTP/S 1.1。为了接收来自后端系统的消息，Business Integration Connect 既支持 HTTP/S V1.0 也支持 V1.1。

HTTP 消息可以包括集成封装属性。是否包括这些属性取决于与参与者连接相关的封装类型，如下所述：

- 如果参与者连接指定了 HTTP 消息包含后端集成封装，那么 HTTP 消息的传输级头包括额外的属性，这些属性包含关于消息的信息，例如内容的协议、消息的标识以及消息的发送方。要获取头中字段的完整列表，请参阅第 10 页的 『 传输级头的内容 』。

EDI、SOAP 和 cXML 消息必须使用“无”封装。

- 如果参与者连接指定了“无”封装，那么 HTTP 消息不包含这些额外的属性，并且 Business Integration Connect 会对消息进行解析以获得该信息。

RosettaNet 消息在需要时必须使用后端集成封装。

注：XML 消息既可以使用“无”也可以使用“后端集成”封装。从后端系统接收的二进制消息必须包含后端集成封装；但是，反之则不然，因为 Business Integration Connect 支持使用任一种封装方式向应用程序发送二进制消息。

过程： 当在 Business Integration Connect 和应用程序之间发送 HTTP 或 HTTPS 消息以进行异步交换时，会发生下面的步骤：

1. 源系统（Business Integration Connect 或后端系统）使用特定的 URL 将 HTTP 消息发布到目标系统。
2. 目标系统接收到消息并发送协议级应答 - HTTP 200 或 202，以表示所有权的变更。源系统忽略该应答消息的主体。如果在该过程中发生错误，目标系统将 HTTP 500 发送回源系统。
3. 如果 Business Integration Connect 是目标系统（换句话说，当 Business Integration Connect 接收消息时），它会持久存储消息并释放与源系统的连接。
4. 然后目标系统可以异步地处理消息。

如果交换是同步（例如，对于 SOAP 或 cXML），在相同的 HTTP 连接中，会返回一个带 HTTP 200 消息的响应。

使用 HTTP 协议发送和接收消息： 要使用 HTTP 协议将消息发送到 Business Integration Connect，后端系统将采用如下步骤：

1. 创建消息。

传输级头中的 Content-Type 属性定义了消息中使用的编码。

2. 根据为连接设置的封装类型对消息进行封装。

对于后端集成封装，后端系统添加了 Business Integration Connect 所需的协议头属性。

3. 将消息发布到 Business Integration Connect 用于接收这些消息的 URL。

4. 如果交换是同步的，则后端系统将等待接收响应，该响应所用的连接与用于请求的连接是相同的。

要在这个方向启用 HTTP 消息交换，使用社区控制台的“目标详细信息”屏幕来为入站文档设置一个目标。要了解更多信息，请参阅第 27 页的『从后端系统接收文档』。

要使用 HTTP 协议从 Business Integration Connect 接收消息，后端系统将采用如下步骤：

1. 在特定的 URL 上侦听消息。

2. 当接收到消息时，处理该消息：

- 如果连接含有“无”封装，则后端系统必须解析该消息以确定如何处理它。
- 如果连接含有“后端集成”封装，则应用程序可以使用后端集成属性来确定如何处理该消息。

3. 如果交换是同步的，则后端系统将返回一个响应，该响应所用的连接与用于请求的连接是相同的。

要在这个方向启用 HTTP 消息交换，使用社区控制台的“网关”屏幕来为出站文档设置一个网关。要了解更多信息，请参阅第 23 页的『向后端系统发送文档』。

JMS 协议

JMS 协议基于 Java 消息服务 (JMS)，它通过事务性的和持久的 JMS 队列来传递消息，JMS 队列是由 WebSphere MQ 之类的工具提供的。JMS 协议支持下列 JMS 消息类型：

- StreamMessage (作为字节数组)
- BytesMessage (作为字节数组)
- TextMessage

在 JMS 协议中，发送系统使用入队操作来把 JMS 消息发送到接收系统。接收系统从队列里取得消息，持久保存消息，然后执行出队操作以从队列中除去消息。从这时起，接收系统可以异步地处理消息。

JMS 消息可以包括集成封装属性。是否包括这些属性取决于与参与者连接相关的封装类型，如下所述：

- 如果参与者连接指定 JMS 消息包括后端集成封装，则 JMS 消息在消息中包含传输级信息 (例如内容的协议、消息标识以及消息发送方) 作为 JMS 属性。要获取属性的完整列表，请参阅第 10 页的『传输级头的内容』。

注：为了与 WebSphere MQ JMS 兼容，JMS 消息中的属性在属性名中使用下划线而不是连字符。例如，在 JMS 消息中，属性是 x_aux_system_msg_id，而等价的

HTTP 头字段是 `x-aux-system-msg-id`。当 Business Integration Connect 处理 JMS 消息时，它会将这些属性中的下划线转换成连字符。

- 如果参与者连接指定了“无”封装，那么 JMS 消息不包含这些额外的属性。

除二进制消息外，Business Integration Connect 支持在需要时发送和接收任何一种封装类型的 JMS 消息。从应用程序接收到的二进制消息必须含有后端集成封装。反之则不然，因为 Business Integration Connect 支持使用任何一种封装类型相应用程序发送二进制消息。

使用 JMS 协议发送消息： 要使用 JMS 协议将消息发送到 Business Integration Connect，后端系统将采用如下步骤：

1. 创建消息。

`content_type` 头属性设置消息的内容类型，而 `content_length` 头属性指定消息的长度（以字节计）。

2. 根据为连接设置的封装类型对消息进行封装。

对于后端集成封装，应用程序添加必需的 JMS 头属性。

3. 将消息发送到后端系统用于将消息发送到 Business Integration Connect 的 JMS 队列。

要在这个方向启用 JMS 消息交换，使用社区控制台的“目标详细信息”屏幕来为入站文档设置一个目标。要了解更多信息，请参阅第 27 页的『从后端系统接收文档』。

使用 JMS 协议接收消息： 要使用 JMS 协议从 Business Integration Connect 接收消息，后端系统将采用如下步骤：

1. 在 JMS 队列上侦听消息。

2. 当接收到消息时，处理该消息：

- 如果连接含有“无”封装，则后端系统必须解析该消息以确定如何处理它。
- 如果连接含有“后端集成”封装，则应用程序可以使用后端集成属性来确定如何处理该消息。

要在这个方向启用 JMS 消息交换，使用社区控制台的“网关”屏幕来为出站文档设置一个网关。要了解更多信息，请参阅第 23 页的『向后端系统发送文档』。

Enterprise Edition 和 Advanced Edition 的文件系统协议

文件系统协议使得 Business Integration Connect 能够通过把消息放入已定义的目录结构中来发送消息。Business Integration Connect 通过从目录结构中读取消息来接收它们。文件系统协议支持下列各项：

- 文档类型：EDI 和 XML 文档
- 集成封装：仅“无”封装类型；即，文件不能包含额外属性。

使用文件系统协议发送消息： 要使用文件系统协议将消息发送到 Business Integration Connect，应用程序必须采用下列步骤：

1. 在临时目录中创建消息文件。

2. 一旦文件准备就绪，就把它移到由 Business Integration Connect 轮询的目录。

要在这个方向启用文件系统消息，使用社区控制台的“目标详细信息”屏幕来为入站文档设置一个目标。消息的目标决定了 Business Integration Connect 轮询的目录。当您创建目标时，Business Integration Connect 会为该目标创建一个 Documents 目录及其子目录，如下所示。

```
<doc_root>
  Documents
    Production
    Test
  <other destination types>
```

Business Integration Connect 定期轮询 Documents 目录及其子目录以检测消息文件。如果它发现消息，则 Business Integration Connect 持久存储该消息，然后从目录中删除该消息。然后 Business Integration Connect 按正常方式处理该消息。请参阅中心配置指南以获取关于如何创建目标的信息。

使用文件系统协议接收消息： 要使用文件系统协议接收消息，应用程序应当执行如下步骤：

1. 轮询相应的目录以获取消息文件。

注： 临时文件（那些扩展名为 .tmp 或 .tmp1 的文件）应该被忽略。应用程序决不能选取或删除这些临时文件。

2. 当发现消息时，持久保存它。
3. 从目录中删除该消息。
4. 处理该消息

要在这个方向下启用文件系统消息交换，使用社区控制台的“网关”屏幕来为出站文档设置一个网关。Business Integration Connect 将消息文件放入网关定义的 Documents 目录。通过根据网关定义目的地目录，每个参与者连接都可以拥有一个不同的目录。要获取关于网关的信息，请参阅中心配置指南。

如何访问您的后端应用程序？

Business Integration Connect 提供了与许多不同后端应用程序集成的能力。通常，通过后端系统访问后端应用程序集成代理程序。在本指南中包含了在表 11 中列出的与后端系统的集成。

表 11. Business Integration Connect 支持的后端系统

后端系统	要获取更多信息
WebSphere InterChange Server	第 33 页的第 2 章, 『InterChange Server 集成简介』
WebSphere Business Integration Message Broker	第 117 页的第 5 章, 『与 WebSphere Business Integration Message Broker 集成』
WebSphere Data Interchange	第 133 页的第 6 章, 『与 WebSphere Data Interchange 集成』

消息处理

本节描述 Business Integration Connect 如何处理下列影响消息传递的情形:

- 『排队的传递』
- 『通信错误处理』
- 『重复消息』

排队的传递

Business Integration Connect 将有关它希望发送到特定网关的所有文档的信息发布进队列。传递管理器系统按队列接收这些消息的顺序 (FIFO) 来处理它们, 并为每个消息使用一个线程来发送它们。注意, 如果网关 (例如, 传输协议是 HTTP 时的 URL, 或传输协议是 JMS 时的 JMS 目的地) 被配置为脱机的 (请参阅通信错误处理), 消息将保存在队列中直到网关被启用 (联机)。如果传递管理器接收到某个线程中的错误, 那么它将使其它线程停止尝试传递消息。传递管理器把这些消息放回队列, 直到它可以传递引起错误的那个消息。

如果失败尝试的次数超过最大尝试次数, 则文档管理器将把消息放入失败目录, 然后尝试传递队列中的下一条消息, 除非网关处于脱机状态。

通信错误处理

当 Business Integration Connect 是发送者并且应用程序返回一个错误 (例如, 当使用 HTTP 协议时, HTTP 响应消息不是 200 或 202 消息) 时, 然后 Business Integration Connect 可以重新尝试再次发送消息, 这取决于如何为这个特定网关配置该协议的。每个网关 (对于 HTTP, 指定 URL) 含有下列选项, 这些选项影响重试次数和消息的发送方式:

表 12. 网关配置选项

配置选项	描述
重试计数	当接收到错误时, 重新尝试发送文档的次数
重试时间间隔	重试尝试之间的时间间隔
联机 / 脱机	启动和停止传递尝试
线程数	每个网关上处理消息的发布线程数

如果 Business Integration Connect 没有配置成重试发送消息或所有传递尝试失败, Business Integration Connect 通过执行下列任何或全部操作来发出问题信号:

- 在社区控制台的各个屏幕 (如文档查看器和 RosettaNet 查看器) 中显示错误
- 如果已经设置了传递失败事件电子邮件警报, 则会向相应的人员发送电子邮件以通知他们该问题, 这样他们可以采取相应的操作
- 创建事件文档, 然后将文档发送给接收者。

请参阅管理员指南中的“管理网关配置”, 以获取更多信息。

重复消息

Business Integration Connect 收发的所有消息必须有全局唯一标识 (GUID)。Business Integration Connect 使用 GUID 来检测重复消息。当使用后端集成封装时, 每个消息在

传输级头中包含其 GUID。例如，对于 HTTP 协议，在 x-aux-system-msg-id 字段中包含 GUID（请参阅第 10 页的『传输级头的内容』）。消息发送方生成 GUID。文件系统协议不支持重复消息的检查。

如果发送消息的尝试导致错误，Business Integration Connect 每次重试时会重用该消息的 GUID。如果 Business Integration Connect 接收了包含重复 GUID 的消息，它返回一个肯定的应答（例如，HTTP 200）但不处理该重复消息。

注：如果使用的是 RosettaNet，Business Integration Connect 在 RosettaNet 过程级别上检查重复消息。如果使用的是 XML，也会检查重复消息。

创建传输协议机制

对于要进行通信的 Business Integration Connect 和后端系统，您必须选择这两个后端集成组件都可以支持的传输协议。表 13 汇总了本指南提供的关于受支持传输协议的信息。

表 13. 如何获取关于受支持的传输协议的信息

后端集成组件	要获取更多信息
Business Integration Connect 后端系统	第 23 页的『配置 Business Integration Connect』 请参阅本指南相应的章节，如第 20 页的『如何访问您的后端应用程序？』所列出的。

对于这两个要进行通信的后端集成组件，您必须确保**传输协议机制**存在；即，要使这两个组件能通过期望的传输协议通信，相应的软件和硬件实体必须存在。本节为下列传输协议汇总了创建传输协议机制的步骤：

- HTTP/S
- JMS
- 文件系统

HTTP/S 传输协议机制

要通过 HTTP 或 HTTPS 传输协议进行通信，Business Integration Connect 和后端系统需要一个两者都可以访问的 URL 地址。因此，您必须提供一个 URL 地址。该地址必须具有如下格式：

```
bcgreceiver
```

JMS 传输协议机制

要利用 JMS 传输协议进行通信，Business Integration Connect 和后端应用程序需要通信的每个方向都有一个 JMS 队列。因此，您必须采取下列步骤来提供相应的 JMS 队列：

- 配置您的 JMS 环境。
- 创建队列管理器和所需的队列，包括传输队列、远程队列和接收方队列。
- 创建发送方和接收方通道。

JMS 队列管理器可以存在于任何机器上，包括：

- 后端系统驻留的机器
- WebSphere Business Integration Connect 驻留的机器

另外，也可以使后端系统驻留的机器和 Business Integration Connect 驻留的机器上都有一个队列管理器。在这种情况下，使用设置通道将两个队列管理器结合起来。使用这种方法，每一方都不需要使客户机连接到网络上。

使用 WebSphere MQ V5.3 配置 JMS 传输协议机制的指示信息在第 159 页的『用 WebSphere MQ 配置 JMS 协议』中提供。

注：也可以使用 LDAP 或 WebSphere Application Server 作为 JNDI 提供程序。

配置 Business Integration Connect

本节总结了配置 Business Integration Connect，使之与后端系统一起使用的步骤。这些配置步骤假设您已经在中心社区中配置了社区参与者。特别地，本节假设下列的配置已经执行：

- “社区管理者”的**参与者**已经创建；其参与者**概要文件**也已创建。
- 文档的源（或目的地）社区**参与者**已经创建。
- 对于社区参与者，社区管理者参与者的参与者**概要文件**已经创建。
- 已经定义**目标**，因此 Business Integration Connect 的社区管理者的接收器可以通过适当的传输协议侦听从社区参与者传入的文档。
- 文档发送方社区参与者中定义并启用了**B2B 功能**，以便社区管理者能确定文档来源于该源。
- **参与者连接**存在于社区管理者和社区参与者之间，因此参与者和社区管理者可以接收（或发送）文档。

注：您应该作为社区管理员登录。

要获取关于如何配置 Business Integration Connect 以支持中心社区的完整描述，请参阅**中心配置指南**。

一旦配置了社区参与者，你必须配置 Business Integration Connect 以使它可以与后端系统通信。本节提供了下列信息以描述如何将后端系统合并到您的中心社区中：

- 『向后端系统发送文档』
- 第 27 页的『从后端系统接收文档』

向后端系统发送文档

要将文档发送到后端系统，社区管理者采用下列步骤：

1. 从某些社区参与者那里接收文档。

接收器从目标检索源文档，在社区管理者中定义了该目标，用于来自社区参与者的入站消息及其相关的传输协议。当向后端系统发送文档时，源文档是从某个社区参与者那里接收的文档；因此它被称为**参与者文档**。

2. 把参与者文档转换成目的地文档，后者使用的是后端系统所需的格式。

Business Integration Connect 文档管理器执行该转换，使参与者文档转换成目的地文档。当向后端系统发送文档时，目的地文档就是发送到后端系统的文档；因此，该文档被称为**后端文档**。

3. 将后端文档发送到后端系统。

文档管理器通过社区管理者中已经定义的网关发送后端文档，用于将消息外发到后端系统。

因此，为了使社区管理者能够向后端系统发送文档，您必须确保在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中执行了表 14 所汇总的配置。

表 14. 向后端系统发送文档的配置步骤

配置步骤	Business Integration Connect 步骤	要获取更多信息
1. 定义文档将发向何处。	1. 创建到后端系统的网关	『定义将参与者文档发向何处』
2. 定义如何处理文档。	2. 为源和目的地格式创建文档流定义。	第 25 页的『定义如何处理参与者文档』
	3. 启用要发送到后端系统的文档的文档流定义的 B2B 功能。	
	4. 创建源和目的地文档流定义之间的文档流定义交互。	
3. 定义如何连接到后端系统。	5. 创建将文档发送到后端系统的参与者连接。	第 26 页的『定义如何连接到后端系统』

定义将参与者文档发向何处

要向后端系统发送文档，社区管理者必须已经定义网关。该网关为已转换文档指定了目的地；即，它指定了社区管理者发送后端文档的目的地位置（以 URI 方式）。该位置和后端系统侦听入站消息的位置相同。网关识别进入企业应用层（后端系统中）的入口点。在 Business Integration Connect 里，检查网关由文档管理器负责。一旦文档管理器处理完文档，它就将已转换文档发送到后端系统，该系统位于网关中指定的位置上。

要在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里定义网关，请单击：

帐户管理 > 概要文件 > 网关

当您定义网关时，您指定了社区管理者和后端系统两者用来传送后端文档的传输协议。如表 15 所示，传输协议的选择取决于文档的格式。它的格式包括其封装类型和业务协议，而封装类型和业务协议是在它的文档流定义中定义的。

注：要获取关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中创建网关的更多信息，请参阅中心配置指南。

表 15. 从 Business Integration Connect 到后端系统的传输所支持的传输协议

封装类型	业务协议	HTTP 或		文件系统？
		HTTPS？	JMS？	
后端集成	RosettaNet (RNSC)	是	是	否
	XML	是	是	否
	Binary	是	是	否
无	仅 EDI	是	是	是
	仅 cXML	是	否	否
	仅 SOAP	是	否	否
	Binary	是	是	否

注： 传输协议的选择还取决于您特定的后端系统所支持的传输协议。要获取更多信息，请参阅本指南中的相关章节，以了解如何集成您的特定后端系统。

一旦您为文档选择了一个有效的传输协议，您可以提供在“网关”屏幕中定义网关所需的其它信息。

定义如何处理参与者文档

要使文档管理器能够处理参与者文档，它必须了解要将文档转换成何种格式；即，它需要知道后端文档的格式。作为后端集成的一部分，您必须确保您的 Business Integration Connect 中已经定义了下列实体：

- 文档流定义必须存在，以定义参与者文档和后端文档两者的格式。
- 社区管理者的 B2B 功能必须包括支持将后端文档的文档流定义作为目的地（目标）。
- 文档流定义交互必须存在，它使得参与者文档能作为源且后端文档能作为目的地。

定义文档流定义： 每个文档流定义都定义了 Business Integration Connect 如何处理特定的文档。它包括了文档的封装类型和业务协议。Business Integration Connect 提供了一些预定义的封装类型和协议定义。如果这些预定义格式正确定义了您的参与者文档和后端文档，那么您就不需要定义任何文档流定义。但是，如果预定义格式不能充分说明您的参与者文档或后端文档，则您必须为该文档创建有效的文档流定义。要在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里定义文档流定义，请使用 Business Integration Connect 的“管理文档流定义”屏幕。要访问该屏幕，请单击：

中心管理 > 中心配置 > 文档流定义 >
创建文档流定义

注： 要获取更多关于预定义文档流定义以及如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中创建文档流定义的信息，请参阅中心配置指南。

对于后端集成，后端文档的封装类型必须是下列之一：

- “无”封装
- “后端集成”封装

您必须根据文档的业务协议和您正在使用的特定后端系统确定应用哪种封装类型。要获取关于后端系统的封装类型信息，请参阅第 9 页的『您将使用哪种封装？』。要获取关于受支持的后端系统的信息，请参阅第 20 页的『如何访问您的后端应用程序？』。

设置用于发送的 B2B 功能： 在文档管理器可以转换源文档前，它必须确定能否处理期望的目的地文档格式。为了作出这个决定，文档管理器检查其 **B2B 功能**，后者定义了哪些文档流定义已经启用。受支持的文档流定义具有已启用各个组件文档类型（例如封装类型、业务协议和文档）。要启用特定的文档流定义，请使用 Business Integration Connect 的“B2B 功能”屏幕。要访问该屏幕，请单击：

帐户管理 > 概要文件 > B2B 功能

注： 要获取关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中设置 B2B 功能的更多信息，请参阅中心配置指南。

对于后端集成，确保后端文档的文档流定义的每个组件文档类型已经启用，以充当目的地（目标）。表 16 汇总了要在“B2B 功能”屏幕采取的操作，以创建用于将文档发送到后端系统的 B2B 功能。

表 16. 创建用于发送文档的 B2B 功能

B2B 功能列	要启用的文档流定义
设置目标	启用后端文档的文档流定义中的每个文档类型组件。

重点：如果您的社区管理者还从后端系统接收文档，您可能希望在显示“B2B 功能”的同时启用 B2B 功能。在这种情况下，您启用后端文档的文档流定义组件文档类型来充当源。第 29 页的表 22 汇总了要在“B2B 功能”屏幕上采取的操作，以创建用于从后端系统接收文档的 B2B 功能。

定义用于发送的文档流交互：为了让文档管理器了解如何转换参与者文档，它必须可以定位文档流定义交互，该交互组合了参与者文档和后端文档的文档流定义，并标识哪个是源参与者和哪个是目的地参与者。

当文档管理器准备好向后端系统发送已转换的文档时，它必须可以定位在源参与者和目的地参与者（后端系统）之间的参与者连接。但是，要使参与者连接存在，在源文档和目的地文档之间必须存在有效文档流定义交互。要在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里定义文档流定义交互，请单击：

中心管理 > 中心配置 > 文档流定义 > 管理交互 > 创建一个有效交互

注：要获取更多关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中创建文档流定义交互的更多信息，请参阅中心配置指南。

要向后端系统发送文档，定义源和目的地（目标）文档之间的交互，如表 17 中所汇总的。

表 17. 创建用于发送文档的交互

管理交互部分	操作
源	在参与者文档的文档流定义中选择组件文档类型。
目标	在后端文档的文档流定义中选择组件文档类型。

注：要获取更多关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中创建文档流定义交互的更多信息，请参阅中心配置指南。

定义如何连接到后端系统

要使文档管理器能够将已转换的文档发送到后端应用程序，它必须找到有效的参与者连接，该连接标识了源参与者和目的地参与者，并提供了这两个参与者的通信位置。要创建参与者连接，请您使用 Business Integration Connect 中的“管理连接”屏幕。要访问该屏幕，请单击：

帐户管理 > 参与者连接

对于要定义的参与者连接，源和目的地文档之间的文档流定义交互必须已经存在。在“管理连接”屏幕上，首先通过指定源和目的地（目标）参与者来检查现有的交互。表 18 列出了“管理连接”屏幕上供选择的参与者，以定义用于将文档发送到后端系统的参与者连接。

表 18. 创建用于发送文档的参与者连接

管理连接下拉列表	社区参与者的名称
源	将文档发送到社区管理者的社区参与者的名称
目标	接收来自社区参与者的文档的社区管理者名称

一旦您指定了源和目标，则单击“搜索”以检查现有的文档流定义交互。如果不存在交互，则在您继续创建参与者连接之前必须创建一个交互。如果交互确实存在（该交互的源是参与者文档流定义，而目标是后端文档流定义），那么您可以配置用于与后端系统通信的参与者连接。

注：要获取关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里创建参与者连接的更多信息，请参阅中心配置指南。

对于后端集成，该参与者连接应该指定您在第 24 页的『定义将参与者文档发向何处』中定义的网关作为其目标网关。您必须保证“网关部分”屏幕上的所有“目标网关”都使用您创建的网关名称。

从后端系统接收文档

要从后端系统接收文档，社区管理者采取下列步骤：

1. 从后端系统接收文档。

Business Integration Connect 接收器从目标检索该源文档，该目标是在社区管理者中定义的，用于来自后端系统的入站消息以及与其相关的传输协议。当从后端系统接收文档时，源文档是从后端系统接收的文档；因此，该文档被称为**后端文档**。

2. 将后端文档转换成目的地文档，后者使用的是指定的社区参与者所需的格式。

文档管理器执行该转换，将后端文档转换成目的地文档。当从后端系统接收文档时，目的地文档就是发送给某个社区参与者的文档；因此，该文档被称为**参与者文档**。

3. 将参与者文档发送给相应的社区参与者发。

文档管理器通过在社区管理者中定义的网关发送参与者文档，该网关是为了将消息外发到相应的社区参与者而定义的。

因此，要使社区管理者能够从后端系统接收文档，您必须确保在 Business Integration Connect Enterprise Edition 或 Advanced Edition 中已经执行了表 19 所汇总的配置。

表 19. 从后端系统接收文档的配置步骤

配置步骤	Business Integration Connect 步骤	要获取更多信息
1. 定义在何处检索文档。	1. 创建接收来自后端系统入站消息的目标。	第 28 页的『定义从何处检索后端文档』

表 19. 从后端系统接收文档的配置步骤 (续)

配置步骤	Business Integration Connect 步骤	要获取更多信息
2. 定义如何处理文档。	2. 为源和目的地格式创建文档流定义。 3. 启用从后端系统所接收文档的文档流定义 B2B 功能。 4. 创建源和目的地文档流定义之间的文档流定义交互。	第 29 页的『定义如何处理后端文档』
3. 定义如何连接到 Business Integration Connect。	5. 创建将文档发送到 Business Integration Connect 的参与者连接。	第 29 页的『定义如何连接到 Business Integration Connect』

定义从何处检索后端文档

要从后端系统接收文档，社区管理者必须已经定义目标。这个目标指定了文档的源；即，它标识了社区管理者侦听入站文档的位置（作为一个 URI）。该位置和后端系统发送文档的位置是相同的。目标标识进入接收器（在 Business Integration Connect 里）的入口点。在 Business Integration Connect 里，检查目标由接收器负责。一旦接收器处理完文档，它会把转换后的文档保存到持久共享存储，以便稍后能由文档管理器检索。

要在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里定义目标，请单击：

中心管理 > 中心配置 > 目标

注：要获取关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里创建目标的更多信息，请参阅中心配置指南。

当您定义目标时，您指定了社区管理者和后端系统两者用来传送后端文档的传输协议。如表 20 所示，传输协议的选择取决于文档的格式。它的格式包括其封装类型和业务协议，而封装类型和业务协议是在它的文档流定义中定义的。

表 20. 从后端系统到 Business Integration Connect 的传输所支持的传输协议

封装类型	业务协议	HTTP HTTPS?	或 JMS?	文件系统?
后端集成	RosettaNet (RNSC)	是	是	否
	XML	是	是	否
	Binary	是	是	否
无	仅 XML	是	是	是
	仅 EDI	是	是	是
	仅 cXML	是	否	否
	仅 SOAP	是	否	否
	仅 Binary	否	否	否

注：传输协议的选择还取决于您特定的后端系统所支持的传输协议。要获取更多的信息，请参阅本指南中关于集成您的特定后端系统的章节

一旦您为文档选择了一个有效的传输协议，您可以在“目标详细资料”屏幕中提供您需要为目标定义的其他信息。

定义如何处理后端文档

要使文档管理器能够处理后端文档，它必须知道需要将该文档转换成何种格式；即，它需要知道参与者文档的格式。作为后端集成的一部分，您必须确保在您的 Business Integration Connect 中已经定义了表 21 中汇总的实体。

表 21. 定义如何转换后端文档

步骤	要获取更多信息
1. 文档流定义必须存在，用于以定义参与者文档 第 25 页的『定义文档流定义』和后端文档的格式。	
2. 社区管理者的 B2B 功能必须包括对后端文档 『设置用于接收的 B2B 功能』文档流定义作为源的支持。	
3. 文档流定义交互必须存在，该交互使得参与者 『定义用于接收的交互』者文档作为目的地且后端文档能作为源。	

设置用于接收的 B2B 功能: 要获取应用于后端集成的 B2B 功能的摘要，请参阅第 25 页的『设置用于发送的 B2B 功能』。本节汇总了如何设置用于从后端系统接收文档的 B2B 功能。

对于后端集成，确保后端文档的文档流定义中每个组件文档类型都已经启用，以充当源。表 22 汇总了在 B2B 功能屏幕上采取的操作，以创建用于从后端系统接收文档的 B2B 功能。

表 22. 创建用于接收文档的 B2B 功能

B2B 功能列	要启用的文档流定义
设置源	启用后端文档的文档流定义中每个文档类型组件

定义用于接收的交互: 要获取应用于后端集成的文档流定义交互的摘要，请参阅第 26 页的『定义用于发送的文档流交互』。本节汇总了如何定义用于从后端系统接收文档的交互。

要从后端系统接收文档，定义源和目的地（目标）的文档流定义之间的交互，如表 23 中所汇总的。

表 23. 创建用于接收文档的交互

管理交互部分	操作
源	选择后端文档的文档流定义中的组件文档类型。
目标	选择参与者文档的文档流定义中的组件文档类型。

定义如何连接到 Business Integration Connect

为了让接收器能够从后端系统检索文档，它必须找到有效的**参与者连接**，该连接识别源和目的地参与者，并提供了这两个参与者通信的位置。要获取用于后端集成的参与者连接的摘要，请参阅第 26 页的『定义如何连接到后端系统』。

表 24 列出了 Business Integration Connect 的“管理连接”屏幕上选择的参与者，以定义用于从后端系统接收文档的参与者连接。

表 24. 创建用于接收文档的参与者连接

管理连接下拉列表	社区参与者的名称
源	接收来自社区参与者文档的社区管理者名称
目标	将文档发送给社区管理者的社区参与者名称

一旦您指定了源和目标，则单击“搜索”以检查现有的文档流定义交互。如果不存在交互，则在您继续创建参与者连接之前必须创建一个交互。如果确实存在交互（该交互的源是后端文档流定义，而目标是参与者文档流定义），那么您可以配置与后端系统进行通信的参与者连接。

注：要获取关于如何在 Business Integration Connect Enterprise Edition 或 Advanced Edition 里创建参与者连接的更多信息，请参阅[中心配置指南](#)。

对于后端集成，该参与者连接应该指定您在第 24 页的『定义将参与者文档发向何处』中定义的网关作为其目标网关。您必须保证“网关部分”屏幕上的所有“目标网关”都使用您创建的网关名称。

第 2 部分 与 WebSphere InterChange Server 集成

第 2 章 InterChange Server 集成简介

本章描述了如何将 WebSphere Business Integration Connect 与 WebSphere InterChange Server 集成在一起。

注:

1. 关于将 Business Integration ConnectFor 与后端系统集成的一般过程的描述, 请参阅第 3 页的第 1 章, 『规划后端集成』。
2. 本章假定您熟悉 WebSphere InterChange Server 和相关组件, 例如协作、业务对象、适配器和 WebSphere InterChange Server Access。

通常 WebSphere Business Integration Connect 和后端系统的集成由两个独自的人员或角色完成。每个角色配置特定的组件, 对于这个特定的组件该角色具有专业知识。因此, 本章将 WebSphere InterChange Server 的集成分成 Business Integration Connect 的配置和 InterChange Server 的配置中这两个部分讲述的。表 25 列出了这些配置角色以及它们在本章中的位置, 从而获得相关的配置信息。

表 25. InterChange Server 集成角色

配置角色	了解更多信息
WebSphere Business Integration Connect 的配置	1. 第 34 页的『规划与 InterChange Server 的集成』。 2. 第 37 页的『为 InterChange Server 配置 Business Integration Connect』。
WebSphere InterChange Server 的配置	1. 第 34 页的『规划与 InterChange Server 的集成』。 2. 第 40 页的『配置 InterChange Server』。

注: 尽管可以分别执行这两个角色, 当每个角色仍需要有公共的信息, 以便这两个组件可以通信。

本章讲述了以下信息:

- 第 34 页的『规划与 InterChange Server 的集成』
- 第 37 页的『为 InterChange Server 配置 Business Integration Connect』
- 第 40 页的『配置 InterChange Server』
- 第 44 页的『处理带附件的文档』

规划与 InterChange Server 的集成

要规划与 WebSphere InterChange Server 的集成，请遵循第 5 页的『规划后端集成』中概述的步骤。表 26总结了将 WebSphere Business Integration Connect 与 InterChange Server (ICS) 集成所需的集成步骤。

表 26. 规划与 WebSphere InterChange Server 的集成

集成步骤	了解更多信息
1. 确认您有支持安装 WebSphere InterChange Server 的版本并可用于 WebSphere Business Integration Connect。	第 2 章: 『Business Integration Connect 支持的 InterChange Server 版本』
2. 确定 WebSphere Business Integration Connect 文档的业务协议。	第 1 章: 第 5 页的 『使用何种业务协议?』
3. 确定文档的打包类型: “无” 或 “后端集成”。	第 1 章: 第 9 页的 『您将使用哪种封装?』
4. 确定 WebSphere Business Integration Connect 和 WebSphere InterChange Server 之间所用的传输协议。	第 2 章: 『InterChange Server 支持的消息传输』
5. 配置 WebSphere Business Integration Connect。	第 2 章: 第 37 页的 『为 InterChange Server 配置 Business Integration Connect』
6. 配置在所选传输协议上使用的 WebSphere InterChange Server 组件。	第 2 章: 第 40 页的 『配置 InterChange Server』

Business Integration Connect 支持的 InterChange Server 版本

Business Integration Connect V4.2.2 可以支持与下列 InterChange Server 版本的集成:

- 4.1.1
- 4.2.0
- 4.2.1
- 4.2.2

InterChange Server 可用于多个平台，包括 Windows 2000 和几个基于 UNIX 的平台。要了解更多信息，请参考 WebSphere InterChange Server 文档集中的 InterChange Server 安装指南。

InterChange Server 支持的消息传输

当 Business Integration Connect 通过某个特定的消息传输协议将信息发送给 InterChange Server 时，它将该信息发送给相应的与 **InterChange Server 兼容组件**，该组件理解这个特定的传输协议并将消息路由给 InterChange Server。类似的，当 InterChange Server 将信息发送给 Business Integration Connect 时，它将该消息发送给相应的与 ICS 兼容的组件，从而通过相应的传输协议路由给 Business Integration Connect。

表 27 总结用于 Business Integration Connect 集成中的与 ICS 兼容的组件。

表 27. 与 InterChange Server 兼容的组件

与 ICS 兼容的组件	描述	传输协议
WebSphere Business Integration Adapter	支持 InterChange Server 与应用程序（或技术）之间的通信。在这里，应用程序是 Business Integration Connect。	HTTP, JMS
WebSphere Business Integration 数据处理程序	处理从串行化数据到业务对象或者从业务对象到串行化数据的实际变换。使用与有效负载类型相对应的数据处理程序来执行这些转换。	HTTP, JMS
WebSphere Business Integration Connect Servlet	通过 HTTP 传输协议将非 SOAP 文档路由到 InterChange Server	HTTP（文档被发送到 InterChange Server）

Business Integration Connect 支持第 17 页的表 10 中所示的消息传输协议。在这些受支持的协议之中，InterChange Server 支持下列两种消息传输协议：

- HTTP 传输协议

注：在单独的一节中讲述了基于 HTTP 的 Web Service 交换，这是因为 Web Service 的交换方式有别于通过 HTTP 传输的其它文档。请参阅第 97 页的『通过 HTTP/S 发送 SOAP 文档』。

- JMS 传输协议

注：InterChange Server 提供了其它类型的集成选项，比如基于文件的集成。请参阅 WebSphere InterChange Server 文档以了解关于通过基于文件的集成支持文档交换的详细信息。

使用最适合您业务需求的传输协议。考虑下列条件：

- 首先，确定所使用集成机制是否适用于社区参与者和 Business Integration Connect 之间所用的传输协议。请参阅第 16 页的『您将使用哪种消息传输？』。
- 发送 SOAP 文档到 WebSphere InterChange Server 和接收来自 InterChange Server 的文档要求使用 HTTP 传输协议。要了解更多信息，请参阅第 97 页的『通过 HTTP/S 发送 SOAP 文档』。

将 HTTP 用于 InterChange Server

需要使用与 ICS 兼容的组件来通过 HTTP 发送和接收 Business Integration Connect 和 InterChange Server 之间的文档，这取决于以下条件：

- 正在发送的文档类型
- 正在集成的 InterChange Server 版本。

注：有关 HTTP 传输协议的所有参考资料也同样适用于 HTTPS。

表 28 总结了去哪里可以找到有关如何配置用于 InterChange Server 的与 ICS 兼容的组件的信息。

表 28. 配置用于 InterChange Server 的 HTTP 传输

条件	了解更多信息
如果您正在用发行版版本为 V4.2.2 之前的 InterChange Server 传输非 SOAP 文档	第 85 页的『将 HTTP 传输协议与 V4.2.2 ICS 一起使用』

表 28. 配置用于 InterChange Server 的 HTTP 传输 (续)

条件	了解更多信息
如果您正在用 InterChange Server V4.2.2 传输非 SOAP 文档	第 61 页的『将 HTTP 传输协议用于 4.2.2 之前版本的 ICS』
如果您正在发送 SOAP 文档	第 97 页的『通过 HTTP/S 发送 SOAP 文档』

当您确定 InterChange Server 采用哪种传输方法时，请考虑使用 HTTP 所具备的好处：

- 如果您需要同步事务，则必须使用 HTTP 传输协议。

将 JMS 用于 InterChange Server

第 101 页的表 71 总结了通过 JMS 发送和接收 Business Integration Connect 和 InterChange Server 之间的文档所需的与 ICS 兼容的组件。基本上，JMS 支持涉及到 WebSphere Business Integration Adapter for JMS 的使用。Adapter for JMS 以异步方式调用 InterChange Server 中的协作。

当您确定 InterChange Server 采用哪种消息传输时，请考虑使用 JMS 所具备的好处：

- Adapter for JMS 可以提供从 Business Integration Connect 到 WebSphere InterChange Server 的“有保证的事件传递”。

“有保证的事件传递”确保事件永远不会丢失或发送两次。

有关如何配置 JMS 的更多信息，请参阅第 33 页的第 2 章，『InterChange Server 集成简介』。

InterChange Server 集成支持

Business Integration Connect 提供了一些样本，它们可帮助您处理与 InterChange Server 的集成。这些样本位于 Business Integration Connect 产品目录的子目录中，如下所示：

Integration/WBI/WICS/samples

表 29 列出了 samples 目录的子目录，每个子目录针对 InterChange Server 支持的各种传输协议。

表 29. InterChange Server 集成样本

传输协议	InterChange Server 版本	Samples 的子目录
HTTP	4.1.1, 4.2.0, 4.2.1	WBICServlet
HTTP	4.2.2	常规样本: HTTP
JMS	支持所有版本	特定于 RosettaNet 的样本: RosettaNet/HTTP 常规样本: JMS
		特定于 RosettaNet 的样本: RosettaNet/JMS

为 InterChange Server 配置 Business Integration Connect

第 23 页的『配置 Business Integration Connect』中一般性地概述了如何配置 Business Integration Connect 与后端系统的通信。本节总结了配置 Business Integration Connect 以便与 InterChange Server 进行通信所需的步骤。要执行该配置，请使用作为中心社区“社区管理者”的 Business Integration Connect Enterprise Edition 或 Advanced Edition 的实例。

配置 Business Integration Connect 涉及下列步骤：

- 配置出站文档支持

有关将文档从 Business Integration Connect 发送到 InterChange Server 的信息，请参阅『提供对出站文档的支持』。

- 配置入站文档

有关将文档从 InterChange Server 发送到 Business Integration Connect 的信息，请参阅第 38 页的『提供对入站文档的支持』。

提供对出站文档的支持

要使 Business Integration Connect 将文档发送到任何后端系统，您必须执行第 24 页的『定义将参与者文档发向何处』中所描述的步骤。当后端系统是 InterChange Server 时，您需要创建网关，该网关的传输类型与 Business Integration Connect 和 InterChange Server 之间传递消息所用的传输协议相匹配。当“社区管理者”将文档发送到 InterChange Server 时，它必须清楚在何处路由文档。这个位置必须符合使用的传输协议。这个传输协议必须是 InterChange Server 所支持的传输协议（请参阅第 34 页的『InterChange Server 支持的消息传输』）。

下面几节总结了如何创建下列 InterChange Server 所支持传输协议的网关：

- 『配置通过 HTTP 传输协议传递的出站文档』
- 第 38 页的『配置通过 JMS 传输协议传递的出站文档』

配置通过 HTTP 传输协议传递的出站文档

当“社区管理者”通过 HTTP 协议将文档发送到 InterChange Server 时，“社区管理者”将消息路由给已定义的网关。这个网关标识了 URL，InterChange Server 可以通过这个 URL 接收文档。当 InterChange Server 使用 HTTP 协议时，ICS 兼容组件在相应的 URL 处接收文档，这里该组件可以将文档发送给 InterChange Server。

要使“社区管理者”可以通过 HTTP 传输协议经由网关发送文档，则必须从“社区控制台”的“网关详细信息”屏幕创建网关。该网关必须配置为可使用 HTTP 1.1 传输协议，并向与 ICS 兼容的相应组件所侦听的 URL 写信息。如同表 30 所示，可以在这个网关定义的“目标 URL”字段中提供这个 URL。

注：有关如何创建网关的概述，请参阅第 24 页的『定义将参与者文档发向何处』。

表 30. 用于与 InterChange Server 进行通信的“网关详细信息”屏幕的 HTTP 值

InterChange		
Server 版本	“目标 URL”文件的值	注意事项和限制
4.1.1, 4.2.0, 4.2.1	该 URL 应当与为 WebSphere Business Integration Connect Servlet 配置的那个 URL 相同。	从 WebSphere InterChange Server 集成中 Connect Servlet 的配置获取这个 URL。
4.2.2	这个 URL 应当与 WebSphere Business Integration Adapter for HTTP 的协议侦听器所使用的用于接收请求的 URL 相同。	从 WebSphere InterChange Server 集成中 Adapter for HTTP 配置获取这个 URL。

配置通过 JMS 传输协议传递的出站文档

当“社区管理者”通过 JMS 协议将文档发送给 InterChange Server 时，“社区管理者”将文档路由给相应的 JMS 队列，InterChange Server 可以在此接收文档。要使“社区管理者”获取这个 JMS 位置，您必须在 Business Integration Connect 中创建一个网关，这个网关使用 JMS 传输协议。必须将这个网关配置成可向 Adapter for JMS 轮询的队列些信息。

注：有关如何创建网关的概述，请参阅第 24 页的『定义将参与者文档发向何处』。

要使“社区管理者”可以通过 JMS 传输协议经由网关发送文档，则必须从“社区控制台”的“网关详细信息”屏幕创建网关。如果您正在使用 WebSphere MQ V5.3 作为 JMS 提供程序，则请使用第 165 页的表 116 中的信息来设置网关字段。另外，在“网关详细信息”屏幕中为 JMS 协议指定表 31 中所确定的信息。

表 31. 用于与 ICS 通信的“网关详细信息”屏幕的 JMS 值

“网关详细信息”字段	值	注意事项和限制
JMS 消息类	TextMessage、BytesMessage 或 StreamMessage	2.4.1 之前的 Adapter for JMS 版只支持 JMS 文本消息。如果您还在使用 V2.4.1 之前适配器，请确信将网关配成仅写入 JMS 文本消息 (TextMessage)
JMS 队列名	JMS 队列的名称与 Adapter for JMS 的输入队列名相同	这个队列必须包含在 Adapter for JMS 的输入队列列表之中，也就是说，这个适配器轮询这个队列以查找入站事件。要了解更多信息，请参阅第 108 页的『标识 JMS 队列』。

提供对入站文档的支持

要使 Business Integration Connect 接收来自任何后端系统的信息，您必须执行第 28 页的『定义从何处检索后端文档』中所描述的步骤。当您的后端系统是 InterChange Server 时，需要在您的“社区管理者”中采取下列步骤：

1. 作为“社区管理者”的参与者概要文件的一部分，定义网关类型并提供接收器将在其上侦听的相关 IP 地址。
2. 创建一个目标，该目标的传输类型与在 Business Integration Connect 和 InterChange Server 之间传递文档所用的传输协议相匹配。

要使“社区管理者”接收来自 InterChange Server 的文档，必须清楚在何处检索消息。这个位置必须符合将要使用的传输协议。这个传输协议必须是 InterChange Server 所支持的传输协议（请参阅第 34 页的『InterChange Server 支持的消息传输』）。

下面几节总结了如何创建 InterChange Server 所支持传输协议的目标。

配置通过 HTTP 传输协议传递的进站文档

当“社区管理者”通过 HTTP 传输协议接收文档时，其“接收器”从已定义的目标中检索文档。这个目标标识了一个 URL，“接收器”在这个 URL 上侦听来自 InterChange Server 的文档。当 InterChange Server 使用 HTTP 传输协议时，适配器将文档发送到相应的 URL，“社区管理者”可以在此接收文档。

