

目录

PowerCenter WebServices 提供程序指南-版权.....	3
前言.....	6
Informatica 资源.....	6
Web 服务概念.....	7
Web 服务概念概览.....	7
简单对象访问协议 (SOAP).....	8
Web 服务说明语言 (WSDL).....	8
了解 Web Services Provider.....	9
了解 Web Services Provider 概览.....	9
Web Services Provider 体系结构.....	10
性能和可扩展性.....	11
Web 服务中心安全.....	12
Web 服务中心日志.....	12
SOAP 故障处理.....	13
使用 Web 服务中心控制台.....	15
使用 Web 服务中心控制台概览.....	15
连接到 Web 服务中心控制台.....	15
了解 Web 服务中心控制台.....	15
测试 Web 服务.....	18
批处理 Web 服务操作.....	21
批处理 Web 服务操作概览.....	21
元数据 Web 服务操作.....	21
数据集成 Web 服务操作.....	23
编写客户端应用程序.....	46
编写客户端应用程序概览.....	46
批处理 Web 服务的客户端应用程序.....	47
批处理 Web 服务的 Java 客户端应用程序.....	48
批处理 Web 服务的 C# 客户端应用程序.....	51
实时 Web 服务的客户端应用程序.....	54
实时 Web 服务的 Java 客户端应用程序.....	55

使用参数数组.....	57
为客户端请求添加安全性.....	59
使用 Web 服务源和目标.....	63
Web 服务源和目标概览.....	63
了解 Web 服务源和目标.....	63
导入 Web 服务源或目标定义.....	65
创建一个源或目标定义.....	68
编辑 Web 服务源和目标.....	70
编辑 Web 服务源和目标概览.....	70
在 Designer 工作区中编辑定义.....	71
在 WSDL 工作区中编辑定义.....	72
使用 Web 服务映射.....	73
使用 Web 服务映射概览.....	73
Web 服务映射类型.....	74
通过 WSDL 生成映射.....	75
通过关系或平面文件源或目标生成映射.....	76
通过转换或 Mapplet 生成映射.....	76
编辑 Web 服务映射中的目标实例.....	78
附件.....	78
使用 Web 服务工作流.....	79
使用 Web 服务工作流概览.....	79
创建和配置 Web 服务工作流.....	80
配置 Web Services Provider 读取器和写入器.....	82
为 Web 服务会话配置分区.....	85
Web 服务工作流故障排除.....	85
Web 服务示例客户端应用程序.....	86
Web 服务示例客户端应用程序概览.....	86
使用批处理 Web 服务示例程序.....	86
批处理 Web 服务示例.....	88
使用实时 Web 服务示例程序.....	93
实时 Web 服务示例.....	96
配置 Web 浏览器.....	97
配置 Web 浏览器.....	97

PowerCenter WebServices 提供程序指南-版权

本软件和文档仅根据包含使用与披露限制的单独许可协议提供。未事先征得 Informatica LLC 同意，不得以任何形式、通过任何手段（电子、影印、录制或其他手段）复制或传播本文档的任何部分。

Informatica、Informatica 标志和 PowerCenter 是 Informatica LLC 在美国和世界其他许多司法管辖区的商标或注册商标。欲获得 Informatica 商标的最新列表，请访问 <https://www.informatica.com/trademarks.html>。其他公司和产品名称可能是其各自所有者的商业名称或商标。

本软件和/或文档的某些部分受第三方版权制约，包括但不限于：版权所有 DataDirect Technologies。保留所有权利。版权所有 (C) Sun Microsystems。保留所有权利。版权所有 (C) RSA Security Inc. 保留所有权利。版权所有 (C) Ordinal Technology Corp. 保留所有权利。版权所有 (C) Aandacht c.v. 保留所有权利。版权所有 Genivia, Inc. 保留所有权利。版权所有 Isomorphic Software。保留所有权利。版权所有 (C) Meta Integration Technology, Inc. 保留所有权利。版权所有 (C) Intalio。保留所有权利。版权所有 (C) Oracle。保留所有权利。版权所有 (C) Adobe Systems Incorporated。保留所有权利。版权所有 (C) DataArt, Inc. 保留所有权利。版权所有 (C) ComponentSource。保留所有权利。版权所有 (C) Microsoft Corporation。保留所有权利。版权所有 (C) Rogue Wave Software, Inc. 保留所有权利。版权所有 (C) Teradata Corporation。保留所有权利。版权所有 (C) Yahoo! Inc. 保留所有权利。版权所有 (C) Glyph & Cog, LLC。保留所有权利。版权所有 (C) Thinkmap, Inc. 保留所有权利。版权所有 (C) Clearpace Software Limited。保留所有权利。版权所有 (C) Information Builders, Inc. 保留所有权利。版权所有 (C) OSS Nokalva, Inc. 保留所有权利。版权所有 Edifecs, Inc. 保留所有权利。版权所有 Cleo Communications, Inc. 保留所有权利。版权所有 (C) International Organization for Standardization 1986。保留所有权利。版权所有 (C) ej-technologies GmbH。保留所有权利。版权所有 (C) Jaspersoft Corporation。保留所有权利。版权所有 (C) International Business Machines Corporation。保留所有权利。版权所有 (C) yWorks GmbH。保留所有权利。版权所有 (C) Lucent Technologies。保留所有权利。版权所有 (C) University of Toronto。保留所有权利。版权所有 (C) Daniel Veillard。保留所有权利。版权所有 (C) Unicode, Inc. 版权所有 IBM Corp. 保留所有权利。版权所有 (C) MicroQuill Software Publishing, Inc. 保留所有权利。版权所有 (C) PassMark Software Pty Ltd. 保留所有权利。版权所有 (C) LogiXML, Inc. 保留所有权利。版权所有 (C) 2003-2010 Lorenzi Davide, 保留所有权利。版权所有 (C) Red Hat, Inc. 保留所有权利。版权所有 (C) The Board of Trustees of the Leland Stanford Junior University。保留所有权利。版权所有 (C) EMC Corporation。保留所有权利。版权所有 (C) Flexera Software。保留所有权利。版权所有 (C) Jinfonet Software。保留所有权利。版权所有 (C) Apple Inc. 保留所有权利。版权所有 (C) Telerik Inc. 保留所有权利。版权所有 (C) BEA Systems。保留所有权利。版权所有 (C) PDFlib GmbH。保留所有权利。版权所有 (C) Orientation in Objects GmbH。保留所有权利。版权所有 (C) Tanuki Software, Ltd. 保留所有权利。版权所有 (C) Ricebridge。保留所有权利。版权所有 (C) Sencha, Inc. 保留所有权利。版权所有 (C) Scalable Systems, Inc. 保留所有权利。版权所有 (C) jQWidgets。保留所有权利。版权所有 (C) Tableau Software, Inc. 保留所有权利。版权所有 (C) MaxMind, Inc. 保留所有权利。版权所有 (C) TMate Software s.r.o. 保留所有权利。版权所有 (C) MapR Technologies Inc. 保留所有权利。版权所有 (C) Amazon Corporate LLC。保留所有权利。版权所有 (C) Highsoft。保留所有权利。版权所有 (C) Python Software Foundation。保留所有权利。版权所有 (C) BeOpen.com。保留所有权利。版权所有 (C) CNRI。保留所有权利。

本产品包括由 Apache Software Foundation (<http://www.apache.org/>) 开发的软件和/或在不同 Apache 许可证版本（以下简称“许可证”）下许可的其他软件。您可从 <http://www.apache.org/licenses/> 获取这些许可证的副本。除非适用法律要求或者有相应书面协议，否则依据这些“许可证”分发的软件以“原样”提供，不附带任何明示或暗示的担保或条件。请参阅“许可证”中规定的具体语言管理权限和限制。

本产品包括由 Mozilla (<http://www.mozilla.org/>) 开发的软件、由 JBoss Group, LLC 开发的软件（版权所有 JBoss Group, LLC 保留所有权利）、由 Bruno Lowagie 和 Paulo Soares 开发的软件（版权所有 (C) 1999-2006 Bruno Lowagie 和 Paulo Soares）以及在 <http://www.gnu.org/licenses/lgpl.html> 网站上的

不同版本 GNU Lesser General 公共许可协议下许可的软件。这些材料由 Informatica 按“原样”免费提供，不附带任何明示或暗示的担保，包括但不限于适销性和特定用途适用性的暗示担保。

本产品包括 ACE(TM) 和 TAO(TM) 软件，这些软件版权归 Douglas C. Schmidt 及其在华盛顿大学、加利福尼亚大学欧芬分校以及范德堡大学的研发团队所有（版权所有 ((C)) 1993-2006，保留所有权利）。

本产品包括由 OpenSSL Project 开发并在 OpenSSL Toolkit（版权所有 OpenSSL Project。保留所有权利）中使用的软件，该软件的再分发受 <http://www.openssl.org> 和 <http://www.openssl.org/source/license.html> 上规定条款之制约。

本产品包括 Curl 软件，版权所有 1996-2013, Daniel Stenberg <daniel@haxx.se>。保留所有权利。有关该软件的权限和限制受 <http://curl.haxx.se/docs/copyright.html> 上规定条款之制约。允许出于任何目的以免费或收费形式使用、复制、修改和分发该软件，但前提是所有副本均应注明上述版权声明以及本许可声明。

本产品包括由 MetaStuff, Ltd. 开发的软件，版权所有 2001-2005 ((C)) MetaStuff, Ltd. 保留所有权利。有关该软件的权限和限制受 <http://www.dom4j.org/license.html> 上规定条款之制约。

本产品包括由 Per Bothner 开发的软件，版权所有 (C) 1996-2006 Per Bothner。保留所有权利。<http://www.gnu.org/software/kawa/Software-License.html> 上的许可证中规定了您使用这些材料的权利。

本产品包括 OSSP UUID 软件，版权所有 (C) 2002 Ralf S. Engelschall，版权所有 (C) 2002 OSSP Project，版权所有 (C) 2002 Cable & Wireless Deutschland。有关该软件的权限和限制受 <http://www.opensource.org/licenses/mit-license.php> 上规定条款之制约。

本产品包括由 Boost (<http://www.boost.org/>) 开发的软件或在 Boost 软件许可证下许可的软件。有关该软件的权限和限制受 http://www.boost.org/LICENSE_1_0.txt 上规定条款之制约。

本产品包括由 University of Cambridge 开发的软件，版权所有 (C) 1997-2007 University of Cambridge。有关该软件的权限和限制受 <http://www.pcre.org/license.txt> 上规定条款之制约。

本产品包括由 The Eclipse Foundation 开发的软件，版权所有 (C) 2007 The Eclipse Foundation。保留所有权利。有关该软件的权限和限制受 <http://www.eclipse.org/org/documents/epl-v10.php> 和 <http://www.eclipse.org/org/documents/edl-v10.php> 上规定条款之制约。

本产品包括在 <http://www.tcl.tk/software/tcltk/license.html>、<http://www.bosrup.com/web/overlib/?License>、<http://www.stlport.org/doc/license.html>、<http://asm.ow2.org/license.html>、<http://www.cryptix.org/LICENSE.TXT>、<http://hsqldb.org/web/hsqLicense.html>、<http://httpunit.sourceforge.net/doc/license.html>、<http://jung.sourceforge.net/license.txt>、http://www.gzip.org/zlib/zlib_license.html、<http://www.openldap.org/software/release/license.html>、<http://www.libssh2.org>、<http://slf4j.org/license.html>、<http://www.sente.ch/software/OpenSourceLicense.html>、<http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3-license-agreement>、<http://antlr.org/license.html>、<http://aopalliance.sourceforge.net/>、<http://www.bouncycastle.org/licence.html>、<http://www.jgraph.com/jgraphdownload.html>、<http://www.jcraft.com/jsch/LICENSE.txt>、http://jotm.objectweb.org/bsd_license.html、<http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>、<http://www.slf4j.org/license.html>、<http://nanoxml.sourceforge.net/orig/copyright.html>、<http://www.json.org/license.html>、<http://forge.ow2.org/projects/javaservice/>、<http://www.postgresql.org/about/licence.html>、<http://www.sqlite.org/copyright.html>、<http://www.tcl.tk/software/tcltk/license.html>、<http://www.jaxen.org/faq.html>、<http://www.jdom.org/docs/faq.html>、<http://www.slf4j.org/license.html>、<http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/License>、<http://www.keplerproject.org/md5/license.html>、<http://www.toedter.com/en/>

jcalendar/license.html、<http://www.edankert.com/bounce/index.html>、<http://www.net-snmp.org/about/license.html>、<http://www.openmdx.org/#FAQ>、http://www.php.net/license/3_01.txt、<http://srp.stanford.edu/license.txt>、<http://www.schneier.com/blowfish.html>、<http://www.jmock.org/license.html>、<http://xsom.java.net>、<http://benalman.com/about/license/>、<https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js>、<http://www.h2database.com/html/license.html#summary>、<http://jsoncpp.sourceforge.net/LICENSE>、<http://jdbc.postgresql.org/license.html>、<http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto>、<https://github.com/rantav/hector/blob/master/LICENSE>；<http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html>、<http://jibx.sourceforge.net/jibx-license.html>、<https://github.com/lyokato/libgeohash/blob/master/LICENSE>、<https://github.com/hjiang/jsonxx/blob/master/LICENSE>、<https://code.google.com/p/lz4/>、<https://github.com/jedisct1/libsodium/blob/master/LICENSE>、<http://one-jar.sourceforge.net/index.php?page=documents&file=license>、<https://github.com/EsotericSoftware/kryo/blob/master/license.txt>、<http://www.scala-lang.org/license.html>、<https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt>、<http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>、<https://aws.amazon.com/asl/>、<https://github.com/twbs/bootstrap/blob/master/LICENSE> 和 <https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt> 下许可的软件。

本产品包括在 Academic 免费许可证 (<http://www.opensource.org/licenses/afl-3.0.php>)、通用开发和分发许可证 (<http://www.opensource.org/licenses/cddl1.php>)、通用公共许可证 (<http://www.opensource.org/licenses/cpl1.0.php>)、Sun Binary Code 许可协议补充许可条款、BSD 许可证 (<http://www.opensource.org/licenses/bsd-license.php>)、新 BSD 许可证 (<http://opensource.org/licenses/BSD-3-Clause>)、MIT 许可证 (<http://www.opensource.org/licenses/mit-license.php>)、Artistic 许可证 (<http://www.opensource.org/licenses/artistic-license-1.0>) 以及原始开发者公共许可证版本 1.0 (<http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/>) 下许可的软件。

本产品包括由 Joe Walnes 和 XStream Committers 开发的软件，版权所有 (C) 2003-2006 Joe Walnes，2006-2007 XStream Committers。保留所有权利。有关该软件的权限和限制受 <http://xstream.codehaus.org/license.html> 上规定条款之制约。本产品包括由 Indiana University Extreme! Lab 开发的软件。有关详细信息，请访问 <http://www.extreme.indiana.edu/>。

本产品包括软件版权所有 (c) 2013 Frank Balluffi 和 Markus Moeller。保留所有权利。有关此软件的权限和限制受 MIT 许可证上规定条款之制约。

请参阅位于以下位置的专利：<https://www.informatica.com/legal/patents.html>。

免责声明：Informatica LLC 以“原样”提供本文档，不附带任何明示或暗示的担保，包括但不限于非侵权、适销性或特定用途适用性的暗示担保。Informatica LLC 不保证本软件和文档中没有错误。本软件或文档中提供的信息可能包括技术上的不准确性或排字错误。本软件和文档中包含的信息随时可能更改，恕不另行通知。

声明

本 Informatica 产品（以下称“软件”）包括由 Progress Software Corporation 的运营公司 DataDirect Technologies（以下称“DataDirect”）提供的某些驱动程序（以下称“DataDirect 驱动程序”），受以下条款和条件制约：

1. DataDirect 驱动程序以“原样”提供，不附带任何明示或暗示的担保，包括但不限于适销性、特定用途适用性以及非侵权的暗示担保。
2. 在任何情况下，DataDirect 或其第三方供应商均不对最终用户客户承担因使用 ODBC 驱动程序而引起的任何直接、间接、偶发、特殊、继发或其他损害赔偿的责任，无论是否已提前告知该种损害的可能性。这些限制适用于所有诉因，包括但不限于违反合同、违反担保、过失、严格责任、虚假陈述以及其他侵权行为。

本文档中的信息如有更改，恕不另行通知。如发现本文档中有什么问题，请通过以下电子邮件地址向我们报告：infa_documentation@informatica.com。

Informatica 产品根据对应协议的条款和条件进行担保。INFORMATICA 按"原样"提供本文档中的信息，无任何明示或暗示的担保，包括但不限于任何适销性和特定用途适用性担保，也没有任何非侵权担保或条件。

前言

有关 Web Services Provider 和 Web 服务中心托管的 PowerCenter Web 服务的信息，请阅读《PowerCenter® Web Services Provider 指南》。阅读本指南可学习如何将 PowerCenter 工作流转变为 Web 服务，以及查看使用 Web 服务中心上的 Web 服务创建客户端应用程序的示例。

Informatica 资源

Informatica 通过 Informatica Network 和其他在线门户为您提供一系列产品资源。使用这些资源，可以充分利用 Informatica 产品和解决方案，并向其他 Informatica 用户和主题专家学习。

Informatica Network

在 Informatica Network 中可以获得许多资源，包括 Informatica 知识库和 Informatica 全球客户支持。要进入 Informatica Network，请访问 <https://network.informatica.com>。

作为 Informatica Network 成员，您可以选择以下服务：

- 在知识库中搜索产品资源。
- 查看产品可用性信息。
- 创建并检查您的支持案例。
- 查找当地的 Informatica 用户组网络并与您的伙伴进行协作。

Informatica 知识库

使用 Informatica 知识库可查找产品资源，例如操作方法文章、最佳实践、视频教程以及常见问题的答案。

要搜索知识库，请访问 <https://search.informatica.com>。如果您对知识库有任何疑问、意见或建议，请与 Informatica 知识库团队联系，电子邮件地址为 KB_Feedback@informatica.com。

Informatica 文档

使用 Informatica 文档门户可浏览大量当前与最近产品版本的文档库。要浏览文档门户，请访问 <https://docs.informatica.com>。

如果您对产品文档有任何疑问、意见或建议，请与 Informatica 文档团队联系，电子邮件地址为 infa_documentation@informatica.com。

Informatica 产品可用性矩阵

产品可用性矩阵 (PAM) 指明了产品版本支持的操作系统版本、数据库以及数据源和目标的类型。您可以在以下网址中浏览 Informatica PAM：

<https://network.informatica.com/community/informatica-network/product-availability-matrices>。

Informatica Velocity

Informatica Velocity 是由 Informatica 专业服务根据数百个数据管理项目的实际经验所开发出来的，其中汇集了大量使用技巧和最佳实践。Informatica Velocity 代表了 Informatica 顾问的集体知识，这些顾问与世界各地的组织合作，共同计划、开发、部署和维护成功的数据管理解决方案。

您可以在以下网址中找到 Informatica Velocity 资源：<http://velocity.informatica.com>。如果您对 Informatica Velocity 有任何疑问、意见或建议，请通过 ips@informatica.com 与 Informatica 专业服务联系。

Informatica Marketplace

Informatica Marketplace 是一个论坛，该论坛中提供的解决方案可扩展和增强您的 Informatica 实施。利用 Informatica 开发人员和合作伙伴在 Marketplace 中提供的数以百计的解决方案，可提高您的工作效率并加快项目实施时间。您可以在以下网址中找到 Informatica Marketplace：

<https://marketplace.informatica.com>。

Informatica 全球客户支持部门

您可以通过电话或 Informatica Network 与全球支持中心联系。

要查找您当地的 Informatica 全球客户支持部门电话号码，请访问 Informatica 网站，链接为：

<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>。

要在 Informatica Network 上查找在线支持资源，请访问 <https://network.informatica.com>，然后选择 eSupport 选项。

Web 服务概念

Web 服务概念概览

Web 服务是通过 Web 操作的业务功能。这些服务描述可通过标准化 XML 消息传输进行网络访问的操作的集合。PowerCenter Web Services Provider 允许您整合 PowerCenter 元数据和数据集成功能并作为 Web 服务公开。您可以编写应用程序，该应用程序可以以任何语言或平台与集成服务进行通信。您可以将这些应用程序轻松嵌入现有的组件和产品。

Web 服务基于 XML、SOAP 和 WSDL 等开放标准，它们可以提供比传统的专有应用程序更好的互操作性。

Web 服务示例包括业务服务，例如股票报价、航空时刻表和信用检查。

启用 Web 服务的组件包括：

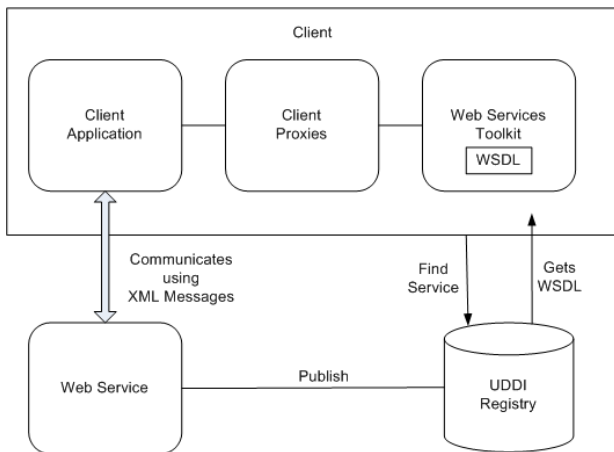
- **简单对象访问协议 (SOAP)**。SOAP 是 Web 服务所使用的通信协议。它定义 Web 服务消息的 XML 格式规范。
- **Web 服务定义语言 (WSDL)**。WSDL 是一个 XML 文档，描述了 Web 服务操作。
- **注册表**。已发布的 Web 服务的目录。部分 Web 服务提供商通过通用描述、发现和集成 (UDDI) 发布服务。可以选择在 UDDI 中注册 Web 服务。

注意：PowerCenter Web Services Provider 不使用 UDDI 注册中心。

要为 PowerCenter Web Services Provider 生成 Web 服务客户端，您可以选择要连接的 Web 服务并为选定的 Web 服务检索 WSDL。使用 Axis 等 Web 服务工具包生成客户端代理。客户端代理包含与 Web 服务交互所需的所有函数调用。

您可以通过检查 WSDL 确定 Web 服务所提供的功能、Web 服务需要的数据和服务的位置。WSDL 描述了 Web 服务界面和服务可用的操作。WSDL 中的信息用于生成客户端应用程序以使用这些服务。

下图显示了一个 Web 服务的构建块：



简单对象访问协议 (SOAP)

SOAP 是 Web 服务的通信协议。它定义了 Web 服务消息的格式。SOAP 编码用于告知 SOAP 运行时环境如何从 Java 等数据结构转化为 SOAP XML。SOAP 和 WSDL 指示 Web 服务及其客户端之间的通信。

SOAP 消息包含以下各部分：

- **SOAP 信封。**信封定义消息的框架，包括消息的内容，处理该消息的对象，以及该消息是可选的还是必需的。
- **SOAP 表头。**表头是 SOAP 信封的一个元素，它允许您以分散方式将功能添加到 SOAP 消息。
- **SOAP 正文。**正文是必需信息的容器，它提供了一个与预期收件人交换信息的机制。

身份验证和事务管理是可以作为表头条目实施的扩展的典型例子。SOAP 表头可以帮助处理 SOAP 消息正文的数据。与身份验证或事务相关的信息通常包含在表头中，因为此信息标识发送 SOAP 消息正文的实体和将处理该正文的上下文。

使用 SOAP 工具包来创建和解析 SOAP 消息。SOAP 工具包将函数调用从另一种语言转换为 SOAP 消息。例如，Apache Axis 工具包将 Java 函数调用转换为 SOAP。

使用 SOAP 在组织内外的不同平台上实施 Web 服务。每个 SOAP 实施支持不同函数调用和参数。因此，用于一个工具包的函数可能无法用于另一个工具包。

Web 服务说明语言 (WSDL)

WSDL 是一个描述 Web 服务所使用的协议和格式的 XML 文档。

WSDL 包含要传递给 Web 服务的数据的说明，以便服务请求的发送方和接收方都能理解所交换的数据。WSDL 元素还包含对该数据执行的操作说明（这样消息接收方便可以了解如何处理该数据），并包含与协议或运输的绑定（这样发件人便可以了解如何发送）。

您可以查看和下载 Web 服务中心控制台上的 PowerCenter Web Services Provider 托管的 Web 服务的 WSDL 文件。

相关主题：

- [“使用 Web 服务中心控制台” 页面上 15](#)

了解 Web Services Provider

了解 Web Services Provider 概览

Web Services Provider 是 PowerCenter Web 服务框架的提供程序实体，可使外部客户端能够通过 Web 服务访问 PowerCenter 工作流和数据集成功能。

Web Services Provider 由以下组件构成：

- **Web 服务中心。**PowerCenter 域中的应用程序服务，该服务使用 SOAP 标准接收请求并将响应发送到 Web 服务客户端。Web 服务中心与集成服务和存储库服务进行交互，以处理请求并生成响应。
- **批处理 Web 服务。**Web Services Provider 提供的一组 Web 服务操作，可允许访问集成服务进程和存储库元数据。
- **实时 Web 服务。**作为 Web 服务启用 PowerCenter 工作流时，将创建实时 Web 服务。将 PowerCenter 工作流转换为 Web 服务时，可以从 Web 服务客户端运行工作流。

Web 服务中心

Web 服务中心是 PowerCenter 域中的 Web 服务网关，可允许客户端应用程序使用 Web 服务标准和协议访问 PowerCenter 功能。通过 Web 服务中心，您可以作为 Web 服务来启用 PowerCenter 工作流。此外，还可以监视 PowerCenter 进程并获取存储库信息。

