

版本 6.1.0



产品概述

版本 6.1.0



产品概述

注意:

在使用本资料之前，请务必阅读文档末尾的声明一节中的一般信息。

2008 年 2 月 1 日

此版本适用于 WebSphere Process Server for Multiplatforms V6.1.0（产品编号为 5724-L01）及所有后续发行版和修订版，直到在新版本中另有声明为止。

要向我们发送您对本文档的意见，请向 ctscrcf@cn.ibm.com 发送电子邮件消息。IBM 欢迎您提出宝贵意见。

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 2005, 2008. All rights reserved.

目录

图	v	消息服务客户机	25
第 1 章 WebSphere Process Server 简介	1	调解模块	26
第 2 章 本发行版的新增内容	3	调解原语	28
第 3 章 产品系列概述	7	服务消息对象	31
第 4 章 WebSphere Process Server 的体系结构概述	11	第 8 章 管理 WebSphere Process Server 上的应用程序	35
面向服务的体系结构核心	11	第 9 章 在 WebSphere Process Server 上开发和部署应用程序	37
服务组件体系结构	12	第 10 章 WebSphere Process Server 上的安全性	39
服务数据对象和业务对象	13	第 11 章 在 WebSphere Process Server 上进行监视	41
WebSphere Process Server 中的公共事件基础结构	14	第 12 章 样本和教程	43
支持服务	15	教程	43
调解流	15	访问样本 (样本库)	43
接口映射	16	第 13 章 标准一致性	47
业务对象映射	16	辅助功能选项	47
关系	16	美国联邦信息处理标准	48
选择器	17	Common Criteria	49
服务组件	17	因特网协议版本 6	49
业务流程	18	第 14 章 全球化	51
人员任务	19	声明	57
业务状态机	19		
业务规则	19		
第 5 章 导入、导出和适配器	21		
第 6 章 WebSphere Process Server 中的部署环境	23		
第 7 章 企业服务总线消息传递基础结构	25		



1. 基于 WebSphere Process Server 组件的框架	11	5. 调解模块的简单示例.	27
2. 基于 WebSphere Process Server 组件的框架	12	6. 包含调解模块的 EAR 文件的简单示例.	28
3. 基于 WebSphere Process Server 组件的框架	15	7. 包含三个调解原语的调解模块.	29
4. 基于 WebSphere Process Server 组件的框架	18	8. SMO 结构概述	33

第 1 章 WebSphere Process Server 简介

IBM® WebSphere Process Server 是一种业务流程集成服务器，它从成熟的业务集成概念、应用程序服务器技术和最新的开放式标准发展而来。WebSphere Process Server 是一个高性能的业务引擎，它可以帮助构建流程以满足业务目标。

WebSphere Process Server 允许在面向服务的体系结构 (SOA) 中部署基于标准的业务集成应用程序。SOA 执行日常业务应用程序，并将它们分解为单独的业务功能和流程，将它们呈现为服务。WebSphere Process Server 基于健壮的 J2EE 1.4 基础结构以及由 WebSphere® Application Server 提供的相关平台服务，来帮助您面对当前的业务集成挑战。这些挑战包括但不限于业务流程自动化。

WebSphere Process Server 支持对跨人员、系统、应用程序、任务、规则的流程的部署以及这些对象之间的交互。它同时支持长期运行和短期运行的业务流程，并为松散耦合的业务流程提供事务回滚等功能。

硬件和软件要求

要查看 WebSphere Process Server 支持的硬件和软件的正式声明，请参阅 WebSphere Process Server 系统要求 Web 站点。

信息指南

为了帮助您在产品信息中心内部和外部浏览可用信息源，在 IBM developerWorks® 上的 WebSphere business process management 区域 (<http://www.ibm.com/developerworks/websphere/zones/bpm/>) 中提供了在线业务流程管理信息指南。

第 2 章 本发行版的新增内容

V6.1.0 提供的增强功能主要集中在以下九个方面：平台调整和通用性；更易于使用和更高的业务灵活性；改进了产品安装和配置；增强了应用程序的人员工作流程能力；增强了应用程序的业务流程能力；增强了业务流程编排器资源管理器和业务流程编排器观察器；增强了应用程序的业务规则能力；改进了与新的和已增强的服务组件体系结构（SCA）绑定的连通性并且改进了服务质量；增加了企业服务总线调解支持。

欢迎使用 WebSphere Process Server V6.1.0，它提供了下列新功能：

- 更易于使用和更高的业务灵活性：
 - WebSphere Process Server 管理控制台通常是“集成解决方案控制台”框架的一部分，特殊情况下是 WebSphere Application Server 管理控制台的一部分。因此，对于 WebSphere Process Server 和 WebSphere Application Server 来说，许多管理任务（例如，设置安全性、查看日志和安装应用程序）都相同。在 WebSphere Application Server 文档中说明了这些任务。
 - Install Factory 增强功能，通过它在特定修订级别定义和构建 WebSphere Process Server 安装，就可以向安装产品中添加更多文件，同时省去不需要的功能部件。
 - 基于 Eclipse 的概要管理工具，适用于所有 WebSphere 平台，通过它可以在同一工具中创建和扩充 WebSphere Process Server 和 WebSphere Enterprise Service Bus 概要文件。
 - 增强对于部署的配置管理。
 - 支持根据模式来进行配置，从而提高了为 WebSphere Process Server 环境配置 WebSphere Application Server Network Deployment 集群和所有必需资源的管理员的效率。
 - 现在，首次访问时将创建必需的数据库表，而不需要在配置期间手动创建数据库。
- 增强了应用程序的人员工作流程能力：
 - 提供了参与者替代功能，在运行时应用程序的用户没空的情况下，它允许他们临时委派工作。
 - 成批 API 允许通过执行一项操作转移多个任务。
 - 支持 Virtual Member Manager，它是 WebSphere Application Server 的新组件，用于集成特定于客户机的人员目录。
 - 可以将自动删除限制为删除已成功完成的任务。
 - 可以将使用 IBM Lotus® Forms Designer（已集成到 WebSphere Integration Developer 中）创建的表单用作人员任务和流程的用户界面。
 - 可以使用在 WebSphere Integration Developer 中生成的 portlet 来扩展 WebSphere Portal Server 的“我的任务”portlet。
- 增强了应用程序的业务流程能力：
 - 业务流程管理器的通用 JMS 接口允许与业务流程模板和实例进行程序交互。
 - 通过对业务流程管理器的通用 Web Service 接口进行扩展，添加了新的运行时功能。

- Web Service 业务流程执行语言 (WS-BPEL) **forEach** 构造允许以并行或串行方式来处理数目动态变化的多个分支。
- 扩展了暂挂能力, 允许指定自动恢复流程实例。
- 可以将自动删除限制为删除已成功完成的流程, 从而允许您只保留需要进一步分析或修复的流程实例。
- 添加了数据处理选项, 在访问期间将忽略缺少的数据而不会产生错误。
- 支持单线程流程中的反向链接。
- 增强了应用程序的业务规则能力:
 - 现在, 定制的业务用户客户机可以使用新的业务规则管理 API 来管理流程中所使用的业务规则, 通过此 API 可以创建、读取、更新和删除业务规则, 而不必使用业务规则管理器。
 - 可以为业务规则组指定新的定制属性和从规则集的规则逻辑和决策表中访问新的定制属性, 以向这些规则提供对属性中捕获的环境信息的访问权。属性还可以用于通过业务规则管理器或定制管理客户机在业务规则组上搜索。
- 增强了业务流程编排器资源管理器和业务流程编排器观察器:
 - 业务流程编排器资源管理器增强功能允许您执行下列任务:
 - 处理用户的缺席和替代。
 - 对流程和任务使用“暂挂结束时间”选项。
 - 查看和编辑 XML 源数据。
 - 使用改进的定制视图来对返回给应用程序用户的数据进行排序并控制数据量。
 - 将流程和任务中的过滤条件与它们的定义和实例进行组合。
 - 浏览相关任务 (子任务和后续任务), 管理和查看有关特定任务的信息。
 - 将人员任务优先级和业务类别作为过滤条件和列表列。
 - 编辑定制属性。
 - 改进了图形流程视图的可用性。
 - 现在, 可以导出“业务流程编排器观察器”中的报告, 以便在 Microsoft® Excel 等工具中进一步分析, 并且还可以保存这些报告, 以便稍后根据安排或要求来自动生成。
- 改进了连通性:
 - 增强了对 Web 服务描述语言 (WSDL) XML 和模式定义 (XSD) 的支持, 从而能够使用许多作为业界标准的 XML 模式, 并且增强了与其他环境进行连接的能力。
 - 新的和已增强的 SCA 绑定提高了连通性:
 - 通过增强功能, 能够与符合任何 JMS 1.1 应用程序服务器工具 (ASF) 的消息传递提供程序进行集成, 并且能够为 WebSphere Application Server 自动设置通用 JMS 资源。
 - 通过新的通用 HTTP 1.0 和 1.1 绑定, 能够与更多应用程序和服务进行连接。
 - 通过新的数据绑定很容易与 WebSphere Transformation Extender 集成。
 - 定制数据样本包括关于按固定宽度导出和导入、定界和名称/值对数据格式的样本方案。
 - 增强了业务故障支持, 从而能够区分业务异常和运行时异常。
 - 新的数据处理程序框架允许灵活地指定数据绑定。

- 新的和已增强的服务质量:
 - 可以调用业务对象实例验证器，以进行基于接口限定符的隐式验证或者进行显式程序验证。
 - 增强的事件排序支持改进了服务质量支持。
- 增加了企业服务总线调解支持:
 - 新的业务对象映射原语使得将映射嵌入到调解流中变得很容易。
 - 新的 **ServiceInvoke** 原语将从请求或响应流中调用目标服务。
 - 新的分割和聚集原语通过分割消息的各个部分并调用多个服务来增强了单条消息或组合消息的处理。
 - 增强的定制调解允许定义新的终端。
 - 对消息元素 **setter** 原语进行了一些改进。
 - **MessageLogger** 原语包括系统范围内的公共数据库支持和模式限定符。
- 平台调整和通用性:
 - 支持 WebSphere Application Server V6.1，从而允许 WebSphere Process Server 使用 WebSphere Application Server 功能并继承 WebSphere Application Server V6.1 中引入的新功能部件和优势。
 - 支持 i5/OS[®]，从而使 WebSphere Process Server 能够在 System i[™] 上运行和使用本机 i5/OS 工具。
 - 对于 System i 系统的远程 DB2[®] 支持实现了与 i5/OS 应用程序集成，并实现了使用 i5/OS 上运行的关系数据库管理服务。
 - 支持从分布式平台上远程访问 z/OS[®] 操作系统上的 DB2 安装。
 - 支持 64 位的 Windows[®] 和 UNIX[®] 平台。
 - 支持将 Windows Vista 作为设计、开发和测试平台（但不用于生产），从而能够使用 Microsoft 的最新版本的桌面操作系统。
- 改进了产品安装和配置:
 - 受指导的部署环境安装使您很容易在多台机器上的多个安装之间定义一个工作集群或一组集群。
 - 非 root 用户安装允许您在 Linux[®] 或 UNIX 操作系统上以非 root 用户身份进行安装，或者在 Windows 操作系统上以非管理员身份进行安装。
 - WebSphere Process Server 为概要管理工具提供了一个插件，以允许在同一工具中创建和扩充 WebSphere Process Server 和 WebSphere Enterprise Service Bus 概要文件。概要管理工具是一个基于 Eclipse、用于创建和扩充概要文件的工具，它是作为 WebSphere Application Server V6.1 的一部分提供的。
 - WebSphere Process Server 为 IBM Installation Factory 提供了一个插件，IBM Installation Factory 是一个基于 Eclipse 的工具，允许您创建 WebSphere Process Server 和 WebSphere Enterprise Service Bus 定制安装包（CIP），定制安装包中可以包括更新包、修订包和临时修订，也可以排除功能部件，还可以包括其他文件和脚本以进行安装和创建概要文件。
 - 升级安装。对于购买并安装了 WebSphere Enterprise Service Bus V6.1 并且后来又购买了 WebSphere Process Server 6.1 的客户，能够通过使用 WebSphere Process Server 6.1 安装程序将他们的 WebSphere Enterprise Service Bus 安装升级为 WebSphere Process Server 安装。

- WebSphere Process Server 安装程序支持在远程连接至 i5/OS 的 Windows 系统上进行安装。概要管理工具可以在远程连接至 i5/OS 系统的 Windows 系统上运行。
- 为基于 Eclipse 的 IBM 用户界面帮助系统提供了新的安装程序，从而能够从中央存储库中下载 WebSphere Process Server 信息中心文档并进行安装。

相关任务

 使用 Lotus Forms Server API 和 Lotus Forms Viewer

可以将 IBM Lotus Forms Designer 创建的表单用作人员任务和流程的用户界面。

第 3 章 产品系列概述

WebSphere Process Server 是 WebSphere Business Process Management 平台的一部分，并且可与 IBM 的许多其他产品配合使用。

WebSphere Application Server Network Deployment

WebSphere Process Server 基于健壮的 J2EE 基础结构以及由 WebSphere Application Server 提供的关联平台服务。WebSphere Process Server 基于 WebSphere Application Server Network Deployment。有关 WebSphere Application Server Network Deployment 的更多信息，请参阅 WebSphere Application Server Network Deployment 文档。