要使“社区管理者”可以通过 HTTP 传输协议经由目标接收文档，则必须从“社区控制台”的“目标列表”屏幕创建目标。这个目标必须使用 HTTP 1.1 传输协议。“社区管理者”确定这个由以下信息组成的 URL：

- 主机的 IP 地址，从“社区管理者”的参与者概要文件中获取
- 目标 URL，从目标定义的 URL 字段中获取

注：第 28 页的『定义从何处检索后端文档』中提供了关于如何创建目标的概述。

要使 InterChange Server 能够将文档发送到该目标，它的 ICS 兼容组件必须配置成可将文档发送到该 URL。因此，您必须确保这个目标 URL 可用于 InterChange Server 配置。

配置通过 JMS 传输协议传递的进站文档

当“社区管理者”通过 JMS 协议接收来自 InterChange Server 的文档时，“社区管理者”从相应的 JMS 队列那里获取文档，InterChange Server 通过这个队列发送文档。要使“社区管理者”能够获取这个 JMS 位置，您必须在 Business Integration Connect 中创建一个目标，这个目标使用 JMS 传输协议。通过这个目标，“社区管理者”在它的人站队列上侦听所有文档并检索它们。

注：有关如何创建目标的概述，请参阅第 28 页的『定义从何处检索后端文档』。

要使“社区管理者”可以通过 JMS 传输协议经由目标接收文档，则必须从“社区控制台”的“目标列表”屏幕创建目标。如果您正在使用 WebSphere MQ V5.3 作为 JMS 提供程序，则请使用第 164 页的表 115 中的信息来设置目标字段。另外，在“目标详细信息”屏幕中为 JMS 协议指定了表 32 中所确定的信息。

表 32. 用于与 ICS 的通信的“目标详细信息”屏幕的 JMS 值

“目标详细信息”字段	值	注意事项和限制
JMS 消息类	TextMessage	2.4.1 之前的 Adapter for JMS 只支持 JMS 文本消息。如果您正在使用这当中的某个版本适配器，请确保将目标配置成仅写入 JMS 文本消息。

表 32. 用于与 ICS 的通信的“目标详细信息”屏幕的 JMS 值 (续)

“目标详细信息” 字段	值	注意事项和限制
JMS 队列名	JMS 队列的名称与 Adapter for JMS 的输出队列名相同	这个队列必须存在于 Adapter for JMS 的输出队列列表中；也就是说，这个适配器必须将文档发送到该队列。要了解更多信息，请参阅第 108 页的『标识 JMS 队列』。

配置 InterChange Server

对于 Business Integration Connect 和 InterChange Server 之间的交互，必须在“系统管理器”工具中创建一个“集成组件库 (ICL)”。该 ICL 将包含以下构件：

- 业务对象定义
- 连接器对象
- 协作模板和协作对象

还必须创建一个用户产品并且从 ICL 中选择在 InterChange Server 和 Business Integration Connect 之间进行特定交互所需的那些构件。

注：有关怎样创建 ICL 和配置 InterChange Server 的更多详细信息，请参阅 WebSphere InterChange Server 文档集中的系统实现指南。

创建业务对象定义

Business Integration Connect 将消息发送到与 ICS 兼容的组件，该组件将消息以一个或多个**业务对象**的形式路由到 InterChange Server。要使 InterChange Server 识别业务对象，首先必须找到模板，该模板名为**业务对象定义**，它来描述了这个业务对象中的信息结构。业务对象定义中的每条信息都保存在**属性**中。因此，必须创建业务对象定义以表示消息中的信息。要创建业务对象定义，请使用“业务对象设计器”工具来创建业务对象定义。

注：业务对象设计器包含在 WebSphere InterChange Server 和 WebSphere Business Integration Adapter 产品中。有关如何使用该工具的更多信息，请参阅**业务对象开发指南**。

InterChange Server 使用业务对象查找一下信息：

- 『用于文档的业务对象』
- 第 42 页的『用于配置信息的业务对象』

用于文档的业务对象

要保存 Business Integration Connect 文档或消息的有效负载，必须定义一个业务对象定义来表示**有效负载业务对象**。ICS 兼容组件以有效负载业务对象形式将文档传输到（或传输出）InterChange Server。本节提供了下列关于有效负载业务对象的信息：

- 第 41 页的『业务对象结构』
- 第 41 页的『业务对象变换』
- 第 42 页的『InterChange Server 术语』

业务对象结构: 必须设计有效负载业务对象，从而使要传输的文档中的每条信息在其相关联的有效负载业务对象定义中必须有一个属性。正如表 33 所示，有效负载业务对象内容取决于文档结构和文档所用的打包类型。

表 33. 打包与有效负载业务对象结构的关系

文档结构	打包类型	有效负载业务对象定义
仅限有效负载 仅限有效负载	无 后端集成	保存文档的有效负载信息。 包含： <ul style="list-style-type: none"> • 文档的有效负载信息 • 传输级的标题头
有效负载和附件	无	不适用。如果文档包含附件则必须使用“后端集成”打包类型。
有效负载和附件 文档包含 XML 名为传输包络的包装器，其中包装了有效负载和附件。	后端集成	包含： <ul style="list-style-type: none"> • 文档的有效负载信息 • 传输级的标题头 • 附件容器，它包含附件数据和任何附件业务对象 <p>Business Intergration Connect-supplied 提供了名为“附件数据处理程序”的数据处理程序，可用它来处理传输包络。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。</p>

还必须根据用于集成 Business Integration Connect 的特定的 ICS 兼容组件的需求来设计有效负载业务对象（请参阅第 35 页的表 27）。表 34 提供了关于去哪里查找发现如何创建有效负载业务对象的详细信息，以通过特定的传输协议进行传输。

表 34. 为不同的传输协议创建有效负载业务对象

传输协议	注意事项和限制	了解更多信息
HTTP	用于 InterChange Server V4.2.2 之前的版本	第 79 页的『为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象』
HTTP	用于 InterChange Server V4.2.2	第 91 页的『为通过 HTTP 收发文档的 V4.2.2 ICS 创建业务对象定义』
JMS	如果文档使用“后端集成”打包类型	第 109 页的『创建 JMS 的业务对象定义』
全部	如果文档有附件	第 54 页的『创建与附件关联的业务对象定义』

业务对象变换: 通常，ICS 兼容组件使用数据处理程序使文档格式和文档的业务对象表示之间进行相互转换。这个数据处理程序称为**有效负载数据处理程序**。必须配置 ICS 兼容组件来调用与有效负载内容类型相对应的数据处理程序。通常，由于用于 XML 的 WebSphere Business Integration 使 XML 消息和业务对象间进行相互转换，因此将它配置成有效负载数据处理程序。然而，您可以为任何消息格式创建定制的数据处理程序，WebSphere Business Integration Server 没有为它们提供相应数据处理程序。

注: 对于 XML 消息的处理，请确保您正在使用 WebSphere Business Integration Data Handler for XML V2.3.1 或更高版本。对于 cXML 消息，必须使用 Data Handler for XML V2.4.1 或更高版本。

必须确保您所使用的有效负载数据处理程序可以忽略所用传输协议所需的子元对象。在使用数据处理程序（不管它是由 WebSphere Business Integration 提供还是一个定制的数据处理程序）之前，请确保它提供了对子元对象的支持。请参阅关于业务对象的特定于应用程序信息中的 `cw_mo_label` 标记，在针对传输协议相应的章节中讲述了该信息（请参阅第 41 页的表 34）。

要指出使用哪种数据处理程序来转换有效负载，必须采取下列步骤：

- 标识数据处理程序为转换有效负载而必须支持的 MIME 类型并查找可以处理该 MIME 类型的数据处理程序。

在 WebSphere Business Integration Adapter 文档集的 *数据处理程序指南* 中描述了 IBM 提供的数据处理程序。如果这些数据处理程序都不适用，那么可以创建一个定制的数据处理程序。

- 在“业务对象设计器”里，为您需要使用的数据处理程序创建一个子元对象。如果您正在使用 IBM 提供的数据处理程序，请参阅 *数据处理程序指南* 以获取关于子元对象结构的信息。
- 在“业务对象设计器”里，更新连接器的顶级数据处理程序元对象以包含受支持的 MIME 类型的属性。该属性的属性类型是数据处理程序的子元对象。
- 在“连接器适配器”里，设置相应的连接器配置属性以标识要使用的数据处理程序：
 - 分别将 `DataHandlerConfigMO` 和 `DataHandlerMimeType` 属性设置成带有顶级数据处理程序元对象的名称和受支持的 MIME 类型。
 - 将 `DataHandlerClassName` 属性设置成带有数据处理程序类的名称，并进行实例化。

注：可以设置 `DataHandlerConfigMO` 与 `DataHandlerMimeType` 属性，也可以设置 `DataHandlerClassName` 属性。

- 在“连接器配置器”里，在受支持的业务对象列表中包含顶级数据处理程序元对象。

InterChange Server 术语： 对于 InterChange Server，有效负载业务对象的名称取决于通信的方向，如下所示：

- 当 Business Integration Connect 将文档发送到 InterChange Server 时，它正在参与 InterChange Server 的 **事件通知**。

在这种情况下，该数据业务对象叫做 **事件业务对象**（有时称为事件），它通知 InterChange Server 某个社区参与者中发生了事件。

- 当 Business Integration Connect 接收来自 InterChange Server 的文档时，它正在参与 InterChange Server 的 **请求处理**。

在这种情况下，该数据业务对象是 **请求业务对象**，InterChange Server 将它发送到来自某个社区参与者的请求信息。作为响应，InterChange Server 可能会将 **响应业务对象** 发送回中心社区。

用于配置信息的业务对象

对于许多 ICS 兼容组件，您创建业务对象定义来保存配置信息。这样的业务对象通常被称为 **元对象**。

至于如何创建数据业务对象，以通过某个特定的传输协议进行传输，表 34 提供了有关在何处查找此类详细说明书的信息。

表 35.

传输协议	相关组件	了解更多信息
HTTP (针对 4.2.2 之前的 InterChange Server)	包装器数据处理程序	第 70 页的『创建用于包装器数据处理程序的配置业务对象』
HTTP (针对 InterChange Server V4.2.2)	Adapter for HTTP	第 82 页的『为 4.2.2 之前版本的 InterChange Server 创建 HTTP 传输级头信息』.
JMS	Adapter for JMS	第 110 页的『创建 JMS 头信息』
全部	附件数据处理程序	第 50 页的『创建附件子元对象』

创建连接器

如果针对传输协议的 ICS 兼容组件是 WebSphere Business Integration Adapter, 您必须为那个适配器创建**连接器对象**。这个连接器对象表示适配器在运行时的一个实例。您可以用 InterChange Server 的“系统管理”工具来创建连接器对象。

注: 有关如何创建连接器对象的信息, 请参阅 WebSphere InterChange Server 文档集中的**系统工具指南**。

表 36 概括了何处可发现有关如何创建连接器对象的信息, 基于您正在使用的传输协议。

表 36. 创建用于不同传输协议的连接器对象

传输协议	ICS 兼容组件	了解更多信息
HTTP (针对 4.2.2 之前的 InterChange Server)	Adapter for XML (仅用于请求处理)	第 85 页的『创建 XML 连接器对象』
HTTP (针对 InterChange Server V4.2.2)	Adapter for HTTP	第 97 页的『创建 HTTP 连接器对象』
JMS	Adapter for JMS	第 113 页的『创建 JMS 连接器对象』

=创建协作

使用 InterChange Server 中的**协作**执行您所需的实际业务流程。因此, InterChange Server 中必须有相应的能正确处理 Business Integration Connect 文档的协作。请务必执行下列步骤, 以便在运行时能使用相应的协作:

1. 确保有能提供您所需业务流程的协作模板:
 - 如果当前不存在这样的协作模板, 则必须创建一个并对其进行编译。
 - 如果存在协作模板, 则必须知道如何有效地使用它, 以便能够配置协作对象。
2. 创建协作对象并绑定其端口, 如下所示:

- 对于请求处理: 应当将发送请求至 Business Integration Connect 的“发送”端口设置成为 ICS 兼容组件。
- 对于事件通知: 应当将接收来自 Business Integration Connect 的事件的“接收”端口设置成 ICS 兼容组件。

表 37 总结了根据正在使用的传输协议去何处查找关于如何创建连接器对象的信息。

表 37. 针对不同传输协议的协作绑定

传输协议	ICS 兼容组件	了解更多信息
HTTP (针对 4.2.2 之前的 InterChange Server)	Adapter for XML (仅用于请求处理)	第 85 页的『绑定协作以与 Adapter for XML 通信』
HTTP (针对 InterChange Server V4.2.2)	Adapter for HTTP	第 97 页的『绑定协作以与 Adapter for HTTP 进行通信』
JMS	Adapter for JMS	第 113 页的『绑定协作以与 Adapter for JMS 通信』

部署项目

一旦您的用户项目包含了定义运行时组件所需的构件, 就必须将它部署到 InterChange Server 资源库。您要用“系统管理器”部署一个用户项目。

处理带附件的文档

Business Integration Connect 提供了“附件”数据处理程序来处理 Business Integration Connect 到 InterChange Server 和 InterChange Server 之间发送的文档。“附件”数据处理程序使用 XML 传输封包(带或不带附件)在支持的串行化格式和其业务对象表现之间进行转换。当出现下列情形之一时, 应当将“附件”数据处理程序配置成有效负载数据处理程序:

- Envelope Flag for Backend Integration 打包已经设置为“是”。

当该标志设置为“是”, Business Integration Connect 总是将文档包装在 XML 传输包络中, 不管它是否包含附件。将 Backend Integration 打包的标志设置为“是”是概要文件的“B2B 功能”屏幕的一部分。要了解更多信息, 请参阅第 14 页的『有效负载』。

- 要处理的文档可包含附件。

当文档包含附件时, Business Integration Connect 将其包装在 XML 传输包络中。在任何文档流中, 都有一个有效负载, 并可能有多个附件。如果发送或接收包含附件的文档, 有效负载业务对象需要包含附件信息。

注: 包含附件的 SOAP 文档不需要附件数据处理程序。有关如何处理 SOAP 文档的信息, 请参阅第 97 页的『通过 HTTP/S 发送 SOAP 文档』。

下列任何一个环境都可调用“附件”数据处理程序:

- 从 WebSphere Business Integration Adapter

- 如果 Business Integration Connect 和 InterChange Server V4.2.2 使用 HTTP 传输协议，则 Adapter for HTTP 调用附件数据处理程序。
 - 如果 Business Integration Connect 和 V4.2.2 之前的 InterChange Server 参与请求处理（InterChange Server 发起请求），则 Adapter for XML 调用附件数据处理程序以将业务对象转换成 XML 传输包络中包装的文档串行化“形式”。
 - 如果 Business Integration Connect 和 InterChange Server（任何受支持版本的）使用 JMS 传输协议，则 Adapter for JMS 调用附件数据处理程序。
- 从 WebSphere Server Access，它驻留在 InterChange Server 中，以处理访问客户机的请求

如果 Business Integration Connect 和 V4.2.2 之前 InterChange Server 参与事件通知（Business Integration Connect 发起该请求），则 WebSphere Business Integration Connect Servlet（一个访问客户机）将文档发送到“服务器访问”，它调用“包装器”数据处理程序将文档转换为其业务对象表示。如果在 XML 传输包络中包装这个文档，则“包装器”数据处理程序调用“附件”数据处理程序以将文档的串行化格式（在它的 XML 传输包络中）转换成相应的业务对象。

不管调用实体是适配器还是“服务器访问”，当调用实体接收 XML 传输包络中的文档时，调用实体调用“附件”数据处理程序来将这个文档转换为与其相对应的业务对象表示。例如，第 103 页的图 20 所示 Adapter for JMS 调用“附件”数据处理程序来将文档的串行化格式转换为其业务对象表示。相反，当调用实体接收 XML 传输包络中文档的业务对象表示时，调用实体调用“附件”数据处理程序将这个业务对象结构转换为与其相对应的文档格式。例如，第 105 页的图 21 所示 Adapter for JMS 调用“附件”数据处理程序将文档的业务对象表示转换为文档的串行化格式。

本节提供了下列关于附件数据处理程序的信息：

- 『执行转换』
- 第 49 页的『设置附件数据处理程序的环境』
- 第 50 页的『配置附件数据处理程序』
- 第 54 页的『创建与附件关联的业务对象定义』

执行转换

“附件”数据处理程序可以解释这个 XML 传输包络的结构并处理包含的数据和相应的业务对象表示间的转换，如下所示：

- 『将文档转换为业务对象』以将文档发送到 InterChange Server
- 第 47 页的『将业务对象转换为文档』以接收来自 InterChange Server 的文档

将文档转换为业务对象

在 Business Integration Connect 将文档发送到 InterChange Server 之前，它必须确定是否将内容包装在 XML 传输包络中。如果 Business Integration Connect 创建传输包络，则有效负载和任何附件都是 Base64 编码。Business Integration Connect 然后根据相应的传输级头将 XML 传输包络发送到相应的 ICS 兼容组件。可以配置这个 ICS 兼容组件（WebSphere Business Integration 适配器或“包装器”数据处理程序）来调用“附件”数据处理程序，以处理将有效负载和任何 XML 打包文档中的附件转换成相应业务对象表示。

为将 XML 传输包络中包装的文档转换为其业务对象表示，调用实体实例化“附件”数据处理程序，传递给文档（在它的传输包络中）。然后，“附件”数据处理程序采取下列步骤：

1. 装载在数据处理程序的子元对象中定义的 content-type 映射。

Content-type 映射是在子元对象的 ContentTypeMap_x 配置特性中定义的。子元对象是包含了“附件”数据处理程序配置信息的业务对象。此业务对象中的属性将内容类型与 content-type 映射相关联。要了解更多信息，请参阅第 50 页的『创建附件子元对象』。

2. 检查文档，查看其是否包装在 XML 传输包络中。

- 如果“附件”数据处理程序未检测到传输包络，则不必从这个“包络”结构中抽取有效负载。

文档仅包含有效负载，“附件”数据处理程序必须将该有效负载转换为与其相对应的业务对象表示。要了解更多信息，请参阅『处理不在传输包络中的文档』。

- 如果“附件”数据处理程序确实检测到传输包络，则必须从这个封包结构中抽取有效负载和所有附件。

文档包含一个有效负载并可能有一些附件。因此，“附件”数据处理程序需要将有效负载和“所有”附件转换为与它们相关联的业务对象表示。要了解更多信息，请参阅『处理传输封包中的文档』。

3. 设置产生的有效负载业务对象并将该业务对象返回给调用实体。

处理不在传输包络中的文档： 如果“附件”数据处理程序确定文档未包含在 XML 传输“包络”中，则不必从这个封包结构中抽取有效负载数据。因此，数据处理程序使用 PayloadDataHandlerMimeType 配置特性（在它的子元对象中定义）来获取 MIME 类型，该类型确认出缺省的有效负载数据处理程序来实例化文档有效负载。这个数据处理程序将有效负载数据转换为与其对应的有效负载业务对象并将产生的有效负载业务对象返回给其调用实体。

处理传输封包中的文档： 如果“附件”数据处理程序确定文档包含在 XML 传输“包络”中，则必须在处理程序处理文档之前从这个包络结构中抽取出有效负载和所有附件。因此，数据处理程序采取以下步骤来处理 and 转换文档：

1. 从传输包络抽取有效负载和所有附件并对有效负载数据进行解码。

有效负载包含在 <payload> XML 标记中。每个附件包含在 <attachment> XML 标记中。

2. 搜索与有效负载相匹配的内容类型的 content-type 映射。

使用匹配的 content-type 映射中的指定的 MIME 类型创建数据处理程序的实例。这个数据处理程序将有效负载数据转换为与其对应的有效负载业务对象并将产生的业务对象返回给“附件”数据处理程序。

3. 为有效负载创建内容信息业务对象。

检查有效负载业务对象定义的业务对象级特定于应用程序的信息，并确定内容信息业务对象的名称，该对象的属性名由 cw_mo_bcg_content_info 标记所指定。创建该内容信息业务对象的实例并设置有效负载的内容类型和编码的值。

4. 为有效负载创建附件容器业务对象。

检查有效负载业务对象的业务对象级的特定于应用程序的信息，并确定附件容器业务对象的名称，该对象的属性名由 `cw_mo_bcg_attachment` 标记指定。创建附件容器业务对象的实例，并将其保存在有效负载业务对象相应的属性中。

如果 `cw_mo_bcg_attachment` 标记不存在（或为空），则认为文档不包含任何附件。因此，不需要采取进一步的处理步骤。“附件”数据处理程序返回经过转换的有效负载业务对象。

5. 为附件容器创建缺省的附件业务对象。

检查附件容器业务对象的业务对象级特定于应用程序的信息，并确定缺省附件业务对象的名称，该对象的属性名由 `cw_mo_bcg_default_attribute` 标记所指定。创建缺省附件业务对象的实例，并将其保存在附件容器业务对象相应的属性中。

6. 通过搜索与附件相匹配的内容类型的 `content-type` 映射来确定是否需要将附件转换为业务对象。

获取来自附件的内容类型和字符集编码，并检查 `content-type` 映射中是否有相应的项。

- 如果未找到相应的 `content-type` 映射，“附件”数据处理程序不会创建附件数据的业务对象。

因此，数据处理程序创建缺省附件业务对象的实例，设置其内容信息业务对象中的内容类型和编码的值，并在 `attachment` 属性中设置 `base64` 编码附件数据（作为字符串）。

然后，附件数据处理程序使用缺省的附件业务对象填充附件容器业务对象。

- 如果找到 `content-type` 映射，则“附件”数据处理程序检查是否需要将“附件”转换为业务对象：
 - 如果在相匹配的 `content-type` 映射中 `ConvertAttachment` 配置特性为 `false`，则“附件”数据处理程序创建缺省的附件业务对象的实例，并设置其的内容信息业务对象的内容类型和编码的值，设置 `attachment` 属性中的 `base64` 编码的附件数据（作为字符串）。

然后，附件数据处理程序使用缺省附件业务对象填充附件容器业务对象。

- 如果在相匹配的 `content-type` 映射中的 `ConvertAttachment` 配置特性为 `true`，则附件数据处理程序对附件数据解码并创建数据处理程序的实例来处理附件数据。这个数据处理程序处理经过解码的字节并返回相应的附件业务对象。

然后，附件数据处理程序检查附件业务对象定义的业务对象级特定应用程序的信息，并确定内容信息业务对象的名称，该对象的属性名是由 `cw_mo_bcg_content_info` 标记指定的。如果该标记存在，则数据处理程序创建附件的内容信息业务对象并设置附件内容类型和编码的值。

最后，“附件”数据处理程序使用附件业务对象填充附件容器业务对象。

将业务对象转换为文档

在 Business Integration Connect 按来自 InterChange Server 中的文档之前，ICS 兼容组件必须确定是否在 XML 传输包络中包装有效负载和所有附件的业务对象表示。InterChange Server 将业务对象发送到相应的 ICS 兼容组件，用该组件处理实际的转换。可以配置这个 ICS 兼容组件（WebSphere Business Integration 适配器或“包装器”数

据处理程序)来调用“附件”数据处理程序以处理有效负载和所有附件业务对象到相应的有效负载和附件的转换, 这跟 XML 传输封包的创建一样。

为将带附件的有效负载业务对象转换为它的传输“包络”表示, 调用实体实例化附件数据处理程序, 将它传递给有效负载业务对象。附件数据处理程序采取以下步骤:

1. 装载在其配置子元对象中定义过的 content-type 映射。

Content-type 映射是在子元对象的 ContentTypeMap_x 配置特性中定义的。子元对象是包含“附件”数据处理程序配置信息的业务对象。此业务对象中的属性将内容类型与 content-type 映射相关联。要了解更多信息, 请参阅第 50 页的『创建附件子元对象』。

2. 检查业务对象以确定是否创建 XML 传输包络。

- 如果“附件”数据处理程序确定文档不需要传输封包, 则不必包装这个“包络”结构里的有效负载。

文档仅包含有效负载, “附件”数据处理程序必须根据与其相关联的业务对象表示创建有效负载。数据处理程序不必为文档创建传输包络。要了解更多信息, 请参阅『不使用传输封包创建文档』。

- 如果附件数据处理程序确定文档需要传输“包络”, 则必须包装该封包结构中的有效负载和所有附件。

文档包含一个有效负载并可能有一些附件。因此, 附件数据管理程序需要将有效负载业务对象表示转换为有效负载和任一附件并将其包装在传输“包络”中。要了解更多信息, 请参阅『使用传输封包创建文档』。

3. 设置 Business Integration Connect 产生的有效负载和任何附件, 并将该文档返回给调用实体。

不使用传输封包创建文档: 如果“附件”数据处理程序确定有效负载业务对象不需要 XML 传输包络, 则不必包装该包络结构中的有效负载数据。因此, 数据处理程序使用缺省的有效负载数据处理程序来将有效负载业务对象转换为与它对应的有效负载文档。PayloadDataHandlerMimeType 配置特性(在“附件”数据处理程序的子元对象中定义的)包含 MIME 类型, 该类型确定了缺省有效负载数据处理程序以实例化有效负载业务对象。这个数据处理程序接收有效负载业务对象; 将其作为参数并将产生的有效负载文档返回给它的调用实体。

使用传输封包创建文档: 如果“附件”数据处理程序确定有效负载业务对象确定需要 XML 传输包络, 则必须该包络结构中包装有效负载和附件文档。因此, 数据处理程序采取以下步骤来处理 and 转换业务对象:

1. 获取的有效负载内容类型和字符集编码。

在有效负载业务对象的业务对象级特定于应用程序的信息中的 cw_mo_bcg_content_info 标记指定了内容信息的属性名。该属性包含关于有效负载的内容类型和编码。

注: 如果内容信息属性不存在, 使用缺省的数据处理程序(由 PayloadDataHandlerMimeType 配置特性中包含的 MIME 类型确定, 在附件数据处理程序的子元对象中)来转换有效负载业务对象。

2. 搜索与有效负载相匹配的内容类型的 content-type 映射。

使用相匹配的 `content-type` 映射中指定的 MIME 类型创建有效负载数据处理程序实例。这个数据处理程序将有效负载业务对象转换为与其对应的有效负载文档，并将产生的文档返回给附件数据处理程序。根据有效负载数据处理程序返回的字符串，“附件”数据处理程序使用 Base64 来对字节进行编码并将结果存储到 XML 传输包络的 `payload` 标记中。

3. 从有效负载业务对象中获取附件容器。

附件容器驻留在有效业务对象的附件容器的属性中。有效负载业务对象的业务对象级特定于应用程序的信息包含了 `cw_mo_bcg_attachment` 标记，该标记确定附件容器的属性。该属性包含了附件。

如果 `cw_mo_bcg_attachment` 标记不存在（或为空），则认为该文档不包含任何附件。因此，不需要采取进一步的处理步骤。“附件”数据处理程序将经过转换的有效负载放在传输包络中返回。

4. 对每个附件，确定将附件表示为业务对象还是仅表示为数据。

- 如果附件仅是附件数据，则附件容器业务对象的业务对象级特定于应用程序的信息包含 `cw_mo_bcg_default_attribute` 标记，该标记确定缺省附件属性。这个属性包含附件数据，附件数据处理程序检索该附件数据，并抽取出 Base64 编码的数据，然后将结果存储在文档中。
- 如果用业务对象表示附件，则它的属性级特定于应用程序信息包含 `wbic_type` 标记以表明它包含附件业务对象。

附件数据处理程序采取以下步骤来处理附件业务对象：

a. 检索附件属性的内容并获取附件的内容类型和编码。

附件业务对象的业务对象级特定于应用程序的信息包含 `cw_mo_bcg_content_info` 标记，该标记确定内容信息属性。该属性包含附件的内容类型和编码。“附件”数据处理程序将该内容信息存储到文档的附件标记。

b. 搜索与附件相匹配的内容类型的 `content-type` 映射。

使用相匹配的 `content-type` 映射中的 MIME 类型创建数据处理程序的实例。这个数据处理程序将附件业务对象转换为与其对应的附件文档并将产生的文档（作为字符串）返回给“附件”数据处理程序。

c. 将经过编码的结果存储到文档的 XML 包装器的附件标记中。

“附件”数据处理程序从返回的字符串（使用字符集，如果出现）中获得字节，并使用 Base64 对字节进行编码。然后将结果存储到附件标记中。

设置附件数据处理程序的环境

关于 Business Integration Connect——提供的“附件”数据处理程序的使用包含以下步骤：

- 第 50 页的『部署附件数据处理程序』
- 第 50 页的『配置附件数据处理程序』

部署附件数据处理程序

可在 Business Integration Connect 安装介质中找到附件数据处理程序和相关资源库文件，在表 38 列出了其所在位置。

表 38. 附件数据处理程序各组件的位置

组件	位置
“附件”数据处理程序	Integration/WBI/WICS/Attachment/ bcgwbiattachmentdh.jar
资源库文件	Integration/WBI/WICS/Attachment/ MO_DataHandler_DefaultAttachmentConfig.in

根据 Web 服务器的文档，将文件部署到 Web 服务器。

指定附件数据处理程序的位置

WebSphere InterChange Server 需要知道“附件”数据处理程序的位置，以便它在运行时装入该程序。要指定附件数据处理程序的位置，请遵循以下步骤：

1. 编辑 ICS 启动脚本 `start_server.bat`，该脚本位于 InterChange Server 产品目录的 `bin` 子目录下（在 InterChange Server 所在的机器上）。
2. 对于该文件中的 `CLASSPATH` 变量，添加附件数据处理程序的 JAR 文件：

将附件数据处理程序的 JAR 文件 `bcgwbiattachmentdh.jar` 添加到 ICS 启动脚本所包含的 JAR 文件列表中。

配置附件数据处理程序

遵循以下步骤配置“附件”数据处理程序来创建配置业务对象：

- 『创建附件子元对象』
- 第 53 页的『更新顶级数据处理程序元对象』

注：您还必须为附件数据处理程序创建与附件相关的业务对象定义。要了解更多信息，请参阅第 54 页的『创建与附件关联的业务对象定义』。

创建附件子元对象

要配置“附件”数据处理程序，必须创建子元对象以提供“附件”数据处理程序所需的类名和配置特性。要创建这个元对象，需创建一个业务对象定义，该定义中包含了表 39 中所列的属性。使用 WebSphere Business Integration 工具集中的业务对象设计器来创建该业务对象定义。

子元对象提供“附件”数据处理程序所需的类名和配置特性。在业务对象设计器工具中，创建包含您期望接收的各种附件和有效负载的 MIME 类型的子元对象。

如表 39 中显示了子元对象的属性。在第 53 页的图 6 中显示了“附件”数据处理程序的子元对象的一个示例。

注：本章显示的样本业务对象不包括 WebSphere InterChange Server 需要的而附件数据处理程序不使用的标准属性（例如 `ObjectEventId`）。

表 39. 配置附件子元对象中特性

属性名	描述
ClassName	类名（必需），它指向下面这个数据处理程序类： com.ibm.bcg.DataHandlers.AttachmentDataHandler
ContentTypeMap_x	您期望在 XML 包装器中的接收有效负载和个类附件的 content-type 映射。 要了解更多信息，请参阅『content-type 映射』。
PayloadDataHandlerMimeType	MIME 类型用于确定缺省的数据处理程序，该程序处理没有相关附件的有效负载。

请注意：要赋值给表 39 中的属性，请设置属性的缺省值。例如，如果“附件”数据处理程序要为自己缺省的数据处理程序使用 XML 数据处理程序，则将 PayloadDataHandlerMimeType 属性的缺省值设置成 text/xml。

content-type 映射： content-type 映射确定“附件”数据处理程序调用的数据处理程序来将格式信息转换为相关的内容类型。例如，如果有效负载的内容类型是 application/xml，则“附件”数据处理程序搜索 content-type 映射，该映射的 ContentType 属性包含值 application/xml。如果没有发现相匹配的内容类型，则数据处理程序认为不应该将相关的附件转换为业务对象。

您将为这些内容类型中的每个创建 content-type 映射，表 41 列出了属性级特定于应用程序的信息。

当在子元对象中创建表示 content-type 映射属性时，牢记以下规定：

- content-type 映射属性的名称具有如下格式：

ContentTypeMap_x

其中 x 是一个整数，它唯一标识业务对象定义中 content-type 映射。

注：您必须按顺序排列 ContentTypeMap_x 属性。例如，如果有三个 content-type 映射，则必须将它们的属性依次命名为 ContentType_1、ContentType_2 和 ContentType_3。

- content-type 映射属性的缺省值必须包含一些有效标记的组合。

表 40 中列出了可以包含该属性缺省值的标记。

表 40. content-type 映射属性缺省值的有效标记

标记名称	描述	是否必需？
ContentType	在传输包络中出现的实际的内容类型（例如，是 text/xml）。	是
MimeType	MIME 类型，用于确定数据处理程序以将相关的内容类型转换为业务对象。如果没有指定 MimeType，则数据处理程序使用 ContentType 的值来实例化数据处理程序。	否

表 40. *content-type* 映射属性缺省值的有效标记 (续)

标记名称	描述	是否必需?
Charset	<p>字符集名称 (例如, UTF-8), 附件数据处理程序用它来将字节转换成字符串或将字符串转换成字节。</p> <p>如果没有指定 Charset, 则“附件”数据处理程序执行下列操作:</p> <ul style="list-style-type: none"> • 对于入站数据, 根据 base64 对消息进行解码, 将得到的数据字节用于转换成业务对象。 • 对于出站数据, 调用返回字节 (而不是字符串) 的子数据处理程序的方法。 	否
ConvertAttachment	布尔值, 用于指出是否应该将附件转换为业务对象。缺省值是 false。	否

content-type 映射还可以用于指定编码的字符集, 以及是否应该将附件转换成业务对象。有关子元对象属性和示例的描述, 请参阅第 50 页的『创建附件子元对象』。

例如, 假定在文档中包含有如下内容:

- application/xml
- text/xml
- application/octet-stream

表 41. 样本 *content-type* 映射

内容类型	属性名	缺省值
text/xml	ContentType_1	ContentType=text/xml;MimeType=myxml;Charset=UTF-8;ConvertAttachment=false;
application/xml	ContentType_2	ContentType=application/xml;MimeType=mynewxml;Charset=UTF-16; ConvertAttachment=true;
application/octet-stream	ContentType_3	ContentType=application/octet-stream; MimeType=myoctet

样本子元对象: Business Integration Connect 提供了以下 InterChange Server 资源库输入文件, 它包含一个“附件”数据处理程序的样本子元对象:

```
ProductDir/Integration/WBI/WICS/Attachment/
MO_DataHandler_DefaultAttachmentConfig
```

其中, *ProductDir* 是 Business Integration Connect 产品的安装目录。这个资源库文件定义了一个单独的“附件”数据处理程序, 该程序的 MIME 类型是 *wbic_attachment*, 并且该程序相关的子元对象是 *MO_DataHandler_DefaultAttachmentConfig*。图 6 显示了附件数据处理程序的样本子元对象。该元对象定义了两个 *content-type* 映射: *ContentTypeMap_1* 和 *ContentTypeMap_2*。

MO_DataHandler_DefaultAttachmentConfig
名称 = ClassName 缺省值 = com.ibm.bcg.DataHandlers. AttachmentDataHandler 名称 = ContentTypeMap_1 缺省值 = ContentType=application/xml; MimeType=text/xml;CharSet=UTF-8; ConvertAttachment=true; 名称 = ContentTypeMap_2 缺省值 = ContentType=text/xml; MimeType=text/xml;CharSet=UTF-8; 名称 = PayloadDataHandlerMimeType 缺省值 = text/xml

图 6. 附件数据处理程序的样本子元对象

更新顶级数据处理程序元对象

WebSphere Business Integration Adapter (例如 Adapter for JMS) 使用 MO_DataHandler_Default 元对象将来确定它可使用的数据处理程序。WebSphere Server Access 使用 MO_Server_DataHandler 元对象也是出于同样的目的。在这些元对象中的某个添加附件数据处理程序的引用。

为了将文档包含的 MIME 类型与提供这些 MIME 类型支持的数据处理程序相关联, ICS 兼容组件引用顶级数据处理程序元对象。表 42 总结了顶级元对象名称, 这取决于需要访问数据处理程序的组件。

表 42. InterChange Server 顶级数据处理程序元对象

WebSphere Business Integration 组件	ICS 兼容组件	顶级数据处理程序员对象
适配器	Adapter for XML, Adapter for HTTP, MO_DataHandler_Default Adapter for JMS	
使用“服务器访问”的访问客户机	Connect Servlet	MO_Server_DataHandler

对于 MO_Server_DataHandler 或 MO_DataHandler_Default 元对象, 进行以下修改:

- 添加属性, 用其名称确定了附件数据处理程序实例关联的 MIME 类型; 即对于包含这个 MIME 类型的文档, 相关联的数据处理程序可以处理至业务对象的转换。

这个属性的属性类型是附件数据处理程序的子元对象的业务对象定义 (请参阅第 50 页的『创建附件子元对象』)。

- 如果在顶级数据处理程序元对象中不存在受支持的附件 MIME 类型, 则为这些类型中每个添加属性。

这些属性的属性类型将是关联数据处理程序的子元对象。

例如，假设您有“附件”数据处理程序，如同第 53 页的图 6 中所配置。图 7 显示了 MO_DataHandler_Default 元对象，该对象具有将 wbic_attachment MIME 类型与附件数据处理程序的实例相关联的属性，这一处理程序是由 MO_DataHandler_DefaultAttachmentConfig 子元对象配置的。这个顶级数据处理程序元对象还将文档 MIME 类型 (text/xml) 与 XML 数据处理程序的子元对象相关联。

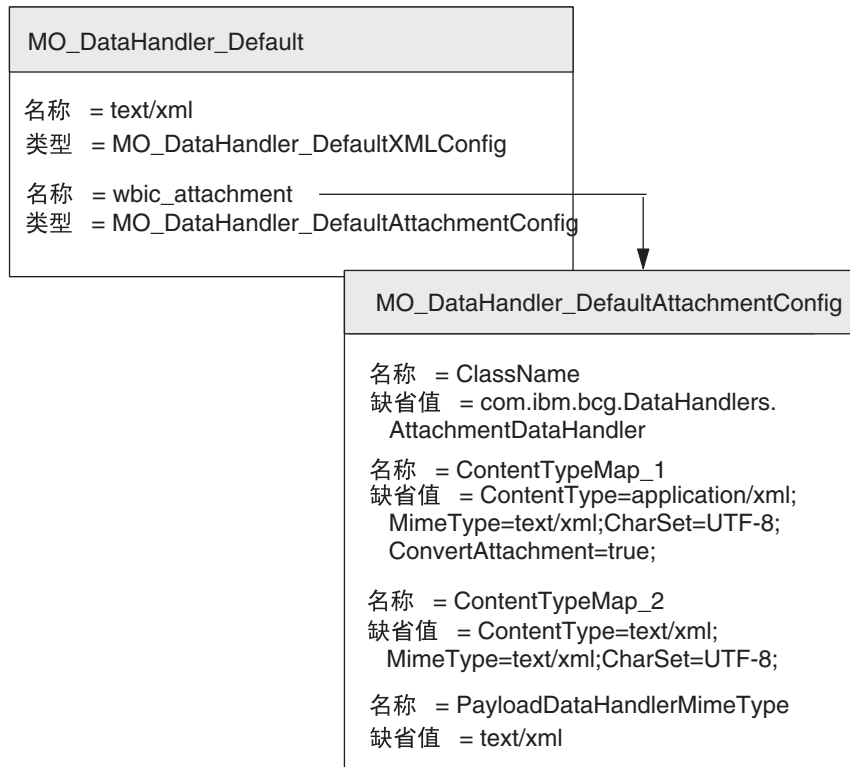


图 7. 将 wbic_attachment MIME 类型与附件数据处理程序相关联

对于您需要支持的每种唯一的受支持内容类型组合，主要向相应的顶级数据处理元对象中添加属性这一过程，该对象的属性名是与附件数据处理实例相关联的 MIME 类型，该对象的类型是相关联子元对象的名称。另外，还要确保经过配置的 MIME 类型（及其子元对象）存在于顶级元对象中。

创建与附件关联的业务对象定义

如果发送或接收包裹在 XML 传输包络中的文档，有效负载业务对象需要包含附件信息。在任何一个文档流中，都有一个有效负载，并可能有多个附件。附件数据处理程序期望该附件信息成为与附件业务对象关联的业务对象里。因此，必须创建业务对象定义以表示该信息。业务对象定义是 InterChange Server 使用的信息格式。可以使用“业务对象设计器”工具创建业务对象定义。

图 8 显示了包装在 XML 传输包络中有效负载的业务对象结构。

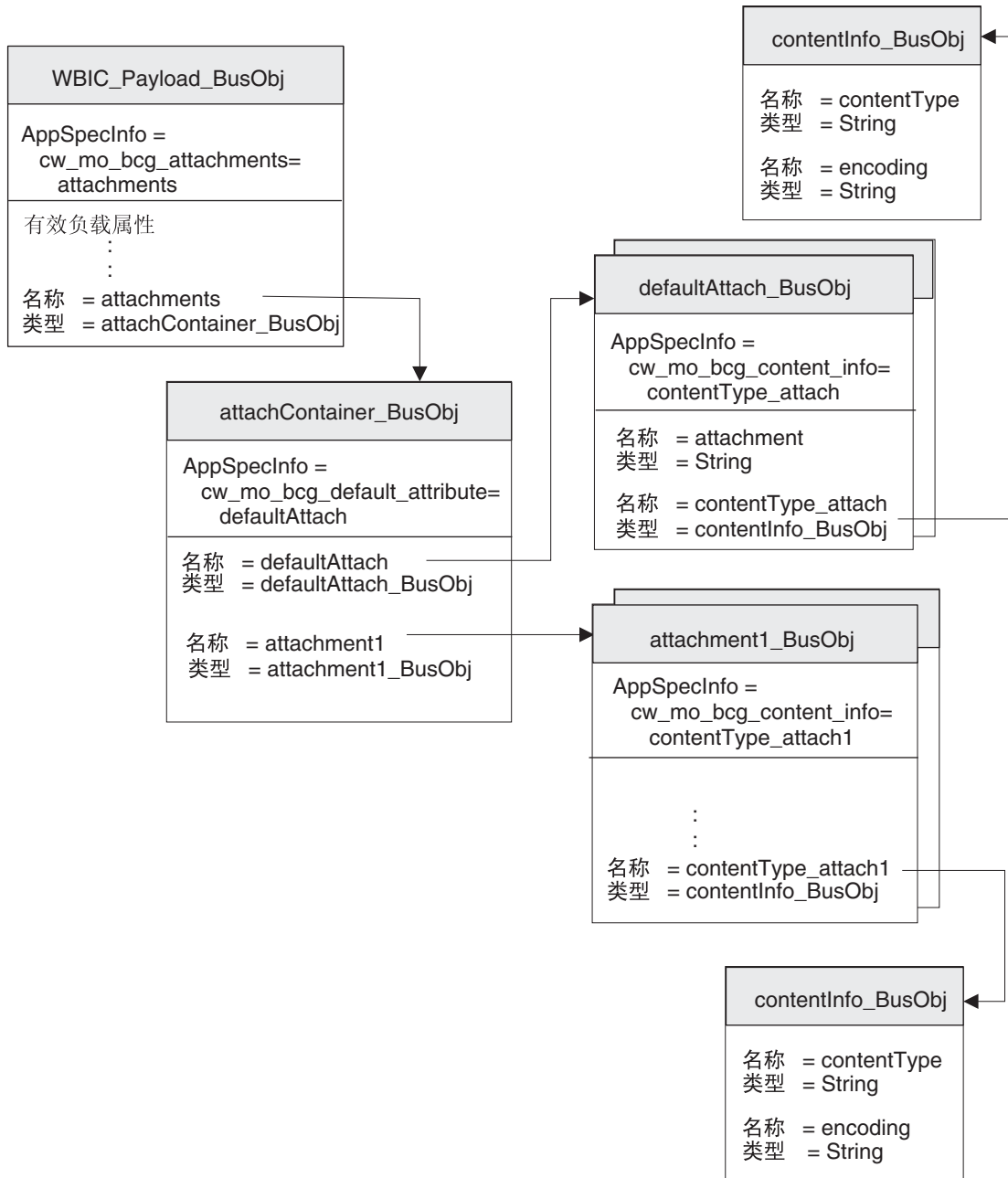


图 8. 有效负载业务对象到附件业务对象间的联系

如图 8 所示，所有附件包含在附件容器业务对象中。如果有附件，则有效负载业务对象具有与附件容器业务对象的属性

通过采取以下步骤来确保您的业务对象结构包含附件所需的业务对象：

1. 创建业务对象定义以保存“后端集成”打包所需的 content-type 编码属性。
2. 为每类附件创建业务对象定义。
3. 为附件容器业务对象创建业务对象定义。
4. 为有效负载业务对象修改业务对象定义。

下面几节中逐一描述了这些步骤。

表示内容信息

为存储相关有效负载或附件的内容类型和编码，要创建**内容信息业务对象**。为创建内容信息业务对象定义，创建表 43 所示的属性。

表 43. 内容信息业务对象的属性

属性	属性类型	描述	是关键属性吗?
contentType	字符串	相关有效负载或附件的内容类型	是
编码	字符串	相关有效负载或附件的字符编码	否

在第 55 页的图 8 中，contentInfo_BusObj 业务对象定义包含了附件内容类型和编码的属性。这些属性都有特定于应用程序的属性级信息以指定相关协议头的名称。例如，x-aux-sender-id 属性具有如下特定于应用程序的信息设置：

```
name=x-aux-sender-id
```

