



Informatica®
10.1

Developer ツールガイド

本ソフトウェアおよびマニュアルには、Informatica LLC の所有権下にある情報が収められています。これらは使用および開示の制限等を定めた使用許諾契約のもとに提供され、著作権法により保護されています。当該ソフトウェアのリバースエンジニアリングは禁じられています。本マニュアルのいかなる部分も、いかなる手段（電子的複写、写真複写、録音など）によっても、Informatica LLC の事前の承諾なしに複製または転載することは禁じられています。このソフトウェアは、米国および/または国際的な特許、およびその他の出願中の特許によって保護されています。

合衆国政府によるソフトウェアの使用、複製または開示は、DFARS 227.7202-1 (a) および 227.7702-3 (a) (1995 年)、DFARS 252.227-7013(C) (1) (ii) (1988 年 10 月)、FAR 12.212 (a) (1995 年)、FAR 52.227-19、または FAR 52.227-14 (ALT III) に記載されているとともに、当該ソフトウェア使用許諾契約に定められた制限によって規制されます。

本製品または本書の情報は、予告なしに変更されることがあります。お客様が本製品または本書内に問題を発見された場合は、書面に当社までお知らせください。

Informatica、Informatica Platform、Informatica Data Services、PowerCenter、PowerCenterRT、PowerCenter Connect、PowerCenter Data Analyzer、PowerExchange、PowerMart、Metadata Manager、Informatica Data Quality、Informatica Data Explorer、Informatica B2B Data Transformation、Informatica B2B Data Exchange、Informatica On Demand、Informatica Identity Resolution、Informatica Application Information Lifecycle Management、Informatica Complex Event Processing、Ultra Messaging、Informatica Master Data Management、および Live Data Map は、Informatica LLC の米国および世界中の管轄地での商標または登録商標です。その他のすべての企業名および製品名は、それぞれの企業の商標または登録商標です。

本ソフトウェアまたはドキュメントの一部は、次のサードパーティが有する著作権に従います（ただし、これらに限定されません）。Copyright DataDirect Technologies. All rights reserved. Copyright (C) Sun Microsystems. All rights reserved. Copyright (C) RSA Security Inc. All rights reserved. Copyright (C) Ordinal Technology Corp. All rights reserved. Copyright (C) Aandacht c.v. All rights reserved. Copyright Genivia, Inc. All rights reserved. Copyright Isomorphic Software. All rights reserved. Copyright (C) Meta Integration Technology, Inc. All rights reserved. Copyright (C) Intalio. All rights reserved. Copyright (C) Oracle. All rights reserved. Copyright (C) Adobe Systems Incorporated. All rights reserved. Copyright (C) DataArt, Inc. All rights reserved. Copyright (C) ComponentSource. All rights reserved. Copyright (C) Microsoft Corporation. All rights reserved. Copyright (C) Rogue Wave Software, Inc. All rights reserved. Copyright (C) Teradata Corporation. All rights reserved. Copyright (C) Yahoo! Inc. All rights reserved. Copyright (C) Glyph & Cog, LLC. All rights reserved. Copyright (C) Thinkmap, Inc. All rights reserved. Copyright (C) Clearpace Software Limited. All rights reserved. Copyright (C) Information Builders, Inc. All rights reserved. Copyright (C) OSS Nokalva, Inc. All rights reserved. Copyright Edifecs, Inc. All rights reserved. Copyright Cleo Communications, Inc. All rights reserved. Copyright (C) International Organization for Standardization 1986. All rights reserved. Copyright (C) ej-technologies GmbH. All rights reserved. Copyright (C) Jaspersoft Corporation. All rights reserved. Copyright (C) International Business Machines Corporation. All rights reserved. Copyright (C) yWorks GmbH. All rights reserved. Copyright (C) Lucent Technologies. All rights reserved. Copyright (C) University of Toronto. All rights reserved. Copyright (C) Daniel Veillard. All rights reserved. Copyright (C) Unicode, Inc. Copyright IBM Corp. All rights reserved. Copyright (C) MicroQuill Software Publishing, Inc. All rights reserved. Copyright (C) PassMark Software Pty Ltd. All rights reserved. Copyright (C) LogiXML, Inc. All rights reserved. Copyright (C) 2003-2010 Lorenzi Davide, All rights reserved. Copyright (C) Red Hat, Inc. All rights reserved. Copyright (C) The Board of Trustees of the Leland Stanford Junior University. All rights reserved. Copyright (C) EMC Corporation. All rights reserved. Copyright (C) Flexera Software. All rights reserved. Copyright (C) Jinfonet Software. All rights reserved. Copyright (C) Apple Inc. All rights reserved. Copyright (C) Telerik Inc. All rights reserved. Copyright (C) BEA Systems. All rights reserved. Copyright (C) PDFlib GmbH. All rights reserved. Copyright (C) Orientation in Objects GmbH. All rights reserved. Copyright (C) Tanuki Software, Ltd. All rights reserved. Copyright (C) Ricebridge. All rights reserved. Copyright (C) Sencha, Inc. All rights reserved. Copyright (C) Scalable Systems, Inc. All rights reserved. Copyright (C) jQWidgets. All rights reserved. Copyright (C) Tableau Software, Inc. All rights reserved. Copyright (C) MaxMind, Inc. All rights reserved. Copyright (C) T-Mate Software s.r.o. All rights reserved. Copyright (C) MapR Technologies Inc. All rights reserved. Copyright (C) Amazon Corporate LLC. All rights reserved. Copyright (C) Highsoft. All rights reserved. Copyright (C) Python Software Foundation. All rights reserved. Copyright (C) BeOpen.com. All rights reserved. Copyright (C) CNRI. All rights reserved.

本製品には、Apache Software Foundation (<http://www.apache.org/>) によって開発されたソフトウェア、およびさまざまなバージョンの Apache License（まとめて「License」と呼んでいます）の下に許諾された他のソフトウェアが含まれます。これらのライセンスのコピーは、<http://www.apache.org/licenses/> で入手できます。適用法にて要求されないか書面に合意されない限り、ライセンスの下に配布されるソフトウェアは「現状のまま」で配布され、明示的あるいは黙示的かを問わず、いかなる種類の保証や条件も付帯することはありません。ライセンス下での許諾および制限を定める具体的文言については、ライセンスを参照してください。

本製品には、Mozilla (<http://www.mozilla.org/>) によって開発されたソフトウェア、ソフトウェア copyright The JBoss Group, LLC、コンテンツの無断複写・転載を禁じます、ソフトウェア copyright, Red Hat Middleware, LLC、コンテンツの無断複写・転載を禁じます、Copyright (C) 1999-2006 by Bruno Lowagie and Paulo Soares および GNU Lesser General Public License Agreement (<http://www.gnu.org/licenses/lgpl.html> を参照) に基づいて許諾されたその他のソフトウェアが含まれています。資料は、Informatica が無料で提供しており、一切の保証を伴わない「現状渡し」で提供されるものとし、Informatica LLC は市場性および特定の目的の適合性の黙示の保証などを含めて、一切の明示的及び黙示的保証の責任を負いません。

製品には、ワシントン大学、カリフォルニア大学アーバイン校、およびバンダービルト大学の Douglas C. Schmidt および同氏のリサーチグループが著作権を持つ ACE (TM) および TAO (TM) ソフトウェアが含まれています。Copyright (C) 1993-2006, All rights reserved.

本製品には、OpenSSL Toolkit を使用するために OpenSSL Project が開発したソフトウェア（copyright The OpenSSL Project. コンテンツの無断複写・転載を禁じます）が含まれています。また、このソフトウェアの再配布は、<http://www.openssl.org> および <http://www.openssl.org/source/license.html> にある使用条件に従います。

本製品には、Curl ソフトウェア Copyright 1996-2013, Daniel Stenberg, <daniel@haxx.se>が含まれます。All Rights Reserved. 本ソフトウェアに関する許諾および制限は、<http://curl.haxx.se/docs/copyright.html> にある使用条件に従います。すべてのコピーに上記の著作権情報とこの許諾情報が記載されている場合、目的に応じて、本ソフトウェアの使用、コピー、変更、ならびに配布が有償または無償で許可されます。

本製品には、MetaStuff, Ltd. のソフトウェアが含まれます。Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved. 本ソフトウェアに関する許諾および制限は、<http://www.dom4j.org/license.html> にある使用条件に従います。

製品には、The Dojo Foundation のソフトウェアが含まれます。Copyright (C) 2004-2007. All Rights Reserved. 本ソフトウェアに関する許諾および制限は、<http://dojotoolkit.org/license> にある使用条件に従います。

本製品には、ICU ソフトウェアおよび他のソフトウェアが含まれます。Copyright International Business Machines Corporation. All rights reserved. 本ソフトウェアに関する許諾および制限は、<http://source.icu-project.org/repos/icu/icu/trunk/license.html> にある使用条件に従います。

本製品には、Per Bothner のソフトウェアが含まれます。Copyright (C) 1996-2006. All rights reserved. お客様がこのようなソフトウェアを使用するための権利は、ライセンスで規定されています。<http://www.gnu.org/software/kawa/Software-License.html> を参照してください。

本製品には、OSSP UUID ソフトウェアが含まれます。Copyright (C) 2002 Ralf S. Engelschall, Copyright (C) 2002 The OSSP Project Copyright (C) 2002 Cable & Wireless Deutschland. 本ソフトウェアに関する許諾および制限は、<http://www.opensource.org/licenses/mit-license.php> にある使用条件に従います。

本製品には、Boost (<http://www.boost.org/>) によって開発されたソフトウェア、または Boost ソフトウェアライセンスの下で開発されたソフトウェアが含まれます。本ソフトウェアに関する許諾および制限は、http://www.boost.org/LICENSE_1_0.txt にある使用条件に従います。

本製品には、University of Cambridge のが含まれます。Copyright (C) 1997-2007. 本ソフトウェアに関する許諾および制限は、<http://www.pcre.org/license.txt> にある使用条件に従います。

本製品には、The Eclipse Foundation のソフトウェアが含まれます。Copyright (C) 2007. All Rights Reserved. 本ソフトウェアに関する許諾および制限は、<http://www.eclipse.org/org/documents/epl-v10.php> および <http://www.eclipse.org/org/documents/edl-v10.php> にある使用条件に従います。

本製品には、<http://www.tcl.tk/software/tcltk/license.html>、<http://www.bosrup.com/web/overlib/?License>、<http://www.stlport.org/doc/license.html>、<http://www.asm.ow2.org/license.html>、<http://www.cryptix.org/LICENSE.TXT>、<http://hsqldb.org/web/hsqldbLicense.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/license.html>、<http://www.jgraph.com/jgraphdownload.html>、<http://www.jcraft.com/jsch/LICENSE.txt>、http://jotm.objectweb.org/bsd_license.html に基づいて許諾されたソフトウェアが含まれています。<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.schneider.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>、および <https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt>。

本製品には、Academic Free License (<http://www.opensource.org/licenses/afl-3.0.php>)、Common Development and Distribution License (<http://www.opensource.org/licenses/cddl1.php>)、Common Public License (<http://www.opensource.org/licenses/cpl1.0.php>)、Sun Binary Code License Agreement Supplemental License Terms、BSD License (<http://www.opensource.org/licenses/bsd-license.php>)、BSD License (<http://opensource.org/licenses/BSD-3-Clause>)、MIT License (<http://www.opensource.org/licenses/mit-license.php>)、Artistic License (<http://www.opensource.org/licenses/artistic-license-1.0>)、Initial Developer's Public License Version 1.0 (<http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/>) に基づいて許諾されたソフトウェアが含まれています。

本製品には、ソフトウェア copyright (C) 2003-2006 Joe Walnes, 2006-2007 XStream Committers が含まれています。All rights reserved.本ソフトウェアに関する許諾および制限は、<http://j.org/license.html> にある使用条件に従います。本製品には、Indiana University Extreme! Lab によって開発されたソフトウェアが含まれています。詳細については、<http://www.extreme.indiana.edu/> を参照してください。

本製品には、ソフトウェア Copyright (C) 2013 Frank Balluffi and Markus Moeller が含まれています。All rights reserved.本ソフトウェアに関する許諾および制限は、MIT ライセンスの使用条件に従います。

特許については、<https://www.informatica.com/legal/patents.html> を参照してください。

免責: 本文書は、一切の保証を伴わない「現状渡し」で提供されるものとし、Informatica LLC は他社の権利の非侵害、市場性および特定の目的への適合性の黙示の保証などを含めて、一切の明示的および黙示的保証の責任を負いません。Informatica LLC では、本ソフトウェアまたはドキュメントに誤りのないことを保証していません。本ソフトウェアまたはドキュメントに記載されている情報には、技術的に不正確な記述や誤植が含まれる場合があります。本ソフトウェアまたはドキュメントの情報は、予告なしに変更されることがあります。

NOTICES

この Informatica 製品（以下「ソフトウェア」）には、Progress Software Corporation（以下「DataDirect」）の事業子会社である DataDirect Technologies からの特定のドライバ（以下「DataDirect ドライバ」）が含まれています。DataDirect ドライバには、次の用語および条件が適用されます。

1. DataDirect ドライバは、特定物として現存するままの状態提供され、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。
2. DataDirect または第三者は、予見の有無を問わず発生した ODBC ドライバの使用に関するいかなる直接的、間接的、偶発的、特別、あるいは結果的損害に対して責任を負わないものとします。本制限事項は、すべての訴訟原因に適用されます。訴訟原因には、契約違反、保証違反、過失、厳格責任、詐称、その他の不法行為を含みますが、これらに限るものではありません。

発行日: 2018-07-23

目次

序文	12
Informatica のリソース.....	12
Informatica Network.....	12
Informatica ナレッジベース.....	12
Informatica マニュアル.....	12
Informatica 製品可用性マトリックス.....	13
Informatica Velocity.....	13
Informatica Marketplace.....	13
Informatica グローバルカスタマサポート.....	13
 第 1 章 : Informatica Developer	14
Informatica Developer の概要.....	14
Informatica Data Quality およびプロファイリング.....	14
Informatica Data Services.....	15
Informatica Developer の起動.....	16
ローカルマシンでの Developer ツールの起動.....	16
リモートマシンでの Developer ツールの起動.....	16
Informatica Developer のインターフェース.....	17
Informatica Developer のビュー.....	18
Informatica Developer のようこそページ.....	20
チートシート.....	20
Informatica Developer のオンラインヘルプ.....	20
Informatica の設定.....	20
Informatica Marketplace.....	21
Informatica Developer の設定.....	21
手順 1。ドメインの追加.....	21
手順 2。モデルリポジトリの追加.....	22
手順 3。デフォルトの Data Integration Service の選択.....	22
ドメイン.....	23
プロジェクト.....	23
プロジェクトの作成.....	23
プロジェクトのフィルタ.....	24
プロジェクトの権限.....	24
外部オブジェクトの権限.....	25
依存オブジェクトインスタンスに対する権限.....	25
親オブジェクトのアクセス.....	26
権限の割り当て.....	26
フォルダ.....	27
フォルダーの作成.....	27
オブジェクトのコピー操作.....	28

オブジェクトのコピー.....	28
オブジェクトのコピーの保存.....	29
タグ.....	29
タグの作成.....	29
タグの割り当て.....	30
タグの表示.....	30
第2章 : モデルリポジトリ.....	31
モデルリポジトリの概要.....	31
Informatica Developer のオブジェクト.....	32
オブジェクトプロパティ.....	34
リポジトリオブジェクトのロック.....	34
ロック管理.....	35
ロック管理のルールとガイドライン.....	35
バージョン管理されたオブジェクトを使用したチームベース開発.....	36
バージョンされたオブジェクトの管理.....	37
オブジェクトの履歴バージョン.....	39
[チェックアウトされたオブジェクト] ビュー.....	40
[バージョン履歴] ビュー.....	40
チームベース開発のトラブルシューティング.....	40
モデルリポジトリへの接続.....	41
モデルリポジトリサービスの更新.....	41
第3章 : Informatica Developer での検索.....	42
Informatica Developer での検索の概要.....	42
モデルリポジトリでの検索.....	42
オブジェクトとプロパティの検索.....	43
ビジネス用語集の検索.....	43
ビジネス用語集デスクトップのルックアップ.....	44
ビジネス用語のルックアップ.....	44
ビジネス用語のルックアップ用にホットキーをカスタマイズ.....	44
エディタでの検索.....	45
第4章 : 接続.....	47
接続の概要.....	47
接続タイプ.....	48
[接続エクスプローラ] ビュー.....	49
接続管理.....	50
接続の作成.....	50
接続の編集.....	51
接続のコピー.....	51
接続の削除.....	52
接続リストの更新.....	52

接続の切り替え.....	52
接続を切り替える前に.....	53
接続の切り替え.....	54
接続を切り替えた後に.....	54
サードパーティ製 JDBC ドライバ.....	56
第 5 章 : 物理データオブジェクト.....	57
物理データオブジェクトの概要.....	57
物理データオブジェクトのタイプ.....	58
リレーショナルデータオブジェクト.....	58
リレーショナルデータオブジェクトのインポート.....	60
キーのリレーション.....	60
カスタマイズデータオブジェクト.....	62
キーのリレーション.....	63
カスタマイズデータオブジェクトの書き込みプロパティ.....	64
カスタマイズデータオブジェクトの作成.....	65
カスタマイズデータオブジェクトへのリレーショナルリソースの追加.....	65
カスタマイズデータオブジェクトへのリレーショナルデータオブジェクトの追加.....	66
カスタマイズデータオブジェクトのキーの作成.....	66
カスタマイズデータオブジェクト内のリレーションの作成.....	67
ターゲットテーブルの作成または置換.....	67
ターゲットテーブルを作成または置換するためのルールおよびガイドライン.....	68
設計時の DDL の生成と実行.....	68
実行時の DDL の生成と実行.....	70
DDL の生成エラー.....	70
カスタムクエリ.....	70
カスタムクエリの最適化.....	71
デフォルトクエリ.....	71
ヒント.....	72
個別に選択.....	73
フィルタ.....	74
ソート済みポート.....	74
ユーザー定義ジョイン条件.....	76
アウタージョインのサポート.....	77
Informatica ジョイン構文.....	77
マッピング実行前および実行後の SQL コマンド.....	81
カスタムクエリの作成.....	82
非リレーショナルデータオブジェクト.....	82
非リレーショナルデータオブジェクトのインポート.....	83
非リレーショナルデータ操作からの読み取り、書き込み、またはルックアップトランスフォーメーションの作成.....	83
WSDL データオブジェクトの概要 WSDL データオブジェクト.....	84
WSDL データオブジェクトの概要ビュー.....	85

WSDL データオブジェクトの詳細ビュー.....	85
WSDL データオブジェクトのインポート.....	85
WSDL の同期.....	86
証明書管理.....	87
同期化.....	88
フラットファイルデータオブジェクトの同期.....	88
リレーショナルデータオブジェクトの同期.....	88
物理データオブジェクトのトラブルシューティング.....	89
第 6 章 : フラットファイルデータオブジェクト.....	90
フラットファイルデータオブジェクトの概要.....	90
ソースファイル名の生成.....	91
フラットファイルデータオブジェクトの概要プロパティ.....	91
フラットファイルデータオブジェクトの詳細プロパティ.....	92
フォーマットのプロパティ.....	93
カラム形式: 区切りプロパティ.....	94
カラム形式: 固定幅プロパティ.....	94
ランタイム:読み取りプロパティ.....	96
ランタイム:書き込みプロパティ.....	98
制御ファイル.....	100
実行時のカラムの更新.....	101
ランタイムカラム名の自動生成.....	101
データファイルヘッダからのランタイムカラム名の生成.....	102
制御ファイルからのカラムメタデータの生成.....	102
制御ファイルの形式.....	103
ランタイムプロパティのパラメータ化.....	103
制御ファイルのランタイム処理.....	103
制御ファイルに関するルールおよびガイドライン.....	104
フラットファイルデータオブジェクトの作成.....	104
空のフラットファイルデータオブジェクトの作成.....	105
既存のフラットファイルからのフラットファイルデータオブジェクトの作成.....	105
制御ファイルからのフラットファイルデータオブジェクトの作成.....	107
第 7 章 : データの論理ビュー.....	109
データの論理ビューの概要.....	109
論理データオブジェクトモデルの例.....	110
データの論理ビューの開発.....	111
論理データオブジェクトモデル.....	111
論理データオブジェクトモデルの作成.....	111
モデリングツールからの論理データオブジェクトモデルのインポート.....	112
論理データオブジェクトモデルのプロパティ.....	113
CA ERwin Data Modeler のインポートプロパティ.....	113

IBM Cognos Business Intelligence Reporting - Framework Manager のインポートプロパティ.....	114
SAP BusinessObjects Designer のインポートプロパティ.....	115
SAP PowerDesigner CDM のインポートプロパティ.....	116
SAP PowerDesigner OOM (9.x - 15.x) のインポートプロパティ.....	117
SAP PowerDesigner PDM のインポートプロパティ.....	118
XSD のインポートプロパティ.....	119
論理データオブジェクト.....	120
論理データオブジェクトのプロパティ.....	120
属性のリレーション.....	121
論理データオブジェクトの作成.....	121
論理データオブジェクトマッピング.....	124
論理データオブジェクト読み取りマッピング.....	124
論理データオブジェクト書き込みマッピング.....	125
論理データオブジェクトマッピングの作成.....	125
第 8 章 : データの表示.....	127
データの表示の概要.....	127
構成.....	128
構成プロパティ.....	128
データビューアの構成.....	131
マッピング構成.....	133
Web サービスの設定.....	134
デフォルトの構成プロパティの更新.....	134
構成のトラブルシューティング.....	135
データのエクスポート.....	135
オブジェクトの依存関係.....	135
オブジェクトの依存関係の表示.....	136
オブジェクトの依存関係の表示.....	136
オブジェクトの依存関係のフィルタリング.....	137
ログ.....	137
ログファイル形式.....	138
検証設定.....	138
エラーメッセージのグループ化.....	138
エラーメッセージの制限.....	139
Developer tool からのジョブの監視.....	139
第 9 章 : アプリケーションデプロイメント.....	140
アプリケーションデプロイメントの概要.....	140
n アプリケーションの作成.....	141
アプリケーションのプロパティ.....	141
アプリケーションデプロイメント.....	143
オブジェクトのデプロイメント.....	143

アプリケーションアーカイブファイルへのデプロイメント.....	144
アプリケーションの再デプロイメント.....	145
アプリケーションの状態情報.....	145
アプリケーションの作成、デプロイ、更新の方法.....	146
アプリケーションの作成.....	147
データ統合サービスへのアプリケーションのデプロイ.....	148
データ統合サービスへのオブジェクトのデプロイ.....	149
アーカイブファイルへのオブジェクトのデプロイ.....	150
アーカイブファイルへのアプリケーションのデプロイ.....	151
アプリケーションアーカイブのインポート.....	152
アプリケーションの更新.....	152
データ統合サービスへのアプリケーションの再デプロイ.....	152
第 10 章 : オブジェクトのインポートとエクスポート.....	154
オブジェクトのインポートとエクスポートの概要.....	154
オブジェクトのインポートとエクスポート.....	155
オブジェクトエクスポート.....	156
オブジェクトのエクスポート.....	156
オブジェクトインポート.....	157
プロジェクトのインポート.....	158
オブジェクトのインポート.....	159
付録 A : データ型リファレンス.....	162
データ型リファレンスの概要.....	162
トランスフォーメーションのデータ型.....	163
Integer データタイプ.....	164
バイナリデータタイプ.....	166
日付/時刻データ型.....	166
10 進型と倍精度浮動小数点数型のデータ型.....	168
文字列データタイプ.....	170
DB2 for i5/OS および DB2 for z/OS のデータ型とトランスフォーメーションのデータ型.....	170
DB2 for i5/OS および DB2 for z/OS のサポートされていないデータ型.....	171
フラットファイルとトランスフォーメーションのデータ型.....	172
DB2 for LUW とトランスフォーメーションデータ型.....	172
JDBC のデータ型とトランスフォーメーションのデータ型.....	174
Microsoft SQL Server のデータ型とトランスフォーメーションデータ型.....	175
Uniqueidentifier データタイプ.....	177
非リレーショナルのデータ型とトランスフォーメーションのデータ型.....	178
ODBC データ型とトランスフォーメーションデータ型.....	180
Oracle のデータ型およびトランスフォーメーションデータ型.....	181
Number (P,S) データ型.....	183
Char、Varchar、Clob データタイプ.....	183
サポートされていない Oracle のデータ型.....	184

SAP HANA およびトランスフォーメーションデータ型.	184
XML データ型とトランスフォーメーションデータ型.	186
データの変換.	188
ポート対ポートのデータ変換.	188
付録 B: キーボードショートカット.	190
オブジェクトのキーボードショートカット.	190
ポートのキーボードショートカット.	191
[トランスフォーメーション] パレットのキーボードショートカット.	192
ワークベンチのキーボードショートカット.	192
付録 C: 接続プロパティ.	194
接続プロパティの概要.	195
Adabas 接続のプロパティ.	195
DataSift 接続プロパティ.	198
Facebook 接続プロパティ.	199
Greenplum 接続プロパティ.	199
HBase 接続プロパティ.	201
HDFS 接続プロパティ.	202
Hive 接続のプロパティ.	203
HTTP 接続プロパティ.	209
IBM DB2 接続のプロパティ.	211
IBM DB2 for i5/OS 接続プロパティ.	214
IBM DB2 for z/OS 接続のプロパティ.	218
IMS 接続のプロパティ.	221
JDBC 接続のプロパティ.	224
JD Edwards EnterpriseOne 接続プロパティ.	228
LDAP 接続プロパティ.	229
LinkedIn 接続プロパティ.	230
Microsoft Azure Blob ストレージ接続のプロパティ.	231
Microsoft Azure SQL Data Warehouse 接続プロパティ.	232
MS SQL Server 接続のプロパティ.	233
Netezza 接続のプロパティ.	237
OData 接続のプロパティ.	238
ODBC 接続のプロパティ.	239
Oracle 接続のプロパティ.	241
Salesforce 接続プロパティ.	243
SAP 接続のプロパティ.	244
シーケンシャル接続のプロパティ.	247
Teradata Parallel Transporter 接続のプロパティ.	249
Twitter 接続プロパティ.	251
Twitter Streaming 接続プロパティ.	252
VSAM 接続のプロパティ.	253

Web Content-Kapow Katalyst 接続プロパティ.....	255
Web サービス接続のプロパティ.....	257
データベース接続における識別子のプロパティ.....	258
通常の識別子.....	259
区切り記号付き識別子.....	259
識別子のプロパティ.....	259
索引.....	262

序文

『Informatica Developer ツールガイド』は、データサービスおよびデータ品質の開発者を対象としています。このガイドでは、Developer ツールの一般的な機能について説明しています。このガイドでは、フラットファイルとリレーショナルデータベースの概念、および作業環境のデータベースエンジンについて理解していることを前提としています。

Informatica のリソース

Informatica Network

Informatica Network は、Informatica グローバルカスタマサポート、Informatica ナレッジベースなどの製品リソースをホストします。Informatica Network には、<https://network.informatica.com> からアクセスしてください。

メンバーは以下の操作を行うことができます。

- 1 つの場所からすべての Informatica のリソースにアクセスできます。
- ドキュメント、FAQ、ベストプラクティスなどの製品リソースをナレッジベースで検索できます。
- 製品の提供情報を表示できます。
- 自分のサポート事例を確認できます。
- 最寄りの Informatica ユーザーグループネットワークを検索して、他のユーザーと共同作業を行えます。