WebSphere Process Server 还会使用 WebSphere Application Server 中的基础结构和平台服务。有关 WebSphere Application Server 的更多信息，请参阅 WebSphere Application Server V6.1 信息中心。

IBM WebSphere Enterprise Service Bus

WebSphere Process Server 与 WebSphere Enterprise Service Bus 基于相同的技术。企业服务总线功能是 WebSphere Process Server 的底层功能的一部分，WebSphere Enterprise Service Bus 不需要其他许可证即可利用这些功能。但是，您可以在企业范围部署已购买的 WebSphere Enterprise Service Bus 的其他独立许可证，以扩展 WebSphere Process Server 支持的流程集成解决方案的连接范围。例如，WebSphere Enterprise Service Bus 可安装在距 SAP 应用程序较近的位置，以主管 IBM WebSphere Adapter for SAP 和变换 SAP 消息，然后通过网络将该信息发送给 WebSphere Process Server 设计的业务流程。

IBM WebSphere Partner Gateway

IBM WebSphere Partner Gateway 在与 WebSphere Process Server 配合使用时可支持企业到企业应用程序。WebSphere Partner Gateway 的受限许可证已随 WebSphere Process Server 提供。有关 WebSphere Partner Gateway 的更多信息，请参阅 WebSphere Partner Gateway 产品文档库。

IBM WebSphere Integration Developer

WebSphere Integration Developer 是 WebSphere Process Server 的开发环境。它是一种常用工具，用于构建跨越 WebSphere Process Server、WebSphere Enterprise Service Bus 和 WebSphere Adapters 的、基于面向服务的体系结构（SOA）的集成解决方案。有关 WebSphere Integration Developer 的更多信息，请参阅 IBM WebSphere Business Process Management 信息中心。

IBM WebSphere Adapters

IBM WebSphere Adapters 允许将现有企业信息系统基础结构与部署在 WebSphere Process Server 上的应用程序集成在一起。WebSphere Adapters 使您能够快速而轻松地创建综合进程，这些进程之间可以交换有关企业资源计划、人力资源、客户关系管理和供应链系统的信息。

应用程序适配器从跨行业和特定行业的已打包应用程序中抽取数据和交易信息，并将它们连接至中央集线器。技术适配器提供了连通性，以访问可增强集成基础结构的数据、技术和协议。可以使用 Adapter Development Toolkit 来创建定制适配器。

WebSphere Adapters 是 WebSphere Integration Developer 中包括的组件。

有关 WebSphere Adapters 的更多信息，请参阅 IBM WebSphere Business Process Management 信息中心中的 WebSphere Integration Developer 文档。

IBM WebSphere Business Modeler 和 IBM WebSphere Business Monitor

WebSphere Process Server 和 WebSphere Integration Developer 包含了其他功能，这些功能使建模、构建、部署、安装、配置、运行、监视和管理集成应用程序成为可能。WebSphere Integration Developer 补充了 IBM WebSphere Business Modeler 和 IBM WebSphere Business Monitor。

有关这些产品的更多信息，请参阅 WebSphere Business Modeler 信息中心以及 IBM WebSphere Business Process Management 信息中心中的 WebSphere Business Monitor 文档。

IBM Rational® Application Developer 和 IBM Rational Software Architect

WebSphere Integration Developer 可以与 IBM Rational Application Developer 或 IBM Rational Software Architect 配合使用，从而创建独一无二、功能强大的综合集成开发平台。

有关这些产品的更多信息，请参阅 Rational Application Developer 信息中心和 Rational Software Architect 信息中心。

IBM CICS® Transaction Gateway 和 IBM WebSphere Host Access Transformation Services

可扩展现有应用程序，以便在企业流程中配合 IBM 企业现代化产品服务组合（包括 CICS Transaction Gateway 和 WebSphere Host Access Transformation Services）重用这些应用程序。

有关这些产品的更多信息，请参阅 CICS Transaction Gateway Library 和 WebSphere Host Access Transformation Services (HATS) 信息中心。

IBM WebSphere Portal

IBM WebSphere Portal 使您能够访问各种管理功能，并允许 portlet 访问 WebSphere Process Server 中的业务流程及其他服务组件体系结构服务。

有关 WebSphere Portal 的更多信息，请参阅 WebSphere Portal 产品文档库。

IBM WebSphere Application Toolkit

WebSphere Application Server Toolkit 是一套工具，可帮您组装、测试和部署 Web Service 以便在 WebSphere Process Server 中使用。

有关更多信息，请参阅 WebSphere Application Server 信息中心中的 WebSphere Application Server Toolkit 文档。

IBM WebSphere Extended Deployment

WebSphere Extended Deployment 使 WebSphere Process Server Network Deployment 环境能够在内部的集群之间调整资源，以满足那些定义为策略的处理目标。由于应用程序数量的消长，在峰值期可能没有足够的处理能力来满足各个请求，而且很难优化资源以便关键应用程序获取所需处理时间。

在这些时段内动态重新分配处理能力可能有助于满足业务需求。WebSphere Extended Deployment 从应用程序数量较少的集群中动态除去资源，并将这些资源添加至正在处理需要额外资源的应用程序的集群。处理优先级是在 WebSphere Extended Deployment 中作为策略指定的。

有关 WebSphere Extended Deployment 的更多信息，请参阅 WebSphere Extended Deployment 信息中心。

第 4 章 WebSphere Process Server 的体系结构概述

WebSphere Process Server 提供了集成平台，该平台具有功能齐全的标准业务流程引擎，并可使用 WebSphere Application Server 的全部功能。

WebSphere Process Server 构建在统一调用编程模型和统一数据表示模型之上，是一个面向服务的体系结构（SOA）集成平台。

WebSphere Process Server 的基本运行时基础结构是 WebSphere Application Server。服务组件体系结构和业务对象是 SOA 核心的一部分，提供了统一的调用和数据表示编程模型。SOA 核心包含公共事件基础结构，用于监视和管理在 WebSphere Process Server 上运行的应用程序生成事件。支持服务为 WebSphere Process Server 提供基本业务对象和变换框架。服务组件代表综合应用程序所需的功能组件。

强大的基础（WebSphere Application Server 和 SOA 核心）与 WebSphere Process Server 中的服务组件相结合，允许您快速开发和部署在 WebSphere Process Server 上运行的复杂综合应用程序。

一个基于组件的框架能够处理所有样式的集成。

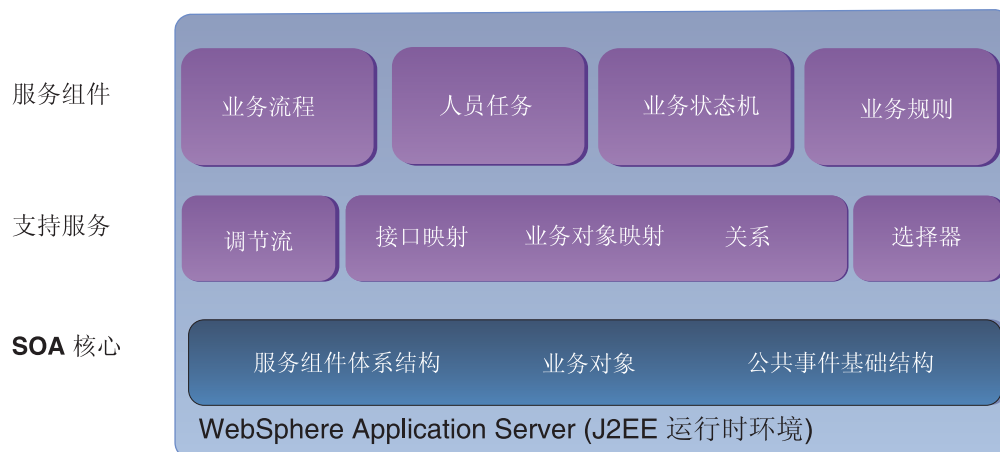


图 1. 基于 WebSphere Process Server 组件的框架

面向服务的体系结构核心

IBM WebSphere Process Server 的“面向服务的体系结构”核心为在 WebSphere Process Server 上运行的应用程序提供统一的调用、数据表示编程模型以及监视和管理功能。

面向服务的体系结构（SOA）就组件及其提供的服务方面而言是软件系统结构的概念性描述，而不会考虑这些组件、服务和组件之间连接的底层实现。WebSphere Process Server 使您能够在 SOA 中部署基于标准的流程集成解决方案。这意味着可以为组件创建和维护明确定义的业务级别接口集合，不受低级别技术更改影响。基于 SOA 的松散

耦合的集成应用程序提供了灵活性和机动性。您可以不受平台、协议和产品限制即可实现集成解决方案。有关 SOA 的更多信息，请参阅 IBM Web 站点中的面向服务体系结构 (SOA)。

服务组件体系结构和业务对象是 SOA 核心的一部分，它们为部署在 WebSphere Process Server 上的应用程序提供统一的调用和数据表示编程模型。SOA 核心还包含公共事件基础结构，用于监视和管理在 WebSphere Process Server 上运行的应用程序生成事件。

一个基于组件的框架能够处理所有样式的集成。

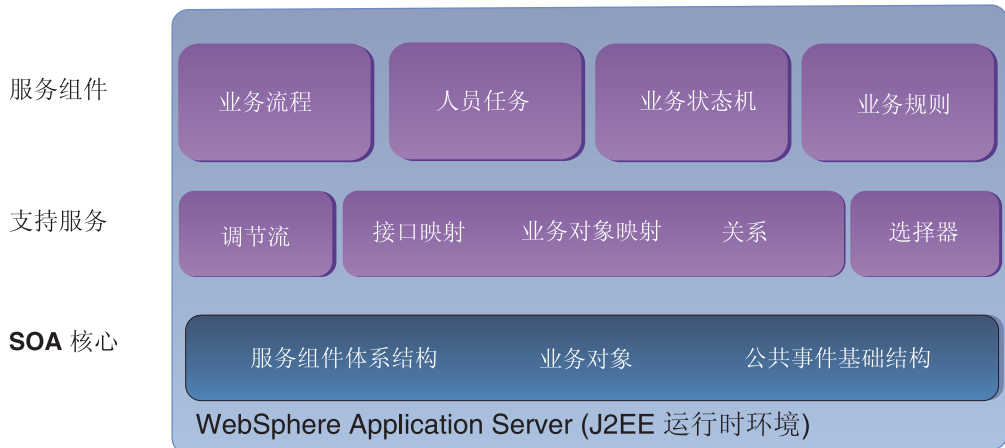


图 2. 基于 WebSphere Process Server 组件的框架

服务组件体系结构

服务组件体系结构在 WebSphere Process Server 运行时环境中以面向服务的方式提供了业务交易的所有元素。

服务组件体系结构 (SCA) 是这样一种体系结构：在该体系结构中，以面向服务的方式表示业务交易的所有元素，例如，访问 Web Service、企业信息系统 (EIS) 服务资产、业务规则、工作流程和数据库等。

服务组件体系结构将业务逻辑与实现分开，以便您可以集中精力组装集成应用程序，而不必知道有关实现的详细信息。业务流程的实现包含在服务组件中。

可在 IBM WebSphere Integration Developer 工具中以图形方式组装服务组件，并且稍后可添加实现。通过服务组件体系结构编程模型，开发者不必掌握很多有关 Java™、J2EE 或在特定方案中语言概念核心集实现过程的知识。目前，采用其他编程语言开发业务应用程序的开发者都熟悉此语言概念核心集。这使开发者可以快速轻松地集成技术。

从传统应用程序开发环境转过来的开发者将会少走很多学习上的弯路；他们可以使用此编程模型快速成为高效多产的开发者。服务组件体系结构编程模型还可帮助有经验的 J2EE 开发者变得更为高效。

服务组件体系结构支持多种标准服务实现类型：

- Java 对象，用于实现 Java 类。与在 Java 编程语言中一样，处于运行时的 Java 组件的实例被称为 Java 对象。
- 用于实现业务流程的业务流程组件。实现语言是业务流程执行语言（BPEL）及其 IBM 扩展。
- 人员任务组件，用于提供和实现通常由人员在业务流程或集成应用程序中执行的任务。
- 业务状态机组件，当应用程序与具有一组状态的工件配合工作时将使用这些组件。状态机定义了工件在某个时间点可执行的操作。
- 业务规则组件，用于根据上下文来确定业务流程的结果，并可设计为 if-then 规则、决策表或决策树。业务流程中的业务规则允许应用程序对不断变化的业务情况快速作出响应。规则独立于业务流程本身，您可以随时更改它们，而不必重复流程。

服务限定词管理服务客户机与 WebSphere Process Server 运行时环境上的服务之间的交互。服务限定词是服务质量规范，它们为传输优先级、传递可靠性级别、事务管理以及安全性级别定义了一组应用程序所需的通信特征。应用程序通过指定服务限定词将其服务质量需求传达给运行时环境。连接 WebSphere Integration Developer 中组装编辑器的组件时，可指定服务质量限定词。当在 WebSphere Process Server 上运行时，这些规范确定客户机如何与目标组件交互。根据指定的限定词，可以在运行时执行其他必需的处理。