Web 服务中心允许数据集成进程保留在 PowerCenter 框架中，但会使用 Web 服务技术处理请求和响应。Web 服务中心从 Web 服务客户端接收 SOAP 消息形式的请求，并将其传递给集成服务。集成服务与存储库服务协作处理请求并将结果发送到 Web 服务中心。Web 服务中心以 SOAP 消息的形式将响应发送回 Web 服务客户端。

Web 服务中心提供 Web 服务中心控制台，您可以从中管理 Web 服务并查看和下载 Web 服务的 WSDL 文件。您可以使用 WSDL 文件创建客户端应用程序，以访问 Web 服务。

PowerCenter 安装包含 Web 服务中心。安装 PowerCenter 后，使用 Informatica Administrator 创建 Web 服务中心，并如同启用域中的其他应用程序服务一样启用它。

批处理 Web 服务

Web Services Provider 提供 Web 服务操作，可用于运行和监视工作流并访问元数据信息。这些 Web 服务操作统称为批处理 Web 服务。批处理 Web 服务操作可以提供有关与 Web 服务中心相关联的存储库中对象的信息，或者连接到集成服务以管理工作流和任务的执行或获取有关工作流和会话的信息。

批处理 Web 服务分为以下类别：

- **数据集成 Web 服务。**使用数据集成 Web 服务可连接到集成服务并运行或监视 PowerCenter 工作流。数据集成 Web 服务提供使您可以获取集成服务详细信息、计划和运行工作流、启动和停止工作流中的任务或监视并获取会话统计信息的操作。
- **元数据 Web 服务。**元数据 Web 服务提供从 PowerCenter 存储库中检索元数据的操作。使用元数据 Web 服务可获取有关存储库对象（例如文件夹、工作流和工作流任务）的信息，以帮助您在运行和监视存储库中的工作流。

实时 Web 服务

在安装后最初启动 Web 服务中心时，没有可用的实时 Web 服务。将 PowerCenter 工作流公开为 Web 服务时，将创建实时 Web 服务。您可以创建客户端以运行 Web 服务工作流并获取工作流进程的结果。Web 服务将获取 SOAP 消息请求并生成 SOAP 消息响应。

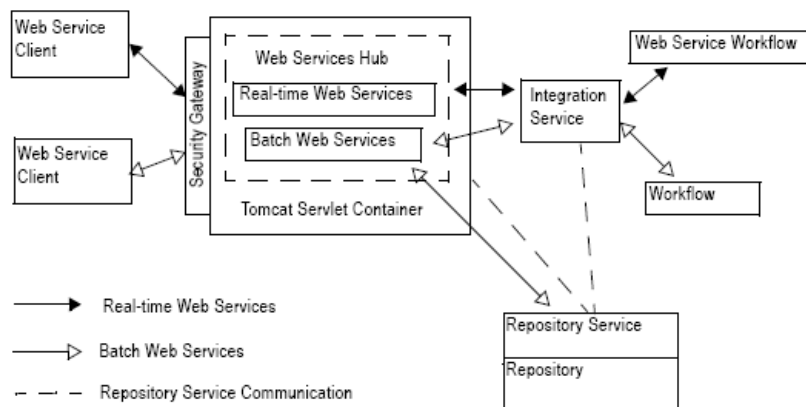
您可以创建服务映射，以从 Web 服务客户端接收消息、转换该消息并将其写入 PowerCenter 支持的任何目标中。还可以创建具有 Web 服务源和目标定义的 Web 服务映射，以从 Web 服务客户端接收消息请求、转换数据并将响应发送回 Web 服务客户端。源和目标定义代表服务操作。源定义用户请求，目标定义响应。

创建映射后，可以创建 Web 服务工作流以运行 Web 服务映射中定义的进程。Web 服务工作流是作为 Web 服务启用的工作流。配置 Web 服务工作流并将会话添加到工作流。保存工作流后，Web 服务中心将在 Web 服务中心控制台上发布该 Web 服务。集成服务可以执行请求-响应和单向服务的并行处理。

Web Services Provider 体系结构

Web Services Provider 包括 Web 服务中心以及 Web 服务中心托管的批处理和实时 Web 服务。Web 服务中心与集成服务和存储库服务协作，以处理 Web 服务请求。

下图显示了 Web Services Provider 体系结构：



Web 服务中心以类似的方式处理实时 Web 服务和批处理 Web 服务的请求。

以下流程介绍了 Web 服务中心处理 Web 服务请求的方式：

1. Web 服务客户端将 SOAP 消息发送到 Web 服务中心，以运行 Web 服务。
2. 对于批处理 Web 服务，Web 服务中心基于在登录过程中生成会话 ID 对 Web 服务客户端进行身份验证。

对于受保护的实时 Web 服务，Web 服务中心基于用户名标志对 Web 服务客户端进行身份验证。

3. Web 服务中心为请求生成消息 ID。

如果请求是针对实时 Web 服务，则 Web 服务中心会将消息发送到集成服务。

如果请求是针对批处理 Web 服务操作，则 Web 服务中心会根据请求类型将消息发送到集成服务或存储库服务。例如，如果请求是要运行工作流或启动或停止集成服务，则 Web 服务中心会将消息发送到集成服务进行处理。如果请求是要获取存储库中工作流的列表，则 Web 服务中心会将消息发送到存储库服务进行处理。

4. 集成服务或存储库服务处理该请求。

如果请求是针对实时 Web 服务，集成服务会将处理后的数据发送到 Web 服务中心，它将使用消息 ID 将请求与响应关联起来。

5. Web 服务中心将 SOAP 响应发送到 Web 服务客户端。

在整个流程中，集成服务和 Web 服务中心都会与存储库服务进行沟通。

性能和可扩展性

您可以在单个节点上运行多个 Web 服务中心。在一个节点上运行多个 Web 服务中心时，可以增加在节点上运行的 Web 服务数量，并最大限度地利用资源。

将 Web 服务中心配置为在域中运行 Web 服务时，可以使用以下选项提高性能并提供灵活性和可扩展性：

- 将多个存储库与一个 Web 服务中心相关联。
- 将一个存储库与多个 Web 服务中心相关联。
- 运行 Web 服务工作流的多个实例。
- 在网格上运行 Web 服务会话或工作流。

将多个存储库与 Web 服务中心相关联

您可以将存储库与多个 Web 服务中心相关联。如果将多个存储库与 Web 服务中心相关联，则 Web 服务中心可以运行位于任何关联存储库中的 Web 服务工作流。这允许您使用一个 Web 服务中心运行可由不同用户在不同的时间访问的 Web 服务，从而最大限度地利用 Web 服务中心。

将一个存储库与多个 Web 服务中心服务相关联

您可以将存储库与多个 Web 服务中心相关联。如果将一个存储库与多个 Web 服务中心服务相关联，则多个 Web 服务中心服务可以运行相同的 Web 服务。

使用第三方负载均衡器来管理请求并将请求分发到 Web 服务中心服务，以便服务请求负载均衡地分布于 Web 服务中心服务。在生产环境中使用的硬件负载均衡器可以优化 Web 服务的性能。在 Administrator 工具中创建 Web 服务中心服务时，设置负载均衡器的 URL。

使用批处理 Web 服务时，负载均衡器无法在多个 Web 服务中心服务间分发请求。Web 服务中心用于验证 Web 服务客户端身份的会话 ID 仅对调用登录操作的 Web 服务中心有效。在同一节点或其他节点上的多个 Web 服务中心服务不能对各个批处理 Web 服务操作使用同一会话 ID。

运行多个 Web 服务工作流实例

如果将工作流配置为在多个实例中运行，Web 服务中心可以动态启动 Web 服务的新实例，以尽可能多地处理 Web 服务请求。Web 服务中心监视 Web 服务的使用情况，以确定资源使用情况和 Web 服务处理时间。您可以为 Web 服务中心处理 Web 服务请求所需的最长时间设置阈值。如果处理时间超过该阈值，Web 服务中心将启动另一个 Web 服务工作流实例来处理新请求。

当请求服务的数量减少时，Web 服务中心可以动态关闭 Web 服务实例，以减少资源使用量。

相关主题：

- [“Web 服务工作流的并发执行” 页面上 82](#)

在网络上运行 Web 服务会话或工作流

如果 PowerCenter 域包含网格，则可以在网络上运行 Web 服务工作流。在 Administrator 工具中创建网格并将集成服务与网格相关联。然后分配要运行 Web 服务工作流的集成服务。

要从客户端应用程序在网络上运行 Web 服务工作流，请在与网格关联的集成服务上运行 Web 服务工作流。

还可以启用会话以使其在网络上运行。当会话在网络上运行时，集成服务会将会话线程分布在网格的节点中。要在网络上运行会话，请将消息 ID 添加到 Web 服务源和目标定义中。集成服务将使用该消息 ID 将分布于各个节点的 Web 服务输入和输出消息相关联。

Web 服务中心安全

Web 服务中心具有以下安全级别：

- **加密。**Web 服务中心对配置文件中用于连接到存储库的存储库登录信息进行加密。您也可以在安全模式下运行 Web 服务中心，并使用 SSL 协议对 Web 服务客户端请求进行加密。
- **身份验证。**对于批处理 Web 服务，Web 服务客户端应用程序必须在调用其他操作之前调用登录操作。Web 服务中心基于会话 ID 对请求进行身份验证。

对于受保护的实时 Web 服务，Web 服务中心基于用户名标志对 Web 服务客户端进行身份验证。Web 服务客户端必须在每个发送到 Web 服务中心的 SOAP 请求中包括用户名标志。用户名标志可以包含纯文本密码、哈希密码或摘要式密码。

Web 服务中心不会对公用实时 Web 服务的 Web 服务请求进行身份验证。

- **授权。**具有存储库访问权限的 Web 服务客户端必须对文件夹具有权限，以运行服务。对于受保护的实时 Web 服务，对文件夹具有适当权限的 Web 服务客户端可以根据服务配置运行该文件夹中的服务。例如，如果服务不可运行，则 Web 服务客户端无法启动服务，但可以在 Web 服务工作流运行时调用该服务。

相关主题：

- [“为客户端请求添加安全性” 页面上 59](#)

Web 服务中心日志

Web 服务中心可为与任务相关的状态和错误消息创建日志，如服务初始化、任务执行和连接状态。日志包括客户端 IP 地址、客户端调用的服务以及关联的工作流。可以通过检查此日志中的错误消息排除问题故障。

您可以在 Administrator 工具中查看和配置 Web 服务中心的日志。

注意：当 Web 服务中心无法处理请求时，它也会在 SOAP 响应的错误元素中写入消息。

相关主题：

- [“SOAP 故障处理” 页面上 13](#)

配置日志

PowerCenter 域中的日志管理器可处理域中所有服务的所有日志记录功能，包括 Web 服务中心。

在 Administrator 工具中，您可以配置 Web 服务中心日志的大小和位置以及将要包含在日志中的错误级别。

查看日志

您可以在管理控制台日志查看器中查看 Web 服务中心日志事件。您可以筛选日志事件，以获取仅包含 Web 服务中心的日志事件的列表。当您在日志查看器中查看日志事件时，日志管理器将在域管理员设置的日志目录中显示所生成文件的日志事件。

SOAP 故障处理

Web 服务中心将错误响应作为 SOAP 错误消息发送。Web 服务中心可以生成以下类型的错误响应：

- 用户定义的错误
- 系统错误

用户定义的错误

要将错误数据发送到目标，您可以在目标定义中定义错误视图。如果 Web 服务映射中的转换逻辑将错误数据发送到目标，集成服务会将消息写入错误目标。如果要捕获并解析特定错误，请将错误数据发送到目标。例如，您希望响应的数据类型为字符串。如果 Web 服务工作流发送数值响应，则可以将该响应发送到错误目标。然后，可以计算响应并解析该错误。

系统错误

如果 Web 服务中心遇到系统错误，将会根据错误类型生成错误消息并将响应发送到 Web 服务客户端。该错误消息基于 Web 服务中心在遇到错误时执行的任务：

- 如果 Web 服务中心无法处理 SOAP 请求消息的表头元素，则会返回与 SOAP 响应表头元素的子元素中 SOAP 请求消息的表头条目相关的错误信息。
- 如果 Web 服务中心遇到 SOAP 请求的表头元素的任何错误，则它将不处理正文元素。请求的 SOAP 响应包含 SOAP 表头中的表头错误元素和不包含详细信息元素的 SOAP 错误元素。
- 如果 Web 服务中心无法处理正文元素的内容，则 SOAP 响应消息中的 SOAP 错误元素将包含具有错误信息的详细信息元素。
- 当遇到任何以下系统错误时，Web 服务中心将生成 SOAP 错误响应，并在详细信息元素中提供错误信息：
 - 集成服务未运行，并且 Web 服务中心无法处理输入消息。
 - Web 服务中心已超时。
 - 受保护的 Web 服务不提供有效的用户名标志。

- 在下列情况下，Web 服务中心将不返回 Web 服务请求的响应：
 - 服务请求的内容格式错误，或生成解析错误。
 - workflow 筛选掉请求。

系统错误架构

默认情况下，系统错误消息包含消息代码，其中包含前缀、代码编号和消息文本。例如，消息代码 WSH_95002 与包含空 workflow 名称的无效请求相关联。

消息代码是 SOAP 错误的详细信息元素中的 ErrorCode 元素，消息文本是 SOAP 错误的 faultstring 元素。

SOAP 错误表头

Web 服务中心在 SOAP 响应表头的表头错误元素中报告表头相关错误。

此元素的架构如下所列：

```
<ns1:HeaderFault xmlns:ns1=" http://www.informatica.com/wsh" >
  <ErrorCode>
    error code
  </ErrorCode>
  <ErrorMessage>
    error message
  </ErrorMessage>
</ns1:HeaderFault>
```

SOAP 错误正文

SOAP 错误正文包含以下子元素：

- **Faultcode.**Faultcode 确定错误来源于 Web 服务客户端还是集成服务。如果消息的结构错误，则该错误可能来源于 Web 服务客户端。
- **Faultstring.**Faultstring 提供错误的说明。Faultstring 值指示错误来源于集成服务、Web 服务中心或存储库服务。
- **详细信息.**详细信息元素包含错误信息，其中包括错误代码；如果 faultstring 是 Web 服务中心或存储库服务错误，扩展详细信息元素将提供详细的错误信息。

Web 服务中心使用以下 SOAP 错误架构：

```
<SOAP-ENV: Fault>
  <faultcode> Client/Server </faultcode>
  <faultstring>Brief Description of Error</faultstring>
  <detail>
    <ns:WSHFaultDetails xmlns:ns="www.informatica.com/wsh">
      <ErrorCode>
        Error Code
      </ ErrorCode >
      <ExtendedDetails>
        Actual Error
      </ ExtendedDetails >
    </ns:WSHFaultDetails>
  </detail>
</SOAP-ENV: Fault>
```


使用 Web 服务中心控制台

使用 Web 服务中心控制台概览

Web 服务中心控制台是用于查看和测试 Web 服务中心中可用的实时 Web 服务和批处理 Web 服务操作的 PowerCenter 应用程序。使用 Web 服务中心控制台可执行以下任务：

- **查看实时 Web 服务的属性。**您可以查看 Web 服务和 Web 服务是否受保护等属性的说明。还可以查看包含 Web 服务的存储库和文件夹。
- **查看实时 Web 服务的 WSDL。**要下载 WSDL，请将 WSDL 保存到硬盘上的文件中。
- **测试实时 Web 服务。**使用 Try-It 客户端应用程序运行有效的 Web 服务并在 Web 服务中心控制台上查看响应。还可以测试批处理 Web 服务的操作。
- **查看批处理 Web 服务操作的说明。**您可以查看批处理 Web 服务操作的说明。可以使用 Try-It 应用程序查看该操作的参数。
- **查看数据集成和元数据 Web 服务的 WSDL。**要下载 WSDL，请将 WSDL 保存到硬盘上的文件中。
- **测试批处理 Web 服务操作。**使用 Try-It 应用程序运行批处理 Web 服务操作并在 Web 服务中心控制台上查看响应。

注意：Web 服务中心控制台不要求进行身份验证。无需登录即可访问 Web 服务中心控制台。为确保安全，请在安全的网络环境中运行 Web 服务中心。

连接到 Web 服务中心控制台

可以从任何浏览器连接到 Web 服务中心控制台。

使用以下 URL 之一连接到 Web 服务中心控制台：

```
http://<WebServicesHubHostName:PortNumber>/wsh  
http://<WebServicesHubHostName:PortNumber>/PowerCenter
```

上下文名称 /wsh 和 /PowerCenter 区分大小写。

在 HTTP 上运行的 Web 服务中心的默认端口是 7333。还可以将 Web 服务中心配置为使用安全的 HTTPS 连接。在 HTTPS 上运行的 Web 服务中心的默认端口是 7343。在 Administrator 工具中创建 Web 服务中心时，可以设置端口号。

还可以从 Administrator 工具连接到 Web 服务中心控制台。查看 Web 服务中心的详细信息，然后单击“服务 URL”。必须启用 Web 服务中心，才能连接到 Web 服务中心控制台。

了解 Web 服务中心控制台

Web 服务中心控制台由以下部分组成：

- **导航器。**导航器显示您可以在 Web 服务中心控制台上查看的服务类型。
- **Web 服务或操作。**对于实时 Web 服务，“Web 服务”部分将显示有效和无效的 Web 服务。对于批处理 Web 服务，“操作”部分将显示可用于元数据 Web 服务和数据集成 Web 服务的操作。

在“Web 服务”部分中，您可以测试 Web 服务或查看 Web 服务的 WSDL。

在“操作”部分中，您可以测试批处理 Web 服务操作或查看并保存批处理 Web 服务 WSDL。

- **Description.** “说明”部分提供有关在导航器中选定的 Web 服务类型的信息。
- **属性。** “属性”部分显示在“Web 服务”或“操作”部分中所选的 Web 服务或 Web 服务操作的属性。

导航器

在导航器中，您可以滚动并选择要显示其信息的 Web 服务的类型。根据在导航器中选择的 Web 服务类型，在控制台的其他部分中显示的信息有所不同。

“Web 服务”和“操作”部分

Web 服务中心控制台将显示“Web 服务”部分或“操作”部分，具体取决于在导航器中选择的 Web 服务类型。

如果在导航器中选择有效的 Web 服务或无效的 Web 服务，“Web 服务”部分将显示有关在 Web 服务中心运行的实时 Web 服务的信息。

如果在导航器中选择元数据 Web 服务或数据集成 Web 服务，“操作”部分将显示 Web 服务中心上可用的批处理 Web 服务操作。

您可以对 Web 服务或操作列表进行排序。要对 Web 服务或操作列表进行排序，请单击进行排序时所依据的列的标签。Web 服务中心控制台将根据您所单击的列按字母顺序列出 Web 服务或操作。列标签旁边的箭头显示列表的排序顺序：升序或降序。

您可以使用 Try-It 应用程序测试“Web 服务”和“操作”部分中列出的 Web 服务操作。要测试 Web 服务操作，请在该 Web 服务操作的输入消息中输入参数的值并查看响应。

在“Web 服务”部分中，您可以显示 Web 服务的 WSDL。在“操作”部分中，您可以显示批处理 Web 服务的 WSDL。WSDL 针对元数据 Web 服务或数据集成 Web 服务而发布，不是针对操作而发布。如果选择某项操作并单击 WSDL，则 Web 服务中心将显示元数据 Web 服务或数据集成 Web 服务的 WSDL。使用 WSDL 编写可调用实时 Web 服务或批处理 Web 服务操作的客户端应用程序。

Web 服务部分

必须创建 Web 服务工作流，才能在 Web 服务中心控制台上查看实时 Web 服务。将 Web 服务工作流配置为可见后，Web 服务中心将在 Web 服务中心控制台上发布 Web 服务和 WSDL。

如果在导航器中选择有效的 Web 服务或无效的 Web 服务，“Web 服务”部分将显示配置为在 Web 服务中心控制台上可见的实时 Web 服务的列表。

如果您有管理存储库中的对象的特权，则可以查看与该存储库相关联的所有 Web 服务。您可以查看但不能运行由其他用户创建的 Web 服务。例如，您对 TestRepo 存储库的运行时代象拥有创建、编辑和删除特权。在与 TestRepo 存储库相关联的 Web 服务中心的 Web 服务中心控制台上，您可以查看 TestRepo 存储库中的所有 Web 服务。您可以查看但不能运行 TestRepo 存储库中由其他用户创建的 Web 服务。

下表介绍了“Web 服务”部分中的可用选项：

标签	说明
Try-It	可用于测试选定 Web 服务的客户端应用程序。单击可运行选定的 Web 服务。 不适用于无效的 Web 服务。
WSDL	选定 Web 服务的 WSDL。单击可查看选定 Web 服务的 WSDL。可以单击该部分顶部或与选定 Web 服务位于同一行的 WSDL 按钮。要下载 WSDL，请查看该 WSDL 并将其保存到本地计算机。 不适用于无效的 Web 服务。
搜索	搜索 Web 服务。输入要搜索的文本字符串并单击“执行”。“Web 服务”部分将列出包含该文本的任何 Web 服务名称、存储库名称或工作流名称。
服务名称	可以在 Web 服务中心运行的 Web 服务的名称。
存储库名称	与 Web 服务关联的存储库的名称。
工作流名称	包含 Web 服务的工作流的名称。

“操作”部分

“操作”部分显示 Web 服务中心上可用的批处理 Web 服务操作的列表。您可以从客户端应用程序调用这些操作，以运行和监视工作流并访问 PowerCenter 元数据。

当您在导航器中选择元数据 Web 服务时，“操作”部分将显示可用元数据 Web 服务操作的列表。当您在导航器中选择数据集成 Web 服务时，“操作”部分将显示可用数据集成 Web 服务操作的列表。

下表介绍了“操作”部分中的图标和信息：

标签	说明
Try-It	测试选定操作的应用程序。单击可运行选定的元数据 Web 服务或数据集成 Web 服务操作。
WSDL	元数据或数据集成 Web 服务的 WSDL。单击可查看 Web 服务中心为所选 Web 服务类型发布的 WSDL。如果在导航器中选择元数据 Web 服务，请单击 WSDL 以查看元数据 Web 服务的 WSDL。如果在导航器中选择数据集成 Web 服务，请单击 WSDL 以查看数据集成 Web 服务的 WSDL。要下载 WSDL，请查看该 WSDL 并将其保存到本地计算机。
搜索	搜索操作。输入要搜索的文本字符串并单击“执行”。“操作”部分将列出包含该文本的任何操作名称或说明。
操作名称	元数据 Web 服务或数据集成 Web 服务操作的名称。
说明	元数据 Web 服务或数据集成 Web 服务操作的说明。

“属性”部分

“属性”部分显示在“Web 服务”或“操作”部分中所选的 Web 服务或 Web 服务操作的信息。

实时 Web 服务的“属性”部分

如果在“Web 服务”部分中选择有效或无效的实时 Web 服务，“属性”部分将显示选定 Web 服务的属性。

下表介绍了实时 Web 服务属性：

属性	说明
服务名称	Web 服务的名称。
域名	包含 Web 服务中心的 PowerCenter 域的名称。
存储库名称	包含 Web 服务工作流的存储库。
文件夹名称	包含 Web 服务工作流的文件夹的名称。
工作流名称	与 Web 服务关联的工作流的名称。
说明	Web 服务的说明。
是否可运行	指示 Web 服务是否可以由客户端应用程序启动。 如果为 True，则 Web 服务客户端可以启动 Web 服务工作流或在工作流运行时调用 Web 服务。 如果为 False，则 Web 服务客户端可以在工作流运行时调用 Web 服务，但不能启动工作流。
是否受保护	指示 Web 服务受保护或公用。 如果为 True，则 Web 服务客户端请求必须通过身份验证。SOAP 请求必须在表头中包含有效的用户名标志。 如果为 False，则任何 Web 服务客户端都可以运行 Web 服务请求，而无需身份验证。
是否为单向	指示 Web 服务使用单向映射还是请求-响应映射。

批处理 Web 服务的“属性”部分

当您在“操作”部分中选择元数据 Web 服务操作或数据集成 Web 服务操作时，“属性”部分将显示选定 Web 服务操作的名称和说明。

测试 Web 服务

Try-It 应用程序是一个客户端应用程序，您可以使用它来运行 Web 服务中心控制台中列出的实时或批处理 Web 服务操作。使用 Try-It 应用程序可测试有效的 Web 服务操作并在 Web 服务中心控制台上查看结果。如果您不确定输入消息中需要哪些参数，或者希望查看特定输入消息的响应，则可以使用 Try-It 应用程序。

可以使用 Try-It 应用程序运行 Web 服务或调用操作，而无需下载 WSDL，并为客户端应用程序生成客户端代理类。您可以在控制台上查看响应，并确定客户端应用程序应如何处理来自 Web 服务的响应。

您可以测试有效的实时 Web 服务应用程序或批处理 Web 服务操作。不能使用 Try-It 应用程序来测试其 WSDL 包含 SOAP 附件的 Web 服务。

受保护的实时 Web 服务要求进行身份验证。要测试受保护的 Web 服务操作，请提供一个有效的用户名标志，以登录到 PowerCenter 存储库。

输入消息

Try-It 应用程序提供了两种方法来创建 Web 服务请求：

- XML 输入
- 表单输入

使用最符合请求要求的方法。例如，如果请求中包含多次出现的元素，请使用 XML 输入创建请求消息。

XML 输入

如果选择“XML 输入”选项卡，Web 服务中心将显示一条 SOAP 输入消息，其中包含运行服务请求操作所需的元素。输入 SOAP 消息中元素的值。或者，您可以在 Web 服务中心控制台以外创建 SOAP 消息，然后将其粘贴到 XML 输入部分。