Informatica ナレッジベース

ドキュメント、ハウツー記事、ベストプラクティス、PAM などの製品リソースを Informatica Network で検索するには、Informatica ナレッジベースを使用します。

ナレッジベースには、<https://kb.informatica.com> からアクセスしてください。ナレッジベースに関する質問、コメント、ご意見の連絡先は、Informatica ナレッジベースチーム (KB_Feedback@informatica.com) です。

Informatica マニュアル

使用している製品の最新のドキュメントを取得するには、https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx にある Informatica ナレッジベースを参照してください。

このマニュアルに関する質問、コメント、ご意見の電子メールの送付先は、Informatica マニュアルチーム (infa_documentation@informatica.com) です。

Informatica 製品可用性マトリックス

製品可用性マトリックス（PAM）には、製品リリースでサポートされるオペレーティングシステム、データベースなどのデータソースおよびターゲットが示されています。Informatica Network メンバである場合は、PAM (<https://network.informatica.com/community/informatica-network/product-availability-matrices>) にアクセスできます。

Informatica Velocity

Informatica Velocity は、Informatica プロフェッショナルサービスによって開発されたヒントおよびベストプラクティスのコレクションです。数多くのデータ管理プロジェクトの経験から開発された Informatica Velocity には、世界中の組織と協力して優れたデータ管理ソリューションの計画、開発、展開、および維持を行ってきた弊社コンサルタントの知識が集約されています。

Informatica Network メンバである場合は、Informatica Velocity リソース (<http://velocity.informatica.com>) にアクセスできます。

Informatica Velocity についての質問、コメント、またはアイデアがある場合は、ips@informatica.com から Informatica プロフェッショナルサービスにお問い合わせください。

Informatica Marketplace

Informatica Marketplace は、お使いの Informatica 製品を強化したり拡張したりするソリューションを検索できるフォーラムです。Informatica の開発者およびパートナーの何百ものソリューションを利用して、プロジェクトで実装にかかる時間を短縮したり、生産性を向上させたりできます。Informatica Marketplace には、<https://marketplace.informatica.com> からアクセスできます。

Informatica グローバルカスタマサポート

Informatica Network の電話またはオンラインサポートからグローバルカスタマサポートに連絡できます。

各地域の Informatica グローバルカスタマサポートの電話番号は、Informatica Web サイト (<http://www.informatica.com/us/services-and-training/support-services/global-support-centers>) を参照してください。

Informatica Network メンバである場合は、オンラインサポート (<http://network.informatica.com>) を使用できます。

第 1 章

Informatica Developer

この章では、以下の項目について説明します。

- [Informatica Developer の概要, 14 ページ](#)
- [Informatica Developer の起動, 16 ページ](#)
- [Informatica Developer のインタフェース, 17 ページ](#)
- [Informatica Developer の設定, 21 ページ](#)
- [ドメイン, 23 ページ](#)
- [プロジェクト, 23 ページ](#)
- [プロジェクトの権限, 24 ページ](#)
- [フォルダ, 27 ページ](#)
- [オブジェクトのコピー操作, 28 ページ](#)
- [タグ, 29 ページ](#)

Informatica Developer の概要

Developer tool は、データ統合、データ品質、データプロファイリング、Data Services、およびビッグデータに関連したソリューションを設計および実装するためのアプリケーションです。

Developer tool を使用して、メタデータのインポート、接続の作成、およびデータオブジェクトの作成を実行できます。また、Developer tool を使用して、プロファイル、マッピング、およびワークフローを作成して実行することもできます。

Informatica Data Quality およびプロファイリング

Developer ツールのデータ品質機能を使用して、データの内容と構造を分析し、ビジネスニーズに合った方法でデータを強化します。

Developer ツールを使用して、次のタスクを完了するプロセスを設計および実行します。

- データのプロファイリング。プロファイリングにより、データの内容と構造が明らかになります。プロファイリングは、どのようなデータプロジェクトにおいても重要です。プロファイリングを実施すると、データの強みと弱みが明らかになり、プロジェクト計画の定義に役立つからです。
- スコアカードを作成して、データ品質を確認します。スコアカードは、プロファイルでの品質解析結果を視覚的に表したものです。

- データ値の標準化。データを標準化し、プロファイルを実行したときに見つかるエラーや不整合を除去します。句読点、形式、およびスペルの差異を標準化できます。例えば、市区町村、州、および郵便番号の値に整合性を持たせることができます。
- データの解析。解析では、複数の値で構成されるフィールドを読み取り、各値のフィールドを、含まれている情報のタイプに従って作成します。レコードに情報を追加することもできます。例えば、製品データに測定単位を追加するように解析操作を定義できます。
- 郵便アドレスの検証。アドレス検証では、郵便アドレスデータの精度と配信性を評価および強化します。アドレス検証では、アドレスレコードと国内の郵便運送業者のアドレス参照データを照合することで、アドレスの誤りを修正し、部分的なアドレスを完全なアドレスにします。アドレス検証では、郵便情報を追加して、郵便の配達に要する時間とコストを削減することもできます。
- 重複レコードの検索。重複分析では、各レコードの 1 つ以上のフィールドのデータを比較して、レコード間の類似度を計算します。分析するフィールドを選択し、データに適用する比較戦略を選択します。Developer ツールでは 2 種類の重複分析を実行できます。類似するレコードまたは重複するレコードを特定するフィールド一致と、レコードデータの類似する ID または重複する ID を特定する ID 一致です。
- 例外の管理。例外とは、手動で修正するデータ品質問題が含まれるレコードです。マッピングを実行し、他のデータ品質処理の実行後にデータセットに残っている例外レコードをキャプチャできます。Analyst ツールで例外レコードの確認と編集を行います。
- 参照データテーブルの作成。Informatica には、標準化や解析などのいくつかのタイプのデータ品質プロセスを強化できる参照データが用意されています。プロファイル結果のデータを使用して参照テーブルを作成できます。
- データ品質ルールの作成と実行。Informatica には、ルールが用意されています。このルールは、そのまま実行することも、プロジェクト目標に合わせて編集することもできます。Developer ツールでマプレットを作成し、ルールとして検証することができます。
- Informatica ユーザーとの共同作業。参照データとルールはモデルリポジトリに格納され、このリポジトリは Developer ツールおよび Analyst ツールで使用できます。ユーザーはプロジェクトを共同で実施できます。また、プロジェクトのさまざまな段階でさまざまなユーザーがオブジェクトの所有権を取得できます。
- PowerCenter へのマッピングのエクスポート。PowerCenter にマッピングをエクスポートして実行できます。PowerCenter にマッピングをエクスポートして、物理データの統合のためにメタデータを再利用したり、Web サービスを作成したりできます。

Informatica Data Services

データサービスは、データにアクセスしてそのデータを変換するために実行できる再利用可能な操作の集まりです。

Developer ツールのデータサービス機能を使用して、以下のタスクを実行できます。

- データの論理ビューの定義。データの論理ビューは、エンタープライズ内のデータの構造および用途を表します。エンタープライズで使用されているデータのタイプとそのデータの構造を示す論理データオブジェクトモデルを作成できます。
- データソースまたはターゲットへの論理モデルのマッピング。論理モデルのオブジェクトをデータソースまたはターゲットにリンクするマッピングを作成します。複数の異なるデータソースのデータをリンクすると、単一のデータビューを作成できます。モデルに準拠するデータを複数の異なるターゲットにロードすることもできます。
- データの仮想ビューの作成。仮想統合データベースを Data Integration Service にデプロイすることができます。エンドユーザーは、実際のソースデータに影響を与えることなく、仮想データに対して SQL クエリを実行することができます。
- Web サービスインタフェースを介したデータ統合機能へのアクセスの提供。Web サービスを Data Integration Service にデプロイできます。エンドユーザーは Web サービスに要求を送信し、SOAP メッセージを介して応答を受信します。

- PowerCenter へのマッピングのエクスポート。PowerCenter にマッピングをエクスポートして、物理データの統合のためにメタデータを再利用したり、Web サービスを作成したりできます。
- ドメインユーザーがコマンドラインから実行できるマッピングの作成とデプロイ。
- データのプロファイリング。[プロファイリング] オプションを使用すると、データのプロファイリングによってデータの内容と構造が明らかになります。プロファイリングは、どのようなデータプロジェクトにおいても重要です。プロファイリングを実施すると、データの強みと弱みが明らかになり、プロジェクト計画の定義に役立つからです。

Informatica Developer の起動

Developer ツールがローカルマシンにインストールされている場合は、Windows の [スタート] メニューからツールを起動します。Developer ツールがリモートマシンにインストールされている場合は、コマンドラインからツールを起動します。

ローカルマシンでの Developer ツールの起動

ローカルマシンにインストールされた Developer ツールを起動するには、Windows の [スタート] メニューを使用します。

1. Windows の [スタート] メニューから、[すべてのプログラム] > [Informatica [バージョン]] > [クライアント] > [Developer クライアント] > [Informatica Developer の起動] をクリックします。

Developer ツールの初回実行時には、ようこそページにいくつかのアイコンが表示されます。それ以降、Developer ツールを実行しても、ようこそページは表示されません。

2. [ワークベンチ] をクリックします。

Developer ツールの初回実行時には、ドメインの追加、モデルリポジトリの追加、デフォルトのデータ統合サービスの選択を行って、ツールを設定します。

リモートマシンでの Developer ツールの起動

リモートマシンにインストールされた Developer ツールを起動するには、コマンドラインを使用します。

Developer ツールがリモートマシンにインストールされている場合、インストールディレクトリへの書き込みアクセス権がないことがあります。そのため、Developer ツールが一時ファイルを書き込むワークスペースディレクトリをローカルマシン上に指定します。管理者は、デフォルトのローカルワークスペースディレクトリをすべてのユーザーに対して設定できます。Developer ツールの起動時に、デフォルトのディレクトリをオーバーライドできます。

設定済みのローカルのワークスペースディレクトリが存在しない場合、Developer ツールは一時ファイルを書き込むときにディレクトリを作成します。

1. コマンドプロンプトを開きます。
2. Developer ツールを起動するためのコマンドを入力します。デフォルトのローカルワークスペースディレクトリを使用することも、デフォルトのディレクトリをオーバーライドすることもできます。
 - デフォルトのローカルワークスペースディレクトリを使用するには、次のコマンドを入力します：


```
\\<remote installation directory>\developer.exe
```


以下に例を示します。

```
\\MyRemoteMachine\Informatica\9.5.1\clients\DeveloperClient\developer.exe
```

- デフォルトのローカルワークスペースディレクトリをオーバーライドするには、次のコマンドを入力します:

```
\\<remote installation directory>\developer.exe -data <local workspace directory>
```

以下に例を示します。

```
\\MyRemoteMachine\Informatica\9.5.1\clients\DeveloperClient\developer.exe -data C:\temp\MyWorkspace
```

ローカルワークスペースディレクトリのフォルダ名に番号記号(#)文字は使用できません。ローカルワークスペースディレクトリのフォルダ名にスペースが含まれている場合は、ディレクトリ全体を二重引用符で囲みます。

Developer ツールの初回実行時には、ようこそページにいくつかのアイコンが表示されます。それ以降、Developer ツールを実行しても、ようこそページは表示されません。

3. **【ワークベンチ】** をクリックします。

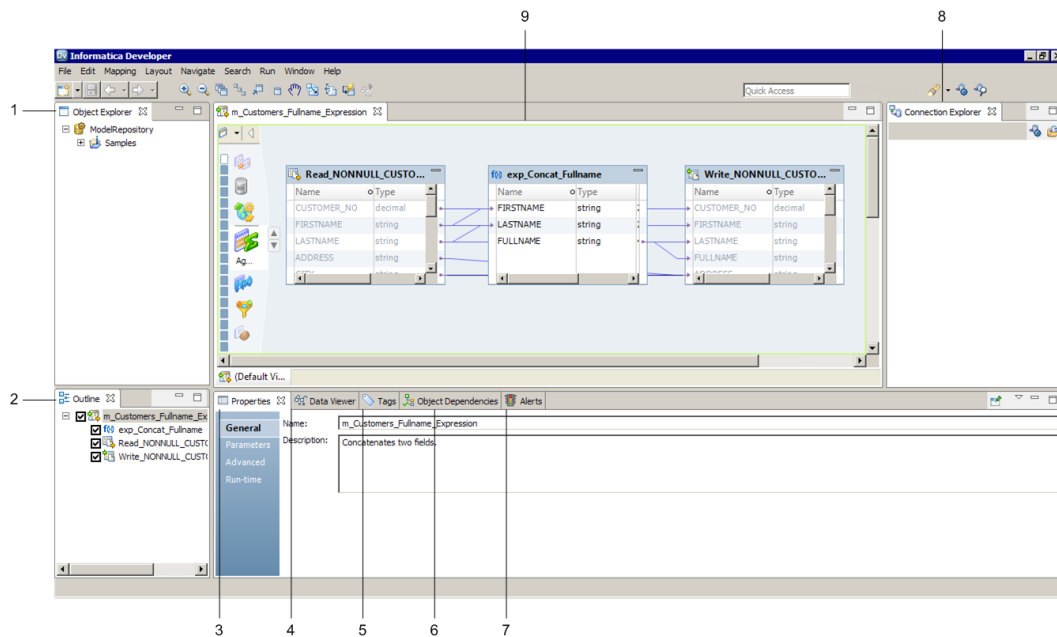
Developer ツールの初回実行時には、ドメインの追加、モデルリポジトリの追加、デフォルトのデータ統合サービスの選択を行って、ツールを設定します。

Informatica Developer のインタフェース

Developer tool を使用すると、データ品質およびデータサービスソリューションを設計、実装できます。

Developer tool で同時に複数のタスクを実行できます。また、同時に複数のフォルダーやプロジェクトで作業することもできます。Developer tool で作業するには、Developer tool のワークベンチにアクセスします。

次の図に Developer tool のワークベンチを示します。



1. [オブジェクトエクスプローラ] ビュー
2. [アウトライン] ビュー
3. [プロパティ] ビュー
4. [データビューア] ビュー
5. [タグ] ビュー
6. [オブジェクトの依存関係] ビュー
7. [警告] ビュー
8. [接続エクスプローラ] ビュー
9. エディタ

Informatica Developer のビュー

Developer tool のワークベンチには、エディタとビューがあります。エディタでマッピングなどのオブジェクトを編集します。Developer tool では、エディタ内で選択されているオブジェクトに応じたビューが表示されます。

追加のビューの選択、ビューの非表示、Developer tool のワークベンチ内の別の場所へのビューの移動などの操作が可能です。

表示するビューを選択するには、[ウィンドウ] > [ビューの表示] をクリックします。

デフォルトビュー

Developer tool では、デフォルトで以下のビューが表示されます。

[オブジェクトエクスプローラ] ビュー

プロジェクト、フォルダ、およびプロジェクトとフォルダ内のオブジェクトが表示されます。

[アウトライン] ビュー

[オブジェクトエクスプローラ] ビューで選択したオブジェクトに応じたオブジェクトが表示されます。

ヘルプビュー

状況依存のオンラインヘルプが表示されます。

【接続エクスペローラ】 ビュー

リレーショナルデータベースへの接続が表示されます。

【プロパティ】 ビュー

エディタで選択されたオブジェクトのプロパティが表示されます。

【データビューア】 ビュー

ソースデータ、プロファイル結果を表示し、トランスフォーメーションの出力をプレビューします。Web サービスメッセージをプレビューしたり、データビューアビューから SQL クエリを実行できます。

【タグ】 ビュー

ビジネス上の用途に基づいてモデルリポジトリ内のオブジェクトを定義するタグが表示されます。

【チェックアウトされたオブジェクト】 ビュー

チェックアウトしたすべてのオブジェクトが表示されます。

【通知】 ビュー

ヒューマンタスクのすべての作業が完了したときにユーザーまたはグループに通知するオプションが表示されます。

【検索】 ビュー

検索結果が表示されます。また、検索オプションダイアログボックスを起動することもできます。

追加のビュー

Developer tool ワークベンチには、次のビューも表示されます。

【警告】 ビュー

接続ステータス警告が表示されます。

【データプロセッサイベント】 ビュー

Developer tool でデータプロセッサトランスフォーメーションを実行するときに起こる初期化、実行、要約イベントに関する情報が表示されます。

【データプロセッサ 16 進数ソース】 ビュー

入力ドキュメントが 16 進形式で表示されます。

【オブジェクトの依存関係】 ビュー

オブジェクトの表示、編集、または削除を行うときにオブジェクトの依存関係が表示されます。

【検証ログ】 ビュー

オブジェクトの検証エラーが表示されます。

【バージョン履歴】 ビュー

選択したオブジェクトのバージョン履歴が表示されます。チェックインコメントを確認し、オブジェクトチェックインに関するユーザー情報を表示できます。

【チートシート】 ビュー

開いているチートシートが表示されます。チートシートを開くには、**【ヘルプ】 > 【チートシート】** をクリックして、チートシートを選択します。

【進捗状況】 ビュー

マッピング実行などの Developer tool での操作の進捗状況が表示されます。

Informatica Developer のようこそページ

Developer tool を初めて開くと、ようこそページが表示されます。ようこそページを使用して、Developer tool に関する知識を深め、Developer tool を設定し、Developer tool での作業を開始します。

ようこそページには次のオプションが表示されます。

- 概要: データ品質およびデータサービスソリューションの概要を表示するには、[概要] ボタンをクリックします。
- 最初の手順: Developer tool の設定および Informatica Data Quality Service と Informatica Data Services のレッスンへのアクセスに関する知識を深めるには、[最初の手順] ボタンをクリックします。
- チュートリアル: Developer tool のチートシートやデータ品質およびデータサービスソリューションのチートシートを表示するには、[チュートリアル] ボタンをクリックします。
- Web リソース: Informatica ナレッジベースへのリンクにアクセスするには、[Web リソース] ボタンをクリックします。Informatica How-To ライブラリにアクセスできます。Informatica How-To ライブラリには、Informatica Data Quality、Informatica Data Services、および他の Informatica 製品に関する記事が掲載されています。
- ワークベンチ: Developer tool での作業を開始するには、[ワークベンチ] ボタンをクリックします。

閉じた後にようこそページにアクセスするには、[ヘルプ] > [ようこそ] をクリックします。

チートシート

Developer ツールには、オンラインヘルプの一部としてチートシートが含まれています。チートシートは段階を追った手引きで、Developer ツールで 1 つまたは複数のタスクを完了するために役立ちます。

チートシートに従って、タスクを完了して結果を確認します。例えば、チートシートを完了して、物理データオブジェクトをインポートおよびプレビューすることができます。

チートシートにアクセスするには、[ヘルプ] > [チートシート] をクリックします。

Informatica Developer のオンラインヘルプ

Informatica Developer のオンラインヘルプシステムには、Developer tool を最大限活用するのに役立つ情報が含まれます。[コンテンツ]、[検索]、および [インデックス] オプションを使用して、Developer tool の特徴と機能について学習します。役に立つページがあれば、そのページへのブックマークを追加します。[関連トピック] オプションを使用して、同様の情報が含まれるページを見つけます。

Informatica の設定

[設定] ダイアログボックスには、Developer ツールおよび Eclipse プラットフォームの設定が含まれます。

Developer ツール内の設定を管理するには、Informatica 設定を使用します。例えば、Informatica 設定を使用して、設定、接続、トランスフォーメーション設定、タグ、使用可能な Data Integration Serviceなどを管理します。

Developer ツールは、Eclipse プラットフォーム上に構築されています。**[設定]** ダイアログには、Eclipse プラットフォームの設定を管理するための設定も含まれます。Informatica では、Informatica 設定のみがサポートされます。

Informatica 設定にアクセスするには、**[ウィンドウ]** > **[設定]** をクリックします。**[設定]** ダイアログボックスで、**[Informatica]** を選択します。

Informatica Marketplace

Informatica Marketplace は、データ統合の実装を強化、拡張、改良するための構築済みのソリューションを提供します。

Informatica Marketplace にアクセスするには、ツールバーで **【マーケットプレイス】** をクリックします。Developer tool に **【マーケットプレイス】** ビューが表示されます。

初めてマーケットプレイスにログインする前に、ユーザーとして登録する必要があります。

ログイン後、エディタで構築済みのソリューションへのリンクを表示できます。**【マーケットプレイス】** 検索ボックスでソリューションを検索し、検索結果を表示してソリューションを見つけることができます。ソリューションには、Developer tool で使用するモデルリポジトリにインポートできるマッピング、マッピングオブジェクト、プロファイルまたはワークフローが含まれている場合があります。

マーケットプレイスソリューションをインポートするには、マーケットプレイスソリューションの横にある **【インポート】** ボタンをクリックし、モデルリポジトリにソリューションをインポートする手順に従います。ソリューションをインポートするには、モデルリポジトリに接続する必要があります。ソリューションの関連ソースファイルおよびマニュアルをコピーするには、インポートプロセス中にフォルダを選択する必要があります。

モデルリポジトリにソリューションをインポートした後、マッピングを実行する、またはマッピングを実行する前に編集することができます。

また、マーケットプレイスコミュニティにソリューションを投稿し、その他のユーザーを支援することもできます。

Informatica Developer の設定

モデルリポジトリオブジェクトにアクセスするように Informatica Developer を設定します。データ統合サービスを選択し、データをプレビューして、プロファイル、マッピング、およびデータフローを実行します。

Developer ツールを設定するには、以下のタスクを実行します。

1. ドメインを追加します。
2. モデルリポジトリを追加します。
3. デフォルトのデータ統合サービスを選択します。

Developer ツールを設定したら、作業を格納するプロジェクトとフォルダーを作成できます。

手順 1. ドメインの追加

ドメインで実行されるサービスにアクセスするには、Developer ツールでドメインを追加します。

ドメインを追加する前に、ドメインに接続するためのドメイン名、ホスト名、およびポート番号があることを確認します。この情報は管理者から提供されます。

1. **【ウィンドウ】** > **【設定】** をクリックします。
【設定】 ダイアログボックスが表示されます。
2. **【Informatica】** > **【ドメイン】** を選択します。
3. **【追加】** をクリックします。
【新しいドメイン】 ダイアログボックスが表示されます。
4. ドメイン名、ホスト名、およびポート番号を入力します。

5. **【完了】** をクリックします。
6. **【OK】** をクリックします。

手順 2。モデルリポジトリの追加

プロジェクトとフォルダーにアクセスするために、モデルリポジトリを追加します。

モデルリポジトリを追加する前に、以下の前提条件を確認します。

- 管理者が Administrator ツールで Model Repository Service を設定済みである。
 - Model Repository Service にアクセスするためのユーザー名とパスワードがある。この情報は管理者から提供されます。
1. **【ファイル】 > 【リポジトリへの接続】** をクリックします。
【リポジトリへの接続】 ダイアログボックスが表示されます。
 2. **【参照】** をクリックして、Model Repository Service を選択します。
 3. **【OK】** をクリックします。
 4. **【次へ】** をクリックします。
 5. ユーザー名とパスワードを入力します。
 6. **【次へ】** をクリックします。
【プロジェクトを開く】 ダイアログボックスが表示されます。
 7. **【Object Explorer】** ビューに表示されるプロジェクトの一覧をフィルタリングするには、開く予定のないプロジェクトを消去します。
 8. **【完了】** をクリックします。
【Object Explorer】 ビューのモデルリポジトリに、選択したプロジェクトが表示されます。

手順 3。デフォルトの Data Integration Service の選択

Data Integration Service は、Developer ツールでデータ統合タスクを実行します。ドメイン内で使用可能な任意の Data Integration Service を選択できます。デフォルトの Data Integration Service を選択します。マッピングまたはデータのプレビューを実行する際、デフォルトの Data Integration Service をオーバーライドできます。

Data Integration Service を選択する前に、ドメインを追加します。

1. **【ウィンドウ】 > 【設定】** をクリックします。
【設定】 ダイアログボックスが表示されます。
2. **【Informatica】 > 【Data Integration Services】** を選択します。
3. ドメインを展開します。
4. Data Integration Service を選択します。
5. **【デフォルトとして設定】** をクリックします。
6. **【OK】** をクリックします。

ドメイン

Informatica ドメインは、Informatica 環境を定義するノードとサービスの集合です。

Developer ツールでドメインを追加します。ドメイン情報を編集したり、ドメインを削除したりすることもできます。Developer ツールの設定でドメイン情報を管理します。

プロジェクト

プロジェクトは、Developer ツールでフォルダおよびオブジェクトを格納するために使用する最上位のコンテナです。

プロジェクトを使用して、データサービスおよびデータ品質ソリューションに使用するオブジェクトを整理および管理します。

[Object Explorer] ビューで、プロジェクトを管理および表示します。プロジェクトを作成すると、Developer ツールにより、プロジェクトがモデルリポジトリ内に格納されます。

作成した各プロジェクトは、Analyst ツールにも表示されます。

以下の表に、プロジェクトに関して実行できるタスクを示します。

タスク	説明
プロジェクトの管理	プロジェクトコンテンツを管理します。プロジェクトの作成、複製、名前変更、および削除が可能です。プロジェクトコンテンツを表示することができます。
プロジェクトのフィルタ	[Object Explorer] ビューに表示されるプロジェクトの一覧をフィルタリングします。
フォルダーの管理	プロジェクトコンテンツをフォルダーで整理します。プロジェクト内のフォルダの作成、複製、名前変更、移動が可能です。
オブジェクトの管理	オブジェクトのコンテンツの表示のほか、プロジェクトまたはプロジェクト内のフォルダーにあるオブジェクトの複製、名前変更、移動、および削除を行います。
プロジェクトの検索	プロジェクト内のフォルダーやオブジェクトを検索します。検索結果を確認し、結果からオブジェクトを選択してコンテンツを表示することができます。
権限の割り当て	プロジェクトのオブジェクトを表示および編集できるユーザーとグループを選択します。他のユーザーおよびグループに権限を割り当てることができるユーザーとグループを指定します。

プロジェクトの作成

プロジェクトを作成して、オブジェクトとフォルダーを格納します。

1. **[Object Explorer]** ビューで Model Repository サービスを選択します。
2. **[ファイル] > [新規] > [プロジェクト]** をクリックします。
[新規プロジェクト] ダイアログボックスが表示されます。
3. プロジェクトの名前を入力します。

4. **【次へ】** をクリックします。
【新規プロジェクト】 ダイアログボックスの **【プロジェクトの権限】** ページが表示されます。
5. 必要に応じて、ユーザーまたはグループを選択して権限を割り当てます。
6. **【完了】** をクリックします。
【Object Explorer】 ビューの **【Model Repository サービス】** の下にプロジェクトが表示されます。

プロジェクトのフィルタ

【Object Explorer】 ビューに表示されるプロジェクトの一覧をフィルタリングできます。膨大な数のプロジェクトのうち、管理の必要なプロジェクトはほんの一部である場合などに、プロジェクトをフィルタリングします。

フィルタリングしたプロジェクトの一覧は、次にリポジトリに接続するときまで保持されます。

以下の時点でプロジェクトをフィルタリングできます。

リポジトリへの接続前

リポジトリへの接続前にプロジェクトをフィルタリングすると、リポジトリに接続するときの所要時間を短縮できます。

【ファイル】 > **【リポジトリへの接続】** を選択します。リポジトリを選択した後、ユーザー名とパスワードを入力し、**【次へ】** をクリックします。**【プロジェクトを開く】** ダイアログボックスに、アクセスできるすべてのプロジェクトが表示されます。リポジトリで開くプロジェクトを選択したら **【完了】** をクリックします。