服务组件体系结构中的导入和导出功能为 WebSphere Process Server 定义了服务模块的外部接口或访问点。可导入和导出同一应用程序中的其他模块，也可以导入和导出企业信息系统（EIS）上的其他应用程序。这允许您使用 IBM WebSphere Adapters。有关更多信息，请参阅导入、导出和适配器。

WebSphere Process Server 解决方案依靠底层的 WebSphere Application Server 功能来处理事务、安全性和工作负载管理，以便提供可伸缩的集成环境。

对于业务流程，WebSphere Process Server 为涉及多个资源管理器的事务提供支持，这些资源管理器使用两阶段落实流程来确保原子、一致、孤立和持久的（ACID）属性。短时间运行的流程（单个事务）和长时间运行的流程（多个事务）均可使用此功能。通过在 WebSphere Integration Developer 中修改事务边界，您可以将业务流程中的多个步骤分组到一个事务中。

由于并非所有服务调用都支持两阶段落实事务，所以 WebSphere Process Server 还将包含恢复功能。如果在运行集成应用程序期间发生故障，那么服务器将检测该故障并允许管理员管理来自失败事件管理器的失败事件。

服务数据对象和业务对象

服务数据对象和业务对象定义了在服务组件体系结构中定义的组件之间流动的数据。

服务数据对象（SDO）技术是一项开放式标准，它使应用程序能够以统一方式处理来自不同种类的数据源的数据。SDO 合并了 J2EE 模式，但简化了 J2EE 数据编程模型。

作为内置到 WebSphere Process Server 中的 IBM WebSphere Application Server 功能的一部分，SDO 提供了用于简化 J2EE 数据编程模型的数据应用程序开发框架。

WebSphere Process Server 包含的业务对象是基于数据访问技术的增强型 SDO。SDO 提供了用于描述异构数据（例如，JDBC ResultSet 和 XML 模式描述的数据）的通用方

法。业务对象包含一些对集成解决方案极为重要的扩展，它们可用于进一步描述在服务组件体系结构服务之间交换的数据。业务对象是 WebSphere Process Server 的面向服务体系结构 (SOA) 核心的一部分。

业务对象是一组属性，描述了业务实体（例如职员）、数据操作（例如创建或更新操作）和处理数据的指示信息。集成应用程序的组件使用业务对象来交换信息和触发器操作。由于业务对象可表示多种数据，所以它们是很灵活的。例如，除了支持传统集成服务器的数据标准模型之外，它们还可以表示从同步 EJB 会话 Bean 外观或同步业务流程返回的数据，然后可绑定到 IBM WebSphere Portal portlet 和 JSF 组件。

业务对象是下列各项的主要机制：表示业务实体或记录文字消息定义、启用从具有标量属性的简单基本对象到大型复杂层次结构或对象图的一切事物。

在 WebSphere Process Server 中，业务对象框架由下列元素组成：

- 业务对象定义
- 业务图定义
- 业务对象元数据定义
- 业务对象服务（服务 API）

业务对象定义是用于指定业务对象类型的名称、有序属性集、属性、版本号和特定于应用程序的文本。业务图定义是添加到简单业务对象或业务对象层次结构周围的包装器，可提供诸如传递与业务图中业务对象相关的更改摘要和事件摘要信息之类的其他功能。业务对象元数据定义是可添加到业务对象定义的元数据，用于在 WebSphere Process Server 上运行时增强业务对象定义的值。此元数据可作为著名的 `xs:annotation` 和 `xs:appinfo` 元素添加至业务对象的 XML 模式定义中。业务对象服务是一组功能，在服务数据对象所提供的基本功能的基础上提供。诸如创建、复制、等式和串行化之类的服务就是一些示例。

有关 WebSphere Application Server 服务数据对象的更多信息，请参阅 WebSphere Application Server Network Deployment 文档。

相关概念

第 16 页的『业务对象映射』

业务对象映射是一种使业务对象互相关联的方法。

WebSphere Process Server 中的公共事件基础结构

公共事件基础结构是 WebSphere Process Server 中嵌入的一项技术，可提供基本事件管理服务。

公共事件基础结构的基础结构部分是 WebSphere Process Server 中的底层 IBM WebSphere Application Server 功能的一部分。事件发出功能是 WebSphere Process Server 的附加功能。

公共事件基础结构 (CEI) 实现了用于创建、传输、持续以及分配业务、系统和网络公共基本事件的一组 API 和基础结构。公共基本事件是一项基于 XML 的规范，它定义用于管理商业企业应用程序中的事件（例如，日志记录、跟踪、管理和业务事件）的机制。




CEI 提供基本的事件管理服务，其中包括合并和保持来自多个不同种类的源的原始事件并将这些事件分发给事件使用者。它提供用于生成、传播、持续和使用表示服务组件

流程的事件的功能。公共基本事件模型是标准的基于 XML 的格式，它定义了这些事件的结构。服务器使用的每类事件均包含多个特定于给定事件类型的标准字段。在某些情况下，它包含服务组件在特定事件点时使用的业务对象数据的封装。

WebSphere Process Server 以近乎独占的方式使用 CEI 中的事件，以启用服务组件监视功能。如果要使用与事件相关的功能，那么必须配置 CEI 服务器，但在这之后不得直接使用 CEI，而是应使用 WebSphere Process Server 中的现有服务。

在 WebSphere Process Server 中，特殊配置的 CEI 服务器（可能是现有流程服务器或另一服务器的一部分）用于所有与事件相关的服务。您必须先创建和部署供 CEI 服务器使用的多个工具，其中包括事件数据库、消息传递引擎、一个或多个企业应用程序以及数据库驱动程序。

相关概念

-  公共事件基础结构
-  公共基本事件模型
-  管理公共事件基础结构

支持服务

IBM WebSphere Process Server 中的支持服务解决了多个连接组件和外部工件时的变换问题。

您可以使用调解流、接口映射、业务对象映射、关系和选择器来集成在 IBM WebSphere Process Server 上运行的应用程序。

一个基于组件的框架能够处理所有样式的集成。

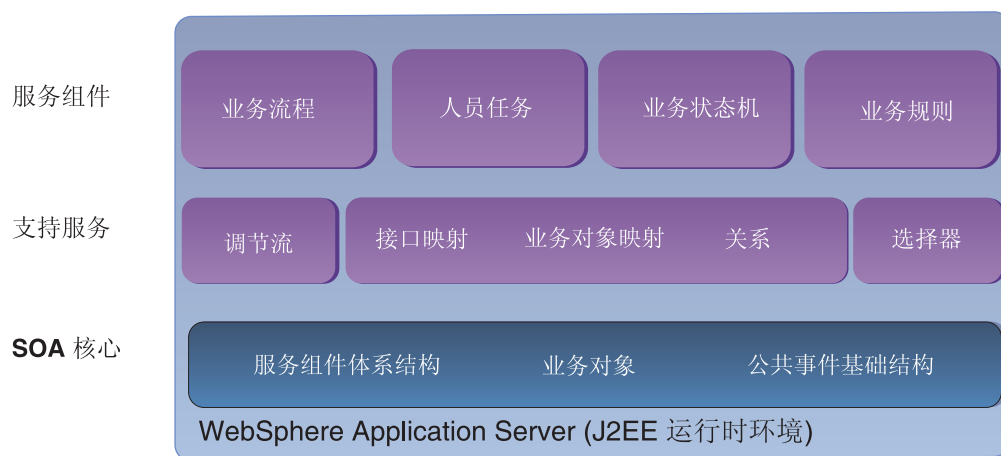


图 3. 基于 WebSphere Process Server 组件的框架

调解流

调解流拦截并修改在现有服务（提供者）与要使用这些服务的客户机（请求者）之间传递的消息。

调解流在导出和导入之间进行调解或干预，以提供消息记录、数据变换和传递等功能。调解流创建于 IBM WebSphere Integration Developer 中，并作为 WebSphere Process Server 中的调解模块的一部分进行部署。

相关概念

第 25 页的第 7 章，『企业服务总线消息传递基础结构』

WebSphere Process Server 包含企业服务总线功能。WebSphere Process Server 支持对面向服务、面向消息和事件驱动的技术进行集成，以在集成企业服务总线中提供基于标准的消息传递基础结构。

接口映射

接口映射协调具有不同接口的组件之间的差别。

接口映射是 WebSphere Process Server 中的支持服务组件，这些组件可以解决并协调其他服务组件体系结构（SCA）组件中接口之间的差别，以使它们能够进行通信。接口映射拥有一流的模式，允许 IBM WebSphere Integration Developer 中的模块设计器通过变换和其他基本操作来协调多个接口的差别。接口映射作为模块的一部分部署在 WebSphere Process Server 上，它们也称为“SCA 模块”。

业务对象映射

业务对象映射是一种使业务对象互相关联的方法。

业务对象映射是 IBM WebSphere Process Server 中的支持服务组件，这些组件根据源业务对象服务组件中的值来为目标业务对象服务组件指定值。一个业务对象成为源，而另一个业务对象成为目标。业务对象映射在源与目标之间建立映射。业务对象映射支持业务对象之间进行“1 对 n”、“多对 1”和“多对 n”映射。这包括映射业务数据以及与业务对象相关联的方面（例如，动词）。

开发者在 IBM WebSphere Integration Developer 中创建和编辑业务对象映射。在运行期间，通过映射可以解决如何在源业务对象与目标业务对象之间表示数据。在运行期间，可以监视 WebSphere Process Server 中发生的映射事件。

相关概念

第 13 页的『服务数据对象和业务对象』

服务数据对象和业务对象定义了在服务组件体系结构中定义的组件之间流动的数据。

关系

关系是用来对业务对象与其他数据之间的关联进行建模和维护的服务。

关系是 IBM WebSphere Process Server 应用程序中的支持服务，用于在两种或更多类型的数据之间建立关联。

关系是业务集成系统中的两个或多个数据实体之间的关联。通常，这些实体是业务对象。关系用来对业务对象之间等价但是以不同方式表示的数据进行变换。

在 WebSphere Process Server 中，关系管理器是一种手动处理关系数据工具，用于更正在自动关系管理中发现的错误或者提供更完整的关系信息。它特别提供了用于检索和修改关系实例数据的工具。关系管理器允许您对关系运行时数据（包括参与者及其数

据) 进行配置、查询、查看和执行操作。您可使用关系设计器来创建关系定义。在运行期间, 将使用关联来自不同应用程序的信息的数据来填充关系实例。

相关概念

管理关系

关系管理器是一种手动控制和处理关系数据工具, 用于更正自动关系管理中发现的错误或提供更完整的关系信息。它特别提供了用于检索和修改关系实例数据的工具。

选择器

选择器使您可以在运行期间灵活地处理服务组件。

选择器又称为选择器组件, 它是 IBM WebSphere Process Server 中的支持服务, 它执行一个调用, 并允许根据选择标准来调用不同目标。

选择器组件是一个组件, 它提供一种将动态选择机制置于客户机应用程序和一组目标实现之间的方法。

与业务规则相比, 选择器提供了更多灵活性。业务规则是业务的基本部分, 它推动应用程序的一般处理, 调用某些服务来通过应用程序获取数据。例如, 规则可以如下, 在学校开学前两周, 对与我们学校相关的商品提供“返校”特价。选择器执行一个调用, 并允许基于选择标准来调用不同目标。例如, 如果时间刚好在学校开学前, 那么将调用先前的“返校”报价。但是, 如果该季节正好是学校放假期间, 那么将调用“让孩子做好夏天准备”报价。

由于应用程序始终调用同一事物, 所以它是可移植的。业务规则从不改变。根据选择器的不同, 实际的处理会不同, 并会调用不同的服务组件。

相关概念

选择器组件概述

随着业务的变化, 用于驱动业务的业务流程也必须相应变化。有些变化可能要求在不更改流程设计的情况下, 某些流程返回与最初设计的结果不同的结果。选择器组件提供了实现这种灵活性的框架。

服务组件

所有在 IBM WebSphere Process Server 上运行的集成工件 (例如, 业务流程、业务规则和人员任务) 均表示为具有严格定义接口的组件。

在服务组件体系结构 (SCA) 中, 服务组件 (又称为 SCA 组件) 将定义服务实现。每个服务组件都有接口, 并且可连接在一起以形成部署到 WebSphere Process Server 的模块。

这会创建灵活的运行时环境, 并使您可以更改应用程序的任何部分, 而不会影响其他部分。例如, 只需通过替换组装图中的服务组件, 无需更改业务流程或业务流程的调用者, 就可以将提供审批的人员任务替换为提供自动审批的业务规则。

使用下列编程构造, 服务组件可以与现有应用程序进行交互:

- Java Beans
- Enterprise Java Beans
- Web Service
- JMS 消息

此外，服务组件可与带有 IBM WebSphere Adapters 的企业信息系统（EIS）上的其他应用程序交互。

在支持服务和面向服务的体系结构核心的运行时基础结构上，WebSphere Process Server 提供各种可供使用的 SCA 组件，这些组件可用在集成应用程序中。