可以选择您想要的任何名字来确定内容信息业务对象定义。附件业务对象的特定于应用程序的信息确定这是否为“内容类型编码”业务对象类型。第 55 页的图 8 显示了名为 contentType_BusObj 的内容类型编码业务对象定义的样本。

表示附件数据

对于没有转换为业务对象的附件数据，创建**缺省的附件业务对象**。这个业务对象有助于包含来自传输封包的 base64 编码数据。

为创建缺省的附件业务对象定义，采取以下步骤：

- 创建 表 43 中所示的属性。
- 如果在缺省附件业务对象定义的特定于应用程序的信息中创建内容信息业务对象，则添加 cw_mo_bcg_content_info 标记以标识包含内容信息的属性。

这个 cw_mo_bcg_content_info 标记有如下格式：

```
cw_mo_bcg_content_info=contentInfoAttr
```

其中 contentInfoAttr 是包含附件容器业务对象的属性名。

表 44. 缺省附件业务对象的属性

属性	属性类型	描述	是关键属性吗?
附件	字符串	附件数据的部分。 注： 这个属性是业务对象定义的关键属性。	是
用于保存内容信息的属性	业务对象	一个可选的属性，用于保存内容信息业务对象，该对象提供内容类型和编码给附件数据。这个属性应当有唯一的基数。 注： 如果这个属性不存在，则附件数据处理程序不在业务对象中设置附件数据。 要了解关于内容信息业务对象的格式的更多信息，请参阅『表示内容信息』。	否

在第 55 页的图 8 中，defaultAttach_BusObj 业务对象定义包含附件数据部分的属性，包括内容信息业务对象以保存它的内容类型和编码。这个缺省附件业务对象表示的附件数据的部分有 content-type 编码，由 contentType_attach 属性表示。因此，缺省附件业务对象定义在它的业务对象级的特定应用程序信息中包含以下标记：

```
cw_mo_bcg_content_info=contentType_attach
```

表示附件

对于转换为业务对象的文档中的每种附件，必须创建一个单独的**附件业务对象定义**。附件业务对象定义表示文档附件中的实际数据。为创建附件业务对象定义，请采取以下步骤：

1. 为附件数据的每个部分创建属性。

属性类型可能包括字符串（针对简单数据部分）或业务对象定义（针对复杂数据）。

2. 如果附件需要 content-type 编码：

- 创建内容类型编码属性。

这个属性的属性类型是内容类型编码业务对象定义（请参阅第 56 页的『表示内容信息』）。

- 添加附件业务对象定义的特定于程序的应用业务对象级 cw_mo_bcg_content_info 的信息。

这个 cw_mo_bcg_content_info 标记有以下格式：

```
cw_mo_bcg_content_info=contentTypeEncodingAttr
```

其中 *contentTypeEncodingAttr* 就是包含 content-type 编码业务对象的属性名。

在第 55 页的图 8 中，有效负载文档有一个属性，由 attachment1_BusObj 业务对象定义表示。这个附件有 content-type 编码，由 contentTypeEncoding 属性表示。因此，附件业务对象定义在它的特定于应用程序的业务对象级信息中包含以下标记：

```
cw_mo_bcg_content_info=contentTypeEncoding
```

表示附件容器

附件容器包含传输封包中的所有文档附件。为表示 InterChange Server 附件容器，要创建**附件容器业务对象**。附件容器业务对象中的每个属性都表示一个附件。

为创建附件容器业务对象定义，请采取以下步骤：

1. 为即将转换为业务对象的文档中的每个附件，添加一个属性。

这些属性中的每个属性类型与附件业务对象关联（请参阅『表示附件』）。每个属性应当有多个基数。

2. 向每个属性的特定于应用程序的信息添加 wbic_type 标记。

wbic_type 标记的格式如下：

```
wbic_type=Attachment
```

注：一个附件属性可以有多个基数。

3. 如果有效负载包含将不转换为业务对象的附件数据：

- 添加缺省附件业务对象的属性。

这个属性的属性类型是缺省附件业务对象（请参阅第 56 页的『表示附件数据』）。这是附件容器业务对象的关键属性。这个属性不需要在特定应用程序信息中的 `wbic_type` 标记。

注：附件容器业务对象可以仅包含一个缺省附件属性。但是，该属性可以有多个基数。

- 附件业务对象定义的特定于应用程序的业务对象级信息，`cw_mo_bcg_default_attribute` 来确认包含附件数据的属性。

这个 `cw_mo_bcg_default_attribute` 标记有如下格式：

```
cw_mo_bcg_content_info=defaultAttachmentAttr
```

其中 `defaultAttachmentAttr` 是包含缺省附件业务对象的属性名。

请注意：如果不存在缺省附件属性，则附件数据处理程序不能转换任何关联 `content-type` 映射的附件，也不能转换未转换为业务对象的附件。这些附件将在至业务对象表示的转换间丢失。

在第 55 页的图 8 中，`attachContainer_BusObj` 业务对象定义表示附件容器。这个附件容器业务对象定义有以下属性：

- `attachment1` 属性表示这个文档的一个附件。因此，附件容器业务对象定义特定于应用程序的属性级信息中包含以下标记：

```
wbic_type=Attachment
```

`attachment1_BusObj` 业务对象定义表示这个附件。

- `defaultAttach` 属性表示不需要转换至业务对象的附件数据。因此，附件容器业务对象定义特定于应用程序的业务信息级信息中包含以下标记。

```
cw_mo_bcg_default_attribute=defaultAttach
```

修改有效负载业务对象定义

有效负载业务对象定义表示文档中的信息。它包含用于各部分信息的属性，这些信息是在 `Business Integration Connect` 和 `InterChange Server` 之间传输信息。关于有效负载业务对象定义的创建信息，请参阅第 40 页的『用于文档的业务对象』。

如果，正在发送或接收包含附件的文档，有效负载业务对象需要包含附件信息。在任何文档流中，都有一个有效负载，并可能有多个附件。如果文档的有效负载包含附件，则必须按以下方式修改有效负载业务对象定义：

- 创建属性以保存有效负载数据。

如果实际的有效负载数据存储于单独的有效负载业务对象定义中，则您会发现更容易使用有效负载。在这种情况下，顶级有效负载业务对象包含有效负载数据的属性，该属性的属性类型是实际有效负载数据的业务对象定义。

- 添加附件容器：

- 添加属性以保存附件容器。

这个属性的属性类型是附件容器业务对象定义（请参阅第 57 页的『表示附件容器』）。这个属性应当有单个的基数。

- 在有效负载业务对象定义的特定于应用程序的信息中添加 `cw_mo_bcg_attachment` 标记以确认包含附件容器的属性。

`cw_mo_bcg_attachment` 标记的格式如下:

```
cw_mo_bcg_attachment=attachContainerAttr
```

其中 `attachContainerAttr` 是包含附件容器业务对象的属性名。

- 可以指定有效负载的内容类型（可选）。附件数据处理程序使用这个内容类型来确定实例化哪个数据处理程序有效负载数据。如果在 `content-type` 映射中发现相匹配的内容类型，则用实例化针对内容类型的数据处理程序。
- 添加内容信息属性，该属性是可选，用它来保存有效负载的内容类型和编码。这个属性应当有单一的基数。

注：如果这个属性不存在，附件数据处理程序获得数据处理程序以在它的子元对象中从 `PayloadDataHandlerMimeType` 配置特性的设置中转换有效负载。

- 在有效负载业务对象定义的特定于应用程序信息中，添加 `cw_mo_bcg_content_info` 标记以确认包含内容信息的属性。

这个 `cw_mo_bcg_content_info` 标记有如下格式:

```
cw_mo_bcg_attachment=contentInfoAttr
```

其中 `contentInfoAttr` 是包含内容信息业务对象的属性名。要了解关于内容信息业务对象格式的更多信息，请参阅第 56 页的『表示内容信息』。

- 传输协议所需任何配置属性。

例如，如果正在使用 JMS 传输协议，则有效负载业务对象定义必须包含 JMS 动态业务对象。有关更多信息，请参阅如何在传输协议支持下创建业务对象定义章节。

第 3 章 通过 HTTP 集成 InterChange Server

本章描述了如何通过 HTTP 传输协议将 WebSphere Business Integration Connect 和 WebSphere InterChange Server 集成在一起。它提供如何配置 InterChange Server (ICS) 的信息, 以及通过 HTTP 进行通信所需的、兼容 ICS 的组件信息。

注: 有关如何配置 WebSphere Business Integration Connect 以便通过 HTTP 与 InterChange Server 进行通信的信息, 请参阅第 37 页的『为 InterChange Server 配置 Business Integration Connect』。要获取有关如何配置 InterChange Server 的常用信息, 请参阅第 40 页的『配置 InterChange Server』。

本章提供了有关如何通过使用 HTTP 传输协议在 WebSphere Business Integration Connect 和 WebSphere InterChange Server 之间发送和接收文档的下列信息:

- 『将 HTTP 传输协议用于 4.2.2 之前版本的 ICS』
- 第 85 页的『将 HTTP 传输协议与 V4.2.2 ICS 一起使用』
- 第 97 页的『通过 HTTP/S 发送 SOAP 文档』

将 HTTP 传输协议用于 4.2.2 之前版本的 ICS

WebSphere Business Integration Connect 可以通过 HTTP 传输协议向 4.2.2 之前版本的 WebSphere InterChange Server (ICS) 发送文档以及从其接收文档。

注:

1. 要通过 HTTP 传输协议在 WebSphere Business Integration Connect 和版本 4.2.2 InterChange Server 之间发送和接收文档, 请参阅第 85 页的『将 HTTP 传输协议与 V4.2.2 ICS 一起使用』。
2. 如果您正通过 HTTP 传输协议交换 SOAP 文档, 则请参阅第 97 页的『通过 HTTP/S 发送 SOAP 文档』。

本节提供如何配置 4.2.2 之前版本的 InterChange Server 以及相应的兼容 ICS 的组件, 以通过 HTTP 与 Business Integration Connect 一起使用的下列信息:

- 『通过 HTTP 将文档发送到 4.2.2 之前版本的 ICS』
- 第 74 页的『通过 HTTP 从 4.2.2 之前版本的 ICS 接收文档』
- 第 79 页的『为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象』
- 第 84 页的『创建用于 HTTP 的 4.2.2 之前版本的 ICS 构件』

通过 HTTP 将文档发送到 4.2.2 之前版本的 ICS

本节提供了下列信息, 以描述如何通过 HTTP 传输协议将文档从 Business Integration Connect 发送到 4.2.2 之前版本的 ICS:

- 第 62 页的『发送所需的组件』
- 第 65 页的『配置 Connect Servlet』
- 第 70 页的『配置包装器数据处理程序』
- 第 73 页的『创建用于发送文档的业务对象定义』

从 Business Integration Connect 发送到 InterChange Server 的文档会启动 InterChange Server 中的事件通知。

发送所需的组件

Business Integration Connect 可以通过 HTTP 传输协议将文档发送到下列 4.2.2 之前版本的 InterChange Server。

- V4.1.1
- V4.2.0
- V4.2.1

要使 Business Integration Connect 能使用 HTTP 传输协议将文档发送到 4.2.2 之前版本的 ICS，需要配置这两个组件。表 45 中概述了这些配置步骤。

表 45. 配置 Business Integration Connect 和 InterChange Server

组件	版本	更多信息
WebSphere Business Integration Connect	4.2.2	第 37 页的『配置通过 HTTP 传输协议传递的出站文档』 第 39 页的『配置通过 HTTP 传输协议传递的入站文档』
WebSphere InterChange Server	4.1.1, 4.2.0, 4.2.1	第 84 页的『创建用于 HTTP 的 4.2.2 之前版本的 ICS 构件』

另外，要通过 HTTP 传输协议将文档发送到 ICS，您可以使用表 46 中列出的兼容 ICS 的组件。其中的大多数组件是作为 Business Integration Connect 发行版一部分提供的。

表 46. 通过 HTTP 将文档发送到 4.2.2 之前版本的 ICS 所需的组件

组件	描述	注意事项和限制
WebSphere Business Integration Connect Servlet (Connect Servlet)	这个 servlet 是 WebSphere InterChange Server 访问客户机。访问客户机是 InterChange Server (ICS) 的外部过程，可以请求执行 ICS 内部的协作。	4.2.2 之前版本的 WebSphere InterChange Server 支持 servlet。 注：4.2.2 版本的 WebSphere InterChange Server 不支持 servlet。
包装器数据处理程序	Connect Servlet 调用这个数据处理程序来将 HTTP 消息转换成相应的数据业务对象。它调用适合于您消息的数据处理程序。例如，如果有效负载是 XML 格式的，则可以配置包装器数据处理程序，以调用 XML 格式的数据处理程序。	无
有效负载数据处理程序	该数据处理程序可以使文档有效负载在它的文档格式（通常是 XML）和它的业务对象表示之间进行转换。	该数据处理程序是必需的，并且它必须支持 MIME 类型的有效负载文档。
附件数据处理程序	该数据处理程序处理文档消息的附件文档。	仅当文档包括附件时，才需要该数据处理程序。

注：表 46 中所列的所有组件都包含在 Business Integration Connect 安装介质中。要了解关于这些组件位置的信息，请参阅第 65 页的『部署 Connect Servlet』。

图 9 提供了 Business Integration Connect 如何通过 HTTP 传输协议将文档发送到 4.2.2 之前版本的 ICS 的概述。

注：包装器数据处理程序、附件数据处理程序和有效负载数据处理程序都是在 InterChange Server 内执行的。

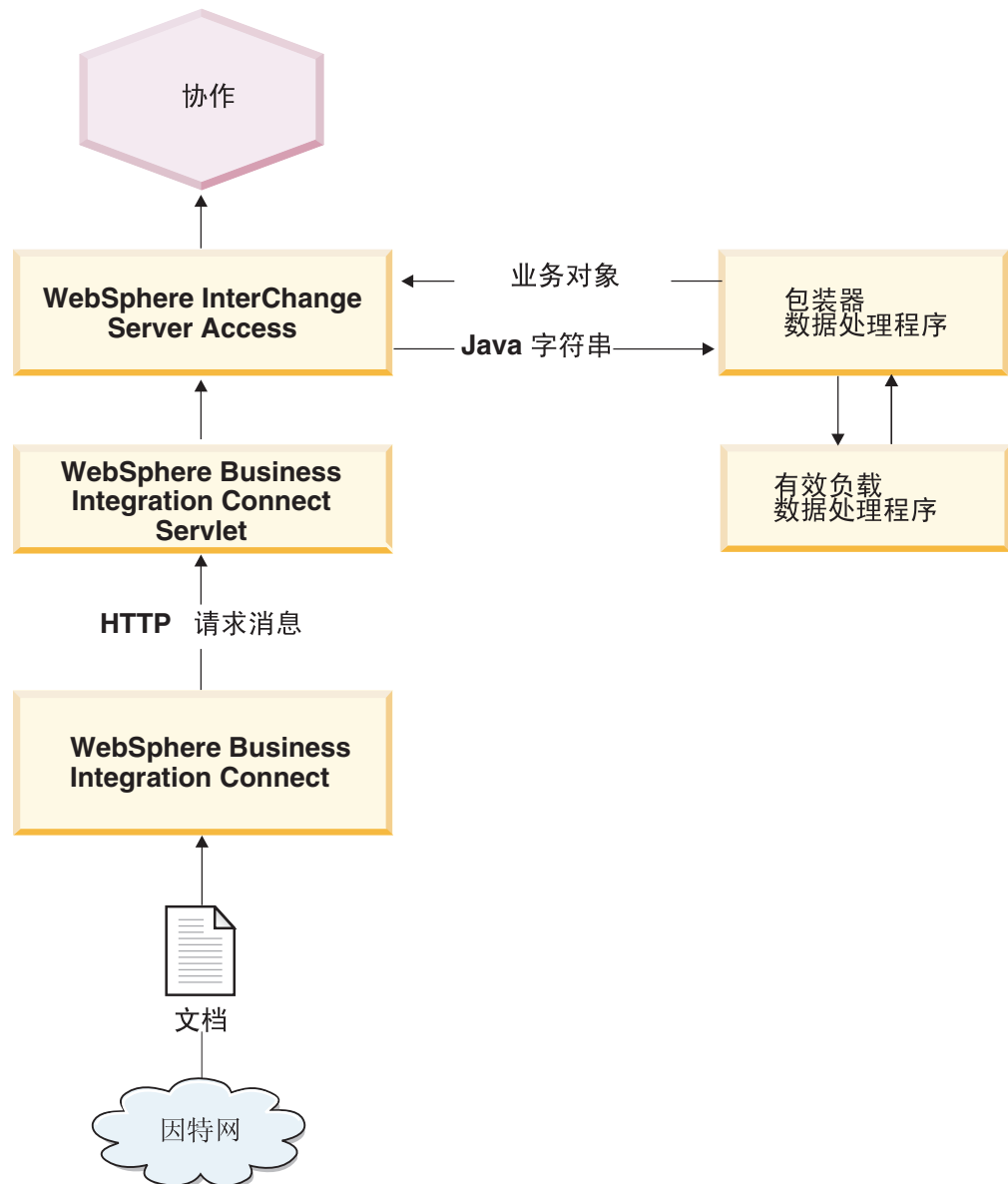


图 9. 通过 HTTP 传输协议从 Business Integration Connect 到协作的消息流

如图 9 所示，WebSphere Business Integration Connect Servlet 是兼容 ICS 的组件，Business Integration Connect 可以与其直接交互。该 Connect Servlet 是访问客户机，它是 InterChange Server 外部过程，可以请求执行 ICS 协作。该访问客户机从名为服务器访问接口的应用程序编程接口（API）发出调用，以与 ICS 进行交互。WebSphere InterChange Server Access 接收并解释这些调用，前者是 ICS 内部的一个组件，ICS 用于处理与访问客户机的交互。服务器访问接口同步调用协作。

注：

1. 即使 Business Integration Connect 和后端系统之间的某些交互是异步的，Server Access 仍然同步调用协作并一直等到该协作执行完毕。

2. 有关访问客户机和 Server Access 的更多详细信息, 请参阅 WebSphere InterChange Server 文档集中的访问开发指南。

以下步骤描述了 Business Integration Connect 如何通过 HTTP 传输协议将文档发送到 ICS 中的协作, 以此来参与事件通知的:

1. Business Integration Connect 调用 WebSphere Business Integration Connect Servlet 以将文档发送到 InterChange Server。

Business Integration Connect 将文档发送到指定为目标网关的 URL。

注: Connect Servlet 可用于调用多个协作。

2. 从 Business Integration Connect 发送的 HTTP 请求消息中, Connect Servlet 创建一个 Java 字符串。

HTTP 请求消息包含两部分:

- HTTP 传输协议头:
 - 标准头
 - 定制头, 如果为文档指定后端集成封装, 则 Business Integration Connect 设置该头
- 消息, 它的格式由使用的封装类型来决定

3. Connect Servlet 检查其 servlet 属性文件, 以确定要调用的协作以及要使用的谓词和 MIME 类型。

每个 URL 都对应一个要调用的协作。(请参阅第 65 页的『配置 Connect Servlet』以获取信息。)

4. Connect Servlet 使用服务器访问接口中的调用将 Java 字符串连同 Servlet 属性文件中的信息发送到 WebSphere InterChange Server Access。

由于 Connect Servlet 仅能将文档发送到 InterChange Server (它不能接收文档), 所以它仅能参与 InterChange Server 的事件通知。

注: 要支持利用 InterChange Server 的请求处理, Business Integration Connect 必须与 WebSphere Business Integration Adapter for XML 进行交互。要了解更多信息, 请参阅第 74 页的『通过 HTTP 从 4.2.2 之前版本的 ICS 接收文档』。

5. WebSphere InterChange Server Access (属于 InterChange Server) 接收 Java 字符串并调用包装器数据处理程序。

包装器数据处理程序的任务是将 Java 字符串转换成相应的业务对象结构。它是 InterChange Server 期望作为输入的业务对象。

6. 包装器数据处理程序采用下列步骤将 Java 字符串转换成业务对象结构:

- a. 从 Java 字符串中抽取头和有效负载。

注: 如果 Business Integration Connect 发送的文档包含附件, 则可以配置包装器数据处理程序, 以调用附件数据处理程序。第 44 页的『处理带附件的文档』中描述了附件数据处理程序的操作。

- b. 验证 MIME 类型的有效负载, 并调用已经为 MIME 类型配置好的数据处理程序, 以将有效负载转换成有效负载业务对象。
- c. 创建 HTTP 属性业务对象和动态业务对象。

在 HTTP 属性业务对象中设置 HTTP 头，该业务对象是这个有效负载业务对象的动态元对象的子代。

- d. 创建顶级业务对象，并将事件业务对象设置成它的请求业务对象。

包装器数据处理程序期望有效负载业务对象能有一个分等级的结构。有关该有效负载业务对象的结构信息，请参阅第 73 页的『创建用于发送文档的业务对象定义』。

- e. 将顶级业务对象返回给 InterChange Server 内部的 Server Access 。
7. Server Access 调用协作，并将它传递给顶级业务对象。

请确保将要调用的协作对象的协作端口配置为外部端口。请参阅 WebSphere InterChange Server 文档，以获取有关如何配置端口的详细信息。

8. 该协作执行顶级业务对象并将它返回到包装器数据处理程序。

是否填充响应业务对象（在顶级业务对象内）取决于 InterChange Server 与 Business Integration Connect 之间交互的类型，具体如下：

- 对于异步交互，协作将不填充响应业务对象。
- 对于同步交互（在这些交互中将在同一个 HTTP 连接中返回响应），协作将填充响应业务对象。

要了解更多信息，请参阅第 81 页的『响应业务对象』。

9. 如果交互成功，Connect Servlet 就将 HTTP 200 OK 应答返回给 Business Integration Connect。

配置 Connect Servlet

WebSphere Business Integration Connect Servlet 是一个访问客户机，它是 InterChange Server 的外部过程，可以请求执行 InterChange Server 内的协作。该访问客户机从名为服务器访问接口的应用程序编程接口（API）使用调用，以与 ICS 进行交互。WebSphere InterChange Server Access（InterChange Server 的组件，负责处理与访问客户机的交互）接收到这些调用并对它们进行解释。

注：有关访问客户机和 Server Access 的更多详细信息，请参阅 WebSphere InterChange Server 文档集中的访问开发指南。

Connect Servlet 的配置包括下列步骤：

- 『部署 Connect Servlet』
- 第 66 页的『设置 Connect Servlet 属性』

部署 Connect Servlet: Connect Servlet、包装器数据处理程序和包装器数据处理程序的库文件在 Business Integration Connect 的安装盘中都可获得的，其位置在表 47 中列出。

表 47. Connect Servlet 组件的位置

组件	位置
Connect Servlet	Integration/WBI/WICS/WBICServlet/ bcgwbiservlet.war
包装器数据处理程序	Integration/WBI/WICS/WBICServlet/ bcgwbirapperdh.jar
包装器数据处理程序的库文件	Integration/WBI/WICS/WBICServlet MO_DataHandler_WBIWrapper.in

注: 如果您想发送包含附件的文档, 也可以如第 50 页的『部署附件数据处理程序』中所描述的那样部署附件数据处理程序及其相关联的库文件。

该 servlet 可以连接至 WebSphere InterChange Server V4.1.1、4.2.0 和 4.2.1。可以在支持任意这些 InterChange Server 版本的平台上部署该 servlet。另外,您需要确保该平台上支持服务器访问接口。参阅 WebSphere InterChange Server 文档, 以获取您正使用的 ICS 版本所支持的平台的列表。

要部署第 65 页的表 47 中的组件, 请遵循这些步骤:

1. 根据 Web 服务器的文档, 将 Connect Servlet 及其关联文件部署到 Web 服务器。
2. 请确保下列文件位于 Connect Servlet 的 CLASSPATH 中:
 - crossworlds.jar
 - vbjorb.jar

这些文件可以在 InterChange Server 产品目录的 lib 子目录中找到。

注:

- a. 这些文件所在的 InterChange Server 必须与您要调用的 InterChange Server 是同一个版本。
 - b. 这些文件对您的 Web 服务器中 Connect Servlet 的 Web 容器必须是可用的。有关如何使这些文件对 Web 容器可用的更多信息, 请查看您的 Web 服务器文档。
3. 请确保下列文件位于 Connect Servlet 的 CLASSPATH 中:
 - mail.jar
 - log4j-1.2.8.jar

可以在 Business Integration Connect 安装介质的下列目录中找到这些文件:

integration/wbi/wics/http/lib/thirdparty

注: 这些文件对您 Web 服务器中 Connect Servlet 的 Web 容器必须是可用的。有关如何使这些文件对 Web 容器可用的更多信息, 请查看您的 Web 服务器文档。

4. 使 InterChange Server 可互操作对象引用 (.ior) 文件可以在部署 Connect Servlet 的机器上使用。

如果 Connect Servlet 与 InterChange Server 不在一台机器上, 则您可以采取下列操作之一, 以使 .ior 文件可用:

- 将 InterChange Server 机器中的文件复制到安装了 Business Integration Connect 的机器中。
- 将文件放到 InterChange Server 和 Business Integration Connect 都可以访问的共享位置中。

注: 您还需要用这个 .ior 文件的位置更新 Connect Servlet 属性文件中的 ICS_IORFILE 属性。要了解更多信息, 请参阅第 67 页的『标识 InterChange Server 实例』。

设置 Connect Servlet 属性: 如第 61 页的『通过 HTTP 将文档发送到 4.2.2 之前版本的 ICS』所述, **servlet** 属性文件中包含了 WebSphere Business Integration Connect

Servlet 调用协作所需的信息，比如端口名和谓词。您必须创建该 servlet 属性文件，指定有关 WebSphere InterChange Server 的常规信息。然后，对于您希望该 servlet 调用的任何协作，提供关于该协作的信息。

本节提供了关于如何设置 Connect Servlet 属性的下列信息：

- 『创建 servlet 属性文件』
- 第 69 页的『标识 servlet 属性文件的位置』

创建 *servlet* 属性文件： servlet 属性文件包含了表 48 中所列的组成部分。

表 48. *servlet* 属性文件的组成部分

Servlet 属性文件的组成部分	描述	了解更多信息
一般信息	用于标识 InterChange Server 实例的属性	『标识 InterChange Server 实例』
协作信息	用于标识每个要调用协作的属性	『标识要调用的协作』
日志记录信息	用于配置 servlet 日志记录的属性	第 69 页的『指定 servlet 日志文件的位置』

标识 *InterChange Server* 实例： Connect Servlet 属性文件的第一部分包含了一般信息，用以标识与 Business Integration Connect 进行通信的 InterChange Server 实例。该 ICS 实例包含 Business Integration Connect 需要调用的协作。表 49 显示了 servlet 属性文件的一般属性。

表 49. *servlet* 属性文件的一般属性

属性名	描述	示例
ICS_SERVERNAME	WebSphere InterChange Server 所运行的主机。	Server1
ICS_VERSION	WebSphere InterChange Server 的版本号。可能的值有 4.1.1、4.2.0 和 4.2.1。	4.2.0
ICS_IORFILE	可互操作对象引用 (.ior) 文件的文件名，它用来访问 WebSphere InterChange Server Access。这个示例显示了如何指定 Windows 系统上的路径。 注： 这个路径应该在一行中输入。	c:/myiorlocation/Server1ICS.ior
ICS_USERNAME	连接 WebSphere InterChange Server 的用户标识。	admin
ICS_PASSWORD	连接 WebSphere InterChange Server 的密码。	null
ICS_ENCRYPTED_PASSWORD	指出 ICS_PASSWORD 是否进行加密。如果对密码进行加密，则 servlet 将这个字段设置成 true。	false
ICS_DISABLEENCRYPTION	指出密码加密功能已禁用 (true) 还是已启用 (false)。如果想让密码进行加密，则将这个字段设置成 false。	true

注： 要获取定义了表 49 “示例” 列中所列值的样本 servlet 属性文件，请参阅第 69 页的『样本 servlet 属性文件』。

标识要调用的协作： Connect Servlet 属性文件的第二部分包含了协作信息，它将协作 URL 与相关的协作属性相关联。这一部分用两部分标识了协作 URL，如下所示：

- WBIC_SERVLET_COUNT 属性指定 servlet 属性文件的这个协作部分所标识的协作数量。

它在这个文件中指定了整数数目的 URL：

- 如果将其设置为 1，则 Connect servlet 将处理用 WBIC_URL_1 属性所定义的 URL。

- 如果它设为 2，则 Connect servlet 将处理用 WBIC_URL_1 和 WBIC_URL_2 属性所定义的 URL。
- WBIC_URL_count 属性标识协作的相对 URL。

相关的协作属性具有格式为 WBIC_URL_count_propertyName 的属性名。表 50 定义了这些 WBIC_URL_count_propertyName 属性的样本。在“示例”列中，该表提供了第一个协作 URL (count 为 1) 的 WBIC_URL_count_propertyName 属性的样本值。

表 50. servlet 属性文件的协作属性

属性名	描述	示例
WBIC_SERVLET_COUNT	这个文件中所配置的 URL 数: <ul style="list-style-type: none"> • 如果将它设置为 1，则 servlet 将处理 WBIC_URL_1 的 URL 和属性。 • 如果将它设置为 2，则 Connect servlet 将处理用 WBIC_URL_1 和 WBIC_URL_2 的 URL 和属性。 	1
WBIC_URL_1	相对 URL 的名称	PurchaseOrder
WBIC_URL_1_COLLAB	协作的名称	PurchaseOrderCollab
WBIC_URL_1_PORT	协作的端口名	From
WBIC_URL_1_VERB	协作所预定的谓词	Create
WBIC_URL_1_WRAPPER_MIME	包装器数据处理程序所支持的 MIME 类型。请注意示例是小写的。	wbic/wrapper
WBIC_URL_1_CHARENCODE	HTTP 请求所用的字符编码。请指定有效的 Java 字符编码。	UTF-8

注：要获取定义了表 50 “示例”列中所列值的样本 servlet 属性文件，请参阅第 69 页的『样本 servlet 属性文件』。

servlet 属性文件的协作部分提供了相对 URL，以标识要执行的协作。为了能够在运行时查找协作，Connect Servlet 将下列几条信息合并起来：

- 标识 Connect Servlet 位置的 URL
- 相对 URL，为 Connect Servlet 属性文件中的协作而指定。

例如，如果您使用表 50 中所示的样本值，则 Connect Servlet 需要获取 PurchaseOrderCollab 协作的 URL。为查找该 URL，servlet 采取了下列步骤：

1. 获取 servlet URL，它标识了 Connect Servlet 的位置。

Servlet 包含来自您 Web 服务器的 servlet URL。例如，假定您在下列位置部署了 Connect Servlet：

`http://www.yourcompany.com/tasks`

2. 将它 WBIC_URL_count 属性中的路径追加到 servlet URL。

在表 50 中，WBIC_URL_1 属性包含值“PurchaseOrder”。因此，Connect Servlet 将把这个字符串追加到 servlet URL，以获得协作的下列 URL：

`http://www.yourcompany.com/tasks/PurchaseOrder`

在协作属性中，WBIC_URL_1_WRAPPER_MIME 属性指定用于包装器数据处理程序的 MIME 类型。如果您指定多个 MIME 类型，则需要多个元对象。请参阅第 70 页的『创建包装器子元对象』以获取信息。

指定 *servlet* 日志文件的位置: 日志记录属性是在 Connect Servlet 属性文件的第三部分中指定的。通过添加下列语句在属性文件中指定 *servlet* 日志文件的位置:

```
log4jappender.RollingFile.File=logFileLocation
```

如图 10 所示, `log4jappender.RollingFile.File` 属性是用来配置 Log4J 的 *servlet* 属性文件的一部分。为配置 Connect Servlet, 您只需指定日志文件的位置, 即设置 `log4jappender.RollingFile.File` 属性。如果您熟悉 Log4J, 您也能设置它的其它属性。

样本 *servlet* 属性文件: 图 10展示了 *servlet* 属性文件的一个示例, 它配置了表 49 和表 50 “示例” 列中的值。

```
# WebSphere Business Integration 的样本属性文件
# Connect Servlet
ICS_SERVERNAME=Server1
ICS_VERSION=4.2
ICS_IORFILE=C:/myiorlocation/Server1InterChangeServer.ior
ICS_USERNAME=admin
ICS_PASSWORD=null
ICS_ENCRYPTED_PASSWORD=false
ICS_DISABLEENCRYPTION=true

# Collaboration properties for single collaboration
WBIC_SERVLET_COUNT=1

WBIC_URL_1=PurchaseOrder
WBIC_URL_1_COLLAB=PurchaseOrderCollab
WBIC_URL_1_CHARENCODE=UTF-8
WBIC_URL_1_PORT=From
WBIC_URL_1_VERB=Create
WBIC_URL_1_WRAPPER_MIME=wbic/wrapper

#Log4J Debug Properties
#Possible Categories - debug/info/warn/error/fatal
#Default Category "error". Output to: stdout and RollingFile
log4j.rootCategory=debug,RollingFile
log4j.appender.RollingFile=org.apache.log4j.RollingFileAppender

#Log File Name
log4j.appender.RollingFile.File=D:\\_DEV\\servlet.log
log4j.appender.RollingFile.MaxFileSize=1000KB

#Number of backup files to keep
log4j.appender.RollingFile.MaxBackupIndex=10
log4j.appender.RollingFile.layout=org.apache.log4j.PatternLayout
log4j.appender.RollingFile.layout.ConversionPattern=
    %d{yyyy-MM-ddHH:mm:ss} %-5p [%c{1}] - %m%n
```

图 10. 样本 *servlet* 属性文件

还可以在 Business Integration Connect 安装介质的 SAMPLES 目录中找到样本 *servlet* 属性文件。

标识 *servlet* 属性文件的位置: Connect Servlet 的部署描述符 `web.xml` 为 *servlet* 提供了初始化参数。要标识 *servlet* 属性文件的位置, 请设置这个部署描述符中的 `WBIC_FILENAME` 参数。该参数指定了 Connect *servlet* 属性文件的绝对路径名。

例如, 如果图 10 中所示的样本 *servlet* 属性文件被称为 `connectServlet.cfg`, 并且它位于 Connect Servlet 的部署目录中 (例如 `C:\WBIC\integration`), 则您需要将 `WBIC_FILENAME` 参数设置成如下值:

```
C:\WBIC\integration\connectServlet.cfg
```

配置包装器数据处理程序

包装器数据处理程序将文档由序列化格式（该格式由 Connect Servlet 从 HTTP 消息创建）转换成其相应的业务对象。当 Connect Servlet 调用协作时，它将 Business Integration Connect 发送给它的序列化格式文档发送给 InterChange Server。该协作请求被驻留在 InterChange Server 中的 WebSphere Server Access 接收。如第 63 页的图 9 所述，Server Access 调用包装器数据处理程序，向其传递 Business Integration Connect 文档。数据处理程序返回相应的有效负载业务对象。

要配置包装器数据处理程序，请遵循如下步骤：

- 『指定包装器数据处理程序的位置』
- 『创建用于包装器数据处理程序的配置业务对象』

下面几节概括了配置包装器数据处理程序的步骤。要获取有关数据处理程序的一般信息，请参阅 WebSphere InterChange Server 文档集中的 *数据处理程序指南*。

指定包装器数据处理程序的位置： InterChange Server 需要知道包装器数据处理程序的位置，以便在运行时可以装入该处理程序。要指定位置，请采用下列步骤：

1. 编辑 ICS 启动脚本 `start_server.bat`，该脚本位于 InterChange Server 产品目录（InterChange Server 所驻留的机器）的 `bin` 子目录中。
2. 对于该文件，将包装器数据处理程序的 JAR 文件（`bcgwbwrapperdh.jar`）添加到包含在 ICS 中的 JAR 文件列表中。通常，将数据处理程序 JAR 文件添加到 ICS 启动脚本中的 `DATAHANDLER` 变量中。

注：如果您安装了可选的附件数据处理程序，那么您必须将它的 JAR 文件添加到 ICS 启动脚本中。要了解更多信息，请参阅第 50 页的『指定附件数据处理程序的位置』。

创建用于包装器数据处理程序的配置业务对象： 为确定要调用的数据处理程序，Server Access（位于 InterChange Server 内）检查顶级数据处理程序元对象 `MO_Server_DataHandler`。该文件位于 InterChange Server 产品目录的下列子目录中：

`repository\edk`

这个顶级元对象将 MIME 类型与子代元对象相关联，这个子代元对象中包含了数据处理程序的配置信息。因此，创建配置业务涉及到下列步骤：

1. 『创建包装器子元对象』

您必须用包装器数据处理程序的配置信息对子代元对象进行初始化。

2. 第 72 页的『编辑 `MO_Server_DataHandler` 元对象』

您必须在这个元对象中创建一个项，它将 MIME 类型与包装器数据处理程序的子代元对象名称相关联。

创建包装器子元对象： 要配置包装器数据处理程序，必须创建其子代元对象，并用配置信息对这个子代元对象进行初始化。数据处理程序使用这个元对象的属性来获取其配置信息，包括要进行实例化的数据处理程序类的名称。要创建这个元对象，需创建一个业务对象定义，该定义中包含表 51 中所列的属性。

注：使用业务对象设计器来创建这个业务对象定义。

表 51. 包装器子代元对象的配置属性

属性	描述
ClassName	类名（必需），它指向下面这个数据处理程序类： <code>com.ibm.bcg.integration.wbi.datahandlers.WBICWrapperDataHandler</code>
TopBOPrefix	用于确定顶级业务对象名的前缀。如果为请求所配置的数据处理程序所返回的请求业务对象在其特定于应用程序的业务对象级信息中没有 <code>wbic_mainboname</code> 标记，则将 <code>TopBOPrefix</code> 添加到请求业务对象名中，从而得到顶级对象的名称。
wbic_request_mime	包装器数据处理程序调用以处理请求消息有效负载的数据处理程序支持的 MIME 类型。请确保已经对这个数据处理程序进行了配置，以便 WebSphere InterChange Server Access 可以调用它。要获取更多信息，请参阅第 72 页的『编辑 <code>MO_Server_DataHandler</code> 元对象』。 注：如果您的文档中包含附件，则该配置属性的 MIME 类型应该就是调用附件数据处理程序的 MIME 类型。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。
wbic_response_mime	包装器数据处理程序将调用以处理响应消息有效负载的数据处理程序的 MIME 类型。 注：如果 Business Integration Connect 没有得到预期的响应，则您没有设置 <code>wbic_response_mime</code> 。

要点：要赋值给表 51 中的属性，请设置属性的缺省值。例如，如果包装器数据处理程序打算对其请求消息使用定界数据处理程序，则将 `wbic_request_mime` 属性的缺省值设为 `text/delimited`。

您可以为您需要使用的每个包装器数据处理程序实例定义子代元对象。例如，如果您需要仅支持一个请求 MIME 类型或一个请求 MIME 类型和响应 MIME 类型的组合，则可以创建单个子代元对象，并相应地设置 `wbic_request_mime` 和 `wbic_response_mime` 属性的缺省值。然而，如果您需要支持请求 MIME 类型和响应 MIME 类型的各种不同组合，则可以为每个受支持的组合创建一个子代元对象。

Business Integration Connect 提供以下 InterChange Server 资源库文件，它包含包装器数据处理程序的样本子元对象：

`ProductDir/Integration/WBI/WICS/WBICServlet/MO_DataHandler_WBICWrapper.in`

此处的 `ProductDir` 是您安装 Business Integration Connect 产品的目录。该资源库文件定义了包装器数据处理程序的一个实例，它被配置成调用请求和响应业务对象的定界数据处理程序。图 11 显示了名为 `MO_DataHandler_WBICWrapper` 的样本子代元对象。

MO_DataHandler_WBICWrapper
名称 = ClassName 缺省值 = com.ibm.bcg.integration.wbi. Datahandlers.WBICWrapperDataHandler
名称 = TopBOPrefix 缺省值 = WBIC
名称 = wbic_request_mime 缺省值 = text/delimited
名称 = wbic_response_mime 缺省值 = text/delimited

图 11. 包装器数据处理程序的样本子代元对象

如果您还需要支持请求消息格式为 XML 的文档，则要创建第二个子代元对象，以表示包装器数据处理程序的第二个实例。在这个子代元对象中，wbic_request_mime 属性的缺省值具有 text/xml MIME 类型。

编辑 MO_Server_DataHandler 元对象: WebSphere InterChange Server Access 使用名为 MO_Server_DataHandler 的顶级元对象，来将访问客户机能处理的 MIME 类型与支持这些 MIME 类型的数据处理程序相关联。特别，这个顶级元对象将 MIME 类型与数据处理程序子代元对象相关联。

MO_Server_DataHandler 元对象是业务对象定义。因此，要编辑这个元对象，请在对象设计器中启动 MO_Server_DataHandler，并给它添加每个受支持的包装器数据处理程序实例的新属性。这个数据处理程序的每个实例都是请求 MIME 类型和响应 MIME 类型的唯一组合。

对于 MO_Server_DataHandler 元对象，您要进行下列修改：

- 添加一个属性，其名称标识了与包装器数据处理程序实例相关联的 MIME 类型；也就是说，对于包含该 MIME 类型的文档来说，相关联的数据处理程序能处理其到业务对象的转换。

该属性的属性类型就是包装器数据处理程序的子代元对象的业务对象定义（请参阅第 70 页的『创建包装器子元对象』）。

- 如果 MO_Server_DataHandler 元对象中尚不存在受支持的请求和响应 MIME 类型，则为每个 MIME 类型添加一个属性。

这些属性的属性类型将是相关数据处理程序的子代元对象。

例如，假设您将包装器数据处理程序配置成如图 11 所示的那样。图 12 显示了 MO_Server_DataHandler 元对象，该元对象的属性将 wbic_wrapper MIME 类型由 MO_DataHandler_WBICWrapper 子代元对象配置的包装器数据处理程序的实例关联起来。这个 MO_Server_DataHandler 元对象还将请求和响应 MIME 类型（text/delimited）与定界数据处理程序的子代元对象相关联。

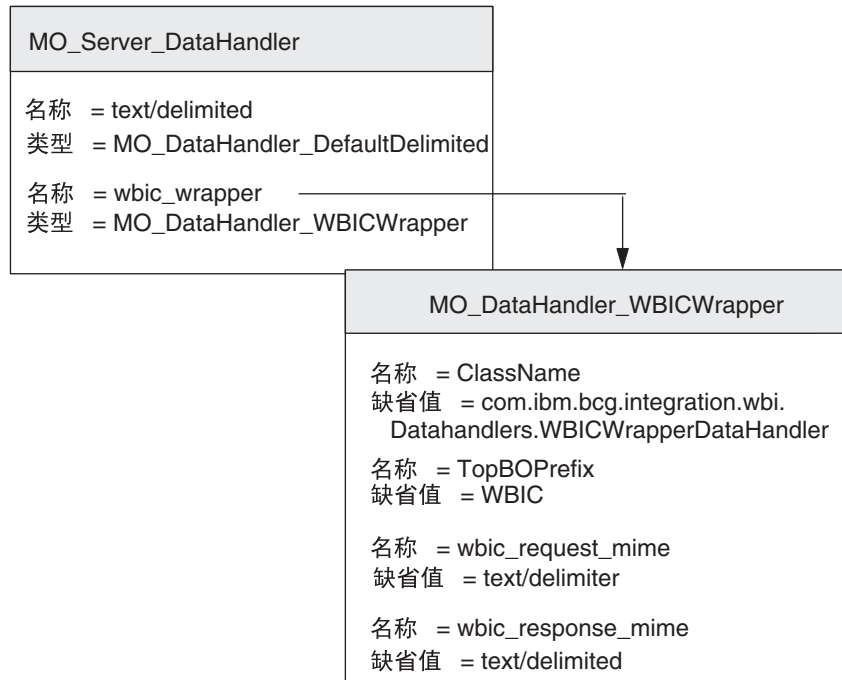


图 12. 将 *wbic_wrapper* MIME 类型和包装器数据处理程序相关联

对于您需要支持的每种唯一的请求和响应 MIME 类型组合，请通过在 `MO_Server_DataHandler` 顶级元对象中添加属性来重复这个过程，其中这个属性的名称是与包装器数据处理程序实例相关联的 MIME 类型，它的类型是相关联的子元对象名。您还要确保已配置的请求和响应 MIME 类型（及其子元对象）存在于 `MO_Server_DataHandler` 中。

注：如果您使用附件数据处理程序来处理 Business Integration Connect 文档中的附件，则还必须修改 `MO_Server_DataHandler` 以支持附件数据处理程序，如第 50 页的『配置附件数据处理程序』所描述的那样。

创建用于发送文档的业务对象定义

WebSphere Business Integration Connect Servlet 将您的文档以有效负载业务对象的格式发送到 InterChange Server。对于 Connect Servlet，有效负载业务对象描述为层次结构的业务对象。包装器数据处理程序在接收到 Business Integration Connect 文档时创建这个业务对象层次结构。因此，您必须创建业务对象定义以表示这个层次结构。

因为 Connect Servlet 仅参与 InterChange Server 的事件通知，所以顶级业务对象的请求和响应属性将是如表 52 所示的那样解释的。