リポジトリへの接続後

リポジトリに接続している場合は、**【ファイル】** > **【プロジェクトを閉じる】** をクリックし、プロジェクトをフィルタリングして **【Object Explorer】** ビューから除去します。**【プロジェクトを閉じる】** ダイアログボックスには、現在 **【Object Explorer】** ビューで開いているすべてのプロジェクトが表示されます。フィルタリングで除去するプロジェクトを選択し、**【完了】** をクリックします。

フィルタリングしたプロジェクトを開く場合は、**【ファイル】** > **【プロジェクトを開く】** をクリックします。

プロジェクトの権限

プロジェクト権限をユーザーまたはグループに割り当てます。プロジェクト権限は、ユーザーまたはグループがオブジェクトの表示や編集を行えるかどうか、また、他のユーザーに権限を割り当てられるかどうかを決定します。

以下の権限を割り当てることができます。

読み取り

ユーザーまたはグループは、プロジェクトのすべてのオブジェクトを開き、プレビュー、エクスポート、検証、およびデプロイを行うことができます。ユーザーまたはグループはプロジェクトの詳細を表示することもできます。

書き込み

ユーザーまたはグループは、プロジェクトのすべてのオブジェクトに対して読み取り権限があります。さらに、プロジェクトのすべてのオブジェクトの編集、プロジェクトの詳細の編集、プロジェクト内のすべてのオブジェクトの削除、プロジェクトの削除を行うことができます。

付与

ユーザーまたはグループは、プロジェクトのすべてのオブジェクトに対して読み取り権限があります。さらに、ユーザーまたはグループは権限を他のユーザーまたはグループに割り当てることができます。

Model Repository サービスの管理者ロールに割り当てられたユーザーは、Model Repository サービスのすべてのプロジェクトに対するすべての権限を継承します。グループに割り当てられたユーザーは、そのグループの権限を継承します。

外部オブジェクトの権限

権限はプロジェクト内のオブジェクトに適用されます。他のオブジェクト内に存在する依存オブジェクトまでは適用されません。

依存オブジェクトとは、他のオブジェクトで使用されるオブジェクトです。例えば、再利用不可能な式トランスフォーメーションを含むマプレットを作成したとします。この場合、マプレットは親オブジェクトです。式トランスフォーメーションは、このマプレットの依存オブジェクトです。

親オブジェクト内で再利用可能なオブジェクトを使用すると、Developer tool によりオブジェクトのインスタンスが作成されます。例えば、再利用可能なルックアップトランスフォーメーションを使用してマッピングを作成したとします。このマッピングは親オブジェクトです。このマッピングには、ルックアップトランスフォーメーションのインスタンスが含まれます。

オブジェクトは、他のプロジェクトに存在する依存オブジェクトのインスタンスを含むことができます。他のオブジェクトにある依存オブジェクトを表示するには、そのプロジェクトに対する読み取り権限が必要です。他のプロジェクトにある依存オブジェクトを編集するには、親オブジェクトプロジェクトに対する書き込み権限と他のプロジェクトに対する読み取り権限が必要です。

依存オブジェクトインスタンスに対する権限

別のプロジェクトからの依存オブジェクトインスタンスを含むオブジェクトへのアクセスが必要な場合があります。その別プロジェクトに対する読み取り権限がない場合、親オブジェクトへのアクセス権限に基づいて、異なるオプションが表示されます。

表示権限のない依存オブジェクトインスタンスを含む親オブジェクトにアクセスしようとすると、Developer ツールに警告メッセージが表示されます。操作を続行すると、操作タイプごとに異なる結果が生成されます。

以下の表に、親オブジェクトに対して実行可能な操作の結果を示します。

操作	結果
親オブジェクトを開く。	親オブジェクトを開く方法を指定するよう求められます。 <ul style="list-style-type: none">- コピーを開く。親オブジェクトのコピーが作成されます。コピーには、表示権限のない依存オブジェクトインスタンスは含まれません。- 開く。Developer ツールでオブジェクトが開かれますが、表示権限のない依存オブジェクトインスタンスは削除されます。親オブジェクトを保存すると、親オブジェクトから依存オブジェクトインスタンスが削除されます。リポジトリからは依存オブジェクトは削除されません。- キャンセル。親オブジェクトは開かれません。
親オブジェクトを XML ファイルにエクスポートすると、Developer ツールで使えるようになります。	依存オブジェクトインスタンスを含まないエクスポートファイルが作成されます。
親オブジェクトを PowerCenter にエクスポートする。	親オブジェクトをエクスポートできません。

操作	結果
親オブジェクトを検証する。	依存オブジェクトが親オブジェクトの一部でないかのように、親オブジェクトが検証されます。
親オブジェクトをデプロイする。	親オブジェクトをデプロイできません。
親オブジェクトをコピーして貼り付ける。	依存オブジェクトインスタンスを含まない新しいオブジェクトが作成されます。

セキュリティの詳細

表示権限のない依存オブジェクトインスタンスを含むオブジェクトにアクセスすると、Developer ツールに警告メッセージが表示されます。警告メッセージからは、その依存オブジェクトに関する詳細を表示できます。

依存オブジェクトに関する詳細を表示するには、警告メッセージの【詳細】ボタンをクリックします。モデルリポジトリサービスのセキュリティの詳細表示権限がある場合、表示できないオブジェクトを含むプロジェクトが表示されます。ユーザーにセキュリティの詳細表示権限がない場合、プロジェクト名を表示する権限がないことが表示されます。

親オブジェクトのアクセス

別のプロジェクトからの依存オブジェクトインスタンスを使用する親オブジェクトを作成した場合、その親オブジェクトを編集することはできません。ユーザーが親オブジェクトを編集し、その親オブジェクトの機能を保持できるようにするには、マプレットに依存オブジェクトのインスタンスを作成します。

例えば、別のプロジェクトからの再利用可能なルックアップトランスフォーメーションを含むマッピングを作成するとします。プロジェクトのユーザーにマッピングの編集を許可し、ルックアップトランスフォーメーションの編集は禁止するとします。

マッピングにルックアップトランスフォーメーションを配置すると、他のプロジェクトに対する読み取り権限がないユーザーがマッピングを開くと警告メッセージが表示されます。ユーザーはマッピングのコピーやマッピング自体を開くことはできますが、Developer ツールによってルックアップトランスフォーメーションインスタンスは削除されます。

ユーザーがマッピングを編集できるようにするには以下のタスクを実行します。

1. プロジェクトにマプレットを作成します。マプレットに入力トランスフォーメーション、再利用可能なルックアップトランスフォーメーション、および出力トランスフォーメーションを追加します。
2. マッピングを編集し、ルックアップトランスフォーメーションをマプレットで置き換えます。
3. マッピングを保存します。

プロジェクトのユーザーがマッピングを開くと、ルックアップトランスフォーメーションの代わりにマプレットが表示されます。ユーザーは、マプレット以外のマッピングを自由に編集できます。

ユーザーがマッピングをエクスポートした場合、エクスポートファイルにルックアップトランスフォーメーションは含まれません。

権限の割り当て

ユーザーとグループをプロジェクトに追加して、それらのユーザーとグループに権限を割り当てることができます。権限を割り当てて、ユーザーがプロジェクト内のオブジェクトに対して実行できるタスクを決定します。

1. 【Object Explorer】ビューでプロジェクトを選択します。

2. **【ファイル】** > **【プロパティ】** をクリックします。
【プロパティ】 ウィンドウが表示されます。
3. **【権限】** を選択します。
4. **【追加】** をクリックしてユーザーを追加し、そのユーザーに権限を割り当てます。
【ドメインユーザーとグループ】 ダイアログボックスが表示されます。
5. ユーザーおよびグループのリストをフィルタリングするには、名前または文字列を入力します。
必要に応じて、フィルタでワイルドカード文字を使用します。
6. セキュリティドメインでフィルタリングするには、**【セキュリティドメインでフィルタ】** ボタンをクリックします。
7. **【ネイティブ】** を選択して、ネイティブセキュリティドメインのユーザーとグループを表示します。または、**【すべて】** を選択してすべてのユーザーとグループを表示します。
8. ユーザーまたはグループを選択し、**【OK】** をクリックします。
ユーザーとグループが**【新規プロジェクト】** ダイアログボックスの**【プロジェクトの権限】** ページに表示されます。
9. ユーザーとグループに対する読み取り、書き込み、または権限の付与を選択します。
10. **【OK】** をクリックします。

フォルダ

フォルダを使用して、プロジェクト内のオブジェクトを整理します。フォルダを作成して、ビジネスニーズに基づいてオブジェクトをグループ化します。例えば、フォルダを作成すると、プロジェクト内の特定のタスクに使用するオブジェクトをグループ化できます。フォルダは、プロジェクトまたは別のフォルダ内に作成することができます。

フォルダは、**【Object Explorer】** ビューのプロジェクト内に表示されます。フォルダには他のフォルダ、データオブジェクト、およびオブジェクトタイプを含めることができます。

フォルダに対しては、以下のタスクを実行できます。

- フォルダを作成する。
- フォルダを表示する。
- フォルダの名前を変更する。
- フォルダを複製する。
- フォルダを移動する。
- フォルダを削除する。

フォルダーの作成

プロジェクトに、関連するオブジェクトを格納するフォルダーを作成します。フォルダーは、プロジェクトまたは別のフォルダー内に作成する必要があります。

1. **【Object Explorer】** ビューで、フォルダを作成するプロジェクトまたはフォルダを選択します。
2. **【ファイル】** > **【新規】** > **【フォルダ】** をクリックします。
【新規フォルダー】 ダイアログボックスが表示されます。

3. フォルダの名前を入力します。
4. **【完了】** をクリックします。
フォルダーがプロジェクトまたは親フォルダーの下に表示されます。

オブジェクトのコピー操作

オブジェクトをプロジェクト内または別のプロジェクトにコピーすることができます。また、オブジェクトを同じプロジェクト内のフォルダーまたは別のプロジェクト内のフォルダーにコピーすることもできます。

オブジェクトのコピーに別の名前を付けて保存できます。オブジェクトをリンクとしてコピーして、Analyst ツールにオブジェクトを表示したり、メールメッセージなどの別のメディアにオブジェクトへのリンクを設定したりすることもできます。

以下のオブジェクトを、他のプロジェクトまたはフォルダにコピーするか、オブジェクトのコピーに別の名前を付けて保存するか、またはオブジェクトをリンクとしてコピーできます。

- アプリケーション
- データサービス
- 論理データオブジェクトモデル
- Mapping
- マップレット
- 物理データオブジェクト
- プロファイル
- 参照テーブル
- 再利用可能なトランスフォーメーション
- ルール
- スコアカード
- 仮想ストアドプロシージャ
- ワークフロー

オブジェクトをコピーする場合は以下のガイドラインを使用してください。

- マッピング、マップレット、ルール、および仮想ストアドプロシージャのセグメントをコピーできます。
- フォルダーを別のプロジェクトにコピーすることができます。
- 論理データオブジェクトをリンクとしてコピーできます。
- コピーしたオブジェクトは複数回貼り付けることができます。
- プロジェクトまたはフォルダーに同じ名前のオブジェクトが含まれている場合は、そのオブジェクトの名前を変更するか上書きすることができます。

オブジェクトのコピー

オブジェクトをコピーして、別のプロジェクトまたはフォルダーでできるようにします。

1. プロジェクトまたはフォルダ内のオブジェクトを選択します。

2. **【編集】** > **【コピー】** をクリックします。
3. オブジェクトをコピーするプロジェクトまたはフォルダを選択します。
4. **【編集】** > **【貼り付け】** をクリックします。

オブジェクトのコピーの保存

オブジェクトに別の名前を付けて保存するために、オブジェクトのコピーを保存します。

モデルリポジトリの外部の場所にある参照データ値が参照データオブジェクトで指定されている場合に、その参照データオブジェクトのコピーを保存すると、モデルリポジトリはそのオブジェクトを無効としてマークします。コピーを作成するには、**【Object Explorer】** ビューでオブジェクトを選択し、**【編集】** > **【コピー】** をクリックした後、必要な場所にオブジェクトを貼り付けます。

1. エディタでオブジェクトを開きます。
2. **【ファイル】** > **【名前を付けてコピーを保存】** をクリックします。
3. オブジェクトのコピーの名前を入力します。
4. **【参照】** をクリックして、オブジェクトをコピーするプロジェクトまたはフォルダを選択します。
5. **【完了】** をクリックします。

タグ

タグは、ビジネス上の用途に基づいてモデルリポジトリ内のオブジェクトを定義するメタデータです。ビジネス上の用途に従ってオブジェクトをグループ化するには、タグを作成します。

タグを作成したら、そのタグを1つ以上のオブジェクトに関連付けることができます。タグとオブジェクトの関連付けは削除可能です。タグを使用して、モデルリポジトリ内のタグに関連付けられたオブジェクトを検索することができます。Developer ツールにはすべてのタグの用語集が表示されます。

例えば、XYZCorp_CustomerOrders という名前のタグを作成し、XYZ 社の顧客の注文に関する情報が格納されたテーブルに割り当てているとします。この場合、XYZCorp_CustomerOrders タグで検索を実行して、そのタグに関連付けられているテーブルを識別することができます。

注: Developer ツールでオブジェクトに関連付けられたタグは、Analyst ツールでも同じオブジェクトのタグとして表示されます。

タグの作成

タグを作成し、ビジネス上の用途に基づいてオブジェクトを定義するメタデータを追加します。

1. 以下のいずれかの方法を使用して、タグを作成します。
 - **【ウィンドウ】** - **【設定】** をクリックします。**【設定】** ダイアログボックスで、**【Informatica】** > **【タグ】** を選択します。モデルリポジトリサービスを選択し、**【追加】** をクリックします。
 - エディタでオブジェクトを開きます。**【タグ】** ビューで、**【編集】** をクリックします。**【オブジェクトのタグの割り当て】** ダイアログボックスで、**【新規】** をクリックします。
2. タグの名前を入力します。
3. 必要に応じて、説明を入力します。
4. **【OK】** をクリックします。

タグの割り当て

タグをオブジェクトに割り当て、メタデータの定義をオブジェクトに関連付けます。

1. エディタでオブジェクトを開きます。
2. **【タグ】** ビューで、**【編集】** をクリックします。
【オブジェクトのタグの割り当て】 ダイアログボックスが表示されます。**【使用可能なタグ】** 領域に、リポジトリで定義されたすべてのタグが表示されます。名前または説明でタグを検索できます。**【タグの割り当て】** 領域に、開かれているオブジェクトとそのオブジェクトに割り当てられているすべてのタグが表示されます。
3. **【使用可能なタグ】** 領域で、タグを選択します。
4. **【タグの割り当て】** 領域で、オブジェクトを選択します。
5. **【割り当て】** をクリックします。
6. オブジェクトからタグを削除するには、**【使用可能なタグ】** 領域でタグを選択し、**【タグの割り当て】** 領域でオブジェクトを選択して、**【削除】** をクリックします。

タグの表示

オブジェクトに割り当てられているすべてのタグを表示したり、モデルリポジトリで定義されたすべてのタグを表示できます。

1. オブジェクトに割り当てられているタグを表示するには、エディタでそのオブジェクトを開きます。
2. **【タグ】** ビューを選択します。
【タグ】 ビューに、オブジェクトに割り当てられているすべてのタグが表示されます。
3. モデルリポジトリに定義されているすべてのタグを表示するには、**【ウィンドウ】** > **【設定】** をクリックします。
【設定】 ダイアログボックスが表示されます。
4. **【Informatica】** > **【タグ】** を選択します。
【タグ】 領域に、モデルリポジトリで定義されたすべてのタグが表示されます。名前または説明でタグを検索できます。

第 2 章

モデルリポジトリ

この章では、以下の項目について説明します。

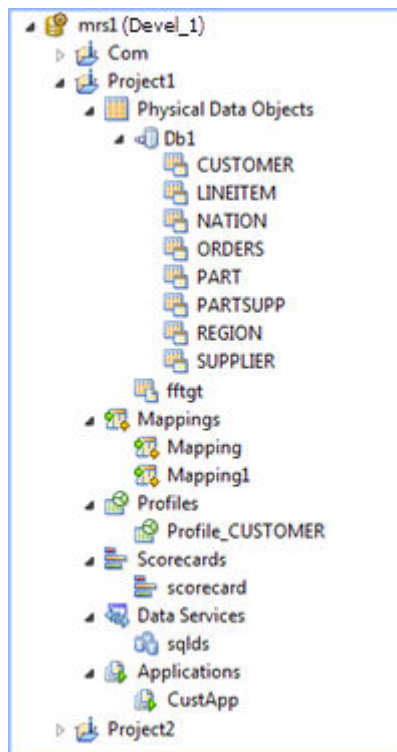
- [モデルリポジトリの概要, 31 ページ](#)
- [Informatica Developer のオブジェクト, 32 ページ](#)
- [リポジトリオブジェクトのロック, 34 ページ](#)
- [バージョン管理されたオブジェクトを使用したチームベース開発, 36 ページ](#)
- [モデルリポジトリへの接続, 41 ページ](#)
- [モデルリポジトリサービスの更新, 41 ページ](#)

モデルリポジトリの概要

モデルリポジトリは、プロジェクトおよびフォルダのメタデータを格納するリレーショナルデータベースです。

モデルリポジトリに接続し、物理データオブジェクト、マッピング、プロファイル、および他のオブジェクトを作成および編集します。アプリケーションにオブジェクトを追加し、エンドユーザーおよびサードパーティのツールでオブジェクトにアクセスできるようにアプリケーションをデプロイします。

以下の図に、**[オブジェクトエクスプローラ]** ビューで mrs1 という名前のモデルリポジトリを開いた状態を示します。



モデルリポジトリサービスでは、モデルリポジトリが管理されます。モデルリポジトリにアクセスするすべてのクライアントアプリケーションおよびアプリケーションサービスは、モデルリポジトリサービスを介して接続します。クライアントアプリケーションには Developer tool および Analyst ツールが含まれます。モデルリポジトリにアクセスする Informatica サービスには、モデルリポジトリサービス、アナリストサービス、およびデータ統合サービスが含まれます。

Developer tool を設定する場合は、モデルリポジトリを追加する必要があります。Developer ツールを開くたびに、モデルリポジトリに接続してプロジェクトおよびフォルダにアクセスします。

オブジェクトを編集するときには、排他的に編集できるように、モデルリポジトリによってそのオブジェクトがロックされます。モデルリポジトリをサードパーティ製のバージョン管理システムと統合することもできます。バージョン管理システムを統合すると、オブジェクトのチェックアウトとチェックイン、オブジェクトのチェックアウトの取り消し、オブジェクトの履歴バージョンの表示と取得を実行できます。

Informatica Developer のオブジェクト

Developer tool の【オブジェクトエクスプローラ】ビューで特定のオブジェクトを作成、管理、または表示できます。

Developer tool では、以下のモデルリポジトリオブジェクトを作成できます。

アプリケーション

データオブジェクト、マッピング、SQL データサービス、Web サービス、ワークフローを含む、デプロイ可能なオブジェクト。アプリケーションの作成、編集、削除を行うことができます。

データサービス

データにアクセスしてそのデータを変換するために実行できる、再利用可能な操作のコレクション。データサービスは、Web サービスを通じてアクセスできる、または SQL クエリを実行する対象となるデータの統合されたモデルを提供します。データサービスの作成、編集、削除を行うことができます。

フォルダ

モデルリポジトリ内のオブジェクトのコンテナ。フォルダを使用してプロジェクト内のオブジェクト体系的に管理したり、フォルダを作成してビジネスニーズに基づいてオブジェクトをグループ化したりできます。フォルダの作成、編集、削除を行うことができます。

論理データオブジェクト

論理データオブジェクトモデル内のオブジェクトであり、企業の論理エンティティを表します。論理データオブジェクトには属性とキーがあり、属性間のリレーションを表します。論理データオブジェクトモデル内の論理データオブジェクトの作成、編集、削除を行うことができます。

論理データオブジェクトマッピング

論理データオブジェクトを 1 つ以上の物理データオブジェクトにリンクするマッピング。論理データオブジェクトマッピングには、トランスフォーメーションロジックを含めることができます。論理データオブジェクトの論理データオブジェクトマッピングの作成、編集、削除を行うことができます。

論理データオブジェクトモデル

論理データオブジェクトを含み、オブジェクト間のリレーションを定義するデータモデル。論理データオブジェクトモデルの作成、編集、削除を行うことができます。

マッピング

データトランスフォーメーションのルールを定義するトランスフォーメーションオブジェクトによりリンクされている入力と出力。マッピングの作成、編集、削除を行うことができます。

マップレット

複数のマッピングで使用したり、ルールとして検証可能なトランスフォーメーション式を含む再利用可能なオブジェクト。マップレットの作成、編集、削除を行うことができます。

操作マッピング

Web サービスクライアントに対して Web サービス操作を実行するマッピング。操作マッピングには、入力トランスフォーメーション、出力トランスフォーメーション、および複数のフォルトトランスフォーメーションを含めることができます。Web サービスの操作マッピングの作成、編集、削除を行うことができます。

物理データオブジェクト

リソースからの読み取り、リソースの検索、またはリソースへの書き込みに使用されるデータの物理的な表現。物理データオブジェクトの作成、編集、削除を行うことができます。

プロファイル

ソースデータ内のパターンを検出するルールが含まれるオブジェクト。データ構造を評価し、必要なタイプの情報がデータカラムに含まれていることを確認するには、プロファイルを実行します。プロファイルの作成、編集、削除を行うことができます。

参照テーブル

参照テーブルには、一連のデータ値の標準バージョンおよび検索する可能性のある値の代替バージョンが含まれます。参照テーブルの表示と削除を行うことができます。

ルール

プロファイルの実行時にソースデータに適用される条件を定義するビジネスロジック。これは、プロファイルで使用する中間ストリームマップレットです。ルールの作成、編集、削除を行うことができます。

スコアカード

ソースカラムの有効値またはプロファイル結果内のルールの出力を視覚的に表したものの。スコアカードの作成、編集、削除を行うことができます。

トランスフォーメーション

データの生成や変更を行ったり、データを渡したりする、マッピング内のリポジトリオブジェクト。各トランスフォーメーションは別の関数を実行します。トランスフォーメーションは再利用可能または再利用不可能にすることができます。トランスフォーメーションの作成、編集、削除を行うことができます。

仮想スキーマ

データベース構造を定義する仮想データベースのスキーマ。SQL データサービスの仮想スキーマの作成、編集、削除を行うことができます。

仮想ストアドプロシージャ

SQL データサービスにおける一連の手順またはデータフローの命令。仮想スキーマの仮想ストアドプロシージャの作成、編集、削除を行うことができます。

仮想テーブル

仮想データベース内のテーブル。仮想スキーマ内の仮想テーブルの作成、編集、削除を行うことができます。

仮想テーブルマッピング

仮想テーブルをターゲットとして含むマッピング。仮想テーブルの仮想テーブルマッピングの作成、編集、削除を行うことができます。

ワークフロー

ビジネスプロセスを定義する一連のイベント、タスク、およびディシジョンを視覚的に表したものの。ワークフローの作成、編集、削除を行うことができます。

オブジェクトプロパティ

プロジェクト、フォルダー、またはモデルリポジトリ内のその他のオブジェクトのプロパティを表示できます。

[全般設定] タブ（**[プロパティ]** ビュー内）には、オブジェクトのプロパティが表示されます。オブジェクトのプロパティには、リポジトリ内のオブジェクトの名前、説明、場所が含まれます。オブジェクトのプロパティには、オブジェクトの作成ユーザー、最終更新ユーザー、およびイベントの発生時刻も含まれます。

オブジェクトのプロパティにアクセスするには、**[Object Explorer]** ビューでオブジェクトを選択して、**[ファイル]** > **[プロパティ]** をクリックします。

リポジトリオブジェクトのロック

モデルリポジトリでは、ユーザーが他の開発者の作業を上書きしないように、オブジェクトがロックされます。モデルリポジトリは、プロジェクトとフォルダを除いて、**[オブジェクトエクスプローラ]** ビューに表示されるすべてのオブジェクトをロックできます。また、このロックは Analyst ツール内で対応するアセットにも影響します。

オブジェクトの編集を開始すると、他のユーザーがそのオブジェクトに対する変更を保存できないように、モデルリポジトリによってそのオブジェクトがロックされます。オブジェクトを保存すると、ロック状態が保持されます。オブジェクトを閉じると、リポジトリではロックが解除されます。

別のユーザーがロックしたオブジェクトを開くと、別のユーザーによってそのオブジェクトがロックされているということが Developer tool によって通知されます。そのオブジェクトは、Analyst ツールまたは

Developer tool でロックされている可能性があります。オブジェクトは、読み取り専用モードで確認するか、別の名前で保存することができます。

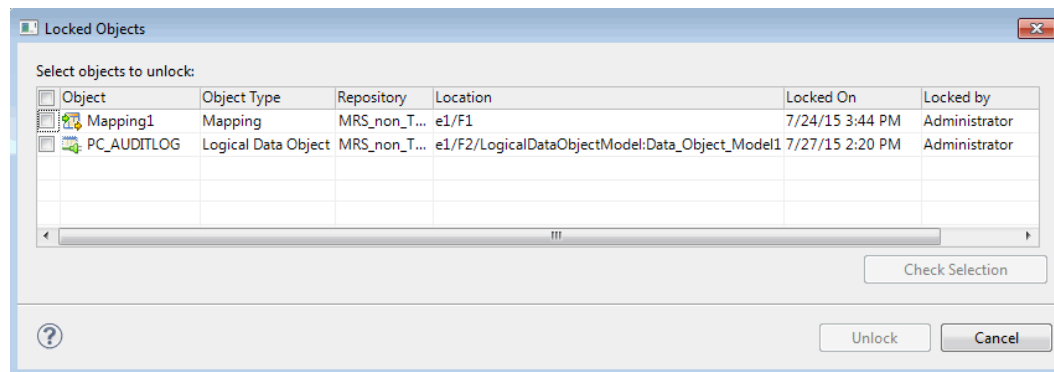
ロック管理

Developer tool が予期せず停止した場合、モデルリポジトリはオブジェクトをロックした状態で保持します。モデルリポジトリに再接続すると、ロックしたオブジェクトを表示できます。オブジェクトの編集を継続するか、オブジェクトのロックを解除することができます。

オブジェクトをロックした開発者が不在で、他の開発者がオブジェクトの開発を任された場合、オブジェクトのロックを解除することが必要になる場合があります。**【ロックされたオブジェクト】** ダイアログボックスでロックされたオブジェクトを表示してロックを解除します。

【ロックされたオブジェクト】 ダイアログボックスを表示するには、**【ファイル】** > **【ロックされたオブジェクト】** をクリックします。

以下の図に、**【ロックされたオブジェクト】** ダイアログボックスを示します。



【ロックされたオブジェクト】 ダイアログボックスでは1つ以上のオブジェクトのロックを解除できます。以下のいずれかのアクションを実行して、ロックを解除するオブジェクトを選択します。

- チェックボックスを使用して1つ以上のオブジェクトを選択する。
- 1つ以上の行を選択して**【選択のチェック】**をクリックし、選択した各オブジェクトのチェックボックスを有効にする。

次に、**【ロック解除】**をクリックして、選択したすべてのオブジェクトのロックを解除します。

ヒント: 作業が中断した場合は、ロックしたオブジェクトを表示し、作業中のオブジェクトを特定することができます。

ロック管理のルールとガイドライン

オブジェクトロックを管理する場合は、以下のルールとガイドラインを考慮してください。

- モデルリポジトリでは、オブジェクトを開くときにそのオブジェクトがロックされることはありません。モデルリポジトリでオブジェクトがロックされるのは、オブジェクトの編集を開始した後だけです。例えば、編集可能なフィールドにカーソルを挿入するときやマッピングオブジェクトに接続するときに、マッピング仕様がロックされます。
- オブジェクトの開発には複数のクライアントツールを使用できます。例えば、1台のマシンでオブジェクトを編集し、その後で別のマシンでそのオブジェクトを開いて編集を続けることができます。最初のマシンに戻る場合は、オブジェクトエディタを閉じてもう一度開くことによってロックを再取得する必要があります。管理特権を持つユーザーが、別のユーザーが開いたオブジェクトのロックを解除するときにも、この同じ原則が適用されます。