一个基于组件的框架能够处理所有样式的集成。

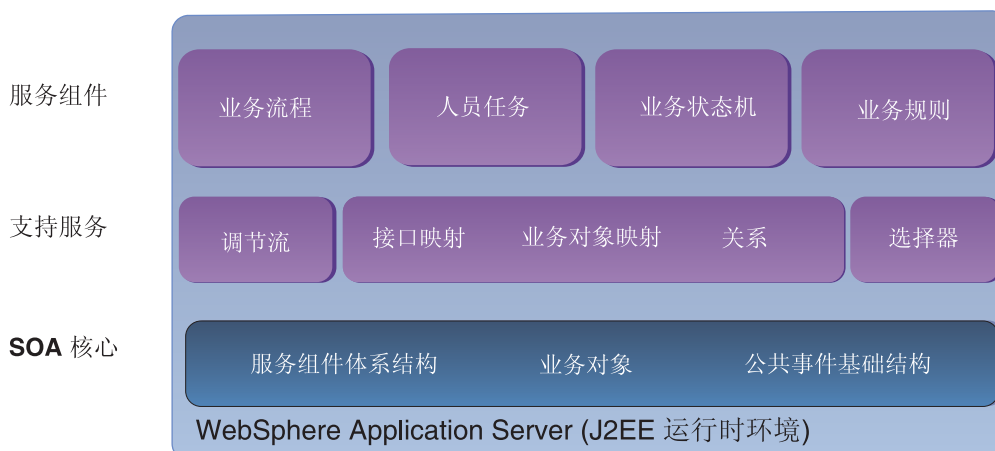


图 4. 基于 WebSphere Process Server 组件的框架

业务流程


业务流程是一些服务组件，提供了用于集成企业服务的主要方法。

业务流程是一个组织用来实现更大业务目标的任何系统或过程。将它分解后，您会看到业务流程实际上是一系列单独的任务，并且每个任务都按特定的顺序执行。作为在 IBM WebSphere Process Server 上运行的应用程序的必要组成部分，业务流程提供了用于集成企业服务的主要方法。

业务流程组件实现完全受支持的 Web Service 业务流程执行语言（BPEL）引擎。WebSphere Process Server 在 WebSphere Application Server 的基础上提供了业务流程编排引擎。您可以凭借对高度可伸缩的基础结构中长时间运行和短时间运行的业务流程的高级支持，在简单的开发模型中开发和部署复杂的业务流程。可以在 WebSphere Integration Developer 中创建 BPEL 模型，也可以从您在 WebSphere Business Modeler 中创建的业务模型中导入 BPEL 模型。

Web Service 业务流程执行语言（BPEL）可用于编排业务流程流。业务流程集成服务在 BPEL4WS V1.1 基础之上构建，并且添加了即将发行的 WS-BPEL V2.0 规范的主要功能。

相关概念

 关于业务流程

人员任务

人员任务是一些服务组件，可用于为职员分配工作或者调用其他服务。

人员任务是一个由人完成的工作单元，它通常涉及到与其他服务交互，因此成为更大业务目标内的任务。

WebSphere Process Server 中提供的人员任务管理器支持在运行时创建和跟踪任务。现有 LDAP 目录（以及操作系统存储库和 WebSphere 用户注册表）均可用于访问用户和组信息。WebSphere Process Server 支持包括电子邮件通知在内的人员任务的多级升级。它还包含用于管理人员任务的 Web 客户机、一组可用于创建定制客户机或将人员任务功能嵌入到其他 Web 应用程序的 Java Server Faces (JSF) 组件。

人员任务服务组件允许根据角色来分配、调用任务和对任务进行升级。

相关概念

 人员任务

业务状态机

业务状态机是一些服务组件，它们允许您根据状态和事件而不是连续的业务流程模型来表示业务流程。

业务状态机指定对象或交互在对事件作出响应时经历的状态顺序、响应顺序以及所执行操作的顺序。

在 IBM WebSphere Integration Developer 中创建和编辑业务状态机，在运行期间在 IBM WebSphere Process Server 中监视业务状态机。

相关概念

 业务状态机事件

业务规则

业务规则是用于声明业务中必须满足的策略或条件的服务组件。

业务规则说明如何将业务策略或实践应用于业务活动。它是用来控制商业实践行为或者对商业实践强加某种结构的任何对象。业务规则可以强制执行业务策略、在组织内部建立共同遵守的准则或者控制在业务环境中进行的访问。

业务规则使业务流程更为灵活。由于业务规则可根据上下文来确定流程的结果，因此在业务流程中使用业务规则将使应用程序能够对瞬息变化的业务情况迅速作出响应。

IBM WebSphere Integration Developer 支持业务规则制订。IBM WebSphere Process Server 包括一个业务规则管理器，该管理器是一个基于 Web 的运行时工具，可供业务分析员根据业务需要的指示更新业务规则，而不影响其他组件或服务组件体系结构 (SCA) 服务。

相关概念

业务规则概述

使用业务规则来控制商业实践的行为。

第 5 章 导入、导出和适配器

服务组件体系结构中的导入和导出功能为 WebSphere Process Server 定义了服务模块的外部接口或访问点。可导入和导出同一应用程序中的其他模块，也可以导入和导出企业信息系统（EIS）上的其他应用程序。

导入过程确定模块外部的服务，使它们可从模块中调用。导出过程允许模块中的组件向外部客户机提供其服务。

导入或导出使模块能够访问其他模块，并使应用程序能够访问 EIS 系统上的应用程序，就像它们是本地组件一样。这允许您使用 IBM WebSphere Adapters。

WebSphere Adapters 提供了一种面向服务的方法来进行 EIS 集成。WebSphere Adapters 符合 J2EE 连接器体系结构（JCA 1.5）。JCA 是用于建立 EIS 连接的 J2EE 标准。EIS 导入和 EIS 导出使 SCA 组件的模块外部的服务具有统一形式。这就允许组件与各种使用一致的 SCA 编程模型的外部 EIS 系统通信。将 WebSphere Adapters 从已导入的 RAR 文件组装到 WebSphere Integration Developer 中，然后作为“企业归档”（EAR）文件导出，再部署到 WebSphere Process Server 上。

WebSphere Adapters 包括以下各项：

- IBM WebSphere Adapter for E-mail V6.1
- IBM WebSphere Adapter for FTP V6.1
- IBM WebSphere Adapter for Flat Files V6.1
- IBM WebSphere Adapter for JDBC V6.1
- IBM WebSphere Adapter for SAP Software V6.1
- IBM WebSphere Adapter for Siebel Business Applications V6.1
- IBM WebSphere Adapter for Oracle E-Business Suite V6.1
- IBM WebSphere Adapter for JD Edwards EnterpriseOne V6.1

WebSphere Adapters 是 WebSphere Integration Developer 中包括的组件。

有关 WebSphere Adapters 的更多信息，请参阅 IBM WebSphere Business Process Management 信息中心中的 WebSphere Integration Developer 文档。

导入和导出需要可指定从模块传输数据的方法的绑定信息。WebSphere Integration Developer 中的组装编辑器设置导入和导出、列示受支持的绑定并简化它们的创建过程。属性视图将显示绑定信息。

第 6 章 WebSphere Process Server 中的部署环境

WebSphere Process Server 允许您将服务组件体系结构 (SCA) 模块的部署环境作为一组服务器来管理。WebSphere Process Server 附带的 WebSphere Application Server Network Deployment 功能为这组服务器提供了元素。

WebSphere Process Server 环境包括互连服务器或拓扑的布局，它支持 WebSphere Process Server SCA 模块 (其中包括业务流程编排器、业务规则、调解和关系)。此拓扑由在一个计算机系统上运行的一个服务器进程组成，它也可以由在多个计算机系统上运行的多个服务器进程组成。服务器进程是作为 SCA 模块部署的组件的运行时环境。在 WebSphere 产品中 (包括 WebSphere Process Server)，服务器进程是 Java 虚拟机 (JVM)。

如果环境由一个系统上的一个服务器进程组成，那么设置的服务器进程称为独立服务器。独立服务器并不与其他服务器进程互连，它的容量仅限于一个计算机系统上的资源，并且不包括故障转移支持。它也是最容易设置的环境。

如果环境中包含多个服务器进程，那么您很可能在单元中设置集群环境。一个单元是分布式计算环境中的一个管理域，由 SCA 模块以及支持这些 SCA 模块所需要的资源组成。在部署环境中，将集中管理服务器进程 (通常位于不同的物理计算机系统上)。一个 Deployment Manager 可以管理多个部署环境。

将部署环境与集群配合使用具有下列优点：

- 容易管理：您可以具有用于配置 SCA 模块的一个视图、支持 SCA 模块的服务器进程的一个视图、用于控制 SCA 模块的运行时操作 (例如，启动、停止、创建和删除) 的一个控制点。
- 工作负载均衡：通过在多个服务器上运行应用程序映像，集群可在集群中的服务器上均衡应用程序工作负载。
- 应用程序的处理能力：通过配置其他服务器硬件作为支持应用程序的集群成员，您可以提高应用程序的处理能力。
- 应用程序可用性：当服务器出现故障时，应用程序将继续在集群中的其他服务器上处理工作，从而允许在不影响应用程序用户的情况下继续进行恢复工作。
- 可维护性：您可以停止服务器以进行有计划的维护，而无需停止应用程序处理。
- 灵活性：您可以根据需要使用管理控制台来添加或除去容量。

第 7 章 企业服务总线消息传递基础结构

WebSphere Process Server 包含企业服务总线功能。WebSphere Process Server 支持对面向服务、面向消息和事件驱动的技术进行集成，以在集成企业服务总线中提供基于标准的消息传递基础结构。

可用于企业应用程序的企业服务功能不仅提供传输层，而且还提供调解支持以加快服务交互。企业服务总线围绕着开放式标准和面向服务的体系结构（SOA）进行构建。它基于健壮的 J2EE 1.4 基础结构以及由 IBM WebSphere Application Server Network Deployment 提供的相关平台服务。

WebSphere Process Server 与 IBM WebSphere Enterprise Service Bus 基于相同的技术。此功能是 WebSphere Process Server 的底层功能的一部分，不需要其他 WebSphere Enterprise Service Bus 许可证即可利用这些功能。

但是，您可以在企业范围部署 WebSphere Enterprise Service Bus 的其他独立许可证，以扩展 WebSphere Process Server 支持的流程集成解决方案的连接范围。例如，WebSphere Enterprise Service Bus 可安装在距 SAP 应用程序较近的位置，以主管 IBM WebSphere Adapter for SAP 和变换 SAP 消息，然后通过网络将该信息发送给 WebSphere Process Server 设计的业务流程。

相关概念

第 15 页的『调解流』

调解流拦截并修改在现有服务（提供者）与要使用这些服务的客户机（请求者）之间传递的消息。



消息服务客户机

WebSphere Process Server 提供了 Message Service Client for C/C++ 和 .NET，它使非 Java 应用程序能够连接至企业服务总线。

第 26 页的『调解模块』

调解模块是可以更改服务请求的格式、内容或目标的服务组件体系结构（SCA）模块。

第 28 页的『调解原语』

调解组件对服务组件之间的消息流执行操作。调解组件的功能是由调解原语实现的，调解原语将实现标准的服务实现类型。

第 31 页的『服务消息对象』

服务消息对象（SMO）提供了一个抽象层，用于处理和控制与服务之间交换的消息。

消息服务客户机

WebSphere Process Server 提供了 Message Service Client for C/C++ 和 .NET，它使非 Java 应用程序能够连接至企业服务总线。

Message Service Clients for C/C++ and .NET 提供了一个称为 XMS 的 API，它与 Java 消息服务（JMS）API 具有相同的一组接口。Message Service Client for C/C++ 包含

XMS 的两种实现，一种实现供 C 应用程序使用，另一种实现供 C++ 应用程序使用。Message Service Client for .NET 包含 XMS 的完全管理实现，任何适用于 .NET 的语言都可以使用这种实现。

还可以安装和使用 WebSphere Application Server Network Deployment V6 中的 J2EE 客户机支持，包括 Web Service 客户机、EJB 客户机和 JMS 客户机。

相关概念

企业服务总线消息传递基础结构

WebSphere Process Server 包含企业服务总线功能。WebSphere Process Server 支持对面向服务、面向消息和事件驱动的技术进行集成，以在集成企业服务总线中提供基于标准的消息传递基础结构。

调解模块

调解模块是可以更改服务请求的格式、内容或目标的服务组件体系结构（SCA）模块。

调解模块将对服务请求者与服务提供者之间传递的消息进行处理。您可以将消息传递给不同的服务提供者，还可以修正消息的内容或格式。调解模块可以提供一些功能（例如，消息日志记录）以及满足您的要求的错误处理。

您可以从 WebSphere Process Server 管理控制台动态地更改调解模块的某些方面，而不必重新部署该模块。

调解模块的组件

调解模块包含下列各项：

- 导入：用于定义 SCA 模块与服务提供者之间的交互。它们允许 SCA 模块将外部服务当作本地服务来调用。您可以查看 WebSphere Process Server 中的调解模块导入和修改绑定。
- 导出：用于定义 SCA 模块与服务请求者之间的交互。它们允许 SCA 模块提供服务并定义 SCA 模块的外部接口（访问点）。可以从 WebSphere Process Server 中查看调解模块导出。
- SCA 组件：它们是诸如调解模块的 SCA 模块的构建块。可以使用 WebSphere Integration Developer 以图形方式创建和定制 SCA 模块和组件。在部署调解模块之后，可以从 WebSphere Process Server 管理控制台中定制该调解模块的某些方面，而不必重新部署该模块。