表 52. 事件通知中的请求和响应业务对象

属性	用途
请求业务对象	包含来自 Business Integration Connect 的请求消息，该消息是触发协作的事件。
响应业务对象	包含响应消息，如果交互是同步的。

有关如何创建该业务对象结构的更多信息，请参阅第 79 页的『为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象』。

通过 HTTP 从 4.2.2 之前版本的 ICS 接收文档

本节提供了下列信息以描述如何通过 HTTP 传输协议从 4.2.2 之前版本的 InterChange Server 接收来自 Business Integration Connect 的文档：

- 『接收操作所需的组件』
- 第 76 页的『设置 4.2.2 之前版本的 ICS 的 HTTP 环境』
- 第 78 页的『创建用于接收文档的业务对象定义』

Business Integration Connect 接收来自 InterChange Server、由 InterChange Server 中的请求处理发送的文档。

接收操作所需的组件

Business Integration Connect 可以通过 HTTP 传输协议接收来自下列 4.2.2 之前版本的 InterChange Server 的文档。

- V4.1.1
- V4.2.0
- V4.2.1

要使 Business Integration Connect 能通过 HTTP 传输协议从 4.2.2 之前版本的 InterChange Server 接收文档，需要配置两个组件。第 62 页的表 45 概括了这些配置步骤。另外，要通过 HTTP 协议从 InterChange Server 接收文档，您可以使用表 53 中列出的兼容 ICS 的组件。

表 53. 通过 HTTP 从 V4.2.2 之前版本的 Interchange Server 接收文档所需的组件

组件	描述	注意事项和限制
WebSphere Business Integration Adapter for XML (Adapter for XML)	该适配器允许 InterChange Server 与应用程序交换业务对象，该应用程序以 HTTP 或消息格式接收数据。Adapter for XML 和 Business Integration Connect 之间通过 URL 地址进行通信。	Adapter for XML 是与 Business Integration Connect 一起提供的。您必须使用 V3.1.x 或更高版本的适配器。 注： 该适配器仅能与 4.2.2 版本的 WebSphere InterChange Server 一起使用。
HTTP 或 HTTPS 协议处理程序	这个协议处理程序与 Adapter for XML 一起工作，以将信息流发送到 URL 以及从 URL 接收信息流。	该协议处理程序是与 Business Integration Connect 一起提供的。要了解更多信息，请参阅第 77 页的『部署 HTTP 协议处理程序』。
有效负载数据处理程序	该数据处理程序可以使文档有效负载在它的文档格式（通常是 XML）和它的业务对象表示之间进行转换。	该数据处理程序是必需的，且必须支持 MIME 类型的有效负载文档。
附件数据处理程序	该数据处理程序使包含附件的文档在其文档格式及业务对象表示之间进行转换。	仅当您的文档包含附件时，才需要该数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

图 13 概述了 Business Integration Connect 如何通过 HTTP 传输协议从 V4.2.2 之前版本的 InterChange Server 接收文档。

注：所有有关 HTTP 协议处理程序的参考资料同样适用于 HTTPS 协议处理程序。

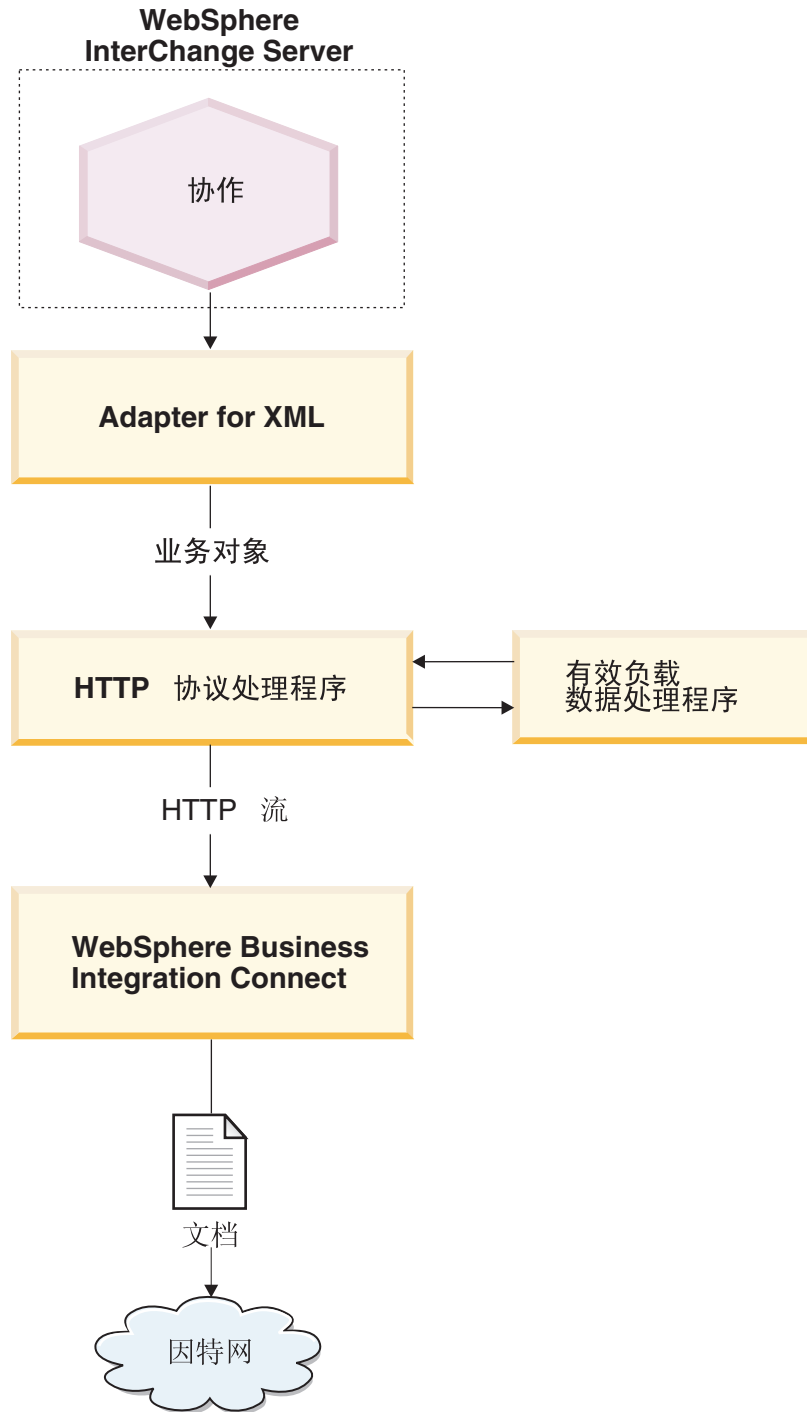


图 13. 通过 HTTP 传输协议从协作发送到 Business Integration Connect 的消息流

下列步骤描述了 Business Integration Connect 如何通过接收由 InterChange Server 内部的协作所发起的文档来参与请求处理。

1. InterChange Server 内的协作对 Adapter for XML 进行服务调用，给它发送包含请求和响应子对象的顶级业务对象。

请求子对象包含了特定于应用程序的信息，这些信息指向包含 Business Integration Connect 所期望的定制 HTTP 头的动态元对象。

2. Adapter for XML 调用 HTTP 协议处理程序。
3. HTTP 协议处理程序使用数据处理程序将协作发送的业务对象转换成 HTTP 流。

协议处理程序读取顶级业务对象的 MIME 类型和 URL，以确定要使用的数据处理程序和接收者地址。

4. HTTP 协议处理程序从顶级业务对象获得了第一个填充的业务对象。这是请求业务对象。

HTTP 协议处理程序调用数据处理程序，以将业务对象转换成 HTTP 流。

注：如果文档有附件，则请安装附件数据处理程序，然后配置 Adapter for XML，使之能调用该数据处理程序，以便将请求业务对象转换成带附件的文档。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

5. HTTP 协议处理程序根据请求业务对象确定动态元对象的名称。

HTTP 协议处理程序在请求业务对象特定于应用程序的信息中搜索 `cw_mo_conn` 标记，该标记标识与动态元对象相对应的属性。如果您对文档使用的是后端集成封装方式，则可以在这个动态元对象中指定定制 HTTP 头信息。

6. HTTP 协议处理程序在动态元对象中搜索 `HTTPProperties` 属性。

如果该属性已填充，则协议处理程序在请求消息中设置传输级的头。在 `HTTPProperties` 属性内，还可以指定 `content-type` 标准 HTTP 头。要了解更多信息，请参阅第 94 页的『为 V4.2.2 ICS 创建 HTTP 传输级头信息』。

7. HTTP 协议处理程序通过使用数据处理程序所返回的字符串来创建 HTTP 流。它还设置任何定制头信息，如动态元对象中所定义的那样。
8. HTTP 协议处理程序将产生的请求消息以流的形式发送至指定的 URL。

Business Integration Connect 侦听这个 URL，该 URL 已被配置成它的目标。

9. Business Integration Connect 用 HTTP 200 OK 响应。

如果 (Adapter for XML 的) `ReturnBusObjResponse` 连接器属性为 `true`，则调用是同步的。协议处理程序将响应消息转换成响应业务对象，并将它返回给 Adapter for XML。适配器在顶级业务对象中设置业务对象。然后将顶级业务对象返回给 InterChange Server 内的协作。

设置 4.2.2 之前版本的 ICS 的 HTTP 环境

因为接收来自 InterChange Server 的文档需要使用兼容 ICS 的组件，所以您必须执行表 54 中描述的设置和配置任务。有关如何配置 Business Integration Connect 以便通过 HTTP 与 V4.2.2 之前的 InterChange Server 进行通信的信息，请参阅第 37 页的『提供对出站文档的支持』。

表 54. 设置用于发送文档的环境

步骤	了解更多信息
1. 部署 HTTP 协议处理程序。	第 77 页的『部署 HTTP 协议处理程序』
2. 配置 WebSphere Business Integration Adapter for XML。	第 77 页的『配置 Adapter for XML』

注：如果文档中包含附件，则还必须如第 44 页的『处理带附件的文档』所描述的那样安装并配置附件数据处理程序。

部署 HTTP 协议处理程序： Business Integration Connect 提供了定制 HTTP 协议处理程序，以便将消息发送到 Business Integration Connect，并从 Business Integration Connection 接收消息。可以在 Business Integration Connect 安装介质的下列文件中找到这个 HTTP 协议处理程序：

Integration/WBI/WICS/WBICServlet/bcgwbiprotocol.jar

可以将这个定制协议处理程序插入 Adapter for XML V3.1.x 或更高版本。有关受支持 InterChange Server 的版本和平台，请参阅 *Adapter for XML 使用指南*，以了解您所使用的适配器的版本。

要将 HTTP 协议处理程序部署到 Adapter for XML，您必须让 Adapter for XML 知道 HTTP 协议处理程序的位置，以便它能在运行时装入该协议处理程序。要指定 HTTP 协议处理程序的位置，请遵循下面这些步骤：

1. 编辑 Adapter for XML 的启动脚本 start_xml.bat，该脚本位于安装了 WebSphere Business Integration Adapter 的产品目录的下列子目录中：
connectors/xml
2. 在这个启动脚本中，将定制 HTTP 协议处理程序的 JAR 文件 bcgwbiprotocol.jar 添加到 Adapter for XML 的 CLASSPATH 的 JAR 文件列表中。

配置 Adapter for XML： Adapter for XML 是兼容 ICS 的组件，它允许 Business Integration Connect 与 InterChange Server 以 HTTP 消息的格式交换文档。它支持与 InterChange Server 进行的请求处理交互，如下所示：

- 它从 InterChange Server 接收业务对象。
- 它使用 HTTP 协议处理程序将业务对象转换成 HTTP 流。
- 它将 HTTP 流发送至特定的 URL，Business Integration Connect 可以在这个 URL 中检索 HTTP 流。

注：未使用适配器的事件通知功能。要将 HTTP 消息从 Business Integration Connect 发送到 InterChange Server，请使用 WebSphere Business Integration Connect Servlet，如第 61 页的『通过 HTTP 将文档发送到 4.2.2 之前版本的 ICS』所描述的那样。

要点：WebSphere Business Integration Connect 不包括 WebSphere Business Integration Adapter for XML。必须单独获取该产品，并根据其 *Adapter for XML 用户指南* 中的指示说明来安装它。请参阅适配器文档，以确保适配器版本与正在使用的 InterChange Server 版本相兼容。

当您为与 InterChange Server 进行通信而已配置了 Adapter for XML 时，请遵循下面这些部分的步骤来配置该适配器，以便它接受来自 Business Integration Connect 的 HTTP 消息。

指定有效负载数据处理程序： 如第 75 页的图 13 所示，Adapter for XML 协议处理程序使用数据处理程序将它从 InterChange Server 收到的业务对象转换成相应的 HTTP 流。

注: Adapter for HTTP 调用的数据处理程序转换文档有效负载。如果您的文档是以 XML 传输包络（它包含附件或者“包络标志”为“是”）打包的，则将附件数据处理程序配置为有效负载数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

要指出使用哪种数据处理程序来转换有效负载，您必须采取第 41 页的『业务对象变换』中列出的步骤。另外，必须配置 Adapter for XML 以使用该有效负载数据处理程序。在连接器配置程序中，设置 DataHandlerConfigMO 连接器配置属性，以指定 Adapter for XML 用来标识数据处理程序的顶级数据处理程序元对象。请确保在用于该适配器的受支持的业务对象列表中包括顶级数据处理程序元对象的名称。

配置协议处理程序包名: Adapter for XML 使用 JavaProtocolHandlerPkgs 连接器配置属性来标识 Java 协议处理程序包的名称。为与 Business Integration Connect 集成，确保将 JavaProtocolHandlerPkgs 属性设置成 Business Integration Connect 提供的 HTTP 协议处理程序的包名。

`com.ibm.bcg.integration.wbi.utils.protocolhandlers`

为响应业务对象指定支持: Adapter for XML 使用 ReturnBusObjResponse 连接器配置属性来指出是否返回响应业务对象。仅当交互为同步时，才返回响应业务对象。缺省情况下，ReturnBusObjResponse 连接器配置属性被设置成“false”。要配置 Adapter for XML 以返回响应业务对象，将 ReturnBusObjResponse 连接器配置属性设置成“true”。

注: 如果希望 Business Integration Connect 允许社区管理员正在使用的封装和业务协议之间进行的同步交互，则请将 ReturnBusObjResponse 连接器配置属性设置成 true，并在您的顶级业务对象中提供响应业务对象。

要设置连接器配置属性，请使用作为 WebSphere Business Integration Adapter for XML 发行版的一部分的连接器配置程序工具。在连接器配置程序内，ReturnBusObjResponse 应该会出现现在连接器配置属性中特定于连接器的选项卡上。

创建用于接收文档的业务对象定义

WebSphere Business Integration Adapter for XML 以有效负载业务对象的格式接收来自 InterChange Server 的信息。对于 Adapter for XML，有效负载业务对象被表示成业务对象的层次结构。Adapter for XML 在接收到 Business Integration Connect 文档时创建这个业务对象层次结构。因此，您必须创建业务对象定义以表示这个层次结构。

因为 Adapter for XML 仅参与 InterChange Server 的请求处理，所以顶级业务对象的请求和响应属性将如表 55 所示的那样进行解释。

表 55. 请求处理中的请求和响应业务对象

属性	用途
请求业务对象	包含来自 InterChange Server 的请求信息；协议处理程序和数据处理程序转换该信息并将它发送到 Business Integration Connect 侦听的 URL。
响应业务对象	如果交互是同步的，则包含来自 Business Integration Connect 的响应信息。

有关如何创建该业务对象结构的更多信息，请参阅第 79 页的『为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象』。

为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象

Connect Servlet 以有效负载业务对象的格式将您的文档发送到 InterChange Server。Adapter for XML 以相同的格式接收来自 InterChange Server 的文档。当接收或发送 Business Integration Connect 文档时，这两个组件都调用有效负载数据处理程序来处理该业务对象，如下所述：

- 对于请求处理，有效负载数据处理程序将请求业务对象转换成相应的 HTTP 流。
- 对于事件通知，数据处理程序将 HTTP 流转换成事件业务对象。

因此，您必须创建表 56 所示的业务对象定义，以表示 Adapter for XML 期望的有效负载业务对象结构。

表 56. 用于 HTTP 传输协议的业务对象定义

条件	业务对象定义	了解更多信息
如果您正对您的文档使用“无”封装或后端集成封装且您的文档没有附件	有效负载业务对象的业务对象层次结构： <ul style="list-style-type: none">• 顶级业务对象• 请求业务对象• 响应业务对象（仅当期望有响应时才使用）	『通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建有效负载业务对象结构』
如果您对您的文档使用后端集成封装	在有效负载业务对象中添加业务对象以保存传输级头信息： <ul style="list-style-type: none">• 动态元对象• HTTP 属性业务对象	第 82 页的『为 4.2.2 之前版本的 InterChange Server 创建 HTTP 传输级头信息』
如果文档包含附件（后端集成封装是必需的）	还必须创建其它业务对象以表示附件。	第 54 页的『创建与附件关联的业务对象定义』

注：如果正在为 cXML 文档定义业务对象，请参阅第 82 页的『创建 cXML 业务对象』。

通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建有效负载业务对象结构

包装器数据处理程序（用于发送文档）以及 Adapter for XML 和 HTTP 协议处理程序（用于接收文档）都期望有效负载业务对象有相同业务对象结构。该业务对象结构由下列业务对象构成：

- 顶级业务对象
- 请求业务对象
- 响应业务对象（可选）

图 14 显示了一个有效负载业务对象定义的样本业务对象结构，其中的业务对象定义用于通过 HTTP 传输协议传递文档的 4.2.2 之前版本的 InterChange Server。

注：有关该业务对象结构的详细描述，请参阅 *Adapter for XML 用户指南*。

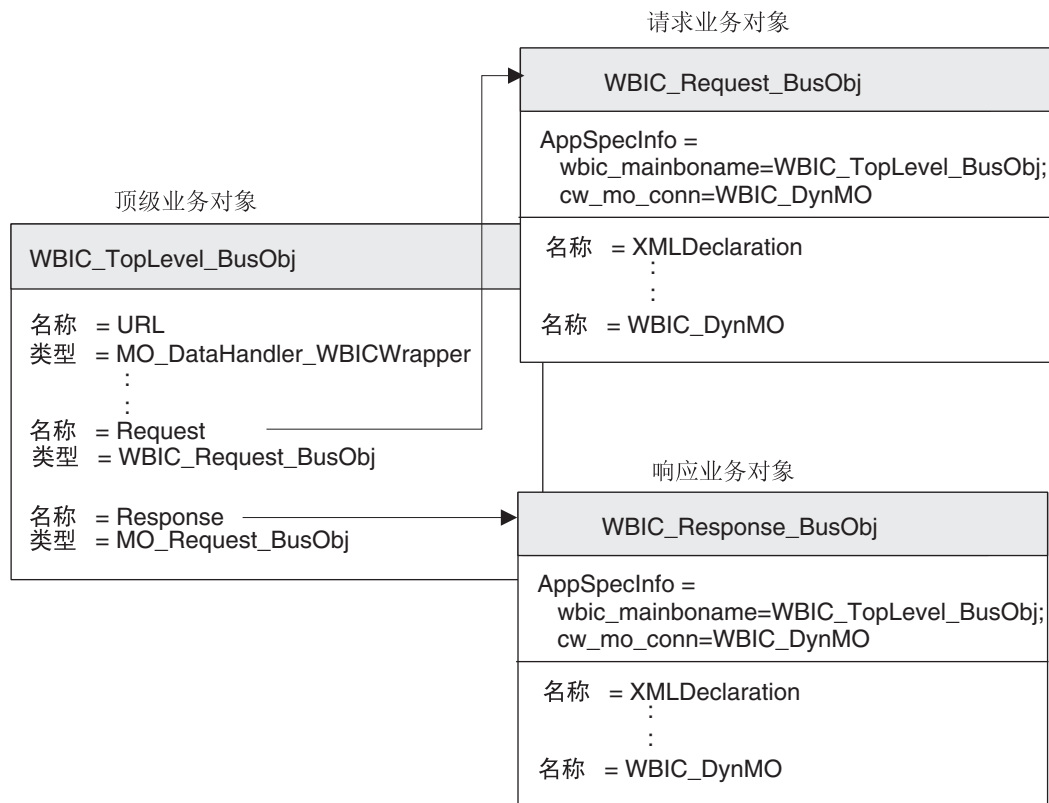


图 14. 用于 4.2.2 之前版本的 *InterChange Server* 的 HTTP 有效负载业务对象的业务对象结构

顶级业务对象: 顶级业务对象是请求和响应业务对象的包装器。您必须为该业务对象创建业务对象定义。表 57 概括了该顶级业务对象定义的属性。

表 57. 顶级业务对象的属性

属性	属性类型	描述
URL	字符串	业务对象中数据的目的地址。 请注意: 包装器数据处理程序不使用该属性。但是, Adapter for XML 使用了该属性。
MimeType	字符串	定义了传递给 URL 的数据的内容类型和格式。 请注意: 包装器数据处理程序不使用该属性。但是, Adapter for XML 使用了该属性。
BOPrefix	字符串	用于确定要调用哪个数据处理程序。 请注意: 包装器数据处理程序不使用该属性。
响应	业务对象	表示响应消息的子业务对象 (如果您期望获得响应)。该业务对象的用途取决于它是参与请求处理还是参与事件通知。要了解关于该业务对象结构的更多信息, 请参阅第 81 页的『响应业务对象』。
请求	业务对象	表示请求消息的子业务对象。该业务对象的用途取决于它是参与请求处理还是参与事件通知。要了解关于该业务对象结构的更多信息, 请参阅第 81 页的『请求业务对象』。

注：如果您正在使用附件数据处理程序处理附件，则您必须如第 54 页的『创建与附件关联的业务对象定义』中所描述的那样修改您的请求业务对象，以保存附件。

要了解有关顶级业务对象结构的完整描述，请参阅 *Adapter for XML 用户指南*。

请求业务对象：请求业务对象包含了要传递到 URL 的数据。它包含了请求消息中各种 XML 标记的属性。该请求业务对象的用途取决于它参与以下哪个 InterChange Server 任务：

- 对于事件通知（发送文档到 InterChange Server），请求业务对象包含来自 Business Integration Connect 的请求消息，它是要发送到 InterChange Server 的事件。

要了解更多信息，请参阅第 73 页的『创建用于发送文档的业务对象定义』。

- 对于请求处理（接收来自 InterChange Server 的文档），请求业务对象包含了 InterChange Server 打算发送到 Business Integration Connect 的请求。

要了解更多信息，请参阅第 78 页的『创建用于接收文档的业务对象定义』。

注：该业务对象结构将它的两个子业务对象标识成“请求业务对象”和“响应业务对象”。但是，请求处理和事件通知都使用该结构。

有关请求业务对象结构的基本描述，请参阅 *Adapter for XML 用户指南*。要和 Business Integration Connect 一起使用，您必须对请求业务对象定义的结构进行两项定制：

- 如果从 Business Integration Connect 发送到 InterChange Server 的文档使用后端集成封装，则您必须在请求业务对象定义中添加一个特殊属性以标识动态元对象。

该属性提供了有关消息头的配置信息。要了解更多信息，请参阅第 82 页的『为 4.2.2 之前版本的 InterChange Server 创建 HTTP 传输级头信息』。

- 对于请求业务对象定义的特定于业务对象级应用程序的信息，请添加表 58 中所示的标记。

表 58. 请求业务对象特定于应用程序的信息中标记

特定于应用程序的信息的标记	描述	是否必需？
wbic_mainboname	给顶级业务对象取名	是
cw_mo_conn	指定动态元对象，该元对象包含了 HTTP 传输级头字段。要了解更多信息，请参阅第 82 页的『为 4.2.2 之前版本的 InterChange Server 创建 HTTP 传输级头信息』。	否（仅当使用后端集成封装时才需要这个标记）

响应业务对象：响应业务对象包含了要从 URL 接收的数据。它包含了响应消息中各种 XML 标记的属性。该响应业务对象的用途取决于它参与以下哪个 InterChange Server 任务：

- 对于事件通知，响应业务对象包含来自协作的响应。要了解更多信息，请参阅第 73 页的『创建用于发送文档的业务对象定义』。
- 对于请求处理，响应业务对象包含了来自响应 InterChange Server 所发送请求的 URL 的信息。要了解更多信息，请参阅第 78 页的『创建用于接收文档的业务对象定义』。

无论该响应是事件通知的一部分还是请求处理的一部分，只有当 Business Integration Connect 和 InterChange Server 之间的交互是同步的，并且期望获得对请求的业务响应时，才发送响应业务对象。如果出现这种情况，那么您还必须采用下列这些额外的步骤：

- 在顶级业务对象中，将 `wbic_type` 标记添加到对应于响应业务属性的特定于属性级应用程序的信息中。

此标记有下列语法：

```
wbic_type=reply
```

- 确定是否添加 `wbic_response_mime` 特定于业务对象级应用程序的信息。这个特定于应用程序的信息是可选的。它为要用于响应业务对象的数据处理程序制定了 MIME 类型。

如果未指定这个标记，则包装器数据处理程序使用（顶级业务对象中的）`wbic_response_mime` 属性所指示的子代元对象来确定用于响应的数据处理程序。

注：响应业务对象未包含动态元对象的属性。

如果 Business Integration Connect 和 InterChange Server 之间的交互是异步的，则 Business Integration Connect 不需要响应，因此您无需创建响应业务对象。

创建 cXML 业务对象：对于 cXML 文档，可以使用 XML 对象发现代理（ODA）来创建业务对象。XML ODA 可以使用 cXML DTD。但是请注意，XML ODA 不支持 ENTITY。因此，在运行带 XML ODA 的 cXML DTD 之前，您需要从 DTD 中除去 ENTITY。

当使用 XML ODA 生成业务对象时，您可以选择 cXML 标记作为您的根元素。但是，这可能会生成一个大业务对象，捕获整个 cXML DTD。如果您想创建较小的业务对象，可以选择另一个标记作为根元素，这需要您为 Data Handler for XML 编写一个定制名称处理程序。数据处理程序将调用这个名称处理程序以进行顶级业务对象名称解析。请参阅 Data Handler for XML 文档以获取关于编写定制名称处理程序的信息。

为 4.2.2 之前版本的 InterChange Server 创建 HTTP 传输级头信息

如果您正通过 HTTP 传输协议发送以后端集成封装的文档，那么您的请求业务对象需要包含定制的传输级头信息。包装器数据处理程序和 Adapter for XML 都期望这个定制头信息位于动态元对象中。

图 15 显示了请求业务对象的业务对象结构，该请求业务对象中表示了通过 HTTP 传输协议传递的、以后端集成封装的 Business Integration Connect 文档。

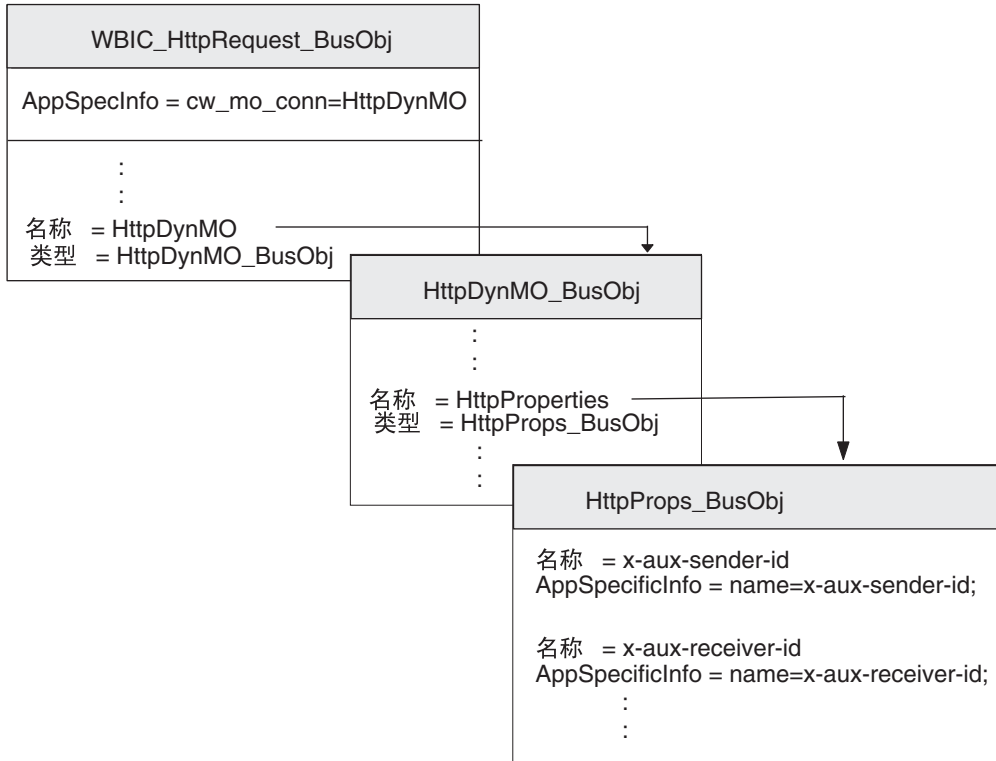


图 15. 请求业务对象与 HTTP 动态元对象的关系

请通过执行下列步骤来确保业务对象结构包含了动态元对象：

1. 创建业务对象定义以保存后端集成封装所需的 HTTP 属性。
2. 为动态元对象创建业务对象定义。
3. 修改请求业务对象的业务对象定义，以包含动态元对象的属性。

下面几节中逐一描述了这些步骤。

创建 HTTP 属性业务对象： HTTP 属性业务对象包含了后端集成封装所需的 HTTP 属性。它还可以包含 Content-Type 属性（该属性指定要在请求消息中设置的 content-type 头）和 content-length 属性（该属性指定消息长度，长度单位为字节）。第 10 页的表 4 描述了各个有效传输头字段。

要创建 HTTP 属性业务对象定义，请执行下列步骤：

1. 在业务对象定义中为每个传输头字段创建属性。

所有属性的属性类型都为“字符串”。您可以用 HTTP 属性的确切名称命名属性（如第 10 页的表 4 的头字段列中所列的那样）。

注： HTTP 属性名称中唯一例外是 content-type 字段，其属性应为 Content_Type。

2. 对于 HTTP 属性业务对象中的每个属性，添加特定于应用程序的信息，以标识相关属性的用途。

这个特定于属性级应用程序的信息具有以下格式：

`name=HTTPproperty`

其中 *HTTPproperty* 是第 10 页的表 4 的头字段列中的值之一。

在第 83 页的图 15 中，*HttpProps_BusObj* 业务对象定义包含了各种传输头字段的属性。这些属性都有特定于属性级应用程序的信息，以指定相关协议头的名称。例如，*x-aux-sender-id* 属性具有如下特定于应用程序的信息：

```
name=x-aux-sender-id
```

创建 HTTP 动态元对象： 动态元对象包含了一个子业务对象，该业务对象中包含了用于 HTTP 头信息的配置信息。请务必确保业务对象结构包含动态元对象。动态元对象的业务对象定义必须包含名为 *HttpProperties* 的属性，其属性类型是 HTTP 属性业务对象的业务对象定义（请参阅第 83 页的『创建 HTTP 属性业务对象』）。

例如，在第 83 页的图 15 中，*HttpDynMO_BusObj* 业务对象定义包含了 *HttpProperties* 属性，其属性类型为 *HttpProps_BusObj*。

修改请求业务对象定义： 请求业务对象定义表示从 Business Integration Connect 请求的信息。有关如何创建请求业务对象的信息，请参阅第 81 页的『请求业务对象』。为将动态元对象并入您的有效负载业务对象结构中，您必须对您的请求业务对象定义进行下列修改：

1. 将某个属性添加到请求业务对象定义中，以保存动态子元对象。

该属性的属性类型是动态元对象的业务对象定义（请参阅『创建 HTTP 动态元对象』）。

2. 将 *cw_mo_conn* 标记添加到请求业务对象定义特定于业务对象级应用程序的信息中，以标识包含动态元对象的属性。

cw_mo_conn 标记的格式如下：

```
cw_mo_conn=dynamicMetaObjAttr
```

其中 *dynamicMetaObjAttr* 是包含动态元对象的请求业务对象中的属性名。

例如，在第 83 页的图 15 中，名为 *HttpDynMO* 的属性被添加到请求业务对象定义 *WBIC_HttpRequest_BusObj* 中。该属性包含了动态元对象，该对象是类型 *HttpDynMO_BusObj* 的子业务对象。另外，已经对请求业务对象的特定于应用程序的信息进行了修改，使之包含下列 *cw_mo_conn* 标记，以标识这个动态元对象：

```
cw_mo_conn=HttpDynMO
```

创建用于 HTTP 的 4.2.2 之前版本的 ICS 构件

要配置 4.2.2 之前版本的 InterChange Server 以通过 HTTP 传输协议与 Business Integration Connect 进行通信，您必须创建如表 59 所示的 InterChange Server 构件。

表 59. 用于通过 HTTP 传输协议与 4.2.2 之前版本的 ICS 进行通信的构件

ICS 构件	目的	了解更多信息
业务对象定义	表示文档	第 79 页的『为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象』
连接器对象（仅需要请求处理）	在运行时表示 Adapter for XML	第 85 页的『创建 XML 连接器对象』

表 59. 用于通过 HTTP 传输协议与 4.2.2 之前版本的 ICS 进行通信的构件 (续)

ICS 构件	目的	了解更多信息
协作模板和协作对象	表示 InterChange Server 用于处理文档的业务流程	『绑定协作以与 Adapter for XML 通信』

创建 XML 连接器对象

为了支持通过 HTTP 传输协议利用 4.2.2 之前版本的 InterChange Server 进行请求处理，您可以使用 Adapter for XML 将文档发送到 InterChange Server。要在运行时获取 Adapter for XML 的实例，请在“系统管理器”中采用下列步骤：

1. 创建连接器对象：

- 创建连接器对象以表示 Adapter for XML 实例。

注：在连接器配置程序的“业务对象”选项卡中，确保您指定了为和 Adapter for XML 一起使用而创建的所有业务对象定义。要了解关于这些业务对象定义的描述，请参阅第 79 页的『为通过 HTTP 传递文档的 4.2.2 之前版本的 ICS 创建业务对象』。

- 如果协作需要连接器对象，则为“端口连接器”创建一个连接器对象。

2. 配置连接器对象。

有关如何配置 Adapter for XML 以和 Business Integration Connect 一起使用的信息，请参阅第 77 页的『配置 Adapter for XML』。

绑定协作以与 Adapter for XML 通信

如第 43 页的『=创建协作』所描述的，要使 InterChange Server 知道从哪接收以及向哪发送业务对象，则协作对象在运行时必须存在。当您为将信息发送到 Business Integration Connect 的协作创建协作对象时，帮定其端口。对于请求处理，设置“目标”协作端口，该端口使用 Adapter for XML 将请求发送到 Business Integration Connect、您为 Adapter for XMP 创建的连接器对象；也就是说，Adapter for XMP 是目的地适配器。

将 HTTP 传输协议与 V4.2.2 ICS 一起使用

WebSphere Business Integration Connect 可以通过 HTTP 传输协议向 V4.2.2 WebSphere InterChange Server (ICS) 发送文档和从其接收文档。

注：

1. 有关通过 HTTP 传输协议在 WebSphere Business Integration Connect 和 4.2.2 之前版本的 WebSphere InterChange Server 之间发送和接收文档的信息，请参阅第 61 页的『将 HTTP 传输协议用于 4.2.2 之前版本的 ICS』。
2. 如果您正通过 HTTP 传输协议交换 SOAP 文档，则请参阅第 97 页的『通过 HTTP/S 发送 SOAP 文档』。

本节提供了有关如何配置 V4.2.2 InterChange Server 和相关兼容 ICS 的组件以与通过 HTTP 传输文档的 Business Integration Connect 一起使用的下列信息：

- 第 86 页的『通过 HTTP 传输协议将文档传递到 V4.2.2 ICS 所需的组件』
- 第 88 页的『设置用于 V4.2.2 ICS 的 HTTP 传输环境』
- 第 91 页的『为通过 HTTP 收发文档的 V4.2.2 ICS 创建业务对象定义』

- 第 97 页的『创建用于 HTTP 的 V4.2.2 ICS 构件』

通过 HTTP 传输协议将文档传递到 V4.2.2 ICS 所需的组件

要使 Business Integration Connect 能使用 HTTP 传输协议与 V4.2.2 InterChange Server 进行通信，需要配置这两个组件。表 60 中概括了这些配置步骤。

表 60. 配置 Business Integration Connect 和 InterChange Server

组件	版本	了解更多信息
WebSphere Business Integration Connect	4.2.2	第 37 页的『配置通过 HTTP 传输协议传递的出站文档』 第 39 页的『配置通过 HTTP 传输协议传递的入站文档』
WebSphere InterChange Server	4.2.2	第 97 页的『创建用于 HTTP 的 V4.2.2 ICS 构件』

另外，为了使用 HTTP 传输协议在 Business Integration Connect 和 V4.2.2 InterChange Server 之间收发文档，您可以使用表 61 中列出的兼容 ICS 的组件。

表 61. 通过 HTTP 使用 V4.2.2 InterChange Server 传送文档所需的组件

组件	描述	注意事项和限制
WebSphere Business Integration Adapter for HTTP (Adapter for HTTP) 有效负载数据处理程序	该适配器允许 InterChange Server 与以 HTTP 流格式发送或接收数据的应用程序交换业务对象。 该数据处理程序使文档有效负载在其文档格式（通常为 XML）和业务对象描述之间进行转换。	该适配器不能用于 4.2.2 之前版本的 WebSphere InterChange Server。 该数据处理程序是必需的，并且它必须支持 MIME 类型的有效负载文档。
附件数据处理程序	该数据处理程序处理文档消息的附件文档。	仅当文档包括附件时，才需要该数据处理程序。

下面几节描述了表 61 中的组件是如何协作以通过 HTTP 传输协议在 Business Integration Connect 和 V4.2.2 InterChange Server 之间发送和接收文档。

通过 HTTP 将文档发送到 V4.2.2 ICS

要使 Business Integration Connect 能使用 HTTP 传输协议将文档发送到 V4.2.2 InterChange Server，可以使用 Adapter for HTTP 检索 Business Integration Connect 以 HTTP 流发送的文档。该适配器然后将文档路由到 InterChange Server。图 16 概述了 Business Integration Connect 如何通过 HTTP 传输协议将文档发送到 V4.2.2 InterChange Server。

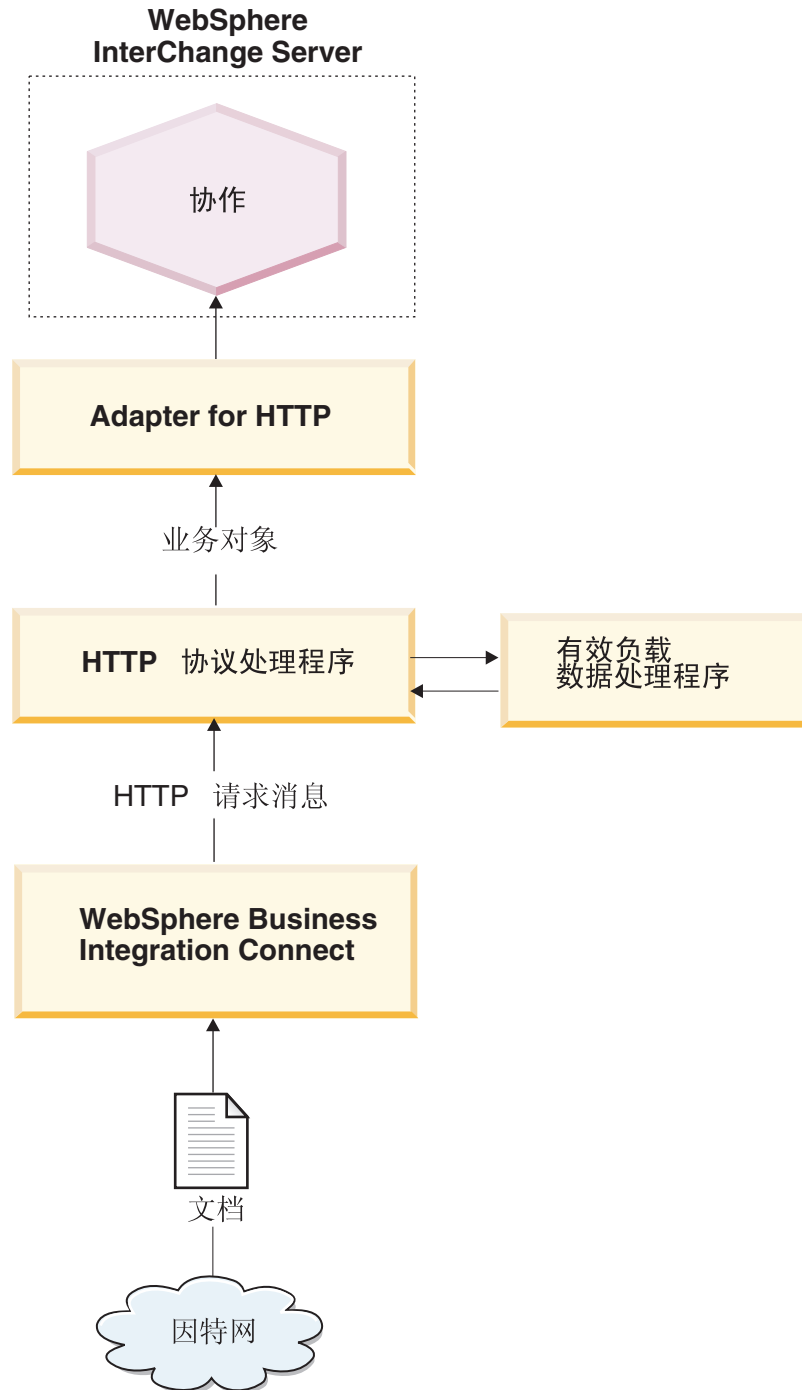


图 16. 通过 HTTP 传输协议从 Business Integration Connect 到协作的消息流

通过 HTTP 接收来自 V4.2.2 ICS 的文档

要使 Business Integration Connect 使用 HTTP 传输协议接收来自 V4.2.2 InterChange Server 的文档。使用 Adapter for HTTP，它以 HTTP 流的形式发送和接收自 InterChange Server 的消息，以供 Business Integration Connect 检索。图 17 概述了 Business Integration Connect 如何使用 HTTP 传输协议接收来自 V4.2.2 InterChange Server 的文档。

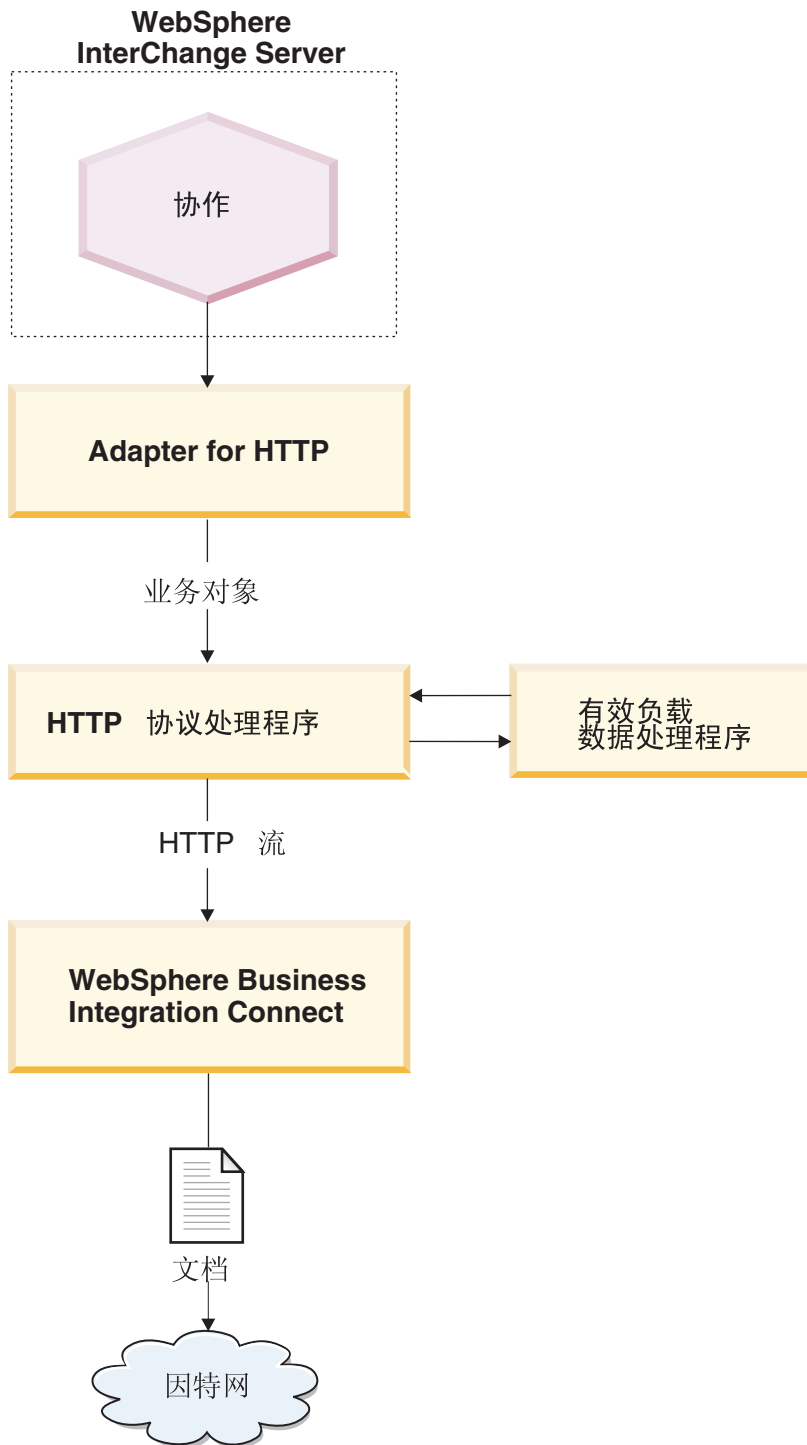


图 17. 通过 HTTP 传输协议从协作到 Business Integration Connect 的消息流

设置用于 V4.2.2 ICS 的 HTTP 传输环境

因为无论是将文档发送到 InterChange Server 还是接收来自 InterChange Server 的文档都要使用兼容 ICS 的组件，所以您必须对 Adapter for HTTP 执行设置和配置任务。有