Web 服务中心使用 SOAP 输入消息来运行 Web 服务。它会将响应显示为 SOAP 输出消息。

表单输入

如果选择“表单输入”选项卡，Web 服务中心将显示 Web 服务请求的参数列表。输入参数的值。如果 Web 服务请求中包含复杂类型的元素，“表单输入”选项卡将显示正确的层次结构中的输入参数。

Web 服务中心使用您输入的参数值创建 SOAP 输入消息并运行 Web 服务。它会将响应显示为 SOAP 输出消息。

测试公共 Web 服务或批处理 Web 服务操作

要测试公共或批处理 Web 服务，请选择一个 Web 服务操作并在该 Web 服务操作的输入消息中输入参数的值。

要测试公共 Web 服务或批处理 Web 服务操作，请执行以下操作：

1. 在“Web 服务”或“操作”部分中，选择有效的实时 Web 服务或操作。
2. 单击 Try-It。

Try-It 应用程序窗口将显示您可以测试的 Web 服务操作的列表以及有关如何运行该 Try-It 应用程序的说明。

3. 选择要测试的操作。

Try-It 应用程序窗口将显示输入消息的参数。

4. 单击“XML 输入”选项卡以输入 SOAP 消息格式的输入参数。

或者，单击“表单输入”选项卡以在参数输入表单中输入输入参数。

5. 输入参数的值。

WSDL 可以包含用户定义的数据类型。为了避免错误响应，请根据数据类型输入参数的值。

6. 单击“发送”。

Web 服务中心将运行 Web 服务操作，并显示 SOAP 消息响应和一条指示成功或失败的消息。

7. 要清除参数并输入新值，请单击“重置”。

8. 单击 Web 浏览器的“关闭”按钮以退出 Try-It 应用程序窗口并返回 Web 服务中心控制台的主页面。

测试受保护的实时 Web 服务

要测试受保护的实时 Web 服务，请在 SOAP 表头中包含有效的用户名标志。您可以在“表单输入”选项卡中输入用户名和密码，或者在“XML 输入”选项卡中修改 SOAP 消息以包括用户名标志的所有元素。

在“表单输入”或“XML 输入”选项卡中，您可以测试具有纯文本或哈希密码的受保护 Web 服务。要测试具有哈希密码的受保护 Web 服务，请使用 MD5 或 SHA-1 哈希函数加密密码，然后再测试 Web 服务。加密必须以 Base64 进行编码。将由此产生的哈希值用作 Web 服务的密码。

在“XML 输入”选项卡中，您可以测试具有摘要式密码的受保护 Web 服务。要测试具有摘要式密码的受保护 Web 服务，请在摘要式密码的 UsernameToken 元素中添加 Password 属性和需要的元素。

要测试受保护 Web 服务，请执行以下操作：

1. 在实时 Web 服务的“Web 服务”部分中，选择要运行的受保护 Web 服务并单击 Try-It。
2. 在 Try-It 应用程序窗口中，选择用于受保护 Web 服务的操作。
3. 要使用表单输入测试 Web 服务操作，请单击“表单输入”选项卡。

在 SOAP 表头部分中，输入用户名以及纯文本或哈希密码。

在 SOAP 正文部分中，输入受保护 Web 服务所需参数的值。

-或-

要使用 XML 输入测试 Web 服务操作，请单击“XML 输入”选项卡并更新 UsernameToken 元素。

要测试使用纯文本或哈希密码的受保护 Web 服务，请将 Username 和 Password 子元素中的值 *[string]* 替换为有效用户名和密码：

```
<UsernameToken>
  <Username>[string]</Username>
  <Password>[string]</Password>
</UsernameToken>
```

要测试使用摘要式密码的受保护 Web 服务，请将 Username 元素中的值 *[string]* 替换为有效用户名。

使用适当的值更新 Password 元素并添加 Nonce 和 Created 元素：

```
<UsernameToken>
  <Username>[string]</Username>
  <Password Type="PasswordDigest">[string]</Password>
  <Nonce>[NonceValue]</Nonce>
  <Created>[RequestCreationTimestamp]</Created>
</UsernameToken>
```

有关 UsernameToken 元素的详细信息，请参阅 [“SOAP 请求中的 UsernameToken” 页面上 59](#)。

在 SOAP 正文部分中，输入受保护 Web 服务所需参数的值。

4. 单击“发送”。

Web 服务中心将运行受保护的 Web 服务操作，并在控制台上显示 SOAP 消息响应。

5. 单击 Web 浏览器的“关闭”按钮以退出 Try-It 应用程序窗口并返回 Web 服务中心控制台的主页面。

批处理 Web 服务操作

批处理 Web 服务操作概览

您可以使用批处理 Web 服务操作来计划、启动或停止现有工作流和任务。您可以获取会话统计信息和性能数据。您可以检索工作流和会话日志。

批处理 Web 服务包括在单独 WSDL 中定义的以下服务组：

- **元数据 Web 服务。**Web 服务中心控制台的“批处理 Web 服务”页面上提供的元数据 WSDL 中定义了元数据 Web 服务操作。
- **数据集成 Web 服务。**Web 服务中心控制台的“批处理 Web 服务”页面上提供的数据集成 WSDL 中定义了数据集成 Web 服务操作。

本章介绍了批处理 Web 服务提供的操作。有关这些操作的请求和响应 XML 文档的详细信息，请参阅 WSDL 文件。

注意：批处理 Web 服务操作调用获取的日志段采用集成服务代码页或 UTF-8 格式。

元数据 Web 服务操作

使用元数据 Web 服务中提供的操作从与 Web 服务中心关联的 PowerCenter 存储库检索元数据。

您可以使用这些操作登录到存储库并获取以下存储库对象的列表：

- 与 Web 服务中心关联的存储库中的所有文件夹
- 文件夹中的所有工作流
- 工作流中的所有工作集和会话任务
- 与存储库关联的所有集成服务
- 与 Web 服务中心关联的所有存储库

本节列出了元数据 Web 服务可用的所有操作。

getAllServers

使用此操作检索与存储库关联的所有集成服务的名称。

您可以将一个或多个集成服务与存储库关联以运行工作流和会话。在多个集成服务环境中，为每个关联服务输入描述性服务名称是很重要的，这有助于用户区分各个集成服务。与存储库关联的每个集成服务必须具有服务名称及主机名与端口号的组合（在与存储库关联的服务间唯一）。

此操作返回与给定存储库关联的所有集成服务的名称。

请使用 SessionID 参数调用 getAllServers 操作。登录到存储库后会生成会话 ID。

getAllFolders

使用此操作检索存储库中的所有文件夹。

请使用 SessionID 参数调用 getAllFolders 操作。登录到存储库后会生成会话 ID。

getAllRepositories

使用此操作查看与 Web 服务中心关联的所有存储库。getAllRepositories 操作不需要参数。

在 Web 服务中心客户端应用程序可以使用存储库之前，您必须将存储库与 Web 服务中心关联。使用 Administrator 工具将存储库与 Web 服务中心关联。

注意: 由于 getAllRepositories 操作不与具体存储库关联，要使用此操作，您无需登录到存储库。您可以在不调用 Login 操作的情况下调用 getAllRepositories 操作。

getAllTaskInstances

使用此操作获取有关指定深度的工作流中的所有工作集和会话任务实例的信息。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
深度	必需	从中获取任务实例信息的工作流任务层次结构中的级别数。
WorkflowName	必需	包含工作集和会话的工作流的名称。
FolderName	必需	包含工作流的文件夹的名称。
IsValid	可选	指定获取有效或无效的任务实例。设置为 True 时可获取有效任务实例。

getAllWorkflows

使用此操作获取有关文件夹中所有工作流的信息。工作流是一组指令，指示集成服务如何执行会话、电子邮件通知和 Shell 命令等任务。工作流信息包括工作流名称、工作流所在文件夹的名称及工作流是否有效的信息。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
名称	必需	包含工作流的文件夹的名称。

登录

使用此操作登录到存储库。Login 操作对指定存储库的用户名和密码进行身份验证。调用其他任何操作之前，客户端应用程序必须调用此操作。调用 Login 操作之后，Web 服务客户端应用程序可以调用任何批处理 Web 服务操作。

Login 操作需要存储库名称、用户名和密码。该操作返回加密的会话 ID。存储库与 Web 服务中心必须位于相同的域。

下表介绍了此操作的参数：

参数名	必需/可选	说明
RepositoryName	必需	登录到的存储库的名称。
用户名	必需	用于登录到存储库的用户名。
密码	必需	用于登录到存储库的用户帐户的密码。
RepositoryDomainName	可选	存储库服务的域名。
UserNameSpace	条件	用于登录到存储库的用户帐户的安全域。如果 PowerCenter 域中有多个安全域，该参数是必需的。

会话到期时间段

超过非活动时间段后会话 ID 到期。配置 Web 服务中心的 **SessionExpiryPeriod** 高级属性可指定会话到期之前允许的非活动时长。您可以在 Administrator 工具中配置 Web 服务中心高级属性。

Web 服务中心在会话开始时缓存用户名和密码。会话到期之前，Web 服务中心将缓存的用户名和密码用于登录身份验证。会话到期前，对用户帐户进行的更改不会影响身份验证。如果删除或禁用用户帐户，则在会话到期前不会删除或禁用该用户帐户。如果锁定用户帐户，会话到期之前，锁定的用户可发送请求。

注意：如果将 **SessionExpiryPeriod** 设置为较高的值，一段时间之后用户帐户更改才会生效。为了避免安全漏洞，请勿删除或修改在会话中使用的用户帐户。

注销

使用此操作注销存储库。Logout 操作断开与存储库和集成服务的连接。可在客户端应用程序运行结束时调用此操作，以此释放资源和防止存储库服务和 Web 服务中心进程中可能产生的内存泄漏。

请使用 SessionID 参数调用 Logout 操作。登录到存储库后会生成会话 ID。

数据集成 Web 服务操作

您可以使用数据集成 Web 服务操作执行以下任务：

- **连接到存储库服务。**您可以使用以下操作登录或注销存储库：
 - 登录
 - 注销
- **连接到集成服务并获取其相关详细信息。**您可以使用以下操作验证集成服务是否正在运行，并连接到集成服务或获取其相关信息。
 - pingDIServer
 - getDIServerProperties
 - initializeDIServerConnection（已弃用）
 - deinitializeDIServerConnection（已弃用）

- **计划和运行工作流。**您可以使用以下操作来管理工作流运行：
 - startWorkflow
 - startWorkflowEx
 - stopWorkflow
 - scheduleWorkflow
 - startWorkflowFromTask
 - unscheduleWorkflow
 - waitTillWorkflowComplete
- **启动和停止工作流中的任务。**您可以使用以下操作来控制工作流中的任务：
 - recoverWorkflow
 - resumeWorkflow（已弃用）
 - startTask
 - stopTask
 - waitTillTaskComplete
- **获取有关会话的统计信息。**您可以使用以下操作来获取有关会话或工作流运行的详细信息。
 - getNextLogSegment（已弃用）
 - getSessionLog
 - getSessionPerformanceData
 - getSessionStatistics
 - getTaskDetails
 - getTaskDetailsEx
 - getWorkflowDetails
 - getWorkflowDetailsEx
 - getWorkflowLog
 - monitorDIServer
 - startSessionLogFetch（已弃用）
 - startWorkflowLogFetch（已弃用）

di.wsdl 中定义数据集成 Web 服务操作。

deinitializeDIServerConnection

已弃用的操作。您无需明确断开客户端应用程序与集成服务的连接。Logout 操作释放客户端应用程序获得的与集成服务的连接并执行清理操作。

此操作需要 initializeDIServerConnection 的 SessionID 参数。登录到存储库后会生成会话 ID。

getDIServerProperties

使用此操作获取集成服务的属性。调用此操作之前，您必须登录到存储库。

集成服务属性包含以下信息：

- 集成服务名称
- 集成服务版本
- 产品名称
- 集成服务启动时间
- 与集成服务关联的存储库的名称
- 数据移动模式（ASCII 或 Unicode）
- 集成服务是否可以调试映射

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
ServiceName	必需	要获取其属性的集成服务的名称。
DomainName	可选	集成服务的域名。

getNextLogSegment

已弃用的操作。此操作以增量方式获取会话或工作流日志的信息。

getNextLogSegment 操作返回会话或工作流日志的一部分。调用此操作之前，您必须登录到存储库。

请将此操作与 startSessionLogFetch 或 startWorkflowLogFetch 操作配合使用。可使用 startSessionLogFetch 或 startWorkflowLogFetch 操作生成的日志句柄调用 getNextLogSegment 操作，直到到达日志结尾。

要在单个操作中获取会话日志信息，请使用 getSessionLog 操作。要在单个操作中获取工作流日志信息，请使用 getWorkflowLog 操作。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
LogHandle	必需	startSessionLogFetch 或 startWorkflowLogFetch 操作生成的日志 ID。调用此操作之前，必须调用 startSessionLogFetch 或 startWorkflowLogFetch 操作。
超时	必需	在此操作过程中，客户端请求保留与 Web 服务中心的连接的时间。如果无法在超时时间段内完成操作，则操作失败。调用 startSessionLogFetch 或 startWorkflowLogFetch 操作时，超时时间段开始。

getSessionLog

使用此操作在单次操作中获取会话日志中的所有信息。调用此操作之前，您必须登录到存储库。

当服务会话运行时，集成服务将进程初始化、会话验证、读取器和写入器线程的 SQL 命令创建、遇到的错误及加载摘要等信息写入会话日志。会话日志中的详细信息量取决于您设置的跟踪级别。getSessionLog 操作返回会话日志中的信息。

要以增量方式获取会话日志信息，请使用 getNextLogSegment 操作。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含会话的工作流的名称。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i>WorkletName.TaskName</i> 。作为完全限定字符串输入 taskInstancePath。
超时	必需	在此操作过程中，客户端请求保留与 Web 服务中心的连接的时间。如果无法在超时时间段内完成操作，则操作失败。
ServiceName	必需	运行工作流的集成服务的名称。
DomainName	可选	集成服务的域名。

getSessionPerformanceData

使用此操作检索在集成服务上运行的会话的性能数据。性能详细信息提供有助于您了解会话和映射效率的计数器。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含会话的工作流的名称。
TaskInstancePath	必需	指定会话位置的路径。如果会话在工作流中，则仅输入会话名称。如果会话在工作集中，则输入 <i>WorkletName.SessionName</i> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
DomainName	可选	集成服务的域名。

getSessionStatistics

使用此操作获取在集成服务上运行的会话的统计信息。当会话不运行时，此操作提供最近运行会话的统计信息。

调用此操作之前，您必须登录到存储库。

会话统计信息包括文件夹和工作流名称、会话和任务运行状态、错误信息、源和目标的成功和失败的行数及已应用、受影响和已拒绝的行数。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含会话任务的工作流的名称。
TaskInstancePath	必需	指定会话位置的路径。如果会话在工作流中，则仅输入会话名称。如果会话在工作集中，则输入 <i>WorkletName.SessionName</i> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
DomainName	可选	集成服务的域名。

getTaskDetails

使用此操作从集成服务中检索任务详细信息。如果父工作流正在运行且任务已经运行，该操作返回正在运行的工作流中的当前任务的详细信息。如果父工作流未运行，该操作返回上次工作流运行的任务详细信息。

调用此操作之前，您必须登录到存储库。

任务详细信息包括文件夹和工作流名称、任务名称和类型、开始时间、运行状态及运行错误代码和消息。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含任务的工作流的名称。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	包含任务的工作流运行实例的 ID。此参数不可用。

参数名	必需/可选	说明
WorkflowRunInstanceName	可选	包含任务的工作流运行实例的名称。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
RequestMode	可选	指明会话任务的恢复策略： <ul style="list-style-type: none"> - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
IsAbort	可选	指明是否中止任务。不适用于此操作。
DomainName	可选	集成服务的域名。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

getTaskDetailsEx

当多个工作流实例同时运行时，使用此操作通过集成服务检索任务的详细信息。

getTaskDetailsEx 操作类似于 getTaskDetails 操作，但返回有关任务的所有实例的信息。如果父工作流正在运行且任务已经运行，该操作返回正在运行的工作流中的所有任务实例的详细信息。如果父工作流未运行，该操作返回上次工作流运行中的任务实例的详细信息。该操作返回的集成服务详细信息包含任务详细信息。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含任务的工作流的名称。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <code><WorkletName>.<TaskName></code> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
DomainName	可选	集成服务的域名。

getWorkflowDetails

使用此操作获取给定工作流的详细信息。如果工作流正在运行，该操作返回正在运行的工作流的详细信息。如果工作流未运行，该操作返回此工作流上次运行的详细信息。

工作流详细信息包括文件夹名称、工作流、工作流日志文件和运行工作流的用户。它包括工作流运行类型、日志文件代码页、开始和结束时间、运行状态及运行错误代码和消息。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none">- 普通。重新启动会话而不恢复。- 恢复。恢复会话。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <code><WorkletName>.<TaskName></code> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。

参数名	必需/可选	说明
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

getWorkflowDetailsEx

当多个工作流实例同时运行时，使用此操作获取工作流的详细信息。

此操作类似于 getWorkflowDetails 操作，但返回有关工作流的所有实例的信息。如果工作流正在运行，该操作返回正在运行的工作流的所有实例的详细信息。如果工作流未运行，该操作返回上次工作流运行的详细信息。该操作返回的集成服务详细信息包含工作流详细信息。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	工作流的名称。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
DomainName	可选	集成服务的域名。

getWorkflowLog

使用此操作在单次操作中获取工作流日志的所有信息。调用此操作之前，您必须登录到存储库。

当 Web 服务工作流运行时，集成服务将进程初始化、工作流任务运行信息、遇到的错误和工作流运行摘要信息写入工作流日志。工作流日志的数量取决于跟踪级别。getWorkflowLog 操作返回工作流日志中的信息。

要以增量方式获取工作流日志信息，请使用 getNextLogSegment 操作。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。

参数名	必需/可选	说明
WorkflowName	必需	工作流的名称。
超时	必需	在此操作过程中，客户端请求保留与 Web 服务中心的连接的时间。如果无法在超时时间段内完成操作，则操作失败。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
DomainName	可选	集成服务的域名。

initializeDIServerConnection

已弃用的操作。您无需初始化与集成服务的连接。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
DIServerName	必需	包含工作流的文件夹的名称。
LoginHandle	可选	与会话 ID 相同。
DIServerDomain	可选	集成服务的域名。

登录

数据集成 Web 服务和元数据 Web 服务包含 Login 操作。

使用此操作登录到存储库。Login 操作对指定存储库的用户名和密码进行身份验证。调用其他任何操作之前，客户端应用程序必须调用此操作。调用 Login 操作之后，Web 服务客户端应用程序可以调用任何批处理 Web 服务操作。

Login 操作需要存储库名称、用户名和密码。该操作返回加密的会话 ID。存储库与 Web 服务中心必须位于相同的域。

下表介绍了此操作的参数：

参数名	必需/可选	说明
RepositoryName	必需	登录到的存储库的名称。
用户名	必需	用于登录到存储库的用户名。

参数名	必需/可选	说明
密码	必需	用于登录到存储库的用户帐户的密码。
RepositoryDomainName	可选	存储库服务的域名。
UserNameSpace	条件	用于登录到存储库的用户帐户的安全域。如果 PowerCenter 域中有多个安全域，该参数是必需的。

会话到期时间段

超过非活动时间段后会话 ID 到期。配置 Web 服务中心的 **SessionExpiryPeriod** 高级属性可指定会话到期之前允许的非活动时长。您可以在 Administrator 工具中配置 Web 服务中心高级属性。

Web 服务中心在会话开始时缓存用户名和密码。会话到期之前，Web 服务中心将缓存的用户名和密码用于登录身份验证。会话到期前，对用户帐户进行的更改不会影响身份验证。如果删除或禁用用户帐户，则在会话到期前不会删除或禁用该用户帐户。如果锁定用户帐户，会话到期之前，锁定的用户可发送请求。

注意: 如果将 **SessionExpiryPeriod** 设置为较高的值，一段时间之后用户帐户更改才会生效。为了避免安全漏洞，请勿删除或修改在会话中使用的用户帐户。

注销

数据集成 Web 服务和元数据 Web 服务包含 Logout 操作。

使用此操作注销存储库。Logout 操作断开与存储库和集成服务的连接。可在客户端应用程序运行结束时调用此操作，以此释放资源和防止存储库服务和 Web 服务中心进程中可能产生的内存泄漏。

请使用 SessionID 参数调用 Logout 操作。登录到存储库后会生成会话 ID。

monitorDIServer

使用此操作检索集成服务的状态、活动工作流和计划的工作流的详细信息，以及工作流内任务和链接的详细信息。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
MonitorMode	必需	要监视的工作流的模式： <ul style="list-style-type: none"> - 运行。返回活动工作流的状态详细信息。活动工作流包括正在运行、已挂起和正在挂起的工作流。 - 计划。返回计划的工作流的状态详细信息。 - 全部。返回所有计划的和活动的工作流的信息。
ServiceName	必需	要监视的集成服务的名称。
DomainName	可选	集成服务的域名。

pingDIserver

使用此操作确定某项集成服务是否在运行。返回值为 ALIVE 或 FAIL。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
超时	必需	在此操作过程中，客户端请求保留与 Web 服务中心的连接的时间。如果无法在超时时间段内完成操作，则操作失败。
ServiceName	必需	要对其执行 ping 操作的集成服务的名称。
DomainName	可选	集成服务的域名。

recoverWorkflow

使用此操作恢复已挂起的工作流。集成服务从所有已挂起和失败的工作集，以及所有已挂起和失败的命令、电子邮件和会话任务中恢复工作流。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要恢复的工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none">- 普通。重新启动会话而不恢复。- 恢复。恢复会话。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <code><WorkletName>.<TaskName></code> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。

参数名	必需/可选	说明
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

resumeWorkflow

已弃用的操作。请改用 `recoverWorkflow` 操作。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要恢复的工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none"> - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。

参数名	必需/可选	说明
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

scheduleWorkflow

使用此操作计划一个工作流。可以计划任何非按需运行的工作流。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要计划的工作流的名称。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	要计划的工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。

参数名	必需/可选	说明
RequestMode	可选	指明会话任务的恢复策略： <ul style="list-style-type: none"> - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于计划工作流的属性名称和值对中的属性名称。
AttributeValue	可选	用于计划工作流的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

startSessionLogFetch

已弃用的操作。此操作开始以增量方式提取会话日志中的信息。

startSessionLogFetch 生成供 getNextLogSegment 操作使用的日志句柄。调用 startSessionLogFetch 操作之后，在到达日志结尾之前，使用 startSessionLogFetch 生成的日志句柄调用 getNextLogSegment 操作。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含会话的工作流的名称。
ServiceName	必需	运行工作流的集成服务的名称。

startTask

使用此操作启动工作流内的具体任务。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含任务的工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none">- 普通。重新启动会话而不恢复。- 恢复。恢复会话。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	包含任务的工作流运行实例的 ID。
WorkflowRunInstanceName	可选	包含任务的工作流运行实例的名称。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
IsAbort	可选	指明是否中止任务。不适用于此操作。
DomainName	可选	集成服务的域名。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

startWorkflow

使用此操作启动工作流。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要运行的工作流的名称。

参数名	必需/可选	说明
RequestMode	必需	指明会话任务的恢复策略： - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	要运行的工作流运行实例的名称。
原因	可选	说明启动工作流的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动工作流的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动工作流的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

startWorkflowEx

使用此操作启动工作流。startWorkflowEx 操作返回工作流的运行实例 ID。请使用 startWorkflowEx 操作而不是 startWorkflow 操作来获取操作启动的工作流的运行 ID。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要运行的工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none">- 普通。重新启动会话而不恢复。- 恢复。恢复会话。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunInstanceName	可选	要运行的工作流运行实例的名称。
原因	可选	说明启动工作流的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动工作流的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动工作流的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

startWorkflowFromTask

使用此操作从一项任务启动工作流。通过任务启动工作流时，集成服务从所选任务开始运行工作流，直到工作流结尾。

您必须指定要启动的任务的任务实例路径。任务实例路径标识工作流内的任务实例。工作流内的任务仅通过其任务名称标识。工作集内的任务通过其工作集和任务名称（由句号分隔）标识：

WorkletName.TaskName。例如，某个工作流包含工作集 A，而工作集 A 包含另一工作集 B。任务 C 为工作集 B 内的任务。任务 C 的任务实例路径为 A.B.C。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要运行的工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none">- 普通。重新启动会话而不恢复。- 恢复。恢复会话。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	要启动的工作流运行实例的名称。
原因	可选	说明启动工作流的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动工作流的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动工作流的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

startWorkflowLogFetch

已弃用的操作。此操作开始以增量方式提取工作流日志中的信息。

startWorkflowLogFetch 生成用于 getNextLogSegment 操作的日志句柄。调用 startWorkflowLogFetch 操作之后，在到达日志结尾之前，使用 startWorkflowLogFetch 生成的日志句柄调用 getNextLogSegment 操作。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	工作流的名称。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
DomainName	可选	集成服务的域名。

stopTask

使用此操作停止集成服务上正在运行的某项任务。可随时停止或中止任务、工作流或工作集。当您停止工作流中的某项任务时，集成服务停止处理该任务及其路径中的所有其他任务。

您还可以通过将 isAbort 参数设置为 true 以中止正在运行的任务。指定要中止的任务的任务实例路径。通常情况下，仅当集成服务无法停止任务时中止任务。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含任务的工作流的名称。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <code><WorkletName>.<TaskName></code> 。作为完全限定字符串输入 taskInstancePath。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	包含任务的工作流运行实例的 ID。
WorkflowRunInstanceName	可选	包含任务的工作流运行实例的名称。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。