通常，调解模块包含特定类型的 SCA 组件（称为*调解流组件*）。调解流组件定义调解流。调解模块只能包含一个调解流组件。

一个调解流组件可以只包含一个调解原语、包含许多调解原语或者不包含任何调解原语。WebSphere Process Server 支持所提供的一组调解原语，它们提供了用于消息传递和变换的功能。如果需要使调解原语具有更高的灵活性，可以使用定制调解原语来调用定制逻辑。

不包含调解流组件的调解模块的作用是将服务请求从一种协议变换为另一种协议。例如，虽然可以使用 SOAP/JMS 来发出服务请求，但是可能需要变换为 SOAP/HTTP 之后才能发送。

注：可以从 WebSphere Process Server 中查看调解模块和对这些模块进行某些更改。但是，不能查看或更改 WebSphere Process Server 模块中的 SCA 组件。使用 WebSphere Integration Developer 来定制 SCA 组件。

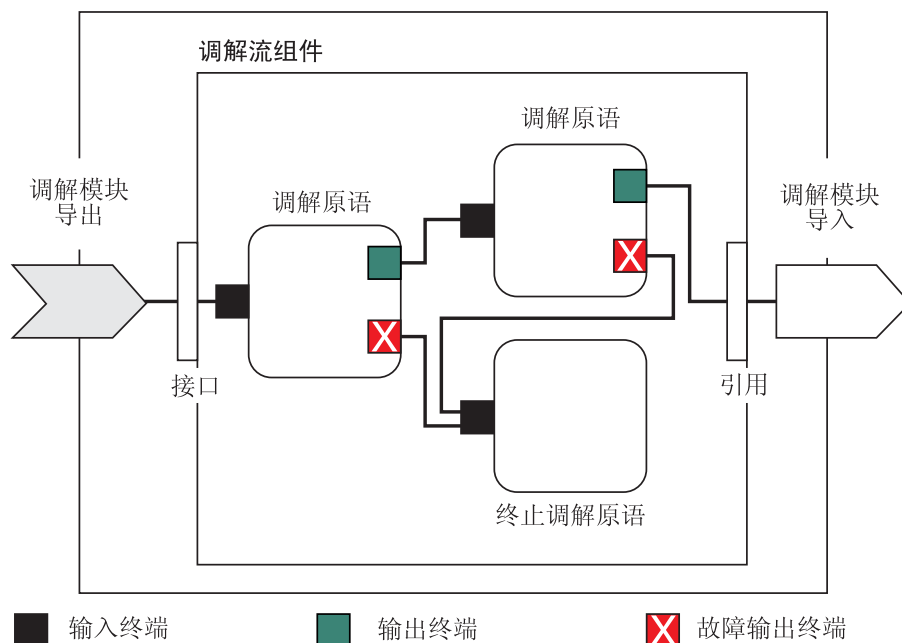


图 5. 调解模块的简单示例. 调解模块中包含一个调解流组件，该调解流组件中包含调解原语。

- 属性

调解原语具有属性，其中一些可以作为 SCA 模块的其他属性显示在管理控制台中。

为了能够从 WebSphere Process Server 管理控制台看到调解原语属性，集成开发者必须将属性标记为已升级。某些属性适合于以管理方式进行配置，WebSphere Integration Developer 将它们描述为可升级属性，其原因是可以将它们从集成周期升级到管理周期。其他属性不适合于管理配置，这是因为修改它们会影响调解流，导致您需要重新部署调解模块。WebSphere Integration Developer 在调解原语的“已升级属性”下列示了一些属性，您可以选择这些属性进行升级。

可以使用 WebSphere Process Server 管理控制台来更改已升级属性的值，而不必重新部署调解模块或重新启动服务器或模块。

除非 Deployment Manager 单元中发生更改，否则新调用的调解流将立即使用属性更改。如果 Deployment Manager 单元中发生更改，那么这些更改在每个节点中都将在该节点完成同步之后就生效。未完成的调解流将继续使用先前的值。

注：如果您要更改调解原语的属性名和类型，而不是更改属性值，则应使用 WebSphere Integration Developer。

部署调解模块

调解模块是使用 WebSphere Integration Developer 创建的，通常被部署到 WebSphere Process Server 的一个企业归档（EAR）文件中。

您可以在部署时更改已升级属性的值。

可以从 WebSphere Integration Developer 导出调解模块，这会导致 WebSphere Integration Developer 将调解模块打包在一个 Java 归档 (JAR) 文件内，并将该 JAR 文件打包在一个 EAR 文件内。然后，您可以通过从管理控制台安装新的应用程序来部署该 EAR 文件。

可以将调解模块认为是一个实体。但是，SCA 模块是由存储在 JAR 文件中的许多 XML 文件定义的。

包含调解模块的 EAR 文件示例

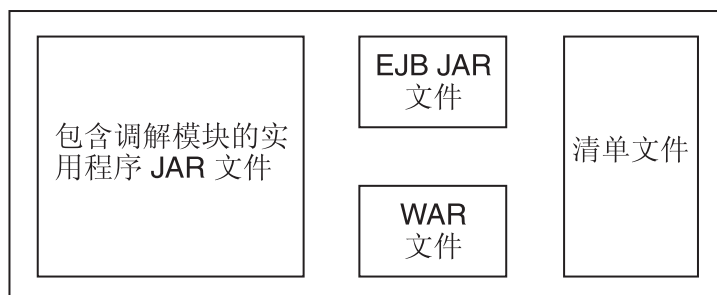


图 6. 包含调解模块的 EAR 文件的简单示例。EAR 文件包含 JAR。实用程序 JAR 文件包含调解模块。

相关概念

企业服务总线消息传递基础结构

WebSphere Process Server 包含企业服务总线功能。WebSphere Process Server 支持对面向服务、面向消息和事件驱动的技术进行集成，以在集成企业服务总线中提供基于标准的消息传递基础结构。

调解原语

调解组件对服务组件之间的消息流执行操作。调解组件的功能是由调解原语实现的，调解原语将实现标准的服务实现类型。

一个调解组件具有一个或多个流。例如，一个流用于请求，另一个流用于应答。

WebSphere Process Server 支持所提供的一组调解原语，它们实现已部署到 WebSphere Process Server 中的调解模块的标准调解功能。如果需要特殊的调解功能，则可以开发您自己的定制调解原语。

调解原语定义一个“输入”操作，用来处理由服务消息对象 (SMO) 表示的消息。调解原语还可以定义一些“输出”操作，用来将消息发送至另一个组件或模块。

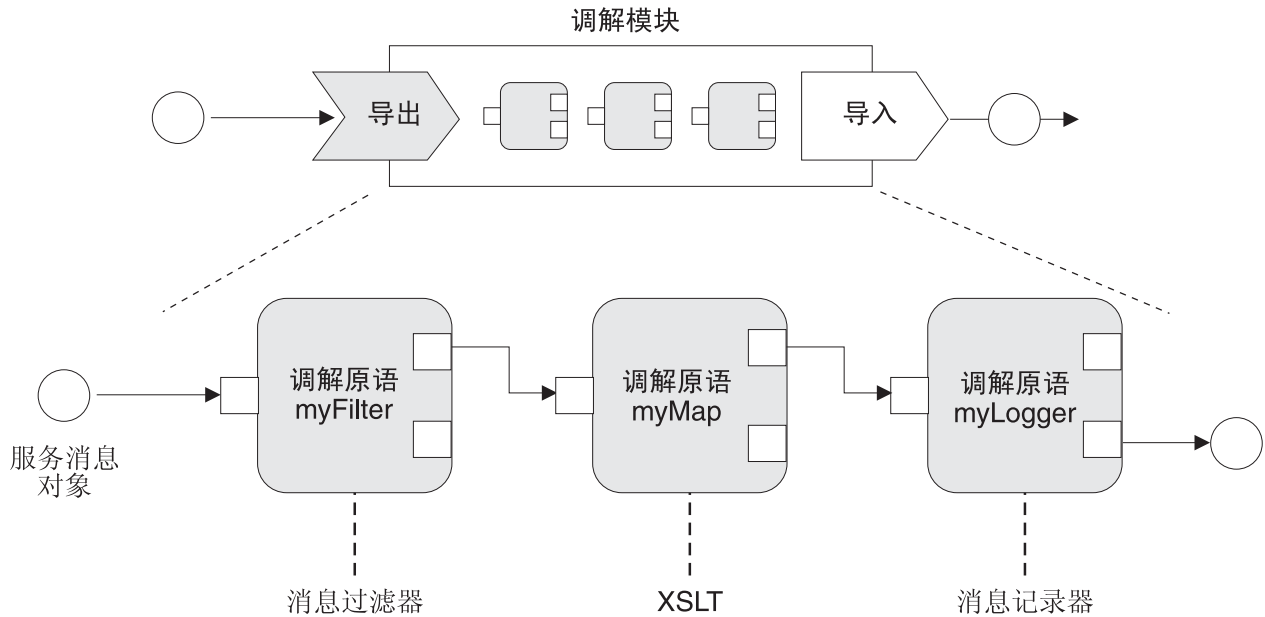


图 7. 包含三个调解原语的调解模块

可以使用 WebSphere Integration Developer 来配置调解原语和设置它们的属性。这些属性中的某些属性在经过提升之后，运行时管理员就可以看到它们。

WebSphere Integration Developer 还允许您以图形方式建立调解组件的模型以及从调解原语来组装调解组件，还可以使用调解组件组装调解模块。

受支持的调解原语

WebSphere Process Server 支持下面这一组调解原语：

业务对象映射

变换消息。

- 使用业务对象映射来定义消息变换，可以重复使用业务对象映射。
- 允许您使用业务对象映射编辑器以图形方式定义消息变换。
- 可以改变消息的内容。
- 可以将输入消息类型变换为另外的输出消息类型。

定制调解

允许您使用 Java 代码来实现您自己的调解逻辑。定制调解原语既具有用户定义的调解原语的灵活性，又具有预定义的调解原语的简单性。可以通过下列方法来创建复杂的变换和路由模式：

- 创建 Java 代码。
- 创建您自己的属性。
- 添加新的终端。

可以从定制调解原语中调用服务，但是“服务调用”调解原语旨在调用服务和提供其他功能（例如，重试）。

数据库查询

使用用户提供的数据库中的信息来修改消息。

- 必须设置数据库查询调解原语要使用的数据库、数据源和任何服务器认证设置。可使用管理控制台来帮助您完成此任务。
- 只能从一个表中读取数据库查询调解原语。
- 指定的键列必须包含唯一值。
- 值列中的数据必须是 Java 原语或 Java 字符串（或者能够通过强制类型转换为 Java 原语或 Java 字符串）。

端点查询

通过在存储库中搜索服务端点来允许请求的动态路由选择。

- 从本地或远程的 WebSphere Service Registry and Repository (WSRR) 中检索服务端点信息。
- 可以从 WSRR 管理控制台进行注册表更改。
- WebSphere Process Server 需要知道要使用的注册表，因此，您必须使用 WebSphere Process Server 管理控制台来创建 WSRR 访问定义。

事件发射器

允许您从调解流组件中发送事件，从而提高监控能力。

- 您可以在 WebSphere Process Server 上使用“公共基本事件”（CBE）浏览器来查看事件发射器事件。
- 出于性能方面的考虑，您应当仅在调解流中的重要位置才发送事件。
- 可以定义事件包含的消息部件。
- 事件以公共基本事件的形式发送给公共事件基础结构服务器。
- 为了充分利用事件发射器信息，事件使用者需要了解公共基本事件的结构。公共基本事件具有整体模式，但这不会对特定于应用程序的数据建立模型，这些数据包含在扩展数据元素中。要建立扩展数据元素的模型，WebSphere Integration Developer 工具为每个已配置的事件发射器调解原语生成公共事件基础结构事件目录定义文件。事件目录定义文件是用于帮助您的导出工件，WebSphere Integration Developer 或 WebSphere Process Server 运行时并不使用这些定义文件。在创建应用程序以处理事件发射器事件时，您应该参阅事件目录定义文件。
- 可以从 WebSphere Process Server 中指定其他监视手段。例如，您可以监视从导入和导出中发射的事件。

失败 停止流中的特定路径并产生异常。

内部使用

可帮助聚集（组合）消息。

- 只能与调解原语“外部使用”组合使用。
- 调解原语“外部使用”和“内部使用”允许将数据聚集到一条输出消息中。
- 在到达决策点之前，调解原语“内部使用”将一直接收消息，然后输出一条消息。
- 应使用共享上下文来保存聚集数据。

外部使用

可帮助分割和聚集（组合）消息。

- 调解原语“外部使用”和“内部使用”允许将数据聚集到一条输出消息中。

- 采用迭代方式时，“外部使用”调解原语允许您遍历包含重复元素的单条输入消息。该元素每次出现时，都会发送一条消息。
- 应使用共享上下文来保存聚集数据。

消息元素设置器

提供用于设置消息内容的简单机制。

- 可以更改、添加或删除消息元素。
- 它不会更改消息的类型。

消息过滤器

根据消息内容的不同，沿着不同的路径传递消息。

消息记录器

将消息记录在关系数据库中。消息是作为 XML 存储的，因此，支持 XML 的应用程序以后也可以处理这些数据。

- 数据库模式（表结构）由 IBM 定义。
- 缺省情况下，“消息记录器”调解原语使用公共数据库。运行时将 jdbc/mediation/messageLog 处的数据源映射至公共数据库。