关于如何配置 Business Integration Connect 以与通过 HTTP 传递文档的 InterChange Server 一起使用的信息，请参阅第 37 页的『为 InterChange Server 配置 Business Integration Connect』。

Adapter for HTTP 是兼容 ICS 的组件，该组件允许 Business Integration Connect 与 InterChange Server 以 HTTP 消息的格式进行文档交换。它支持与 InterChange Server 进行下列交互：

- 对于请求处理，它接收来自 InterChange Server 的请求业务对象，将其转换成 HTTP 流并将其发送到指定的 URL，在那由 Business Integration Connect 接收。
- 对于事件通知，它侦听指定的 URL，Business Integration Connect 将文档发往该 URL。当它接收到文档时，它将该文档转换成事件对象（使用数据处理程序）并将该对象发送到 InterChange Server。

请注意： WebSphere Business Integration Connect 不包括 WebSphere Business Integration Adapter for HTTP。您必须单独获取该产品并按 *Adapter for HTTP 用户指南* 中的指示信息安装它。请参阅适配器文档，以确保适配器版本与您正在使用的 InterChange Server 版本相兼容。

当您已配置好用以与 InterChange Server 通信的 Adapter for HTTP 时，请遵循下面几节中的步骤来配置该适配器，以侦听来自 Business Integration Connect 的 HTTP 消息：

指定有效负载数据处理程序

如第 88 页的图 17 所示，Adapter for HTTP 使用数据处理程序将接收自 InterChange Server 的业务对象转换成相应的 HTTP 流。

注： Adapter for HTTP 调用的数据处理程序转换文档有效负载。如果您的文档使用 XML 传输包络（它包含附件或者“包络标志”为“是”），则请将附件数据处理程序配置为有效负载数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

要指出使用哪个数据处理程序来转换有效负载，您必须采取第 41 页的『业务对象变换』中列出的步骤。另外，必须配置 Adapter for HTTP 以使用该有效负载数据处理程序。您可以以下列任一方式设置有效负载数据处理程序：

- 在连接器配置程序中，设置 `DataHandlerMetaObjectName` 连接器配置属性以指定顶级数据处理程序元对象的名称，Adapter for HTTP 使用该元对象名称来标识数据处理程序。确保将顶级数据处理程序元对象包含在受支持的适配器业务对象列表中。
- 在顶级业务对象中，使用 `MimeType` 属性来保存用于标识有效负载数据处理程序的 MIME 类型。有关该业务对象的更多信息，请参阅第 92 页的『顶级业务对象』。

配置协议处理程序包名

Adapter for HTTP 使用 `JavaProtocolHandlerPackages` 连接器配置属性来标识 Java 协议处理程序包的名称。为与 Business Integration Connect 集成，确保将 `JavaProtocolHandlerPackage` 的属性设置成其缺省值：

```
com.ibm.net.ssl.internal.www.protocol
```

配置 HTTP 协议侦听器

Adapter for HTTP 支持层次结构性的配置属性以获取配置其协议侦听器所需的信息。顶级配置属性称为 `ProtocolListenerFramework`。该顶级属性内存在多个级别的子属性。要配置这里的协议处理程序以和 Adapter for HTTP 一起使用，确保按照以下步骤配置 `ProtocolListener` 中的属性：

1. 在下列配置属性下配置带子属性的协议侦听器：

```
ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
```

要配置协议侦听器，请设置表 62 中列出的子属性。

表 62. 配置协议侦听器

属性	描述	值
协议	协议侦听器类型： <ul style="list-style-type: none">• HTTP• HTTPS	http 或 https
主机	协议侦听器侦听的 IP 地址	WebSphere Business Integration Connect 在其上运行的本地机器的 IP 地址
端口	协议侦听器侦听请求的端口	8080

2. 通过配置下列配置属性的子属性，配置协议侦听器支持的 URL：

```
ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
      URLsConfiguration
        URL1
```

将 `ContextPath` 属性设置成协议侦听器接收到的 HTTP 请求的 URI。

注：该目录必须与 Business Integration Connect 网关指定作为其“目标 URI”的目录相一致。要了解更多信息，请参阅第 37 页的『配置通过 HTTP 传输协议传递的出站文档』。

3. 如果您的文档包含附件，您必须配置协议侦听器转换，通过设置下列配置属性的子属性来完成上述配置：

```
ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
      URLsConfiguration
        URL1
          TransformationRules
            TransformationRule1
```

要配置您协议侦听器的附件转换，请设置表 63 列出的子属性。对于您所使用的每个附件数据处理程序的实例，都需要一个转换规则。有关附件数据处理程序的更多信息，请参阅第 44 页的『处理带附件的文档』。

表 63. 配置协议侦听器的附件转换

属性	描述	值
ContentType	要使用数据处理程序转换的数据的内容类型	附件数据相关的内容类型
MimeType	用来标识调用的数据处理程序的 MIME 类型	与附件数据处理程序实例相关的 MIME 类型
Charset	转换指定内容类型的数据时使用的字符集	附件数据字符集

有关这些属性的更多信息，请参阅 *Adapter for HTTP 用户指南*。

为通过 HTTP 收发文档的 V4.2.2 ICS 创建业务对象定义

Adapter for HTTP 以有效负载业务对象的形式将文档发送到 InterChange Server 以及从其接收文档。当收到或发送 Business Integration Connect 文档时，Adapter for HTTP 调用有效负载数据处理程序来处理该业务对象，如下所述：

- 对于请求处理，有效负载数据处理程序将请求业务对象转换成其相应的 HTTP 流。
- 对于事件通知，数据处理程序将 HTTP 流转换成事件业务对象。

因此，您必须创建如表 64 所示的业务对象定义，以表示 Adapter for HTTP 所期望的有效负载业务对象结构。

表 64. Adapter for HTTP 的业务对象定义

条件	业务对象定义	了解更多信息
如果您的消息使用且您的文档没有附件	有效负载业务对象： <ul style="list-style-type: none"> • 顶级业务对象 • 请求业务对象 • 响应业务对象（可选） • 故障业务对象（可选） 	『为通过 HTTP 传递文档的 V4.2.2 ICS 创建有效负载业务对象结构』
如果您对您的消息使用后端集成封装	将业务对象添加到有效负载业务对象以保存消息头信息： <ul style="list-style-type: none"> • 动态元对象 • HTTP 属性业务对象 	第 94 页的『为 V4.2.2 ICS 创建 HTTP 传输级头信息』。
如果文档包含附件	还必须创建其它业务对象来表示这些附件。	第 54 页的『创建与附件关联的业务对象定义』

为通过 HTTP 传递文档的 V4.2.2 ICS 创建有效负载业务对象结构

Adapter for HTTP 期望有效负载业务对象由以下业务对象组成：

- 顶级业务对象
- 请求业务对象
- 故障业务对象（可选）
- 响应业务对象（可选）

图 18 显示了用于 V4.2.2 InterChange Server（通过 HTTP 传输文档）的有效负载业务对象定义的样本业务对象结构。

注：有关该业务对象结构的详细描述，请参阅 *Adapter for HTTP* 用户指南。

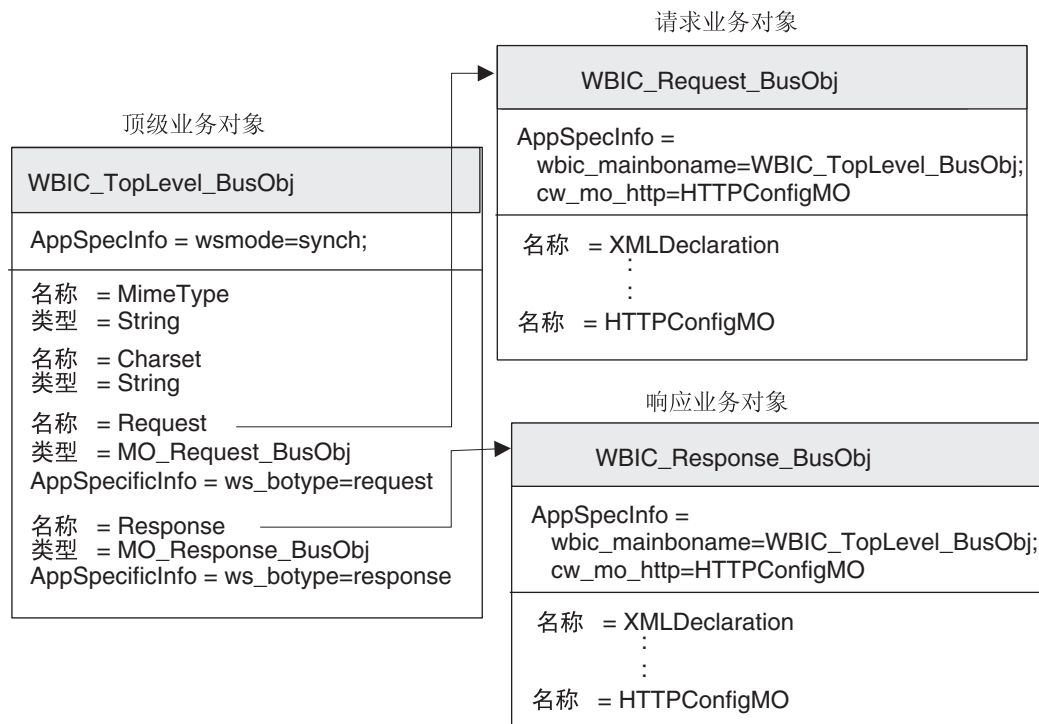


图 18. V4.2.2 ICS 的 HTTP 有效负载业务对象的业务对象结构

顶级业务对象: 顶级业务对象是请求和响应业务对象的包装器。您必须为该业务对象创建业务对象定义。表 57 汇总了顶级业务对象定义的属性。

表 65. 顶级业务对象的属性

属性	属性类型	描述
MimeType	字符串	定义了被传递给 URL 的数据的内容类型和格式。
Charset	字符串	用于确定要调用哪个数据处理程序。
请求	业务对象	表示请求消息的子业务对象。该业务对象的用途取决于它是参与请求处理还是参与事件通知。要了解关于该业务对象结构的更多信息，请参阅第 81 页的『请求业务对象』。
响应	业务对象	表示请求消息的子业务对象（如果您期望获得响应）。该业务对象的用途取决于它是参与请求处理还是参与事件通知。要了解关于该业务对象结构的更多信息，请参阅第 94 页的『响应业务对象』。

注：当将 Adapter for HTTP 和 Business Integration Connect 一起使用时，在您的顶级业务对象中不需要包括故障业务对象。

表 66 汇总了顶级业务对象定义可以拥有的特定于应用程序的信息。

表 66. 顶级业务对象定义的特定于应用程序的信息

特定于应用程序的信息	标记	描述
业务对象级别	ws_mode	定义交互是同步的还是异步的
属性级别	ws_botype	定义哪个属性包含请求业务对象或响应业务对象

要了解有关顶级业务对象结构及其特定于应用程序信息的完整描述，请参阅 *Adapter for HTTP 用户指南*。

请求业务对象： 请求业务对象包含了要传递到 URL 的数据。它表示了 HTTP 请求消息。该请求业务对象的用途取决于它参与以下哪个 InterChange Server 任务：

- 对于事件通知（发送文档到 InterChange Server），请求业务对象包含来自 Business Integration Connect 的请求消息，它是被发送到 InterChange Server 的事件。
- 对于请求处理（接收来自 InterChange Server 的文档），请求业务对象包含了 InterChange Server 打算传递给 Business Integration Connect 的请求。

注： 顶级业务对象将其两个子业务对象标识成“请求业务对象”和“响应业务对象”。但是，请求处理和事件通知都使用该结构。

有关请求业务对象结构的基本描述，请参阅 *Adapter for HTTP 用户指南*。要和 Business Integration Connect 一起使用，必须对请求业务对象定义的结构进行两个定制操作：

- 如果 Business Integration Connect 发送到 InterChange Server 的文档使用了后端集成封装，则您必须在请求业务对象定义中添加一个特殊的属性以标识 HTTP 协议配置元对象。

该属性提供了有关消息的传输级头的配置信息。要了解更多信息，请参阅第 94 页的『为 V4.2.2 ICS 创建 HTTP 传输级头信息』。

- 对于请求业务对象定义的特定于业务对象级应用程序的信息，请添加表 67 中所示的标记。

表 67. 请求业务对象特定于应用程序的信息中的标记

特定于应用程序的信息的标记	描述	是否必需？
ws_tloname	给顶级业务对象取名	只有当业务对象定义参与事件通知时该标记才是必需
cw_mo_http	指定 HTTP 协议配置元对象，该元对象包含了 HTTP 传输级头字段。要了解更多信息，请参阅第 94 页的『为 V4.2.2 ICS 创建 HTTP 传输级头信息』。	只有当您使用后端集成封装时该标记才是必需的

注： 如果您正使用附件数据处理程序来处理打包在 XML 传输包络中的文档，则您必须如第 54 页的『创建与附件关联的业务对象定义』中所描述的那样修改您的请求业务对象以保存附件。

响应业务对象: 响应业务对象包含了要从 URL 接收的数据。它包含了响应消息中各种 XML 标记的属性。该响应业务对象的用途取决于它参与以下哪个 InterChange Server 任务:

- 对于事件通知, 响应业务对象包含响应消息, 该消息是由 InterChange Server 中的协作发送的。
- 对于请求处理, 响应业务对象包含来自 Business Integration Connect 的响应 InterChange Server 所发送请求的信息。

无论该响应是事件通知的一部分还是请求处理的一部分, 只有当 Business Integration Connect 和 InterChange Server 之间的交互是同步的且期望对您的请求做出业务响应时, 才发送响应业务对象。

有关故障业务对象结构的基本描述, 请参阅 *Adapter for HTTP 用户指南*。要和 Business Integration Connect 一起使用, 您必须对请求业务对象定义的结构进行一些定制:

- 如果 Business Integration Connect 发送到 InterChange Server 的文档使用了后端集成封装, 您必须在响应业务对象定义中添加一个特殊的属性, 以标识 HTTP 协议配置元对象。

该属性提供了有关消息的传输级头的配置信息。要了解更多信息, 请参阅『为 V4.2.2 ICS 创建 HTTP 传输级头信息』。

- 对于业务对象定义的特定于业务对象级应用程序的信息, 如第 93 页的表 67 所示的标记。
- 在顶级业务对象中, 将 `ws_botype` 标记添加到对应于响应业务对象的属性的特定于属性级应用程序的信息中。

此标记有下列语法:

```
ws_botype=response
```

- 确定是否添加 `wbic_response_mime` 特定于业务对象级应用程序的信息。这个特定于应用程序的信息是可选的。它为要用于响应业务对象的数据处理程序指定了 MIME 类型。

如果未指定这个标记, 则包装器数据处理程序使用 (顶级业务对象中的) `wbic_response_mime` 属性所指示的子代元对象来确定用于响应的数据处理程序。

如果 Business Integration Connect 与 InterChange Server 之间的交互是异步的, 则 Business Integration Connect 不需要响应, 因此您无需创建响应业务对象。

为 V4.2.2 ICS 创建 HTTP 传输级头信息

如果您正通过 HTTP 传输协议发送用后端集成封装的文档, 那么您的请求业务对象需要包含定制的传输级头信息。Adapter for HTTP 希望这个定制的头信息位于**动态元对象**中。

图 19 显示了请求业务对象的业务对象结构, 该请求业务对象表示通过 HTTP 传输协议传递的以后端集成封装的 Business Integration Connect 文档。

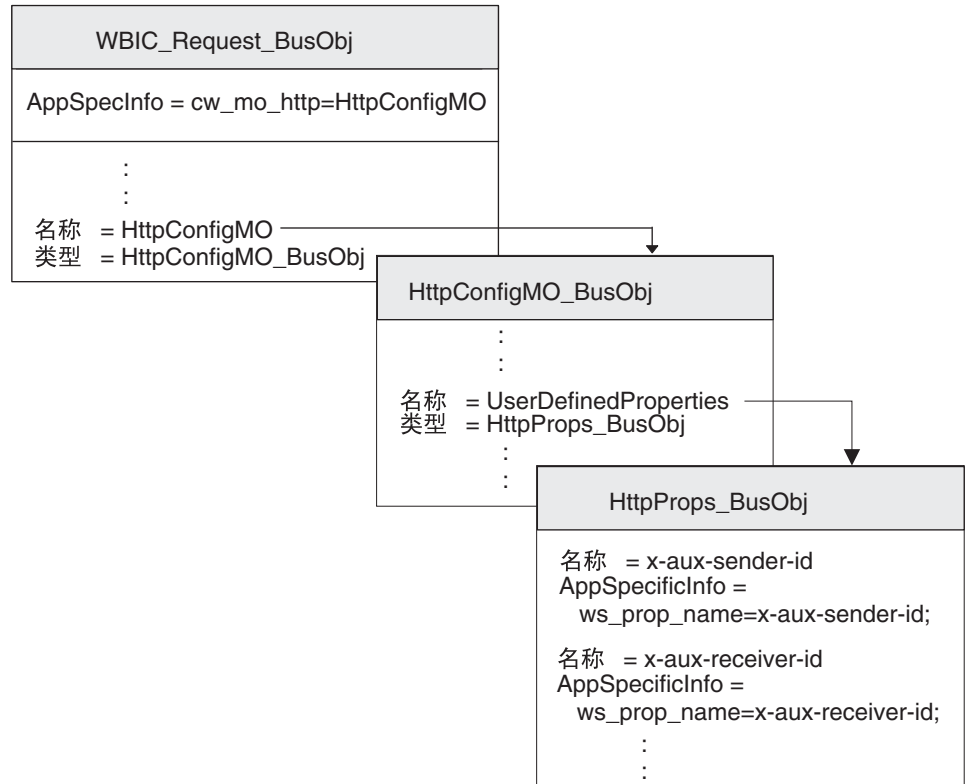


图 19. 请求业务对象与 HTTP 协议配置元对象的关系

请通过执行下列步骤来确保业务对象结构中包含 HTTP 协议配置元对象：

1. 创建业务对象定义以保存后端集成封装所需的 HTTP 属性。
2. 为 HTTP 协议配置元对象创建业务对象定义。
3. 修改请求业务对象的业务对象定义，以包含 HTTP 协议配置元对象的属性。

下面几节中逐一描述了这些步骤。

创建用户定义属性的业务对象： Adapter for HTTP 支持用户定义属性的业务对象，以便在 HTTP 协议配置元对象中保存定制属性。Business Integration Connect 使用该业务对象来保存后端集成封装所需的 HTTP 属性。它还可以包含 Content-Type 属性（该属性指定了要在请求消息中设置的 content-type 头）和 content-length 属性（该属性指定了消息长度，长度单位为字节）。第 10 页的表 4 描述了各个有效的传输头字段。

要创建 HTTP 头字段的用户定义属性的业务对象定义，请遵循以下步骤：

1. 在业务对象定义中为每个传输头字段创建属性。

所有属性的属性类型都应当为“字符串”。您可以用 HTTP 属性的确切名称命名属性（如第 10 页的表 4 的头字段列中所列的那样）。

2. 对于 HTTP 属性业务对象中的每个属性，添加特定于应用程序的信息，以标识相关属性的用途。

这个特定于属性级应用程序的信息具有以下格式：

`ws_prop_name=HTTPproperty`

其中 *HTTPproperty* 是第 10 页的表 4 的头字段列中的值之一。

在第 95 页的图 19 中，*HttpProps_BusObj* 业务对象定义包含了各种传输头字段的属性。这些属性都有特定于属性级应用程序的信息，用于指定相关协议头的名称。例如，*x-aux-sender-id* 属性具有如下所示的特定于应用程序的信息：

```
ws_prop_name=x-aux-sender-id
```

创建 HTTP 协议配置元对象： 对于事件通知，请求、响应或故障业务对象可能包含名为 **HTTP 协议配置元对象** 的动态元对象，该元对象用来保存配置信息（如头信息）。

有关 HTTP 协议配置业务对象结构的信息，请参阅 *Adapter for HTTP 用户指南*。要和 Business Integration Connect 一起使用，您必须对 HTTP 协议配置业务对象定义的结构进行以下定制：

1. 在业务对象定义内为任何需要的字段创建属性。

所有属性的属性类型都应当为“字符串”。

注：有关 HTTP 协议配置元对象中属性的完整列表，请参阅 *Adapter for HTTP 用户指南*。

2. 将 *UserDefinedProperties* 属性添加到该业务对象定义中。

该属性的属性类型是用户定义属性业务对象的业务对象定义（参阅第 95 页的『创建用户定义属性的业务对象』）。

例如，在第 95 页的图 19 中，*HttpConfigM0_BusObj* 业务对象定义包含 *UserDefinedProperties* 属性，其属性类型是 *HttpProps_BusObj*。

修改请求业务对象定义： 请求业务对象定义表示了 Business Integration Connect 所请求的信息。有关如何创建请求业务对象的信息，请参阅第 93 页的『请求业务对象』。为将动态元对象并入您的有效负载业务对象结构，您必须对您的请求业务对象定义进行下列修改：

1. 在您的请求业务对象定义中添加属性以保存 HTTP 协议配置元对象。

该属性的属性类型是 HTTP 协议配置元对象的业务对象定义（参阅『创建 HTTP 协议配置元对象』）。

2. 在您的请求业务对象定义的特定于业务对象级应用程序的信息中添加 *cw_mo_http* 标记，以标识包含 HTTP 协议配置元对象的属性。

cw_mo_http 标记的格式如下：

```
cw_mo_http=HttpConfigMetaObjAttr
```

其中 *HttpConfigMetaObjAttr* 是保存 HTTP 协议配置元对象的请求业务对象中的属性名。

例如，在第 95 页的图 19 中，名为 *HttpConfigM0* 的属性被添加到请求业务对象定义 *WBIC_HttpRequest_BusObj* 中。该属性包含动态元对象，该对象是类型为 *HttpConfigM0_BusObj* 对象的子业务对象。另外，已修改请求业务对象的特定于应用程序的信息，使其包含以下 *cw_mo_http* 标记，以标识该动态元对象：

```
cw_mo_http=HttpConfigM0
```

创建用于 HTTP 的 V4.2.2 ICS 构件

为配置 V4.2.2 InterChange Server 以通过 HTTP 传输协议与 Business Integration Connect 进行通信，您必须创建如表 68 所示的 InterChange Server 构件。

表 68. 通过 HTTP 传输协议 V4.2.2 ICS 进行通信的构件

ICS 构件	目的	了解更多信息
业务对象定义	表示文档	第 91 页的『为通过 HTTP 收发文档的 V4.2.2 ICS 创建业务对象定义』
连接器对象	在运行时表示 Adapter for HTTP	『创建 HTTP 连接器对象』
协作模板和协作对象	表示 InterChange Server 用于处理文档的业务流程	『绑定协作以与 Adapter for HTTP 进行通信』

创建 HTTP 连接器对象

要在运行时获取 Adapter for HTTP 的实例，您必须在“系统管理器”中采取以下步骤：

1. 创建连接器对象：

- 创建用于表示 Adapter for HTTP 实例的连接器对象。

注：在“连接器配置程序”的“受支持业务对象”选项卡中，请确保指定所有业务对象定义，这些定义是您为与 Adapter for HTTP 一起使用而创建的。要了解关于这些业务对象定义的描述，请参阅第 91 页的『为通过 HTTP 收发文档的 V4.2.2 ICS 创建业务对象定义』。

- 如果协作需要连接器对象，则为“端口连接器”创建一个连接器对象。

2. 配置连接器对象

有关如何配置 Adapter for HTTP 连接器对象以便与 Business Integration Connect 一起使用的信息，请参阅第 88 页的『设置用于 V4.2.2 ICS 的 HTTP 传输环境』。

绑定协作以与 Adapter for HTTP 进行通信

如第 43 页的『=创建协作』中描述的，要使 InterChange Server 从何处接收业务对象以及向何处发送业务对象，协作对象在运行时必须存在。当您为使用 adapter for HTTP 将信息发送 Business Integration Connect 以及从其接收信息的协作创建协作对象时，可以绑定协作端口，如下所示：

- 对于请求处理：应当将“发送”端口（该端口接收来自 Business Integration Connect 的事件）设置成为 Adapter fo HTTP 所创建的连接器对象；也就是说 Adapter for HTTP 是目的地适配器。
- 对于事件通知：应当将“接收”端口（该端口接收来自 Business Integration Connect 的事件）设置成为 Adapter for HTTP 所创建的连接器对象；也就是说 Adapter for HTTP 是源适配器。

通过 HTTP/S 发送 SOAP 文档

SOAP 文档不同于通过 HTTP/S 交换的其它类型文档。前者使用标准的 Adapter for Web Service，该适配器调用 SOAP 数据处理程序来将 SOAP 消息转换成业务对象，以及将业务对象转换成 SOAP 消息。本节描述了如何通过 HTTP 传输协议在 WebSphere Business Integration Connect 和 WebSphere InterChange Server 之间收发文档。

注:

1. 有关在 WebSphere Business Integration Connect 和 4.2.2 之前版本的 WebSphere InterChange Server 之间通过 HTTP 传输协议收发非 SOAP 文档的信息, 请参阅第 61 页的『将 HTTP 传输协议用于 4.2.2 之前版本的 ICS』。
2. 有关在 WebSphere Business Integration Connect 和 V4.2.2 WebSphere InterChange Server 之间通过 HTTP 传输协议收发非 SOAP 文档的信息, 请参阅第 85 页的『将 HTTP 传输协议与 V4.2.2 ICS 一起使用』。

请参阅 Adapter for Web Service 文档, 以获取有关业务对象结构和 WSDL 对象发现代理 (ODA) 的信息, ODA 是一种设计时工具, 您可以用它来生成包括有关目标 Web Service 信息的 SOAP 业务对象。

如“管理员指南”所描述的, 您必须已设置一个目标以接收来自后端系统的 Web Service 调用 (Web Service 目标), 还需设置一个目标以接收来自社区参与者的 Web Service 调用 (外部 Web Service 目标)。

发送和接收所需的组件

要使用 HTTP 传输协议将 SOAP 文档从 Business Integration Connect 发送到 InterChange Server, 您可以使用表 69 中列出的组件。所有这些组件都是作为 Business Integration Connect 发行版的一部分提供的。

表 69. 通过 HTTP 将 SOAP 文档发送到 InterChange Server 所需的组件

组件	描述	注意事项和限制
WebSphere Business Integration Adapter for Web Services	这个适配器允许 InterChange Server 与以 HTTP 消息形式发送或接收数据的应用程序交换业务对象。	<ol style="list-style-type: none">1. 该适配器不能用于非 SOAP 文档。2. 请确保您使用 Adapter for Web Service 3.1.0 (或更高版本)。请参阅 <i>Adapter for Web Service 用户指南</i> 以确保适配器的级别跟您正在使用的 WebSphere InterChange Server 的版本是兼容的。

注: 如果 SOAP 文档包含附件, 则您不必使用附件数据处理程序来处理它们。

社区参与者如何调用 Web Service

当社区参与者发送协作请求, 该协作是作为社区管理者提供的 Web Service 公开的, 会发生下列步骤:

1. 社区参与者将 SOAP 请求消息发送到 WSDL 文档 (为协作生成的) 中指定的目的地。注: WSDL 中指定的端点是 Business Integration Connect 的 Web Service 目标 (URL), 而不是实际的端点。
2. Business Integration Connect 接收消息并将其路由到 Adapter for Web Service。
3. Adapter for Web Service 将 SOAP 消息发送到 SOAP 数据处理程序, 以将 SOAP 消息转换成业务对象。适配器调用作为 Web Service 公开的协作。
4. 如果这是请求 / 响应操作, 则协作返回 SOAP 响应 (或故障) 业务对象。
5. 如果协作返回了 SOAP 响应 (或故障) 业务对象, 则 Adapter for Web Service 调用 SOAP 数据处理程序来将 SOAP 响应 (或故障) 业务对象转换成 SOAP 响应消

息。适配器将响应返回给 Business Integration Connect。如果协作未返回 SOAP 响应（或故障）业务对象，则 Adapter for Web Service 返回相应的 HTTP 响应状态码。

6. Business Integration Connect 将响应路由至 Web Service。

社区管理者如何调用 Web Service

由 Business Integration Connect 提供的公共 WSDL 可用于创建使用 WSDL ODA 的业务对象。请注意，当社区参与者提供的 Web Service 供社区管理者使用时，社区管理者用来调用 Web Service 的公共 URL 应该包含下列查询字符串，这一点很重要：

`?to=<Community participant Web Service Provider's business ID>`

例如，下列地址告诉：Business Integration Connect Web Service 的供应商是企业标识为 123456789 的参与者：

`http://WBICHost/bcgreceiver/Receiver?to=123456789`

WSDL ODA 将不会在 Web Service 顶级业务对象的 URL 属性缺省值中添加查询字符串。

当协作发送请求（到 Adapter for Web Service）以调用社区参与者的 Web Service 时，将发生以下步骤：

1. 协作向适配器发送服务调用请求，这会调用 SOAP 数据处理程序来将业务对象转换成 SOAP 请求消息。
2. 通过将 SOAP 消息发送到 Business Integration Connect 上的外部 Web Service 目标（URL），适配器调用 Web Service。
3. Business Integration Connect 充当代理，将 SOAP 消息发送到对应于目标（社区参与者）Web service 的端点。这会调用 Web Service。
4. 被调用的 Web Service 接收 SOAP 请求消息，并执行所请求的处理。
5. 被调用的 Web Service 发送 SOAP 响应（或故障）消息。如果是单向操作，则返回相应的 HTTP 状态码。
6. 如果这是请求 / 响应 Web Service，Business Integration Connect 将 SOAP 响应（或故障）消息路由至适配器，适配器调用数据处理程序来将其转换成响应或故障业务对象。连接器将 SOAP 响应或故障业务对象返回给协作。

第 4 章 JMS 上的 InterChange Server 集成

本章描述了如何在 JMS 传输协议上将 WebSphere Business Integration Connect 和 WebSphere InterChange Server 集成在一起。并提供了关于如何在 JMS 上配置 InterChange Server 以及基于 JMS 的通信所需的与 ICS 兼容的组件。

注：有关如何在 JMS 上配置 WebSphere Business Integration Connect 以与 InterChange Server 通信的信息，请参阅第 37 页的『为 InterChange Server 配置 Business Integration Connect』。有关如何配置 InterChange Server 的一般信息，请参阅第 40 页的『配置 InterChange Server』。

本章提供了以下有关如何通过使用 JMS 传输协议在 WebSphere Business Integration Connect 和 WebSphere InterChange Server 间接收和发送文档的信息：

- 『通过 JMS 传输的文档所需的组件』
- 第 106 页的『设置 JMS 传输的环境』
- 第 109 页的『创建 JMS 的业务对象定义』

通过 JMS 传输的文档所需的组件

对于 JMS 传输协议上与 InterChange Server 通信的 Business Integration Connect 需要配置这两个组件以使用 JMS。表 70 总结了这些配置步骤。

表 70. 配置使用 JMS 传输协议的 Business Integration Connect 和 InterChange Server。

组件	版本	了解更多信息
WebSphere Business Integration Connect	4.2.2	第 38 页的『配置通过 JMS 传输协议传递的出站文档』 第 39 页的『配置通过 JMS 传输协议传递的入站文档』
WebSphere InterChange Server	4.1.1, 4.2.0, 4.2.1, 4.2.2	第 112 页的『创建 JMS 的 ICS 构件』

另外，为了在 JMS 传输协议上 Business Integration Connect 和 InterChange Server 间收发文档，可以使用表 71 里列出的与 ICS 兼容的组件。

表 71. 通过 JMS 与 InterChange Server 传输文档所需的组件

组件	描述	注意事项和限制
WebSphere Business Integration Adapter for JMS (JMS 适配器)	该适配器允许 InterChange Server 与以 JMS 消息形式发送或接收数据的应用程序交换业务对象。Adapter for JMS 和 Business Integration Connect 通过 JMS 队列通信。	<ol style="list-style-type: none">1. 请确保使用 Adapter for JMS V2.3.1 (或更高版本) 它提供对定制头属性的支持。请参阅该适配器的文档，以确保适配器版本与正在使用的 InterChange Server 版本相兼容。2. Adapter for JMS 仅支持 JMS 文本消息。如果打算使用 JMS 字节消息，则使用 Adapter for JMS V2.5.0。

表 71. 通过 JMS 与 InterChange Server 传输文档所需的组件 (续)

组件	描述	注意事项和限制
有效负载数据处理程序	该数据处理程序在文档有效负载的文档格式及其业务对象表示之间进行转换。	要了解更多信息，请参阅第 108 页的『指定有效负载数据处理程序』。
附件数据处理程序	该数据处理程序把带有附件的文档转换成业务对象。	仅当文档包括附件时，才需要该数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

以下几节描述了在表 71 中的各组件如何一起工作，通过 JMS 传输协议在 Business Integration Connect 和 InterChange Server 间收发文档。

通过 JMS 传输发送文档

要使 Business Integration Connect 使用 JMS 传输协议将文档发送给 InterChange Server，请使用 Adapter for JMS 检索 Business Integration Connect 放入 JMS 队列中的消息。该适配器然后将消息路由给 InterChange Server。图 20 概述了 Business Integration Connect 如何通过 JMS 传输协议将文档发送到 InterChange Server。

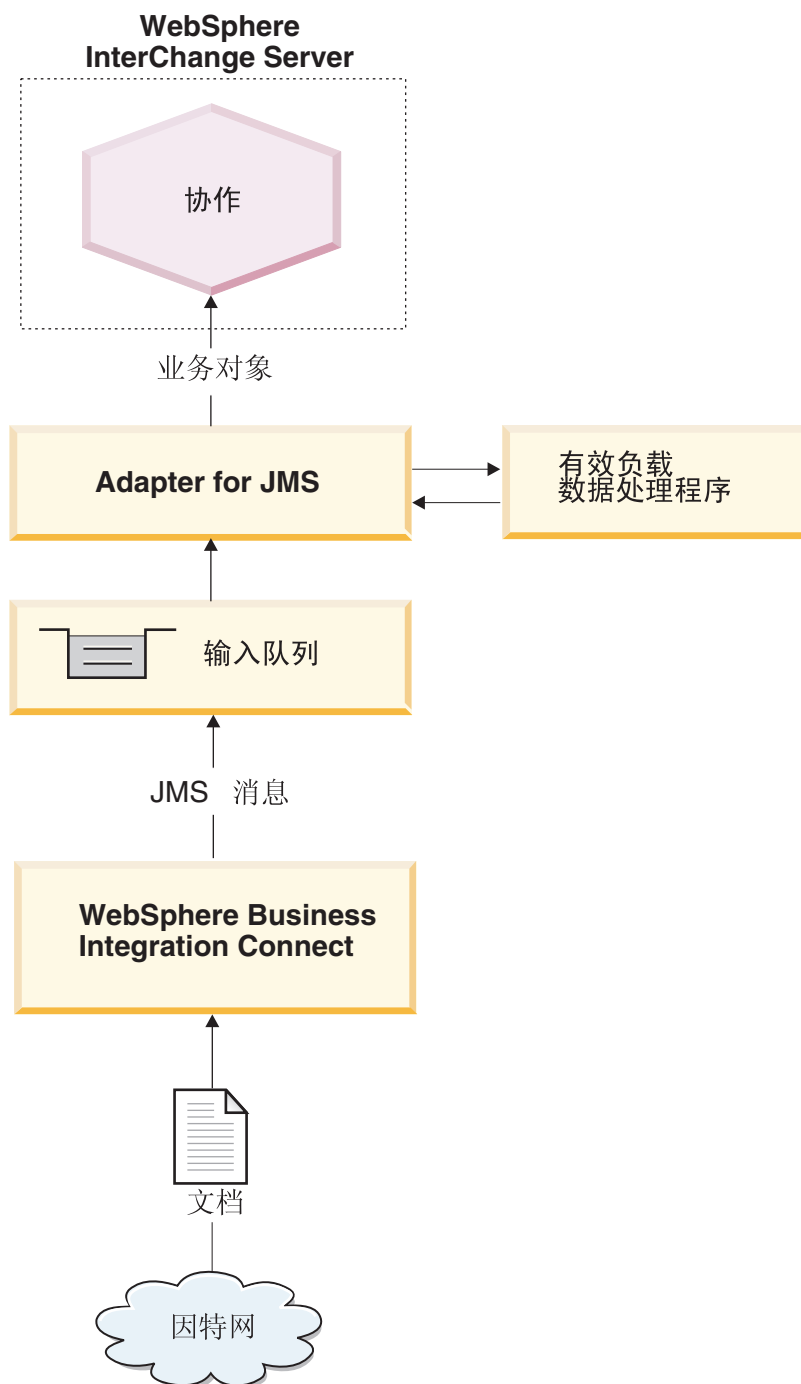


图 20. 通过 JMS 传输协议，从 Business Integration Connect 到协作的消息流

下列步骤描述了 Business Integration Connect 如何通过 JMS 传输协议将文档发送到 InterChange Server 内的协作来参与事件通知。

1. Business Integration Connect 将消息放入其 JMS 出站队列。

如果文档的封装类型是“后端集成”，那么 Business Integration Connect 已在该消息中提供了定制属性。用有效负载的内容类型设置 JMS 消息头 JMSType。

注：在 Business Integration Connect 内，必须配置一个标识 JMS 队列的网关，Business Integration Connect 将消息发送到该队列，而 Adapter for JMS 则会轮询该队列。要了解更多信息，请参阅第 38 页的『配置通过 JMS 传输协议传递的出站文档』。

2. 当 Adapter for JMS 在其任何一个输入队列上看到消息时，它就检索该消息。

Business Integration Connect 作为出站队列使用的 JMS 队列就是 Adapter for JMS 使用的输入队列。有关如何创建这个队列的信息，请参阅第 107 页的『配置 JMS 队列』。有关 Adapter for JMS 处理的详细信息，请参阅 *Adapter for JMS 用户指南*。

3. Adapter for JMS 将消息移到其正在处理的队列中。
4. Adapter for JMS 抽取 JMS 消息的主体，并根据该消息主体调用数据处理程序。该数据处理程序将 JMS 消息主体转换成业务对象。

注：如果消息有附件，则安装附件数据处理程序，然后配置 Adapter for JMS 使之能调用该数据处理程序，以便将 JMS 信息的主体转换成业务对象。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

当“后端集成”是封装类型且文档包含附件时，已配置的数据处理程序负责处理有效负载和附件。

5. 数据处理程序将业务对象返回给 Adapter for JMS。

注：如果使用附件数据处理程序，则这个业务对象包含有效负载和附件。

6. 如果 Adapter for JMS 找到子代动态元对象（使用特定于应用程序的业务对象级信息的 `cw_mo_conn` 指定），那么该适配器用 JMS 消息中的头填充业务对象中用户定义的 JMS 头。
7. Adapter for JMS 传递业务对象给 InterChange Server，将其作为订阅传递的部分。

通过 JMS 传输接收文档

要让 Business Integration Connect 使用 JMS 传输协议接收来自 InterChange Server 的文档，请使用 Adapter for JMS，它把接收自 InterChange Server 的消息放置在 Business Integration Connect 的 JMS 队列以供检索。图 21 概述了 Business Integration Connect 如何通过 JMS 传输协议接收来自 InterChange Server 的文档。

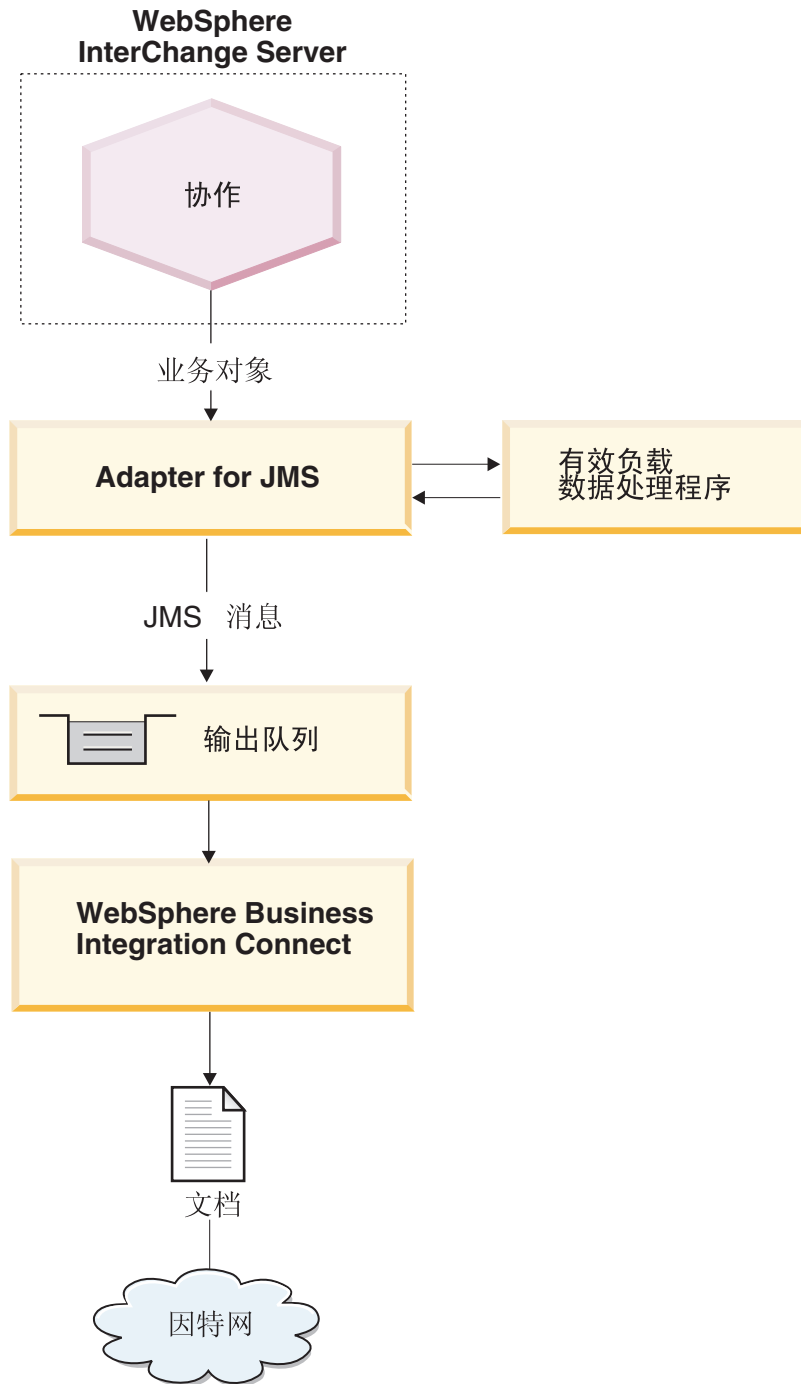


图 21. 通过 JMS 传输协议从协作到 Business Integration Connect 的消息流

下列步骤描述了 Business Integration Connect 如何通过 JMS 传输协议从 InterChange Server 内的协作接收文档来参与请求处理。

1. InterChange Server 内的协作对 Adapter for JMS 进行服务调用，向它发送请求业务对象。

请求业务对象包含了特定于应用程序的信息，该信息指向包含 JMS 传输级头信息（该信息是 Business Integration Connect 所期望的）的动态元对象。

2. Adapter for JMS 使用数据处理程序将由协作发送的业务对象转换成 JMS 消息。

该适配器读取 `DataHandlerMimeType` 和 `DataHandlerConfigMO` 属性以确定要使用的数据处理程序。要了解更多信息，请参阅第 108 页的『指定有效负载数据处理程序』。

注：如果文档有附件，则安装附件数据处理程序，然后配置 Adapter for JMS 使之能调用该数据处理程序，以便将请求业务对象转换成带附件的文档。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

3. 数据处理程序将业务对象转换成字符串，然后将它返回给 Adapter for JMS。
4. Adapter for JMS 根据请求业务对象确定定制 JMS 属性的动态元对象的名称。

适配器在请求业务对象的特定于应用程序信息中搜索 `cw_mo_conn` 标记，该标记标识包含动态元对象的属性。如果您对文档使用后端集成封装方式，那么可以在这个动态元对象中指定传输级头信息。