参数名	必需/可选	说明
RequestMode	可选	指明会话任务的恢复策略： - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
IsAbort	可选	指明是否中止任务。将此参数设置为 True 后可中止任务。
DomainName	可选	集成服务的域名。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

相关主题：

- [“startWorkflowFromTask” 页面上 39](#)

stopWorkflow

使用此操作停止正在运行的工作流。停止工作流时，集成服务尝试停止工作流中当前正在运行的所有任务。如果工作流包含工作集，集成服务还尝试停止工作集中当前正在运行的所有任务。

除了停止工作流外，还可以将 isAbort 参数设置为 true 以中止正在运行的工作流。通常情况下，仅当集成服务无法停止工作流时中止工作流。

调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要停止的工作流的名称。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	要停止的工作流运行实例的 ID。
WorkflowRunInstanceName	可选	要停止的工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
RequestMode	可选	指明会话任务的恢复策略： - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。

参数名	必需/可选	说明
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止工作流。将此参数设置为 True 后可中止工作流。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

unscheduleWorkflow

使用此操作取消工作流的计划。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	要对其取消计划的工作流的名称。
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	要对其取消计划的工作流运行实例的 ID。
WorkflowRunInstanceName	可选	要对其取消计划的工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。

参数名	必需/可选	说明
RequestMode	可选	指明会话任务的恢复策略： - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

waitTillTaskComplete

使用此操作等待集成服务上正在运行的任务完成。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	包含任务的工作流的名称。
TaskInstancePath	必需	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
RequestMode	必需	指明会话任务的恢复策略： - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。

参数名	必需/可选	说明
ServiceName	必需	运行工作流的集成服务的名称。
WorkflowRunId	可选	包含任务的工作流运行实例的 ID。
WorkflowRunInstanceName	可选	包含任务的工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

waitTillWorkflowComplete

要仅在其他工作流完成后运行某项操作或工作流，请使用此操作以在其他工作流完成后获取通知。调用此操作之前，您必须登录到存储库。

下表介绍了此操作的参数：

参数名	必需/可选	说明
SessionID	必需	登录后生成的会话 ID。
FolderName	必需	包含工作流的文件夹的名称。
WorkflowName	必需	工作流的名称。
RequestMode	必需	指明会话任务的恢复策略： <ul style="list-style-type: none"> - 普通。重新启动会话而不恢复。 - 恢复。恢复会话。
ServiceName	必需	运行工作流的集成服务的名称。

参数名	必需/可选	说明
WorkflowRunId	可选	工作流运行实例的 ID。
WorkflowRunInstanceName	可选	工作流运行实例的名称。
原因	可选	说明启动工作流或任务的原因。
ParameterFileName	可选	运行工作流时使用的参数文件的名称。
TaskInstancePath	可选	指定任务位置的路径。如果任务在工作流中，则仅输入任务名称。如果任务在工作集中，则输入 <i><WorkletName>.<TaskName></i> 。作为完全限定字符串输入 taskInstancePath。
IsAbort	可选	指明是否中止任务。不适用于此操作。
OSUser	可选	指定分配给工作流的操作系统配置文件。
DomainName	可选	集成服务的域名。
AttributeName	可选	用于启动或计划工作流或任务的属性名称和值对中的属性名称。
AttributeValue	可选	用于启动或计划工作流或任务的属性名称和值对中的属性值。
键	可选	用于启动工作流或任务的键。
MustUse	可选	指明是否必须使用键启动工作流或任务。
ParameterScope	可选	参数数组定义中的参数的范围。
ParameterName	可选	参数数组定义中的参数的名称。
ParameterValue	可选	参数数组定义中的参数的值。

编写客户端应用程序

编写客户端应用程序概览

本章提供如何编写客户端应用程序以使用 PowerCenter Web Services Provider 提供的 Web 服务的概览。在有关创建客户端应用程序的步骤的一般论述之后，还提供了在 Java 和 .NET 框架中创建客户端应用程序的示例。

要创建 PowerCenter Web 服务的客户端应用程序，您需要 Web 服务 WSDL 和 Web 服务工具包。Web 服务工具包有助于通过从 Web 服务 WSDL 生成客户端代理类来创建客户端应用程序。您可以使用 Microsoft .NET 和 Apache Axis Web 服务工具包编写 PowerCenter Web 服务的客户端应用程序。

您可以创建客户端应用程序以运行 PowerCenter 批处理或实时 Web 服务。应用程序开发遵循相同的基本步骤。

注意: Web 服务中心可以处理块消息。要在客户端请求中启用分块的传输编码, 请将以下表头添加到 SOAP 消息:

```
TRANSFER_ENCODING=chunked
```

批处理 Web 服务的客户端应用程序

开发可访问 PowerCenter Web Services Provider 中提供的批处理 Web 服务的客户端应用程序时, 将涉及以下元素:

- 客户端代理类
- 初始化
- 会话维护
- 操作调用
- 资源清理
- 错误处理
- 代理对象

生成客户端代理类

要使用 Web 服务中心中提供的批处理 Web 服务操作, 您需要使用 Web 服务工具包生成 Web 服务 WSDL 的客户端代理类。

要生成客户端代理, 请完成以下步骤:

1. 为开发时所用的平台和语言选择 Web 服务工具包。
2. 从 Web 服务中心控制台下载元数据 Web 服务和数据集成 Web 服务的 WSDL 文件。默认情况下, 从 Web 服务中心控制台下载 WSDL 文件时, 端点 URL 设置为 Web 服务中心的主机名和端口号。生成代理类之前, 请验证 WSDL 文件是否包含正确的端点 URL。
3. 使用 Web 服务工具包通过 WSDL 文件生成客户端代理类。有关生成代理类的详细信息, 请参阅 Web 服务工具包文档。各种工具包以特定方式生成客户端代理类。

初始化

执行元数据和数据集成 Web 服务操作调用之前, 客户端应用程序执行初始化步骤。

要执行初始化, 请完成以下步骤:

1. 实例化元数据 API 代理类。
在示例中, 元数据 API 代理对象的名称为 *MWSPProxy*。本节使用名称 *MWSPProxy* 指代该元数据 API 代理对象。
2. 实例化数据集成 API 代理类。
在示例中, 数据集成 API 代理对象的名称为 *DIWSPProxy*。本节使用名称 *DIWSPProxy* 指代该数据集成 API 代理对象。

3. 使用 MWSPProxy 对象调用 Login 操作。

Login 操作需要存储库名称、用户名和密码，并返回会话 ID。此操作调用将 MWSPProxy 对象与存储库名称和用户名对关联。所有使用 MWSPProxy 对象的后续批处理 Web 服务操作请求都使用这些存储库和用户名。

会话维护

Web 服务中心需要会话维护来缓存资源。SOAP 消息中的 SOAP 表头附带方便维护会话的会话信息。

要设置和执行会话维护，请完成以下步骤：

1. 从 Login 操作调用的响应中提取具有根元素名称 Context 和命名空间 `http://www.informatica.com/wsh` 的表头。此 SOAP 表头包含由 Web 服务中心发送的会话 ID。
2. 调用 Login 操作后，设置 MWSPProxy 对象中的 SOAP 表头。这会针对使用 MWSPProxy 对象的所有后续请求发送 SOAP 表头中的会话 ID。
3. 在 DIWSPProxy 对象中使用同一会话 ID 设置 SOAP 表头，这样对于随后执行的所有请求，都将使用 DIWSPProxy 对象发送相同的会话 ID。

操作调用

此时您可以使用 MWSPProxy 和 DIWSPProxy 对象调用元数据 Web 服务和数据集成 Web 服务操作。使用 MWSPProxy 对象调用元数据 Web 服务操作。使用 DIWSPProxy 对象调用数据集成 Web 服务操作。

资源清理

Web 服务中心针对性能和资源清理实施会话到期。Logout 操作释放客户端应用程序获得的 Web 服务中心资源并执行清理操作。

要释放资源，请使用 MWSPProxy 对象调用 Logout 操作。

如果登录到存储库但不调用 Logout 操作，Web 服务中心在会话到期后执行资源清理。

错误处理

SOAP 响应中的 SOAP 故障元素包含调用 Web 服务时出现的错误。

调用任何批处理 Web 服务操作时，客户端应用程序应当实施相应的错误处理方案以检索 SOAP 故障。该方案因工具包而异。

Web 服务工具包提供了用于获得故障元素的 faultcode 和 faultstring 字段的异常处理架构。但是，您可能需要使用 XML 解析器来解析详细信息元素字段，以获得错误代码和扩展详细信息。

代理对象

Login 操作调用为提供的存储库和用户名创建会话。从 Login 操作调用获得的会话 ID 标识此会话。会话 ID 对应于元数据代理对象。只要会话 ID 有效，元数据代理对象便有效。调用 Logout 操作后，会话 ID 及对应的元数据和数据集成代理对象均为无效。

批处理 Web 服务的 Java 客户端应用程序

本节着重介绍使用 Axis Web 服务工具包编写 Java 客户端应用程序的步骤。

注意: 以下各节中的示例代码段来自 Web 服务中心附带的批处理 Web 服务示例程序。您可以在以下目录中查看 PowerCenter Web 服务的示例程序。

`<PowerCenterInstallationDir>/server/samples/WebServices/samples/BatchWebServices/axis`

在 Axis 中生成客户端代理类

您可以使用 Axis Web 服务工具包生成 Java 客户端代理类。要使用该工具包，请将 axis.jar 文件包含在 CLASSPATH 环境变量中。

要生成 Java 客户端代理类，请完成以下步骤：

1. 通过 Web 服务中心控制台下载元数据 Web 服务和数据集成 Web 服务的 WSDL 文件。
验证 WSDL 的端点 URL 是否包含正确的 Web 服务中心主机名和端口号。如果端点 URL 不正确，请更新地址元素，可在 WSDL 中的 definitions\service\port 层次结构中获得该元素。

2. 使用以下命令生成客户端代理类：

```
java org.apache.axis.wsdl.WSDL2Java --NStoPkg  
http://www.informatica.com/wsh=ProxyClasses -W <WSDLFile>
```

-W 选项关闭对包装的文档文字服务的支持。

例如，对名为 Metadata.wsdl 和 DataIntegration.wsdl 的 WSDL 文件运行以下命令：

```
java org.apache.axis.wsdl.WSDL2Java --NStoPkg  
http://www.informatica.com/wsh=ProxyClasses -W Metadata.wsdl  
java org.apache.axis.wsdl.WSDL2Java --NStoPkg  
http://www.informatica.com/wsh=ProxyClasses -W DataIntegration.wsdl
```

这些命令将在 ProxyClasses 包中生成客户端代理类。这些命令生成以下代理类：

- **MetadataInterface.java**. 包含元数据 Web 服务的接口。
- **DataIntegrationInterface.java**. 包含数据集成 Web 服务的接口。

Axis 中的初始化

调用元数据 Web 服务和数据集成 Web 服务之前，客户端应用程序必须执行初始化步骤。

要执行初始化，请完成以下步骤：

1. 通过实例化服务定位器类创建 MetadataService 和 DataIntegrationService 对象：

```
MetadataService mdService = new MetadataServiceLocator();  
DataIntegrationService diService = new DataIntegrationServiceLocator();
```

2. 通过步骤 1 中创建的 MetadataService 对象获得 MetadataInterface 对象 (MWSPProxy)。

如果 Metadata.wsdl 中的元数据服务端点 URL 包含正确的 URL，则获得 MWSPProxy 对象：

```
MWSPProxy=mdService.getMetadata();
```

否则，获得 MWSPProxy 对象：

```
MWSPProxy=mdService.getMetadata(new java.net.URL(MWS_URL));
```

MWS_URL 是包含元数据 Web 服务端点 URL 的变量。

使用 MWSPProxy 对象调用元数据 Web 服务操作。

3. 通过步骤 1 中创建的数据集成服务 DataIntegrationService 对象获得 DataIntegrationInterface 对象 (DIWSPProxy)。

如果 DataIntegration.wsdl 中的服务端点 URL 包含正确的 URL，获得 DIWSProxy 对象：

```
DIWSProxy=diService.getDataIntegration();
```

否则，获得 DIWSProxy 对象：

```
DIWSProxy=diService.getDataIntegration(new java.net.URL(DIWS_URL));
```

DIWS_URL 是包含数据集成 Web 服务的端点 URL 的变量。

使用 DIWSProxy 对象调用数据集成 Web 服务操作。

4. 使用 MWSProxy 对象调用 Login 操作来为客户端应用程序用户帐户创建会话 ID。Login 操作需要包装在对象 LoginRequest 中的域、存储库、用户名和密码，并返回会话 ID。

```
LoginRequest loginReq = new LoginRequest();
loginReq.setRepositoryDomainName(REPO_DOMAIN_NAME);
loginReq.setRepositoryName(REPO_NAME);
loginReq.setUserName(USER_NAME);
loginReq.setPassword(PASSWORD);
String sessionID = MWSProxy.login(loginReq);
```

REPO_DOMAIN_NAME 是包含 PowerCenter 域名的字符串，REPO_NAME 是包含域中存储库的名称的字符串，USER_NAME 是包含存储库的有效用户名的字符串，PASSWORD 是包含登录到存储库的用户的密码字符串。

5. 将 MWSProxy 和 DIWSProxy 对象与会话 ID 中的存储库和用户名关联。随后执行的所有使用 MWSProxy 或 DIWSProxy 对象的批处理 Web 服务请求使用会话 ID 中的存储库和用户名。

```
((org.apache.axis.client.Stub)MWSProxy).setHeader(createSessionHeader(sessionID));
((org.apache.axis.client.Stub)DIWSProxy).setHeader(createSessionHeader(sessionID));
```

Axis 中的会话维护

Web 服务中心需要会话维护来缓存资源。SOAP 消息中的 SOAP 表头附带方便维护会话的会话信息。

要执行会话维护，请完成以下步骤：

1. 从使用 MWSProxy 对象的 Login 操作调用的响应提取具有根元素名称 Context 和命名空间 http://www.informatica.com/wsh 的 SOAP 表头。此 SOAP 表头包含由 Web 服务中心发送的会话 ID。

```
/** Create session ID in the Soap message header */
public static SOAPHeaderElement createSessionHeader(String sessID) throws SOAPException
{
    String WSSE_NS = "http://www.informatica.com/";
    String WSSE_PREFIX = "infa";
    Name hdrname = SOAPFactory.newInstance().createName("Context", WSSE_PREFIX, WSSE_NS);
    SOAPHeaderElement header = new SOAPHeaderElement(hdrname);
    SOAPElement token = header.addChildElement("SessionId");
    token.addTextNode(sessID);
    return header;
}
```

2. 对于所有后续请求，使用 MWSProxy 对象发送 SOAP 表头中的会话 ID。调用 Login 操作后，您仅需在 MWSProxy 对象中设置一次 SOAP 表头。

```
((org.apache.axis.client.Stub) MWSProxy).setHeader(createSessionHeader(sessionID));
```

3. 使用同一 SOAP 表头设置 DIWSProxy 对象中的 SOAP 表头。

```
((org.apache.axis.client.Stub) DIWSProxy).setHeader(createSessionHeader(sessionID));
```

在 Axis 中执行操作调用

此时您可以使用 MWSProxy 和 DIWSProxy 对象调用元数据 Web 服务和数据集成 Web 服务操作。

例如，您可以调用 `getAllDIServers` 操作以获得集成服务列表：

```
DIServerInfoArray servers = MWSPProxy.getAllDIServers(null);
if (servers.getDIServerInfo() != null) {
    for(int i=0; i < servers.getDIServerInfo().length ; i++) {
        System.out.println("("+(i+1)+") "+servers.getDIServerInfo(i).getName());
    }
}
```

您可以调用 `pingDIServer` 操作来检查集成服务的状态：

```
DIServiceInfo diInfo = new DIServiceInfo();
diInfo.setDomainName(DI_DOMAIN_NAME);
diInfo.setServiceName(SERVICE_NAME);
PingDIServerRequest pingReq = new PingDIServerRequest();
pingReq.setDIServiceInfo(diInfo);
pingReq.setTimeout(100);
EPingState eps = DIWSPProxy.pingDIServer(pingReq);
```

`DI_DOMAIN_NAME` 是包含了包含集成服务的域的名称的变量。`SERVICE_NAME` 是包含集成服务名称的变量。

Axis 中的清理

清理操作释放客户端应用程序获得的 Web 服务中心资源。要清理和释放资源，请使用 `MWSPProxy` 对象调用 `Logout` 操作：

```
MWSPProxy.logout(null);
```

Axis 中的错误处理

您可以在 Axis 中实施客户端应用程序错误处理，方法是将代码置于 `try` 块内并捕获 `FaultDetails` 对象。作为客户端代理的一部分生成 `FaultDetails` 类。

您可以在 `try` 块内使用以下代码捕获 `FaultDetails` 对象：

```
try {
    // Code for steps explained above.
}
catch(FaultDetails fault) {
    // Display fault code
    System.out.println("fault code : " + fault.getFaultCode());
    // Display fault string
    System.out.println("fault string : " + fault.getFaultString());
    // Display error code
    System.out.println("error code is : " + fault.getErrorCode());
    // Display extended details
    System.out.println("extended detail is : " + fault.getExtendedDetails());
}
```

批处理 Web 服务的 C# 客户端应用程序

本节着重介绍使用 .NET Web 服务工具包编写 C# 客户端应用程序的步骤。

注意：以下各节中的示例代码段来自批处理 Web 服务示例程序。可在以下目录中查看这些示例程序：

```
<PowerCenterInstallationDir>\server\samples\WebServices\samples\BatchWebServices\dotnet\csharp
```

在 .NET 中生成客户端代理类

您可以使用 Microsoft .NET Web 服务工具包创建 Web 服务中心的 C# 客户端代理类。

要生成 C# 客户端代理，请完成以下步骤：

1. 通过 Web 服务中心控制台下载元数据 Web 服务和数据集成 Web 服务的 WSDL 文件。验证 WSDL 的端点 URL 是否包含正确的 Web 服务中心主机名和端口号。如果端点 URL 不正确，请更新地址元素，可在 WSDL 中的 definitions\service\port 层次结构中获得该元素。
2. 使用以下命令生成客户端代理类：

```
wsdl <WSDLFile>
```

例如，对名为 Metadata.wsdl 和 DataIntegration.wsdl 的 WSDL 文件运行以下命令：

```
wsdl Metadata.wsdl  
wsdl DataIntegration.wsdl
```

这些命令生成以下代理类：

- **MetadataService.cs**. 包含元数据 Web 服务的接口。
- **DataIntegrationService.cs**. 包含数据集成 Web 服务的接口。

.NET 中的初始化

调用元数据 Web 服务和数据集成 Web 服务之前，客户端应用程序必须执行初始化步骤。

要执行初始化，请完成以下步骤：

1. 实例化 MetadaService 类对象 (MWSPProxy)：

```
MWSPProxy= new MetadaService();
```

如果 Metadata.wsdl 中的元数据服务端点 URL 不包含正确的 URL，可使用以下代码设置 URL：

```
MWSPProxy.Url = MWS_URL;
```

MWS_URL 是包含元数据 Web 服务端点 URL 的变量。

使用 MWSPProxy 对象调用元数据 Web 服务操作。

2. 实例化 DataIntegrationService 类对象 (DIWSPProxy)：

```
DIWSPProxy= new DataIntegrationService ();
```

如果 DataIntegration.wsdl 中的数据集成服务端点 URL 不包含正确的 URL，可使用以下代码设置 URL：

```
DIWSPProxy.Url = DIWS_URL;
```

DIWS_URL 是包含数据集成 Web 服务端点 URL 的字符串。

使用 DIWSPProxy 对象调用数据集成 Web 服务操作。

3. 使用 MWSPProxy 对象调用 Login 操作以创建客户端应用程序用户帐户的会话 ID。Login 操作需要包装在对象 LoginRequest 中的域、存储库、用户名和密码，并返回会话 ID。

```
LoginRequest loginReq = new LoginRequest();  
loginReq.RepositoryDomainName = REPO_DOMAIN_NAME;  
loginReq.RepositoryName = REPO_NAME;  
loginReq.UserName = USER_NAME;  
loginReq.Password = PASSWORD;  
String sessID = MWSPProxy.Login(loginReq);
```

REPO_DOMAIN_NAME 是包含 PowerCenter 域名的字符串, REPO_NAME 是包含域中存储库的名称的字符串, USER_NAME 是包含存储库的有效用户名的字符串, PASSWORD 是包含登录到存储库的用户的密码字符串。

4. 将 MWSPProxy 和 DIWSPProxy 对象与会话 ID 中的存储库和用户名关联。随后执行的所有使用 MWSPProxy 或 DIWSPProxy 对象的批处理 Web 服务请求使用会话 ID 中的存储库和用户名。

```
MWSPProxy.Context.SessionId = sessID;  
DIWSPProxy.Context.SessionId = sessID;
```

.NET 中的会话维护

Web 服务中心需要会话维护来缓存资源。SOAP 消息中的 SOAP 表头附带方便维护会话的会话信息。

您无需执行其他步骤。.NET 客户端代理类为您处理会话维护。

在 .NET 中执行操作调用

此时您可以使用 MWSPProxy 和 DIWSPProxy 对象调用元数据 Web 服务和数据集成 Web 服务操作。

例如, 您可以调用 getAllDIServers 操作以获得集成服务列表:

```
DIServerInfo[] servers = MWSPProxy.GetAllDIServers(null);  
if (servers != null) {  
    for(int i=0; i < servers.Length ; i++) {  
        Console.WriteLine("("+(i+1)+") "+servers[i].Name);  
    }  
}
```

您可以调用 pingDIServer 操作来检查集成服务的状态:

```
PingDIServerRequest pingReq = new PingDIServerRequest();  
pingReq.Timeout = (PING_TIME_OUT);  
DIServiceInfo diInfo1 = new DIServiceInfo();  
diInfo1.DomainName = DI_DOMAIN_NAME;  
diInfo1.ServiceName = DI_SERVICE_NAME1;  
pingReq.DIServiceInfo = diInfo1;  
EPingState pingResult = DIWSPProxy1.pingDIServer(pingReq);
```

DI_DOMAIN_NAME 是包含了包含集成服务的域的名称的变量。DI_SERVICE_NAME 是包含集成服务名称的变量。

.NET 中的错误处理

您可以在 .NET 中实施客户端应用程序错误处理, 方法是代码置于 try 块内并捕获 SOAP 异常对象。SOAP 异常类是 .NET 框架 SDK 的一部分。

您可以在 try 块内使用以下代码来捕获 SOAP 异常对象:

```
try {  
    //Code for steps explained above.  
}  
catch(SoapException fault) {  
    // Display fault code  
    Console.WriteLine( "fault code is : " + fault.Code);  
    // Display fault string  
    Console.WriteLine( "fault string is : " + fault.Message);  
    // Parsing detail element  
    XmlNode detail = fault.Detail;  
    XmlElement WSHFaultDetails = detail[ "WSHFaultDetails" , "http://www.informatica.com/PowerCenter" ];
```

```

        XmlElement ErrorCode= WSHFaultDetails [ "ErrorCode" ];
        XmlElement ExtendedDetails= WSHFaultDetails [ "ExtendedDetails" ];
// Display error code
        Console.WriteLine ( "error code is : " + ErrorCode.InnerText);
// Display extended details
        Console.WriteLine ( "extended detail is : " + ExtendedDetails.InnerText);
    }

```

实时 Web 服务的客户端应用程序

实时 Web 服务客户端应用程序涉及以下元素：

- Web 服务工作流
- 客户端代理类
- 初始化
- 操作调用
- 错误处理

Web 服务工作流

您可以构建实时 Web 服务客户端应用程序来运行 Web 服务工作流。创建客户端应用程序之前，请在 PowerCenter 中创建映射和工作流。启用工作流中的以下选项以允许客户端应用程序运行工作流：

- **Web 服务。**启用“Web 服务”选项将工作流转换为 Web 服务工作流。
- **可运行。**启用“可运行”选项以允许客户端应用程序运行 Web 服务工作流。
- **可见。**启用“可见”选项以使 Web 服务中心在 Web 服务中心控制台中发布 Web 服务的 WSDL。

生成客户端代理类

要使用在 PowerCenter 中创建的实时 Web 服务，您需要通过要访问的 Web 服务的 WSDL 生成客户端代理类。

要生成客户端代理，请完成以下步骤：

1. 为开发时所用的平台和语言选择 Web 服务工具包。
2. 通过 Web 服务中心控制台下载实时 Web 服务的 WSDL。
3. 使用 Web 服务工具包通过 WSDL 生成客户端代理类。有关生成代理类的详细信息，请参阅 Web 服务工具包文档。各种工具包以特定方式生成客户端代理类。

初始化

执行 Web 服务操作调用之前，客户端应用程序必须实例化客户端代理类中的 Web 服务对象，并且获得 Web 服务的端口。

操作调用

要调用 Web 服务操作，客户端应用程序必须创建请求对象并将其传递到端口操作。当 Web 服务发回响应时，客户端应用程序必须根据需要处理响应。

错误处理

实时 Web 服务客户端应用程序中的错误处理与批处理 Web 服务客户端应用程序中的相同。SOAP 响应中的 SOAP 故障元素包含调用 Web 服务时出现的错误。客户端应用程序应当实施相应的错误处理架构以检索 SOAP 故障。

实时 Web 服务的 Java 客户端应用程序

本节提供有关使用 Axis Web 服务工具包创建调用 PowerCenter 实时 Web 服务的 Java 客户端应用程序的说明。有关使用 Axis Web 服务工具包的详细信息，请参阅 Apache 网站上文档：