服务调用

在调解流中调用服务，而不是等到调解流结束再使用调出机制来调用服务。

- 如果该服务返回故障，那么您可以重试此服务，也可以调用另一个服务。
- 调解原语“服务调用”是一个功能强大的调解原语，可以将它单独用于简单的服务调用，也可以将它与其他调解原语一起用于复杂的调解。

设置消息类型

在集成开发期间，允许将强类型的消息字段当作弱类型的消息字段来处理。如果一个字段中可以包含多种类型的数据，那么此字段是一个弱类型字段。如果已经知道一个字段的类型和内部结构，那么此字段是一个强类型字段。

- 在运行时，调解原语“设置消息类型”允许您检查消息的内容是否与您期望的数据类型相匹配。

终止 终止流中的特定路径，而不会产生异常。

XSL 变换

变换消息。

- 允许您执行可扩展样式表语言（XSL）变换。
- 使用 XSLT 1.0 变换来变换消息。变换是针对经过 XML 序列化后的消息进行的。

相关概念

企业服务总线消息传递基础结构

WebSphere Process Server 包含企业服务总线功能。WebSphere Process Server 支持对面向服务、面向消息和事件驱动的技术进行集成，以在集成企业服务总线中提供基于标准的消息传递基础结构。

服务消息对象

服务消息对象（SMO）提供了一个抽象层，用于处理和控制在服务之间交换的消息。

SMO 模型

调解原语将消息作为 SMO 进行处理。SMO 是经过增强的服务数据对象 (SDO)，而 SMO 模型是使用 SDO 数据对象来表示消息的一种模式。SMO 包含下列各组数据的表示：

- 与消息相关联的头信息。例如，Java 消息服务 (JMS) 头，如果消息已使用 JMS API 进行传递；或 MQ 头，如果消息来自 WebSphere MQ。
- 消息有效内容。消息有效内容是在服务端点之间交换的应用程序数据。
- 上下文信息（除了消息有效内容之外的数据）。

此信息的所有内容均作为 SDO 数据对象进行访问，并且存在用来指定 SMO 整体结构的模式声明。该模式是由 WebSphere Integration Developer 生成的。

SMO 内容

所有 SMO 都具有相同的基本结构。该结构由称为 ServiceMessageObject 的根数据对象组成，包含用来表示头、主体和上下文数据的其他数据对象。头、主体和上下文的准确结构取决于您在集成开发期间是如何定义调解流的。调解流在运行时用于在服务之间进行调解。

SMO 头包含源于特定导出或导入绑定的信息（绑定用于指定消息格式和协议详细信息）。消息可以来自于许多消息源，因此 SMO 必须能够携带不同种类的消息头。将处理下列类型的消息头：

- Web Service 消息头。
- 服务组件体系结构 (SCA) 消息头。
- Java 消息服务 (JMS) 消息头。
- WebSphere MQ 消息头。
- WebSphere Adapters 消息头。

通常，SMO 主体的结构由您在配置调解流时指定的“Web Service 定义语言” (WSDL) 消息确定，SMO 主体用于存储应用程序数据。

SMO 上下文对象是由用户或系统定义的。可以使用用户定义的上下文对象来存储属性，调解原语稍后可以在调解流中使用该属性。在业务对象中定义用户定义的上下文对象的结构，并在请求流的输入节点中使用该业务对象。相关上下文、瞬态上下文和共享上下文都是由用户定义的上下文对象。

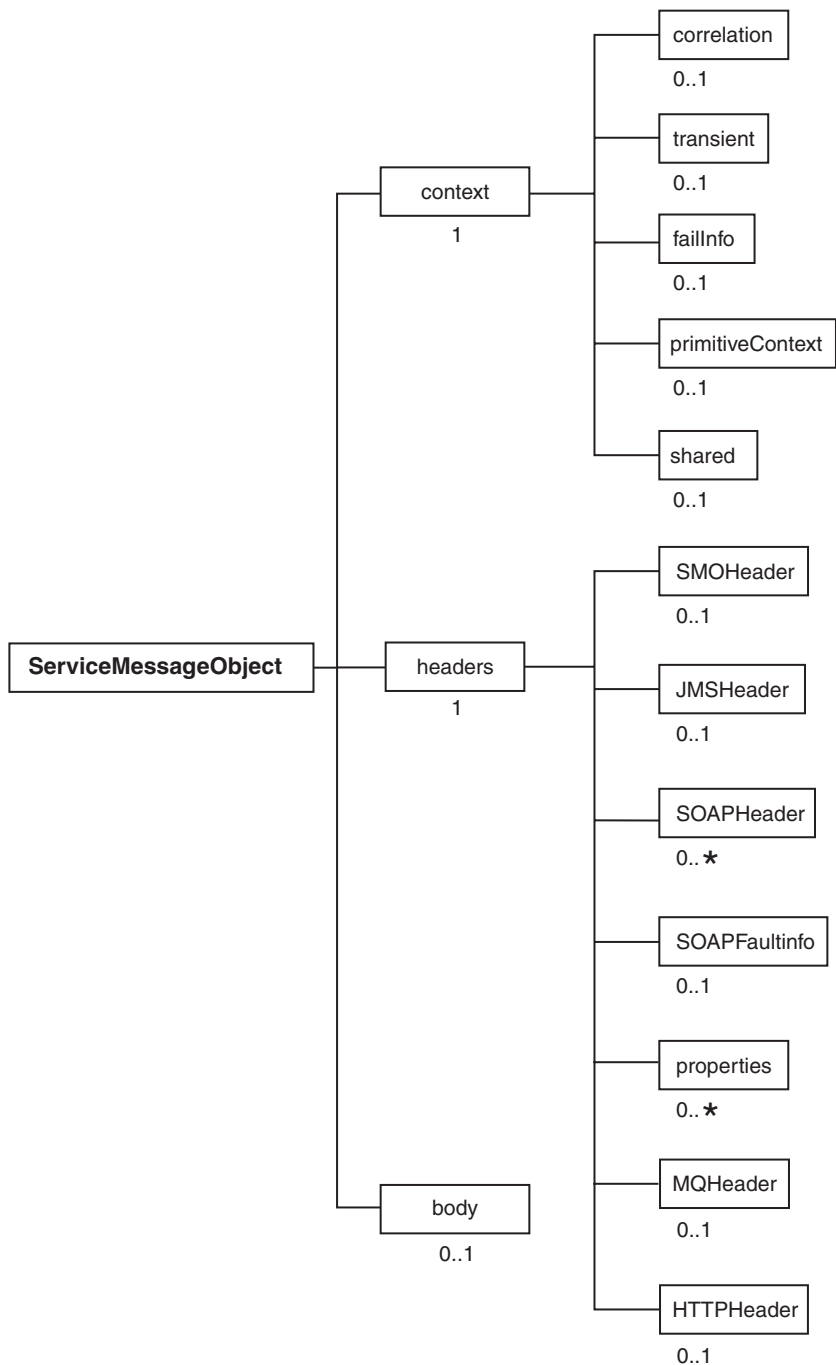


图 8. SMO 结构概述. ServiceMessageObject 的上下文、头和主体

SMO 提供了一个接口用来访问和修改消息头、消息有效内容和消息上下文。

运行时使用 SMO 的方式

运行时将对在交互端点之间传递的消息进行处理。运行时将创建 SMO 对象，调解流使用 SMO 对象来处理消息。

当创建调解流时，WebSphere Integration Developer 将指定每个终端的消息体的类型（输入、输出或失败），还可以选择指定上下文信息的类型。运行时使用此信息将消息转换为指定类型的 SMO 对象。

为了提供动态路由选择，可以使用 WebSphere Service Registry and Repository (WSRR) 或数据库来查询交互端点。可以将 WSRR 查询或数据库查询的结果存储在 SMO 中的特定位置，运行时将从此位置获取动态端点。

相关概念

企业服务总线消息传递基础结构

WebSphere Process Server 包含企业服务总线功能。WebSphere Process Server 支持对面向服务、面向消息和事件驱动的技术进行集成，以在集成企业服务总线中提供基于标准的消息传递基础结构。

第 8 章 管理 WebSphere Process Server 上的应用程序

管理 IBM WebSphere Process Server 涉及准备、监视和修改要将“服务组件体系结构”（SCA）模块作为应用程序和资源部署到的环境，还涉及到使用应用程序和资源本身。

有关管理应用程序的更多信息，请参阅管理 *WebSphere Process Server* PDF 文件。

WebSphere Process Server 提供了多个用于管理运行时环境的界面：

- 管理控制台

管理控制台是一个基于浏览器的界面，可用于监视、更新、停止和启动大量应用程序、服务与用于 WebSphere Process Server 上运行的应用程序的资源。管理控制台还可以用于处理各种关系以及找到并解决失败的 WebSphere Process Server 事件。

管理控制台还为 WebSphere Application Server 和其他客户定义的产品提供管理功能。WebSphere Process Server 管理控制台通常是“集成解决方案控制台”框架的一部分，而在特殊情况下是 WebSphere Application Server 管理控制台的一部分。因此，对于 WebSphere Process Server 和 WebSphere Application Server 来说，许多管理任务（例如，设置安全性、查看日志和安装应用程序）都相同。

- 命令行工具

命令行工具是一些简单的程序，从操作系统命令行提示符处运行这些程序来执行特定任务。通过使用这些工具，可以启动和停止应用程序服务器、检查服务器状态、添加或除去节点以及完成其他任务。WebSphere Process Server 命令行工具包括 `serviceDeploy` 命令，它处理已从 WebSphere Integration Developer 环境中导出的 `.jar`、`.ear`、`.war` 和 `.rar` 文件，并准备将它们安装到生产服务器中。

- WebSphere 管理（wsadmin）脚本编制程序

wsadmin 脚本编制程序是一个非图形命令解释器环境，它使您能够运行采用脚本语言的管理选项以及提交脚本语言程序以便执行。它与管理控制台支持相同的任务。wsadmin 工具用于生产环境和无人照看操作。

- 管理程序

Java 管理扩展（JMX）规范中的一组 Java 类和方法为管理服务组件体系结构（SCA）和业务对象提供支持。每个编程接口都包括其用途的描述、一个用来演示如何使用接口或类的示例，还包括对各个方法描述的引用。

- 业务流程编排器资源管理器

“业务流程编排器资源管理器”是一个独立的 Web 应用程序，它提供了一组用于管理业务流程和人员任务的基本管理功能。可以查看有关流程模板、流程实例、任务实例及其相关对象的信息。您还可以对这些对象执行操作；例如，可以启动新的流程实例，修复和重新启动失败的活动，管理工作项以及删除已完成的流程实例和任务实例。

- 业务流程编排器观察器

业务流程编排器观察器是一个 Web 应用程序，可创建有关在执行业务流程和人员任务期间所发生事件的报告。然后，您可以使用这些报告来评估流程和活动的效率与可靠性。

- 业务规则管理器

业务规则管理器是一个基于 Web 的工具，它可以帮助业务分析员浏览和修改业务规则值。此工具是 WebSphere Process Server 的一个选项，您可以选择在创建概要文件时或者在最初安装服务器之后安装此工具。

相关概念



WebSphere Process Server 的管理控制台

管理控制台是一个基于浏览器的界面，用于管理单元、节点或服务器作用域内的 WebSphere Process Server 应用程序、服务和其他资源。可以在独立的流程服务器中使用管理控制台，还可以在用于管理联网环境中单元内的所有服务器的 Deployment Manager 中使用管理控制台。



管理 WebSphere Process Server

本节中的主题描述了如何管理 WebSphere Process Server 运行时环境，包括部署在此环境中的应用程序和资源。

第 9 章 在 WebSphere Process Server 上开发和部署应用程序

用于在 WebSphere Process Server 上开发和部署集成应用程序的选项包括：在 WebSphere Integration Developer 开发环境中工作、使用服务组件体系结构 API 以及在使用 WebSphere Process Server 的测试或生产服务器环境中启用应用程序。

IBM WebSphere Integration Developer 是 WebSphere Process Server 的开发环境。有关在 WebSphere Integration Developer 中开发集成应用程序的更多信息，请参阅 IBM WebSphere Business Process Management 信息中心。

除了 WebSphere Integration Developer 开发环境以外，还为开发者发布了服务组件体系结构 API。有关服务组件体系结构 API 的更多信息，请参阅“开发和部署模块”。

模块，也称为服务组件体系结构（SCA）模块，将它们部署到 WebSphere Process Server 之后，可以确定哪些工件包含在已部署至运行时环境的企业归档（EAR）文件中。

在 WebSphere Integration Developer 中，您可以使用组装编辑器将服务分组到模块中，并指定该模块对外部使用者显示哪些服务。然后，连接模块以形成完整的集成解决方案。将集成逻辑包括在各个模块中，以便在对一个模块中的服务进行更改，但该模块的接口保持不变时，所作的更改不会影响解决方案中包括的任何其他模块。