5. Adapter for JMS 搜索动态元对象以查找 `JMSProperties` 属性。

如果已填入该属性，那么适配器在请求文档中设置传输级的头。在 `JMSProperties` 属性内，还可以指定 `content-type` 标准 JMS 头。要了解更多信息，请参阅第 110 页的『创建 JMS 头信息』。

6. Adapter for JMS 通过使用由数据处理程序返回的字符串来创建 JMS 消息。它还设置动态元对象中所定义的任何定制属性。

注：在 2.4.1 版之前的 Adapter for JMS 版本只能写 JMS 文本消息。

7. Adapter for JMS 将产生的请求消息发送给输出队列。

可以在静态元对象或动态元对象中指定队列。有关指定队列的信息，请参阅第 108 页的『标识 JMS 队列』。Business Integration Connect 侦听该 JMS 队列，将该队列配置成其目标地的为入站队列。要了解更多信息，请参阅第 39 页的『配置通过 JMS 传输协议传递的入站文档』。

8. Business Integration Connect 从它的 JMS 入站队列接收该消息，正如它在目标中的配置。

注：Business Integration Connect 只支持通过 JMS 与 InterChange Server 进行的异步交互。因此，可能不需要等待响应。来自社区参与者或 Business Integration Connect 的响应可以来自不同队列。可以配置 Adapter for JMS 来轮询该队列。可以将来自队列的响应可以作为事件传递的一部分传递到 InterChange Server。

设置 JMS 传输的环境

因为接收来自 InterChange Server 的文档或向其发送文档涉及到与 ICS 兼容的组件，您必须执行表 72 中描述的设置和配置任务。有关如何配置 Business Integration Connect 以通过 JMS 与 InterChange Server 协作的信息，请参阅第 37 页的『为 InterChange Server 配置 Business Integration Connect』。

表 72. 设置 JMS 传输协议使用的环境

配置步骤	了解更多信息
1.配置 JMS 队列。	第 107 页的『配置 JMS 队列』
2.配置 WebSphere Business Integration Adapter for JMS。	第 107 页的『配置 Adapter for JMS』

注：如果文档中包含附件，您还必须安装和配置附件数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

配置 JMS 队列

为了和 InterChange Server 一起使用 JMS 传输协议，必须安装 WebSphere MQ 提供的 JMS 系统。InterChange Server 支持的版本使用 WebSphere MQ 作为 JMS 提供程序。因此，您可以使用第 159 页的『用 WebSphere MQ 配置 JMS 协议』中的步骤来设置 JMS 传输协议机制。

请注意：第 159 页的『用 WebSphere MQ 配置 JMS 协议』中的步骤必须在 WebSphere Business Integration Connect 驻留的机器上执行。本指南假定 Adapter for JMS 和 InterChange Server 所需的 JMS 传输机制已经被作为 InterChange Server 安装的一部分而设置定成。

当创建在 Business Integration Connect 和 InterChange Server 之间使用的 JMS 队列时，请考虑以下几点：

- InterChange Server 安装过程部分包括 WebSphere MQ 队列管理器的创建。您可以和 Business Integration Connect 一起使用这个队列管理器。
- 当创建 JMS 队列别名时，可能希望给它们的命名能反映 Business Integration Connect 和 InterChange Server 之间流的方向。例如，如果创建表 73 的 Original queue name 列中列出的队列，可以将这些队列重命名，以表明 InterChange Server 的方向性，就如同表 73 的 Directional queue name 列中显示的一样。

表 73. 命名为 JMS 队列以反映 InterChange Server 方向性

初始队列名称	定向队列名称
inQ	ICS2WBIC
outQ	WBIC2ICS

配置 Adapter for JMS

Adapter for JMS 是与 ICS 兼容的组件，它允许 Business Integration Connect 以 JMS 消息形式与 InterChange Server 交换文档。它支持与 InterChange Server 进行下列交互：

- 对于请求处理，它接收来自 InterChange Server 的请求业务对象，将其转换为 JMS 消息（使用数据处理程序），并将 JMS 消息放在 JMS 队列上（请参阅第 105 页的图 21），Business Integration Connect 在那里从 JMS 队列上获取 JMS 消息。
- 对于事件通知，它轮询 JMS 队列以查找来自 Business Integration Connect 的 JMS 消息。当它找到一个 JMS 消息时，就把它转换成事件业务对象（使用数据处理程序）并把它发送给 InterChange Server。

请注意：WebSphere Business Integration Connect 不包括 WebSphere Business Integration Adapter for JMS。您必须单独获取该产品，并根据其 *Adapter for JMS 用户指南* 中的指示说明来安装它。请阅读本指南中描述的步骤以正确地安装和配置 Adapter for JMS，这一点非常重要。

当您配置好 Adapter for JMS 使之可以与 InterChange Server 进行通信时，请执行本节中的步骤来配置这个适配器，以接受来自 Business Integration Connect 的 JMS 消息。

- 第 108 页的『指定有效负载数据处理程序』

- 『标识 JMS 队列』

指定有效负载数据处理程序

如第 105 页的图 21 所示，Adapter for JMS 使用数据处理程序以将接收来自 InterChange Server 的业务对象转换成相应的 JMS 消息。

注： Adapter for JMS 调用的数据处理程序转换文档的有效负载。如果您的文档包装在 XML 传输封包（该封包包含附件或者“封包标志”是“是”），将附件数据处理程序配置成有效负载数据处理程序。要了解更多信息，请参阅第 44 页的『处理带附件的文档』。

要指出使用哪种数据处理程序来转换有效负载，必须采取第 41 页的『业务对象变换』中列出的步骤。另外，必须配置 Adapter for JMS 以使用该有效负载数据处理程序。在“连接器配置器”中，请采取以下步骤：

- 设置下列连接器配置属性以标识有效负载数据处理程序：
 - 用顶级数据处理程序元对象和受支持的 MIME 类型的名称分别设置 DataHandlerConfigMO 和 DataHandlerMimeType 属性。
 - 使用数据处理程序类的名称来设置 DataHandlerClassName 属性，以进行实例化。

注： 可以设置 DataHandlerConfigMO 与 DataHandlerMimeType 属性，也可以设置 DataHandlerClassName 属性。

- 在受支持的业务对象列表中包含了顶级数据处理程序元对象。

您也可以在静态或动态元对象里指定要使用的数据处理程序。在这些元对象里，相同的特性（DataHandlerMimeType, DataHandlerConfigMO 和 DataHandlerClassName）可作为属性使用。有关完整的描述，请参阅 *Adapter for JMS 用户指南*。

标识 JMS 队列

当 Adapter for JMS 接收来自 InterChange Server 的文档时，将消息放到它的出站队列里，该队列也就是 Business Integration Connect 接收器要轮询的队列。类似的，当 Business Integration Connect 向 InterChange Server 发送文档时，它将文档放到它的出站队列里，该队列也就是 Adapter for JMS 要轮询的队列。

表 74 总结了如何配置 Adapter for JMS 用来接收和发送文档的 JMS 队列。

注： 有关如何配置 JMS 队列的完整描述，请参阅 *Adapter for JMS 用户指南*。

表 74. JMS 队列

JMS 队列	配置设置
输入队列	<p>将 InputDestination 连接器配置属性设置为 JMS 队列名，Adapter for JMS 将轮询该队列以查找入站消息。</p> <p>请确保该队列的名称和 Business Integration Connect 将其作为 JMS 出站队列的名称一样。如果没有在 InputDestination 中指定该队列，Adapter for JMS 将不轮询该队列。</p> <p>注： InputDestination 属性包含一个用逗号分隔的输入队列列表。如果 Adapter for JMS 轮询多个队列，那么请确保该列表包含 Business Integration Connect 将其用作 JMS 出站队列的 JMS 队列名。</p>

表 74. JMS 队列 (续)

JMS 队列	配置设置
输出队列	在运行时，协作可以将动态元对象中的 <code>OutputQueue</code> 属性动态地设置成 JMS 队列名，Adapter for JMS 将用它发送出站消息。

必须确保配置了静态或动态元对象，从而可以将它们写到 Business Integration Connect 在侦听的队列。

创建 JMS 的业务对象定义

Adapter for JMS 以有效负载业务对象的形式向 InterChange Server 发送文档以及接收来自它的文档。当 Adapter for JMS 接收或发送 Business Integration Connect 文档时，Adapter for JMS 调用有效负载数据处理程序以处理该业务对象，如下所示：

- 对于请求处理，有效负载数据处理程序将请求业务对象转换成相应的 JMS 消息。
- 对于事件通知，数据处理程序将 JMS 消息转换成一个事件业务对象。

因此，您必须创建表 75 中显示的业务对象定义，以表示 Adapter for JMS 期望的有效负载业务对象结构。

表 75. Adapter for JMS 的业务对象定义

条件	业务对象定义	了解更多信息
对于消息，如果您正在使用“无”或“后端集成”封装，并且文档没有附件	有效负载业务对象	『创建用于 JMS 的有效负载业务对象结构』。
对于文档，如果正在使用“后端集成”封装	用于保存消息头信息的业务对象： <ul style="list-style-type: none"> • 动态元对象 • JMS 属性业务对象 	第 110 页的『创建 JMS 头信息』。
如果文档包含附件	还必须创建其它业务对象来表示这个附件。	第 54 页的『创建与附件关联的业务对象定义』

创建用于 JMS 的有效负载业务对象结构

JMS 传输协议有效负载业务对象的结构取决于如下的封装类型：

- 如果文档使用“无”封装，则对于通过 JMS 文档传输协议发送的文档没有创建有效负载业务对象这一特定需求。

正如在第 40 页的『用于文档的业务对象』中讨论的一样，必须为每部分您需要传输的有效负载信息创建属性。

- 如果文档使用“后端集成”封装，则必须采取以下步骤：
 - 添加有效负载业务对象标识特定属性以定义动态元对象。该属性提供了有关传输级消息头的配置信息。
 - 将 `cw_mo_conn` 标记添加到特定于应用程序的业务对象级信息中，以标识包含动态元对象的属性。

有关这些步骤的更多信息，请参阅第 110 页的『创建 JMS 头信息』。

注：对于请求处理，JMS 传输协议仅支持异步交互。您可以发送请求业务对象，但不能获得响应。因此，必须创建请求业务对象定义，而不是用于响应的业务对象定义。

创建 JMS 头信息

如果通过 JMS 传输协议接收或发送文档，而这些文档使用“后端集成”封装，请求业务对象需包含定制传输级头信息。Adapter for JMS 期望该定制的头信息在其动态元对象中。

图 22 显示了业务对象结构，结构是 Adapter for JMS 将该结构用于请求业务对象，以表示使用后端集成封装的 Business Integration Connect 文档。

注：Adapter for JMS 用户指南提供了关于这个必需的业务对象结构的信息。当定义业务对象定义时，请参阅本指南。

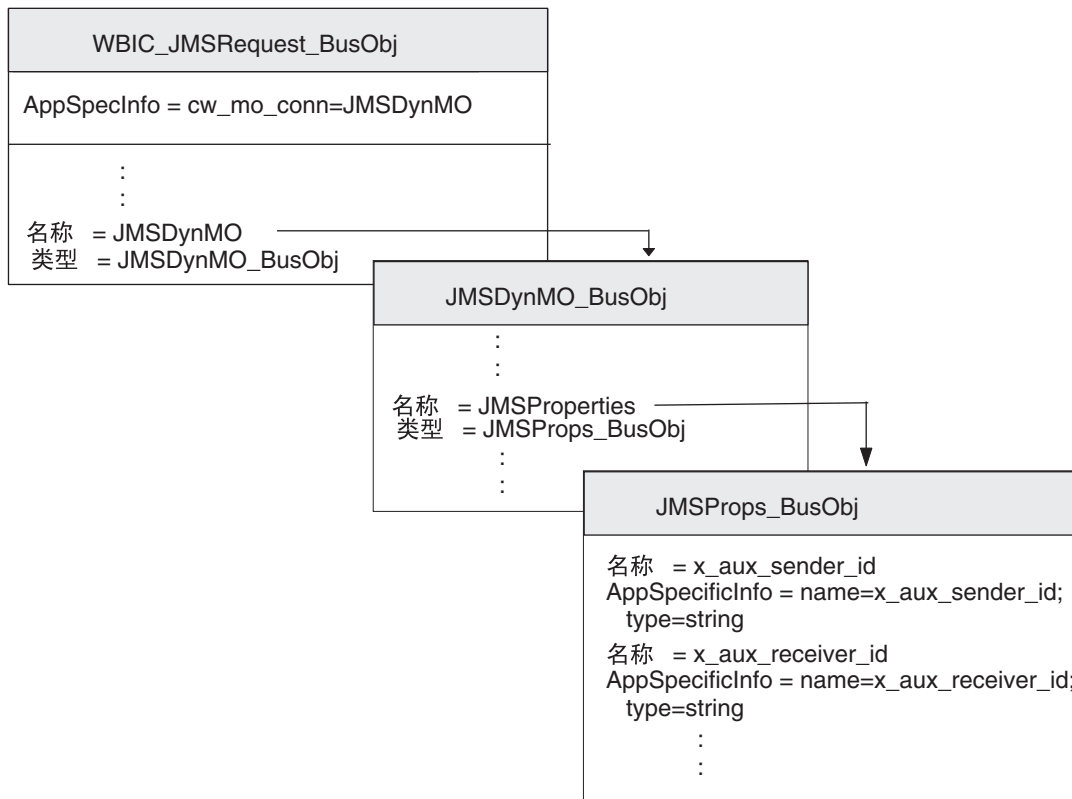


图 22. 请求业务对象与 JMS 动态元对象的关系

请通过采用下列步骤来确保业务对象结构包含了动态子元对象：

1. 创建业务对象定义以保存“后端集成”封装所需的 JMS 属性。
2. 为动态子元对象创建业务对象定义。
3. 修改请求业务对象的业务对象定义，以包含动态元对象的属性。

下面几节逐一描述了这些步骤。

创建 JMS 属性业务对象

JMS 属性业务对象包含传输级头所需的 JMS 属性，它是“后端集成”封装所需的。它还可以包含内容类型属性（该属性指定要在请求消息中设置的内容类型头）和内容长度属性（以字节为单位，指定了消息长度）。第 10 页的表 4 描述了各个有效传输头字段。

要创建 JMS 属性业务对象定义，请采用下列步骤：

1. 在业务对象定义中为每个传输级头的字段创建属性。

所有属性的属性类型都为“字符串”。对于 JMS 信息，传输头字段的名称使用下划线代替连字符，如表 76 中所示。

2. 对于 JMS 属性业务对象定义中的每个属性，添加特定于应用程序的信息，以标识相关属性的用途。

这个特定于应用程序的属性级信息必须具有如下格式：

```
name=JMSproperty;type=string
```

其中 *JMSproperty* 是表 76 中的 JMS 属性名列的一个值。

3. 对于 JMS 属性业务对象定义中的任何属性，可以为传输级字段添加缺省值来指定公共（或仅有效）值。

表 76. JMS 属性业务对象定义的属性

传输头字段	JMS 属性名
x-aux-sender-id	x_aux_sender_id
x-aux-receiver-id	x_aux_receiver_id
x-aux-protocol	x_aux_protocol
x-aux-protocol-version	x_aux_protocol_version
x-aux-process-type	x_aux_process_type
x-aux-process-version	x_aux_process_version
x-aux-create-datetime	x_aux_create_datetime
x-aux-msg-id	x_aux_msg_id
x-aux-production	x_aux_production
x-aux-system-msg-id	x_aux_system_msg_id
x-aux-payload-root-tag	x_aux_payload_root_tag
x-aux-process-instance-id	x_aux_process_instance_id
x-aux-event-status-code	x_aux_event_status_code
x-aux-third-party-bus-id	x_aux_third_party_bus_id
x-aux-transport-retry-count	x_aux_transport_retry_count
content-type	content_type
content-length	content_length

注：表 76 没有提供后端集成所需头的详尽列表。有关头的完整列表和描述，请参阅第 10 页的『传输级头的内容』。请务必用下划线替代传输头字段名中的所有连字符。

在第 110 页的图 22 中，JMSProps_BusObj 业务对象定义包含了各种传输级头字段的属性。这些属性都有特定于应用程序的属性级信息以指定相关协议头的名称。例如，x_aux_sender_id 属性有如下的特定于应用程序的信息设置：

```
name=x_aux_sender_id;type=string
```

创建 JMS 动态元对象

这个动态元对象包含了一个子业务对象以及 Adapter for JMS 的配置信息。要创建动态元对象的业务对象定义，请遵循以下步骤：

1. 创建名为 JMSProperties 的属性，该属性类型是 JMS 属性业务对象的业务对象定义（请参阅第 111 页的『创建 JMS 属性业务对象』）。
2. 相应地添加其它配置属性。有关动态元对象中有效属性的列表，请参阅 *Adapter for JMS 用户指南*。请查阅本指南以获取关于如何创建属性以配置动态元对象的信息。

为了使 Adapter for JMS 和 Business Integration Connect 一起工作，动态元对象的业务对象定义必须包含名为 JMSProperties 的属性，这个属性的属性类型是 JMS 属性业务对象（请参阅第 111 页的『创建 JMS 属性业务对象』）的业务对象定义。例如，在第 110 页的图 22 中，JMSDynMO_BusObj 业务对象定义包含了各配置属性的属性（未列出）并包含了 JMSProperties 属性。

修改请求业务对象定义

为将业务对象结构合并到请求业务对象中，必须对请求业务对象定义做以下修改：

1. 将一个属性添加到请求业务对象定义中，以保存动态子元对象。

该属性的属性类型是动态元对象的业务对象定义（请参阅『创建 JMS 动态元对象』）。

2. 将 `cw_mo_conn` 标记添加到请求业务对象定义的特定于应用程序的业务对象级信息中，以标识包含动态元对象的属性。

`cw_mo_conn` 标记的格式如下：

```
cw_mo_conn=dynamicMetaObjAttr
```

其中 `dynamicMetaObjAttr` 是包含了动态元对象的请求业务对象中的属性名。

例如，在第 110 页的图 22 中，已经将名为 JMSDynMO 的属性添加到请求业务对象定义 WBIC_JMSRequest_BusObj 中。该属性包含了动态元对象，这是类型为 JMSDynMO_BusObj 的子业务对象。另外，已经对请求业务对象的特定于应用程序的信息进行了修改，使之包含下列 `cw_mo_conn` 标记，以标识这个动态元对象：

```
cw_mo_conn=JMSDynMO
```

创建 JMS 的 ICS 构件

为了配置 InterChange Server，使之通过 JMS 传输协议与 Business Integration Connect 通信，必须创建表 77 中所示的 InterChange Server 构件。

表 77. 通过 JMS 传输协议用于通信的 ICS 构件

ICS 构件	目的	了解更多信息
业务对象定义	表示文档	第 109 页的『创建 JMS 的业务对象定义』
连接器对象	表示位于运行时的 Adapter for JMS	第 113 页的『创建 JMS 连接器对象』
协作模板和协作对象	表示 InterChange Server 用于处理文档的业务流程	第 113 页的『绑定协作以与 Adapter for JMS 通信』

创建 JMS 连接器对象

为获得 Adapter for JMS 运行时实例，在“系统管理器”中采取以下步骤：

1. 创建连接器对象：

- 创建一个表示 Adapter for JMS 实例的连接器对象。

注：在“连接器配置器”的“受支持的业务对象”选项卡中，请务必指定所有的业务对象定义，创建这些定义是为了与 Adapter for JMS 一起使用。要了解关于这些业务对象定义的描述，请参阅第 109 页的『创建 JMS 的业务对象定义』。

- 如果协作需要连接器对象，则为“端口连接器”创建一个。

2. 配置连接器对象。

有关如何配置 Adapter for JMS 以便与 Business Integration Connect 一起使用的信息，请参阅第 107 页的『配置 Adapter for JMS』。

绑定协作以与 Adapter for JMS 通信

如同第 43 页的『创建协作』中描述的，协作对象必须存在于 InterChange Server 的运行中，从而知道在何处接收和发送业务对象。当为协作创建协作对象时，该协作使用 Adapter for JMS 向 Business Integration Connect 发送信息和接收来自它的信息，绑定协作端，如下所示：

- 对于请求处理：“发送”端口，它将请求发送到 Business Integration Connect，因此将其设置到为 Adapter for JMS 创建的连接器对象；即 Adapter for JMS 是目标适配器。
- 对于事件通知：“接收”端口，它接收来自 Business Integration Connect 的事件，应该将其设置到 Adapter for JMS 创建的连接器对象；即 Adapter for JMS 是源适配器。

第 3 部分 与其它后端系统集成

第 5 章 与 WebSphere Business Integration Message Broker 集成

本章描述了如何将 WebSphere Business Integration Connect 与 WebSphere Business Integration Message Broker 集成在一起。

注:

1. 要获取用于将 Business Integration Connect 与后端系统集成在一起的一般过程的描述, 请参阅第 3 页的第 1 章, 『规划后端集成』。
2. 本章假定您熟悉 WebSphere Business Integration Message Broker 和相关组件, 例如项目和消息流。

通常 WebSphere Business Integration Connect 和后端系统的集成由两个独立的人员或角色完成。每个角色配置一个特定的组件, 相应的角色对每个组件具有专业知识。因此, 本章将与 WebSphere Business Integration Message Broker 的集成划分成 WebSphere Business Integration Connect 的配置和 Message Broker 的配置。表 78 列出了配置角色, 以及在哪可以获得相关的配置消息。

表 78. 用于 Message Broker 集成的角色

配置角色	更多信息
WebSphere Business Integration Connect 的配置	1. 第 118 页的『规划与 Message Broker 的集成』 2. 第 119 页的『为 Message Broker 配置 Business Integration Connect』
WebSphere Business Integration Message Broker 的配置	1. 第 118 页的『规划与 Message Broker 的集成』 2. 第 121 页的『配置 Message Broker』

注: 尽管这些配置角色中的每一个都可以单独执行, 但每个角色也需要公用信息, 以便两个组件可以通信。

本章提供了下列信息:

- 第 118 页的『规划与 Message Broker 的集成』
- 第 119 页的『为 Message Broker 配置 Business Integration Connect』
- 第 121 页的『配置 Message Broker』
- 第 122 页的『将 HTTP 传输协议和 Message Broker 一起使用』
- 第 125 页的『发送 SOAP 文档』
- 第 126 页的『将 JMS 传输协议和 Message Broker 一起使用』

规划与 Message Broker 的集成

要规划至 WebSphere Business Integration Message Broker 的集成，请遵循第 5 页的『规划后端集成』中概述的步骤。表 79 总结了将 Business Integration Connect 与 Message Broker 集成在一起所要进行的集成步骤。

表 79. 规划与 WebSphere Business Integration Message Broker 的集成

集成步骤	更多信息
1. 确认您已安装支持的 WebSphere Business Integration Message Broker 版本并可用于 WebSphere Business Integration Connect。	第 5 章: 『Business Integration Connect 支持的 Message Broker 版本』
2. 确定 WebSphere Business Integration Message Broker 文档的业务协议。	第 1 章: 第 5 页的 『使用何种业务协议?』
3. 确定文档的封装类型: “无” 或 “后端集成”。	第 1 章: 第 9 页的 『您将使用哪种封装?』
4. 确定要在 WebSphere Business Integration Connect 和 WebSphere Business Integration Message Broker 之间。	第 5 章: 『Message Broker 支持的消息传输』
5. 配置 WebSphere Business Integration Connect。	第 5 章: 第 119 页的 『为 Message Broker 配置 Business Integration Connect』

Business Integration Connect 支持的 Message Broker 版本

Business Integration Connect V4.2.2 可以支持与 Message Broker V5.0 版的集成。Message Broker 可用于多个平台，包括 Windows 2000 和多个基于 UNIX 的平台。要获取更多信息，参考在 WebSphere Business Integration Message Broker 文档集中的 Message Broker 安装指南。

Message Broker 支持的消息传输

Business Integration Connect 支持第 17 页的表 10 中所示的消息传输协议。在这些受支持的协议中，WebSphere Business Integration Message Broker 支持下列两种消息传输协议:

- HTTP 传输协议 (包括 Web Service)
- JMS 传输协议

要支持这些传输协议，需要安装和配置 IBM WebSphere MQ。

将 HTTP 和 Message Broker 一起使用

Message Broker 将 HTTP 传输协议用于其 Web Service 事务。要通过 HTTP 协议在 Business Integration Connect 和 Message Broker 之间接收和发送文档不需要其它软件。但是，要从 Message Broker 中发送文档到其它目的地则需要 WebSphere MQ。

注: Business Integration Connect 支持异步和通过 HTTP 与 Message Broker 进行异步和同步交互。

将 JMS 和 Message Broker 一起使用

Message Broker 将 JMS 传输协议用于其大多数事务。要通过 JMS 传输协议在 Business Integration Connect 和 Message Broker 之间发送与接收文档，则您必须使用 JMS 队列。如果这两个组件驻留在不同机器上，您必须在每台机器上创建 JMS 队列。基本上，对

JMS 的支持涉及了使用 Message Broker 和底层 JMS 队列的消息流。有关如何配置 JMS 的更多信息，请参阅第 126 页的『将 JMS 传输协议和 Message Broker 一起使用』。

注：Business Integration Connect 仅支持通过 JMS 与 Message Broker 进行异步交互。

支持 Message Broker 集成

Business Integration Connect 提供样本以在与 Message Broker 的集成过程中为您提供辅助。这些样本位于 Business Integration Connect 产品目录的下列子目录中：

Integration/WBI/WBIMB/samples

为 Message Broker 配置 Business Integration Connect

第 23 页的『配置 Business Integration Connect』中提供了有关如何配置 Business Integration Connect 以与后端系统进行通信的一般概述。本节总结了配置 Business Integration Connect 以与 Message Broker 进行通信所需的步骤。要执行该配置，请使用在中心社区中充当社区管理者的 Business Integration Connect Enterprise Edition 或 Advanced Edition 的一个实例。

配置 Business Integration Connect 包括下列步骤：

- 配置出站文档支持

有关从 Business Integration Connect 发送文档到 Message Broker 的信息，请参阅『提供对出站文档的支持』。

- 配置入站文档

有关从 Message Broker 发送文档到 Business Integration Connect 的信息，请参阅第 120 页的『提供对入站文档的支持』。

提供对出站文档的支持

要使 Business Integration Connect 能将文档发送到任何后端系统，您必须执行第 24 页的『定义将参与者文档发向何处』中所描述的步骤。后端系统是 Message Broker 时，您需要创建网关，该网关的传输类型与在 Business Integration Connect 和 Message Broker 之间传递的消息所用的传输协议相匹配。当社区管理者将文档发送到 Message Broker 时，它必须清楚要将文档路由到何处。这个位置必须符合使用的传输协议。传输协议必须是 Message Broker 所支持的传输协议（请参阅第 118 页的『Message Broker 支持的消息传输』）。

下面几节总结了如何创建下列 Message Broker 所支持传输协议的网关：

- 『配置通过 HTTP 传输协议传递的出站文档』
- 第 120 页的『配置通过 JMS 传输协议传递的出站文档』

配置通过 HTTP 传输协议传递的出站文档

当社区管理者通过 HTTP 协议将文档发送到 Message Broker 时，社区管理者通过已定义的网关来路由消息。这个网关标识了 Message Broker 可以从其接收文档的 URL。当 Message Broker 使用 HTTP 协议时，它将文档路由到与指定 URL 相关的消息流的 HTTPInput 节点。

要使社区管理者可以在 HTTP 传输协议上通过网关发送文档，则必须从社区控制台的“网关详细信息”屏幕创建网关。该网关必须配置为使用 HTTP 1.1 传输协议。并写到相应 HTTPInput 节点正在侦听的 URL。如同表 80 所示，在网关定义的目标 URL 字段中提供的 URL。

注：有关如何创建网关的概述，请参阅第 24 页的『定义将参与者文档发向何处』。

表 80. 用以与 Message Broker 通信的“网关详细信息”屏幕上的 HTTP 值

“目标详细信息”字段	值	注意事项和限制
目标 URI	该 URL 应当与为 Message Broker 中的 HTTPInput 节点所配置的那个 URL 相同。	从 WebSphere Business Integration Message Broker 集成中的消息流配置获取这个 URL。

配置通过 JMS 传输协议传递的出站文档

当社区管理者通过 JMS 协议将文档发送给 Message Broker 时，社区管理者将文档路由给相应的 JMS 队列，在此可将它传输到 Message Broker 能检索它的 JMS 队列。要使社区管理者获取这个 JMS 位置，您必须在 Business Integration Connect 中创建一个网关，这个网关使用 JMS 传输协议。必须对该网关进行配置，以写这样一个队列：其内容被传输到 Message Broker 在其上接收消息的队列。

注：有关如何创建网关的概述，请参阅第 24 页的『定义将参与者文档发向何处』。

要使社区管理者可以在 JMS 传输协议上通过网关发送文档，则必须从社区控制台的“网关详细信息”屏幕创建网关。当您使用 WebSphere MQ V5.3 作为您的 JMS 提供程序时，则请使用第 165 页的表 116 中的信息来设置目标字段。另外，在“网关详细信息”屏幕中为 JMS 协议指定表 81 中所确定的信息。

表 81. 用以与 Message Broker 通信的“网关详细信息”屏幕上的 JMS 值

“网关详细信息”字段	值	注意事项和限制
JMS 队列名	Business Integration Connect 所驻留的机器上的 JMS 队列的名称	该队列上接收到的文档被传输到 Message Broker 所驻留机器上的 JMS 队列。

提供对入站文档的支持

要使 Business Integration Connect 能接收来自任何后端系统的信息，您必须执行第 28 页的『定义从何处检索后端文档』中所描述的步骤。当您的后端系统是 Message Broker 时，需要以社区管理者身份采取下列步骤：

1. 作为社区管理者的参与者概要文件的一部分，定义网关类型并提供接收器所侦听的相关 IP 地址。
2. 创建一个目标，该目标的传输类型与 Business Integration Connect 和 Message Broker 之间传递的文档所用的传输协议相匹配。

要使社区管理者接收来自 Message Broker 的文档，必须清楚在何处检索消息。这个位置必须符合将要使用的传输协议。传输协议必须是 Message Broker 所支持的传输协议（请参阅第 118 页的『Message Broker 支持的消息传输』）。

下面几节总结了如何创建 Message Broker 所支持传输协议的目标。

配置通过 HTTP 传输协议传递的进站文档

当社区管理者通过 HTTP 传输协议接收文档时，其接收器从已定义的目标中检索文档。这个目标标识了接收器在其上侦听来自 Message Broker 的文档的 URL。当 Message Broker 使用 HTTP 传输协议时，HttpRequest 节点将文档发送到相应的 URL，社区管理者可以在此接收文档。

要使社区管理者可以在 HTTP 传输协议上通过目标接收文档，则必须从社区控制台的“目标列表”屏幕创建目标。这个目标必须使用 HTTP 1.1 传输协议。社区管理者确定这个由以下信息组成的 URL：

- 主机的 IP 地址，从社区管理者的参与者概要文件中获取
- 目标 URL，从目标定义的 URL 字段中获取

注：第 28 页的『定义从何处检索后端文档』中提供了关于如何创建目标的概述。

要使 Message Broker 能够将文档发送到该目标，必须配置消息流的 HttpRequest 节点以发送文档到该 URL。因此，您必须确保该目标可用于 Message Broker 配置。

配置通过 JMS 传输协议传递的进站文档

当社区管理者通过 JMS 协议从 Message Broker 接收文档时，社区管理者从相应的 JMS 输入队列中获得文档，该文档是从 Message Broker 发送到 JMS 输出队列，再从 JMS 输出队列发送到 JMS 输入队列的。要使社区管理者能够获取这个 JMS 位置，您必须在 Business Integration Connect 中创建一个目标，这个目标使用 JMS 传输协议。通过这个目标，社区管理者在它的进站队列上侦听任何文档并检索它们。

注：有关如何创建目标的概述，请参阅第 28 页的『定义从何处检索后端文档』。

要使社区管理者可以在 JMS 传输协议上通过目标接收文档，则必须从社区控制台的“目标列表”屏幕创建目标。当使用 WebSphere MQ V5.3 作为您的 JMS 提供程序，则请使用在第 164 页的表 115 中的信息来设置网关字段。另外，在“目标详细信息”屏幕中为 JMS 协议指定表 82 中所确定的信息。

表 82. 用以与 Message Broker 通信的“目标详细信息”屏幕上的 JMS 值

“目标详细信息”字段	值	注意事项和限制
JMS 队列名	自 Message Broker 输出队列接收文档的 JMS 输入队列名	本输入队列中的文档传输自 Message Broker 所驻留机器上的 JMS 输出队列。

配置 Message Broker

对于 Business Integration Connect 和 Message Broker 之间的交互，您必须在 Message Broker 工具箱的“代理应用程序开发透视视图”内创建一个消息流项目。该项目将包含下列构件：

- 消息流
- PIP 文件（仅 RosettaNet）或消息定义文件

注：要获取有关如何创建消息项目的更多信息，请参阅 WebSphere Business Integration Message Broker 文档集。

创建消息流

Message Broker 中的消息流执行处理信息所需的实际业务逻辑。因此，要使 Message Broker 能正确处理您的 Business Integration Connect 文档，相应的消息流必须存在。确保存在提供您所需业务逻辑的消息流：

- 如果这样一个消息流当前并不存在，您必须创建或导入一个。
- 如果确实存在消息流，则您必须了解如何有效地使用它，以便能使用它。

要使 Message Broker 能处理入站和出站文档，其消息流需要使用特定传输节点。要使用的传输节点类型取决于特定的传输协议，如下所示。

表 83. 创建用于不同传输协议的消息流

传输协议	更多信息
HTTP	第 124 页的『创建用于 HTTP 传输的消息流』
HTTP (SOAP 文档)	
JMS	第 131 页的『创建用于 JMS 传输的消息流』

部署项目

一旦您的消息流项目包含正确的构件，则您必须将其部署到 Message Broker。可以用 Message Broker 工具箱的“代理管理员透视图”来部署消息流项目。

将 HTTP 传输协议和 Message Broker 一起使用

本节描述了如何通过使用 HTTP 传输协议在 WebSphere Business Integration Connect 和 WebSphere Business Integration Message Broker 之间收发文档。

注：有关 HTTP 传输协议的所有参考资料也同样适用于 HTTPS。

通过 HTTP 传输传递文档所需的组件

使用 HTTP 传输协议在 Business Integration Connect 和 Message Broker 之间收发文档不需任何专门的组件。仅需 Business Integration Connect 和 Message Broker。要 Business Integration Connect 能使用 HTTP 传输协议与 Message Broker V5.0 进行通信，需要配置两个组件。表 84 中概述了这些配置步骤。

表 84. 配置 Business Integration Connect 与 Message Broker

组件	版本	更多信息
WebSphere Business Integration Connect	4.2.2	第 119 页的『配置通过 HTTP 传输协议传递的出站文档』 第 121 页的『配置通过 HTTP 传输协议传递的入站文档』
WebSphere Business Integration Message Broker	5.0	第 121 页的『配置 Message Broker』

另外，要使用 HTTP 传输协议在 Business Integration Connect 与 Message Broker 之间收发文档，您必须使用 IBM WebSphere MQ V5.3 作为 JMS 提供出程序。

通过 HTTP 传输发送文档

要使 Business Integration Connect 能通过 HTTP 传输协议将文档发送到 Message Broker，需要使用在 Message Broker 消息流内专门的 HTTP 传输节点来 Business Integration Connect 将其作为 HTTP 流发送的文档。消息流节点执行所需的计算，然后路由文档到特定的目的地(JMS 输出队列)。

下列步骤描述了 Business Integration Connect 如何通过 HTTP 传输协议将文档发送到 Message Broker 内的消息流:

1. Business Integration Connect 发送 HTTP 消息到 Message Broker。

如果文档的封装类型是后端集成，那么 Business Integration Connect 已在该消息中提供了定制属性。

注：在 Business Integration Connect 内，您必须配置一个网关，该网关标识了 Business Integration Connect 将发送消息以及 Message Broker 将在其上轮询的 URL。要了解更多信息，请参阅第 119 页的『配置通过 HTTP 传输协议传递的出站文档』。

2. 消息流的 HTTPInput 节点选取文档并将它发送至消息流的下一个节点。该节点通常时一个计算节点。
3. 消息流的节点执行业务逻辑的操作。

当完成了业务逻辑，消息流发送生成的文档到它的 HTTPReply 节点。

4. 缺省情况下，HTTPReply 节点将输出消息发送回客户机 (Business Integration Connect)。

可选的是，消息流可以把该消息放入 MQOutput 节点。MQOutput 节点接收文档并将它发送到相应的 JMS 队列或其它应用程序。

通过 HTTP 传输接收文档

要使 Business Integration Connect 能使用 HTTP 传输协议从 Message Broker 接收文档，可以在 Message Broker 消息流内使用特殊的 HTTP 传输节点来发送 Business Integration Connect 将作为 HTTP 轮流接收的文档。消息流节点执行所需的计算然后以 Business Integration Connect 处理请求和响应 (如果交互是同步的)。

下列步骤描述了 Business Integration Connect 如何通过 HTTP 传输协议从 Message Broker 内的消息流接收文档:

1. Message Broker 内的消息流接收其 MQInput 节点 (JMS 输入队列) 中的文档。
2. 消息流的 MQInput 节点接收文档并发送它到 HTTPRequest 节点。
3. HTTPRequest 节点通过客户机 (Business Integration Connect) 处理请求与响应交互，使用指定的 URL。
4. Business Integration Connect 从它的 URL 接收该消息，正如在其目标中配置的。

有关该目标的更多信息，请参阅第 121 页的『配置通过 HTTP 传输协议传递的入站文档』。

创建用于 HTTP 传输的消息流

为使 Message Broker 消息流通过 HTTP 传输协议来处理文档，它使用下面的传输节点：

- HTTPInput
- HTTPReply
- HTTPRequest

这些传输节点的使用顺序取决于通信方向，如下所示：

- 当 Business Integration Connect 发送文档到 Message Broker 时，消息流包括表 85 中（按照显示的顺序）节点类型来描述业务逻辑。
- 当 Business Integration Connect 自 Message Broker 接收文档时，消息流包括在表 86 中（按照显示的顺序）来描述业务逻辑。

表 85. 通过 HTTP 将文档发送到 Message Broker 的节点

节点类型	目的	注意事项
HTTPInput	将 Business Integration Connect 请求文档接收进消息流。	将该传输节点的“URL 选择器”字段（在基本属性中）设置为 Business Integration Connect 发送其文档的 URL。因此，必须将其设置为在 Business Integration Connect 网关中配置的 URL。URL 应当具有如下格式： <code>http://hostName:port/path</code> 其中 <i>hostName</i> 是 Message Broker 所驻留机器的名称， <i>port</i> 是 Message Broker 正在监听的 HTTP 端口号，而 <i>path</i> 标识在这台机器上的位置。 要了解更多信息，请参阅第 119 页的『配置通过 HTTP 传输协议传递的出站文档』。
计算	执行业务逻辑任务，例如更新标题信息	使用 ESQL 以执行业务逻辑。计算节点发送生成的消息到 HTTPReply 节点。
HTTPReply	将响应返回给 Business Integration Connect	缺省情况下，该节点发送输出信息到客户机。但是，您可以配置它，以将消息发送到 MQOutput 节点。
MQOutput	从 HTTPReply 节点接收文档将它发送到 Business Integration Connect	传输节点发送生成的文档到 JMS 输出队列，该队列将文档路由到下一个目的地。

表 86. 通过 HTTP 自 Message Broker 接收文档的节点

节点类型	目的	注
MQInput	从 Business Integration Connect 接收文档	该传输节点从 JMS 输入队列接收入站文档。

表 86. 通过 HTTP 自 Message Broker 接收文档的节点 (续)

节点类型	目的	注
HTTPRequest	与 Business Integration Connect 一起处理请求/响应交互	<p>该传输节点必须将其“Web Service URL”字段（在基本属性中）设置为 Business Integration Connect 正在其中侦听文档的 URL。对文档进行监听。因此，必须将其设置为在 Business Integration Connect 目标中配置的 URL。URL 应当具有如下格式：</p> <pre>http://hostName:port/bcgreceiver/path</pre> <p>其中 <i>hostName</i> 是 Business Integration Connect 所驻留机器的名称，<i>port</i> 是 Business Integration Connect 的 HTTP 端口号，而 <i>path</i> 标识在这台机器上的位置。</p> <p>要了解更多信息，请参阅第 121 页的『配置通过 HTTP 传输协议传递的入站文档』。</p>

要获得更多有关如何创建与配置消息流节点的详细信息，请参阅您的 WebSphere Business Integration Message Broker 文档。

发送 SOAP 文档

SOAP 文档不同于通过 HTTP/S 交换的其它类型文档。该节描述了如何通过 HTTP 传输协议在 WebSphere Business Integration Connect 和 WebSphere Business Integration Message Broker 之间来收发 SOAP 文档。

配置 Business Integration Connect 和 Message Broker 以传输 SOAP 文档的方法与配置通过 HTTP 协议传输非 SOAP 文档的配置方法非常相似。下表汇总了在何处查找有关如何配置这两个集成组件的信息。

表 87. 配置 Business Integration Connect 和 Message Broker 以传输 SOAP 文档

集成组件	配置步骤	更多信息
WebSphere Business Integration Connect	对于通过 HTTP 传递的非 SOAP 文档，请使用与非 SOAP 相同的方法来配置目标和网关。	<p>第 119 页的『配置通过 HTTP 传输协议传递的出站文档』</p> <p>第 121 页的『配置通过 HTTP 传输协议传递的入站文档』</p>
WebSphere Business Integration Message Broker	处理通过 HTTP 传递的非 SOAP 文档的消息流和处理 SOAP 文档的消息流非常相似。只是处理 SOAP 文档需要多一个传输节点。	<p>有关发送 SOAP 文档到 Message Broker 的消息，请参阅第 126 页的表 88。</p> <p>有关自 Message Broker 接收 SOAP 文档，请参阅第 124 页的『创建用于 HTTP 传输的消息流』。</p>

要使 Message Broker 正确处理 Business Integration Connect 发送的 SOAP 文档，消息流必须包含 HTTPRequest 节点以处理与 Web Service 客户机的通信。表 88 列出处理由 Business Integration Connect 发送的 SOAP 文档所需的 Message Broker 消息流中的节点。

表 88. 用于将 SOAP 文档发送到 Message Broker 的节点

节点类型	目的	注
HTTPInput	将 Business Integration Connect 请求文档接收消息流	<p>将该传输节点的“URL 选择器”字段（在基本属性中）设置为 Business Integration Connect 发送其文档的 URL。因此，必须将其设置为在 Business Integration Connect 网关中配置的 URL。URL 应当具有如下格式：</p> <p><code>http://hostName:port/path</code></p> <p>其中 <i>hostName</i> 是 Business Integration Connect 所驻留机器的名称，<i>port</i> 是 Business Integration Connect 的 HTTP 端口号，而 <i>path</i> 标识在这台机器上的位置。</p> <p>要了解更多信息，请参阅第 119 页的『配置通过 HTTP 传输协议传递的出站文档』。</p>
计算	执行业务逻辑任务，例如更新标题信息	使用 ESQL 以执行业务逻辑。计算节点发送生成的消息到 HTTPReply 节点。
HTTPRequest	发送 SOAP 请求到外部“Web Service 提供程序”（WebServices）并从 WebService 那获取返回的响应。	无
HTTPReply	返回响应到 Business Integration Connect	缺省情况下，该节点发送输出信息到客户机。

将 JMS 传输协议和 Message Broker 一起使用

本节描述了如何配置组件以通过使用 JMS 传输协议在 WebSphere Business Integration Connect 和 WebSphere Business Integration Message Broker 之间收发文档。它提供了下列有关如何发送与接收文档的信息：

- 『通过 JMS 传输文档所需的组件』
- 第 102 页的『通过 JMS 传输发送文档』
- 第 104 页的『通过 JMS 传输接收文档』

通过 JMS 传输文档所需的组件

使用 JMS 传输协议在 Business Integration Connect 和 Message Broker V5.0 之间收发文档需要 WebSphere MQ 作为提供程序。下面几节描述了 Business Integration Connect、Message Broker 以及 WebSphere MQ 如何协作以通过 HTTP 传输协议 JMS 交换文档。

通过 JMS 传输发送文档

要使 Business Integration Connect 使用 JMS 传输协议将文档发送到 Message Broker 需要使用 JMS 队列。Business Integration Connect 发送文档到它的 JMS 输出队列，在此将其传输到 Message Broker 监听的 JMS 输入队列。当 Message Broker 接收到文档时，它从它的输入队列中检索文档。Message Broker 的消息流包含特殊的 WebSphere MQ (JMS) 传输节点，这些节点处理对 JMS 队列的访问。图 23 概述了 Business Integration Connect 如何通过 JMS 传输协议发送文档到 Message Broker。