- 編集のためにオブジェクトを開いた状態で、モデルリポジトリサービスを再起動した場合は、オブジェクトのロックが解除されます。オブジェクトのロックを再取得するまで、別のユーザーがオブジェクトを開いて編集することができます。オブジェクトのロックを再取得するには、オブジェクトへの変更を保存してから閉じ、次に、編集のためにオブジェクトを再度開きます。
- オブジェクトを含むフォルダを削除する場合、どのオブジェクトの削除も許可されていなければ、フォルダを削除することはできません。例えば、あるオブジェクトに対するロックを所有していないためにそのオブジェクトを削除できない場合、そのオブジェクトとフォルダは残ります。
- 複数の開発者が同時に SQL データサービスオブジェクトのコンテンツを開き、編集することができます。例えば、SQL データサービスを UserA が開いて編集を開始し、次に UserB が同じオブジェクトを開いて編集を開始することができます。UserB が UserA より先にオブジェクトを保存して閉じると、モデルリポジトリは UserA がオブジェクトを保存するまで競合の可能性を UserA に伝えません。この場合、UserA は別の名前で SQL データサービスを保存することで変更を保存できます。
- 管理者は、ロックされたオブジェクトに対する書き込み権限を取り消したり、ロックを別のユーザーに再割り当てしたりできます。この場合、オブジェクトの編集や保存はできません。オブジェクトは別のユーザー名で保存できます。

バージョン管理されたオブジェクトを使用したチームベース開発

チームベース開発は、モデルリポジトリをサードパーティ製のバージョン管理システムと統合するものです。バージョン管理システムでは、複数のオブジェクトバージョンが保存され、各バージョンにバージョン番号が割り当てられます。

オブジェクトバージョンの管理は、Developer tool を使用して行います。オブジェクトのチェックアウトとチェックイン、オブジェクトのバージョン履歴の表示と取得、チェックアウトの取り消しなどのアクションを実行できます。

モデルリポジトリでは、開発チームの他のメンバによって上書きされないようにオブジェクトが保護されます。別のユーザーによってすでにチェックアウトされているオブジェクトを開くと、そのオブジェクトをチェックアウトしたユーザーを特定する通知が送られてきます。チェックアウト済みのオブジェクトは、読み取り専用モードで開くことができるほか、別の名前で保存することもできます。

バージョン管理システムへの接続がアクティブなとき、モデルリポジトリには各オブジェクトの最新バージョンが含まれています。

バージョン管理システムへの接続を失うと、モデルリポジトリはチェックアウトされたオブジェクトの状態を維持します。バージョン管理システムとの接続が遮断している間、ユーザーはオブジェクトのオープン、編集、保存、およびクローズを継続できます。モデルリポジトリによってオブジェクト状態が追跡および維持されます。

接続が復旧した時点で、バージョン管理システムに関連するアクション（オブジェクトのチェックインやオブジェクトのチェックアウトの取り消しなど）を再開できます。接続が遮断していた間にオブジェクトを開いて編集した場合は、モデルリポジトリによってそのオブジェクトがユーザーに対してチェックアウトされます。

注: SQL データサービスオブジェクトはバージョン管理に含まれません。

バージョンされたオブジェクトの管理

モデルリポジトリがバージョン管理システムと統合されている場合は、オブジェクトのバージョンを管理できます。例えば、オブジェクトのチェックアウトとチェックイン、チェックアウトの取り消し、チェックアウトしたオブジェクトの表示などが可能です。

以下のアクションを実行できます。

オブジェクトをチェックアウトする。

オブジェクトをチェックアウトすると、そのオブジェクトは、チェックインするかチェックアウトを取り消すまでチェックアウト状態を維持します。チェックアウトされたオブジェクトは、**【チェックアウトされたオブジェクト】** ビューで表示できます。オブジェクトをチェックアウトするには、最初にオブジェクトを編集するか、**【オブジェクトエクスプローラ】** でオブジェクトを右クリックして **【チェックアウト】** を選択します。

オブジェクトのチェックアウトを取り消す。

チェックアウトを取り消すと、変更は適用されず、バージョン番号やバージョン履歴が増分されずに、オブジェクトがチェックインされます。チェックアウト後にオブジェクトに対して行った変更はすべて失われます。チェックアウトを取り消すには、**【チェックアウトされたオブジェクト】** ビューまたはオブジェクトの右クリックメニューを使用できます。例えば、チェックアウトを取り消すことによって、オブジェクトに加えた変更を削除できます。

注: オブジェクトがチェックアウトされているときにユーザーがオブジェクトを移動し、チェックアウトを取り消すと、オブジェクトは現在の場所に残り、そのバージョン履歴が再開します。チェックアウトを取り消しても、チェックアウト前の場所にはリストアされません。

オブジェクトのチェックイン。

オブジェクトをチェックインすると、バージョン管理システムによってバージョン履歴が更新され、バージョン番号が増分されます。チェックインコメントを最大 4KB の制限内で追加できます。オブジェクトをチェックインするには、**【チェックアウトされたオブジェクト】** ビューまたはオブジェクトの右クリックメニューを使用できます。

注: Perforce バージョン管理システムでは、パスの長さを 260 バイトに制限する Windows の制限のために、ファイルパスの長さが制限される場合があります。この制限を超えるパスを持つファイルをチェックインしようとする、Perforce はエラーを返します。パスの長さが 260 バイトの場合、使用しているシステムの文字セットに応じて、パスの長さが変わります。

オブジェクトを削除する。

バージョンされたオブジェクトを削除するには、まずそのオブジェクトをチェックアウトする必要があります。削除アクションを行う時点でオブジェクトをチェックアウトしていないと、モデルリポジトリによってそのオブジェクトがチェックアウトされ、削除対象としてマークされます。削除アクションを完了するには、そのオブジェクトをチェックインする必要があります。

バージョンされたオブジェクトを削除すると、バージョン管理システムによってすべてのバージョンが削除されます。

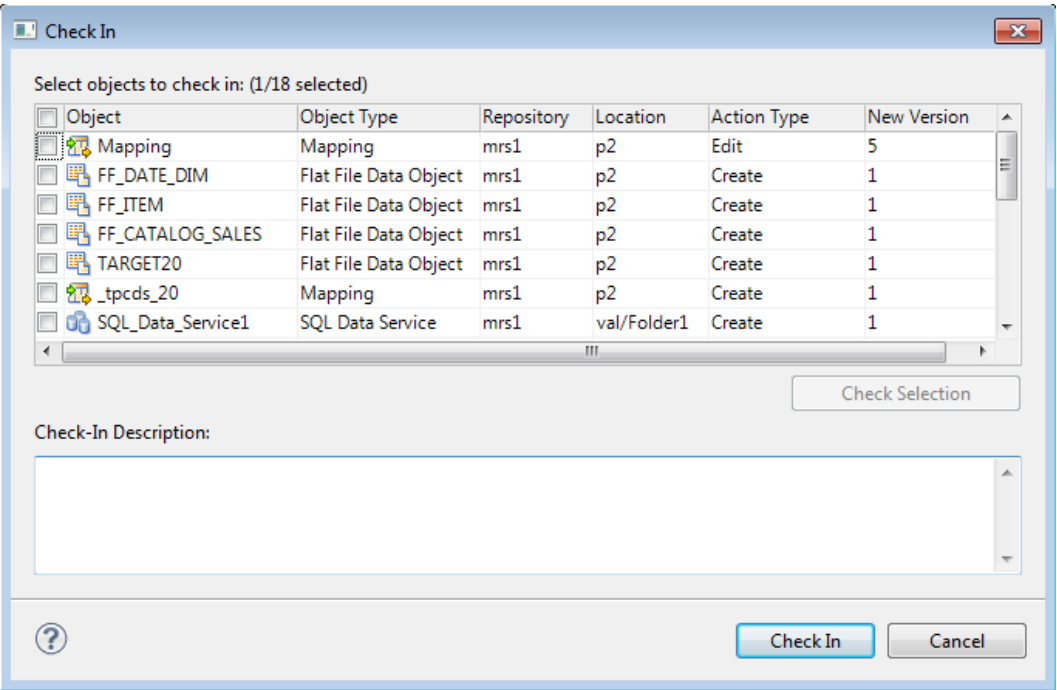
オブジェクトを削除するには、**【チェックアウトされたオブジェクト】** ビューまたはオブジェクトの右クリックメニューを使用できます。

【チェックイン】 ダイアログボックス

【チェックイン】 ダイアログボックスには、チェックアウトしたすべてのオブジェクトが表示されます。

【チェックイン】 ダイアログボックスを使用してオブジェクトをチェックインします。またこれを使用して、削除対象としてマークしたオブジェクトを削除することもできます。

以下の図に、**【チェックイン】** ダイアログボックスを示します。



【アクションタイプ】 列には、各アイテムをチェックインするときにそのアイテムに対してモデルリポジトリサービスが実行するアクションが表示されます。モデルリポジトリサービスは以下のいずれかのアクションタイプを実行します。

アクションタイプ	説明
作成	このオブジェクトは新しいオブジェクトであり、チェックインされたことがありません。これを選択し、 【チェックイン】 をクリックすると、モデルリポジトリサービスは、このオブジェクトをバージョン管理システムにチェックインし、このオブジェクトにバージョン番号 1 を割り当てます。
削除	オブジェクトが削除対象としてマークされます。これを選択し、 【チェックイン】 をクリックすると、モデルリポジトリサービスはこのオブジェクトを削除します。
編集	オブジェクトが編集対象としてマークされます。これを選択し、 【チェックイン】 をクリックすると、モデルリポジトリサービスは、バージョン管理システム内のバージョンを新しいバージョンに置き換えます。
移動	オブジェクトをチェックアウトすると、オブジェクトが 1 つのモデルリポジトリプロジェクトまたはフォルダから別のモデルリポジトリプロジェクトまたはフォルダに移動します。これを選択し、 【チェックイン】 をクリックすると、モデルリポジトリサービスは、このオブジェクトを新しい場所のバージョン管理システムにチェックインします。

【場所】 列には、オブジェクトの現在の場所が表示されます。オブジェクトを移動した場合、**【アクションタイプ】** は **【移動】** です。

ヒント: **【バージョン履歴】** ビューでは、移動したオブジェクトの以前の場所を確認できます。

【新しいバージョン】 列には、チェックイン完了後のオブジェクトのバージョン番号が表示されます。

【チェックイン】ダイアログボックスでは、1つ以上のオブジェクトをチェックインできます。以下のいずれかのアクションを実行して、チェックインするオブジェクトを選択します。

- チェックボックスを使用して1つ以上のオブジェクトを選択する。
- 1つ以上の行を選択して【選択のチェック】をクリックし、選択した各オブジェクトのチェックボックスを有効にする。

オプションで、【チェックインの説明】テキストペインで説明を追加できます。次に【チェックイン】をクリックすると、選択したすべてのオブジェクトをチェックインできます。

バージョンされたオブジェクトの削除

バージョンされたオブジェクトを削除する場合は、削除するオブジェクトに印を付けてチェックインします。

1. 【オブジェクトエクスプローラ】ビューまたは【チェックアウトされたオブジェクト】ビューでオブジェクトを右クリックし、【削除】を選択します。
アクション属性が【編集】から【削除】に変わります。
2. オブジェクトを選択して【チェックイン】を選択します。
モデルリポジトリからオブジェクトが削除されます。

オブジェクトの履歴バージョン

バージョン管理システムによって、オブジェクトのチェックインのたびにオブジェクトのコピーが保存されます。

バージョン管理システムは、オブジェクトのバージョンを保存し、そのバージョンにバージョン番号を割り当てます。最も大きな番号が最新のバージョンになります。

【バージョン履歴】ビューを使用してオブジェクトのバージョン履歴を表示し、読み取り専用ビューで履歴バージョンを開いて、オブジェクトの履歴バージョンをリストアします。オブジェクトの履歴バージョンをリストアし、以前の機能をリストアすることができます。

オブジェクトの履歴バージョンをリストアすると、そのバージョンが最新バージョンになります。モデルリポジトリは、オブジェクトの現行バージョンを履歴バージョンに置き換え、オブジェクトをチェックアウトします。オブジェクトがエディタで開かれると、Developer tool によってオブジェクトがリストアしたバージョンに更新されます。

オブジェクトの履歴バージョンのリストア

バージョン管理システムからオブジェクトの履歴バージョンを表示および取得できます。

1. 【オブジェクトエクスプローラ】のオブジェクトを右クリックし、【バージョン履歴の表示】を選択します。
【改訂履歴】ビューが表示されます。
2. 履歴バージョンを表示するには、【改訂履歴】に一覧表示されるバージョンの1つを右クリックし、【バージョンの表示】を選択します。
Developer tool では、オブジェクトの読み取り専用バージョンが開きます。バージョン番号はエディタウィンドウのタイトルバーに表示されます。
3. 履歴バージョンをリストアするには、【改訂履歴】に一覧表示されるバージョンの1つを右クリックし、【バージョンをリストア】を選択します。
オブジェクトがエディタで表示されます。オブジェクトを編集したり、編集せずにチェックインしたりすることができます。

4. オブジェクトをチェックインします。

リストアされたオブジェクトのバージョンは、モデルリポジトリおよびバージョン管理システムで最新バージョンになります。

[チェックアウトされたオブジェクト] ビュー

[チェックアウトされたオブジェクト] ビューには、チェックアウトしたオブジェクトがすべて一覧表示されます。

[チェックアウトされたオブジェクト] ビューでは、以下のアクションを実行できます。

- オブジェクトのチェックアウトの取り消し
- オブジェクトのチェックイン

これらのどちらかを実行するには、オブジェクトを右クリックし、アクションを選択します。

[アクションタイプ] 列で削除対象としてマークされたオブジェクトを削除するには、オブジェクトを右クリックし、**[チェックイン]** を選択します。

[バージョン履歴] ビュー

[バージョン履歴] ビューには、選択したオブジェクトのバージョン履歴が表示されます。チェックインコメントを確認し、オブジェクトチェックインに関するユーザー情報を表示できます。

このビューには以下の情報が表示されます。

- バージョン。バージョンが保存された順序を表示する整数。バージョン管理システムでは、最新のチェックインバージョンに最も大きな番号を割りてます。
- 名前。オブジェクトの名前。
- 場所。モデルリポジトリ内のオブジェクトへのパス。
- アクションタイプ。オブジェクトのチェックインの際にユーザーが実行したアクション。オブジェクトのチェックインでは、オブジェクトを追加、編集、または削除することができます。
- ユーザー。オブジェクトをチェックインしたチームメンバーのユーザー ID。
- バージョン日付。ユーザーがオブジェクトをチェックインした日時。
- バージョンの説明。選択したバージョンのチェックインコメント。

バージョン履歴を表示するには、**[オブジェクトエクスプローラ]** のオブジェクトを右クリックし、**[バージョン履歴の表示]** を選択します。

チームベース開発のトラブルシューティング

チームベースの開発に関連する機能を使用する場合は、以下のトラブルシューティングのヒントを考慮してください。

Perforce バージョン管理システムが一部のオブジェクトのチェックインに失敗し、**オブジェクトパス名が長すぎる**というエラーが表示されます。

ファイルパスの文字数に関する Windows OS の制限のため、長いパスおよびファイル名を持つモデルリポジトリオブジェクトをチェックインしようとする場合、失敗します。「送信が強制終了されました」という Perforce エラーメッセージが表示され、ファイルパスが内部の長さの制限を超えたことが示されます。

この問題を回避するには、モデルリポジトリ内のプロジェクト名、フォルダ名、オブジェクト名の長さを制限します。各インスタンスに短い名前を使用することで、オブジェクトパス名の合計文字数を制限することができます。

モデルリポジトリへの接続

Developer ツールを開くたびに、モデルリポジトリに接続してプロジェクトおよびフォルダーにアクセスします。モデルリポジトリに接続するときに、接続情報を入力して、モデルリポジトリを管理するモデルリポジトリサービスを含むドメインにアクセスします。

1. **[Object Explorer]** ビューで、モデルリポジトリを右クリックして **[接続]** をクリックします。
[リポジトリへの接続] ダイアログボックスが表示されます。

2. ドメインユーザー名とパスワードを入力します。

3. 名前空間を選択します。

4. **[OK]** をクリックします。

Developer ツールがモデルリポジトリに接続します。Developer ツールにリポジトリ内のプロジェクトが表示されます。

モデルリポジトリサービスの更新

モデルリポジトリで新しいオブジェクトと更新されたオブジェクトを確認するために、モデルリポジトリサービスを更新できます。

モデルリポジトリサービスの更新は、他のユーザーが Developer ツールまたは Analyst ツールのオブジェクトの追加、削除、更新を行い、それらの変更をモデルリポジトリに保存し終わってから行ってください。モデルリポジトリサービスの更新を行った場合、Developer tool で開いているエディタが閉じることはありません。

モデルリポジトリサービスを更新するには、**[Object Explorer]** ビューの **[モデルリポジトリサービス]** を右クリックし、**[更新]** を選択します。

第 3 章

Informatica Developer での検索

この章では、以下の項目について説明します。

- [Informatica Developer での検索の概要, 42 ページ](#)
- [モデルリポジトリでの検索, 42 ページ](#)
- [ビジネス用語集の検索, 43 ページ](#)
- [エディタでの検索, 45 ページ](#)

Informatica Developer での検索の概要

Informatica Developer で検索を行い、オブジェクトの検索とビジネス用語のルックアップを行えます。

モデルリポジトリを検索して、保存されたオブジェクトやオブジェクトプロパティの最新版を見つけることができます。Developer ツールからビジネスグロッサリデスクトップにアクセスして Developer ツールのオブジェクト名をビジネス用語として検索できます。エディタ内のオブジェクト、ポート、グループ、式、属性などを検索することもできます。

モデルリポジトリでの検索

モデルリポジトリでオブジェクトやオブジェクトプロパティを検索できます。

検索クエリを作成して検索結果をフィルタリングすることができます。検索結果を確認し、結果からオブジェクトを選択してコンテンツを表示することができます。検索結果は【検索】ビューに表示されます。検索で 2048 個を超えるオブジェクトが見つかった場合には、結果は表示できません。2048 を超える数のオブジェクトが結果に含まれるために検索が失敗したときは、検索条件に一致するオブジェクトの数が少なくなるように検索オプションを変更します。

以下の表に、オブジェクトの検索に使用可能な検索オプションを一覧表示します。

検索オプション	説明
含まれるテキスト	検索対象のオブジェクトまたはプロパティ。正確な文字列を入力するか、ワイルドカードを使用します。大文字と小文字は区別されません。
名前	名前を含む 1 つ以上のオブジェクト。正確な文字列を入力するか、ワイルドカードを使用します。大文字と小文字は区別されません。

検索オプション	説明
タグ	タグを使用する 1 つ以上のオブジェクト。正確な文字列を入力するか、ワイルドカードを使用します。大文字と小文字は区別されません。
検索	検索対象の 1 つ以上のオブジェクトタイプ。
スコープ	ワークスペースまたは選択したオブジェクトを検索します。

モデルリポジトリでの検索では、ポートと動的ポートは返されますが、生成されたポートは返されません。

注: マッピングを検索すると、検索パターンに似た名前を持つマップレットが検索結果に表示されます。

モデルリポジトリサービスでは、検索エンジンを使用して、モデルリポジトリ内のメタデータのインデックスを作成します。メタデータのインデックスを適切に作成するために、検索エンジンでは、インデックスを作成するメタデータの言語に適した検索アナライザが使用されます。Developer tool は、検索エンジンを使用してモデルリポジトリオブジェクトの検索を行います。オブジェクトを検索する前に保存する必要があります。

検索はさまざまな言語で実行できます。別の言語で検索する場合は、管理者が検索アナライザを変更して、その検索アナライザを使用するようにモデルリポジトリを設定する必要があります。

注: モデルリポジトリを更新すると、検索ビュー内の検索結果は表示されなくなります。更新されたモデルリポジトリ内に結果を表示するには、検索をもう一度行う必要があります。

オブジェクトとプロパティの検索

モデルリポジトリ内のオブジェクトおよびプロパティを検索します。

1. **【検索】** > **【オブジェクト検索】** の順にクリックします。
【検索】 ダイアログボックスが表示されます。
2. 検索対象のオブジェクトまたはプロパティを入力します。必要に応じて、ワイルドカード文字を含めます。
3. オブジェクト内のプロパティを検索する場合、必要に応じて 1 つまたは複数の名前またはタグをカンマで区切って入力することができます。
4. 必要に応じて、検索するオブジェクトタイプを選択します。
5. ワークスペースの中から検索するか、選択したオブジェクトの中から検索するかを選択します。
6. **【検索】** をクリックします。
検索結果が **【検索】** ビューに表示されます。
7. **【検索】** ビューで、オブジェクトをダブルクリックしてエディタで開きます。

ビジネス用語集の検索

Developer ツールのオブジェクト名の意味を、ビジネス用語として Business Glossary Desktop でルックアップし、そのビジネス要件と現在の実装を理解します。

ビジネス用語集とは、ビジネスユーザー向けに、ビジネス言語を使用して概念を定義した用語の集まりです。ビジネス用語は、概念のビジネス上の定義および使用方法を説明したものです。Business Glossary Desktop は、ビジネス用語集をホストする Metadata Manager サービスに接続するクライアントです。Business Glossary Desktop を使用して、ビジネス用語集内のビジネス用語をルックアップします。

Business Glossary Desktop がご使用のマシンにインストールされている場合は、Developer ツールでオブジェクトを選択し、ホットキーか [検索] メニューを使用して、ビジネス用語集でオブジェクトの名前をルックアップします。オブジェクトの名前をルックアップするには、[Object Explorer] ビューなど、Developer tool のビューを使用します。カラム、プロファイル、トランスフォーメーションポートの名前をルックアップするにはエディタを使用します。

例えば、開発者が Developer ツール内の Sales_Audit データオブジェクトに対応するビジネス用語をビジネス用語集で検索するとします。開発者は、そのビジネス用語の詳細を確認して、ビジネス要件と、Developer ツールにおける Sales_Audit オブジェクトの現在の実装を理解したいと考えています。これは、そのデータオブジェクトの意味や、オブジェクトへの実装が必要な可能性がある変更を把握ことができます。

ビジネス用語集デスクトップのルックアップ

ビジネス用語集デスクトップは、ビジネス用語集でオブジェクト名をルックアップし、オブジェクト名に一致するビジネス用語を返します。

オブジェクト名にハイフン、アンダースコア、大文字が含まれていると、ビジネス用語集デスクトップでは、名前が 2 つに分割されます。

例えば、開発者が Sales_Audit という名前のデータオブジェクトを検索する場合、ビジネス用語集デスクトップでは検索ボックスに Sales_Audit と表示されますが、Sales と Audit に名前を分割して両方のビジネス用語を検索します。

ビジネス用語のルックアップ

ビジネス要件と現在の実装を理解するため、Developer ツールのオブジェクト名をビジネス用語としてビジネス用語集デスクトップでルックアップします。

ビジネス用語集デスクトップは、使用するマシンにあらかじめインストールしておく必要があります。

1. オブジェクトを選択します。
2. ホットキーを使用するか、[検索] メニューを使用するかを選択して、ビジネス用語集デスクトップを開きます。
 - ホットキーを使用する場合は、以下のホットキーの組み合わせを使用します。
CTRL+Shift+F
 - [検索] メニューを使用する場合は、[検索] > [Business Glossary] をクリックします。

ビジネス用語集デスクトップが表示され、オブジェクト名に一致するビジネス用語が表示されます。

ビジネス用語のルックアップ用にホットキーをカスタマイズ

ホットキーをカスタマイズして、ビジネス用語集デスクトップを開くキーの組み合わせを変更します。

1. Developer ツールのメニューから、[ウィンドウ] > [設定] > [全般] > [キー] をクリックします。
2. [Business Glossary の検索] をコマンドのリストで見つかり、検索したりするには、以下のいずれかを選択します。
 - 検索してキーを探すには、検索ボックスに「ビジネス用語集検索」と入力します。
 - スクロールしてキーを探すには、[コマンド] カラムでスクロールして [Business Glossary の検索] コマンドを見つけます。
3. [Business Glossary の検索コマンド] をクリックします。
4. [コマンドのバインド解除] をクリックします。
5. [バインディング] フィールドで、キーの組み合わせを入力します。

6. **【適用】** をクリックしてから、**【OK】** をクリックします。

エディタでの検索

エディタ内で開いているオブジェクト、ポート、グループ、式、属性を検索できます。Developer ツールで開いているエディタ内のオブジェクトがハイライトされます。オブジェクトをモデルリポジトリに置く必要はありません。

エディタの下に検索フィールドを表示するには、**【編集】** > **【検索/置換】** を選択します。オブジェクトを検索するには、検索対象オブジェクトの文字列および型を入力します。検索可能なオブジェクトタイプは、エディタによって異なります。オブジェクトのタイプを指定しない場合は、トランスフォーメーション内の検索文字列が検索されます。

ポート、カラム、または属性を検索する場合、データ型も選択できます。例えば、文字列「_ID」を含む名前を持つ integer 型または bigint 型のポートを検索できます。

以下の表に、各エディタで検索可能なオブジェクトのタイプを示します。

エディタ	オブジェクトタイプ
マッピング	マッピングオブジェクト、式、グループ、ポート
マップレット	マップレットオブジェクト、式、グループ、ポート
論理データオブジェクトモデル	論理データオブジェクトおよび属性
物理データオブジェクトの読み取りまたは書き込みマッピング	オブジェクトおよびカラムのマッピング
SQL データサービス	仮想テーブルおよび属性
仮想ストアドプロシージャ	トランスフォーメーション、式、グループ、ポート
仮想テーブルマッピング	仮想テーブルのマッピングオブジェクト、式、グループ、ポート
Web サービス操作のマッピング	Web サービス操作のマッピングオブジェクト、式、グループ、ポート
ワークフロー	ワークフローオブジェクト
注: モデルリポジトリには生成されたポートの名前は格納されないため、エディタでポートを検索することはできません。	

検索文字列が見つかったと、そのオブジェクトの場所が表示されます。また、検索文字列が出現するオブジェクトもハイライトされます。検索文字列がマッピングエディタ内のアイコン化したトランスフォーメーションに含まれている場合は、アイコン化したトランスフォーメーションがハイライトされます。

以下のオプションを選択して、検索の結果をナビゲートすることができます。

- 次の一致。検索文字列の次のオカレンスを検索します。
- 直前の一致。検索文字列の直前のオカレンスを検索します。
- すべてハイライト。検索文字列のすべてのオカレンスをハイライトします。

- アイコン化したトランスフォーメーションの展開。検索文字列を含む、アイコン化したすべてのトランスフォーメーションを展開します。

第 4 章

接続

この章では、以下の項目について説明します。

- [接続の概要, 47 ページ](#)
- [\[接続エクスプローラ\] ビュー, 49 ページ](#)
- [接続管理, 50 ページ](#)
- [接続の切り替え, 52 ページ](#)
- [サードパーティ製 JDBC ドライバ, 56 ページ](#)