<http://ws.apache.org/axis/java/user-guide.html>

创建调用 PowerCenter Web 服务器工作流的客户端应用程序之前，您必须先创建该 Web 服务工作流并为 Web 服务生成 WSDL。然后，根据 Web 服务 WSDL 创建客户端应用程序。

要创建 PowerCenter Web 服务并生成 WSDL，请完成以下步骤：

1. 为 Web 服务工作流创建映射。您可以创建映射以通过 Web 服务客户端接收消息，转换数据，并将响应发回 Web 服务客户端或写入任何 PowerCenter 支持的目标。
2. 创建工作流并将其作为 Web 服务启用。创建运行映射的工作流，然后启用工作流属性中的“Web 服务”选项。选择“可运行”选项以使 PowerCenter 外的客户端应用程序能够运行工作流。
3. 找到并下载 Web 服务工作流的 WSDL。创建 Web 服务工作流时，PowerCenter 为 Web 服务生成 WSDL。如果将 Web 服务配置为可见，您可以在与 Web 服务关联的 Web 服务中心控制台上查看 WSDL。

创建 Web 服务后，您就可以开发运行 Web 服务工作流的客户端应用程序。

相关主题：

- [“创建和配置 Web 服务工作流” 页面上 80](#)
- [“Web 服务部分” 页面上 16](#)

创建实时 Web 服务的客户端应用程序

要创建调用实时 Web 服务的客户端应用程序，请完成以下步骤：

1. 生成 Web 服务的客户端代理类。
创建代理类后，创建 Java 应用程序以调用 Web 服务。在 Java 应用程序内执行后面的步骤。
2. 初始化 Web 服务对象。
3. 创建请求对象。
4. 将请求对象传递到端口操作并处理响应。

注意：以下各节中的示例代码段来自多行查找实时 Web 服务示例程序。可在以下目录中查看这些示例：

```
<PowerCenterInstallationDir>/server/samples/WebServices/samples/RealTimeWebServices/  
UnprotectedWebServices/axis/CustomLookup_MULTIPLEROW
```

步骤 1。在 Axis 中生成客户端代理类

您可以使用 Axis Web 服务工具包为 Web 服务 WSDL 生成 Java 客户端代理类。具体来说，您可以运行 WSDL2Java 工具生成 Java 代理类文件。

验证 WSDL 是否包含端点 URL 中的 Web 服务的正确主机名和端口号。如果端点 URL 不正确，请更新地址元素，可在 WSDL 中的 \definitions\service\port 层次结构中获得该元素。

使用以下命令生成客户端代理类：

```
java org.apache.axis.wsdl.WSDL2Java -W <WSDLFile>
```

例如，对名为 SampleWS.wsdl 的 WSDL 运行以下命令：

```
java org.apache.axis.wsdl.WSDL2Java -W SampleWS.wsdl
```

-W 选项关闭对包装的文档文字服务的支持。

WSDL2Java 为 WSDL 中定义的每种数据类型生成一个类。默认情况下，WSDL2Java 根据 WSDL 中的命名空间生成包名称。通常情况下，如果命名空间的格式为 *http://x.y.com* 或 *urn:x.y.com*，对应的包则是 *com.y.x*。

步骤 2。初始化 Web 服务对象

调用任何 Web 服务操作之前，您必须在客户端代理类中创建 Web 服务对象并获得 Web 服务的端口。

要创建 Web 服务对象，请实例化服务定位器类。在该示例程序中，以下代码实例化服务定位器：

```
CustomerLookup_MULTIPLEROW service = new CustomerLookup_MULTIPLEROWLocator();
```

要获得 Web 服务的端口，请使用针对该端口类型创建的代理类。在该示例程序中，以下代码获得 Web 服务的端口：

```
CustomerLookup_MULTIPLEROWPort port =  
    service.getCustomerLookup_MULTIPLEROWPort(new java.net.URL(END_POINT_URL);
```

变量 END_POINT_URL 包含 WSDL 的 URL。

步骤 3。创建请求对象

您必须创建请求对象和任何要传递到 Web 服务的所需参数。在客户端应用程序示例中，以下代码创建查找请求对象：

```
CustomerLookupRequest request = new CustomerLookupRequest();  
request.setCustomerID_in(CustomerID);
```

步骤 4。发送请求并处理响应

创建请求对象后，将其传递到端口操作。Web 服务发回响应。您可以按需处理响应。

在客户端应用程序示例中，以下代码将请求对象传递到端口并显示响应：

```
CustomerLookupResponse[] response = port.customerLookup_MULTIPLEROWOperation(requestOperation);  
System.out.println();  
if (response[0].getCustomerID_out() == 0)  
{  
    System.out.println("Customer(s) with the ID as " + CustomerID + " does not exist!!!");  
}  
else  
{  
    System.out.println("***** Customer(s) that matches with the Customer ID is/are ...");  
    for (int i = 0; i < response.length; i++)  
    {  
        System.out.println("***** Customer ID: " + response[i].getCustomerID_out());  
        System.out.println("***** Customer Name: " + response[i].getCustomerName_out());  
        System.out.println("***** Customer Age: " + response[i].getCustomerAge_out());  
        System.out.println("***** Customer Gender: " +
```

```

        response[i].getCustomerGender_out());
    System.out.println("***** Customer Address: " +
        response[i].getCustomerAddress_out());
    if (i < response.length - 1) System.out.println ();
}
}

```

使用参数数组

在 PowerCenter 中，参数表示可在会话间更改的值，如数据库连接、源文件或目标文件。您可以创建与工作流或会话关联的参数，以在每次运行工作流或会话时提供灵活性。

对于 Web 服务客户端应用程序，您可以在参数文件或参数数组中定义与工作流或会话关联的参数的值。要使用参数文件中的参数，请在客户端应用程序中指定参数文件名称。集成服务必须能够访问该参数文件。要使用参数数组，请在客户端应用程序中提供参数数组的元素的参数值。

例如，启动工作流或任务的请求可以使用参数文件名或参数数组中的参数和值的列表来指定与工作流或任务关联的参数。

参数数组定义

SOAP 请求中的参数定义由参数的范围、名称和值组成。当集成服务运行工作流或任务时，它使用数组中的参数的方法和使用参数文件中的参数的方法相同。

WSDL 包含以下参数数组元素的定义：

```

<complexType name="Parameter">
  <sequence>
    <element name="Scope" type="xsd:string" />
    <element name="Name" type="xsd:string" />
    <element name="Value" type="xsd:string" />
  </sequence>
</complexType>

<complexType name="ParameterArray">
  <sequence>
    <element maxOccurs="unbounded" minOccurs="0" name="Parameters"
      nillable="true" type="impl:Parameter" />
  </sequence>
</complexType>

```

例如，参数文件包含以下参数：

```

[s_m_A]
$a=1
$b=2
$c=3
[WSH_Folder.s_m_B]
$d=4

```

对于参数数组中包含相同参数的 StartWorkflow 操作的 Web 服务调用，其 SOAP 请求应包含以下元素：

```

<StartWorkflow>
...
  <Parameters>
    <Parameter>
      <Scope>s_m_A</Scope>
      <Name>$a</Name>
      <Value>1</Value>

```



```

    </Parameter>
    <Parameter>
      <Scope>s_m_A</Scope>
      <Name>$b</Name>
      <Value>2</Value>
    </Parameter>
    <Parameter>
      <Scope>s_m_A</Scope>
      <Name>$c</Name>
      <Value>3</Value>
    </Parameter>
    <Parameter>
      <Scope>WSH_Folder.s_m_B</Scope>
      <Name>$d</Name>
      <Value>4</Value>
    </Parameter>
    ...
  </StartWorkflow>

```

WorkflowRequest 和 TaskRequest 类型包含 ParameterArray 元素。您可以在参数数组中指定任意数量的参数。

以下示例代码出自在 Axis 中生成的 Web 服务客户端应用程序，显示了如何在 WorkflowRequest 中创建参数数组：

```

Parameter[] parameters = new Parameter[4];

Parameter param1 = new Parameter();
Param1.setScope( "s_m_A" );
Param1.setName( "$a" );
Param1.setValue( "1" );
Parameters[0] = param1;

Parameter param2 = new Parameter();
Param2.setScope( "s_m_A" );
Param2.setName( "$b" );
Param2.setValue( "2" );
Parameters[1] = param2;

Parameter param3 = new Parameter();
Param3.setScope( "s_m_A" );
Param3.setName( "$c" );
Param3.setValue( "3" );
Parameters[2] = param3;

Parameter param4 = new Parameter();
Param4.setScope( "WSH_Folder.s_m_B" );
Param4.setName( "$d" );
Param4.setValue( "4" );
Parameters[3] = param4;

WorkflowRequest wfReq = new WorkflowRequest();
wfReq.setParameters(parameters);

```

您可以在以下操作中使用参数数组：

- startWorkflow
- startWorkflowFromTask
- recoverWorkflow

- startTask

使用参数数组的规则和准则

在 Web 服务请求中使用参数数组时，请使用以下规则和准则：

- **使用一个参数文件和参数数组。**执行 Web 服务操作调用时，请勿在 SOAP 请求中同时指定参数文件名和参数数组。如果在 SOAP 请求中同时指定参数文件和参数数组，Web 服务中心会返回错误消息警告，指明请求指定了参数列表和参数文件。
- **如果定义了参数文件和参数数组，集成服务使用 SOAP 请求中的参数数组的参数值。**在以下情况下，集成服务使用在参数数组中定义的值：
 - 在启动工作流的 Web 服务请求中指定参数数组。
 - 工作流具有在工作流属性中定义的关联参数文件。

为客户端请求添加安全性

Web 服务中心使用以下类型的 Web 服务安全机制：

- **用户凭据。**要获得请求的凭据，客户端必须登录到包含要运行的 Web 服务的 PowerCenter 存储库。登录生成会话 ID，Web 服务客户端必须将其包含在 SOAP 请求中。
Web 服务中心将此安全选项用于批处理 Web 服务。调用任何其他操作之前，调用批处理 Web 服务的客户端应用程序必须登录到存储库。
- **用户名标志。**使用用户名标志且基于 OASIS Web 服务安全标准（包括一组确保内容完整和 SOAP 消息安全的 SOAP 扩展）的 Web 服务安全。
用户名标志是受保护 Web 服务的默认安全选项。默认情况下，由 Web 服务中心为受保护 Web 服务生成的 WSDL 包含具有 UsernameToken 元素的安全表头。
有关 OASIS Web 服务安全标准的详细信息，请参阅 OASIS 网站上的 Web 服务安全规范：

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ws-sx

注意：如果客户端应用程序发送运行 Web 服务的登录请求，它也应在收到响应后发送一个明确的注销请求。无对应注销请求的登录请求可能会导致存储库服务和 Web 服务中心进程出现内存泄漏。

SOAP 请求中的 UsernameToken

当您根据 Web 服务中心生成的 WSDL 生成客户端应用程序时，请求对象默认包含表头中的 UsernameToken 元素。

SOAP 请求中的 UsernameToken 元素可以具有以下密码安全选项之一：

- **纯文本密码。**包含纯文本密码。
- **哈希密码。**包含使用 MD5 或 SHA-1 哈希函数生成的哈希加密密码。
- **摘要式密码。**包含由使用临时值和时间戳的哈希运算生成的加密密码。

将用户密码包含在 UsernameToken 的 Password 元素中。

密码元素包含一个“类型”属性，用于指示所用的密码安全类型。如果省略类型属性，密码类型默认为 *PasswordText*。

注意：如果 Informatica 域使用 Kerberos 网络身份验证，则无法在 SOAP 请求中使用哈希密码或摘要式密码。

纯文本密码

UsernameToken 元素包含以下子元素：

- **Username 元素。**包含可在 PowerCenter 本地安全域或任何 LDAP 安全域中找到的用户名。默认安全域为本地安全域。如果用户名属于本地安全域，Username 不需要安全域的名称。如果用户名属于 LDAP 安全域，用户名之前必须有安全域名称和斜杠 (/)。

以下 Username 元素示例表格显示了用于指明用户帐户的安全域的格式：

Username 元素的值	安全域
<UsernameToken> <Username>Native/Administrator</Username> <Password>Administrator</Password> </UsernameToken>	本地
<UsernameToken> <Username>/Administrator</Username> <Password>Administrator</Password> </UsernameToken>	本地
<UsernameToken> <Username>Administrator</Username> <Password>Administrator</Password> </UsernameToken>	本地
<UsernameToken> <Username>LDAPAdm/Administrator</Username> <Password>Administrator</Password> </UsernameToken>	名为 LDAPAdm 的 LDAP 安全域。

- **Password 元素。**包含纯文本密码。可省略 Password 元素的 Type 属性，或者将其设置为 *PasswordText*。

哈希密码

UsernameToken 元素包含以下子元素：

- **Username 元素。**包含可在 PowerCenter 本地安全域中找到的用户名。
- **Password 元素。**包含哈希密码。必须使用 MD5 或 SHA-1 哈希函数对该密码进行哈希处理，并将其转换为 Base64 编码。可省略 Password 元素的类型属性，或者将其设置为 “PasswordText”。

以下代码显示了使用哈希密码的请求的安全表头示例：

```
<soapenv:Header>  
  <!-- UsernameTokens -->  
  <inf:Security>  
    <UsernameToken>  
      <Username>Native/Administrator</Username>  
      <Password>Ntm58Cxf7SB0QAz30lsTq1nv-D7</Password>  
    </UsernameToken>  
  </inf:Security>  
</soapenv:Header>
```

使用第三方工具创建哈希密码

您可以使用第三方工具（如 OpenSSL 和 Java MessageDigest 类）创建哈希密码。

在 UNIX 上使用 OpenSSL

要在 UNIX 计算机上使用 OpenSSL 创建哈希密码，请使用消息摘要命令 `dgst` 运行 OpenSSL。

以下示例显示了如何使用 MD5 哈希函数创建密码字符串 Administrator 的哈希密码并以 Base64 对其编码：

```
echo -n "Administrator" | openssl dgst -md5 -binary | openssl base64
```

以下示例显示了如何使用 SHA-1 哈希函数创建密码字符串 Administrator 的哈希密码并以 Base64 对其编码：

```
echo -n "Administrator" | openssl dgst -sha1 -binary | openssl base64
```

注意：在这些例子中，`echo` 命令将换行符添加到字符串。命令中的 `-n` 选项可删除换行符。

在 Windows 上使用 OpenSSL

要在 Windows 计算机上使用 OpenSSL 创建哈希密码，请使用消息摘要命令 `dgst` 运行 OpenSSL。

以下示例显示了如何使用 SHA-1 哈希函数创建哈希密码及使用 Base64 对其编码。

```
openssl dgst -sha1 -binary -out <output file name> <input file name>
openssl enc -base64 -in <output file name>
```

输入文件包含要对其进行哈希处理的密码字符串。OpenSSL 将哈希密码写入输出文件。

Java MessageDigest

以下示例显示了如何使用 Java MessageDigest 类以 MD5 哈希函数创建哈希密码并以 Base64 对其编码：

```
public static String md5hash(String password) throws Exception{
    MessageDigest digest = java.security.MessageDigest.getInstance("MD5");
    digest.reset();
    digest.update(password.getBytes());
    byte[] hash = digest.digest();
    return new String(org.apache.commons.codec.binary.Base64.encodeBase64(hash));
}
```

以下示例显示了如何使用 Java MessageDigest 类以 SHA-1 哈希函数创建哈希密码并以 Base64 对其编码：

```
public static String sha1hash(String password) throws Exception{
    MessageDigest digest = java.security.MessageDigest.getInstance("SHA-1");
    digest.reset();
    digest.update(password.getBytes());
    byte[] hash = digest.digest();
    return new String(org.apache.commons.codec.binary.Base64.encodeBase64(hash));
}
```

摘要式密码

UsernameToken 元素包含以下子元素：

- **Username 元素。**包含可在 PowerCenter 本地安全域中找到的用户名。

- **Password 元素。**包含摘要式密码。该密码是对由 Nonce 元素的临时值和 Created 元素的时间戳连接而成的密码进行哈希处理后生成的值。密码必须使用 SHA-1 哈希函数进行哈希处理，并进行 Base64 编码。

要使用摘要式密码安全选项，必须将 Password 元素的类型属性设置为 *PasswordDigest*。

- **Nonce 元素。**包含临时值，该值是仅能使用一次的随机值。
- **Created 元素。**包含指明请求的创建时间的时间戳值。时间戳使用 UTC 格式 yyyy-MM-dd'T'HH:mm:ss.SSS'Z'。For example:2008-08-11T18:06:32.425Z。

包含在 SOAP 请求中的临时值只能使用一次。默认情况下，创建时间（由 Created 元素中的值指定）之后的 300 秒（五分钟）内请求是有效的。客户端应用程序必须在临时值的有效时间内发送请求。例如，已创建值指明在上午 10:00 创建请求。该请求在上午 10:00 和上午 10:05 间有效。如果客户端应用程序在上午 10:00 前或上午 10:05 后将请求发送到 Web 服务中心，则请求和临时值无效且请求失败。

摘要式密码使用标准 OASIS 密码摘要算法：

Password_Digest = Base64 (SHA-1 (nonce + created + password))

您可以使用任何工具生成临时值、时间戳和摘要式密码。

以下代码显示了使用摘要式密码的请求的安全表头示例：

```
<soapenv:Header>
  <!-- UsernameTokens -->
  <inf:Security>
    <UsernameToken>
      <Username>Administrator</Username>
      <Password Type="PasswordDigest"> Xty5lCAf5SV00AY30tsYq7nv/DI=</Password>
      <Nonce>KjsaeiuDFKJEwkr4332rL=</Nonce>
      <Created>2008-08-12T01:11:47.013Z</Created>
    </UsernameToken>
  </inf:Security>
</soapenv:Header>
```

使用第三方工具创建摘要式密码

您可以使用第三方工具（如 Java MessageDigest 类）创建摘要式密码。

以下示例显示了如何使用 Java MessageDigest 类使用时间戳和临时值创建摘要式密码并以 Base64 对其编码：

```
public static String oasisDigest(String password, String nonce) throws Exception{
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss.SSS'Z'");
    String created = sdf.format(new Date());
    System.out.println("Created : " + created);
    System.out.println("Nonce : " + new
String(org.apache.commons.codec.binary.Base64.encodeBase64(nonce.getBytes())));
    String toDigest = nonce + created + password;
    MessageDigest digest = java.security.MessageDigest.getInstance("SHA-1");
    digest.reset();
    digest.update(toDigest.getBytes());
    byte[] hash = digest.digest();
    return new String(org.apache.commons.codec.binary.Base64.encodeBase64(hash));
}
```

如果使用 Web 服务测试工具（如 soapUI）测试客户端应用程序，您可以使用该工具生成客户端请求的摘要式密码。

使用 Web 服务源和目标

Web 服务源和目标概览

创建要在 Web 服务工作流程中使用的映射后，该映射的源和目标必须定义 Web 服务输入和输出消息。Web 服务源定义 Web 服务操作的输入的消息，并代表 Web 服务 SOAP 请求的元数据。Web 服务目标定义 Web 服务操作的输出消息，并代表 Web 服务 SOAP 响应的元数据。

Web 服务说明语言 (WSDL) 介绍了 Web 服务的输入和输出消息。如果要创建的 Web 服务工作流程具有 WSDL，您可以从 WSDL 导入源和目标定义。如果没有 WSDL，您可以根据关系或平面文件源或目标中的列创建输入和输出消息。或者，可以从您定义的列创建输入和输出消息。

使用 Designer 可创建 Web 服务源和目标定义。您可以通过以下方式创建 Web 服务源和目标定义：

- **从 WSDL 导入定义。** 导入在 WSDL 中定义的操作的源或目标定义。当您导入源定义时，Designer 将导入输入消息的定义。当您导入目标定义时，Designer 将导入输出和错误消息的定义。
- **从关系或平面文件源和目标创建定义。** 您可以根据文件夹中定义的关系或平面文件源和目标定义创建 Web 服务源和目标定义。也可以手动定义列并指定数据类型和列大小。从关系或平面文件源和目标创建 Web 服务源和目标定义或手动定义列时，无需 WSDL。

了解 Web 服务源和目标

与 XML 源和目标类似，Web 服务源和目标定义被组织为 XML 视图。XML 视图是表示在输入和输出消息定义的元素和属性的列组。

当您导入源和目标定义时，Designer 可根据 WSDL 中定义的输入或输出消息中的元素生成 XML 视图。它还生成输入或输出消息的 mime 附件的视图。

当您创建 Web 服务源和目标定义时，Designer 将根据关系或平面文件源或目标中定义的列或您手动定义的列创建 XML 视图。

XML 视图和组

Web 服务源和目标定义可以包含以下视图：

- **信封。** 与 SOAP 信封和正文元素对应的 XML 视图。信封视图是主视图，其中包含主键和输入或输出消息的端口。

如果正文消息部分很简单，Designer 将生成信封视图。

如果正文消息部分很复杂，Designer 将生成其他正文视图：

- **元素。** 输入或输出消息中包含多次出现的元素时创建的视图。Designer 可为输入或输出消息中每个多次出现的元素生成元素视图。元素视图与信封视图具有 $n:1$ 关系。
- **类型。** 输入或输出消息中包含复杂类型的定义时创建的视图。Designer 可为输入或输出消息中的每个复杂类型元素生成类型视图。类型视图与信封视图具有 $n:1$ 关系。

Designer 可为在实体关系模式下导入的 Web 服务源和目标定义生成类型视图。

- **表头。** 与 SOAP 表头元素对应的 XML 视图。如果表头消息部分很复杂，Designer 可以将表头视图拆分成单独的元素和类型视图。

- **错误。**为操作的输出消息定义错误消息时创建的视图。Designer 可为每个为操作定义的错误消息生成错误视图。错误视图与信封视图具有 $n:1$ 关系。只有 Web 服务目标定义包含错误视图。

Designer 可为在规范化层次结构关系模式或实体关系模式下导入的 Web 服务目标定义生成错误视图。

- **附件。**为包含 mime 附件的 WSDL 生成的附件视图。附件视图与信封视图具有 $n:1$ 关系。

Designer 可根据包含 mime 附件的元素定义的 WSDL 为 Web 服务源或目标定义生成附件视图。

相关主题：

- [“WSDL 附件” 页面上 79](#)

源定义

Designer 可根据输入消息的定义为 Web 服务源定义生成 XML 视图。

下表显示了 Designer 可以为 Web 服务源定义生成的 XML 视图：

导入模式	信封	元素	类型	错误	附件
实体关系	是	是	是	否	是
规范化层次结构关系	是	是	否	否	是

目标定义

为 WSDL 中的操作创建目标定义时，Designer 将导入输出消息和任何与该操作关联的错误消息。如果操作中的函数导致不同的错误，Designer 将在目标定义中创建多个错误视图。错误消息表示处理请求时出错。

下表显示了 Designer 可以为 Web 服务目标定义生成的 XML 视图：

导入模式	信封	元素	类型	错误	附件
实体关系	是	是	是	是	是
规范化层次结构关系	是	是	否	是	是

注意：要为错误消息创建单独的目标定义，请在 PowerCenter Designer 中配置 Web 服务选项。

导入或创建 Web 服务源和目标的规则和准则

导入或创建 Web 服务源和目标时，请使用以下规则和准则：

- **对于具有复杂关系的元素使用 WSDL。**要创建具有复杂元素关系的 Web 服务源或目标，请首先创建 WSDL 以定义元素层次结构，然后从该 WSDL 导入源或目标。使用 WSDL 创建包含多次出现的元素或复杂类型的元素的 Web 服务源或目标。
- **手动定义简单的 Web 服务源或目标定义。**要创建包含简单列集而不包含嵌套元素的 Web 服务源或目标，请手动创建定义或使用关系或平面文件源或目标定义。您可以指定 Web 服务源或目标定义中的所有列均出现多次。
- **在同一进程中创建源和目标定义。**在创建源定义的同时，从关系或平面文件源或目标创建 Web 服务目标定义。要同时创建 Web 服务映射的源和目标，请验证“创建源”和“创建目标”选项在“创建 Web 服务

定义”窗口中是否处于选中状态。例如，在 Source Analyzer 中，选择“源”>“Web Service Provider”>“创建 Web 服务定义”。在“创建 Web 服务定义”窗口中，选择“创建目标”选项。

- **使用相同的方法为请求-响应映射创建源和目标定义。**如果创建了请求-响应 Web 服务映射，请使用相同的方法创建源和目标定义。例如，如果从 WSDL 导入了源定义，请从 WSDL 中的相同操作导入目标定义。如果通过定义列或使用关系或平面文件源和目标来创建源定义，请使用相同的方法创建目标定义。
- **使用 WSDL 创建包含错误视图的目标。**如果您希望目标定义中具有特定数据错误的错误视图，请使用 WSDL 创建 Web 服务目标定义。如果根据平面文件或关系源或目标创建目标定义，则不能在目标定义中定义错误视图。如果根据平面文件或关系源或目标定义 Web 服务目标定义，则 Web 服务中心可以仅生成系统错误的错误消息。
- **WSDL 中的输入和输出消息必须具有相同的编码样式。**如果从 WSDL 导入 Web 服务源和目标，则输入和输出消息的编码样式必须相同。如果输入消息使用 RPC/SOAP 编码样式，则输出消息也必须使用 RPC/SOAP 编码样式。如果输入消息使用文档/文字编码样式，则输出消息也必须使用文档/文字编码样式。