部署是指在测试环境或生产环境中启用应用程序（SCA 模块）的操作。虽然对于这两种环境来说部署是相同的，但是每种环境中的部署任务还是有些差别。由于最好是在测试服务器上测试对 SCA 模块所作的任何更改之后再将这些更改落实到生产环境中，因此，应先使用 WebSphere Integration Developer 将这些模块部署到测试环境中，然后再使用 WebSphere Process Server 将这些模块部署到生产环境中。

如果需要部署许多应用程序文件，也就是要安装许多 SCA 模块，那么您可能需要使用批处理文件。有关批处理文件的更多信息，请参阅“在生产服务器上安装模块”和“使用 Apache Ant 任务部署应用程序”。

相关概念



Developing and deploying modules

开发和部署模块是基本的任务。



在生产服务器上安装模块

本主题描述从测试服务器上获取应用程序并将其部署到生产环境时需要执行的步骤。



使用 ANT 任务部署应用程序

本主题描述如何使用 Apache™ Ant 任务将应用程序自动部署到 WebSphere Process Server。通过使用 Apache Ant 任务，可定义多个应用程序的部署并让它们在服务器上以无人照管的方式运行。

第 10 章 WebSphere Process Server 上的安全性

WebSphere Process Server 提供了基于 IBM WebSphere Application Server 安全性的运行时安全性基础结构和机制。

有关安全性的更多信息，请参阅保护应用程序及其环境。

第 11 章 在 WebSphere Process Server 上进行监视

监视 WebSphere Process Server 中的事件，以评估问题确定、调整性能和衡量业务流程的有效性。

WebSphere Process Server 事件监视功能包括性能监视和服务组件监视。

监视性能：性能测量可用于服务组件事件点，并可通过性能监视基础结构（PMI）和 Tivoli® Performance Viewer 进行处理。

您可以监视给定事件的特定性能测量，例如调用事件的次数或该事件从开始到结束所花费的时间量。此外，还可以监视事件，并且稍后可通过查看日志文件中的事件或查询存储在事件数据库中的事件来查看其内容。在这两种情况下，您可以临时指定要监视的一个或多个事件点，以便发现与应用程序逻辑或系统性能有关的问题。

监视服务组件事件：WebSphere Process Server 监视在某个事件点可以捕获服务组件中的数据。这些事件采用名为“公共基本事件”的标准进行格式化。您可以让流程服务器向记录工具发布这些事件，也可以使用公共事件基础结构服务器数据库的更为多用途的监视功能来存储并分析这些事件。

在流程服务器上运行的某些应用程序包含部署应用程序之后持续监视的事件点。如果您是业务分析员，并且要观察部署到流程服务器上的应用程序中建模和执行的业务流程的有效性，那么可以执行此操作。这使您可以使用 IBM WebSphere Business Monitor 等产品来创建定制面板或“仪表板”以查看关键业务流程度量。

相关概念



监视服务组件事件的性能和业务流程是 IBM WebSphere Process Server 的一项重要功能。

第 12 章 样本和教程

为了帮助您了解如何使用 WebSphere Process Server 来完成目标，提供了包含教程和样本的培训资料。

IBM Education Assistant 和样本库提供了 WebSphere Process Server 教程和样本，而 WebSphere Process Server 信息中心提供了有关管理任务的教程。

可直接在 <http://publib.boulder.ibm.com/bpcsamp/index.html> 中获得业务流程编排器样本。

教程

IBM Education Assistant 和 WebSphere Process Server 文档中提供了有关公共任务的教程。

IBM Education Assistant 教程

IBM Education Assistant 站点提供了便于您使用的教程。要查看此培训内容，请参阅 IBM Education Assistant (<http://publib.boulder.ibm.com/infocenter/ieduasst/v1r1m0>)。

WebSphere Process Server 教程

IBM WebSphere Process Server 文档中包含可帮助您进行某些管理任务、安全性任务和监视任务的教程主题。

教程：关系管理器管理。

教程：编写列示安全角色的 Jacl 脚本。

教程：使用 CEI 服务器进行事件监视。

教程：记录服务组件事件。

教程：服务组件性能监视。

访问样本（样本库）

样本库中提供了集成应用程序工件的样本，当您安装本产品时可选择安装样本库。

关于此任务

样本库包含简单工件的样本，例如，由 IBM WebSphere Integration Developer 生成并部署在 IBM WebSphere Process Server 上的工件。可直接在 <http://publib.boulder.ibm.com/bpcsamp/index.html> 中获得业务流程编排器样本。

要安装和查看 WebSphere Process Server 样本库，请执行下列步骤。

过程

1. 安装 WebSphere Process Server 并在“选择功能部件”面板上选择样本包，并创建概要文件作为产品安装的一部分。

注：如果要在 WebSphere Application Server 基础上安装 WebSphere Process Server，那么必须安装基本的 WebSphere Application Server 样本库，以便使用这些样本。样本将安装在 *install_root/samples* 目录中。

2. 启动服务器。
3. 通过在“第一步”控制台上选择**样本库**来启动样本库。 WebSphere Process Server 样本最初作为可安装的样本列示在样本库中。可以展开**可安装的样本**，并在**业务集成**下查找您要部署并运行的样本。

在 WebSphere Process Server 上运行的应用程序具有诸如业务对象、关系定义和业务规则之类的 XML 工件。在安装应用程序之前，必须部署上述工件。 WebSphere Process Server 提供名为 *serviceDeploy* 的实用程序来构建和部署这些工件。每个样本应用程序的 *install_root/samples/lib* 中的企业归档 (EAR) 文件都包含这些工件。 *sampleDeploy* 实用程序使用样本所需的特定参数来调用 *serviceDeploy*。运行 *sampleDeploy* 将在原始 EAR 文件所在的目录中创建名为 *sample_nameDeployed.ear* 的第二个 EAR 文件。这个新的 EAR 文件包含原始 EAR 文件中的 Web 归档 (WAR) 文件，以及其他包含已部署工件的 Java 归档 (JAR) 和 WAR 文件。已部署的 EAR 文件可作为企业应用程序安装在 WebSphere Process Server 中。

4. 如果 WebSphere Process Server 可安装样本未自动安装在样本库上，那么请手动安装并部署这些样本。
 - 要在具有集群功能的分布式 WebSphere Process Server 部署环境中安装和部署样本，请完成下列步骤。

- a. 在管理控制台中，展开**应用程序**，然后单击**安装新的应用程序**。
- b. 单击“浏览”按钮并在以下目录中找到 *SamplesGallery.ear* 文件：

– **Linux** **UNIX** **i5/OS** 在 **UNIX、Linux 和 i5/OS** 平台上：
install_root/samples/lib/SamplesGallery

– **Windows** 在 **Windows** 平台上：*install_root\samples\lib\SamplesGallery*

- c. 安装该 EAR 文件，同时接受除目标映射面板的缺省值之外的所有缺省值，您可在该面板上指定要安装样本库的服务器或集群。
- d. 对 *SamplesGallery* 目录中的 *WBISamplesGallery.ear* 文件重复先前的步骤。
- e. 启动刚安装的应用程序。
- f. 打开浏览器以访问位于以下网址的样本库：http://host_name:host_port /WSsamples/index.jsp。
- g. 按照样本库中的指示信息来部署和运行每个样本，但是将在管理控制台上使用**安装新的应用程序**（而不是使用不支持集群的 *installwbi* 命令）。可以在每个样本的以下目录中找到已部署的 EAR 文件：

– **Linux** **UNIX** **i5/OS** 在 **UNIX、Linux 和 i5/OS** 平台上：
install_root/samples/lib/sample_name

– **Windows** 在 **Windows** 平台上：*install_root\samples\lib\sample_name*

- 要在不带集群功能的分布式 WebSphere Process Server 部署环境中安装和部署样本，请执行下列步骤。
 - a. 在具有 *Deployment Manager* 节点的机器上，运行以下命令：

- **Linux** **UNIX** **i5/OS** 在 **UNIX**、**Linux** 和 **i5/OS** 平台上:
`install_root/samples/bin/installwbi -node node_name -server server_name -samples SamplesGallery WBISamplesGallery`
- **Windows** 在 **Windows** 平台上: `install_root\samples\bin\installwbi -node node_name -server server_name -samples SamplesGallery WBISamplesGallery`

注: 如果在 WebSphere Process Server 概要文件上启用了管理安全性, 那么还必须输入 `-samplepw` 参数并输入您在创建概要文件时所创建的密码。

- b. 在管理控制台中, 展开 **应用程序**, 接着单击 **企业应用程序**, 然后启动 **SamplesGallery** 和 **WBISamplesGallery**。
 - c. 打开浏览器以访问位于以下网址的样本库: `http://host_name:host_port /WSsamples/index.jsp`。
 - d. 按照样本库中的指示信息部署和运行每个样本, 并确保将 **-node node_name** **-server server_name** 参数与 `installwbi` 命令配合使用。
5. 启动样本库并单击**刷新**。可以按照浏览器窗口中显示的指示信息来部署任何可安装的样本。先前已部署的样本被列示为已安装的样本, 可以通过在浏览器窗口中选择一个选项来运行这些样本。
 6. 运行已部署的每个样本。

相关概念



“第一步”控制台上的选项

安装 WebSphere Process Server 后, 使用“第一步”控制台启动产品工具、访问产品文档或管理诸如与各个概要文件相关的服务器和管理控制台等元素。提供了控制台的一个通用版本以及一个适合于安装中各个概要文件的版本。每个控制台上的选项是动态显示的, 取决于您在特定操作系统上安装的功能部件及特定元素的可用性。选项包括验证安装、启动或停止服务器或 **Deployment Manager**、访问管理控制台、启动概要管理工具、访问样本库、访问产品文档或者启动迁移向导。

第 13 章 标准一致性

WebSphere Process Server 符合多项政府标准和业界标准，其中包括：辅助功能选项标准、信息处理标准、软件下载安全性标准和因特网协议标准。

辅助功能选项

IBM 致力于让每个人都可以使用这些产品，而不管他们的年龄大小或能力高低。

此产品使用标准的 Windows 导航键。

WebSphere Process Server 的辅助功能部件

辅助功能部件可以帮助有生理缺陷（例如，行动不便或视力不好）的用户成功使用信息技术产品。

辅助功能部件

以下列表包含 WebSphere Process Server 中的主要辅助功能部件。这些辅助功能部件包含下列功能：

- 仅键盘操作
- 屏幕阅读器常用的界面

当您使用 WebSphere Process Server 时，支持辅助功能选项的操作系统功能部件可用。

提示： WebSphere Process Server 信息中心已启用辅助功能选项，适用于包括 IBM Home Page Reader 等在内的屏幕阅读器。您可以使用键盘而非鼠标来操作所有功能部件。

键盘导航

此产品使用标准 Web 浏览器导航键和标准 Installshield Multiplatform 导航键。

（有关受支持 Web 浏览器的信息，请参阅 WebSphere Process Server 系统需求，网址为：<http://www.ibm.com/software/integration/wps/sysreqs/>。）

界面信息

- 安装

可以采用图形方式或静默方式来安装 WebSphere Process Server。建议需要使用辅助功能选项的用户执行静默安装程序。

有关指示信息，请参阅以静默方式安装产品。

注： WebSphere Process Server 安装程序不支持 Installshield Multiplatform 控制台方式。

- 管理

管理控制台是与产品交互的主要接口。此控制台显示在标准 Web 浏览器中。通过使用诸如 Microsoft Internet Explorer 之类的可访问 Web 浏览器，管理员可以执行下列操作：

- 使用屏幕阅读器软件和数字语音合成器来听取屏幕上显示的内容
- 使用诸如 IBM ViaVoice® 之类的语音识别软件来输入数据和浏览用户界面
- 使用键盘（而非鼠标）来操作功能部件

通过使用所提供的标准文本编辑器和脚本编制或命令行界面（而非图形界面），您可以配置和管理产品功能部件。

在适用的情况下，特定产品功能部件的文档包含有关功能部件的辅助功能选项的其他信息。

供应商软件

本产品包括 IBM 许可协议未涵盖的某些第三方软件。IBM 对于这些产品是否满足美国联邦政府康复法案的条款 508 不作任何说明。请与供应商联系，以了解有关它们提供的产品是否满足条款 508 的信息。可以在 IBM 产品辅助功能选项信息 Web 页面（www.ibm.com/able/product_accessibility）上请求满足美国联邦政府康复法案条款 508 的“自发的产品辅助功能模板”（VPAT）。

相关的辅助功能选项信息

请参阅 IBM 辅助功能选项中心，以了解有关 IBM 对辅助功能选项所提供的更多信息。

美国联邦信息处理标准

“美国联邦信息处理标准”（FIPS）是“美国国家标准技术学会”（NIST）针对联邦政府的计算机系统发布的标准和准则。

WebSphere Process Server 的所有密码功能都依赖于 IBM WebSphere Application Server 来实现，后者符合“美国联邦信息处理标准”。

在联邦政府对标准（例如，安全性和互操作性）具有强制性要求，但又不存在可接受的业界标准或解决方案的情况下制定了 FIPS。政府机构和金融机构都按照这些标准来确保产品符合指定的安全性要求。有关这些标准的更多信息，请访问“美国国家标准技术学会”的网站 <http://www.nist.gov/>。