接続の概要

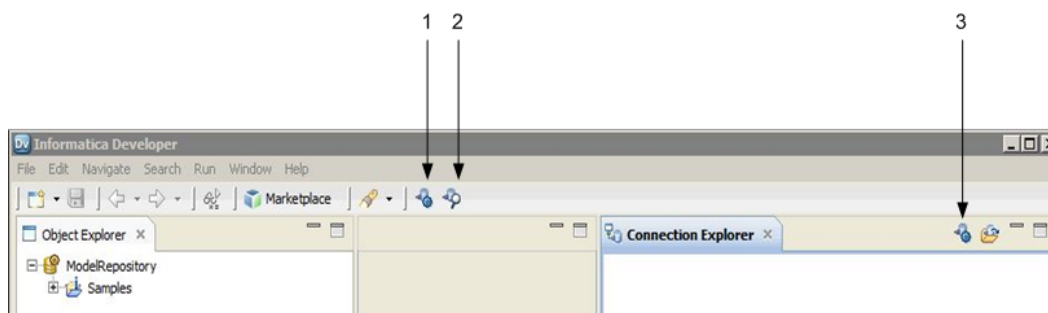
接続は、ドメイン環境設定リポジトリ内の接続を定義するリポジトリオブジェクトです。

接続を作成することで、データオブジェクトのインポート、データのプレビュー、データのプロファイリング、およびマッピングの実行が行えます。Developer ツールは、データオブジェクトをインポートする際に接続を使用します。データ統合サービスは、データのプレビュー、マッピングの実行、または Web サービスの使用時に接続を使用します。

注: Developer tool は、フラットファイルデータオブジェクトのインポートや、フラットファイルデータのプレビュー、読み取り、または書き込みに接続を使用しません。

Developer ツールは接続をドメイン環境設定リポジトリに格納します。Developer ツールで作成した接続は、Analyst ツールや Administrator ツールで使用できます。

【設定】 ダイアログボックスまたは **[接続エクスプローラ]** ビューで接続を作成および管理します。



1. 接続の作成
2. 接続の表示
3. [接続の作成] - [接続エクスプローラ] ビュー

接続を作成したら、その接続に対して以下のアクションを実行できます。

接続を編集する。

接続名および説明を変更できます。接続の詳細（ユーザー名、パスワード、および接続文字列など）の編集もできます。

データ統合サービスは、接続 ID で各接続を識別します。したがって、接続名を変更することができます。接続の名前を変更すると、Developer ツールは、その接続を使用しているオブジェクトを更新します。

デプロイされたアプリケーションおよびパラメータファイルは、接続 ID ではなく名前で接続を識別します。したがって、接続の名前を変更する場合、その接続を使用するすべてのアプリケーションをデプロイしなおす必要があります。さらに、接続パラメータを使用するパラメータファイルもすべて更新する必要があります。

接続をコピーする。

接続をコピーして、別の接続に類似の接続を作成できます。例えば、ユーザー名とパスワードのみが異なる 2 つの Oracle 接続を作成できます。

接続を削除する。

接続を削除すると、その接続を使用するオブジェクトは無効になります。接続を誤って削除した場合、削除した接続と同じ接続 ID で接続を再作成できます。

接続リストを更新する。

接続リストを更新して、ドメインの接続の最新のリストを確認できます。ユーザーが Administrator ツールまたは Analyst ツールで接続を追加、削除、または名前変更した場合は、接続リストを更新します。

接続タイプ

接続を使用して、データソースに対してデータの読み取りと書き込みを実行できます。

次のタイプのデータソースに対して接続を作成および管理できます。

- Adabas データベース
- Amazon Redshift データベース
- AzureBlob
- Datasift ストリーム
- Facebook
- Greenplum データベース
- HBase
- Hadoop ファイルシステム(HDFS)
- Hive データベース
- Web サービスへの HTTP 接続
- IBM DB2 データベース
- IBM DB2 for i5/OS
- IBM DB2 for z/OS
- IMS データベース
- JD Edwards Enterprise One
- JDBC

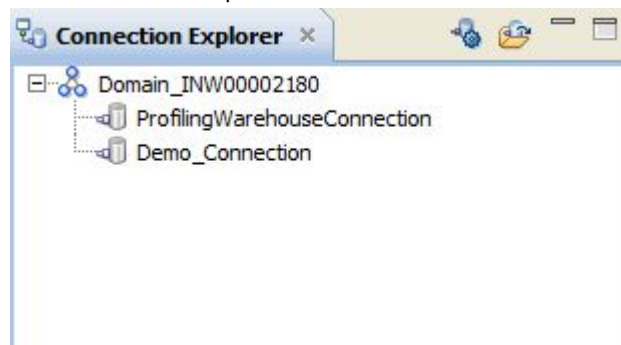
- LDAP
- LinkedIn
- Microsoft Azure SQL Data Warehouse
- Microsoft SQL Server データベース
- Netezza データベース
- OData Web 接続
- ODBC
- Oracle データベース
- Salesforce オブジェクト
- SAP データソース
- シーケンシャルデータソース
- Teradata テーブル
- Tableau
- Twitter および Twitter ストリーム
- VSAM データテーブル
- Web コンテンツ-Kapow Katalyst
- Web サービス

各データソースのサポート対象バージョンについては、[Informatica Product Availability Matrix](#) を参照してください。

[接続エクスプローラ] ビュー

[接続エクスプローラ] ビューを使用して、リレーショナルデータベースまたは非リレーショナルデータベースの接続を表示し、リレーショナルまたは非リレーショナルのデータオブジェクトを作成します。

以下の図に、Developer ツールでの **[接続エクスプローラ]** ビューを示します。



[接続エクスプローラ] ビューでは、以下のタスクを実行できます。

- ビューに接続を追加します。**[接続の選択]** ボタンをクリックして 1 つ以上の接続を選択し、**[接続エクスプローラ]** ビューに追加します。

- リレーショナルまたは非リレーショナルのデータベースへの接続。データベースを右クリックして、**[接続]** をクリックします。
- リレーショナルまたは非リレーショナルデータベースの接続解除。データベースを右クリックして、**[切断]** をクリックします。
- リレーショナルデータオブジェクトの作成。リレーショナルデータベースに接続し、テーブルが表示されるようにデータベースを拡大します。テーブルを右クリックし、**[プロジェクトに追加]** をクリックして **[新しいリレーショナルデータオブジェクト]** ダイアログボックスを開きます。
- 非リレーショナルデータオブジェクトの作成。非リレーショナルデータベースに接続してから、データマップが表示されるようにデータベースを拡大します。データマップを右クリックし、**[プロジェクトに追加]** をクリックして **[新しい非リレーショナルデータオブジェクト]** ダイアログボックスを開きます。
- 接続の更新。接続を右クリックして、**[更新]** をクリックします。
- デフォルトスキーマのみ表示。接続を右クリックして、**[デフォルトのスキーマのみを表示]** をクリックします。デフォルトでは有効になっています。
- **[接続エクスプローラ]** ビューからの接続の削除。接続はモデルリポジトリに残ります。接続を右クリックして、**[削除]** をクリックします。

接続管理

[設定] ダイアログボックスまたは **[接続エクスプローラ]** ビューで接続を作成および管理します。

接続の作成

Hadoop クラスタ、データベース、エンタープライズアプリケーション、ファイルシステム、非リレーショナルデータベース、NoSQL データベース、ソーシャルメディアアプリケーション、または Web サービスにアクセスするための接続を作成します。物理データオブジェクトのインポート、データのプレビュー、データのプロファイリング、またはマッピングを行う前に、接続を作成します。

1. **[ウィンドウ]** > **[設定]** をクリックします。
2. 作成する接続のタイプを選択します。
 - 非 Web サービス接続を選択するには、**[Informatica]** > **[接続]** を選択します。
 - Web サービス接続を選択するには、**[Informatica]** > **[Web サービス]** > **[接続]** を選択します。
3. **[使用可能な接続]** リストでドメインを展開します。
4. **[使用可能な接続]** リストで接続タイプを選択して、**[追加]** をクリックします。
[新しい<接続タイプ>接続] ダイアログボックスが表示されます。
5. 次の情報を入力します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /

プロパティ	説明
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明（オプション）。
場所	接続が存在するドメイン。
タイプ	Oracle、Twitter、または Web サービスなど、特定の接続タイプ。

6. **【次へ】** をクリックします。
7. 接続のプロパティを設定します。
8. **【テスト接続】** をクリックして、入力した接続のプロパティが正しく、データベース、アプリケーション、ファイルシステム、または URI に接続できることを確認します。
9. **【完了】** をクリックします。

接続を作成したら、**【接続エクスプローラ】** ビューに追加できます。

接続の編集

接続名、説明、および接続プロパティを編集できます。

1. **【ウィンドウ】** > **【設定】** をクリックします。
2. 編集する接続のタイプを選択します。
 - 非 Web サービス接続を選択するには、**【Informatica】** > **【接続】** を選択します。
 - Web サービス接続を選択するには、**【Informatica】** > **【Web サービス】** > **【接続】** を選択します。
3. **【使用可能な接続】** リストでドメインを展開します。
4. **【使用可能な接続】** で接続を選択して、**【編集】** をクリックします。
【接続の編集】 ダイアログボックスが表示されます。
5. 必要に応じて、接続名と説明を編集します。
注: 接続名を変更する場合、その接続を使用するすべてのアプリケーションを再デプロイする必要があります。接続パラメータを使用するパラメータファイルもすべて更新する必要があります。
6. **【次へ】** をクリックします。
7. 必要に応じて、接続のプロパティを編集します。
8. **【テスト接続】** をクリックして、入力した接続のプロパティが正しく、データベースに接続できることを確認します。
9. **【OK】** をクリックして、**【接続の編集】** ダイアログボックスを閉じます。
10. **【OK】** をクリックして、**【設定】** ダイアログボックスを閉じます。

接続のコピー

ドメイン内、またはドメイン間で接続をコピーできます。

1. **【ウィンドウ】** > **【設定】** をクリックします。
2. コピーする接続のタイプを選択します。

- 非 Web サービス接続を選択するには、[Informatica] > [接続] を選択します。
 - Web サービス接続を選択するには、[Informatica] > [Web サービス] > [接続] を選択します。
3. [使用可能な接続] リストでドメインを展開します。
 4. [使用可能な接続] で接続を選択して、[コピー] をクリックします。
[接続のコピー] ダイアログボックスが表示されます。
 5. 接続名および ID を入力し、ドメインを選択します。
名前と ID は、ドメイン内で一意である必要があります。
 6. [OK] をクリックして、[接続のコピー] ダイアログボックスを閉じます。
 7. [OK] をクリックして、[設定] ダイアログボックスを閉じます。

接続の削除

[設定] ダイアログボックスで接続を削除すると、モデルリポジトリから接続が削除されます。

1. [ウィンドウ] > [設定] をクリックします。
2. 削除する接続のタイプを選択します。
 - 非 Web サービス接続を選択するには、[Informatica] > [接続] を選択します。
 - Web サービス接続を選択するには、[Informatica] > [Web サービス] > [接続] を選択します。
3. [使用可能な接続] リストでドメインを展開します。
4. [使用可能な接続] で接続を選択して、[削除] をクリックします。
5. [OK] をクリックして、[設定] ダイアログボックスを閉じます。

接続リストの更新

ドメインの接続の最新のリストを確認するには、接続リストを更新します。

1. [ウィンドウ] > [設定] をクリックします。
2. 更新する接続のタイプを選択します。
 - 非 Web サービス接続を選択するには、[Informatica] > [接続] を選択します。
 - Web サービス接続を選択するには、[Informatica] > [Web サービス] > [接続] を選択します。
3. [使用可能な接続] リストでドメインを選択します。
4. [更新] をクリックします。
5. [使用可能な接続] リストでドメインを展開して、接続の最新のリストを表示します。
6. [OK] をクリックして、[設定] ダイアログボックスを閉じます。

接続の切り替え

リレーショナルデータオブジェクトまたはカスタマイズデータオブジェクトの接続を切り替えて、異なるリレーショナルデータベース接続を使用できます。複数のデータオブジェクトの接続を同時に切り替えることもで

きます。接続を切り替えると、新しい接続を使用するために各マッピングを更新する必要がないため、時間と労力が節約されます。

接続を切り替えた後に、Developer tool はデータオブジェクトに基づいてすべての読み取り、書き込み、およびルックアップトランスフォーメーションのデータオブジェクトの接続の詳細を更新します。Developer tool は、新しい接続が参照するデータベースに基づいてデータオブジェクトのデータベースタイプも更新します。

次のいずれかの接続タイプに接続を切り替えることができます。

- IBM DB2
- Microsoft SQL Server
- ODBC
- Oracle

接続を切り替えるとき、Developer tool はメタデータの互換性を検証しません。このため、接続を切り替える前に、新しい接続が参照するデータベースに、接続を切り替えるデータオブジェクトと同じカラムとメタデータを持つテーブルが含まれることを確認する必要があります。新しく参照されるデータベースに同じカラムとメタデータが含まれていない場合、データの消失またはデータの矛盾が発生する可能性があります。

例

Developer tool で Oracle リレーショナルデータオブジェクトを作成して、書き込みトランスフォーメーションとして複数のマッピングに追加しました。

Oracle データベースを IBM DB2 データベースに移行します。Developer tool で既存のマッピングを更新して、データを IBM DB2 データベースに書き込みます。

各マッピングで書き込みトランスフォーメーションが基にしているデータオブジェクトを置き換える代わりに、Oracle データオブジェクトの接続を IBM DB2 の接続に切り替えることができます。新しい接続はデータを書き込む IBM DB2 データベースを指しています。Oracle リレーショナルデータオブジェクトを含むマッピングを実行すると、データ統合サービスは新しい IBM DB2 接続を使用してマッピングを実行します。

接続を切り替える前に

接続を切り替える前に、以下の要件が満たされていることを確認します。

- 接続を切り替えるデータオブジェクトに対する書き込み権限がある。
- 新しい接続が参照するデータベースに、接続を切り替えるデータオブジェクトと同じカラムとメタデータを持つテーブルが含まれる。テーブル内のカラムの精度とスケールがデータオブジェクトのそれらよりも低い場合、データの損失や不整合が発生する可能性があります。
- 接続を切り替えるデータオブジェクトに、保存されていない変更が含まれない。保存されていない変更が含まれる場合、Developer tool は接続を切り替えません。
- 接続を切り替えるカスタマイズデータオブジェクトに、異なるデータベースタイプのリレーショナルデータオブジェクトが含まれない。含まれる場合、接続を切り替えるときに Developer tool はエラーを表示しません。ただし、そのカスタマイズデータオブジェクトを使用してマッピングを実行した場合、マッピングは失敗します。
- 元の接続と新しい接続の参照先のテーブル名が大文字小文字を含め正確に一致する。引用識別子および大文字小文字の混在したテーブル名が接続でサポートされる場合、Developer tool は大文字小文字を区別してテーブル名を処理します。
- 接続を切り替えるデータオブジェクトがバージョン管理されたリポジトリに保存されている場合、そのオブジェクトがチェックアウトされている。

接続の切り替え

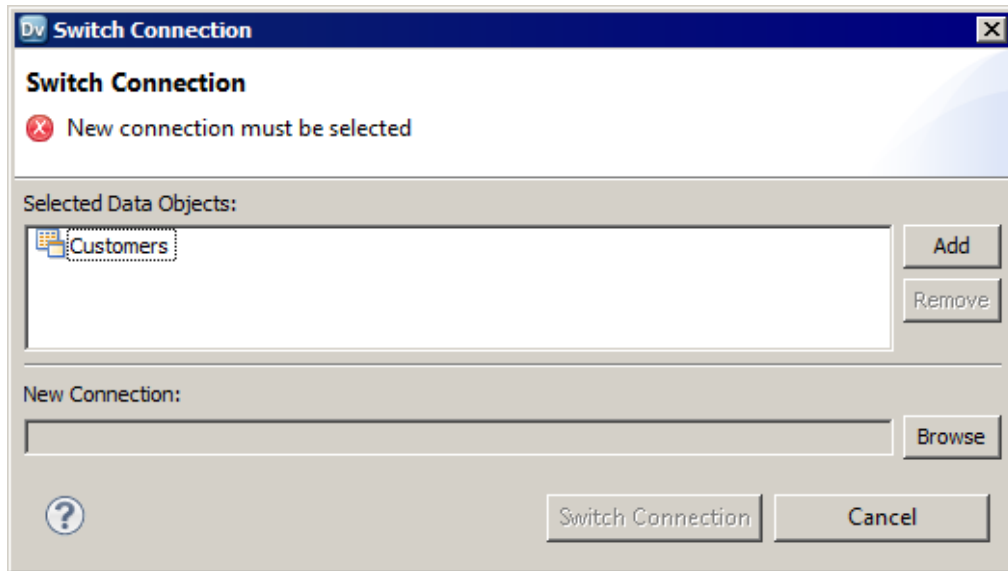
リレーショナルデータオブジェクトまたはカスタマイズデータオブジェクトの接続を切り替えて別のリレーショナルデータベース接続を使用し、同時に既存のマッピングで新しい接続が使用されるように更新できます。

1. **【オブジェクトエクスプローラ】** ビューで、接続を切り替えるデータオブジェクトを右クリックします。
1つのプロジェクトまたは複数のプロジェクトの別々のフォルダにある複数のデータオブジェクトを選択できます。

接続を選択して、その接続を使用するすべてのリレーショナルデータオブジェクトおよびカスタマイズデータオブジェクトの接続を同時に切り替えることもできます。

2. **【接続の切り替え】** をクリックします。

【接続の切り替え】 ダイアログボックスが表示され、選択したデータオブジェクトが表示されます。



3. **【追加】** をクリックして、接続を切り替えるデータオブジェクトのリストを更新します。

オブジェクトを削除するには、オブジェクトを選択して **【削除】** をクリックします。

4. **【新しい接続】** フィールドの横にある **【参照】** をクリックします。

Developer tool で、データオブジェクトに使用できる接続が表示されます。

5. データオブジェクトに使用する新しい接続を選択して、**【OK】** をクリックします。

6. **【接続の切り替え】** をクリックします。

新しい接続が参照するデータベースに、接続を切り替えるデータオブジェクトと同じカラムとメタデータを持つテーブルが含まれることを確認するように求めるメッセージが表示されます。

7. **【OK】** をクリックして、データオブジェクトの接続を切り替えます。

接続が正常に切り替えられたことを示すメッセージが表示されます。Developer tool は、そのデータオブジェクトに関連付けられているすべての Developer tool オブジェクトのデータオブジェクトの接続の詳細およびデータベースタイプを更新します。そのデータオブジェクトを含むマッピングを実行すると、データ統合サービスが新しい接続を使用します。

接続を切り替えた後に

接続を切り替えた後、データオブジェクトプロパティを確認し、必要に応じて手動でプロパティを編集します。

接続の切り替え後、次のタスクを実行します。

- データ型マッピングの検証。
- テーブル所有者名の検証。
- ルックアップトランスフォーメーションの検証。
- ヒントの再設定。
- データオブジェクトの同期。

データ型マッピングの検証

接続を切り替える際に、Developer tool は元の接続および新しい接続が参照するデータベース間のデータ型の最適な一致を特定します。マッピング内のデータオブジェクトおよびトランスフォーメーションのデータ型が、データオブジェクトに基づいてそれぞれ設定されます。マッピングを実行する前に、データ型を確認して必要な場合はそれらを手動で更新します。

例えば、Oracle から Microsoft SQL Server に接続を切り替えるとします。固定精度のデータ型の場合、デフォルトでは、Developer tool は Microsoft SQL Server データベースの精度に基づいて精度を設定します。一方、可変精度のデータ型の場合、Developer tool は Oracle データベースに基づいて精度およびスケールを設定します。Timestamp with Time Zone のようなデータ型の場合、Developer tool は、データ型を Varchar (0,0)、または Microsoft SQL Server データベースが Varchar としてサポートしている同等のデータ型に設定します。

テーブルオーナー名の確認

接続を切り替える場合、Developer tool によりデータオブジェクトのテーブル所有者名が保持されます。必要に応じてデータオブジェクトプロパティのテーブル所有者名を手動で編集できます。

テーブル所有者名を空白に設定し接続を切り替えると、新規の接続が指すデータベースのデフォルトスキーマまたはパブリックスキーマにそのテーブルが存在する場合のみ、データを正常にプレビューできます。そうでない場合、データプレビューは失敗します。データを正常にプレビューするには、データオブジェクトのランタイムプロパティのテーブル所有者名を手動で更新する必要があります。

ルックアップトランスフォーメーションの検証

接続を切り替えた後、必要に応じて、データオブジェクトに基づきルックアップトランスフォーメーションのルックアップキャッシュを再構築し、ルックアップ条件の検証を行います。

次のタスクを実行します。

ルックアップキャッシュの再構築

ルックアップキャッシュを保持し、関連付けられたデータオブジェクトの接続の切り替えを行うようにデータ統合サービスを設定する場合、ルックアップトランスフォーメーションを更新して新規の接続に基づくルックアップキャッシュを再構築する必要があります。再構築しないと、マッピング実行時にキャッシュファイルが異なるデータベース接続で作成されたというエラーが発生します。

ルックアップ条件の検証

接続を切り替えると、Developer tool は元の接続と新規の接続が指すデータベース間で最適なデータ型の一致を識別し、データ型を適切に設定します。接続を切り替えると、ルックアップ条件は、データ型が変更になるため、有効ではないことがあります。ルックアップ条件を検証し手動で更新する必要があります。

例えば、IBM DB2 ソーステーブル、ルックアップテーブル、およびターゲットテーブルを含むマッピングを作成するとします。ソーステーブルおよびルックアップテーブルの Integer カラムでルックアップ条件を設定します。IBM DB2 から Oracle ヘルックアップテーブルの接続を切り替えると、ルックアップテー

ブルの Integer カラムのトランスフォーメーションデータ型が Decimal に変更されます。Integer カラムを Decimal カラムと比較できないため、ルックアップ条件は無効になります。

ヒントの再設定

カスタマイズデータオブジェクトの接続を切り替えた場合、Developer tool では、そのカスタマイズデータオブジェクトに設定されたヒントを保持しません。このため、ヒントを手動で再設定する必要があります。

データオブジェクトの同期

接続を切り替えると、Developer tool はアクティブな参照キー制約のみを保持します。

複数のデータオブジェクトの接続を同時に切り替える場合、データオブジェクトを同期して、正確なキーリレーションを維持する必要があります。

データオブジェクトに複数のテーブルが含まれ、それらのテーブル間に循環参照キー制約がある場合に、テーブルのサブセットの接続を別のデータベースタイプに切り替えると、Developer ツールはエラーを表示することなく接続を切り替えます。ただし、データオブジェクトのキーリレーションを表示すると、Developer tool には、元のデータベースを参照するキーリレーションが表示されます。キーリレーションを更新し、新しいデータベースを参照させるには、データオブジェクトを同期する必要があります。

サードパーティ製 JDBC ドライバ

JDBC を介してソースおよびターゲットに接続する場合は、サードパーティ製の JDBC タイプ 4 のドライバをインストールして設定します。

Developer ツールでメタデータをインポートする際は、サードパーティ製の JDBC ドライバ jar ファイルを次の場所にコピーします。

```
<InformaticaInstallationDir>\clients\externaljdbcjars
```

データのプレビュー、プロファイル、またはマッピングを実行するには、JDBC ドライバ jar ファイルを次の場所にコピーします。

```
<InformaticaInstallationDir>\externaljdbcjars
```

JDBC ドライバへの完全修飾パスを含めるように、CLASSPATH 環境変数を更新します。

第 5 章

物理データオブジェクト

この章では、以下の項目について説明します。

- [物理データオブジェクトの概要, 57 ページ](#)
- [物理データオブジェクトのタイプ, 58 ページ](#)
- [リレーショナルデータオブジェクト, 58 ページ](#)
- [カスタマイズデータオブジェクト, 62 ページ](#)
- [ターゲットテーブルの作成または置換, 67 ページ](#)
- [カスタムクエリ, 70 ページ](#)
- [非リレーショナルデータオブジェクト, 82 ページ](#)
- [WSDL データオブジェクトの概要 WSDL データオブジェクト, 84 ページ](#)
- [同期化, 88 ページ](#)
- [物理データオブジェクトのトラブルシューティング, 89 ページ](#)

物理データオブジェクトの概要

物理データオブジェクトは、データソースからの読み取り、データソースのルックアップ、またはデータソースへの書き込みに使用されるデータの物理的な表現です。データオブジェクトソースが変更された場合、物理データオブジェクトを同期させることができます。物理データオブジェクトを同期化すると、Developer ツールによってオブジェクトメタデータが再インポートされます。

物理データオブジェクトは、プロジェクトやフォルダー内に作成できます。プロジェクトやフォルダー内の物理データオブジェクトは再利用可能なオブジェクトです。再利用可能なオブジェクトはどのタイプのマッピング、マップレット、またはプロファイルでも使用できますが、マッピング、マップレット、またはプロファイル内のデータオブジェクトを変更することはできません。物理データオブジェクトを更新するには、プロジェクトまたはフォルダー内のオブジェクトを編集します。

物理データオブジェクトは、マッピング、マップレット、またはプロファイルに含めることができます。物理データオブジェクトは、読み取り、書き込み、またはルックアップトランスフォーメーションとして、マッピングまたはマップレットに追加できます。物理データオブジェクトを論理データオブジェクトマッピングに追加して、論理データオブジェクトにマッピングできます。

SQL データサービスを定義する際に、物理データオブジェクトを仮想テーブルマッピングに含めることもできます。また、Web サービスを定義する際に、操作マッピングに物理データオブジェクトを含めることもできます。

物理データオブジェクトのタイプ

データを読み取る/書き込むデータソースの種類に基づいて、さまざまな種類の物理データオブジェクトを作成できます。

物理データオブジェクトには、次のようなタイプがあります。

リレーショナルデータオブジェクト

ソースとしてリレーショナルテーブル、ビュー、またはシノニムを使用する物理データオブジェクト。例えば、Oracle ビューからリレーショナルデータオブジェクトを作成できます。

オブジェクトタイプに応じて、マッピングまたはマップレットにソース、ターゲット、またはルックアップトランスフォーメーションとしてリレーショナルデータオブジェクトを追加できます。

カスタマイズデータオブジェクト

ソースとして1つ以上の関連するリレーショナルリソースまたはリレーショナルデータオブジェクトを使用する物理データオブジェクト。リレーショナルリソースには、テーブル、ビュー、およびシノニムが含まれます。例えば、プライマリキー-外部キーリレーションを持つ2つの Microsoft SQL Server テーブルから、カスタマイズデータオブジェクトを作成できます。

データの結合、行のフィルタリング、ポートのソート、または再利用可能なデータオブジェクトでカスタムクエリの実行を行う場合は、カスタマイズデータオブジェクトを作成します。

非リレーショナルデータオブジェクト

ソースとして非リレーショナルデータベースリソースを使用する物理データオブジェクト。例えば、VSA ソースから非リレーショナルデータオブジェクトを作成できます。

フラットファイルデータオブジェクト

ソースとしてフラットファイルを使用する物理データオブジェクト。区切りフラットファイルまたは固定長フラットファイルから、フラットファイルデータオブジェクトを作成できます。

WSDL データオブジェクト

ソースとして WSDL ファイルを使用する物理データオブジェクト。

Informatica PowerExchange アダプタは、SAP、Salesforce、Netezza などの物理データオブジェクトを作成するために使用できる多くのデータソースへのアクセスも可能にします。

リレーショナルデータオブジェクト

リレーショナルデータオブジェクトは、リレーショナルテーブル、ビュー、またはシノニムをソースとして持つ物理データオブジェクトです。リレーショナルデータオブジェクトは、マッピング、マップレット、またはプロファイルに含めることができます。