如果您手动或根据关系或平面文件源或目标创建 Web 服务源或目标定义，Designer 会将文档/文本编码样式用于输入和输出消息。

- **WSDL 中的元素不能引用标准 W3C XML 架构。**不能从包含引用标准 W3C XML 架构的元素的 WSDL 导入 Web 服务源和目标定义。
- **在实体关系模式下导入包含空 complexType 元素的 WSDL。**如果 WSDL 包含的 complexType 元素不包含 Web 服务请求的输入消息中的值，则可以在实体关系模式下从 WSDL 导入源和目标定义。如果在规范化层次结构模式下从 WSDL 导入了源和目标定义，则当您发送包含空 complexType 元素的请求时，Web 服务将生成错误响应。
- **从包含有效 XML 语法的 WSDL 导入源和目标。**如果从无效的 WSDL 导入，Designer 将无法在 Web 服务向导中正确显示 WSDL 定义。在某些情况下，Designer 不会生成错误消息，但会部分解析 WSDL 并仅显示成功解析的服务和操作。如果您从 WSDL 导入且 Web 服务向导不显示正确的 WSDL 定义，请作为 XML 文件打开 WSDL 并验证语法是否正确。
- **使用正确的语法定义二维数组。**如果要将 WSDL 中的 complexType 元素定义为二维字符串数组，请使用以下语法：

```
wsdl:arrayType="xsd:string[][]"
```

不能从包含使用不同的语法定义的二维数组的 WSDL 导入 Web 服务源和目标定义。

- **不能从生成大量 XML 视图的 WSDL 导入 Web 服务源和目标。**可以从 WSDL 文件生成的 XML 视图的数量限制为 400。要创建包含超过 400 个 XML 视图的 Web 服务源或目标，请在 WSDL 工作区中手动创建组。

相关主题：

- [“SOAP 故障处理” 页面上 13](#)

导入 Web 服务源或目标定义

如果有定义要创建的 Web 服务的输入和输出消息的 WSDL，则可以从该 WSDL 导入源和目标定义。可以使用 Web 服务向导从 WSDL 导入 Web 服务源或目标定义。可以在同一进程中导入源和目标定义。

导入模式

您可以根据导入 WSDL 的模式为 Web 服务源或目标定义创建 XML 视图：

- **实体关系。**此模式是从 WSDL 导入的源或目标定义的默认导入模式。使用此导入模式可在视图之间创建关系，而非创建一个大型层次结构。当您创建具有实体关系的 Web 服务源或目标时，Designer 将为多次出现的元素和复杂类型生成单独的视图。Designer 包括所有派生复杂类型的视图。
- **规范化层次结构关系。**在规范化层次结构视图中，每个元素或属性都会出现一次。一对多关系成为单独的 XML 视图，其中包含用于关联视图的键。
- **不生成 XML 视图。**使用此导入模式可创建源或目标定义，而无需定义 XML 视图。您可以使用 WSDL 工作区添加 XML 视图和端口。

相关主题：

- [“从 WSDL 导入而不创建 XML 视图” 页面上 67](#)

消息 ID

要在暂存映射中使用 Web 服务源和目标定义，必须在 Web 源和目标定义中包含消息 ID。Web 服务中心使用消息 ID 作为主键，以绑定 Web 服务的请求和响应。例如，第一个会话从 Web 服务源读取并写入关系目标。第二个会话将关系目标用作源并写入 Web 服务目标。Web 服务中心使用消息 ID 将第一个会话中请求的输入消息与第二个会话的响应中的输出消息相关联。

当您在网络上运行 Web 服务会话时，还必须在 Web 源和目标定义中包括消息 ID。在网络上运行 Web 服务会话时，集成服务会将会话线程分发到网格中节点上的多个 DTM 进程。集成服务将使用该消息 ID 将分布于各个节点的 Web 服务输入和输出消息相关联。

如果在创建 Web 服务源或目标定义时添加消息端口，Designer 会将消息 ID 和客户端端口添加到信封视图中。

下表介绍了添加到信封视图的消息 ID 和客户端端口：

端口名称	说明
MessageID	Web 服务中心收到请求后，将生成消息 ID。它将使用此 ID 关联传入请求与传出响应。
ClientIP	Web 服务客户端的 TCP/IP 地址。

高级选项

从 WSDL 导入 Web 服务源或目标时，可以指定未定义长度的字段的长度和 XML 列的命名约定。

下表显示了从 WSDL 导入 Web 服务源或目标时可以设置的高级选项：

选项	说明
替代所有无限长度	您可以指定未定义长度的字段的默认长度，如字符串。默认情况下，此选项被选中。
为 XML 列生成名称	<p>您可以选择使用数字序列或架构中的元素或属性名称命名 XML 列。如果使用名称，请从以下选项中进行选择：</p> <ul style="list-style-type: none">- 当 XML 列引用某个属性时，将元素名称作为其前缀。对于 XML 列的名称，PowerCenter 使用以下格式： <i>NameOfElement_NameOfAttribute</i>- 为每个 XML 列添加 XML 视图名称作为其前缀。对于 XML 列的名称，PowerCenter 使用以下格式： <i>NameOfView_NameOfElement</i>- 为每个外键列添加 XML 视图名称作为其前缀。对于所生成外键列的名称，PowerCenter 使用以下格式： <i>FK_NameOfView_NameOfParentView_NameOfPKColumn</i> <p>列名称的最大长度为 80 个字符。PowerCenter 将截断长度超过 80 个字符的列名称。如果列名称不唯一，PowerCenter 将添加数字后缀以使名称保持唯一。</p>
映射为字符串的 anyType 元素的默认长度	<p>为 anyType 类型的元素创建的字符串端口的默认长度。您可以为 anyType 类型的元素创建字符串类型的端口。默认情况下，字符串的长度是您在此处设置的值。</p> <p>要更改字符串长度，请在 WSDL 工作区中编辑 Web 服务源或目标定义。默认值为 10,000。</p>

从 WSDL 导入而不创建 XML 视图

如果您从 WSDL 导入源或目标定义，并希望手动定义 XML 视图和端口，则可以创建空源或目标定义。

例如，您有一个在输入消息中定义十个元素的 WSDL，但您希望在源定义中仅包括其中两个元素，则可以创建空源定义并手动定义两个元素。目标定义不受影响。您可以导入目标定义并创建 XML 视图。

要从 WSDL 导入源或目标定义而无需创建 XML 视图，请在导入进程的步骤 2 中选择“不生成 XML 视图”。创建空源或目标定义后，请使用 WSDL 工作区定义 XML 视图和端口，以及视图之间的关系。右键单击源或目标定义的标题并选择“WSDL 工作区”。

相关主题：

- [“在 WSDL 工作区中编辑定义” 页面上 72](#)

从 WSDL 导入 Web 服务源或目标定义

按照相同的步骤从 WSDL 导入 Web 服务源或目标定义。由于源和目标定义代表 WSDL 中的不同元素，因此 Designer 所创建的源定义与目标定义不同。

您可以从可本地访问或通过 URL 访问的 WSDL 导入 Web 服务源或目标。您可以从某个具有 RPC/SOAP 编码或文档/文本编码样式的 WSDL 导入定义。

注意：当 Designer 导入 Web 服务目标定义时，它将根据操作和目标类型（如输出或目标）命名该定义。如果在导入后重命名该定义，则可以在“元数据扩展”选项卡上验证目标类型。

要从 WSDL 导入 Web 服务源或目标定义，请执行以下操作：

1. 要导入源定义，在 Source Analyzer 中，单击“源” > “Web Service Provider” > “从 WSDL 导入”。要导入目标定义，在 Target Designer 中，单击“目标” > “Web Service Provider” > “从 WSDL 导入”。

2. 单击“高级选项”。
此时将显示“XML 视图创建和命名选项”窗口。
3. 指定未定义长度的字段的默认长度并选择如何生成 XML 列的名称。
4. 选择从本地文件或 URL 导入。
如果从 URL 导入，请键入 URL 或从地址列表中选择 URL 并单击“打开”。
如果从本地文件导入，请在本地文件夹中选择 WSDL 文件，然后单击“打开”。
5. 选择在要为其创建源或目标定义的 WSDL 中定义的操作。
注意：如果从包含错误的 WSDL 导入，“Web 服务向导（步骤 1）”窗口将无法正确显示 WSDL 中定义的服务、绑定、端口或操作的列表。该窗口将显示空 WSDL 定义树或部分 WSDL 定义树。例如，如果 WSDL 在类型定义中包含错误，该窗口将显示空 WSDL 定义树。
6. 单击“下一步”。
此时将显示“Web 服务定义创建选项”对话框。
7. 选择导入模式。
导入模式确定要生成的 XML 视图的类型。可以将 XML 视图生成为实体关系或层次结构关系。默认导入模式为实体关系。
8. 要创建未定义任何视图或端口的源或目标定义，请选择“不生成 XML 视图”。
如果不生成 XML 视图，则 Designer 将创建空源或目标定义。源或目标定义不包含任何视图或端口。必须使用 WSDL 工作区为源或目标定义手动添加视图和端口。
9. 选择是否将消息和表头端口添加到源或目标定义。
有关详细信息，请参阅 [“消息 ID” 页面上 66](#)。
10. 要在同一导入进程中生成源和目标定义，请同时选择“创建源”和“创建目标”选项。
Designer 将根据选择的选项创建源和目标定义。
11. 单击“完成”。
此时工作区中将显示 Web 服务源或目标定义。

创建一个源或目标定义

如果没有要从中导入 Web 服务源或目标定义的 WSDL，您可以从关系或平面文件源或目标创建定义。也可以手动定义源或目标的端口，并指定数据类型和出现次数。

当您从其他源和目标或从手动定义的列创建 Web 服务源和目标定义时，Designer 将在实体关系模式下创建视图。不能在层次结构关系模式下从其他源或目标创建 Web 服务源或目标。

可以使用从以下关系数据库创建的关系源和目标创建 Web 服务源和目标定义：

- Oracle
- DB2
- Informix
- Teradata
- Microsoft SQL Server
- Sybase

从关系或平面文件源或目标创建 Web 服务源或目标定义时，Designer 将列出文件夹中提供的源和目标，包括源和目标的快捷方式。

在同一进程中创建源和目标定义。根据关系或平面文件源或目标创建 Web 服务源或目标后，可以在 Designer 工作区中编辑源或目标定义的列。

从 WSDL 导入 Web 服务源或目标定义时，可以选择以下选项：

- 多次出现的元素
- 消息端口

相关主题：

- [“在 Designer 工作区中编辑定义” 页面上 71](#)

多次出现的元素

从关系或平面文件源或目标创建 Web 服务源和目标定义时，必须指示源或目标定义中的列是否出现多次。选择“多次出现的元素”选项指示作为组的列出现多次。列表中的列表示一个数组。

编辑使用此选项创建的 Web 服务源或目标时，不能更改源或目标定义列的多次出现的属性。

消息端口

您可以在 Web 源和目标定义中包括消息 ID。Web 服务中心使用消息 ID 作为主键，以绑定 Web 服务的请求和响应。

注意：当您在 Designer 工作区中编辑 Web 服务源或目标定义时，消息和客户端端口不会出现在“Web 服务定义”选项卡中。不能修改消息和客户端端口。

相关主题：

- [“消息 ID” 页面上 66](#)
- [“Web 服务定义选项卡” 页面上 72](#)

通过关系或平面文件源或目标创建源或目标

使用此过程通过以下方法之一创建 Web 服务源或目标：

- 从平面文件源或目标
- 从关系源或目标
- 通过手动添加列并指定其名称、数据类型和精度

要从关系或平面文件源或目标创建 Web 服务源或目标，请执行以下操作：

1. 要创建 Web 服务源，在 Source Analyzer 中，单击“源” > “Web Services Provider” > “创建 Web 服务定义”。要创建 Web 服务目标，在 Target Designer 中，单击“目标” > “Web Services Provider” > “创建 Web 服务定义”。
2. 输入计划使用其源和目标定义的 Web 服务映射的名称。

Designer 将该 Web 服务定义名称用作源和目标定义的名称。它会将后缀 *_input* 添加到源定义名称或将 *_output* 添加到目标定义名称。

如果您知道要在 Web 服务源或目标定义中包括的列，则可以直接将其添加到源或目标定义。如果文件夹中存在关系或平面文件源或目标，则可以根据关系或平面文件列创建源或目标定义。

3. 要向列表中添加列，请单击“添加”按钮并指定列名称、数据类型和精度。
当您创建包含多次出现的元素的 Web 服务源或创建 Web 服务目标时，Web 服务中心将忽略“非空”属性。
 4. 要根据关系或平面文件源或目标定义中的列创建 Web 服务源或目标，请单击“从源/目标导入”按钮并选择源或目标定义。
Designer 列出了在选定的源或目标中找到的列。
 5. 单击“确定”。
 6. 如果必要，编辑列名称和字符串列的精度。
可以将列添加到定义并定义其数据类型和属性。可以删除不希望使用的列。
可以指定在源或目标定义中定义的列出现一次还是多次。
 7. 如果列出现多次，请选择“多次出现的元素”。
此选项指示作为组的列出现多次。选中此选项后，Designer 将生成包含所有列的元素视图。
 8. 如果您正在创建源定义并且还需要创建目标定义，请单击“创建目标”并重复步骤 3 至 7，以将端口添加到目标定义。
如果您正在创建目标定义并且还需要创建源定义，请单击“创建源”并重复步骤 3 至 7，以将端口添加到源定义。
 9. 要将消息和客户端端口添加到源或目标定义，请单击“添加消息端口”。
Designer 会将消息和客户端端口添加到源或目标定义的信封视图。如果要在同一进程中创建源和目标定义，Designer 会将消息和客户端端口添加到源和目标定义的信封视图。
 10. 单击“确定”。
Designer 将创建 Web 服务源或目标定义。
- 查看源和目标定义中的 XML 视图，以验证视图和端口是否与 Web 服务映射要求匹配。要添加、删除或修改列，请在 Designer 工作区中编辑源或目标定义。

相关主题：

- [“多次出现的元素” 页面上 69](#)
- [“消息 ID” 页面上 66](#)
- [“在 Designer 工作区中编辑定义” 页面上 71](#)

编辑 Web 服务源和目标

编辑 Web 服务源和目标概览

您可以根据创建方式编辑 Web 服务源或目标定义：

- **从 WSDL 导入的源或目标定义。**如果您从 WSDL 导入 Web 服务源或目标定义，则可以在 WSDL 工作区中编辑源或目标定义。您可以使用 WSDL 工作区在源或目标定义中添加、修改或删除视图。
您可以在 Designer 工作区中查看定义并编辑某些属性。

- **通过关系或平面文件源和目标创建的源或目标定义。**如果您根据关系或平面文件目标创建一个 Web 服务源或目标定义，您可以在 Designer 工作区编辑这些列。

您可以在 WSDL 工作区中查看源或目标定义。您无法在 WSDL 工作区中编辑源或目标定义。

注意: 您可以编辑在 PowerCenter 9.0.1 或更高版本中创建的 Web 服务源和目标。为了在您运行 Web 服务工作流时避免发生错误，请不要编辑从以前版本升级的 Web 服务源和目标。要从以前版本的 PowerCenter 更新 Web 服务源或目标，请在 PowerCenter 9.0.1 或更高版本中重新创建源或目标。

在 Designer 工作区中编辑定义

在 Designer 工作区中，您可以随时为 Web 服务源或目标定义添加描述或指定业务文档的链接。如果您根据关系或平面文件目标手动创建一个源或目标定义，您可以在源或目标定义中修改列的列表。当您更改这些列时，所做的更改会立即反映在 XML 视图中。

要查看或编辑源或目标定义的属性，请在 Source Analyzer 双击源定义或在 Target Designer 中双击目标定义。或者您可以右键单击源或目标定义的标题并选择“编辑”。

您可以在以下选项卡中查看或编辑 Web 服务源和目标定义：

- **表。**在“表”选项卡上可以提供所有者名称和描述，并可以更改定义的名称。您不能更改表类型。
- **列。**在“列”选项卡上，您可以编辑字符串数据类型的精度。您还可以添加业务名称和列说明。
- **属性。**在“属性”选项卡上，您可以查看源或目标的定义中每列的属性值。
- **元数据扩展。**在“元数据扩展”选项卡上，您可以查看“Web 服务域”元数据扩展。您还可以在“用户定义的元数据域”中添加元数据扩展。
- **Web 服务定义。**当您编辑通过关系或平面文件目标创建的源或目标定义时，将显示此选项卡。您可以在源或目标的定义中添加、编辑或删除列。您所做的更改立即显示在“列”选项卡上。

表选项卡

“表”选项卡显示源或目标定义的表信息。您可以更改源或目标定义的名称。可以修改源或目标定义的所有者和描述。

列选项卡

“列”选项卡显示 Web 服务源或目标定义中的 XML 视图。您可以编辑字符串和二进制数据类型的精度，并可以添加业务名称和列说明。

字符串数据类型的默认精度是在 WSDL 导入进程中设置的无限长度数据的精度值。当您通过 WSDL 导入源或目标的定义时，您可以设置字符串数据类型的精度。当您编辑定义时，可以设置单个列的精度。

注意: Mapping Designer 使用总列长度大于 500 MB 的源和目标 Web 服务定义的映射无效。

属性选项卡

“属性”选项卡是只读选项卡，显示 Web 服务的源或目标定义中的每个字段的 XPath 和 XMLDataType 值。如果定义具有附件组，“属性”选项卡显示数据字段中的 MIME 类型。

元数据扩展选项卡

在“元数据扩展”选项卡上，您可以创建元数据扩展。您还可以在“Web Services Provider 域”中查看供应商定义的扩展。这些元数据扩展标识消息类型，包括输入、输出或错误。

有关元数据扩展的详细信息，请参阅《PowerCenter 存储库指南》。

Web 服务定义选项卡

对于手动定义的或基于关系或平面文件源或目标的 Web 服务源或目标定义，将显示“Web 服务定义”选项卡。您可以添加或删除源或目标定义中的列。您可以更改列的名称和数据类型，并修改特定数据类型的精度和小数位数。您还可以指定列出现一次还是多次。

当您在“Web 服务定义”选项卡中对列进行更改时，所做的更改将反映在“列”选项卡中。

在 WSDL 工作区中编辑定义

如果您从 WSDL 导入源或目标定义，并创建 XML 视图，则可以编辑 WSDL 工作区中的 XML 视图、端口以及关系。如果您从 WSDL 导入源或目标定义，但不生成 XML 视图，您可以使用 WSDL 工作区创建视图、修改组件、添加列并维护工作区中的视图关系。当您更新源或目标定义时，Designer 将更改传播到包括该源或目标的任何映射。

要在 WSDL 工作区中查看或编辑源或目标定义，请在 Source Analyzer 中右键单击源定义标题或在 Target Designer 中右键单击目标定义标题。然后选择 WSDL 工作区。

WSDL 工作区相当于 XML 编辑器。WSDL 工作区的使用方式与 XML 编辑器使用方式相同。然而，WSDL 工作区对特定于 Web 服务源和目标定义视图的更改执行验证。此外，您不能在 WSDL 工作区中执行 XML 工作区中可以执行的一些任务。

您不能在 WSDL 工作区中执行以下任务：

- 当您在 Web 服务源或目标定义中添加或编辑 XML 视图时转换列。
- 创建 XPath 查询谓词来在 XML 视图中筛选元素或属性。
- 预览 XML 数据。
- 将文件名列添加到 XML 视图。
- 添加引用端口。
- 重新创建实体关系。
- 在“列”窗口中设置 XML 视图选项。

WSDL 工作区规则和准则

当您在 WSDL 工作区中向 Web 服务源或目标定义添加或编辑 XML 视图时，请使用以下规则和准则：

- Web 服务映射的源和目标定义必须包含信封视图，该视图相当于 Web 服务请求、响应和错误消息的 SOAP:envelope。
- 一个源定义必须为输入消息定义视图。它不能为输出或错误消息定义视图。

- 针对源或目标定义的根组和根组主键的名称必须使用以下命名约定，其中 *<NameString>* 可以是任意字母数字字符串：
 - 根组必须命名为 Message 或 X_*<NameString>*_Envelope。
 - 根组的主键必须命名为 PK_Message 或 PK_*<NameString>*_Envelope。
 - 根组和其主键的 *<NameString>* 必须相同。
- 目标定义必须为输出或错误消息定义视图。它不能为输入消息定义视图。
- 您可以使用类型 anyType 或 any 定义元素。您可以为类型为 anyType 的元素创建字符串端口，或将其映射到类型 complexType 的元素。
- 目标定义中的信封视图必须包含视图根和视图行作为信封节点。
- 您不能更改源或目标定义中的 soap:Body 和 soap:Header 元素的类型定义。
- 您可以设置默认命名空间并更改在源或目标的定义视图中定义的命名空间的前缀。您不能更改命名空间。您不能使用任何以下字符串作为命名空间的前缀：
 - mime
 - wsdl
 - soap
 - soapenc
 - http

使用 Web 服务映射

使用 Web 服务映射概览

创建 Web 服务源和目标定义之后，您可以创建映射以确定集成服务如何处理 Web 服务请求中收到的数据并发送 Web 服务响应。Web 服务映射作为 SOAP 请求接收输入消息，转换数据，然后作为 SOAP 响应发送输出消息。

您可以在 Mapping Designer 中创建 Web 服务映射，方法与创建其他 PowerCenter 映射相同。将 Web 服务源和目标定义及转换添加到映射中。

您还可以生成包含 Web 服务源定义、源限定符转换和 Web 服务目标定义的映射。PowerCenter Designer 提供多种生成 Web 服务映射的方法。

您可以使用以下方法生成 Web 服务映射：

- **通过 WSDL 导入源和目标定义。**您可以通过 WSDL 创建映射，方法与通过 WSDL 创建 Web 服务源或目标相同。
- **通过关系或平面文件源或目标定义生成。**您可以通过关系或平面文件源或目标创建映射，方法与通过关系或平面文件源或目标创建 Web 服务源或目标相同。
- **通过转换或 Mapplet 生成。**您可以通过可重用转换或具有单输入和单输出的 Mapplet 创建映射。

生成映射后，您可以添加更多转换、链接及其他任何完成 Web 服务映射所需的映射对象。

Web 服务映射类型

您可以创建映射以通过 Web 服务客户端接收消息，转换数据，将响应发送回 Web 服务客户端或写入任何 PowerCenter 支持的目标。集成服务能够根据源和目标定义作为 SOAP 请求的一部分接收和发送附件。

您还可以使用平面文件或 XML 源和目标创建映射，并将其用于 Web 服务工作流。这允许您通过 SOAP 调用以附件形式接收消息数据，而不是从文件中读取它。

要创建的映射取决于要运行的 Web 服务类型：

- **请求-响应 Web 服务。** 请求-响应 Web 服务从 Web 服务客户端接收传入请求，转换数据，然后将响应发送回 Web 服务客户端。请求-响应 Web 服务使用 Web 服务源和 Web 服务目标。

您可以创建一个或多个映射来处理请求-响应 Web 服务：

- **单个映射。** 创建一个包含 Web 服务源和 Web 服务目标定义的映射。集成服务在单个会话中接收传入请求，转换数据，然后发回响应。
 - **多个映射。** 创建多个映射以在将响应发回 Web 服务客户端前暂存数据。您可以针对每个映射创建包含一个会话的工作流。
- **单向 Web 服务。** 如果从 Web 服务客户端接收更新和通知，但不需要发回响应，您可以创建单向映射。单向映射使用 Web 服务客户端作为源。集成服务将数据加载到目标（通常由实时事件通过 Web 服务请求触发）。

映射中包含的 Web 服务源和目标定义取决于要创建的映射的类型。

下表根据映射类型介绍了所用的 Web 服务源和目标定义：

映射类型	Web 服务源	Web 服务目标
请求-响应	必须包含一个 Web 服务源定义的实例。	必须包含一个 Web 服务目标定义的实例。目标定义中可以有多于一个错误视图。
单向	必须包含一个 Web 服务源定义的实例。	无 Web 服务目标定义。

相关主题：

- [“附件” 页面上 78](#)

请求-响应映射

请求-响应映射使用 Web 服务源和 Web 服务目标。

如果创建请求-响应映射，请使用以相同方法创建的源和目标定义。如果通过 WSDL 导入源定义，请通过 WSDL 中的相同操作导入目标定义。如果通过定义列或使用关系或平面文件源和目标来创建源定义，请使用相同的方法创建目标定义。

要确保使用相同的方法创建 Web 服务的源和目标定义，请在同一进程中创建源和目标定义。

注意：如果没有从 WSDL 中的相同操作导入源和目标定义，或者没有使用相同方法创建它们，您可能会得到意外的结果。

您可以使用 SQL 转换在请求-响应映射中更新数据库或检索多个数据库行。SQL 转换可以向目标返回多个数据库行。当处理过程中出现数据库错误时，SQL 转换接收来自数据库的错误并将错误文本发送到目标。

有关使用 SQL 转换获取多个行的 Web 服务示例，请参阅 PowerCenter 附带的实时 Web 服务示例。默认情况下，实时 Web 服务示例程序安装在以下目录：

`/<PowerCenterInstallDir>/server/samples/WebServices/samples/RealTimeWebServices`

相关主题：

- [“创建一个源或目标定义” 页面上 68](#)

暂存映射

如果要运行请求-响应会话，但需要先暂存数据，则可以创建多个映射来处理数据。

例如，要接收需要处理的消息数据，您必须通过 WebSphere MQ 执行异步外部系统调用。创建以下映射：

1. 使用 Web 服务源定义创建请求映射。此映射写入平面文件目标和 WebSphere MQ 目标。您可以将所有消息数据写入这两个目标。
外部应用程序从 WebSphere MQ 目标接收并处理消息，然后将消息发送到其他 WebSphere MQ 队列。
2. 使用 Web 服务目标定义创建响应映射。此映射使用第一个映射中的平面文件目标作为源。它还使用包含处理过的数据的 WebSphere MQ 队列作为源。

Web 服务中心使用消息 ID 连接暂存映射中的请求和响应。要在暂存映射中使用 Web 服务源和目标定义，必须在 Web 源和目标定义中包含消息 ID。

相关主题：

- [“消息 ID” 页面上 66](#)