WebSphere Application Server 集成了密码模块，包括“Java 安全套接字扩展”（JSSE）和“Java 密码术扩展”（JCE），它们都已经通过了 FIPS 140-2 认证。在 WebSphere Application Server 文档中，将已经通过了 FIPS 认证的 IBM JSSE 和 JCE 模块称为 IBMJSSEFIPS 和 IBMJCEFIPS。

有关更多信息，请参阅 WebSphere Application Server 信息中心中的配置联合信息处理标准 Java 安全套接字扩展文件。当启用 FIPS 时，该服务器的以下几个组件会受到影响：密码套件、密码提供者、负载均衡器、高速缓存代理、高可用性管理器和数据复制服务。

Common Criteria

美国国家标准技术学会（NIST）制定了 Common Criteria，以确保您将软件下载到自己的系统上使用时选择安全选项。

WebSphere Process Server 通过 WebSphere Application Server Network Deployment 来提供安全功能，它并不包括额外的安全功能。有关 Common Criteria 在 WebSphere Application Server Network Deployment 中的一致性的更多信息，请参阅 WebSphere Application Server 信息中心中的 Common Criteria (EAL4) 支持。

因特网协议版本 6

WebSphere Process Server 依赖 WebSphere Application Server 来实现所有因特网协议版本 6.0 兼容性。

IBM WebSphere Application Server V6.1 及其 JavaMail 组件支持因特网协议版本 6 (IPv6)。

有关 WebSphere Application Server 中此兼容性的更多信息，请参阅 WebSphere Application Server Network Deployment 文档中的 IPv6 支持。

有关 IPv6 的更多信息，请参阅 www.ipv6.org。

第 14 章 全球化

全球化产品的使用没有语言或文化障碍，并且可以为特定语言环境而启用。

WebSphere Process Server 为所有语言环境提供基本启用支持。将提供下列本地语言的翻译：

- 巴西葡萄牙语
- 捷克语
- 法语
- 德语
- 匈牙利语
- 意大利语
- 日语
- 韩国语
- 波兰语
- 俄罗斯语
- 简体中文（符合 GB18030）
- 西班牙语
- 繁体中文

另外，WebSphere Process Server 提供以下本地语言的部分翻译：

- 阿拉伯语
- 希伯来语

有关通过 WebSphere Application Server 提供的全球化应用程序和国际化服务的信息，请参阅 WebSphere Application Server Network Deployment 文档中的 WebSphere 扩展。

双向语言支持

WebSphere Process Server 通过双向启用来支持双向语言。双向启用是一种机制，可精确地显示和处理随 WebSphere Process Server 捆绑的组件（诸如公共基本事件浏览器或业务规则管理等基于 Web 的工具）或该产品所支持的组件（例如，服务组件）中的双向字母表数据。

WebSphere Process Server 将所有双向语言数据都按照从左到右的逻辑格式（也就是 Windows 标准双向语言格式）来处理。它采用这种格式来处理传递给内部组件的数据、存储数据和输出数据。WebSphere Adapters 和其他企业信息系统（EIS）必须将数据转换为该格式之后，才能将这些数据发送给 WebSphere Process Server 处理。由于 WebSphere Process Server 输出的数据也是采用从左到右的逻辑格式，因此，用于接收数据的应用程序必须将它转换为外部 EIS 需要的正确双向格式。

下表显示了必须与 Windows 标准双向格式匹配的属性和设置。

表 1. 双向语言格式字符串值

字母位置	用途	允许的值	缺省值	含义
1	排序模式	I	I	隐式
		V		可视
2	方向	L	L	从左到右
		R		从右到左
		C		上下文从左到右
		D		上下文从右到左
3	对称交换	Y	Y	打开对称交换
		N		关闭对称交换
4	塑形	S	N	对文本进行塑形
		N		不对文本进行塑形
		I		初始塑形
		M		中间塑形
		F		最终塑形
		B		隔离塑形
5	数字	H	N	印地语（本地语言）
		C		上下文
		N		名词性

对于来自未实施双向支持的外部组件（例如，不支持处理双向数据处理的 Web Service 或连接器）的数据，您可以使用基于 IBM Java Development Kit (JDK) 的示例双向 API 来创建 API，以便将来自外部源的数据变换为受支持的双向语言格式以及将从 WebSphere Process Server 发送给外部 EIS 的数据变换为该特定 EIS 所使用的双向格式。

要创建 API 以变换字符串对象，请参阅将字符串对象从一种双向语言格式变换为另一种双向语言格式。

要创建 API 以变换数据对象，请参阅将数据对象从一种双向语言格式变换为另一种双向语言格式。

注： 用户界面（浏览器）的语言环境设置用于定义双向语言显示和编辑格式。

有关双向语言的更多信息，请参阅 IBM developerWorks 上的技术文章，网址为：
www.ibm.com/developerworks/websphere/library/techarticles/bidi/bidigen.html。

将字符串对象从一种双向语言格式变换为另一种双向语言格式

对于来自外部企业信息系统（EIS）的数据，您可以创建 API 以将字符串数据变换为受支持的双向语言格式以及将从 WebSphere Process Server 发送给外部 EIS 的数据变换为该特定 EIS 所使用的双向格式。

开始之前

有关双向语言支持的更多信息，请参阅全球化。使用“全球化”中的表来确定将字符串数据从一种格式变换为另一种格式时，要使用的输入字符串或输出字符串的正确值。

要创建 API 以变换字符串对象的双向语言格式，请执行下列步骤：

过程

1. 包括所有包含双向引擎实现的双向类。 例如：

```
import com.ibm.bidiTools.bdlayout.*;
```

2. 定义字符串以便包含要变换的数据对象以及输入和输出格式值。

输入格式是当前存储字符串对象时所采用的双向格式。输出格式是要用于存储字符串对象的双向格式。 例如：

```
String strIn = new String("Hello world");  
String formatIn = "ILYNN";  
String formatOut = "VLYNN";
```

3. 调用 `BidiStringTransformation` 函数。 例如：

```
String strOut = BiDiStringTransformation(strIn, formatIn, formatOut);  
String BiDiStringTransformation(String strIn, String formatIn, String formatOut) {
```

- a. 测试输入字符串是否为 `null`。 例如：

```
    if (strIn == null) return null;
```

- b. 执行变换。 例如：

```
        BidiFlagSet flagsIn;  
        BidiFlagSet flagsOut;  
        formatIn = formatIn.toUpperCase();  
        formatOut = formatOut.toUpperCase();  
  
        if (formatIn != null)  
            flagsIn = new BidiFlagSet(formatIn.toCharArray());  
        else  
            flagsIn = new BidiFlagSet();  
  
        if (formatOut != null)  
            flagsOut = new BidiFlagSet(formatOut.toCharArray());  
        else  
            flagsOut = new BidiFlagSet();  
  
        if (flagsIn.equals(flagsOut)) return strIn;  
        String strOut = BiDiStringTransformation(strIn, flagsIn, flagsOut);  
        return strOut;  
    }
```

将数据对象从一种双向语言格式变换为另一种双向语言格式

对于来自外部企业信息系统（EIS）的数据，您可以创建 API 以将服务数据对象变换为受支持的双向语言格式以及将从 WebSphere Process Server 发送给外部 EIS 的数据变换为该特定 EIS 所使用的双向格式。

开始之前

有关双向语言支持的更多信息，请参阅全球化。使用“全球化”中的表来确定将数据对象类型的数据从一种格式变换为另一种格式时，要使用的输入字符串或输出字符串的正确值。

要创建 API 以变换数据对象的双向语言格式，请执行下列步骤：

过程

1. 包括所有包含双向引擎实现的双向类。 例如:

```
import com.ibm.bidiTools.bdlayout.*;
```

2. 包括处理数据对象类型对象所需的全部类。 例如:

```
import commonj.sdo.DataObject;  
import commonj.sdo.Type;  
import commonj.sdo.Property;
```

3. 定义字符串变量以包含数据对象类型对象所拥有的不同类型的字符串。在以递归方式遍历数据对象时, 这将过滤 type String 的属性。 例如:

```
String STRING_STR_TYPE = "String";  
String NORM_STRING_STR_TYPE = "normalizedString";  
String TOKEN_STR_TYPE = "token";  
String LANG_STR_TYPE = "language";  
String NAME_STR_TYPE = "Name";  
String NM_TOKEN_STR_TYPE = "NM_TOKEN";  
String Ncname_STR_TYPE = "NCName";  
String ID_STR_TYPE = "ID";  
String IDREF_STR_TYPE = "IDREF";  
String IDREFS_STR_TYPE = "IDREFS";  
String ENTITY_STR_TYPE = "ENTITY";  
String ENTITIES_STR_TYPE = "ENTITIES";
```

4. 定义用于验证属性类型是否为字符串的函数。 例如:

```
private static boolean isStringFamilyType (Property property) {  
    boolean rc = false;  
    if ((property.getType().getName().equalsIgnoreCase(STRING_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(NORM_STRING_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(TOKEN_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(LANG_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(NAME_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(NM_TOKEN_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(NCNAME_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(ID_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(IDREF_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(IDREFS_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(ENTITY_STR_TYPE)) ||  
        (property.getType().getName().equalsIgnoreCase(ENTITIES_STR_TYPE)))  
        rc = true;  
    return rc;  
}
```

5. 定义对整个数据对象应用双向变换的递归函数。

注: 代码逻辑作了如下假定:

- 仅对字符串类型的属性应用了双向变换。
- 数据对象中的字符串类型的属性以一种双向格式进行存储。

例如:

```
DataObject BiDiDataObjTransformationB0(DataObject boIn, String formatIn, String formatOut){  
  
    Type type;  
    Property property;  
  
    if (boIn == null) return null;  
  
    type = boIn.getType();  
    List propertyList = type.getProperties();  
    for (int propertyNumber = 0; propertyNumber < propertyList.size(); propertyNumber++){  
        property = (Property) propertyList.get(propertyNumber);  
        String propertyName = property.getName();
```

- a. 跳过所有非字符串属性。 例如:

```
if (!isStringFamilyType(property))
    continue;
```

```
if (property.isContainment()) {
    if (property.isMany()) {
        List childsList = boIn.getList(property);
```

b. 以递归方式调用变换以处理子对象。 例如:

```
        for (int childNumber = 0; childNumber < childsList.size();
            childNumber++){
            BiDiDataObjTransformationBO(connectionContext,
            ((DataObject)childsList.get(childNumber)),formatIn, formatOut);
        }
    } else {
```

c. 以递归方式调用变换以处理任何已包含的业务对象的子对象。 例如:

```
        BiDiDataObjTransformationBO(connectionContext,
        ((DataObject)boIn.get(property)),formatIn, formatOut);
    }
} else {
```

d. 变换简单的字符串属性。 例如:

```
        String str = BiDiStringTransformation(
            (boIn.getString(propertyName),formatIn, formatOut);
        boIn.setString(propertyName, str);
    }
}
return boIn;
}
```

声明

本信息是为在美国提供的产品和服务编写的。

IBM 可能在其他国家或地区不提供本文中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 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.*

有关双字节（DBCS）信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

*IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan*

本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区： 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 Corporation
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。

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。

本信息包含在日常业务操作中使用的数据和报告的示例。为了尽可能全面地说明这些数据和报告，这些示例中包括个人、公司、商标和产品的名称。所有这些名称都是虚构的，与实际商业企业所用的名称和地址的任何雷同纯属巧合。

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。

凡这些样本程序的每份拷贝或其任何部分或任何衍生产品，都必须包括如下版权声明：©（贵公司的名称）（年）。此部分代码根据 IBM 公司的样本程序衍生。© Copyright IBM Corp.（输入年份）。All rights reserved.

如果您正以软拷贝格式查看本信息，图片和彩色图例可能无法显示。

编程接口信息

编程接口信息（如果提供了的话）用来帮助您使用此程序来创建应用软件。

通用编程接口允许您编写获取此程序工具的的服务的应用软件。

然而，本信息还可能包含诊断、修改和调整信息。诊断、修改和调整信息用来帮助您调试应用软件。

警告： 不要将此诊断、修改和调整信息用作编程接口，因为它是会更改的。

商标和服务标记

IBM、IBM 徽标、CICS、DB2、developerWorks、i5/OS、Lotus、Rational、Tivoli、ViaVoice、WebSphere、z/OS 和 zSeries 是 International Business Machines Corporation 在美国和/或其他国家或地区的注册商标，System i 是 International Business Machines Corporation 在美国和/或其他国家或地区的商标。

Adobe 是 Adobe Systems Incorporated 在美国和其他国家或地区的注册商标。

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。

Microsoft 和 Windows 是 Microsoft Corporation 在美国和/或其他国家或地区的注册商标。

UNIX 是 The Open Group 在美国和其他国家或地区的注册商标。

Linux 是 Linus Torvalds 在美国和/或其他国家或地区的注册商标。

其他公司、产品或服务名称可能是其他公司的商标或服务标记。

本产品包括由 Eclipse Project (<http://www.eclipse.org>) 开发的软件。



IBM WebSphere Process Server for Multiplatforms V6.1.0



中国印刷