リレーショナルデータオブジェクトは、データベース内のリソースを説明するものです。リポジトリにリレーショナルデータオブジェクトを追加する場合、ソースデータベースからリレーショナルデータオブジェクトをインポートします。リポジトリにリレーショナルデータオブジェクトをインポートした後、その定義を変更できます。ポートの追加と削除や、プライマリキーの定義ができるほか、リポジトリ内の複数のリレーショナルデータオブジェクト間のリレーションを設定できます。デフォルトの接続、データベースの所有者、およびリソース名の変更またはパラメータ化ができます。

リレーショナルデータオブジェクトをマッピング、マップレット、またはプロファイルに追加する場合、読み取りオブジェクトまたは書き込みオブジェクトのどちらを作成するかを示します。オブジェクトがソース、タ

ーゲット、またはルックアップテーブルのいずれであるかに基づいて、異なるランタイムプロパティを設定できます。

以下の図に、エディタで開いているサンプルのリレーショナルデータオブジェクトを示します。

Overview

General

Name: ReportDefinition_RelationalDataObject

Description:

ReportDefinition_RelationalDataObject

Name	Native Type
NAME	varchar2
DESCRIPTION	varchar2
POSITION	number(p,s)

Columns

	Name	Native Type	Precision	Scale	Primary...	Nullable	Description
1	NAME	varchar2	80	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
2	DESCRIPTION	varchar2	80	0	<input type="checkbox"/>	<input type="checkbox"/>	
3	POSITION	number(p,s)	10	0	<input type="checkbox"/>	<input type="checkbox"/>	

Overview | Keys | Relationships | Advanced

ソースデータベース内でのリレーシンの有無にかかわらず、リレーショナルデータオブジェクト間の、プライマリキー-外部キーのリレーシオンを作成できます。

複数のリレーショナルデータオブジェクトを、ソースとしてマッピングまたはマップレットに含めることができます。同時に複数のリレーショナルデータオブジェクトを追加する場合、以下のどちらかの方法でオブジェクトを追加するように Developer ツールにメッセージが表示されます。

- 関連するデータオブジェクトとして。複数のリレーショナルリソースを持つ読み取りトランスフォーメーションが 1 つ作成されます。読み取りトランスフォーメーションは、カスタマイズデータオブジェクトと同じ機能を持ちます。
- 独立したデータオブジェクトとして。リレーショナルデータオブジェクトごとに読み取りトランスフォーメーションが 1 つ作成されます。読み取りトランスフォーメーションは、リレーショナルデータオブジェクトと同じ機能を持ちます。

次のタイプのリレーショナルデータオブジェクトをインポートすることができます。

- DB2 for i5/OS
- DB2 for z/OS
- HAWQ
- IBM DB2
- JDBC
- Microsoft SQL Server
- Netezza
- ODBC
- Oracle
- SAP HANA

リレーショナルデータオブジェクトのインポート

リレーショナルデータオブジェクトをインポートし、マッピング、マプレット、またはプロファイルに追加します。

リレーショナルデータオブジェクトをインポートする前に、データベースへの接続を設定する必要があります。

1. **[Object Explorer]** ビューで、プロジェクトまたはフォルダーを選択します。
2. **[ファイル]** > **[新規]** > **[データオブジェクト]** をクリックします。
[新規] ダイアログボックスが表示されます。
3. **[リレーショナルデータオブジェクト]** を選択して、**[次へ]** をクリックします。
[新しいリレーショナルデータオブジェクト] ダイアログボックスが表示されます。
4. **[接続]** オプションの横にある **[参照]** をクリックして、データベースへの接続を選択します。
5. **[既存のリソースからデータオブジェクトを作成]** をクリックします。
6. **[リソース]** オプションの隣にある **[参照]** をクリックします。
[リソースの選択] ダイアログボックスが表示されます。
7. インポートする必要があるテーブル、ビュー、同義語を選択します。
8. データオブジェクトをフィルタリングするときは、**[フィルタ]** セクションで名前を入力し、**[検索]** をクリックします。
大文字小文字を区別してオブジェクト名で検索するときは名前を二重引用符で囲みます ("")。
9. 物理データオブジェクトの名前を入力します。
10. **[場所]** オプションの横にある **[参照]** をクリックして、リレーショナルデータオブジェクトをインポートするプロジェクトを選択します。
11. **[完了]** をクリックします。
データオブジェクトが、**[Object Explorer]** ビューのプロジェクトまたはフォルダーにある **[物理データオブジェクト]** の下に表示されます。

キーのリレーション

リレーショナルデータオブジェクト間にはキーリレーションを作成できます。キーリレーションを使用すると、リレーショナルデータオブジェクトをカスタマイズデータオブジェクトでデータソースとして使用する場合、またはマッピングやマプレットで読み取りトランスフォーメーションとして使用する場合に、リレーショナルデータオブジェクトを結合できます。

リレーショナルデータオブジェクトをインポートする場合、データベースに定義されているプライマリキー情報は保持されます。関連するリレーショナルデータオブジェクトを同時にインポートする場合、外部キーとキーリレーションも保持されます。ただし、関連するリレーショナルデータオブジェクトを個別にインポートする場合は、オブジェクトをインポートした後にキーリレーションを再作成する必要があります。

リレーショナルデータオブジェクト間にキーリレーションを作成するには、最初に、参照先オブジェクトにプライマリキーを作成します。次に、外部キーを含むリレーショナルデータオブジェクトにリレーションを作成します。

作成したキーリレーションは、リレーショナルデータオブジェクトのメタデータに存在します。ソースのリレーショナルリソースを変更する必要はありません。

リレーショナルデータオブジェクトのキーの作成

キーカラムは、リレーショナルデータオブジェクトの各行を識別するために作成します。リレーショナルデータオブジェクトごとに1つのプライマリキーを作成できます。

1. リレーショナルデータオブジェクトを開きます。
2. **【キー】** ビューを選択します。

以下の図に、エディタ内で開いているリレーショナルデータオブジェクト例の**【キー】** ビューを示します。

The screenshot shows the 'Keys' view of a database editor. The window title is 'Keys'. On the left, there is a list of keys with 'SYS_C006665' selected. To the right of the list are buttons for 'Add...' and 'Remove'. Further right are input fields for 'Name' (containing 'SYS_C006665') and 'Description'. Below these are two panes: 'Available Columns' and 'Selected Columns'. The 'Available Columns' pane contains 'DESCRIPTION' and 'POSITION'. The 'Selected Columns' pane contains 'NAME'. There are arrows between the panes to move columns. At the bottom of the window are tabs: 'Overview', 'Keys' (which is active), 'Relationships', and 'Advanced'.

3. **【追加】** をクリックします。
【新しいキー】 ダイアログボックスが表示されます。
4. キー名を入力します。
5. キーがプライマリキーの場合、**【プライマリキー】** を選択します。
6. キーカラムを選択します。
7. **【OK】** をクリックします。
8. リレーショナルデータオブジェクトを保存します。

リレーショナルデータオブジェクト間のリレーションの作成

リレーショナルデータオブジェクト間にはキーリレーションを作成できます。リレーショナルデータオブジェクトとカスタマイズデータオブジェクト間にキーリレーションを作成することはできません。

参照するリレーショナルデータオブジェクトにはプライマリキーが必要です。

1. 外部キーを作成するリレーショナルデータオブジェクトを開きます。
2. **【リレーション】** ビューを選択します。
3. **【追加】** をクリックします。
【新しいリレーション】 ダイアログボックスが表示されます。
4. 外部キーの名前を入力します。
5. 参照先リレーショナルデータオブジェクトからプライマリキーを選択します。
6. **【OK】** をクリックします。
7. **【リレーション】** プロパティで、外部キーカラムを選択します。
8. リレーショナルデータオブジェクトを保存します。

カスタマイズデータオブジェクト

カスタマイズデータオブジェクトは、1つ以上のリレーショナルソースを持つ物理データオブジェクトです。データ統合サービスでのソースデータの読み込み時に、データのジョイン、行のフィルタリング、ポートのソート、カスタムクエリの実行を行う場合は、カスタマイズデータオブジェクトを作成します。カスタマイズデータオブジェクトは、マッピング、マップレット、またはプロファイルで再利用することができます。

カスタマイズデータオブジェクトはプロジェクトまたはフォルダーに作成できます。カスタマイズデータオブジェクトは、マッピング、マップレット、またはプロファイルから変更することはできません。プロジェクトまたはフォルダー内のカスタマイズデータオブジェクトを変更すると、そのオブジェクトを使用するすべてのマッピング、マップレット、およびプロファイル内でオブジェクトが更新されます。

以下の図に、エディタ内で開いているカスタムデータオブジェクト例を示します。

Overview

General

Name:

Description:

Columns

Candidate columns:

	Name	Native Type	Precision	Scale	Visibility	Description
ReportDefinition						
1	NAME	varchar2	80	0	Read and...	
2	DESCRIPTION	varchar2	80	0	Read and...	
3	POSITION	number(p,s)	10	0	Read and...	

When column metadata changes:

☒ Synchronize input and output ☐ Do not synchronize

Overview Read Write Parameters Advanced

カスタマイズデータオブジェクトは、以下のタスクを実行するために作成します。

- カスタムクエリを作成し、データ統合サービスでソースデータを読み込むためのデフォルトクエリと置き換える。デフォルトクエリは、データ統合サービスがソースから読み込む各カラムを参照する SELECT 文です。
- データオブジェクトのパラメータを定義する。接続を表すカスタマイズデータオブジェクトに、パラメータを定義して割り当てることができます。接続名、テーブル所有者、およびテーブル名のパラメータを定義できます。カスタマイズデータオブジェクトを使用するマッピングを実行するとき、ランタイムの接続パラメータに対して個別に値を定義することができます。
- 同じソースデータベースから生成されたソースデータを結合する。データベース内にリレーションが存在するかどうかにかかわらず、プライマリキー-外部キーのリレーションで複数のテーブルを結合できます。
- オブジェクトをソースと同期するときにキーリレーションを保持する。複数のテーブルを含むカスタマイズデータオブジェクトを作成し、データベースに存在しないキーリレーションを定義する場合、データオブジェクトを同期するときにリレーションを保持することができます。
- ソースから個別の値を選択する。[個別に選択]を選択すると、データ統合サービスは SELECT DISTINCT 文をデフォルトの SQL クエリに追加します。

- データ統合サービスがソースデータを読み込むときに行をフィルタする。フィルタ条件を含めると、データ統合サービスはデフォルトクエリに WHERE 句を追加します。
- ソート済みポートを指定する。ソート済みポートに対して数値を指定すると、データ統合サービスは ORDER BY 句をデフォルトの SQL クエリに追加します。
- デフォルトのインナージョインの代わりにアウトジョインを指定する。ユーザ定義ジョインを含めると、データ統合サービスは SQL クエリのメタデータで指定された結合情報を置き換えます。
- マッピング実行前およびマッピング実行後 SQL コマンドを追加する。データ統合サービスは、ソースデータベースに対してマッピング実行前の SQL コマンドを実行した後、ソースを読み込みます。また、ターゲットへ書き込みをした後にソースデータベースに対してマッピング実行後 SQL コマンドを実行します。

以下のタイプの接続およびオブジェクトからカスタマイズデータオブジェクトを作成できます。

- DB2 i5/OS 接続
- DB2 z/OS 接続
- IBM DB2 接続
- JDBC 接続
- Microsoft SQL Server 接続
- ODBC 接続
- Oracle 接続
- リレーショナルデータオブジェクト

カスタム SQL クエリを介して、ソースをカスタマイズデータオブジェクトに追加できます。

キーのリレーション

ソースがリレーショナルリソースの場合、カスタマイズデータオブジェクトのソース間にキーリレーションを作成できます。キーリレーションを使用して、カスタマイズデータオブジェクト内のソースを結合することができます。

注: カスタマイズデータオブジェクトがソースとしてリレーショナルデータオブジェクトを使用する場合、カスタマイズデータオブジェクトにキーリレーションを作成することはできません。代わりに、リレーショナルデータオブジェクト間にキーリレーションを作成する必要があります。

リレーショナルデータオブジェクトをカスタマイズデータオブジェクトにインポートする場合、データベースに定義されているプライマリキー情報は保持されます。関連するリレーショナルリソース同士をカスタマイズデータオブジェクトに同時にインポートする場合、キーリレーション情報も保持されます。ただし、関連するリレーショナルデータリソースを個別にインポートする場合は、オブジェクトをカスタマイズデータオブジェクトにインポートした後にキーリレーションを再作成する必要があります。

カスタマイズデータオブジェクトのソース間にキーリレーションが存在する場合、Data Integration Service は、各ソースの関連キーに基づいてテーブルを結合します。このデフォルトジョインは、WHERE 句に次の構文を使用するインナー等価ジョインです。

```
Source1.column_name = Source2.column_name
```

ユーザ定義ジョインを入力するか、またはカスタムクエリを作成することで、デフォルトジョインをオーバーライドできます

カスタマイズデータオブジェクトにキーリレーションを作成するには、最初に、参照先ソーストランスフォーメーションにプライマリキーを作成します。次に、外部キーを含むソーストランスフォーメーションにリレーションを作成します。

作成したキーリレーションは、カスタマイズデータオブジェクトのメタデータに存在します。ソースのリレーショナルリソースを変更する必要はありません。

カスタマイズデータオブジェクトの書き込みプロパティ

Data Integration Service は、データをリレーショナルリソースに書き込むときに書き込みプロパティを使用します。書き込みプロパティを編集するには、**【書き込み】** ビューで入力トランスフォーメーションを選択し、**【詳細】** プロパティを選択します。

以下の表に、カスタマイズデータオブジェクトに対して設定する書き込みプロパティを示します。

プロパティ	説明
Hive ターゲットパーティションを切り詰めます	データが挿入されている Hive ターゲットのパーティションを上書きします。このオプションを有効にするには、ターゲットテーブルを切り詰めるオプションも選択する必要があります。 デフォルトは 【無効】 です。
ロードタイプ	ターゲットロードのタイプ。 【ノーマル】 または 【一括】 を選択します。 【ノーマル】 を選択した場合、データ統合サービスはターゲットを通常どおりにロードします。DB2、Sybase、Oracle、Microsoft SQL Server にロードする場合には、 【一括】 を選択できます。他のデータベースタイプに 【一括】 を指定すると、データ統合サービスがノーマルロードに戻します。バルクロードによってマッピングのパフォーマンスは向上しますが、データベースロギングが発生しないので、リカバリ機能が制限されます。 マッピングにアップデイトストラテジトランスフォーメーションが含まれる場合には、 【ノーマル】 モードを選択してください。 【ノーマル】 を選択したときに Microsoft SQL Server ターゲット名にスペースが含まれる場合は、接続オブジェクトで以下の環境 SQL を設定します。 SET QUOTED_IDENTIFIER ON
更新オーバーライド	ターゲットのデフォルトの UPDATE 文をオーバーライドします。
削除	削除のフラグが設定された行をすべて削除します。 デフォルトは 【有効】 です。
挿入	挿入のフラグが設定された行をすべて挿入します。 デフォルトは 【有効】 です。
実行時にテーブルを作成または置換	データ統合サービスは、実行時にターゲットテーブルを削除して、指定したターゲットテーブルに基づくテーブルで置換します。
ターゲットテーブルの切り詰め	データをロードする前にターゲットを切り詰めます。 デフォルトは 【無効】 です。
アップデイトストラテジ	既存の行のアップデイトストラテジ。次のいずれかのストラテジを選択します。 <ul style="list-style-type: none">- 更新時に更新。Data Integration Service は更新のフラグが設定された行をすべて更新します。- 挿入時に更新。Data Integration Service は更新のフラグが設定された行をすべて挿入します。ターゲットの 【挿入】 オプションも選択する必要があります。- 更新しない場合は挿入。更新のフラグが設定された行がターゲット内に存在している場合、Data Integration Service は、これらの行を更新した後、挿入のマークが付いている残りの行を挿入します。ターゲットの 【挿入】 オプションも選択する必要があります。

プロパティ	説明
PreSQL	ソースを読み込む前にターゲットデータベースに対してデータ統合サービスが実行する SQL コマンド。Developer tool では、SQL は検証されません。
PostSQL	ソースを読み込んだ後にターゲットデータベースに対してデータ統合サービスが実行する SQL コマンド。Developer tool では、SQL は検証されません。

カスタマイズデータオブジェクトの作成

マッピング、マップレット、またはプロファイルに追加するカスタマイズデータオブジェクトを作成します。カスタマイズデータオブジェクトを作成したら、ソースに追加します。

1. **[Object Explorer]** ビューで、プロジェクトまたはフォルダを選択します。
2. **[ファイル] > [新規] > [データオブジェクト]** をクリックします。
[新規] ダイアログボックスが表示されます。
3. **[リレーショナルデータオブジェクト]** を選択して、**[次へ]** をクリックします。
[新しいリレーショナルデータオブジェクト] ダイアログボックスが表示されます。
4. [接続] オプションの横にある **[参照]** をクリックして、データベースへの接続を選択します。
5. **[カスタマイズデータオブジェクトの作成]** をクリックします。
6. カスタマイズデータオブジェクトの名前を入力します。
7. [場所] オプションの横にある **[参照]** をクリックして、カスタマイズデータオブジェクトを作成するプロジェクトを選択します。
8. **[完了]** をクリックします。

カスタマイズデータオブジェクトが、**[Object Explorer]** ビューのプロジェクトまたはフォルダに表示されます。

カスタマイズデータオブジェクトにソースを追加します。リレーショナルリソースまたはリレーショナルデータオブジェクトを、ソースとして追加できます。カスタム SQL クエリを使用してソースを追加することもできます。

カスタマイズデータオブジェクトへのリレーショナルリソースの追加

カスタマイズデータオブジェクトを作成したら、ソースに追加します。リレーショナルリソースをソースとして使用できます。

リレーショナルリソースをカスタマイズデータオブジェクトに追加する前に、データベースへの接続を設定します。

1. **[Object Explorer]** ビューで、同一のリレーショナル接続でリレーショナルリソースを 1 つ以上選択します。
2. **[接続エクスプローラ]** ビューを右クリックして、**[プロジェクトに追加]** を選択します。
[プロジェクトに追加] ダイアログボックスが表示されます。
3. **[既存のカスタマイズされたデータオブジェクトに関連リソースとして追加]** を選択して、**[OK]** をクリックします。
[データオブジェクトに追加] ダイアログボックスが表示されます。
4. カスタマイズデータオブジェクトを選択し、**[OK]** をクリックします。

5. 複数のリソースをカスタマイズデータオブジェクトに追加する場合、書き込み先のリソースを選択するように求められます。リソースを選択して **[OK]** をクリックします。

マッピングのカスタマイズデータオブジェクトを書き込みトランスフォーメーションとして使用すると、Developer ツールはデータをこのリソースに書き込みます。

リソースがカスタマイズデータオブジェクトに追加されます。

カスタマイズデータオブジェクトへのリレーショナルデータオブジェクトの追加

カスタマイズデータオブジェクトを作成したら、ソースに追加します。リレーショナルデータオブジェクトをソースとして使用できます。

1. カスタマイズデータオブジェクトを開きます。
2. **[読み取り]** ビューを選択します。
3. **[Object Explorer]** ビューで、1 つのリレーショナル接続からリレーショナルデータオブジェクトを 1 つ以上選択します。
4. オブジェクトを **[Object Explorer]** ビューからカスタマイズデータオブジェクトの **[読み取り]** ビューにドラッグします。
5. 複数のリレーショナルデータオブジェクトをカスタマイズデータオブジェクトに追加する場合、書き込み先のオブジェクトを選択するように求められます。オブジェクトを選択して **[OK]** をクリックします。

マッピングのカスタマイズデータオブジェクトを書き込みトランスフォーメーションとして使用すると、Developer ツールはデータをこのリレーショナルデータオブジェクトに書き込みます。

リレーショナルデータオブジェクトがカスタマイズデータオブジェクトに追加されます。

カスタマイズデータオブジェクトのキーの作成

ソーストランスフォーメーションの各行を識別するためのキーカラムを作成します。各ソーストランスフォーメーションごとに 1 つのプライマリキーを作成できます。

1. カスタマイズデータオブジェクトを開きます。
2. **[読み取り]** ビューを選択します。
3. キーを作成するソーストランスフォーメーションを選択します。

ソースは、リレーショナルデータオブジェクトではなくリレーショナルリソースにする必要があります。ソースがリレーショナルデータオブジェクトの場合、リレーショナルデータオブジェクトにキーを作成する必要があります。

4. **[キー]** プロパティを選択します。
5. **[追加]** をクリックします。
[新しいキー] ダイアログボックスが表示されます。
6. キー名を入力します。
7. キーがプライマリキーの場合、**[プライマリキー]** を選択します。
8. キーカラムを選択します。
9. **[OK]** をクリックします。
10. カスタマイズデータオブジェクトを選択します。

カスタマイズデータオブジェクト内のリレーションの作成

カスタマイズデータオブジェクトにソース間のキーリレーションを作成できます。

参照するソーストランスフォーメーションにはプライマリキーが必要です。

1. カスタマイズデータオブジェクトを開きます。
2. **【読み取り】** ビューを選択します。
3. 外部キーを作成するソーストランスフォーメーションを選択します。
ソースは、リレーショナルデータオブジェクトではなくリレーショナルリソースにする必要があります。
ソースがリレーショナルデータオブジェクトの場合、リレーショナルデータオブジェクトにリレーションを作成する必要があります。
4. **【リレーション】** プロパティを選択します。
5. **【追加】** をクリックします。
【新しいリレーション】 ダイアログボックスが表示されます。
6. 外部キーの名前を入力します。
7. 参照先のソーストランスフォーメーションからプライマリキーを選択します。
8. **【OK】** をクリックします。
9. **【リレーション】** プロパティで、外部キーカラムを選択します。
10. カスタマイズデータオブジェクトを選択します。

ターゲットテーブルの作成または置換

Developer tool では、モデルリポジトリで 1 つ以上のリレーショナルデータオブジェクトの DDL スクリプトを生成し、DDL スクリプトを実行してターゲットデータベースでテーブルの作成または置換ができます。データベースにすでにターゲットが存在する場合、そのターゲットを削除して再作成できます。

設計時または実行時にターゲットを作成または置換できます。設計時にターゲットを作成または置換するには、マッピング実行前に DDL スクリプトを作成して実行する必要があります。マッピングで書き込みトランスフォーメーションを設定し、実行時にリレーショナルターゲットを作成または置換できます。

Developer tool ですべてのサポートされた接続タイプに対応したデータベース固有バージョンの DDL スクリプトが生成されます。JDBC または ODBC ターゲットを選択すると、Developer tool は ANSI SQL-92 generic データ型形式を生成します。次のデータベースタイプの DDL スクリプトを生成できます。

- IBM DB2
- Greenplum
- Hive
- JDBC
- Microsoft SQL Server
- Netezza
- ODBC
- Oracle
- Teradata

注: OLE DB SQL Server 接続を使用する場合は、DDL スクリプトを生成できません。

ターゲットテーブルを作成または置換するためのルールおよびガイドライン

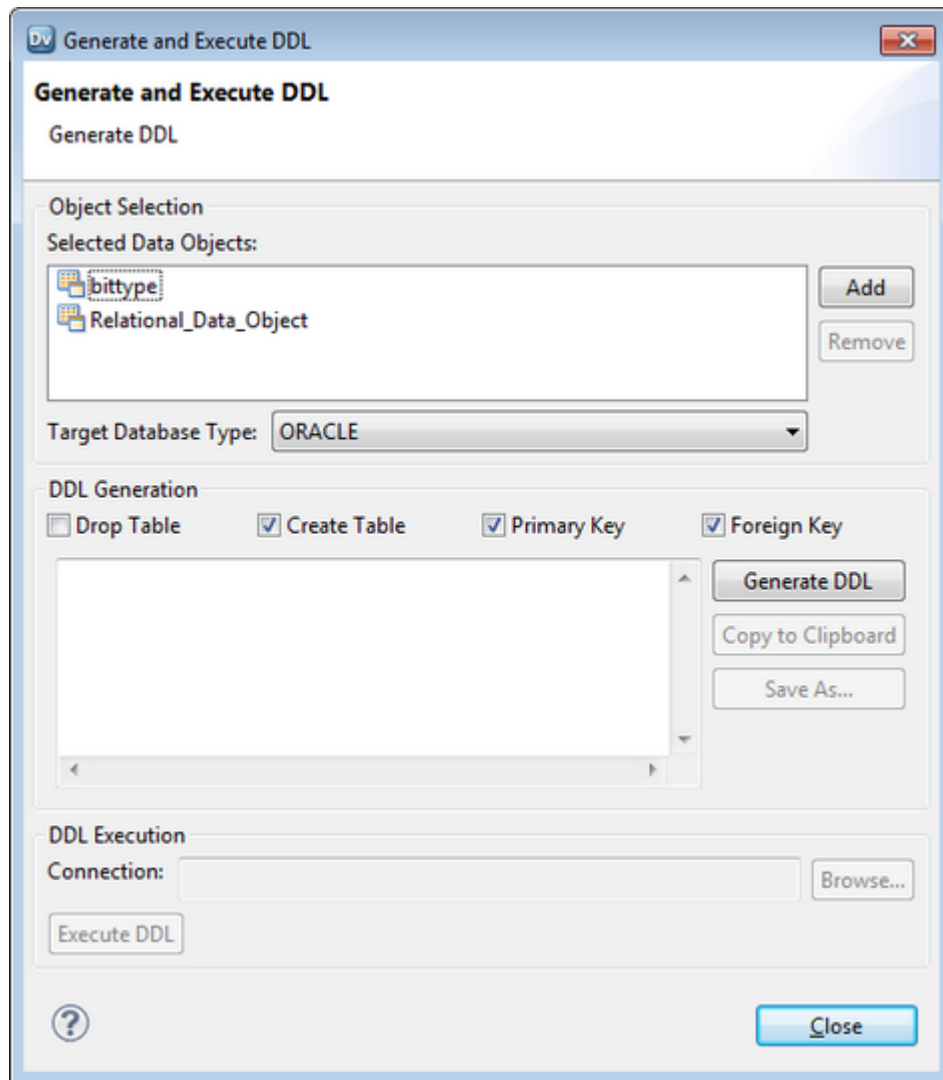
DDL スクリプトを生成および実行する場合は、以下のルールとガイドラインを考慮します。

- 同じソーステーブル名を持つ複数のオブジェクトを選択しないようにします。同じソーステーブル名を持つ複数のオブジェクトを選択すると、DDL コードが失敗します。同じソーステーブル名を持つ3つのオブジェクトに「テーブルの削除」オプションおよび「テーブルの作成」オプションを選択した場合、DDL コードは最初の「テーブルの削除」コマンドおよび「テーブルの作成」コマンドでは成功しますが、後続のコマンドでは失敗します。
- マッピングを実行する前に、データ型を確認して必要な場合はそれらを手動で更新します。DDL スクリプトを生成するときに、Oracle の char および byte のセマンティクスは無視されます。Oracle データベースの char カラムおよび varchar2 カラムを含むテーブルを作成する場合は、char および byte セマンティクスのためのストレージを定義できます。Oracle テーブルのメタデータを Developer tool にインポートする場合、char および byte セマンティクスは無視されます。Oracle テーブルの DDL スクリプトを生成する場合、Developer tool はデータ型を Char として定義します。
- ODBC 接続の DDL を生成する場合、Developer tool は DDL スクリプトを ANSI SQL-92 の汎用データ型形式で作成します。ANSI SQL-92 形式は、ターゲットデータベースでデータ型またはデータ長がサポートされないことがあるため、一部のデータベースでは実行されない場合があります。
- Greenplum から Netezza にデータベーステーブルを移行するために DDL スクリプトを作成する場合、Netezza データベースの NVARCHAR データ型では最大 16000 文字のみがサポートされるため、NVARCHAR カラムに含めることができるのは 16000 文字のみです。
- DDL スクリプトを生成する際に、Developer tool は元の接続および新しい接続が参照するデータベース間のデータ型の最適な一致を特定します。データ型の精度およびスケールはデータベースによって異なります。Oracle データベースでは、Timestamp データ型のデフォルトの精度およびスケールは(29, 9)です。Oracle から Microsoft SQL Server への DDL スクリプトを生成する場合、Timestamp データ型の精度およびスケールは(26, 6)に減少します。Oracle から DB2 への DDL スクリプトを生成する場合、Timestamp データ型の精度およびスケールは(27, 7)に減少します。