通过 WSDL 生成映射

您可以通过从 WSDL（可在本地访问或通过 URL 访问）导入 Web 服务源或目标来生成 Web 服务映射。

如果通过从 WSDL 导入源和目标定义生成 Web 服务映射，Designer 通过所选操作的输入消息创建源定义。通过所选操作的输出消息创建目标定义。

通过 WSDL 生成的 Web 服务映射包含以下对象：

- Web 服务源定义
- 源限定符
- Web 服务目标定义

Designer 链接从源实例到目标实例的端口。要完成映射，请为要创建的 Web 服务添加转换和所需的其他映射组件。

要通过 WSDL 生成映射，请按以下步骤操作：

1. 在 PowerCenter Designer 中，打开 Mapping Designer。
2. 单击“映射”>“创建 Web 服务映射”>“从 WSDL 导入”。

通过从 WSDL 导入源和目标生成 Web 服务映射的步骤与从 WSDL 创建 Web 服务源或目标定义的步骤相同。有关详细信息，请参阅[“导入 Web 服务源或目标定义” 页面上 65](#)。

3. 将映射保存到存储库。

通过关系或平面文件源或目标生成映射

您可以基于关系或平面文件源或目标生成 Web 服务映射。可以使用关系或平面文件源或目标来定义 Web 服务源和目标定义中的列。

如果通过关系或平面文件源或目标生成映射，生成的 Web 服务映射包含以下对象：

- Web 服务源定义
- 源限定符
- Web 服务目标定义

Designer 链接从源实例到目标实例的端口。要完成映射，请为要创建的 Web 服务添加转换和所需的其他映射组件。

注意：当通过关系或平面文件源或目标生成映射时，您需要以相同的步骤创建 Web 服务源和目标。如果运行的工作流所包含的映射具有在不同时间创建的 Web 服务源和目标，工作流可能失败。

要通过关系或平面文件源或目标生成映射，请按以下步骤操作：

1. 在 PowerCenter Designer 中，打开 Mapping Designer。
2. 单击“映射”>“创建 Web 服务映射”>“使用源/目标定义”。

通过关系或平面文件源或目标生成 Web 服务映射的步骤与通过关系或平面文件源或目标创建 Web 服务的源和目标定义的步骤相同。有关详细信息，请参阅[“创建一个源或目标定义”](#)页面上 68。

3. 将映射保存到存储库。

通过转换或 Mapplet 生成映射

您可以通过可重用转换或 Mapplet 生成映射。Designer 使用转换或 Mapplet 中的端口生成 Web 服务源和目标定义。

注意：通过转换或 Mapplet 生成映射时，您必须在同一进程中创建 Web 服务源和目标。如果运行的工作流所包含的映射具有在不同时间创建的 Web 服务源和目标，工作流可能失败。

通过可重用转换生成映射

下表介绍了可从中生成 Web 服务映射的转换类型：

转换	类型	组
表达式	被动	单个
HTTP	被动	一个输入和一个输出
Java	主动或被动	一个输入和一个输出
查找	被动	单个
SQL	主动或被动	一个输入和一个输出
存储过程	被动	单个

用于生成 Web 服务映射的转换必须是可重用转换。当基于转换生成 Web 服务映射时，Designer 列出可重用转换以及文件夹中可用的可重用转换的快捷方式。

当通过转换生成 Web 服务映射时，Designer 使用转换中的端口为源和目标定义定义列，然后创建包含源定义（具有反映转换输入端口的 XML 视图）和目标定义（具有反映转换输出端口的 XML 视图）的映射。

通过转换生成的 Web 服务映射包含以下对象：

- Web 服务源定义
- 源限定符
- 用于生成映射的转换
- Web 服务目标定义

Designer 链接从源实例到目标实例的端口。

通过 Mapplet 生成映射

您可以通过以下类型的 Mapplet 生成 Web 服务映射：

- 包含一个输入转换和一个输出转换的 Mapplet
- 包含非主动转换的 Mapplet

通过 Mapplet 生成 Web 服务映射时，Designer 列出 Mapplet 和进程允许的 Mapplet 的快捷方式。

通过 Mapplet 生成 Web 服务映射时，Designer 使用 Mapplet 中的端口定义源和目标定义的列，然后创建包含源定义（具有反映 Mapplet 输入端口的 XML 视图）和目标定义（具有反映 Mapplet 输出端口的 XML 视图）的映射。

通过 Mapplet 生成的 Web 服务映射包含以下对象：

- Web 服务源定义
- 源限定符
- 用于生成映射的 Mapplet
- Web 服务目标定义

Designer 链接从源实例到目标实例的端口。

通过可重用转换或 Mapplet 生成映射

使用相同的步骤通过可重用转换或 Mapplet 生成 Web 服务映射。

要通过可重用转换或 Mapplet 生成 Web 服务映射，请按以下步骤操作：

1. 在 Mapping Designer 中，单击“映射” > “创建 Web 服务映射” > “使用转换/Mapplet 定义”。
2. 选择要用于 Web 服务映射的转换或 Mapplet。
Designer 显示输入端口及数据类型、精度和小数位数的列表。
您可以指定映射中的源和目标定义中的列出现一次或多次。
3. 如果列出现多次，请选择“源和目标为多次出现对象”。
此选项指明源和目标中的列是数组。作为组的列出现多次。
4. 要将消息和客户端端口添加到源或目标定义，请单击“添加消息端口”。

Designer 将消息和客户端端口添加到源和目标定义的信封视图。

5. 单击“确定”。

Designer 创建 Web 服务映射，并显示已成功创建映射的消息。它使用转换或 Mapplet 的名称加 *m_* 前缀作为源和目标定义的名称。此外，为源定义名称添加后缀 *_input*，为目标定义名称添加后缀 *_output*。

编辑 Web 服务映射中的目标实例

生成 Web 服务映射后，您可以编辑映射中的目标实例。在 Mapping Designer 中编辑目标实例时，您可以编辑 Target Designer 中不可用的属性。

要编辑 Web 服务映射中的目标定义，请在 Mapping Designer 中双击目标定义实例。

您可以在“属性”选项卡上编辑以下转换属性：

- 加载范围
- 部分加载恢复

加载范围

加载范围属性指定目标的加载范围。Web 服务目标定义中的加载范围类似于转换中的转换范围。

可将加载范围设置为以下值：

- 事务。将加载范围设置为事务时，集成服务接收事务中的所有数据后生成响应。目标中的所有组必须从同一事务生成器接收数据。
- 全部输入。将加载范围设置为“全部输入”时，集成服务接收所有传入数据后生成响应。目标中的不同组可以从不同事务生成器接收数据。当加载范围为“全部输入”时，集成服务忽略提交。

默认为事务。有关转换范围的详细信息，请参阅《PowerCenter 高级工作流指南》。

部分加载恢复

部分加载恢复属性指定目标在恢复过程中如何处理以前的部分加载内容。

对于 Web 服务目标，请使用默认值“无”。您不能为 Web 服务指定恢复。

附件

您可以配置 PowerCenter Web 服务工作流通过以下方法使用附件：

- 使用平面文件或 XML 源或目标作为 SOAP 消息的附件
- 使用具有 MIME 附件的 WSDL

平面文件或 XML 源和目标附件

您可以以 SOAP 消息请求或响应附件的形式接收或发送数据。源或目标可以是平面文件或 XML 文档。例如，您定期使用 FTP 访问包含来自 Web 服务应用程序的消息的平面文件。您可以不使用 FTP，而是创建 Web 服务工作流以作为 SOAP 请求附件从平面文件接收数据。

要以 SOAP 消息请求附件的形式接收数据，请在映射中使用平面文件或 XML 源定义。要将平面文件用作 Web 服务的源，请配置读取器，为平面文件使用 Web Services Provider 读取器。编辑运行映射的 Web 服

务会话。在会话属性中，单击“映射”选项卡，然后选择源。将读取器从“平面文件读取器”更改为“针对平面文件的 Web Services Provider 读取器”。

要以 SOAP 消息响应附件的形式发送数据，请在映射中使用平面文件或 XML 目标定义。要将平面文件用作 Web 服务的目标，请配置写入器以使用针对平面文件的 Web Services Provider 写入器。编辑运行映射的 Web 服务会话。在会话属性中，单击“映射”选项卡，然后选择目标。将写入器从“平面文件写入器”更改为“针对平面文件的 Web Services Provider 写入器”。

WSDL 附件

根据源和目标定义，您可以作为 SOAP 请求的一部分接收和发送附件。附件必须是文本文件，如 XML 文档。您不能附加二进制文档，如 JPEG、GIF 或 PDF 文件。例如，您可以从 Oracle 数据库提取 XML 文档，然后将其作为响应消息的附件传递到 Web 服务客户端。

要将二进制文件用作源，将该文件传递到 Web 服务源之前，请将其转换为 hexbinary 或 base64binary。Hexbinary 或 base64binary 文件被视为文本文件。同样，您可以将 Web 服务目标生成的文本文件响应转换为二进制文件。

下表介绍了 Web 服务定义中的附件组端口：

端口名称	说明
FK_Att_Name	生成的指向根组中的 PK_Message 的外键。
Att_Data_Name	包含附件。您可以在“属性”选项卡上查看附件的 MIME 类型。
Att_Index_Name	消息中的各个附件的唯一标识符。
Att_Type_Name	附件的类型。

使用具有 MIME 附件的 WSDL 的规则和准则

使用附件时，请遵循以下规则和准则：

- 一个请求或响应可以包含一个附件。
- 附件必须是文本文件，并且使用 UTF-8 代码页或作为 UTF-8 代码页子集的代码页。
- 要通过请求或响应传递附件，您必须连接附件组中的所有端口。
- 当映射中的定义包含附件组时，如果不希望发送或接收附件，则无需连接组中的任何端口。
- 如果从其他源接收消息且各个消息包含附件，请使用“序列生成器”转换来为响应中发送的各个附件生成唯一索引。
- 要发送或接收附件，请使用支持 MIME 附件的工具包创建客户端应用程序。

使用 Web 服务工作流

使用 Web 服务工作流概览

使用 Workflow Manager 创建 Web 服务工作流。要创建 Web 服务工作流，请启用工作流的 Web 服务选项，然后配置 Web 服务属性。

当您在 Web 服务工作流中创建会话时，该会话称为 Web 服务会话。您可以在 Web 服务会话中包括以下类型的映射：

- Web 服务映射
- 平面文件映射
- XML 映射

Web 服务会话使用 Web Services Provider 读取器和写入器。如果 Web 服务映射包含 Web 服务源和目标，则默认情况下，会话将使用 Web Services Provider 读取器和写入器。如果 Web 服务映射包含平面文件或 XML 源或目标，则必须将读取器和写入器类型更改为 Web Services Provider 读取器或写入器。

当 Web 服务会话包含 XML 或平面文件源或目标时，客户端应用程序将向 Web 服务中心发送请求，作为 SOAP 消息的 MIME 附件。要发送或接收附件，必须使用支持 MIME 附件的工具包来创建客户端应用程序。

当 Web 服务中心收到 SOAP 消息请求，要求运行 Web 服务工作流时，它会将请求传递给集成服务。集成服务运行 Web 服务请求后，会将响应传递给 Web 服务中心。Web 服务中心将生成 SOAP 消息响应并将其传回 Web 服务客户端。

在包含 Web 服务源和目标定义的会话中，您可以设置多个分区。集成服务将根据会话中源、目标和分区的数量创建与 Web 服务中心的连接。

注意：在运行 Web 服务工作流之前，必须在 Administrator 工具中创建和配置 Web 服务中心，并将其与包含要运行的 Web 服务工作流的存储库相关联。

创建和配置 Web 服务工作流

要创建 Web 服务工作流，请将工作流配置为处理 Web 服务映射并启用工作流属性中的“Web 服务”选项。您可以将 Web 服务配置为允许 Web 服务客户端运行工作流。

要创建和配置 Web 服务工作流，请完成以下任务：

- 创建 Web 服务工作流。
- 配置 Web 服务。

创建 Web 服务工作流

要创建 Web 服务工作流，请启用工作流的“Web 服务”选项。然后，配置 Web 服务并将 Web 服务会话添加到工作流。Web 服务会话基于 Web 服务映射。

在大多数情况下，Web 服务工作流包含输入消息的一个 Web 服务源和输出消息的一个 Web 服务目标。会话可以写入目标中的多个错误视图。单向 Web 服务不会发送响应，并且不需要 Web 服务目标。

确保在创建 Web 服务工作流时指定集成服务。使用“浏览集成服务”按钮从可用集成服务列表中进行选择。

创建 Web 服务工作流后，您可以添加会话以运行 Web 服务映射。创建会话并将其添加到 Web 服务工作流的方式与创建会话并将其添加到任何工作流的方式相同。

注意：请勿使用工作流向导创建 Web 服务工作流。如果使用工作流向导，则无法选择“Web 服务”选项。

要创建 Web 服务工作流，请执行以下操作：

1. 在 Workflow Manager 中，打开 Workflow Designer 并单击“工作流”>“创建”。
2. 输入工作流的名称。

3. 要选择要运行工作流的集成服务，请单击“浏览集成服务”按钮，从列表中选择。
4. 启用“Web 服务”选项，然后单击“配置服务”以配置 Web 服务工作流。
当您启用“Web 服务”选项时，默认情况下也将启用“配置并发执行”选项。Web 服务工作流配置属性包括 Web 服务并发执行的设置。
5. 根据需要配置 Web 服务工作流属性。
6. 单击“确定”。

配置 Web 服务工作流

配置 Web 服务工作流时，可以分配运行 Web 服务工作流的 Web 服务中心并配置用于运行和访问 Web 服务的选项。

下表介绍了您可以为 Web 服务配置的属性：

属性	说明
服务名称	Web 服务的名称。当您签入工作流且服务可见时，Web 服务中心将发布此名称。默认名称由存储库名称、文件夹名称和工作流名称串联组成。此名称必须是唯一的。
超时（秒）	Web 服务中心处理请求并生成 SOAP 响应需要的最长时间，超过此时间后，该请求将超时。如果 Web 服务中心无法在超时时限内生成响应，它将向 Web 服务客户端发送错误消息，然后删除连接。 默认值为 60 秒。设置为 0 可禁用超时时限。
服务时间阈值（毫秒）	Web 服务中心处理请求需要的最长时间，超过此时间后，它将启动另一个实例来处理下一个请求。服务时间段为从 Web 服务中心接收 SOAP 请求的时间到它生成 SOAP 响应的的时间。如果 Web 服务中心处理请求所需的平均时间超过服务时间，Web 服务中心将启动新的 Web 服务实例，以处理新请求。 例如，服务时间设置为 1000 毫秒。如果 Web 服务中心无法在 1000 毫秒内处理请求，它将启动另一个 Web 服务实例来处理下一个 SOAP 请求。 默认值为 1000。 注意:要避免性能下降，请勿将服务时间阈值设置为低于 100 毫秒。
Web 服务中心	要运行工作流的 Web 服务中心服务。单击“浏览”按钮以选择一个或多个要运行 Web 服务工作流的 Web 服务中心服务。默认情况下，Web 服务工作流可以在与存储库相关联的任何 Web 服务中心服务上运行。 注意: 如果计划手动启动工作流，请选择要运行该工作流的 Web 服务中心。请勿选择“在所有 Hub 上运行”。在启动该 Web 服务工作流之前，请验证是否启用了 Web 服务中心。
每 Hub 最大运行计数	可以由 Web 服务中心启动的最大 Web 服务实例数量。在 Web 服务中心上运行的 Web 服务工作流的所有实例都包含在计数中，无论动态或手动启动实例均是如此。达到最大数量后，Web 服务中心将无法启动其他 Web 服务实例。
受保护	在运行 Web 服务之前需要进行身份验证。Web 服务中心基于用户名标志对请求进行身份验证。您可以选择保护服务或将其公开。 可以运行工作流的任何 PowerCenter 用户都可以使用 Workflow Manager、 <i>pmcmd</i> 或 LMAPI 运行受保护的 Web 服务工作流。如果 Web 服务不受保护，则任何 Web 服务客户端都可以启动服务而无需身份验证。 有关详细信息，请参阅 “为客户端请求添加安全性” 页面上 59 。

属性	说明
可见	使 Web 服务在 Web 服务中心控制台中可见。使服务可见后，Web 服务中心将在 Web 服务中心控制台上发布该 Web 服务和 WSDL。可以测试该 Web 服务并从 Web 服务中心控制台查看和下载 WSDL。 如果服务不可见，则 Web 服务中心不会发布 Web 服务 WSDL。
可运行	允许 Web 服务客户端通过将请求发送到 Web 服务中心启动工作流。如果 Web 服务工作流可运行，则 Web 服务客户端请求可以启动工作流或在工作流运行时运行 Web 服务。如果您希望 Web 服务客户端启动工作流，请计划工作流按需运行。如果 Web 服务工作流不可运行，则 Web 服务客户端可以在工作流运行时调用 Web 服务，但不能启动工作流。如果禁用，则可以通过 Workflow Manager、LMAPI 或 <i>pmcmd</i> 启动该工作流。

Web 服务工作流的并发执行

Web 服务中心根据资源可用性以及您为 Web 服务的属性设置的值确定何时启动新的 Web 服务工作流实例。它根据您为 Web Services Provider 读取器的属性设置的值确定何时关闭 Web 服务工作流实例。

启动新实例

Web 服务中心根据当前资源使用情况以及 Web 服务工作流的以下属性确定何时启动另一个 Web 服务工作流实例。

- **服务时间阈值。**如果 Web 服务中心处理 Web 服务所需的平均时间超过服务时间阈值，Web 服务中心将启动另一个 Web 服务实例。
- **每 Hub 最大运行计数。**Web 服务中心启动 Web 服务实例，直到实例的数量达到 Hub 的最大运行计数。如果达到最大运行计数，则即使超过平均服务时间阈值，Web 服务中心也不会启动新的 Web 服务实例。

关闭实例

Web 服务中心根据当前资源使用情况以及 Web Services Provider 读取器的以下属性关闭 Web 服务工作流实例：

- **空闲时间。**如果工作流实例未在空闲时间段内收到任何请求，Web 服务中心将关闭该工作流实例。
- **消息计数。**当工作流实例所接收的消息数达到配置集成服务在会话中读取的最大消息数时，Web 服务中心将关闭该工作流实例。
- **读取器时限。**当集成服务达到可以从 Web 服务中心读取输入消息的最长时间时，集成服务将停止从 Web 服务中心读取输入消息。Web 服务中心将关闭工作流实例。

如果其中任何属性达到为工作流配置的阈值，Web 服务中心将关闭 Web 服务工作流实例。

相关主题：

- [“配置 Web 服务工作流” 页面上 81](#)
- [“配置 Web Services Provider 读取器” 页面上 83](#)

配置 Web Services Provider 读取器和写入器

当您配置 Web 服务会话时，可以配置会话读取器和写入器。默认情况下，具有 Web 服务源和目标的 Web 服务会话使用 Web Services Provider 读取器和写入器。

如果 Web 服务会话包含平面文件或 XML 源或目标，则必须将会话配置为使用 Web Services Provider 读取器或写入器。Web 服务中心作为 SOAP 消息的 MIME 附件发送请求和响应。

当您为 Web 服务会话配置读取器时，可以配置终止条件，如空闲时间和消息计数。

当您为 Web 服务会话配置写入器时，可以配置集成服务用于缓存目标数据的缓存信息。还可以配置目标数据的输出格式。

使用 Workflow Manager 配置 Web 服务会话。在 Workflow Designer 中，编辑 Web 服务工作流的会话。要配置 Web Services Provider 读取器，请单击“映射”选项卡并选择源。要配置 Web Services Provider 写入器，请选择目标。

相关主题：

- [“附件” 页面上 78](#)

配置 Web Services Provider 读取器

为 Web Services Provider 读取器配置的属性取决于映射中使用的源类型。

下表介绍了为 Web 服务会话配置的源属性：

属性	读取器类型	说明
空闲时间	<ul style="list-style-type: none">- Web 服务- Web Services Provider 读取器平面文件- Web Services Provider 读取器 XML 文件	集成服务等待接收消息的时间（以秒为单位），超过此时间后，集成服务将停止从源读取，并且 Web 服务中心将关闭工作流实例。符合此属性的条件时，会话将停止。默认值为 180。
消息计数	<ul style="list-style-type: none">- Web 服务- Web Services Provider 读取器平面文件- Web Services Provider 读取器 XML 文件	集成服务读取的消息数，超过此数量后，Web 服务中心将关闭工作流实例。值为 -1 表示无限多个消息。如果会话使用平面文件或 XML 目标，请始终将消息计数配置为 1。有关详细信息，请参阅 “为 XML 和平面文件会话配置读取器和写入器” 页面上 85 。符合此属性的条件时，会话将停止。默认值为 -1。
读取器时限	<ul style="list-style-type: none">- Web 服务- Web Services Provider 读取器平面文件- Web Services Provider 读取器 XML 文件	集成服务从 Web 服务中心读取源消息的时间（以秒为单位）。例如，如果将读取器时限设置为 10，则 10 秒后集成服务将停止从 Web 服务中心读取。符合此属性的条件时，会话将停止。默认为 0，表示无限的时间段。
将空内容视为空值	Web Services Provider 读取器 XML 文件	将空字符串视为空值。默认情况下，空内容不为空。
恢复缓存文件夹	n/a	此属性未被 Web Services Provider 使用。

配置 Web Services Provider 写入器

当您为 Web Services Provider 写入器配置会话属性时，可以配置缓存大小和缓存目录。

下表介绍了为 Web 服务会话配置的目标属性：

属性	写入器类型	说明
XML 日期时间格式	Web Services Provider 写入器 XML 文件	传递给服务目标的数据的日期时间格式。精确到纳秒。 从以下日期时间格式中选择： <ul style="list-style-type: none"> - 本地时间。基于集成服务服务器时区的时间。 - 带时区格式的本地时间。集成服务时区与格林尼治标准时间之间的时差（以小时为单位）。 - UTC。格林尼治标准时间。
空内容表示形式	Web Services Provider 写入器 XML 文件	确定空内容在目标中的表示形式。 从以下选项中选择： <ul style="list-style-type: none"> - 无标记。将不输出标记。 - 具有空内容的标记。将仅输出标记。 默认为“无标记”。
空字符串内容表示形式	Web Services Provider 写入器 XML 文件	确定空字符串在目标中的表示形式。 从以下选项中选择： <ul style="list-style-type: none"> - 无标记。将不输出标记。 - 具有空内容的标记。将仅输出标记。 默认为“具有空内容的标记”。
重复组行处理	Web Services Provider 写入器 XML 文件	确定集成服务如何在会话期间处理重复组行。 从以下选项中选择： <ul style="list-style-type: none"> - 第一行。集成服务将第一个重复行传递给目标。集成服务拒绝它在此行之后处理的具有相同主键的行。 - 最后一行。集成服务将最后一个重复行传递给目标。 - 错误。集成服务将第一行传递给目标。之后具有重复主键的行将增加错误计数。当错误计数超出错误阈值时，会话失败。 默认值为“错误”
孤立行处理	Web Services Provider 写入器 XML 文件	确定集成服务如何在会话期间处理孤立行。 从以下选项中选择： <ul style="list-style-type: none"> - 忽略。集成服务将忽略孤行。 - 错误。当错误计数超出错误阈值时，会话失败。
缓存大小	<ul style="list-style-type: none"> - Web 服务 - Web Services Provider 写入器 XML 文件	写入器所使用的内存缓存的总大小（以字节为单位）。它包括目标实例中每个组的主键和外键索引缓存以及所有组的一个数据缓存。总缓存要求是针对每个目标组的数据缓存和索引缓存要求的总和。 默认值为 10,000,000 个字节。
缓存目录	<ul style="list-style-type: none"> - Web 服务 - Web Services Provider 写入器 XML 文件	目标缓存文件的目录。默认为 \$PMCacheDir 服务进程变量。

将写入器类型更改为 Web Services Provider 写入器时，请使用以下规则和准则：

- 当您更改平面文件目标的写入器类型时，集成服务不会缓存目标消息。
- 当您更改写入器类型为平面文件或 XML 目标时，请将目标用作 Web 服务输出消息，而非错误消息。
- 当您更改 XML 目标的写入器类型时，仍可配置 XML 写入器属性。

为 XML 和平面文件会话配置读取器和写入器

要创建基于包含 XML 或平面文件源和目标的映射的 Web 服务会话，请将读取器或写入器类型设置为 Web Services Provider 读取器或写入器。为了运行具有 XML 或平面文件读取器的 Web 服务工作流，客户端应用程序将向 Web 服务中心发送请求，作为 SOAP 消息的 MIME 附件。Web 服务中心会将包含该附件的 SOAP 消息传递给集成服务，以处理附件。

如果 Web 服务工作流配置了 XML 或平面文件写入器，则集成服务将生成响应并将该响应传递给 Web 服务中心。Web 服务中心将响应发送回 Web 服务客户端作为 SOAP 消息的 MIME 附件。

当您配置具有平面文件或 XML 源或目标的请求-响应 Web 服务会话时，请使用以下规则和准则：

- 在读取器属性中将消息计数设置为 1。
- 在工作流中包括一个会话，可在其中将读取器或写入器类型更改为 Web Services Provider。
- 如果在会话属性中将读取器或写入器类型更改为 Web Services Provider 读取器或写入器，则必须使用支持 MIME 附件的工具包创建客户端应用程序。

为 Web 服务会话配置分区

在包含 Web 服务源和目标定义的会话中设置多个分区时，集成服务将根据该会话中源、目标和分区创建与 Web 服务中心的连接。例如，如果在会话中配置三个包含一个源和一个目标的分区，则集成服务将创建六个与 Web 服务中心的连接，三个针对源，三个针对目标。分区允许并发执行 Web 服务请求。