WebSphere Business Integration Message Broker

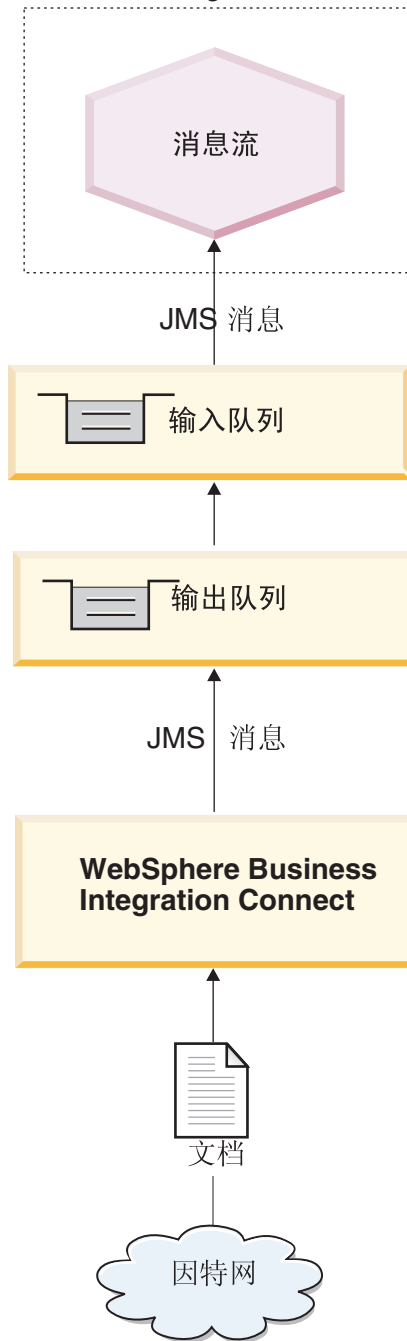


图 23. 通过 JMS 传输协议从 Business Integration Connect 到消息流的消息流

下列步骤描述了 Business Integration Connect 如何通过 JMS 传输协议将文档发送到 Message Broker 内的消息流:

1. Business Integration Connect 将消息发布到 JMS 输出队列信息。

如果文档的封装类型是后端集成, 那么 Business Integration Connect 已在该消息中提供了定制属性。JMS 消息头 JMSType 被设置为有效负载的内容类型。

注: 在 Business Integration Connect 内, 您必须配置一个网关, 该网关标识了 Business Integration Connect 将用于发送消息的以及 Message Broker 正在其上轮询的 JMS 输出队列。要了解更多信息, 请参阅第 120 页的『配置通过 JMS 传输协议传递的出站文档』。

2. WebSphere MQ 将文档从 Business Integration Connect 所驻留的机器上的输出队列传输到 Message Broker 所轮询的输入队列转换文档。
3. 当 Message Broker 在它的输入队列中发现消息时, 它检索信息并发送它到相应的消息流。

有关如何设置这个队列的信息, 请参阅 第 130 页的『设置 JMS 传输的环境』。

4. MQInput 节点将文档发送到消息流下一个节点。该节点通常是一个计算节点。
5. 消息流的节点执行业务逻辑的操作。

当完成了业务逻辑时, 消息流发送生成的文档到它的 MQOutput 节点。

6. MQOutput 节点将文档发送到相应的队列。

通过 JMS 传输接收文档

为了使 Business Integration Connect 通过 JMS 传输协议从 Message Broker 接收文档, 请使用 JMS 队列。Message Broker 发送文档到它的 JMS 输出队列, 在此将文档传输到 Business Integration Connect 侦听的 JMS 输入队列。当 Business Integration Connect 接收到文档。它从它的输入队列中检索文档。Message Broker 的消息流包含特殊的 WebSphere MQ (JMS) 传输节点, 这些节点处理对 JMS 队列的访问。图 24 概述了如何将文档从 Message Broker 发送到 Business Integration Connect。

WebSphere Business Integration Message Broker

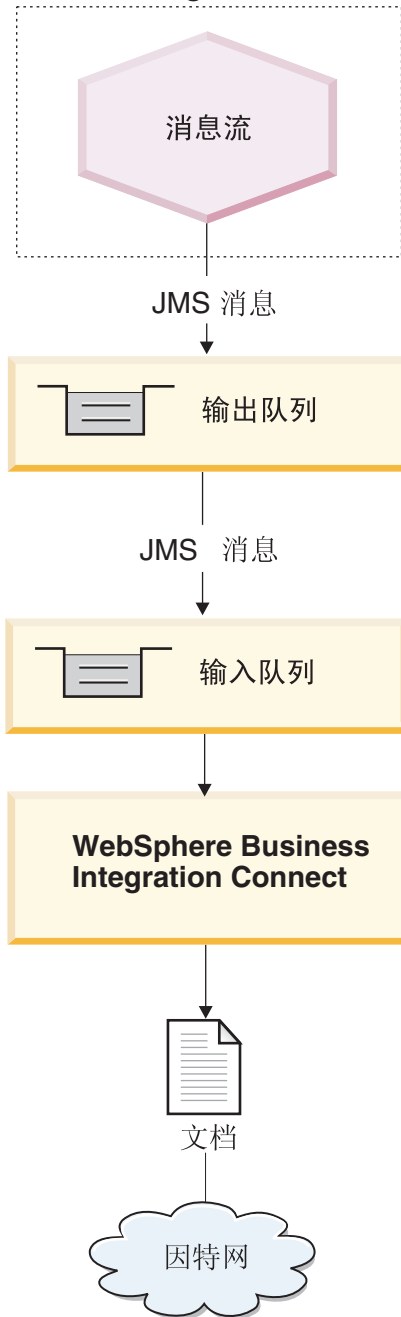


图 24. 通过 JMS 传输协议从消息流到 Business Integration Connect 的消息流

下列步骤描述了 Business Integration Connect 如何通过 JMS 传输协议接收来自 Message Broker 内消息流接收的文档:

1. Message Broker 内的消息流接收其 MQInput 节点中的文档。

消息流从 JMS 输入队列接收它的入站消息。

2. 消息流的 MQInput 队列接收文档并将他发送到消息流的下一个节点。该节点通常时一个计算节点。
3. 消息流的节点执行业务逻辑的操作。

当完成了业务逻辑时，消息流发送生成的文档到它的 MQOutput 节点。

4. MQOutput 节点发送文档到相应的 JMS 输出队列。
5. WebSphere MQ 将文档从 Message Broker 所驻留的机器上的队列传输到 Business Integration Connect 正在轮询的队列。
6. Business Integration Connect 从它的 JMS 输入队列接收该消息，正如其目标中配置的。

有关该目标的更多信息，请参阅第 121 页的『配置通过 JMS 传输协议传递的入站文档』。有关如何设置这个队列的信息，请参阅『设置 JMS 传输的环境』。

设置 JMS 传输的环境

将文档发送到 Message Broker 以及从它接收文档都涉及了 JMS 队列（远程和本地）。有关如何配置 Business Integration Connect 以通过 JMS 与 Message Broker 协作的信息，请参阅第 119 页的『为 Message Broker 配置 Business Integration Connect』。为了和 Message Broker 一起使用 JMS 传输协议，您可以设置 WebSphere MQ 提供的 JMS 系统。Message Broker V5.0 使用 WebSphere MQ V5.3 作为 JMS 提供程序。因此，您可以使用第 159 页的『用 WebSphere MQ 配置 JMS 协议』中的步骤来设置 JMS 传输协议机制。

要点：第 159 页的『用 WebSphere MQ 配置 JMS 协议』中的步骤必须在 WebSphere Business Integration Connect 驻留的机器上执行。本指南假定 Message Broker 所需的 JMS 传输机制已经被作为 Message Broker 安装的一部分而被设置。

当创建在 Business Integration Connect 和 Message Broker 之间使用的 JMS 队列时，请考虑以下几点：

- Message Broker 安装过程中应涉及下面队列管理器的创建：
 - 与代理域相关联的 WebSphere MQ 队列管理器

您可以使用以下命令来创建该队列管理器以及一组已命名的队列：

```
mqsicreatebroker
```

- Message Broker 的 WebSphere MQ 队列管理器

因为 Message Broker 使用一组预确定的队列名，因此对于每个代理都需要独立的 WebSphere MQ 队列管理器。Message Broker 可以与它的“配置管理器”和或可选的“用户名服务器”共享该队列管理器主机托管。

要获得更多的信息，请参阅您的 *WebSphere Business Integration Message Broker 安装与配置指南*。

- 当创建您的 JMS 队列别名时，您可能应该取那些能指明 Business Integration Connect 和 Message Broker 之间流方向的名称。

例如，如果创建表 73 的初始队列名栏中列出的队列，您可以将重命名这些队列，以表明 Message Broker 的方向性，就如同表 89 的方向队列名栏中显示的一样。

表 89. 命名 JMS 队列以表明 Message Broker 的方向性

初始队列名	方向队列名称
inQ	MB2WBIC
outQ	WBIC2MB

创建用于 JMS 传输的消息流

为了使 Message Broker 消息流能通过 JMS 传输协议处理文档，它使用下面的传输节点：

- MQInput
- MQOutput

这些传输节点的使用顺序取决于通信方向，如下所示：

- 当 Business Integration Connect 发送文档到 Message Broker 时，消息流包括表 90 中（按照所显示的顺序）的节点类型来描述业务逻辑。
- 当 Business Integration Connect 自 Message Broker 接收文档，消息流包括表 91 中（按照所显示的顺序）的节点类型来描述业务逻辑。

表 90. 用于通过 JMS 将文档发送到 Message Broker 的节点

节点类型	目的	注意事项和限制
MQInput	从 Business Integration Connect 接收文档	该传输节点的队列名字段（在基本属性中）的值是消息流的输入队列。必须设置 WebSphere MQ 以便于该 JMS 队列自 Business Integration Connect 的输出队列接收文档。要了解更多信息，请参阅第 120 页的『配置通过 JMS 传输协议传递的出站文档』。
计算	执行业务逻辑任务，例如除去标题信息	无
MQOutput	从计算节点接收文档并将它作为消息输出进行发送	该传输节点将生成的文档发送到 JMS 输出队列，该队列将它路由到下一个目的地。

表 91. 用于从 Message Broker 接收文档的节点

节点类型	目的	注意事项和限制
MQInput	将文档接收进消息流	该传输节点从 JMS 输入队列接收入站文档。
计算	执行业务逻辑任务，例如更新标题信息	无
MQOutput	从计算节点接收文档并将它发送到 Business Integration Connect	该传输节点的队列名字段（在基本属性中）的值是消息流的输出队列。必须设置 WebSphere MQ 以便于该 JMS 队列发送文档到 Business Integration Connect 的输出队列。要了解更多信息，请参阅第 119 页的『配置通过 HTTP 传输协议传递的出站文档』。

要获得更多的有关如何创建与配置消息流节点的详细信息，请参阅您的 WebSphere Business Integration Message Broker 文档。

第 6 章 与 WebSphere Data Interchange 集成

本章描述了如何将 WebSphere Business Integration Connect 与 WebSphere Data Interchange 集成在一起。

注：要获取用来集成 Business Integration ConnectFor 与后端系统的一般过程的描述，请参阅第 3 页的第 1 章，『规划后端集成』。

本章提供了关于进行文档交换所用过程的说明，并列出了设置这种交换的样本环境的步骤。整章所用的方案类似于 *Integrating WebSphere Data Interchange V3.2 with WebSphere Business Integration Connect V4.2* 教程中出现的那个方案，可以在下面的 Web 站点找到该教程：

www.ibm.com/developerworks/websphere/

该教程还提供了额外的脚本（在配置 WebSphere MQ 一节中）和样本转换映射。通过学习该教程，您可以设置本章中所描述的环境。

学习本章的前提是假定您熟悉 WebSphere Data Interchange 的使用。当您阅读本章时，可参阅 WebSphere Data Interchange 文档以获取更多信息。

简介

WebSphere Data Interchange 对电子数据交换 (EDI) 集成进 WebSphere 业务流程、消息传递以及基于因特网的 B2B 功能。您可在 Business Integration Connect 与 WebSphere Data Interchange 之间通过 JMS 传输协议来交换文档与消息。将文档发送到 WebSphere Data Interchange 时，您必须将封装方式指定为“无”。

注：WebSphere Data Interchange 还提供了其它类型的集成选项类型，例如基于文件的集成。请参阅 WebSphere Data Interchange 文档以了解关于启用通过基于文件的集成进行文档交换的详细信息。

将文档发送到 WebSphere Data Interchange

在 Business Integration Connect 将 EDI 文档发送到 WebSphere Data Interchange 的过程中，会出现以下步骤：

1. 社区参与者发送 EDI 文档到 Business Integration Connect。该文档是通过 HTTP 传输协议以 AS2 封装发送的。Business Integration Connect 将 AS2 封装与 EDI 文档剥离。
2. Business Integration Connect 把 EDI 文档放置到一个队列中。

注：Business Integration Connect 通过检查 EDI 文档的前 3 个字符来确定所使用的协议。然后它根据协议类型确定发送方和接收方的信息。详细信息请参阅第 153 页的『EDI 路由概述』。

3. WebSphere Data Interchange 从队列中读取 EDI 文档。然后执行解包、验证以及转换 EDI 文档的任务。

注：必须针对用户概要文件及期望映射配置 WebSphere Data Interchange。

4. WebSphere Data Interchange 将文档分发到后端系统。如果后端系统是 WebSphere InterChange Server, 则 WebSphere Data Interchange 发送文档到 WebSphere Business Integration Adapter for MQ 以创建业务对象并调用 InterChange Server 中的协作。

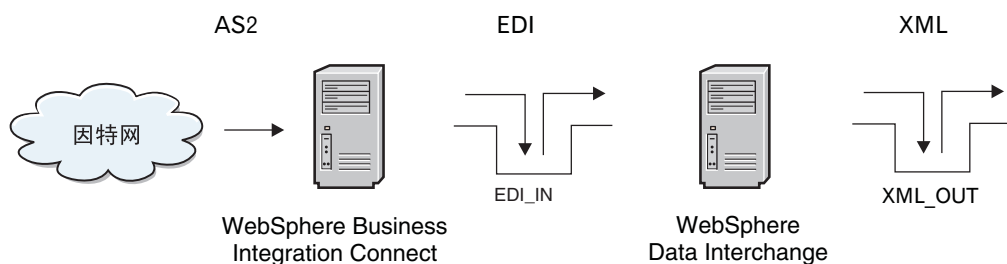


图 25. 来自 Business Integration Connect 的 EDI 文档

在图 25 中, 社区参与者发送 AS2 文档到 Business Integration Connect, 接下来, 将它发送到 WebSphere Data Interchange 端的 EDI_IN 队列。请注意, 必须设置远程队列、传输队列、接收方队列 (在本例中为 EDI_IN) 以及发送方与接收方通道, 以便将发送到 Business Integration Connect 的消息传输到 EDI_IN 队列。WebSphere Data Interchange 服务器获取 EDI 文档, 搜索用户概要文件和映射等等, 把文档转换成 XML, 再把它放入 XML_OUT 队列。

接收来自 WebSphere Data Interchange 的文档

在 Business Integration Connect 接收来自 WebSphere Data Interchange 的 EDI 文档过程中, 会出现以下步骤:

1. WebSphere Data Interchange 将 EDI 文档放置在某个队列中。
2. Business Integration Connect 从该队列读取消息。

注: Business Integration Connect 如第 153 页的『EDI 路由概述』所描述那样, 确定如何路由文档, 。

3. Business Integration Connect 将文档路由给相应的社区参与者。

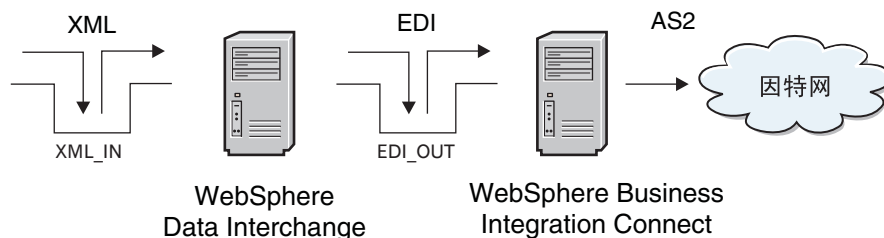


图 26. 将 EDI 文档发送到 WebSphere Business Integration Connect

在图 26 中, XML 文档被放置到 WebSphere Data Interchange 的 XML_IN 队列以进行转换。假定已经执行用户概要文件和映射等。一收到有效的 XML 文档, WebSphere Data Interchange 就把它转换成 EDI 格式并将输出放置在 EDI_OUT 队列 (一个远程队列) 中。还假定设置了 Business Integration Connect 端的传输队列、发送方与接收方通道以及接收方队列。一接收到文档, Business Integration Connect 就把它路由给社区参与者。

本章中使用的样本方案

通过本章，您将了解在两个贸易合作伙伴之间设置 EDI 文档交换的步骤。通过因特网发送 EDI 文档并且使用 AS2（通过 HTTP）作为通信协议。

在该样本中，贸易合作伙伴是 Partner One 和 Partner Two。图 27 说明了这两个合作伙伴的配置。

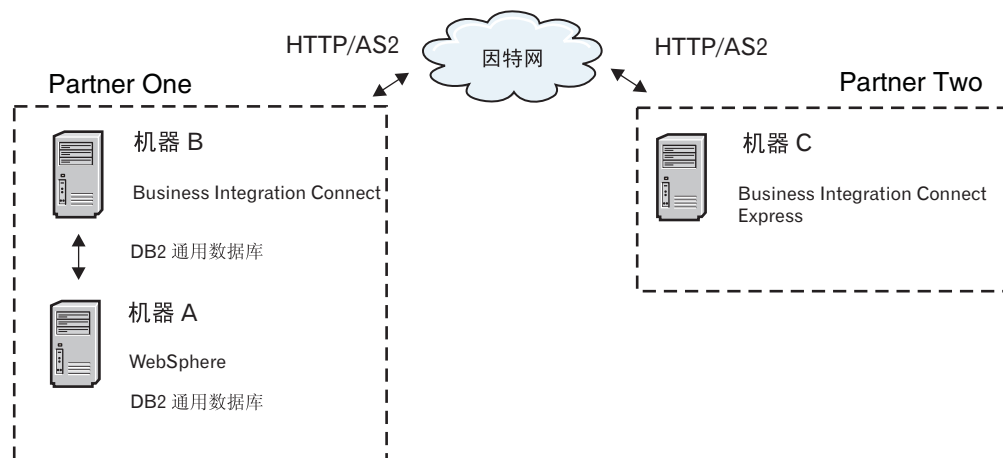


图 27. 样本案例中两个合作伙伴的配置

使用下面的软件来实现该样本案例:

- 在机器 A 上 (Partner One) :
 - 操作系统: Microsoft Windows 2000 Professional
 - 带有 CSD 07 的 WebSphere Data Interchange Server V3.2 (或更高版本)
 - 带有修改包 7 的 WebSphere Data Interchange Client V3.2 (或更高版本)
 - 带有 CSD 04 的 WebSphere MQ V5.3
 - 带有修改包 10 的 IBM DB2 V7.2
- 在机器 B 上 (Partner One) :
 - 操作系统: Red Hat Linux Advanced Server V2.1
 - WebSphere Business Integration Connect Enterprise Edition V4.2.0 (或更高版本)
 - 带有 CSD 04 的 WebSphere MQ V5.3
 - 带有修改包 2 的 IBM DB2 V8.1
- 在机器 C 上 (Partner Two) :
 - 操作系统: Windows 2000 Professional
 - WebSphere Business Integration Connect Express V4.2.0 (或更高版本)

请参阅 *Business Integration Connect 安装指南* 和 *WebSphere Data Interchange 文档* 以获取完整的必备软件列表。

在本例中，partnerOne 操作两台机器。机器 A 既安装了 WebSphere MQ 也安装了 WebSphere Data Interchange Server。机器 B 安装了 WebSphere MQ 与 WebSphere Business Integration Connect Enterprise Edition。机器 B 支持两个贸易合作伙伴之间的通信。

WebSphere Data Interchange 支持与 WebSphere MQ 的集成，从而支持与广泛的企业应用程序和业务流程引擎进行相互操作。WebSphere Business Integration Connect 使用 WebSphere MQ 作为 JMS 提供程序。同样，WebSphere Data Interchange 与 WebSphere Business Integration Connect 之间的集成通过最终传递到 JMS API 客户端的 MQ 消息来实现。

在因特网上，通过使用 AS2 协议用 WebSphere Business Integration Connect 来传递 EDI 事务。

请注意，在本例中，partnerTwo 使用 WebSphere Business Integration Connect- Express 来接收通过 AS2 进行的事务，并且它有自己的 WebSphere Data Interchange 环境可用来处理转换与应答。

学习完本章，您会了解到关于配置此样本方案中所用机器的详细信息。消息流是双向的，因此既包括了发送构件又包含了接收构件。

配置您的消息交换环境

要使得能够在 WebSphere Data Interchange 和 Business Integration Connect 之间进行通信，您需执行下面的设置和配置任务：

- 『配置 WebSphere MQ 通信』
- 第 137 页的 『配置 WebSphere Data Interchange』
- 第 142 页的 『设置 JMS 环境』
- 第 143 页的 『配置 Business Integration Connect Enterprise Edition』

配置 WebSphere MQ 通信

设置环境的第一步是配置 WebSphere MQ 互通信。互通信意味着将消息从一个队列管理器发送到另一个队列管理器。第一步是为 WebSphere Data Interchange 系统与 Business Integration Connect 系统定义队列管理器（及相关对象）。如果您打算双向发送消息，那么请在两个系统上都设置源队列管理器和目标队列管理器。在源队列管理器上，定义发送方通道、远程队列定义以及传输队列。在目标队列管理器上，定义接收方通道和目标队列。

注：请参阅 WebSphere MQ 文档以获取有关定义队列管理器的其它详细信息。

本节向您说明了用来设置样本方案所需的队列管理器和相关对象的值。在此方案中，机器 A 和机器 B 都安装了 WebSphere MQ V5.3。所以第一步是在机器 A 和机器 B 上创建队列管理器，以分别供 WebSphere Data Interchange 和 WebSphere Business Integration Connect Enterprise Edition 使用。

注：应该将您的 WebSphere Data Interchange 队列管理器配置成可以触发使用 WDI Adapter 应用程序的 WebSphere Data Interchange Server。

- 在机器 A 上，您将使用为和 WebSphere Data Interchange 一起使用而定义的队列管理器。在本章的剩余部分，该队列管理器将称为 WDI32_QM。
- 在机器 B 上，您将使用在 WebSphere Business Integration Connect Enterprise Edition 初始安装和配置过程中创建的队列管理器。在本章的剩余部分，该队列管理器将称为 WBIC42_QM。

要使用 WebSphere MQ 将消息从一个队列管理器发送到另一个队列管理器，您需要定义以下对象：

- 在源队列管理器上：
 - 发送方通道
 - 远程队列定义
 - 传输队列
- 在目标队列管理器上：
 - 接收方通道
 - 目标队列

在此样本方案中，机器 A 和机器 B 都充当发送方和接收方。因此，您必须在每台机器上定义一些对象。

表 92 列出了将机器 A 和机器 B 设置为发送方和接收方所要创建的对象。

表 92. 要创建的 WebSphere MQ 对象

WebSphere MQ 对象	机器 A	机器 B
队列管理器	WDI32_QM	WBIC42_QM
发送方通道	TO.WBIC42	TO.WDI32
接收方通道	TO.WDI32	TO.WBIC42
远程队列	EDI_OUT_A	EDI_OUT_B
传输队列	XMITQ_A	XMITQ_B
本地队列	EDI_IN_A	EDI_IN_B
本地队列	XML_IN_A	XML_IN_B
本地队列	XML_OUT_A	XML_OUT_B

图 28 显示了机器 A 与机器 B 之间消息流，从而指出了表 92 中列出的 WebSphere MQ 对象的角色。

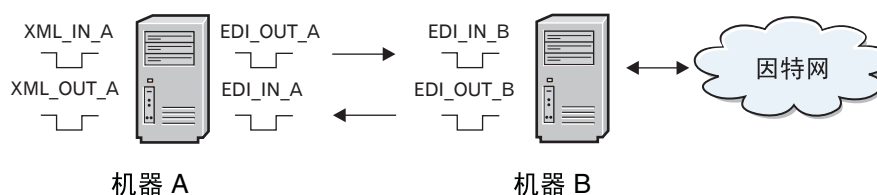


图 28. 机器 A 与机器 B 之间的消息流

根据您的 WebSphere MQ 平台，您可以使用一些不同的方法来定义这些目标。例如，您可以在 Windows 上使用 WebSphere MQ Explorer 来定义对象。

配置 WebSphere Data Interchange

要使 WebSphere Data Interchange 从 WebSphere MQ 队列接收消息并将 EDI 消息写到队列，您必须在 WebSphere Data Interchange 客户机中配置概要文件。通过使用 WebSphere Data Interchange 客户机，您将创建以下概要文件，在接下来的几节中描述了这些概要文件：

- MQ Series 队列概要文件
- 网络概要文件

- 邮箱概要文件
- 服务概要文件

在此样本方案中，WebSphere Data Interchange 接收来自 WebSphere MQ 队列 XML_IN_A 的 XML 消息并将转换结果写入 WebSphere MQ 队列 EDI_OUT_A 中。这称为 XML 到 EDI 转换。WebSphere Business Integration 还将来自 WebSphere Business Integration Connect Enterprise Edition 的 EDI 接收在 WebSphere MQ 队列 EDI_IN_A 上的，并将转换结果写到 XML_OUT_A。

MQSeries 队列概要文件

“MQSeries 队列概要文件”包含有关 WebSphere MQ 消息队列的信息。表 93 显示要为每个概要文件配置的属性。

表 93. “MQSeries 队列概要文件”中的属性

MQ 属性	描述
队列概要文件标识	用来命名概要文件的唯一标识（逻辑名）
完整队列名	WebSphere MQ 队列的实际名称
队列管理器名	WebSphere MQ 队列管理器的实际名称
描述	用来标识概要文件用途的任何字符串
最大长度	WebSphere MQ 中配置的可达到的最长消息
破坏性读取	如果选择这项，那么这些读取操作会导致 WebSphere Data Interchange 在读取 WebSphere MQ 队列中的消息时除去该消息。
同步点控制	选中该属性时，队列消息的读和写都处于同步点控制之下。如果同步点控制有效的话，那么直到 WebSphere Data Interchange 发出同步点，才会对消息队列进行修改。

因为您正在处理 WebSphere MQ 队列，所以对于每一个队列都需要 WebSphere Data Interchange 中有一个“MQSeries 队列概要文件”。总共要创建四个 MQSeries 队列概要文件，与消息流中使用的每个 WebSphere MQ 队列一一对应。从 WebSphere Data Interchange 客户机的设置区域，您将：

1. 为 XML_IN_A 与 EDI_OU_A 各创建一个 MQSeries 队列概要文件。

表 94 列出了您创建的每个 MQSeries 队列概要文件中指定的实际参数。对这里给出的队列使用从 XML 到 EDI 的转换。

表 94. XML_IN_A 与 EDI_OU_A 的 MQSeries 队列概要文件

队列属性	XML_IN_A 的值	EDI_OU_A 的值
队列概要文件标识	XML_IN_A	EDI_OU_A
完整队列名	XML_IN_A	EDI_OUT_A
队列管理器名	WDI32_QM	WDI32_QM
破坏性读取	选中	选中
同步点控制	选中	选中

注：“队列概要文件标识”最多只能是八个字符。因此，EDI_OUT_A 队列的概要文件标识为必须命名为 EDI_OU_A。对 WebSphere Data Interchange 中 WebSphere MQ EDI_OUT_A 队列的所有引用都使用 EDI_OU_A。

2. 创建 EDI_IN_A 与 XML_OU_A 的 MQSeries 队列概要文件。表 95 定义了 EDI 到 XML 的转换中所使用的每个队列的属性。

表 95. 针对 EDI_IN_A 与 XML_OU_A 的 MQSeries 队列概要文件

队列属性	EDI_IN_A 的值	XML_OU_A 的值
队列概要文件标识	EDI_IN_A	XML_OU_A
完整队列名	EDI_IN_A	XML_OUT_A
队列管理器名	WDI32_QM	WDI32_QM
破坏性读取	选中	选中
同步点控制	选中	选中

网络概要文件

网络概要文件为 WebSphere Data Interchange 定义了您与贸易伙伴进行通信所用网络的特征。对于本方案，您将创建并配置与早先创建的 WebSphere MQ 队列进行通讯的网络概要文件。

表 96 显示了要为网络概要文件配置的属性。

表 96. 网络概要文件中的属性

网络属性	描述
网络标识	用来命名概要文件的唯一标识
通信例程	构建网络命令并调用网络程序来处理命令的程序名
网络程序	被通信例程调用以处理请求的程序
网络参数	网络程序需要的参数

对于该样本方案，您将创建并配置与早先创建的 WebSphere MQ 队列进行通讯的网络概要文件（请参阅 第 138 页的『MQSeries 队列概要文件』），如下所示：

1. 创建名为 WBIC_IN 的新网络概要文件。

此网络概要文件在从 XML 到 EDI 方案使用。表 97 列出了为 WBIC_IN 指定的实际参数。

表 97. WBIC_IN 的网络概要文件

网络属性	WBIC_IN 概要文件的值
网络标识	WBIC_IN
通信例程	VANIMQ
网络程序	EDIMQSR
网络参数	SENDMQ=EDI_OU_A RECEIVEMQ=XML_IN_A

2. 创建名为 WBIC_OUT 的第二个网络概要文件。

在转换从 WebSphere Business Integration Connect Enterprise Edition 接收到的 EDI 使用该网络概要文件。第二个网络概要文件是必需的，因为 WebSphere Business Integration Connect Enterprise Edition 将消息放在含 RFH2 头的 WebSphere MQ 队列上。第 140 页的表 98 列出了 WBIC_OUT 的属性。

表 98. WBIC_OUT 的网络概要文件

网络属性	WBIC_OUT 概要文件的值
网络标识	WBIC_OUT
通信例程	VANIMQ
网络程序	EDIRFH2
网络参数	SENDMQ=XML_OU_A RECEIVEMQ=EDI_IN_A

邮箱概要文件

邮箱概要文件中包含了一些信息，WebSphere Data Interchange 需要用这些信息标识组织中接收要转换文档的个人和组。表 99 显示了要为每个邮箱概要文件配置的属性。

表 99. 邮箱概要文件中的属性

邮箱属性	描述
邮箱标识	用来命名概要文件的唯一标识
网络标识	早先创建的网络概要文件的网络标识

您为每个 WebSphere MQ 队列创建邮箱概要文件，以标识组织中的个人与组，如下所示：

1. 为使用每个 WebSphere MQ 队列创建邮箱概要文件。

表 100 列出了每个“邮箱概要文件”中的实际参数。

表 100. XML_IN_A 与 EDI_OU_A 的邮箱概要文件

邮箱属性	XML_IN_A 的值	EDI_OU_A 的值
邮箱标识	XML_IN_A	EDI_OU_A
网络标识	WBIC_IN	WBIC_IN
接收文件	XML_IN_A	EDI_OU_A

2. 再创建一对邮箱。

表 101 列出了每个邮箱的属性。

表 101. EDI_IN_A 与 XML_OU_A 的邮箱概要文件

邮箱属性	EDI_IN_A 的值	XML_OU_A 的值
邮箱标识	EDI_IN_A	XML_OU_A
网络标识	WBIC_OUT	WBIC_OUT
接收文件	EDI_IN_A	XML_OU_A

服务概要文件

“服务概要文件”允许您输入实用程序命令并定义在执行该命令期间将用到的所有文件。

对于本样本方案，您将采取下列步骤：

1. 为 XML_IN_A 创建新的服务概要文件。如下所示定义常规选项卡下的属性：
 - 持续命令链: **On Success**
 - PERFORM 命令:


```
PERFORM TRANSFORM WHERE INFILE(XML_IN_A) SYNTAX(X)
OUTTYPE(MQ)OUTFILE(EDI_OU_A)
```

表 102 列出了公共文件属性。

表 102. XML_IN_A 的公共文件

公共文件属性	值
跟踪文件	..\trk\xml_in.trk
异常文件	..\xex\xml_in.xex
工作文件	..\wrk\xml_in.wrk
报告文件	..\rpt\xml_in.rpt
查询文件	..\qry\xml_in.qry

2. 在输出文件选项卡中输入下列详细信息:

- 命令名: **EDI_OU_A**
- 系统文件名: **..\edi\edi_out.txt**

注: 因为字符长度限制的原因, 所以使用 EDI_OU_A 而不是 EDI_OUT_A。

3. 为 EDI_IN_A 再创建一个服务概要文件。如下所示定义常规选项卡下的属性:

- 持续命令链: **On Success**
- PERFORM 命令:

```
PERFORM TRANSFORM WHERE INFILE(XML_IN_A) SYNTAX(X) OUTTYPE(MQ) OUTFILE(XML_OU_A)
```

表 103 列出了公共文件属性。

表 103. EDI_IN_A 的公共文件

公共文件属性	值
跟踪文件	..\trk\edi_in.trk
异常文件	..\xex\edi_in.xex
工作文件	..\wrk\edi_in.wrk
报告文件	..\rpt\edi_in.rpt
查询文件	..\qry\edi_in.qry

4. 在输出文件选项卡中输入下列详细信息:

- 命令名: **XML_OU_A**
- 系统文件名: **..\xml\xml_out.txt**

注: 因为字符长度限制的原因, 所以使用 XML_OU_A 而不是 XML_OUT_A。带 CSD 10 的 WebSphere Interchange Server 已消除了该限制。

导入并编译数据转换映射

如前一节所描述的那样, 当创建了概要文件之后, 可以导入转换数据所需的任何映射。然后编译转换映射并为每个映射设置规则。您使用 WebSphere Data Interchange 客户机来执行这些任务。请参阅 WebSphere Data Interchange 文档以获取信息。

设置 JMS 环境

在本章前面提到过的，WebSphere Business Integration Connect Enterprise Edition 可以使用 Java Message Service (JMS) 的 WebSphere MQ 实现来与 WebSphere Data Interchange 进行集成。

注：另外，可以使用 LDAP 或 WebSphere Application Server 作为 JNDI 提供程序。

本节概括了在机器 B 上创建 JMS 环境所涉及到的步骤：

- 『配置 JMSAdmin』
- 『创建 JMS 对象』

WebSphere MQ 的 Java 类和 WebSphere MQ 的 JMS 类是内置在 WebSphere MQ for Windows V5.3 中的。

配置 JMSAdmin

使用 WebSphere MQ 内可用的 JMSAdmin 工具在 JNDI 中来创建 JMS 对象。要了解有关如何创建缺省配置文件 JMSAdmin.config 的信息，请参阅第 159 页的『用 WebSphere MQ 配置 JMS 协议』。

创建用于本教程的 JMS 对象：

1. 为使用基于文件的 JNDI 提供程序，您要确保 JMSAdmin.config 文件包含如下所示的行：

```
INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory
PROVIDER_URL=file:/opt/mqm/java/JNDI
```

2. 如果 JNDI 目录尚未存在，则在下列目录中创建 JNDI 目录：

```
/opt/mqm/java/bin
```

在调用 JMSAdmin 工具之前，请确保您的 CLASSPATH 包含下列各项：

```
/opt/mqm/java/lib/jms.jar
/opt/mqm/java/lib/com.ibm.mq.jar
/opt/mqm/java/lib/com.ibm.mqjms.jar
/opt/mqm/java/lib/jta.jar
/opt/mqm/java/lib/connector.jar
/opt/mqm/java/lib/jndi.jar
/opt/mqm/java/lib/providerutil.jar
/opt/mqm/java/lib/fscontext.jar
```

注：上面这些与 Linux 相关的项假定您使用的是基于文件的 JNDI。

创建 JMS 对象

要创建所需的 JMS 对象，请使用 JMSAdmin 工具。对于本样本方案，您将：

1. 定义新的上下文：

```
DEF CTX(WdiJms)
```

2. 更改至新的上下文：

```
CHG CTX(WdiJms)
```

3. 定义队列连接工厂：

```
DEF QCF(WBIC42_QM_QCF)TRAN(CLIENT)HOST(IP_MACHINE_B)
PORT(9999)CHAN(java.channel)QMANAGER(WBIC42_QM)
```

4. 定义 EDI_IN_B 队列：

```

DEF Q(EDI_IN_B)QMANAGER(WBIC42_QM)QUEUE(EDI_IN_B)
5. 定义 EDI_OUT_B 队列:
DEF Q(EDI_OUT_B)QMANAGER(WBIC42_QM)QUEUE(EDI_OUT_B)
6. 结束 JMSAdmin 会话
END

```

配置 Business Integration Connect Enterprise Edition

WebSphere Business Integration Connect 是全异的社区参与者与内部过程之间的通信层。在设置 Business Integration Connect 以处理 EDI 文档时，您可以将它配置成：

- 将 EDI 文档发送到 WebSphere Data Interchange 和从 WebSphere Data Interchange 接收 EDI 文档
- 使用 AS2 与外部贸易合作伙伴进行 EDI 事务通信

中心配置指南提供了完整的有关如何配置 WebSphere Business Integration Connect Enterprise Edition 和 Advanced Edition 的信息。本节提供了一个配置本样本方案中所描述的 WebSphere Business Integration Connect Enterprise Edition 的示例。它描述了下列步骤：

1. 『创建参与者』
2. 第 145 页的『定义 B2B 功能』
3. 第 146 页的『创建网关』
4. 第 148 页的『定义文档流定义和交互』
5. 第 148 页的『创建参与者连接』
6. 第 149 页的『创建目标』

注：要获取有关如何配置 WebSphere Business Integration Connect - Express 的信息，请参阅第 150 页的『配置 Business Integration Connect - Express』。

创建参与者

参与者概要文件将公司标识到系统。您在 WebSphere Business Integration Connect Enterprise Edition 社区控制台中创建 Partner One 与 Partner Two 的参与者。

创建 *Partner One* 的参与者： 创建一个参与者概要文件以表示机器 A 和机器 B，这两个机器是 Partner One 所拥有的两个系统。

要创建该参与者概要文件，请采取以下步骤：

1. 打开 WebSphere Business Integration Connect 社区控制台
2. 以中心操作者身份登录。
3. 确认从“帐户管理”菜单中选中了“概要文件”。
4. 单击**创建**并输入下表 104 中所列的详细信息。

表 104. *Partner One* 的参与者属性

字段名	值
参与者登录名	partnerOne
参与者名	Partner One
参与者类型	社区管理者
状态	已启用

表 104. Partner One 的参与者属性 (续)

字段名	值
供应商类型	其它
Web 站点	http://IP_MACHINE_A
	IP_MACHINE_A 是机器 A 的因特网协议 (IP) 地址
企业标识类型	自由格式
企业标识的标识符	123456789
IP 地址网关类型	生产
IP 地址	IP_MACHINE_A
	IP_MACHINE_A 是机器 A 的因特网协议 (IP) 地址

注: 要创建“企业标识”类型和企业标识的标识符, 首先请单击企业标识下的**新建**按钮。企业标识必须是唯一的。类似地, 要创建与 IP 地址相关的详细信息, 请单击在“IP 地址”标题下的**新建**按钮。

5. 单击**保存**。

WebSphere Business Integration Connect Enterprise Edition 使用企业标识符 (在第 143 页的表 104 中定义的) 以标识文档的发送方或接收方。当接收到 ANSI X12 EDI 事务时, 读取交换发送方数据与接收方数据, 以确定事务的源与目标。

要点: 请记住 Partner One 的管理员密码, 因为您以后将会用到它。当您以 Partner One 身份登录到社区控制台时, 将会要求您输入该密码, 然后才能进行更改。

创建 Partner Two 的参与者: 接下来创建用以表示 Partner Two 的社区参与者。要创建该参与者, 请采取以下步骤:

1. 单击主菜单上的**帐户管理**, 然后单击水平导航栏上的**概要文件**。
2. 请单击**创建**。
3. 输入下表 105 中所列的值。

表 105. Partner Two 的参与者属性

字段名	值
参与者登录名	partnerTwo
参与者名	Partner Two
参与者类型	社区参与者
状态	已启用
供应商类型	其它
Web 站点	http://IP_MACHINE_C
	IP_MACHINE_C 是机器 C 的因特网协议 (IP) 地址
企业标识类型	自由格式
企业标识的标识符	987654321
IP 地址网关类型	生产

表 105. Partner Two 的参与者属性 (续)

字段名	值
IP 地址	IP_MACHINE_C IP_MACHINE_C 是机器 C 的因特网协议 (IP) 地址


4. 单击保存。


要点: 请记住 Partner Two 的管理员密码, 因为您以后将会用到它。当您以 Partner Two 身份登录到社区控制台时, 将会要求您输入该密码, 然后才能进行更改。

定义 B2B 功能






通过社区控制台为 WebSphere Business Integration Connect Enterprise Edition 中的每个参与者定义 B2B 功能。为参与者定义好 B2B 功能之后, 您可以定义用于支持参与者之间特定业务协作类型的有效文档流定义。

为 Partner One 设置 B2B 功能: 要为 Partner One 设置 B2B 功能, 请采取以下步骤:


1. 单击主菜单上的**帐户管理**, 然后单击水平导航栏上的**概要文件**。
2. 单击**搜索**, 显示系统中已定义的所有参与者的列表。
3. 单击 **Partner One** 旁边的  图标, 然后单击 **B2B 功能**。

通过单击  图标, 把 B2B 功能设置为活动状态。对于本样本, 您只要配置实现方案所需的 B2B 功能。






要将 Partner One 的源封装方式和目标封装方式设置成“无”, 您将:

1. 单击与“包: 无”对应的“设置源”底下的  图标来启用它。对“设置目标”重复该步骤。
2. 请单击  图标进行向下钻取。
3. 请单击与“协议: EDI-X12 (全部)”对应的源和目标的  图标。
4. 单击  图标。
5. 请单击与“文档流: 全部”对应的源和目标  图标。

为 Partner Two 设置 B2B 功能: 要为 Partner Two 设置 B2B 功能, 请采取以下步骤:

1. 单击主菜单上的**帐户管理**, 然后单击水平导航栏上的**概要文件**。
2. 单击**搜索**, 显示系统中已定义的所有参与者的列表。
3. 单击 **Partner Two** 旁边的  图标, 然后请单击 **B2B 功能**。

要将 Partner Two 的源封装方式和目标封装方式设置成“无”, 请采取下列步骤:

1. 单击与“包：无”对应的“设置源”下的  图标来启用它。对“设置目标”重复该步骤。
2. 请单击  图标进行向下钻取。
3. 请单击与“协议：EDI-X12（全部）”对应的源和目标的  图标。
4. 单击  图标。
5. 请单击与“文档流：全部”对应的源和目标  图标。

接下来，为 Partner Two 更新 AS 定义，以确保发送到 Partner Two 的 AS2 的消息处置通知（MDN）返回到正确的地址，如下所示：

1. 单击编辑图标 ()。
2. 输入 AS MDN 电子邮件地址。

该地址用于接收 AS1 的 MDN。

3. 输入 AS MDN HTTP URL:

`http://IP_MACHINE_B:PORT/bcgreceiver/submit`

注：为 AS2 定义的 URL 所使用参数与本章稍后为 AS2 目标所定义的参数相同。

创建网关

Business Integration Connect 中的网关定义了一个网络点，该点充当到另一个网络的入口。网关包含了告诉 WebSphere Business Integration Connect 如何将文档传递到企业应用集成（EAI）层的信息。

为 Partner One 创建网关： Partner Two 使用 AS2 将 EDI 文档发送到 Partner One。Partner One 的网关用于将通过 AS2 接收到的 EDI 文档发送到 JMS 队列，最后将该文档发送到 WebSphere Data Interchange 以进行转换。

要为 Partner One 创建新的网关，请采取以下步骤：


1. 单击主菜单上的帐户管理，然后单击水平导航栏上的概要文件。
2. 单击搜索。
3. 通过单击  图标来选择 Partner One，然后选择网关。
4. 单击创建，为 Partner One 创建新网关。
5. 为这个新网关输入表 106 中所示的值。

表 106. Partner One 网关的属性

字段名	值
网关名	JMStoPartnerOne
传输	JMS
目标 URI	file:///opt/mqm/java/JNDI/WdiJms
JMS 工厂名	WBIC42_QM_QCF
JMS 消息类	TextMessage

表 106. Partner One 网关的属性 (续)

字段名	值
JMS 消息类型	TextMessage
JMS 队列名	EDI_OUT_B
JMS JNDI 工厂名	com.sun.jndi.fscontext.RefFSContextFactory

6. 单击保存。

使 JMStoPartnerOne 成为 Partner One 的缺省网关，如下所示：

1. 单击查看缺省网关。
2. 从生产列表中选择 **JMS2toPartnerOne**。
3. 单击保存。

注：只能为社区管理者(在该样本方案中为 Partner One) 定义 JMS 网关。

为 Partner Two 创建网关： Partner One 通过 JMS 队列将 EDI 文档发送到 WebSphere Business Integration Connect Enterprise Edition。Partner Two 的网关用于通过 AS2 将接收到的 EDI 文档发送到 Partner Two。

要为 Partner Two 创建新的网关，请采取以下步骤：


1. 单击主菜单上的**帐户管理**，然后单击水平导航栏上的**概要文件**。
2. 单击**搜索**。
3. 通过单击  图标来选择 Partner Two，然后选择**网关**。
4. 单击**创建**，为 Partner Two 创建新网关。
5. 为该网关输入表 107 中所示的值。

表 107. Partner Two 网关的属性

网关名	AS2toPartnerTwo
传输	HTTP/1.1
目标 URI	http://IP_MACHINE_C/input/AS2
用户名	partnerOne
密码	partnerOne