設計時の DDL の生成と実行

DDL スクリプトを生成および実行する前に、ユーザーにターゲットデータベースにアクセスするための適切な読み取り権限および書き込み権限があることを確認します。

1. Developer tool の「**オブジェクトエクスプローラ**」ビューで、ターゲットデータベースにテーブルを作成するリレーショナルデータオブジェクトを選択します。複数のテーブルを作成する場合は、**Shift** キーと **Ctrl** キーを押しながら、複数のデータオブジェクトを選択します。
2. 選択したデータオブジェクトを右クリックして、「**DDL の生成と実行**」を選択します。
「DDL の生成と実行」ダイアログボックスが表示されます。



3. [オブジェクトの選択] 領域で、DDL を生成する **【ターゲットデータベースタイプ】** を選択します。

4. [DDL の生成] 領域では、次のオプションを選択できます。

- **テーブルの削除。** データベースにテーブルを作成する前にテーブルを削除します。このオプションは、テーブルを置き換える場合に選択します。
- **テーブルの作成。** ターゲットデータベースにテーブルを作成します。
- **プライマリキー。** 選択したデータオブジェクトに基づいてプライマリキーを作成します。
- **外部キー。** 選択したデータオブジェクトに基づいて外部キーを作成します。
- **クリップボードにコピー。** DDL スクリプトをクリップボードにコピーします。
- **名前を付けて保存。** DDL スクリプトをファイルに保存します。

5. [DDL の生成] 領域で、**【DDL の生成】** をクリックします。

選択したデータオブジェクトのために生成された DDL が [DDL の生成] 領域に表示されます。

警告: [DDL の生成] 領域に生成された DDL スクリプトを編集した場合、その DDL スクリプトを実行したときにエラーが発生することがあります。

6. [DDL の実行] 領域で、**【参照】** をクリックしてターゲットデータベースの接続を選択します。

【接続の選択】ダイアログボックスには、【オブジェクトの選択】領域で選択したターゲットデータベースタイプに基づいてターゲットデータベースの接続が一覧表示されます。例えば、ターゲットデータベースタイプとして Oracle を選択した場合、Developer tool は Oracle の接続を表示します。

7. 【DDL の実行】をクリックします。
8. 【閉じる】をクリックします。

実行時の DDL の生成と実行

実行時にリレーショナルターゲットを作成または置換するように、マッピング内の書き込みトランスフォーメーションを設定できます。

データ統合サービスは実行時に DDL スクリプトを生成して実行します。詳細については、『*Informatica Developer マッピングガイド*』を参照してください。

DDL の生成エラー

DDL スクリプトを生成して実行するときに、エラーが発生する場合があります。エラーの原因として以下のいずれかが考えられます。

- 選択したデータ型をターゲットデータベースがサポートしていない。
- 選択した物理データオブジェクトに循環依存関係が含まれる。
- 選択した物理データオブジェクトにカラムが含まれない。

カスタムクエリ

カスタム SQL クエリは、カスタマイズデータオブジェクト内のデフォルトの SQL クエリを上書きする SELECT 文です。

カスタムクエリは、データ統合サービスがリレーショナルソースからデータを読み取るために使用するデフォルトの SQL クエリをオーバーライドします。カスタムクエリは、ソースフィルタの入力、ソート済みポートの使用、ユーザー定義ジョインの入力、または異なるポートの選択時に定義する単純クエリの設定もオーバーライドします。

データベース言語では有効であるがトランスフォーメーション言語では使用できない SQL 操作を実行するためのカスタムクエリを作成できます。カスタマイズデータオブジェクトにカスタムクエリを定義する場合、複数のマッピングまたはプロファイルでオブジェクトを再利用することができます。

カスタマイズデータオブジェクトにカスタムクエリを作成する場合のガイドラインは以下のとおりです。

- SELECT 文では、ソーストランスフォーメーションに表示される順序でカラム名を記述する必要があります。
- データベース予約語はすべて引用符で囲んでください。
- ドル記号 (\$) の前にエスケープ文字を追加します。\$ の前にスラッシュ (\) 文字がある場合、エスケープ文字 (\) をスラッシュとドル記号の両方に追加します。例えば、\$ を \\$ と入力し、\\$ を \\\$ と入力します。

カスタマイズデータオブジェクトを使用して自己結合を実行する場合、自己結合を含むカスタム SQL クエリを入力する必要があります。カスタムクエリが設定されたカスタマイズデータオブジェクトを、マッピングのソースとして使用できます。ソースデータベースは、データをデータ統合サービスに渡す前にクエリを実行します。ソースを空のカスタマイズデータオブジェクトに追加するカスタムクエリを作成できます。カスタムクエリを使用して、デフォルトの SQL クエリをオーバーライドすることもできます。

カスタムクエリの最適化

データ統合サービスは、パフォーマンスを向上させるために、カスタムクエリをプッシュしてリレーショナルデータオブジェクトで実行できます。クエリがデータベースに対して有効なサブクエリを形成する場合は、カスタムクエリをプッシュするようにしてください。

カスタムクエリを使用してリレーショナルデータオブジェクトのデータを読み取ると、データ統合サービスはデータベース内でクエリを実行することによりクエリを最適化できます。クエリがデータベースに対して有効なサブクエリを形成する場合、データ統合サービスはカスタムクエリをプッシュできます。カスタムクエリの SQL 構文がデータベースに対するサブクエリとして有効ではない場合、結果のクエリは実行に失敗します。

IBM DB2 以外のリレーショナルデータベースに対してカスタムクエリをプッシュする場合は、カラム参照ではない選択リストでそれぞれの式に対するエイリアスを指定する必要があります。これらのエイリアスにより、データ統合サービスが選択リスト内の式を参照できるようになります。

エイリアスとサブクエリに対する有効な SQL 構文に関する詳細は、データベースのマニュアルを参照してください。

デフォルトクエリ

Data Integration Service は、リレーショナルソースからデータを読み取るデフォルトの SQL クエリを生成します。カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスでデフォルトクエリをオーバーライドすることができます。

単純クエリまたは詳細クエリでデフォルトクエリをオーバーライドすることができます。単純クエリを使用して、個別の値の選択、ソースフィルタの入力、ポートのソート、またはユーザー定義ジョインの入力を行うことができます。ソースからデータを読み取るためのカスタム SQL クエリを作成するには、詳細クエリを使用します。カスタムクエリはデフォルトクエリおよび単純クエリをオーバーライドします。

テーブル名またはカラム名にデータベース予約語が含まれる場合、予約語ファイル reswords.txt を作成および管理できます。reswords.txt ファイルは、Data Integration Service がアクセスできる任意のマシンに作成します。

Data Integration Service は、マッピングの実行時に reswords.txt ファイルを検索します。ファイルが存在する場合、Data Integration Service はデータベースに対して SQL 文を実行する際に、一致した予約語を引用符で囲みます。デフォルトクエリをオーバーライドする場合、データベース予約語はすべて引用符で囲む必要があります。

Data Integration Service は、デフォルトクエリを生成するときに以下の文字を含むテーブル名およびフィールド名を二重引用符で区切ります。

/ + - = ~ ` ! % ^ & * () [] { } ' ; ? , < > \ | <space>

予約語ファイルの作成

カスタマイズデータオブジェクトにデータベース予約語が含まれる場合、予約語ファイルを作成します。

予約語ファイルを使用するように Data Integration Service を設定するには、管理者特権が必要です。

1. 「reswords.txt」というファイルを作成します。
2. 角括弧で囲んでデータベース名を入力して（例 [Oracle]）、各データベースのセクションを作成します。
3. 予約語をデータベース名の下に追加します。

以下に例を示します。

```
[Oracle]
OPTION
START
where
```



```
number
[SQL Server]
CURRENT
where
number
```

エントリでは、大文字と小文字は区別されません。

4. reswords.txt ファイルを保存します。
5. Informatica Administrator で、[Data Integration Service] を選択します。
6. カスタムプロパティを編集します。
7. 以下のカスタムプロパティを追加します。

名前	値
予約語ファイル	<path>\reswords.txt

8. Data Integration Service をリスタートします。

ヒント

データベースオブティマイザに対する指示として、ソースの SQL クエリにヒントを追加することができます。オブティマイザはヒントを使用して、ソースにアクセスするためのクエリ実行プランを選択します。

[ヒント] フィールドは、リレーショナルデータオブジェクトインスタンスまたはカスタマイズデータオブジェクトの **【クエリ】** ビューに表示されます。ソースデータベースは、Oracle、Sybase、IBM DB2、または Microsoft SQL Server でなければなりません。それ以外のデータベースタイプでは、[ヒント] フィールドは表示されません。

Data Integration Service によるソースクエリの生成時に、Developer ツールで入力した SQL ヒントがクエリに追加されます。ヒントは Data Integration Service では解析されません。ソースを含むマッピングを実行したときに、マッピングログにクエリとヒントが表示されます。

Data Integration Service によって SQL ヒントが挿入されるクエリ内の位置は、データベースタイプによって異なります。ヒントの構文については、データベースのマニュアルを参照してください。

Oracle

ヒントが SELECT/UPDATE/INSERT/DELETE キーワードのすぐ後に追加されます。

```
SELECT /*+ <hints> */ FROM ...
```

'+'はヒントの先頭を示します。

ヒントはコメント（/* ... */または--...行の末尾まで）に格納されます。

Sybase

ヒントがクエリのすぐ後に追加されます。ヒントでプラン名を設定します。

```
SELECT ... PLAN <plan>
```

```
select avg(price) from titles plan "(scalar_agg (i_scan type_price_ix titles ))"
```

IBM DB2

ヒントとして optimize-for 句を入力できます。この句はクエリの末尾に追加されます。

```
SELECT ... OPTIMIZE FOR <n> ROWS
```


optimize-for 句では、クエリで処理する行数をデータベースオブティマイザに示します。これは行数の制限ではありません。データベースで処理する行数が<n>行を超えると、パフォーマンスが低下する可能性があります。

Microsoft SQL Server

ヒントがクエリの末尾に OPTION 句の一部として追加されます。

```
SELECT ... OPTION ( <query_hints> )
```

ヒントに関するルールとガイドライン

SQL クエリのヒントを設定するときは、次のルールとガイドラインに従います。

- プッシュダウンの最適化を有効にする場合やリレーショナルデータオブジェクトで準結合を使用する場合は、元のソースクエリが変わります。変更後のクエリにはヒントは適用されません。
- ヒントはジョインやフィルタのオーバーライドと組み合わせることができますが、SQL のオーバーライドを設定した場合は SQL のオーバーライドが優先され、他のオーバーライドは適用されません。
- **【クエリ】** ビューには、簡易ビューと詳細ビューがあります。簡易ビューでフィルタ、ソート、またはジョインのオーバーライドを使用してヒントを入力した場合、完全なクエリオーバーライドは詳細ビューに表示されます。

ヒントの作成

データベースオブティマイザにクエリプランの決定に関する指示を伝えるには、ヒントを作成します。

1. カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスを開きます。
2. **【読み取り】** ビューを選択します。
3. **【出力トランスフォーメーション】** を選択します。
4. **【クエリ】** プロパティを選択します。
5. 単純クエリを選択します。
6. **【ヒント】** フィールドの横にある **【編集】** をクリックします。
【ヒント】 ダイアログボックスが表示されます。
7. **【SQL クエリ】** フィールドにヒントを入力します。
Developer ツールでは、ヒントは検証されません。
8. **【OK】** をクリックします。
9. データオブジェクトを保存します。

個別に選択

カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスのソースから、個別選択オプションを使用して一意の値を選択できます。個別に選択を有効にすると、SELECT DISTINCT 文がデフォルトの SQL クエリに追加されます。

個別選択オプションを使用して、ソースデータをフィルタします。例えば、**【個別に選択】** オプションを使用して、総売上高の一覧のテーブルから一意の顧客 ID を抽出できます。マッピングでリレーショナルデータオブジェクトを使用する場合、データフローの早い段階でデータがフィルタされるため、パフォーマンスが向上します。

個別に選択の使用

【個別選択】 プロパティを使用して、リレーショナルソースから一意の値を選択します。

1. カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスを開きます。
2. 【読み取り】 ビューを選択します。
3. 【出力トランスフォーメーション】 を選択します。
4. 【クエリ】 プロパティを選択します。
5. 単純クエリを選択します。
6. 【個別に選択】 オプションを選択します。
7. カスタマイズデータオブジェクトを選択します。

フィルタ

カスタムクエリにフィルタ値を入力できます。 フィルタはクエリ SELECT 文の WHERE 句に変換されます。 Data Integration Service がソースのテーブルから読み取る行数を減らすにはフィルタを使用します。

ソースフィルタの入力

Data Integration Service がリレーショナルリソースから読み取る行数を減らすには、ソースフィルタを入力します。

1. カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスを開きます。
2. 【読み取り】 ビューを選択します。
3. 【出力トランスフォーメーション】 を選択します。
4. 【クエリ】 プロパティを選択します。
5. 単純クエリを選択します。
6. 【フィルタ】 フィールドの横にある【編集】 ボタンをクリックします。
【SQL クエリ】 ダイアログボックスが表示されます。
7. 【SQL クエリ】 フィールドにフィルタ条件を入力します。
【カラム】 リストからカラムを選択することもできます。
8. 【OK】 をクリックします。
9. 【検証】 をクリックしてフィルタ条件を検証します。
10. データオブジェクトを保存します。

ソート済みポート

カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスのデフォルトクエリで行をソートできます。 ソート基準のポートを選択します。 Data Integration Service はそのポートをデフォルトクエリの ORDER BY 句に追加します。

マッピングに以下のトランスフォーメーションを含めるときに、ソース行をソートしてパフォーマンスを向上させることができます。

- アグリゲータ。アグリゲータトランスフォーメーションをソート済み入力用に設定すると、ソート済みポートを使ってソート済みデータを送ることができます。アグリゲータトランスフォーメーションの Group By ポートは、カスタマイズデータオブジェクトのソート済みポートの順序と一致する必要があります。

- ジョイナ。ジョイナトランスフォーメーションをソート済み入力用に設定すると、ソート済みポートを使ってソート済みデータを送ることができます。各カスタマイズデータオブジェクト内のソート済みポートの順序は同一に設定してください。

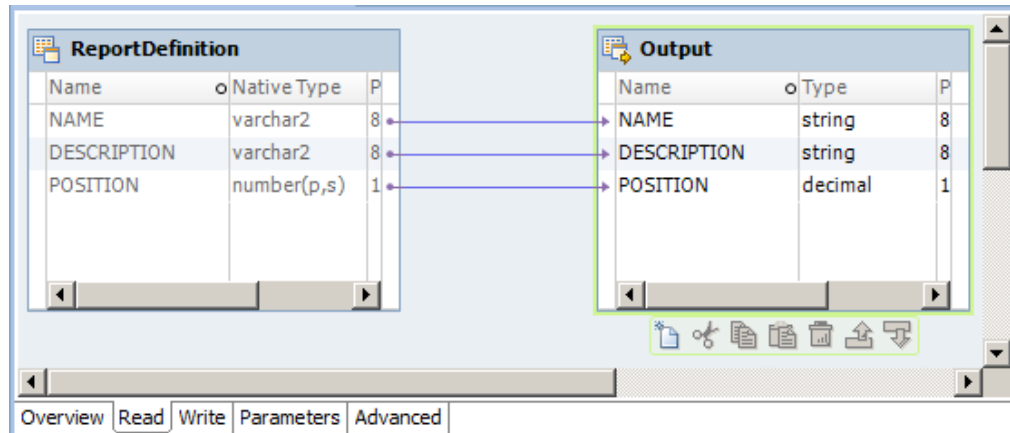
注: また、アグリゲータおよびジョイナトランスフォーメーションの前にソータートランスフォーメーションを使用して、リレーショナルデータおよびフラットファイルデータをソートすることもできます。

コラムデータのソート

ソート済みポートを使用して、カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスのコラムデータをソートします。このデータオブジェクトをマッピングまたはマップレットで読み取りトランスフォーメーションとして使用する場合、ソート済みデータを、読み取りトランスフォーメーションからトランスフォーメーションダウンストリームに送信できます。

1. カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスを開きます。
2. **【読み取り】** ビューを選択します。

以下の図に、エディタ内で開いているカスタマイズデータオブジェクトの**【読み取り】** ビューを示します。



3. 出力トランスフォーメーションを選択します。
4. **【クエリ】** プロパティを選択します。
5. 単純クエリを選択します。
6. **【編集】** ボタン (**【ソート】** フィールドの横) をクリックします。
【ソート】 ダイアログボックスが表示されます。
7. ソート済みポートとしてコラムを指定するには、**【新規】** ボタンをクリックします。
8. コラムとソートタイプ（昇順または降順）を選択します。
9. 手順 [7](#)~[8](#) を繰り返して、ソートする他のコラムを選択します。
【ソート】 ダイアログボックスに表示される順序に従って、コラムがソートされます。
10. **【OK】** をクリックします。
【クエリ】 プロパティに、**【ソート】** フィールドのソースコラムが表示されます。
11. **【検証】** をクリックしてソート構文を検証します。
12. データオブジェクトを保存します。

ユーザー定義ジョイン条件

ユーザー定義ジョインをカスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスで設定できます。ユーザー定義ジョインは、複数のソースのデータを同じデータオブジェクトに結合するための条件を定義します。

ユーザー定義ジョインをカスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスに追加すると、マッピングの読み取りトランスフォーメーションとしてそのデータオブジェクトを使用できます。ソースデータベースは、データを Data Integration Service に渡す前にジョインを実行します。ソーステーブルのインデックスを作成すると、マッピングのパフォーマンスが向上します。

関連ソースからのデータを結合するにはユーザー定義ジョインを作成します。ユーザー定義ジョインは、各ソースの関連キーに基づいて作成されるデフォルトのインナージョインをオーバーライドします。ユーザー定義ジョインを入力する場合、結合条件を指定する WHERE 句の内容を入力します。ユーザー定義ジョインがアウタージョインを実行すると、データベース構文に応じて WHERE 句または FROM 句にジョイン構文が挿入されます。

以下の環境では、ユーザー定義ジョインの入力が必要になることがあります。

- カラムがプライマリキーと外部キーの関係を持たない。
- 結合に使用するカラムのデータ型が一致しない。
- アウタージョインのような、別の種類の結合を指定したい。

カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトにユーザー定義ジョインを入力する場合のガイドラインは以下のとおりです。

- ユーザー定義ジョインには WHERE キーワードを含めないでください。
- データベース予約語はすべて引用符で囲んでください。
- Informatica ジョイン構文を使用し、**【SQL 内の引用符を有効にする】**が接続に対して有効になっている場合、テーブル名や接続名を手動で入力するときは引用符で囲む必要があります。ユーザー定義ジョインを入力するときにテーブルやカラムを選択すると、Developer ツールによってテーブル名やカラム名が引用符で囲まれます。

ユーザー定義ジョインは、データベースの関連リソースのデータを結合します。異種データソースを結合する場合、ソースからデータを読み取るマッピングでジョイナトランスフォーメーションを使用します。自己結合を実行するには、自己結合を含むカスタム SQL クエリを入力する必要があります。

ユーザー定義ジョインの入力

ユーザー定義ジョインをカスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトで設定し、データオブジェクトソースの結合条件を定義します。

1. カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスを開きます。
2. **【読み取り】** ビューを選択します。
3. **【出力トランスフォーメーション】** を選択します。
4. **【クエリ】** プロパティを選択します。
5. 単純クエリを選択します。
6. **【結合】** フィールドの横にある **【編集】** ボタンをクリックします。
【SQL クエリ】 ダイアログボックスが表示されます。
7. **【SQL クエリ】** フィールドにユーザー定義ジョインを入力します。
【カラム】 リストからカラムを選択することもできます。
8. **【OK】** をクリックします。

9. **【検証】** をクリックしてユーザー定義ジョインを検証します。
10. データオブジェクトを保存します。

アウトージョインのサポート

カスタマイズデータオブジェクトを使用して、同じデータベース内の 2 つのソースのアウトージョインを実行できます。Data Integration Service では、アウトージョイン実行時に、1 つ目のソースリソースからすべての行を返し、2 番目のソースリソースからジョイン条件に一致する行を返します。

2 つのリソースを結合し、一方のリソースのすべての行を返したい場合には、アウトージョインを使用します。たとえば、登録済み顧客テーブルを月ごとの購入テーブルと結合して登録済み顧客の行動を調べたい場合に、アウトージョインを使用することができます。登録済み顧客テーブルを月ごとの購入テーブルと結合し、前月に何も購入しなかった顧客も含めて、登録済み顧客テーブルのすべての行を返すことができます。Normal ジョインを実行した場合、Data Integration Service はその月に何か購入した登録済み顧客のみ、および登録済み顧客による購入のみを返します。

アウトージョインの場合、ジョイナトランスフォーメーションの Master Outer（マスターアウト）ジョインまたは Detail Outer（明細アウト）ジョインとして同じ結果を生成することができます。しかし、アウトージョインを使用するとデータフロー内の行数が削減され、パフォーマンスが向上します。

2 種類のアウトージョインを入力できます。

- レフト。Data Integration Service は、ジョイン構文の左辺のリソースからすべての行を、また、両方のリソースからジョイン条件に一致する行を返します。
- ライト。Data Integration Service は、ジョイン構文の右辺のリソースからすべての行を、また、両方のリソースからジョイン条件に一致する行を返します。

注: デフォルトクエリをオーバーライドする場合、ネストされたクエリ文でアウトージョインを使用します。

アウトージョインはユーザー定義ジョインまたはカスタム SQL クエリに入力できます。

Informatica ジョイン構文

ジョイン構文を入力する場合、Informatica ジョイン構文またはデータベース固有のジョイン構文を使用します。Informatica ジョイン構文を使用すると、Data Integration Service はマッピング実行中に構文を変換してソースデータベースに渡します。

注: ジョイン条件については、必ずデータベース固有の構文を使用してください。

Informatica ジョイン構文を使用する場合、結合文全体を「{Informatica 構文}」のようにブレースで囲みます。データベース構文を使用する場合、ソースデータベースがサポートしている構文をブレースで囲まずに使用します。

Informatica ジョイン構文を使用する場合、テーブル名をカラム名の前に付けます。たとえば、REG_CUSTOMER テーブルの FIRST_NAME という名前のカラムの場合、ジョイン構文には「REG_CUSTOMER.FIRST_NAME」と入力します。また、テーブル名にエイリアスを使用している場合、Data Integration Service がエイリアスを確実に認識するように、Informatica ジョイン構文内でエイリアスを使用します。

1 つのデータオブジェクトでレフトアウトージョインまたはライトアウトージョインを Normal ジョインと組み合わせることができます。レフトアウトージョインとライトアウトージョインを組み合わせることはできません。複数の Normal ジョイン、および複数のレフトアウトージョインを使用します。一部のデータベースでは、ライトアウトージョインの使用に制限があります。

ジョインを組み合わせる場合、Normal ジョインを最初に入力します。

Normal ジョイン構文

カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスの結合条件を使用して、Normal ジョインを作成できます。

アウタージョインを作成する場合、デフォルトジョインをオーバーライドする必要があります。したがって、Normal ジョインをジョインオーバーライドに含める必要があります。Normal ジョインをジョインオーバーライドに含める場合、アウタージョインの前に Normal ジョインを記述します。ジョインオーバーライドには複数の Normal ジョインを入力することができます。

Normal ジョインを作成するには、以下の構文を使用します。

```
{ source1 INNER JOIN source2 on join_condition }
```

以下の表に、ジョインオーバーライド内の Normal ジョインの構文を示します。

構文	説明
<i>source1</i>	ソースリソース名。Data Integration Service は、このリソースからジョイン条件に一致する行を返します。
<i>source2</i>	ソースリソース名。Data Integration Service は、このリソースからジョイン条件に一致する行を返します。
<i>join_condition</i>	ジョイン条件。ソースデータベースがサポートしている構文を使用してください。複数のジョイン条件を AND 演算子によって組み合わせることができます。

たとえば、登録済み顧客のデータを含む REG_CUSTOMER テーブルがあるとします。

CUST_ID	FIRST_NAME	LAST_NAME
00001	Marvin	Chi
00002	Dinah	Jones
00003	John	Bowden
00004	J.	Marks

毎月更新される PURCHASES テーブルに以下のデータが含まれているとします。

TRANSACTION_NO	CUST_ID	DATE	AMOUNT
06-2000-0001	00002	6/3/2000	55.79
06-2000-0002	00002	6/10/2000	104.45
06-2000-0003	00001	6/10/2000	255.56
06-2000-0004	00004	6/15/2000	534.95
06-2000-0005	00002	6/21/2000	98.65
06-2000-0006	NULL	6/23/2000	155.65
06-2000-0007	NULL	6/24/2000	325.45

6 月の各取り引きの顧客名を表示する行を返すには、次の構文を使用します。

```
{ REG_CUSTOMER INNER JOIN PURCHASES on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID }
```

Data Integration Service は以下のデータを返します。

CUST_ID	DATE	AMOUNT	FIRST_NAME	LAST_NAME
00002	6/3/2000	55.79	Dinah	Jones
00002	6/10/2000	104.45	Dinah	Jones
00001	6/10/2000	255.56	Marvin	Chi
00004	6/15/2000	534.95	J.	Marks
00002	6/21/2000	98.65	Dinah	Jones

Data Integration Service は、顧客 ID が一致する行を返します。6 月に何も購入していない顧客は含まれません。非登録顧客による購入も含まれません。

レフトアウタージョイン構文

ジョインオーバーライドでレフトアウタージョインを作成することができます。1 つのジョインオーバーライドに複数のレフトアウタージョインを入力することができます。他のジョインと共にレフトアウタージョインを使用する場合、文においては、レフトアウタージョインをすべてまとめて Normal ジョインの後に一覧表示します。

レフトアウタージョインを作成するには、以下の構文を使用します。

```
{ source1 LEFT OUTER JOIN source2 on join_condition }
```

以下の表に、ジョインオーバーライド内のレフトアウタージョインの構文を示します。

構文	説明
<i>source1</i>	ソースリソース名。レフトアウタージョインの場合、Data Integration Service はこのリソースのすべての行を返します。
<i>source2</i>	ソースリソース名。Data Integration Service は、このリソースからジョイン条件に一致する行を返します。
<i>join_condition</i>	ジョイン条件。ソースデータベースがサポートしている構文を使用してください。複数のジョイン条件を AND 演算子によって組み合わせることができます。

たとえば、[「Normal ジョイン構文」 \(ページ 78\)](#)と同じ REG_CUSTOMER および PURCHASES テーブルを使用した場合、以下のジョインオーバーライドによって、6 月に何か購入した顧客の数を求めることができます。

```
{ REG_CUSTOMER LEFT OUTER JOIN PURCHASES on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID }
```