当您运行多分区会话时，Web 服务中心将使用源连接将请求传递给集成服务。集成服务将使用目标连接将响应发送到 Web 服务中心。Web 服务中心和集成服务以轮询方式使用源和目标连接。

当您为 Web 服务映射配置分区时，可以为 Web 服务源和目标配置传递分区。

Web 服务工作流故障排除

我试图对 Web 服务会话运行调试器，但会话失败，并且会话日志包含错误消息，指示运行该会话需要工作流上下文。

如果要调试 Web 服务会话，则必须对 Web 服务工作流运行调试器。如果没有工作流，则无法针对 Web 服务映射或可重用会话运行调试器。

我更新了源 WSDL 并重新导入了我的源和目标定义。工作流有效，但服务 WSDL 未更新。

对映射的更改未动态反映在 Web 服务中心中。要生成 WSDL 以反映映射更改，需要编辑并保存工作流。当您保存工作流时，Web 服务中心将为服务生成 WSDL。

我的 Web 服务工作流在 Workflow Manager 中有效，但在我启动 Web 服务中心时变为无效。

启动 Web 服务中心之后，除了 Workflow Manager 的验证规则以外，它还将根据自己的验证规则验证每个 Web 服务工作流。

Web 服务中心根据以下规则验证 Web 服务工作流：

- 映射中最多只能有一个 Web 服务源定义。
- 映射中最多只能有一个 Web 服务目标定义。

- 如果映射中没有 Web 服务目标定义，则 Web 服务中心会将 Web 服务视为单向服务。
- 存储库服务必须与 Web 服务中心相关联。
- 集成服务必须与工作流相关联。

有关 Web 服务中心错误消息，请参阅 Workflow Manager 中的“验证”选项卡，并更正错误消息中指示的错误。

尝试在 Web 服务中心上提取工作流时，我收到错误消息，指示没有为服务工作流指定的集成服务，并且该服务工作流无效。

创建 Web 服务工作流时，必须分配集成服务。有关详细信息，请参阅[“配置 Web 服务工作流” 页面上 81](#)。

我向配置为在 Web 服务中心上运行多个实例的 Web 服务工作流发送了一个请求。发送请求后，我停止了该 Web 服务工作流。我收到错误响应。

Web 服务中心定期检查工作流的状态。如果它在注册工作流未运行之前向该工作流发送请求，则会生成错误响应。如果工作流配置为运行多个实例，Web 服务中心将启动该工作流的另一个实例。但是，由于 Web 服务中心不缓存请求，它无法将请求重新发送到新的工作流实例。

我在受版本控制的存储库中对实时 Web 服务工作流进行了更改。当我运行该工作流时，更改未生效。

当您在受版本控制的存储库中修改实时 Web 服务工作流时，必须签入工作流，更改才会生效。

例如，修改实时 Web 服务工作流，以将其与其他 PowerCenter 集成服务相关联。如果签入更改，则 Web 服务中心将使用新集成服务来运行工作流。如果不签入更改，Web 服务中心将不使用新集成服务，除非您重新启动 Web 服务中心。

Web 服务示例客户端应用程序

Web 服务示例客户端应用程序概览

Informatica 提供了示例客户端应用程序，用于演示如何使用 PowerCenter Web 服务。这些示例包括以 Java 和 C# 编写的程序。Java 示例程序使用 Axis Web 服务工具包生成的代理类。C# 示例程序使用 wsdl.exe 工具为 .NET 平台生成的代理类。示例程序使用 PowerCenter 批处理 Web 服务和实时 Web 服务。

Web 服务示例程序安装在以下目录中：

```
<PowerCenterInstallDir>/server/samples/WebServices
```

在运行 Web 服务示例程序之前，在 PowerCenter 域上创建并启用 Web 服务中心。使用 Administrator 工具来创建、配置和启用 Web 服务中心。

使用批处理 Web 服务示例程序

在使用批处理 Web 服务示例程序之前，PowerCenter 必须已经安装并且正在运行。PowerCenter 域必须包含与存储库服务关联的 Web 服务中心。

批处理 Web 服务示例程序安装在以下目录中：

```
<PowerCenterInstallDir>/server/samples/WebServices
```

下表描述了 /WebServices 目录中的文件和目录：

目录	说明
/lib	包含运行示例程序所需的库文件。
/ssl	包含在安全模式 (HTTPS) 下运行客户端应用程序的示例密钥库。
/samples/BatchWebServices/axis/ <SampleProgramDirectory>	包含 Java 示例程序。在一个单独的目录中可以找到每个批处理 Web 服务示例程序的源文件。目录名称指示示例程序演示的批处理 Web 服务操作。例如，/multiservers 目录中的示例程序演示如何登录到与存储库服务相关联的多个集成服务。 此目录还包含编译并运行示例程序的批处理和脚本文件。
/samples/BatchWebServices/axis/ proxyclasses	包含 Java 示例程序的代理类。
/samples/BatchWebServices/dotnet/csharp/ <SampleProgramDirectory>	包含 C# 示例程序。在一个单独的目录中可以找到每个批处理 Web 服务示例程序的源文件。目录名称指示示例程序演示的批处理 Web 服务操作。例如，/multiservers 目录中的示例程序演示如何登录到与存储库服务相关联的多个集成服务。 每个示例程序目录还包含编译示例程序的批处理文件。
/samples/BatchWebServices/dotnet/csharp/ proxyclasses	包含 C# 示例程序的代理类。此目录还包含编译代理类的批处理文件。

编译批处理 Web 服务示例程序

编译批处理 Web 服务示例程序的步骤由编程语言确定。

编译示例 Java 程序

要编译示例 Java 程序，请转至示例程序目录并运行编译批处理文件或脚本文件。运行与您要编译的示例程序的名称相匹配的批处理或脚本文件。

例如，要编译 /axis/multithreaded 目录中的 Sample1.java 程序，请转至该目录并运行 CompileSample1.bat (Windows) 或 CompileSample1.sh (UNIX)。编译过程在相同的目录中为示例程序创建一个 .class 文件。

编译示例 C# 程序

要编译示例 C# 程序，请完成以下步骤：

1. 转到 /dotnet/csharp/proxyclasses 目录并运行 compile.bat。
编译过程在以下目录中创建名为 WebServicesHub.dll 的动态链接库：
/dotnet/csharp/bin directory.
2. 转到示例程序目录并运行与您要编译的示例程序的名称相匹配的编译批处理文件。
编译过程使用编译的程序文件名称和 .exe 扩展名创建一个可执行文件。

运行批处理 Web 服务示例程序

当运行客户端应用程序时，必须运行 Web 服务中心。
使用所需参数运行示例程序。运行批处理 Web 服务示例程序的步骤由编程语言确定。

运行示例 Java 程序

要运行示例 Java 程序，请转到示例程序目录并运行与您要运行的示例程序的名称相匹配的批处理或脚本文件。例如，要运行 /axis/multithreaded 目录中的 Sample1.java 程序，请转至该目录并运行 RunSample1.bat (Windows) 或 RunSample1.sh (UNIX)。

运行示例 C# 程序

要运行示例 C# 程序，请转至示例程序目录并运行要运行的示例程序的可执行文件。

批处理 Web 服务示例

下表列出了包含示例程序的目录：

平台	目录
Java	/WebServices/samples/BatchWebServices/axis/<SampleProgramDirectory>
C#	/WebServices/samples/BatchWebServices/dotnet/csharp/<SampleProgramDirectory>

为 Java 和 C# 提供了相同的示例程序集。每个平台都有相同的目录，每个目录包含用于演示 Web 服务的不同用法的示例程序。本节介绍了 Java 和 C# 示例程序。

浏览

/browsing 目录中的示例程序说明从存储库中获取信息的 Web 服务操作的使用方法。

Sample1.java 和 Sample1.cs

此示例程序登录到一个存储库，然后使用元数据 Web 服务操作获取有关存储库中文件夹、工作流和任务及使用该存储库注册的集成服务的信息。

目录： /browsing

编译 Java 和 C# 示例的文件： CompileSample1.bat 或 CompileSample1.sh

运行 Java 示例的文件： RunSample1.bat 或 RunSample1.sh

运行 C# 示例的文件： Sample1.exe

下表描述了您用来运行 Sample1 应用程序的参数：

参数	说明
安全模式	指示运行应用程序的安全模式。传递参数 <i>-ns</i> 可以在不安全模式 (HTTP) 下运行该应用程序。 这些示例不支持安全模式 (HTTPS)。
主机名	运行 Web 服务中心的计算机名称或 IP 地址。
端口号	运行 Web 服务中心的端口号。
存储库域名	包含存储库服务的域的名称。
存储库名称	存储库服务的名称。
用户名	登录到存储库的用户名。
密码	登录到存储库的用户名的密码。

Sample2.java 和 Sample2.cs

此示例程序登录到一个存储库，并连接到关联的集成服务。它使用元数据和数据集成 Web 服务操作来访问存储库中的文件夹并开始和停止文件夹中找到的第一个工作流。

目录： /browsing

编译 Java 和 C# 示例的文件： CompileSample2.bat 或 CompileSample2.sh

运行 Java 示例的文件： RunSample2.bat 或 RunSample2.sh

运行 C# 示例的文件： Sample2.exe

下表描述了您用来运行 Sample2 应用程序的参数：

参数	说明
安全模式	指示运行应用程序的安全模式。传递参数 <i>-ns</i> 可以在不安全模式 (HTTP) 下运行该应用程序。 这些示例不支持安全模式 (HTTPS)。
主机名	运行 Web 服务中心的计算机名称或 IP 地址。
端口号	运行 Web 服务中心的端口号。
存储库域名	包含存储库服务的域的名称。
存储库名称	存储库服务的名称。
用户名	登录到存储库的用户名。
密码	登录到存储库的用户名的密码
集成服务域名	包含集成服务的域的名称。
集成服务名称	集成服务的名称。

数据集成

/dataintegration 目录中的示例程序演示如何使用数据集成 Web 服务中可用的工作流和任务操作。

Sample1.java 和 Sample1.cs

此示例程序登录到一个存储库，并连接到关联的集成服务。它使用数据集成 Web 服务操作来启动和停止在集成服务上运行的工作流。

目录： /dataintegration

编译 Java 和 C# 示例的文件： CompileSample1.bat 或 CompileSample1.sh

运行 Java 示例的文件： RunSample1.bat 或 RunSample1.sh

运行 C# 示例的文件： Sample1.exe

下表描述了您用来运行 Sample1 应用程序的参数：

参数	说明
安全模式	指示运行应用程序的安全模式。传递参数 <i>-ns</i> 可以在不安全模式 (HTTP) 下运行该应用程序。 这些示例不支持安全模式 (HTTPS)。
主机名	运行 Web 服务中心的计算机名称或 IP 地址。
端口号	运行 Web 服务中心的端口号。
存储库域名	包含存储库服务的域的名称。
存储库名称	存储库服务的名称。
用户名	登录到存储库的用户名。
密码	登录到存储库的用户名的密码。
集成服务域名	包含集成服务的域的名称。
集成服务名称	集成服务的名称。
文件夹名称	存储库中的文件夹名称。
工作流名称	包含会话的工作流的名称。
任务名称	要开始的任务的名称。

多个集成服务

/multiservers 目录中的示例程序演示如何登录到与存储库服务相关联的多个集成服务。可以使用相同的技术同时访问与存储库服务相关联的任意数量的集成服务。

Sample1.java 和 Sample1.cs

此示例程序登录到一个存储库，并连接到与存储库关联的其中两个集成服务。它使用数据集成 Web 服务操作来获取这两种集成服务的属性。

注意: 如示例所示, 您必须为这些数据集成 Web 服务创建两个代理对象, 以登录到两个集成服务。为每个要登录的集成服务创建一个代理对象。

目录: /multiservers

编译 Java 和 C# 示例的文件: CompileSample1.bat 或 CompileSample1.sh

运行 Java 示例的文件: RunSample1.bat 或 RunSample1.sh

运行 C# 示例的文件: Sample1.exe

下表描述了您用来运行 Sample1 应用程序的参数:

参数	说明
安全模式	指示运行应用程序的安全模式。传递参数 <i>-ns</i> 可以在不安全模式 (HTTP) 下运行该应用程序。 这些示例不支持安全模式 (HTTPS)。
主机名	运行 Web 服务中心的计算机名称或 IP 地址。
端口号	运行 Web 服务中心的端口号。
存储库域名	包含存储库服务的域的名称。
存储库名称	存储库服务的名称。
用户名	登录到存储库的用户名。
密码	登录到存储库的用户名的密码。
集成服务域名	包含集成服务的域的名称。
集成服务名称 1	与存储库关联的一个集成服务的名称。
集成服务名称 2	与存储库关联的第二个集成服务的名称。

多线程

/multithreaded 目录中的示例程序演示了如何在多个线程中使用代理对象以并行执行操作。您可以使用相同的技术使客户端应用程序继续运行, 并在等待操作完成时调用其他操作。例如, 如果一个客户端应用程序在线程上调用 WaitTillWorkflowComplete 操作, 应用程序可以继续对其他线程执行其他操作。

Sample1.java 和 Sample1.cs

此示例程序登录到一个存储库, 并连接到关联的集成服务。它启动两个线程, 并将数据集成 Web 服务代理对象传递给这两个线程。在一个线程上, 它在集成服务上启动一个工作流并等待该工作流完成。在另一个线程上, 它获取集成服务的属性。同样, 您可以在多个线程中使用元数据 Web 服务代理对象。

目录: /multithreaded

编译 Java 和 C# 示例的文件: CompileSample1.bat 或 CompileSample1.sh

运行 Java 示例的文件: RunSample1.bat 或 RunSample1.sh

运行 C# 示例的文件: Sample1.exe

下表描述了您用来运行 Sample1 应用程序的参数：

参数	说明
安全模式	指示运行应用程序的安全模式。传递参数 <i>-ns</i> 可以在不安全模式 (HTTP) 下运行该应用程序。 这些示例不支持安全模式 (HTTPS)。
主机名	运行 Web 服务中心的计算机名称或 IP 地址。
端口号	运行 Web 服务中心的端口号。
存储库域名	包含存储库服务的域的名称。
存储库名称	存储库服务的名称。
用户名	登录到存储库的用户名。
密码	登录到存储库的用户名的密码。
集成服务域名	包含集成服务的域的名称。
集成服务名称	集成服务的名称。
文件夹名称	存储库中包含该工作流的文件夹的名称。
工作流名称	存储库中的工作流的名称。

Web 服务中心测试

/testsamples 目录中的示例程序演示如何确认在 PowerCenter 域中运行的 Web 服务中心有效。

Sample1.java 和 Sample1.cs

此示例程序登录到一个存储库，并连接到关联的集成服务。它使用元数据和数据集成 Web 服务操作来获取有关存储库服务和集成服务的信息。

目录： /testsamples

编译 Java 和 C# 示例的文件： CompileSample1.bat 或 CompileSample1.sh

运行 Java 示例的文件： RunSample1.bat 或 RunSample1.sh

运行 C# 示例的文件： Sample1.exe

下表描述了您用来运行 Sample1 应用程序的参数：

参数	说明
安全模式	指示运行应用程序的安全模式。传递参数 <i>-ns</i> 可以在不安全模式 (HTTP) 下运行该应用程序。 这些示例不支持安全模式 (HTTPS)。
主机名	运行 Web 服务中心的计算机名称或 IP 地址。

参数	说明
端口号	运行 Web 服务中心的端口号。
存储库域名	包含存储库服务的域的名称。
存储库名称	存储库服务的名称。
用户名	登录到存储库的用户名。
密码	登录到存储库的用户名的密码。
集成服务域名	包含集成服务的域的名称。
集成服务名称	集成服务的名称。

使用实时 Web 服务示例程序

在使用实时 Web 服务示例程序之前，PowerCenter 必须已经安装并且正在运行。PowerCenter 域必须包含与存储库服务关联的 Web 服务中心。

实时 Web 服务示例程序安装在以下目录中：

```
<PowerCenterInstallDir>/server/samples/WebServices
```

实时 Web 服务示例包括要创建示例程序使用的查找表和 Web 服务工作流的文件。

下表列出了 /RealTimeWebServices 目录中的文件和目录：

目录	说明
/samples/RealTimeWebServices/ImportXML	包含由实时 Web 服务示例程序调用的 Web 服务工作流。要使用示例程序，将 XML 文件导入到存储库中，并设置 Web 服务工作流中的 SQL 和查找转换的数据库连接。
/lib	包含运行示例程序所需的库文件。
/samples/RealTimeWebServices/SQLScripts /SINGLEROWLOOKUP	包含创建在示例程序中进行单行查找的查找表的 SQL 脚本。运行 SQL 脚本在选择数据库中创建表。
/samples/RealTimeWebServices/SQLScripts /MULTIPLEROWLOOKUP	包含创建在示例程序中进行多行查找的查找表的 SQL 脚本。运行 SQL 脚本在选择数据库中创建表。
/samples/RealTimeWebServices/Unprotected WebServices/axis/<SampleProgramDirectory>	包含 Java 示例程序。在一个单独的目录中可以找到每个实时 Web 服务示例程序的源文件。每个目录包含批处理和脚本文件，用于为示例程序使用的代理类编译和运行示例程序和子文件夹。

要使用实时 Web 服务示例，您必须完成以下步骤：

1. 创建示例程序将用作查找表的数据库表。
2. 将映射和 Web 服务工作流导入到与 Web 服务中心相关联的存储库中。
3. 为 m_CustomerLookup_MULTIPLEROW 映射中的 SQL 转换修改数据库和数据类型。

4. 设置示例工作流中的数据库连接设置。
5. 编译实时 Web 服务示例程序。
6. 运行实时 Web 服务示例程序。

步骤 1。创建查找表

使用批处理 Web 服务示例程序附带的 SQL 脚本文件在关系数据库中创建查找表。您可以在以下数据库中创建查找表：

- IBM DB2
- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

注意: 如果您在 Teradata 中创建查找表，必须将数据库服务器的默认模式设置为 ANSI。

下表描述了 SQL 脚本：

脚本文件名	说明
CustomerLookup_SINGLEROW_<Database>.sql	创建一个名为 SINGLEROWLOOKUP 的客户表，以用于示例程序来进行单行查找。
CustomerLookup_MULTIPLEROW_<Database>.sql	创建一个名为 MULTIPLEROWLOOKUP 的客户表，以用于示例程序来进行多行查找。

注意: 数据库连接设置。将示例工作流导入存储库中后，您需要修改工作流中转换的数据库连接设置，以匹配您的数据库设置。

步骤 2。导入映射和工作流

实时 Web 服务示例程序运行示例 Web 服务工作流。要使用示例程序，请将示例映射和工作流导入与 Web 服务中心相关联的存储库。

下表描述了这些 XML 文件：

脚本文件名	说明
wf_CustomerLookup_SINGLEROW.XML	包含带有查找转换（用于单行查找的示例程序）的 Web 服务工作流。
wf_CustomerLookup_MULTIPLEROW.XML	包含带有 SQL 转换（用于多行查找的示例程序）的 Web 服务工作流。

步骤 3。为 SQL 转换修改数据库和数据类型

演示多行查找如何使用一个 SQL 转换的 Web 服务示例。您使用的数据库决定可用于 SQL 转换中的端口的本地数据类型。您必须配置 SQL 转换以使用适当的数据库，并将端口设置为使用适当的本地数据类型。

要修改 SQL 转换的数据库和数据类型，请完成以下步骤：

1. 在 PowerCenter Designer 中，在 Mapping Designer 中打开 m_CustomerLookup_MULTIPLEROW 映射并编辑 sql_Customer 转换实例。
2. 在“编辑转换”窗口中，转到“SQL 设置”选项卡并将“数据库类型”属性的值设置为您为该示例使用的数据库。
3. 转到“SQL 端口”选项卡，验证端口的数据类型是否映射到正确的本地数据类型。
对于大多数数据库，默认的本地数据类型映射是正确的。对于 Microsoft SQL Server 和 Sybase，将字符串数据类型映射到 varchar 本地数据类型。
4. 将所做的更改保存到 m_CustomerLookup_MULTIPLEROW 映射。
修改映射后，刷新运行映射的工作流。
5. 在 PowerCenter Workflow Manager 中，打开运行映射的工作流并刷新映射。

步骤 4。修改数据库连接设置

导入的工作流中的 SQL 和查找转换必须能够连接到您在步骤 1 中创建的示例查找表。

导入进程不导入示例工作流中的转换的连接对象。您必须创建一个连接对象，并在会话中使用它。

要更新转换的连接设置，请完成以下步骤：

1. 在 PowerCenter Workflow Manager 中，创建一个连接对象，以连接到示例表。
2. 编辑 s_m_CustomerLookup_SINGLEROW 会话并更新 lkp_Customer 转换中的关系连接信息。
将关系连接设置为新连接对象的名称。使用新设置保存会话。
3. 编辑 s_m_CustomerLookup_MULTIPLEROW 会话并更新 sql_Customer 转换中的关系连接信息。
将关系连接设置为您的连接对象的名称。使用新设置保存会话。

步骤 5。编译实时 Web 服务示例程序

要编译示例 Java 程序，请转至示例程序目录并运行编译批处理文件或脚本文件。运行与您要编译的示例程序的名称相匹配的批处理或脚本文件。

例如，要编译 /axis/CustomerLookup_SINGLEROW 目录中的 Sample.java 程序，请转至该目录并运行 CompileSample.bat (Windows) 或 CompileSample.sh (UNIX)。

编译过程在相同的目录中为示例程序创建一个 .class 文件。

步骤 6。运行实时 Web 服务示例程序

您必须在运行示例程序的计算机上安装 Java 版本 1.5.0_11-b03。当您运行示例程序时，必须运行 Web 服务中心。

要运行示例 Java 程序，请转至示例程序目录并运行要运行的示例程序的批处理或脚本文件。例如，要运行 /axis/CustomerLookup_MULTIPLEROW 目录中的 Sample.java 程序，请转至该目录并运行 RunSample.bat (Windows) 或 RunSample.sh (UNIX)。

使用所需参数运行示例程序。

实时 Web 服务示例

本节介绍了用于实时 Web 服务的示例程序。每个目录包含一个演示使用实时 Web 服务的不同方式的示例程序。

多行查找

/CustomerLookup_MULTIPLEROW 目录中的示例程序演示客户端应用程序运行 Web 服务工作流的方法，以执行查找并处理包含多行数据的响应。

Sample.java

此示例程序调用 PowerCenter Web 服务工作流，用于在数据库中查找客户 ID，并打印出客户信息。该工作流使用 SQL 转换从数据库中检索多个行。

目录： /CustomerLookup_MULTIPLEROW

编译 Java 示例的文件： CompileSample.bat 或 CompileSample.sh

运行 Java 示例的文件： RunSample.bat 或 RunSample.sh

下表描述了您用来运行 Sample 应用程序的参数：

参数	说明
客户 ID	用于查找的客户 ID。将客户 ID 作为整数传递。
端点 URL	用于找到 Web 服务的 URL。将端点 URL 作为字符串传递。 可以在 Web 服务 WSDL 中的服务元素的 soap:address 位置元素中找到实时 Web 服务的端点 URL。示例 Web 服务的默认端点 URL 是 <code>http://<WSHHostName>:<WSHPort>/wsh/services/ts/CustomerLookup_MULTIPLEROW</code> 。 如果 Web 服务中心以 HTTPS 方式运行，使用 HTTPS 启动端点 URL。

单行查找

/CustomerLookup_SINGLEROW 目录中的示例程序演示客户端应用程序运行 Web 服务工作流的方法，以执行查找并处理包含单行数据的响应。

Sample.java

此示例程序调用 PowerCenter Web 服务工作流，用于在数据库中查找客户 ID，并打印出客户信息。映射使用查找转换以从数据库中检索一行。

目录： /CustomerLookup_SINGLEROW

编译 Java 示例的文件： CompileSample.bat 或 CompileSample.sh

运行 Java 示例的文件： RunSample.bat 或 RunSample.sh

下表描述了您用来运行 Sample 应用程序的参数：

参数	说明
客户 ID	用于查找的客户 ID。将客户 ID 作为整数传递。
端点 URL	用于找到 Web 服务的 URL。将端点 URL 作为字符串传递。 可以在 Web 服务 WSDL 中的服务元素的 soap:address 位置元素中找到实时 Web 服务的端点 URL。示例 Web 服务的默认端点 URL 是 <code>http://<WSHHostName>:<WSHPort>/wsh/services/ts/CustomerLookup_SINGLEROW</code> 。 如果 Web 服务中心以 HTTPS 方式运行，使用 HTTPS 启动端点 URL。

配置 Web 浏览器

配置 Web 浏览器

您可以使用 Microsoft Internet Explorer 或 Google Chrome 浏览器在 Informatica 平台上启动 Web 服务中心控制台。

要运行 Web 服务中心控制台，请在您的浏览器中配置以下选项：

脚本和 ActiveX

在 Microsoft Internet Explorer 上启用以下控制项：

- 活动脚本
- 允许编程式剪贴板访问
- 运行 ActiveX 控件和插件
- 标记为脚本安全的脚本 ActiveX 控件

要配置控件，单击**工具 > Internet 选项 > 安全 > 自定义级别**。

受信任站点

如果 Informatica 域运行在使用 Kerberos 身份验证的网络上，必须将浏览器配置为允许访问 Informatica Web 应用程序。在 Microsoft Internet Explorer 和 Google Chrome 中，将 Informatica Web 应用程序的 URL 添加到受信任站点列表。如果使用 Chrome 版本 41 或更高版本，还必须设置 AuthServerWhitelist 和 AuthNegotiateDelegateWhitelist 策略。