6. 单击保存。

注：上面输入的用户名和密码指的是 WebSphere Business Integration Connect - Express 中定义的 HTTP 入站参与者映射方法。

要获取在 WebSphere Business Integration Connect - Express 中设置这些属性的示例，请参阅第 143 页的『配置 Business Integration Connect Enterprise Edition』。

注意 AS2toPartnerTwo 显示为“联机”，其状态为**已启用**。

使 AS2toPartnerTwo 成为 PartnerTwo 的缺省网关，采取以下步骤：

1. 单击查看缺省网关。
2. 从生产列表中选择 **AS2toPartnerTwo**。
3. 单击保存。

定义文档流定义和交互

文档流定义是“元信息”的集合，它定义了参与者的文档处理能力。要使系统能处理业务文档，必须将两个或多个文档流定义链接在一起以创建交互。

要创建 Partner One 与 Partner Two 之间的文档流定义与有效交互，请采取以下步骤：

1. 单击主菜单上的**中心管理**，然后单击水平导航栏上的**文档流定义**。
2. 单击**管理交互**，然后单击**创建有效交互**。
3. 从“源”列中选择：
 - a. 包：**无**
 - b. 协议：**EDI-X12**
 - c. 文档流：**全部**
4. 从“目标”列中选择：
 - a. 包：**AS**
 - b. 协议：**EDI-X12**
 - c. 文档流：**全部**
5. 将操作设置为**通过**。
6. 单击**保存**。
7. 再次单击**创建有效交互**。
8. 从“源”列中选择：
 - a. 包：**AS**
 - b. 协议：**EDI-X12**
 - c. 文档流：**全部**
9. 从“目标列”中选择：
 - a. 包：**无**
 - b. 协议：**EDI-X12**
 - c. 文档流：**全部**
10. 将操作设置为**通过**。
11. 单击**保存**。

创建参与者连接

参与者连接是一种机制，它使系统能够处理文档，并使系统能在社区管理者及其各个参与者之间路由文档。连接包含了正确交换每个文档流所必需的信息。

要在 Partner One 和 Partner Two 之间创建参与者连接，请采取以下步骤：

1. 单击主菜单上的**帐户管理**，然后单击水平导航栏上的**参与者连接**。
2. 从源列表中，选择 **Partner One**。
3. 从目标列表中，选择 **Partner Two**。
4. 单击**搜索**。
5. 通过单击**激活按钮**来激活下面所示的参与者连接。这会显示如第 149 页的表 108 所示的 B2B 功能。

表 108. 激活 Partner One 到 Partner Two 的参与者连接

文档流类型	源	目标
包	无 (N/A)	AS (N/A)
协议	EDI-X12 (ALL)	EDI-X12 (ALL)
文档流	全部 (ALL)	全部 (ALL)

要创建以 Partner Two 为源而以 Partner One 为目标的参与者连接，请采取以下步骤：

1. 单击主菜单上的**帐户管理**，然后单击水平导航栏上的**参与者连接**。
2. 从源列表中选择 **Partner Two**。
3. 从目标列表中选择 **Partner One**。
4. 单击**搜索**。
5. 用表 109 中显示的详细信息激活连接：

表 109. 激活 Partner Two 到 Partner One 的参与者连接

文档流类型	源	目标
包	AS (N/A)	无 (N/A)
协议	EDI-X12 (ALL)	EDI-X12 (ALL)
文档流	全部 (ALL)	全部 (ALL)

创建目标

“目标列表”屏幕提供了位置信息，该位置信息使 Business Integration Connect 的文档管理器能够根据入站文档的传输类型从适当的系统位置访问文档。您可以根据传输类型创建单独的目标配置。文档管理器然后可以轮询多个 Web、FTP 以及 POP 邮件服务器的文档资源库位置（包括内部目录和 JMS 队列）以获取入站文档。

在文档管理器根据预定义的目标从该位置检索文档后，路由基础结构可以根据通道配置处理该文档。

要从 WebSphere Data Interchange 接收 EDI 事务，请通过执行下列步骤来创建新的 JMS 目标：

1. 单击顶级菜单中的**中心管理**。
2. 单击二级菜单中的**目标**，然后单击**创建**。
3. 如表 110 所示对属性进行赋值。

表 110. JMS 接收方的目标属性

目标属性	值
目标名	WdiJmsListener
传输	JMS
网关类型	生产
JMS 提供程序 URL	file:///opt/mqm/java/JNDI/WdiJms
JMS 队列名	EDI_IN_B
JMS 工厂名	WBIC42_QM_QCF
JNDI 工厂名	com.sun.jndi.fscontext.RefFSContextFactory

还需要另一个目标来从 Partner Two 通过 AS2 来接收 EDI。请采取下列步骤以创建该目标:

1. 单击顶级菜单中的**中心管理**。
2. 单击二级菜单中的**目标**，然后单击**创建**。
3. 根据表 111 对属性进行赋值:

表 111. AS2 接收方的目标属性

目标名	WbicAS2Listener
传输	HTTP/S
网关类型	生产
URI	/bcgreceiver/submit 注: HTTP/S 接收方的 URI 必须总是以 /bcgreceiver 开始

4. 单击**保存**。

配置 Business Integration Connect - Express

该节为您提供配置社区参与者环境的步骤。在本例中，该环境是用 WebSphere Business Integration Connect - Express 系统处理。在本章所提供的样本方案中，partnerTwo 使用 WebSphere Business Integration Connect - Express 通过 HTTP AS2 收发 EDI。

要通过 HTTP AS2 成功接收 EDI，请采取下列步骤:

1. 『配置“我的概要文件”』
2. 第 151 页的『创建 Partner One 的参与者』
3. 第 151 页的『配置 Partner One 参与者』

配置“我的概要文件”

作为第一步，您必须在 WebSphere Business Integration Connect - Express 中创建 Partner Two 的概要文件。要创建 Partner Two 的概要文件，请采取以下步骤:

1. 单击主菜单中的**配置**。
2. 单击水平导航栏上的**我的概要文件**。
3. 输入表 112 中概括的详细信息:

表 112. “我的概要文件”详细信息

接收方地址非安全域	IP_MACHINE_C
接收方地址非安全端口	80
AS2 发送方标识	987654321
企业标识类型	DUNS

IP_MACHINE_C 是机器 C 的因特网协议 (IP) 地址，WebSphere Business Integration Connect - Express 运行在该机器上

其中 80 是在安装过程中分配供 WebSphere Business Integration Connect - Express 使用的端口。

表 112. “我的概要文件” 详细信息 (续)

企业标识	987654321
------	-----------

4. 单击保存。

创建 Partner One 的参与者

必须将 Partner One 作为参与者标识到 WebSphere Business Integration Connect - Express。要把 Partner One 创建为参与者，请采取下列步骤：

1. 单击主菜单中的**配置**。
2. 单击水平导航栏上的**参与者**。
3. 请单击**创建参与者**按钮。
4. 赋予下列值：
 - a. 参与者名: **partnerOne**
 - b. AS2 参与者标识: **123456789**
5. 单击**保存**。

从“管理参与者”视图，您可以看到 partnerOne 的详细信息。

配置 Partner One 参与者

一旦存在 Partner One 的参与者，您必须配置 Partner One 以支持 AS2 和 HTTP。该配置标识了 WebSphere Business Integration Connect - Express 将 HTTP 与 AS2 发送到 partnerOne 以及从 partnerOne 接收 HTTP 和 AS2 所需的参数。

配置 partnerOne 以支持 HTTP 与 AS2，请执行下列步骤：

1. 单击主菜单中的**配置**。
2. 单击水平导航栏上的 **AS2**。
3. 从**选中参与者**列表中选择 **partnerOne**，然后单击**编辑**。
4. 将 partnerOne 的“出站目的地地址”定义成：

```
http://IP_MACHINE_B:7080/bcgreceiver/submit
```

IP_MACHINE_B 是机器 B 的 IP 地址。

5. 单击**保存**。
6. 单击水平导航栏上的 **HTTP**。（**partnerOne** 仍应显示在“选中参与者”列表中。）
7. 单击**编辑**。
8. 设置入站用户名和密码：

用户名: **partnerOne**

密码: **partnerOne**

还记得吗？先前在机器 B 上的 WebSphere Business Integration Connect Enterprise Edition 中为 Partner Two 创建缺省网关的样本步骤中曾引用过这些信息。

9. 将“出站目的地地址”设置成：

```
http://IP_MACHINE_B:7080/bcgreceiver/submit
```
10. 单击**保存**。

要点: 在 WebSphere Business Integration Connect - Express 中进行这些更改之后，注销控制台并终止网关。重新启动网关与控制台，使所有更改生效。

结束语

本章描述了 Business Integration Connect 与 WebSphere Data Interchange 进行交互的过程。它还提供了设置第 135 页的『本章中使用的样本方案』所述样本方案的过程。

正如在本章开头所述，您可以按照 *Integrating WebSphere Data Interchange V3.2 with WebSphere Business Integration Connect V4.2* 教程实际创建一个样本配置。该教程提供了样本脚本和映射，以帮助您配置环境，并且向您展示了如何测试样本交换。要访问教程，请转至：

www.ibm.com/developerworks/websphere/

然后搜索该教程的标题。

第 7 章 路由 EDI 文档

本节描述了 Business Integration Connect 确定它所收发的电子数据交换 (EDI) 文档路由信息的过程。它描述了:

- 这个处理活动的常规流程 (请参阅 EDI 路由概述)
- 当指定了 AS 封装之后所需的其它处理 (请参阅 第 154 页的『AS 封装的特殊考虑事项』)

有关在路由 EDI 文档时如何使用基于文件集成的其它信息, 可在 第 19 页的『Enterprise Edition 和 Advanced Edition 的文件系统协议』中找到。

EDI 路由概述

一个 EDI 文档包含了有关文档的发送方和接受方的信息。当 Business Integration Connect 路由 EDI 文档时使用这些信息。常规流程如下:

1. Business Integration Connect 通过检查文档的前 3 个字符来确定所使用的协议。表 113 显示了与每个代码关联的文档类型协议。

表 113. EDI 代码及关联文档类型和协议

代码	文档类型	文档类型协议	出站内容类型:
ISA	X12	EDI-X12	应用程序/EDI-X12
GS	X12	EDI-X12	应用程序/EDI-X12
UNB	Edifact	EDI-EDIFACT	应用程序/EDIFACT
UNA	Edifact	EDI-EDIFACT	应用程序/EDIFACT
ICS	ICS	EDI-X12	应用程序/EDI-X12
STX	UNTDI	EDI-Consent	应用程序/edi-consent
BG	UCS	EDI-Consent	应用程序/edi-consent

2. Business Integration Connect 根据特定文档类型的元素及其位置从 EDI 文档中抽取发送方的信息, 具体描述如表 114 所示。

表 114. EDI 代码与发送方和接受方的位置信息

代码	发送限定符	发送标识	接收限定符	接收标识
ISA	元素 105 在位置 5 上	元素 107 在位置 6 上	元素 105 在位置 7 上	元素 106 在位置 8 上
GS	N/A	元素 142 在位置 2 上	N/A	元素 124 在位置 3 上
UNB	复合元素 S002 中位置 2 上的子元素	复合元素 S002 中位置 2 上的子元素	复合元素 S003 中位置 2 上的子元素	复合元素 S003 中位置 1 上的子元素
UNA	0007, S002 位于 UNB 段的位置 20 (第二个复合) 上	0004, S002 位于 UNB 段的位置 20 (第二个复合) 上	0007, S003 位于 UNB 段的位置 30 (第三个复合) 上	0010, S003 位于 UNB 段的位置 30 (第三个复合) 上
ICS	元素 X05 在位置 4 上	元素 X06 在位置 5 上	元素 X05 在位置 6 上	元素 X08 在位置 7 上
STX	元素 FROM1 在位置 3 上	元素 FROM2 在位置 3 上	元素 UNT1 在位置 4 上	元素 UNT2 在位置 4 上

表 114. EDI 代码与发送方和接受方的位置信息 (续)

代码	发送限定符	发送标识	接收限定符	接收标识
BG	N/A	元素 BG03 在位置 3 上	N/A	元素 BG04 在位置 4 上
UCS	N/A	元素 142 在位置 3 上	N/A	元素 124 在位置 4 上

3. Business Integration Connect 根据 EDI 文档的发送方标识和限定符来确定发送方标识。

注: 某些 EDI 包络 (例如, GS) ↓ 没有限定符概念。这种情况下, Business Integration Connect 只使用标识。

4. Business Integration Connect 用连字符 (-) 将限定符和标识连接在一起, 以查找 Business Integration Connect 概要资料库中的发送方标识。例如, 在发送方的 EDI 消息中, 限定符是 AB, 标识是 1234567, Business Integration Connect 期望在概要资料库中找到一个标识为 AB-1234567 的社区参与者。如果 Business Integration Connect 没有发现该标识, 则不路由 EDI 文档。
5. 为了查找接收方合作伙伴, Business Integration Connect 根据 EDI 消息确定接收方的限定符和标识。
6. Business Integration Connect 用连字符 (-) 将限定符和标识连接在一起, 以在概要资料库中查找接收方标识。
7. Business Integration Connect 将文档路由给预期的接收方。

AS 封装的特殊考虑事项

当文档的封装指定为 AS 时, Business Integration Connect 执行一些额外的处理活动。

路由进站文档

当收到社区参与者的 EDI 文档后:

1. Business Integration Connect 首先检查 AS1 或 AS2 的头信息。尤其是检查发送方和接收方的信息, 以确定是否与有效的社区参与者的标识匹配。
 - 对于 AS1, 它使用主题头字段, 其格式 “ToID;FromID”。
 - 对于 AS2, 使用 “AS - 自” 和 “AS - 至”。

如果头字段中的值与有效的标识不匹配, Business Integration Connect 不路由文档。

2. 然后 Business Integration Connect 执行第 153 页的『EDI 路由概述』中描述的步骤。

路由出站文档

当收到来自后端系统的 EDI 文档时, Business Integration Connect 确定是否为源封装 (无) 和目标封装 (AS) 指定 AS BusinessID 属性。

- 如果指定了 AS BusinessId 属性, Business Integration Connect 用这些信息来在 AS1 或 AS2 的头中生成 “发送方” 和 “接受方” 标识。
- 如果未指定该属性, Business Integration Connect 确定文档的协议, 抽取发送方和接收方的信息并连接结果 (如第 153 页的『EDI 路由概述』中描述的那样) 然后填充头信息。

在参与者概要文件中设置标识

因为 Business Integration Connect 使用 AS1 或 AS2 的头信息，以及从 EDI 文档中得到的信息，所以同一参与者的标识的格式可能不同。例如，发送方 AS 的头信息可能是 123456789，而从 EDI 中得到的信息可能是 AB-12345678。

请确保您已经列出社区参与者概要文件中的所有标识。请参阅[管理员指南](#)以获取信息。

第 4 部分 附录

附录. 用 WebSphere MQ 配置 JMS 协议

本附录描述配置 V5.3 IBM WebSphere MQ 产品中 JMS 传输协议的执行步骤。您必须配置 JMS 协议，因为 WebSphere MQ 缺省情况下不配置 JMS。这些步骤使用基于文件的 JNDI 来配置 JMS 以与 WebSphere Business Integration Connect 和后端系统一起使用。

注：本部分中的指示信息假设 WebSphere MQ 队列管理器是本地的；也就是说，该队列管理器驻留在与 WebSphere Business Integration Connect 相同的机器内。如果您的队列管理器是远程的（其它某台机器上），请参考 WebSphere MQ 文档以获取有关如何设置您队列的信息。

要配置 JMS 传输协议，通过后端系统来发送和接收文档，您需要：

1. 配置 JMS 配置目录，该目录包含 JMS 队列接收的文件。
2. 创建 JMS 队列和通道，它们由队列管理器托管。
3. 为 WebSphere MQ 5.3 创建 JMS 绑定文件。
4. 创建 JMS 目标以使用 JMS 进站队列。
5. 创建 JMS 网关以使用 JMS 出站队列。

配置 JMS 配置目录

JMS 配置目录将包含一些文件，这些文件从由 WebSphere MQ 队列管理器托管的 JMS 队列进入服务器。要创建 JMS 绑定文件，请您使用 JMSAdmin 应用程序。然而，运行该应用程序以前，您必须定制它的配置文件，以符合您的 JMS 环境。

注：WebSphere MQ 文档提供了如何创建 JMS 绑定文件的全面描述。本节概述了该过程。

要配置 JMS 配置目录，请采用以下步骤：

1. 在 Business Integration Connect 所驻留的那台机器上的某一位置创建 JMS 配置目录。
2. 打开 JMSAdmin 应用程序的配置文件“JMSAdmin.config”以供编辑。您必须定制该文件以给 JMSAdmin 应用程序提供关于您的 JMS 配置的信息。该文件位于下列目录：

```
WebSphereMQ_Root\java\bin
```

*WebSphereMQ_Root*是 WebSphere MQ 的产品目录。

3. 通过插入一个英磅字符（#）作为各行的首字符来注释掉下列行：

```
INITIAL_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory  
PROVIDER_URL=ldap://polaris/o=ibm,c=us
```

4. 通过删除作为行首字符的英磅字符（#）来除去下列行的注释：

```
#INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory  
#PROVIDER_URL=fiTe://C:/JNDI-Directory
```

5. 把 PROVIDER_URL 变量的路径更改成 JMS 配置目录。这就是您在上面的步骤 1 中所创建的目录。

注：该目录必须存在并且您的用户帐户必须拥有对该文件夹的“写”许可权。

6. 保存 JMSAdmin.config 文件。

假设您创建如下 JMS 配置目录：

C:/filesender/config

对于 JMS 配置目录，图 29 显示文件 JMSAdmin.config 中被修改的行。

```
#INITIAL_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory
INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory
#
# The following line specifies the URL of the service provider's initial
# context. It currently refers to an LDAP root center. Examples of a
# file system URL and WebSphere's JNDI namespace are also shown, commented
# out
#PROVIDER_URL=ldap://polaris/o=ibm,c=us
PROVIDER_URL=file://C:/filesender/config
```

图 29. 文件 JMSAdmin.config 的样本行

创建 JMS 队列

JMS 传输协议使用 JMS 队列来保存在 Business Integration Connect 和后端系统之间传递的消息。就 WebSphere MQ V5.3 而言，创建 JMS 队列包括下面的步骤：

1. 『创建 MQ 队列管理器』
2. 『创建 MQ 通道和传输队列』
3. 第 162 页的『创建 MQ JMS 本地队列』

创建 MQ 队列管理器

如果还没有为 Business Integration Connect 和后端系统定义队列管理器，您必须在创建 MQ JMS 队列之前创建 MQ 队列管理器。如果您正在做 JMS 集成，则需要队列管理器。你必须决定是安装新的队列管理器还是使用一个现成的。现有的队列管理器可以是 Business Integration Connect 的队列管理器、后端系统队列管理器或其它现有的队列管理器。参阅 WebSphere MQ 文档获取关于如何创建队列管理器的指示信息。

注：如果您的后端系统和 Business Integration Connect 驻留在不同的机器，队列管理器可以驻留在任一机器。Business Integration Connect 和后端系统在访问队列管理器时指定机器的主机名。

本章中的样本假设 WebSphere MQ 队列管理器存在并且具有以下名称：

bcg.queue.manager

创建 MQ 通道和传输队列

一旦队列管理器存在，请确保下面的对象已经创建：

- 传输队列
- 远程队列
- 接收器队列

- 发送方通道
- 接收器通道

要创建 MQ 通道和传输队列，使用 WebSphere MQ 命令环境 “runmqsc”，它由 WebSphere MQ 提供。该工具提供命令行界面，您可以输入队列管理命令。

注：创建这些 MQ 对象的方法依赖于您使用的平台。请参阅 WebSphere MQ 文档以获取关于创建这些对象的指示信息。

要使用 runmqsc 工具来创建您的 MQ 通道和传输队列，请遵循以下步骤：

1. 打开命令提示符并转至下列目录：

```
WebSphereMQ_Root\java\bin
```

其中 *WebSphereMQ_Root* 是 WebSphere MQ 的安装目录。

2. 输入下面命令以启动 WebSphere MQ 命令环境：

```
runmqsc queueManager
```

其中 *queueManager* 是您的 WebSphere MQ 队列管理器的名称。

注：您必须以 mqm 用户的身份登录以使用 runmqsc 工具。

3. 在 runmqsc 的命令行，您可以输入下面的命令：

- a. 定义 JMS 传输队列：

```
define qlocal(transQueueName) usage(xmitq) put(enabled)
get(enabled)
```

其中 *transQueueName* 是期望的 JMS 传输队列名称。

- b. 定义发送方通道：

```
define channel(sndrChannelName) chltype(sdr) xmitq(transQueueName)
conname('remote m/c ip')
stop channel(sndrChannelName)
```

其中 *sndrChannelName* 是期望的 JMS 发送方通道名称，而 *transQueueName* 是步骤 3a 中创建的 JMS 传输队列的名称。

- c. 定义接收器通道：

```
define channel(rcvrChannelName) chltype(rcvr) xmitq(transQueueName)
stop channel(rcvrChannelName)
```

其中 *rcvrChannelName* 是期望的 JMS 接收器通道名称并且 *transQueueName* 是您在步骤 3a 中创建的 JMS 传输队列的名称。

4. 您可以让 WebSphere MQ 命令环境保持打开状态，因为您在后面的队列管理命令中将需要它。

图 30 显示创建 `bcg.queue.manager` 队列管理器托管的样本 JMS 通道和传输队列。

```
runmqsc bcg.queue.manager
define qlocal(TRANSQ) usage(xmitq) put(enabled) get(enabled)
define channel(SENDER) chltype(sdr) xmitq(TRANSQ)
    conname('remote m/c ip')
stop channel(SENDER)
define channel(RECEIVER) chltype(rcvr)
stop channel(RECEIVER)
```

图 30. 创建样本 JMS 通道和传输队列的命令

在图 30 中的命令创建以下 MQ 对象:

- 名为 TRANSQ 的传输队列
- 名为 SENDER 的发送方通道
- 名为 RECEIVER 的接收器通道

创建 MQ JMS 本地队列

要创建 MQ JMS 本地队列, 使用 WebSphere MQ 命令环境 “runmqsc”。

注: 您必须在 Business Integration Connect 所驻留的机器上创建这些 MQ JMS 队列。

要使用 runmqsc 工具来创建您的本地 MQ 队列, 遵循这些步骤:

1. 在 runmqsc 的命令行, 您可以输入下面的命令:
 - a. 定义从后端系统接收消息的 JMS 入站队列:

```
define qlocal(inQueueName)
```

其中 *inQueueName* 是期望的 JMS 入站队列名称。

- b. 定义将消息发送给后端系统的 JMS 出站队列:

```
define qlocal(outQueueName)
```

其中 *outQueueName* 是期望的 JMS 出站队列名称。

2. 退出 WebSphere MQ 命令环境:

```
end
```

您可以让命令提示符保持打开状态, 因为您在后面的配置步骤中将需要它。

注: 如果您与 Business Integration Connect 的交互仅包括与后端系统通信的一个方向, 您可以只创建 Business Integration Connect 支持该方向的队列。

图 31 显示创建 bcg.queue.manager 队列管理器托管的样本 JMS 队列。

```
define qlocal(JMSIN)
define qlocal(JMSOUT)
```

图 31. 创建样本 JMS 队列的命令

图 31 中的命令创建了下列 JMS 队列:

- 一个名为 JMSIN 的本地入站队列
- 一个名为 JMSOUT 的本地出站队列

创建 JMS 绑定文件

要创建 JMS 绑定文件，请您使用 JMSAdmin 应用程序。本节提供了如何创建 JMS 绑定文件的摘要。获取如何使用 JMSAdmin 应用程序的完整信息，参阅您的 WebSphere MQ 文档。

下列步骤描述了如何创建 JMS 绑定文件：

1. 在命令提示符中（在目录 *WebSphereMQ_Root\java\bin* 里），通过输入下列命令来启动 JMSAdmin 应用程序：

```
JMSAdmin
```

2. 通过在命令行提示符里输入下列命令来定义新的 JMS 配置：

- a. 用下列命令来创建 JMS 上下文：

```
def ctx(contextName)
```

- b. 要改变您的活动上下文，请输入：

```
chg ctx(contextName)
```

其中 *contextName* 是您在前一步里创建的上下文。

- c. 用下列命令来定义队列连接工厂：

```
def qcf(connectionFactoryName) qmgr (queueManagerName) tran(client)  
chan(javaChannelName) host (MQHostName) port (MQport)
```

其中：

- *connectionFactoryName* 是为队列连接工厂指定的名称
 - *queueManagerName* 是您的 WebSphere MQ 队列管理器的名称（管理 Business Integration Connect 使用的队列）
 - *javaChannelName* 是用于建立客户机与 WebSphere MQ 通信通道的名称。缺省通道名称是 `java.channel`。
 - *MQHostName* 是主机机器（WebSphere MQ 队列管理器所驻留的机器）的 IP 地址
 - *MQport* 是主机的端口号
- d. 通过对每个队列输入下列命令为每个队列定义队列：

```
def q(queueAliasName) qmgr (queueManagerName) queue (queueName)
```

- e. 用下列命令来退出 JMSAdmin：

```
end
```

绑定文件在 JMSAdmin.config 文件 PROVIDER_URL 字段中配置的文件夹的子文件夹中创建。该绑定文件的名称如下：

```
.bindings
```

子文件夹名称就是您为 JMS 上下文选择的名称。

图 32 显示了样本 JMS 配置的计算。

```

def ctx(JMS)
change ctx(JMS)

define qcf(WBICHub) qmgr(bcg.queue.manager) tran(CLIENT) chan(java.channel)
host(127.0.0.1) port(1414)

define q(inQ) queue (JMSIN) qmgr(bcg.queue.manager)
define q(outQ) queue (JMSOUT) qmgr(bcg.queue.manager)

```

图 32. 创建样本 JMS 配置的命令

图 32 中从 JMSAdmin 应用程序发出的命令创建了下列 JMS 对象:

- 名为 JMS 的 JMS 上下文
- 本地进站队列 (JMSIN) 的队列别名 inQ
- 本地出站队列 (JMSOUT) 的队列别名 outQ
- 名为 WBICHub 的队列连接工厂

当这些命令完成后, JMSAdmin 应用程序已经在下列目录中创建一个 .bindings 文件:

C:/filesender/config/jms

创建 JMS 目标

将在第 159 页的『配置 JMS 配置目录』中创建的绑定文件复制到您希望它驻留的目录。如果您想保持 JMS 上下文, 请把子文件夹 (和上下文同名) 和绑定文件复制到该目录, 这样一来绑定文件的完整路径如下所示:

/parentDirectory/contextSubdirectory/.bindings

绑定文件的样本路径是:

/mydir/myctx/.bindings

从社区控制台的“目标屏幕”创建一个目标, 指定了下列信息, 如表 115 所示

表 115. 社区控制台“目标详细信息”屏幕的信息

字段名	值	示例
传输方法	JMS	相同
JMS 提供程序 URL	JMS 配置目录的文件系统路径, 它是上下文子文件夹 (如果有上下文的话) 和绑定文件所在的位置, 其格式如下: <i>file://JMSConfigDirectory</i>	该值包含 JMS 提供程序 URL 的 JMS 上下文: <i>file://C:/filesender/config/jms</i>
JMS 队列名	其中 <i>JMSConfigDirectory</i> 是全路径名。该 <i>JMSConfigDirectory</i> 可以有包含 .bindings 文件的上下文子目录 (这是可选的)。 JMS 队列别名在创建 JMS 绑定文件时指定。该队列别名与您在 JMS 提供程序 URL 字段指定的路径相关。 <ul style="list-style-type: none"> • 如果 JMS 提供程序 URL 包括上下文名称, 您不需要提供上下文名称作为 JMS 队列别名的一部分。 • 如果 JMS 提供程序 URL 不包括上下文名称, 在 JMS 队列别名中您必须用以下格式提供它: <i>JMScontext/JMSqueueAlias</i> 	

表 115. 社区控制台 “目标详细信息” 屏幕的信息 (续)

字段名	值	示例
JMS 工厂名	<p>队列连接工厂。在创建绑定文件时，用命令 <code>define qcf</code> 指定了这个名称。该工厂名称与您在 JMS 提供程序 URL 字段指定的路径相关。</p> <ul style="list-style-type: none"> • 如果 JMS 提供程序 URL 包括上下文名称，您不需要提供上下文名称作为 JMS 工厂名称的一部分。 • 如果 JMS 提供程序 URL 不包括上下文名称，在 JMS 工厂名称中您必须用以下格式提供它： <i>JMScontext/JMSfactory</i> 	WBICHub
JNDI 工厂名	<code>com.sun.jndi.fscontext.ReffSContextFactory</code>	相同

注：目标必须可以访问子文件夹和 JMS 绑定文件所在的目录。

在表 115 里，“示例”列中列出了第 162 页的图 31 and 第 164 页的图 32 中定义的 JMS 配置的目标值。

创建 JMS 网关

从社区控制台的“网关”屏幕创建了一个网关，并指定了表 116 中的信息

表 116. 社区控制台的 “网关详细信息” 屏幕的信息

字段名	值	示例
传输方法	JMS	相同
目标 URI	<p>JMS 配置目录的文件系统路径，它是上下文子文件夹（如果有上下文的话）和绑定文件所在的位置，其格式如下： <i>file://JMSConfigDirectory</i></p> <p>其中 <i>JMSConfigDirectory</i> 是全路径名。该 <i>JMSConfigDirectory</i> 可以具有包含 <code>.bindings</code> 文件的上下文子文件夹。</p>	该值包括在目标 URI 的 JMS 上下文内： <code>file:///C:/filesender/config/jms</code>
JMS 工厂名	<p>当创建 JMS 绑定文件时您指定的 JMS 工厂名。该工厂名与您在目标 URI 字段中指定的路径有关：</p> <ul style="list-style-type: none"> • 如果目标 URI 包括上下文名称，您不需要上下文名称提供为 JMS 工厂名称的一部分。 • 如果目标 URI 不包括上下文名称，您必须在 JMS 工厂名中用以下格式提供它： <i>JMScontext/JMSfactory</i> 	WBICHub
JMS 消息类	<p>下列 JMS 消息类之一：</p> <ul style="list-style-type: none"> • <code>StreamMessage</code> • <code>BytesMessage</code> • <code>TextMessage</code> 	取决于后端系统支持的消息类

表 116. 社区控制台的“网关详细信息”屏幕的信息 (续)

字段名	值	示例
JMS 队列名	<p>JMS 队列别名，创建 JMS 绑定文件时指定的名称。该名称与您在目标 URI 字段中指定的路径有关：</p> <ul style="list-style-type: none"> • 如果目标 URI 包括上下文名称，您不需要提供上下文名称作为 JMS 队列别名的一部分。 • 如果目标 URI 不包括上下文名称，您必须用以下格式在 JMS 队列别名中提供它： <i>JMScontext/JMSqueueAlias</i> 	outQ
JNDI 工厂名	com.sun.jndi.fscontext.RefFSContextFactory	相同

在第 165 页的表 116 中，“示例”列中列出了第 162 页的图 31 和第 164 页的图 32 中定义的 JMS 配置的网关值。

声明和商标

声明

IBM 可能不会在所有国家或地区提供本文档中讨论的产品、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。您可以用书面方式将许可查询寄往：

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

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：

International Business Machines Corporation “按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许免除明示或暗含的保证。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。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

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 Burlingame Laboratory Director
IBM Burlingame Laboratory
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A

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 国际软件许可协议或任何同等协议中的条款提供。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的，实际结果可能会有差异。本文档的用户应当验证其特定环境的适用数据。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

本信息可能包含日常运营中使用的数据和报告的示例。为了尽可能完整地说明它们，这些示例可能包含了个人、公司、品牌和产品的名称。所有这些名称都是虚构的，如果与实际的企业使用的名称和地址有任何雷同则纯属巧合。

所有关于 IBM 未来方向或意向的声明都可随时更改或收回，而不另行通知，它们仅仅表示了目标和意愿而已。

版权许可

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。

Websphere Business Integration Connect 包含名为 ICU4J 的代码，IBM 根据国际程序许可证协议条款，遵照其“排它组件”条款授予您使用许可权。但是，IBM 有义务向您提供以下声明：

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2003 International Business Machines Corporation and others

All rights reserved.

据此免费授予获得本软件及相关文档文件副本（统称“软件”）的任何人许可权，允许他们没有限制地处理该“软件”，包括有权无限制地使用、复制、修改、合并、发布、分发和/或销售“软件”的副本，并允许该“软件”的接受者也能执行这些操作。前提是以上版权声明和本许可权声明出现在所有“软件”的副本中；并且，以上版权声明和本许可权声明也要出现在支持文档中。

本软件是“按现状”提供的，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关适销、适用于某种特定用途以及不侵犯第三方权利的保证。在任何情况下，对于因本软件的使用或性能所引起的，或与本软件的使用或性能相关的任何索赔、特殊的间接或由此产生的损失，或任何由无法使用、数据丢失、利润减少而造成的损失，无论是合同行为、疏忽或其它侵权行为，本声明中包括的版权所有人概不负责。

除在本声明中使用外，未经版权所有人事先书面授权，不得将版权所有人的名字用于广告中，也不得用于本软件的促销、使用或其它行为。

编程接口信息

编程接口信息（如果提供了的话）旨在帮助您使用该程序创建应用软件。

通用的编程接口允许您编写能获得该程序的工具服务的应用软件。

但是，该信息还可能包含诊断、修改和调优信息。提供诊断、修改和调优信息是为了帮助您调试您的应用软件。

警告： 请勿使用该诊断、修改和调优信息作为编程接口，因为它是不断变化的。

商标和服务标记

下列术语是国际商业机器公司在美国和 / 或其他国家或地区的商标或注册商标：

IBM
IBM 徽标
AIX
CrossWorlds
DB2
DB2 Universal Database
Domino
Lotus
Lotus Notes
MQIntegrator
MQSeries
Tivoli
WebSphere

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

MMX、Pentium 和 ProShare 是 Intel Corporation 在美国和 / 或其他国家或地区的商标或注册商标。

Java 和所有基于 Java 的商标是 Sun Microsystems,Inc. 在美国和 / 或其他国家或地区的商标。

其他公司、产品和服务名称可能是其他公司的商标或服务标记。

WebSphere Business Integration Connect Enterprise Edition 和 Advanced Edition 包括了由 Eclipse Project (www.eclipse.org) 开发的软件。



索引

[B]

- 包装器数据处理程序 62, 70
 - 创建配置业务对象 70
 - 位置 65, 70
 - 业务对象结构 79
 - 资源库文件 65, 71

[C]

- 参与者连接 26, 29, 148
- 参与者文档 27
- 传输包络 14, 15, 41, 44
- 传输级头 10
 - 为 JMS 110
 - 针对 HTTP 82, 94
- 传输协议 16
 - 选择 34, 118
 - 有效负载位置 14
- 传输协议机制 22
 - HTTP 22
 - JMS 22

[D]

- 顶级业务对象 80, 92
- 动态元对象
 - 为 JMS 112
 - 针对 HTTP 82, 84, 94, 96

[E]

- 二进制文档 24, 28

[F]

- 访问客户机 53, 63
- 封装 9
 - 后端集成 10
 - 无 10
- 附件 10, 15
 - 编码 15
 - 附件容器 57, 58
 - 附件数据 56
 - 内容类型 15
 - 内容信息 56
 - 数据处理程序 44
- 附件容器业务对象 57
- 附件数据处理程序 44, 62, 86
 - 表示附件 57

- 附件数据处理程序 (续)
 - 创建配置业务对象 50
 - 创建业务对象定义 54
 - 配置 50
 - 位置 50
 - 资源库文件 50, 52
 - SOAP 文档 98
- 附件业务对象 57

[G]

- 故障业务对象 92

[H]

- 后端集成 3
 - 概述 3
- 后端集成封装
 - 包络标志 14
 - 在需要时 17, 19
 - HTTP 传输协议 17
 - JMS 传输协议 18
- 后端文档 27, 28
- 后端系统 3
 - 将文档发送到 23
 - 文档接收自 27
 - 选择 20

[J]

- 计算节点 124, 126, 131
- 交互服务器 33
 - 被支持的版本 34
 - 集成样本 36
 - 计划集成 34
 - 配置 40
 - 与 InterChange Server 兼容的组件 34
 - 支持的传输协议 34
- 交互。
 - 请参阅 异步交互, 同步交互
- 具有用户定义的属性的业务对象 95

[M]

- 目标 28
 - 来自数据交换 149
 - 来自 InterChange Server 38
 - 来自 Message Broker 120
 - HTTP 传输协议 39, 121
 - JMS 传输协议 39, 121, 164

[N]

内容信息业务对象 56

[Q]

请求处理 42

通过 HTTP 74, 77, 78, 89

在 JMS 上 107

请求业务对象 42

角色 73, 78

结构 81, 93

位置 80, 92

消息头信息 84, 96

缺省的附件业务对象 56

[S]

社区管理者 3

事件通知 6, 42

通过 HTTP 62, 64, 73, 77, 89

在 JMS 上 107

事件业务对象 42

数据处理 35

包装器 70

顶级元对象 53, 72

附件 44

子元对象 50, 70

数据交换 133

[T]

特定于 InputDestination 连接器的属性 108

同步交互

基于 HTTP 的 ICS 65, 78, 82, 94

基于 HTTP 的 ICS 63

通过 HTTP 与 Message Broker 118

在 HTTP 上的 ICS 36

HTTP 传输协议 17

[W]

网关 24

到数据交换 146

到 Message Broker 119

给 InterChange Server 37

HTTP 传输协议 37, 119

JMS 传输协议 38, 120, 165

文档

参与者 27

传输包络 14, 15, 41

附件。

请参阅 附件

后端 27, 28

业务对象 40

文档 (续)

有效负载。

请参阅 有效负载

AS1 13

AS2 12, 14

cXML 5, 82

EDI 133, 153

RosettaNet 6, 11

SOAP 5, 97, 125

文档流定义 25, 148

文档流交互 29, 148

文件系统协议 19

[X]

响应业务对象 42, 94

角色 73, 78

结构 81

位置 80, 92

消息流 122, 124, 125, 131

消息。

请参阅 文档

协作 43

Adapter for HTTP 97

Adapter for JMS 113

Adapter for XML 85

Connect Servlet 67

[Y]

业务对象 40

顶级 80, 92

动态元对象 84, 96, 112

附件容器 57

内容信息 56

请求。

请参阅 请求业务对象

缺省附件 56

事件 42

响应。

请参阅 响应业务对象

用户定义的属性 95

用于文档 40

用 HTTP 协议配置的元对象 96

有效负载 40, 58

HTTP 属性 83

JMS 属性 111

业务对象定义 40

创建 40, 54, 79, 91, 109

附件数据处理程序 54

为 JMS 109

针对 HTTP 79, 91

业务协议 5

异步交互

基于 HTTP 的 ICS 65, 82

异步交互 (续)
 使用 HTTP 的 ICS 94
 通过 HTTP 的 ICS 63
 通过 HTTP 与 Message Broker 118
 在 HTTP 上的 ICS 110
 在 JMS 上的 Message Broker 119
 cXML 文档 6
 HTTP 传输协议 17
 JMS 传输协议 18
 JMS 上的 ICS 36
用 HTTP 协议配置的元对象 96
有效负载 10, 14, 40
 编码 15
 内容类型 15
有效负载数据处理程序 41
 Adapter for HTTP 89
 Adapter for JMS 108
 Adapter for XML 77
有效负载业务对象 40, 58
 为 JMS 109
 针对 HTTP 79, 91

[特别字符]

“附件”数据处理程序
 位置 50
“后端集成”封装 10
 示例 16
 在需要时 16
 HTTP 头信息 82, 94
 HTTP 业务对象 79, 91
 JMS 传输级头 110
 JMS 业务对象 109
“无”封装 10
 文件系统协议 19
 在需要时 16, 17, 19
 HTTP 传输协议 17, 19
 HTTP 业务对象 79, 91
 JMS 业务对象 109

A

Adapter for HTTP 86
 安装 89
 绑定协作 97
 配置 88
 协议处理程序 89
 业务对象结构 91
 有效负载数据处理程序 89
Adapter for JMS
 安装 107
 绑定协作 113
 配置 107
 设置输入队列 108
 业务对象结构 109

Adapter for JMS (续)
 有效负载数据处理程序 108
Adapter for Web Services 98
Adapter for XML
 安装 77
 绑定协作 85
 配置 77, 78
 业务对象结构 79
 有效负载数据处理程序 77
AS 封装 154
AS1 文档 13
AS2 文档 12, 14

B

B2B 功能 25, 29, 145
Business Integration Connect
 发送文档 23
 到数据交换 146
 到 ICS 37
 到 Message Broker 119
 封装类型 9
 关于 InterChange Server 的配置 37
 关于 Message Broker 的配置 119
 关于“数据交换”的配置 143
 接收文档 27
 从 ICS 38
 来自数据交换 149
 来自 Message Broker 120
 配置 23
 业务协议 5

C

Connect Servlet 35, 62, 63
 部署 65
 配置 65
 位置 65
 指定协作 67
 servlet 属性文件 66
cXML 文档 4, 5, 17, 82

E

EDI 文档 4, 133
 封装 24, 28
 路由 153
 有效负载位置 14
 支持的传输协议 24, 28
 HTTP 传输协议 17

H

HTTP 传输协议 17
 传输协议机制 22

HTTP 传输协议 (续)

- 创建数据业务对象 109
 - 创建头信息 82, 94
 - 发送文档
 - 到 ICS 61, 86
 - 到 Message Broker 123
 - 接收文档
 - 从 ICS 74, 87
 - 来自 Message Broker 123
 - 有效负载位置 14
 - Business Integration Connect
 - 和 ICS 37, 39
 - 和 Message Broker 119, 121
 - ICS 集成样本 36
 - ICS 业务对象结构 79, 91
 - InterChange Server 和 35, 61, 85, 97
 - Message Broker 和 118, 122
- ## HTTP 属性业务对象 83
- HTTP 协议处理程序 76, 77, 78, 89
 - HTTPInput 节点 124, 126
 - HTTPReply 节点 124, 126
 - HTTPRequest 节点 125, 126

I

ICS。

请参阅 InterChange Server

InterChange Server

- 创建复制 84
- 创建构件 97, 112
- 发送所需的组件
 - 在 JMS 上 101
- 发送所需组件
 - 通过 HTTP 62, 86
- 集成 101
- 接收所需的组件
 - 通过 HTTP 74, 86
 - 在 JMS 上 101
- 配置 61, 84, 97, 112
- 请求处理 77, 78, 89, 107
- 事件通知 64, 73, 77, 89, 107
- HTTP 传输协议 61, 85, 97
- JMS 传输协议 101

J

- JMS 绑定文件 159
- JMS 传输协议 18
 - 传输协议机制 22
 - 创建头信息 110
 - 发送文档
 - 到 Message Broker 126
 - 给 ICS 102
 - 接收文档
 - 从 ICS 104

JMS 传输协议 (续)

- 接收文档 (续)
 - 来自 Message Broker 128
 - 有效负载位置 14
 - Business Integration Connect
 - 和 ICS 38, 39
 - 和 Message Broker 120, 121
 - 与数据交换 146, 149
 - ICS 集成样本 36
 - ICS 业务对象构造 109
 - InterChange Server 和 36, 101
 - Message Broker 118
 - Message Broker 和 118, 126
 - WebSphere MQ 159
- ## JMS 属性业务对象 111

M

Message Broker 117

- 被支持的版本 118
 - 发送所需的组件
 - 通过 HTTP 122
 - 在 JMS 上 126
 - 集成样本 119
 - 计划集成 118
 - 接收所需的组件
 - 通过 HTTP 122
 - 在 JMS 上 126
 - 配置 121
 - 支持的传输协议 118
 - HTTP 传输协议 122
 - JMS 传输协议 126
- ### MO_DataHandler_Default 顶级元对象 53
- ### MO_Server_DataHandler 顶级元对象 53, 70, 72
- MQInput 节点 124, 131
 - MQOutput 节点 124, 131

R

- RosettaNet 文档 6, 11
 - 封装 17, 24, 28
 - 有效负载位置 14
 - 支持的传输协议 24, 28
 - ICS 集成样本 36

S

- SOAP 文档 4, 5
 - 封装 24, 28
 - 基于 HTTP 的 ICS 97
 - 在 HTTP 上的 Message Broker 125
 - 支持的传输协议 24, 28
 - HTTP 传输协议 17

W

WebSphere Business Integration Adapter for HTTP。

请参阅 Adapter for HTTP

WebSphere Business Integration Adapter for JMS。

请参阅 Adapter for JMS

WebSphere Business Integration Adapter for Web Services。

请参阅 Adapter for Web Services

WebSphere Business Integration Adapter for XML。

请参阅 Adapter for XML

WebSphere Business Integration Connect Servlet。

请参阅 Business Integration Connect Servlet

WebSphere Business Integration Connect。

请参阅 Business Integration Connect

WebSphere Business Integration Message Broker。

请参阅 Message Broker

WebSphere Data Interchange。

请参阅 Data Interchange

WebSphere InterChange Server。

请参阅 InterChange Server

WebSphere MQ 159

X

XML 传输包络。

请参阅 传输包络



中国印刷