Data Integration Service は以下のデータを返します。

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT
00001	Marvin	Chi	6/10/2000	255.56
00002	Dinah	Jones	6/3/2000	55.79
00003	John	Bowden	NULL	NULL
00004	J.	Marks	6/15/2000	534.95
00002	Dinah	Jones	6/10/2000	104.45

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT
00002	Dinah	Jones	6/21/2000	98.65

Data Integration Service は REG_CUSTOMERS テーブル内の登録済み顧客をすべて返し、6 月に何も購入しなかった顧客には NULL 値を使用します。非登録顧客による購入は含まれません。

複数のジョイン条件を使用して、6 月において 1 回の購入金額が\$100.00 を超える登録済み顧客の人数を求めることができます。

```
{REG_CUSTOMER LEFT OUTER JOIN PURCHASES on (REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID AND PURCHASES.AMOUNT > 100.00) }
```

Data Integration Service は以下のデータを返します。

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT
00001	Marvin	Chi	6/10/2000	255.56
00002	Dinah	Jones	6/10/2000	104.45
00003	John	Bowden	NULL	NULL
00004	J.	Marks	6/15/2000	534.95

同じ期間の返品に関する情報を組み込みたい場合、複数のレフトアウタージョインを使用することができます。たとえば、RETURNS テーブルには以下のデータが含まれているとします。

CUST_ID	CUST_ID	RETURN
00002	6/10/2000	55.79
00002	6/21/2000	104.45

6 月に何か購入および返品した顧客の人数を求めるために、2 つのレフトアウタージョインを使用します。

```
{ REG_CUSTOMER LEFT OUTER JOIN PURCHASES on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID LEFT OUTER JOIN RETURNS on REG_CUSTOMER.CUST_ID = PURCHASES.CUST_ID }
```

Data Integration Service は以下のデータを返します。

CUST_ID	FIRST_NAME	LAST_NAME	DATE	AMOUNT	RET_DATE	RETURN
00001	Marvin	Chi	6/10/2000	255.56	NULL	NULL
00002	Dinah	Jones	6/3/2000	55.79	NULL	NULL
00003	John	Bowden	NULL	NULL	NULL	NULL
00004	J.	Marks	6/15/2000	534.95	NULL	NULL
00002	Dinah	Jones	6/10/2000	104.45	NULL	NULL
00002	Dinah	Jones	6/21/2000	98.65	NULL	NULL
00002	Dinah	Jones	NULL	NULL	6/10/2000	55.79
00002	Dinah	Jones	NULL	NULL	6/21/2000	104.45

Data Integration Service は、欠落値には NULL を使用します。

ライトアウタージョイン構文

ジョインオーバーライドでライトアウタージョインを作成することができます。ライトアウタージョインは、ジョイン構文でリソースの順序を逆にした場合、レフトアウタージョインと同じ結果を返します。ジョインオーバーライドには、ライトアウタージョインを 1 つだけ使用します。複数のライトアウタージョインを作成する場合、ソースリソースの順序を逆にして、結合タイプをレフトアウタージョインに変更してみてください。

他のジョインと共にライトアウタージョインを使用する場合、ジョインオーバーライドの最後にライトアウタージョインを入力します。

ライトアウタージョインを作成するには、以下の構文を使用します。

```
{ source1 RIGHT OUTER JOIN source2 on join_condition }
```

以下の表に、ジョインオーバーライドでのライトアウタージョインの構文を示します。

構文	説明
<i>source1</i>	ソースリソース名。Data Integration Service は、このリソースからジョイン条件に一致する行を返します。
<i>source2</i>	ソースリソース名。ライトアウタージョインの場合、Data Integration Service はこのリソースのすべての行を返します。
<i>join_condition</i>	ジョイン条件。ソースデータベースがサポートしている構文を使用してください。複数のジョイン条件を AND 演算子によって組み合わせることができます。

マッピング実行前および実行後の SQL コマンド

SQL コマンドをカスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスで作成できます。Data Integration Service は、ソースのリレーショナルリソースに対して SQL コマンドを実行します。

マッピングを実行すると、Data Integration Service はソースデータベースに対してマッピング実行前の SQL コマンドを実行した後、ソースを読み込みます。また、ターゲットへ書き込みをした後にソースデータベースに対してマッピング実行後 SQL コマンドを実行します。

マッピング実行前および実行後の SQL コマンドを設定するときは以下のガイドラインに従います。

- そのデータベースタイプで有効な任意のコマンドを使用します。Data Integration Service では、データベース側で許可されている場合も、ネストされたコメントは使用できません。
- 複数の文を区切るにはセミコロン (;) を使用します。Data Integration Service は各文の後にコミットを発行します。
- Data Integration Service は、/* ... */内のセミコロンは無視されます。
- コメントの外部でセミコロンを使用する必要がある場合は、バックスラッシュ (\) でセミコロンをエスケープします。セミコロンをエスケープすると、Data Integration Service はバックスラッシュを無視し、セミコロンを文の区切り文字として使用しません。
- Developer ツールはマッピング実行前および実行後の SQL コマンドを検証しません。

マッピング実行前/実行後 SQL コマンドの追加

マッピング実行前および実行後の SQL コマンドを、カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスに追加できます。マッピングでこのデータオブジェクトを使用すると、Data Integration Service で SQL コマンドが実行されます。

1. カスタマイズデータオブジェクトを開きます。
2. **【読み取り】** ビューを選択します。
3. 出力トランスフォーメーションを選択します。
4. **【詳細】** プロパティを選択します。
5. **【PreSQL】** フィールドにマッピング実行前 SQL コマンドを入力します。
6. **【PostSQL】** フィールドにマッピング実行後 SQL コマンドを入力します。
7. カスタマイズデータオブジェクトを選択します。

カスタムクエリの作成

カスタムクエリを作成し、ソースからデータを読み取るための特別な SELECT 文を発行します。カスタムクエリは、データ統合サービスがソースデータを読み取るためのデフォルトクエリをオーバーライドします。

1. カスタマイズデータオブジェクトまたはリレーショナルデータオブジェクトインスタンスを開きます。
2. **【読み取り】** ビューを選択します。
3. 出力トランスフォーメーションを選択します。
4. **【クエリ】** プロパティを選択します。
5. 詳細クエリを選択します。
6. **【カスタムクエリーを使用】** を選択します。
データ統合サービスがソースデータを読み取るために発行するクエリが表示されます。
7. クエリを変更するか、カスタムクエリで置き換えます。
8. カスタムクエリをリレーショナルデータソースにプッシュする場合は、**【カスタムクエリをデータベースにプッシュ】** を選択します。
データ統合サービスは、デフォルトではカスタムクエリをデータベースにプッシュしません。クエリがデータベースに対して有効なサブクエリを形成する場合は、カスタムクエリをプッシュするようにしてください。
9. データオブジェクトを保存します。

非リレーショナルデータオブジェクト

非リレーショナルデータオブジェクトは、インポートしてマッピング、マプレット、またはプロファイルで使います。非リレーショナルデータオブジェクトは、非リレーショナルデータソースを使用する物理データオブジェクトです。

非リレーショナルデータオブジェクトは、以下の接続タイプにインポートできます。

- Adabas
- IMS
- シーケンシャル

- VSAM

非リレーショナルデータオブジェクトをインポートする場合、Developer ツールは PowerExchange データマップからオブジェクトのメタデータを読み取ります。データマップは非リレーショナルレコードをリレーショナルテーブルにマッピングし、データへのアクセスに SQL 言語を使用できるようにします。データマップを作成するには、PowerExchange Navigator を使用します。

オブジェクトをインポートした後、そのオブジェクトの非リレーショナル操作を読み取り、書き込み、またはルックアップトランスフォーメーションとしてマッピングおよびマプレットに追加できます。各非リレーショナル操作は、データマップが定義するリレーショナルテーブルに対応します。1 つ以上の非リレーショナルレコードのフィールドからリレーショナルテーブルのカラムへのマッピングを表示するには、**[Object Explorer]** ビューで非リレーショナル操作をダブルクリックします。

データマップの詳細については、『*PowerExchange Navigator Guide*』を参照してください。

注: Informatica 9.0.1 で作成した非リレーショナルデータオブジェクトを使用する場合、操作を実行する前にアップグレードする必要があります。非リレーショナルデータオブジェクトをアップグレードするには、`infacmd pwx UpgradeModels` コマンドを実行します。

非リレーショナルデータオブジェクトのインポート

非リレーショナルデータオブジェクトは、インポートしてマッピング、マプレット、またはプロファイルで使用します。

非リレーショナルデータオブジェクトをインポートする前に、データベースまたはデータセットへの接続を設定する必要があります。オブジェクトのデータマップを作成する必要もあります。

1. **[Object Explorer]** ビューで、プロジェクトまたはフォルダーを選択します。
2. **[ファイル]** > **[新規]** > **[データオブジェクト]** をクリックします。
3. **[非リレーショナルデータオブジェクト]** を選択して、**[次へ]** をクリックします。
[新しい非リレーショナルデータオブジェクト] ダイアログボックスが表示されます。
4. 物理データオブジェクトの名前を入力します。
5. **[接続]** オプションの横にある **[参照]** をクリックして、接続を選択します。
6. **[データマップ]** オプションの横にある **[参照]** をクリックし、インポートするデータマップを選択します。
[リソース] 領域に、データマップが定義しているリレーショナルテーブルの一覧が表示されます。
7. 必要に応じて、**[リソース]** 領域でテーブルの追加または削除を行えます。
8. **[完了]** をクリックします。

非リレーショナルデータオブジェクトおよびその非リレーショナル操作が、**[Object Explorer]** ビューのプロジェクトまたはフォルダーにある **[物理データオブジェクト]** の下に表示されます。

注: **[接続エクスプローラ]** ビューを使用して、非リレーショナルデータオブジェクトをインポートすることもできます。

非リレーショナルデータ操作からの読み取り、書き込み、またはルックアップトランスフォーメーションの作成

非リレーショナルデータ操作は、読み取り、書き込み、またはルックアップトランスフォーメーションとして、マッピングまたはマプレットに追加できます。

1. 読み取り、書き込み、またはルックアップトランスフォーメーションを作成するマッピングまたはマプレットを開きます。

2. **[Object Explorer]** ビューで、非リレーショナルデータオブジェクトを 1 つ以上選択します。
3. 非リレーショナルデータ操作をマッピングエディタにドラッグします。
[マッピングに追加] ダイアログボックスが表示されます。
4. **[読み取り]**、**[書き込み]**、または **[ルックアップ]** オプションを選択します。
[独立したデータオブジェクトとして] が自動的に選択されます。
5. **[OK]** をクリックします。
非リレーショナルデータ操作ごとに、読み取り、書き込み、またはルックアップトランスフォーメーションが作成されます。

WSDL データオブジェクトの概要 WSDL データオブジェクト

WSDL データオブジェクトは、WSDL ファイルをソースとして使用する物理データオブジェクトです。WSDL データオブジェクトを使用して、Web サービスまたは Web サービスコンシューマトランスフォーメーションを作成できます。WSDL データオブジェクトを作成するには、WSDL ファイルをインポートします。

WSDL データオブジェクトをインポートした後に、**概要ビュー**と**詳細ビュー**で、全般プロパティと詳細プロパティを編集できます。**WSDL** ビューには、WSDL ファイルのコンテンツが表示されます。

WSDL をインポートする際には、以下のガイドラインを考慮してください。

- WSDL ファイルは WSDL 1.1 に準拠している必要があります。
- WSDL ファイルは有効でなければなりません。
- Web サービスまたは Web サービスコンシューマトランスフォーメーションに組み込む操作が、ドキュメント/リテラルのエンコーディングを使用している必要があります。WSDL ファイル内のすべての操作がドキュメント/リテラル以外のエンコーディングタイプを使用している場合、WSDL のインポートは失敗します。
- WSDL ファイルで参照されているすべてのスキーマに Developer ツールからアクセスできる必要があります。
- WSDL ファイルにスキーマが含まれている場合または外部スキーマがある場合は、Developer ツールで WSDL データオブジェクト内に埋め込みスキーマが作成されます。
- WSDL ファイルに別の WSDL ファイルをインポートすると、Developer ツールによって両方の WSDL が結合されて、WSDL データオブジェクトが作成されます。
- WSDL ファイルに複数の操作が定義されている場合は、Developer ツールによってすべての操作が WSDL データオブジェクトに組み込まれます。WSDL データオブジェクトから Web サービスを作成する場合は、1 つ以上の操作を組み込むことができます。

WSDL データオブジェクトの概要ビュー

WSDL データオブジェクトの概要ビューには、WSDL および WSDL 内の操作についての全般的な情報が表示されます。

以下の表に、WSDL データオブジェクトに対して設定する全般プロパティを示します。

プロパティ	説明
名前	WSDL データオブジェクトの名前。
説明	WSDL データオブジェクトの説明。

以下の表に、WSDL データオブジェクトで定義される操作のカラムを示します。

プロパティ	説明
操作	WSDL で操作のメッセージ形式とプロトコルが定義されている場所。
入力	操作入力に関連付けられた WSDL メッセージ名。
出力	操作出力に関連付けられた WSDL メッセージ名。
フォールト	操作フォールトに関連付けられた WSDL メッセージ名。

WSDL データオブジェクトの詳細ビュー

WSDL データオブジェクトの詳細ビューには、WSDL データオブジェクトの詳細プロパティが表示されます。

以下の表に、WSDL データオブジェクトの詳細プロパティを示します。

プロパティ	説明
Connection	Web サービスコンシューマトランスフォーメーション用のデフォルトの Web サービス接続。
ファイルの場所	WSDL ファイルが存在する場所。

WSDL データオブジェクトのインポート

WSDL から Web サービスを作成する場合、または Web サービスコンシューマトランスフォーメーションを作成する場合は、WSDL データオブジェクトをインポートします。WSDL データオブジェクトは、WSDL ファイルから、または WSDL の場所を指す URI からインポートできます。WSDL データオブジェクトは、SOAP 1.1 バインディング操作と SOAP 1.2DL バインディング操作またはそのいずれかを含む WSDL ファイルからインポートできます。

1. **【ファイル】 > 【新規】 > 【データオブジェクト】** をクリックします。
2. **【WSDL データオブジェクト】** を選択し、**【次へ】** をクリックします。
【新しい WSDL データオブジェクト】 ダイアログボックスが表示されます。
3. **【WSDL】** オプションの横にある **【参照】** をクリックして、WSDL の場所を入力します。次に、**【OK】** をクリックします。

WSDL の場所を入力する際に、WSDL ファイルを参照するか、または WSDL の URI を入力できます。

注: URI に英語以外の文字が含まれていると、インポートが失敗する場合があります。任意のブラウザのアドレスバーに URI をコピーします。ブラウザから場所をコピーします。Developer ツールは、エンコードされた URI をブラウザから受け付けます。

4. WSDL の名前を入力します。
5. **【場所】** オプションの横にある **【参照】** をクリックして、WSDL データオブジェクトをインポートするプロジェクトまたはフォルダの場所を選択します。
6. **【次へ】** をクリックして、WSDL 内の操作を表示します。
7. **【完了】** をクリックします。

Object Explorer ビューで、プロジェクトまたはフォルダー内の **【物理データオブジェクト】** の下にデータオブジェクトが表示されます。

WSDL の同期

WSDL ファイルを変更したときに WSDL データオブジェクトを同期できます。WSDL データオブジェクトの同期化では、WSDL ファイルからオブジェクトメタデータが再インポートされます。

WSDL データオブジェクトを使用して、Web サービスまたは Web サービスコンシューマトランスフォーメーションを作成できます。WSDL データオブジェクトを更新すると、その WSDL を参照するオブジェクトが更新され、オブジェクトを開くと更新済みとしてマークされています。Developer ツールが新旧の WSDL を比較するときは、名前属性によって WSDL コンポーネントを識別します。

名前属性が変更されていない場合、WSDL コンポーネントを参照するオブジェクトが更新されます。たとえば、WSDL ファイルを編集して、単純要素 "CustID" の型を xs:string から xs:integer に変更します。WSDL データオブジェクトを同期すると、その CustID 要素を参照するすべての Web サービスおよび Web サービスコンシューマトランスフォーメーションの要素型が更新されます。

名前属性が変更された場合、その WSDL コンポーネントを参照するオブジェクトを開くと変更済みとしてマークされています。例えば、WSDL を編集して、要素の名前を「Resp」から「RespMsg」に変更します。それから、WSDL の同期を実行します。その要素を参照する Web サービスを開くと、その要素名はエディタでアスタリスク付きになり、Web サービスに変更が含まれることがわかります。Developer ツールは Web サービスの要素名を更新しますが、新しい要素をポートにマップする方法を特定できません。Resp 要素が入力トランスフォーメーションまたは出力トランスフォーメーションのポートにマップされている場合、RespMsg 要素を適切なポートにマップする必要があります。

Developer ツールによって、WSDL データオブジェクトを更新する前に WSDL ファイルが検証されます。WSDL ファイルにエラーがある場合、Developer ツールはファイルをインポートしません。

WSDL データオブジェクトの同期

WSDL ファイルを変更したときに WSDL データオブジェクトを同期します。

1. **【Object Explorer】** ビューの WSDL データオブジェクトを右クリックして、**【同期】** を選択します。
【WSDL データオブジェクトの同期化】 ダイアログボックスが表示されます。
2. **【WSDL】** フィールドの横にある **【参照】** をクリックして、WSDL の場所を入力します。次に、**【OK】** をクリックします。

WSDL の場所を入力する際に、WSDL ファイルを参照するか、または WSDL の URI を入力できます。

注: URI に英語以外の文字が含まれていると、インポートが失敗する場合があります。任意のブラウザのアドレスバーに URI をコピーします。ブラウザから場所をコピーします。Developer ツールは、エンコードされた URI をブラウザから受け付けます。

3. WSDL の名前と場所を確認します。

4. **【次へ】** をクリックして、WSDL 内の操作を表示します。
5. **【完了】** をクリックします。

WSDL を参照するオブジェクトが更新され、オブジェクトを開くと更新済みとマークされています。

証明書管理

Developer ツールでは、クライアント認証を必要とする URL から WSDL データオブジェクトとスキーマオブジェクトをインポートするために、証明書を使用する必要があります。

クライアント認証を必要とする URL をホストするサーバーが信頼されている証明書を使用している場合は、デフォルトで Developer ツールがその URL からオブジェクトをインポートします。この URL をホストするサーバーが信頼されていない証明書を使用している場合は、その証明書を Developer ツールに追加する必要があります。信頼されていない証明書を Developer ツールに追加しないと、Developer ツールではオブジェクトをインポートできません。オブジェクトのインポート元となる URL のサーバー管理者に、証明書ファイルとパスワードを要求してください。

Developer ツールに追加した証明書は、Developer ツールマシン上で実行するインポートに適用されます。Developer ツールでは、証明書がモデルリポジトリに保管されません。

Informatica Developer の証明書プロパティ

クライアント認証を必要とする URL から、信頼されていない証明書を使用してオブジェクトをインポートする場合は、Developer ツールに証明書を追加します。

以下の表に、証明書のプロパティを示します。

プロパティ	説明
ホスト名	URL をホストするサーバーの名前。
ポート番号	URL のポート番号。
証明書ファイルのパス	クライアント証明書ファイルの場所。
パスワード	クライアント証明書ファイルのパスワード。

Informatica Developer への証明書の追加

証明書を追加する場合は、証明書プロパティを設定します。これは、クライアント認証を必要とする URL から信頼されていない証明書を使用してオブジェクトをインポートする場合に、Developer ツールで 사용되는ものです。

1. **【ウィンドウ】** > **【設定】** をクリックします。
2. **【Informatica】** > **【Web サービス】** > **【証明書】** を選択します。
3. **【追加】** をクリックします。
4. 証明書プロパティを設定します。
5. **【OK】** をクリックします。

同期化

物理データオブジェクトのソースが変更されたときに、同期させることができます。物理データオブジェクトの同期化では、選択したソースからオブジェクトメタデータが再インポートされます。

すべての物理データオブジェクトを同期できます。リレーショナルデータオブジェクトまたはカスタマイズデータオブジェクトを同期する際、Developer ツールで定義したキーリレーションを保持するか上書きすることができます。

ソースの変更時に同期するようにカスタマイズデータオブジェクトを設定できます。例えば、カスタマイズデータオブジェクトでリレーショナルデータオブジェクトがソースとして使用されている場合に、そのリレーショナルデータオブジェクトにカラムを追加すると、Developer ツールはそのカラムをカスタマイズデータオブジェクトに追加します。ソースの変更時にカスタマイズデータオブジェクトを同期するには、カスタマイズデータオブジェクトの【概要】プロパティで【入力と出力を同期】オプションを選択します。

物理データオブジェクトを同期するには、【Object Explorer】ビューで右クリックして、【同期】を選択します。

フラットファイルデータオブジェクトの同期

Informatica Developer で、外部フラットファイルデータソースに加えられた変更をそのデータオブジェクトに同期させることができます。データオブジェクトを同期させるには、【フラットファイルの同期】ウィザードを使用します。

1. 【オブジェクトエクスプローラ】ビューで、フラットファイルデータオブジェクトを選択します。
2. 右クリックし、【同期】を選択します。
【フラットファイルデータオブジェクトの同期】ウィザードが表示されます。
3. 【既存のフラットファイルの選択】フィールドのフラットファイルパスを確認します。
4. 【次へ】をクリックします。
5. 必要に応じ、コードページ、形式、区切り形式プロパティ、およびカラムプロパティを選択します。
6. 【完了】をクリックし、【OK】をクリックします。

リレーショナルデータオブジェクトの同期

Informatica Developer で、リレーショナルデータソースの外部データソース変更をそのデータオブジェクトに同期させることができます。外部データソース変更には、カラムの追加、変更、削除や、ルールの変更などがあります。

1. 【Object Explorer】ビューで、リレーショナルデータオブジェクトを選択します。
2. 右クリックし、【同期】を選択します。
操作の確認を求めるメッセージが表示されます。
3. 同期プロセスを実行するには、【OK】をクリックします。このプロセスを取り消すには、【キャンセル】をクリックします。
【OK】をクリックすると、同期プロセスステータスメッセージが表示されます。
4. 「同期が完了しました」というメッセージが表示された時点で、【OK】をクリックします。
このメッセージには、データオブジェクトに加えられたメタデータ変更の概要が示されます。

物理データオブジェクトのトラブルシューティング

リレーショナルデータオブジェクトまたはカスタマイズデータオブジェクトのソーストランスフォーメーションのプレビューが失敗する。

リソースの所有者名が正しいことを確認します。

リレーショナルデータオブジェクトをインポートする際、ユーザー名とテーブルのインポート元のスキーマが一致しない場合、Developer tool は所有者名をインポートします。ユーザー名とテーブルのインポート元スキーマが一致しても、データベースのデフォルトスキーマ名が異なる場合、プレビューは失敗します。Data Integration Service が、テーブルが存在しないデータベースのデフォルトスキーマに対するクエリをプレビューするためです。

リレーショナルデータオブジェクトまたはソーストランスフォーメーションを更新し、正しいリソース所有者名を入力します。所有者名は、リレーショナルデータオブジェクトまたはソーストランスフォーメーションの【詳細】プロパティに表示されます。

第 6 章

フラットファイルデータオブジェクト

この章では、以下の項目について説明します。

- [フラットファイルデータオブジェクトの概要, 90 ページ](#)
- [ソースファイル名の生成, 91 ページ](#)
- [フラットファイルデータオブジェクトの概要プロパティ, 91 ページ](#)
- [フラットファイルデータオブジェクトの詳細プロパティ, 92 ページ](#)
- [制御ファイル, 100 ページ](#)
- [実行時のカラムの更新, 101 ページ](#)
- [制御ファイルからのカラムメタデータの生成, 102 ページ](#)
- [フラットファイルデータオブジェクトの作成, 104 ページ](#)

フラットファイルデータオブジェクトの概要

マッピング、マップレット、またはプロファイルに含めるフラットファイル物理データオブジェクトを作成します。フラットファイルデータオブジェクトは、読み取り、書き込み、およびルックアップトランスフォーメーションとして、マッピングおよびマップレットに追加できます。フラットファイルデータオブジェクトに対してプロファイルを作成し、実行できます。

フラットファイルデータオブジェクトは区切り形式または固定長です。バイナリデータを含まない固定長および区切り形式のフラットファイルからフラットファイルデータオブジェクトを作成できます。

実行時にフラットファイルデータソースの変更に対応するようにフラットファイルデータオブジェクトを設定できます。また、フラットファイルまたは制御ファイルの情報を使用して、フラットファイルデータオブジェクトのカラム名を生成することもできます。制御ファイルには、カラム名、精度、スケール、処理バイト数などの情報が含まれます。

フラットファイルデータオブジェクトを作成したら、以下の各ビューを使用してフラットファイルプロパティを設定します。

【概要】 ビュー

フラットファイルデータオブジェクトの名前と説明を設定し、カラムプロパティを更新します。

【パラメータ】 ビュー

フラットファイルデータオブジェクトのパラメータを作成します。

【詳細】 ビュー

データ統合サービスがフラットファイルに対してデータの読み取りや書き込みを実行するときに使用する、フォーマットプロパティとランタイムプロパティを設定します。

フラットファイルデータオブジェクトを読み取りまたは書き込みトランスフォーメーションとしてマッピングに追加すると、**【プロパティ】** ビューでフラットファイルデータオブジェクトのフォーマットプロパティとランタイムプロパティを確認できます。 マッピング内のフラットファイルプロパティは編集できません。

ソースファイル名の生成

ファイル名カラムをフラットファイルデータオブジェクトに追加できます。ファイル名カラムは、特定のデータレコードが含まれるソースファイルを特定するのに役立ちます。これは、データソースが複数ある場合に便利です。

ファイル名カラムはフラットファイルデータオブジェクトの**【概要】** ビューで設定できます。ファイル名カラムは、フラットファイルデータオブジェクトのオプションのカラムです。フラットファイルデータオブジェクトのファイル名カラムポートを使用して、ソースファイル名を各ソース行に書き込むようにマッピングを設定できます。ファイル名カラムには、完全修飾パスおよびファイル名が含まれます。

ファイル名カラムポートはソースデータオブジェクトの最後のカラムとして表示されます。ソースデータオブジェクトに追加できるファイル名カラムポートは1つのみです。ポートに「FileName」という名前が付いていて、ファイル名カラムポートの作成を試みると、ファイル名カラムポートは「FileName1」という名前になります。

例えば、マッピングでファイルのリストが含まれるソースファイルが使用されているとします。ファイルリスト内のファイル名は組織の部署名を参照しています。例えば、SYSA_Finance.txt ファイルには経理部のデータが含まれています。このマッピングでは、文字列関数を使用してファイル名カラムの出力から部署名を抽出できます。抽出された部署名を使用して、各部署ごとにデータを処理することができます。

フラットファイルデータオブジェクトの概要プロパティ

Data Integration Service は、フラットファイルに対してデータの読み取りや書き込みを行う際に概要プロパティを使用します。概要プロパティには全般プロパティが含まれ、フラットファイルデータオブジェクトに適用されます。カラムのプロパティも含まれ、フラットファイルデータオブジェクト内のカラムに適用されます。フラットファイルの概要プロパティは、Developer ツールにより**【概要】** ビューに表示されます。

以下の表に、フラットファイルに設定する全般プロパティを示します。

プロパティ	説明
名前	フラットファイルデータオブジェクトの名前。
説明	フラットファイルデータオブジェクトの説明。

以下の表に、フラットファイルに設定するカラムプロパティを示します。

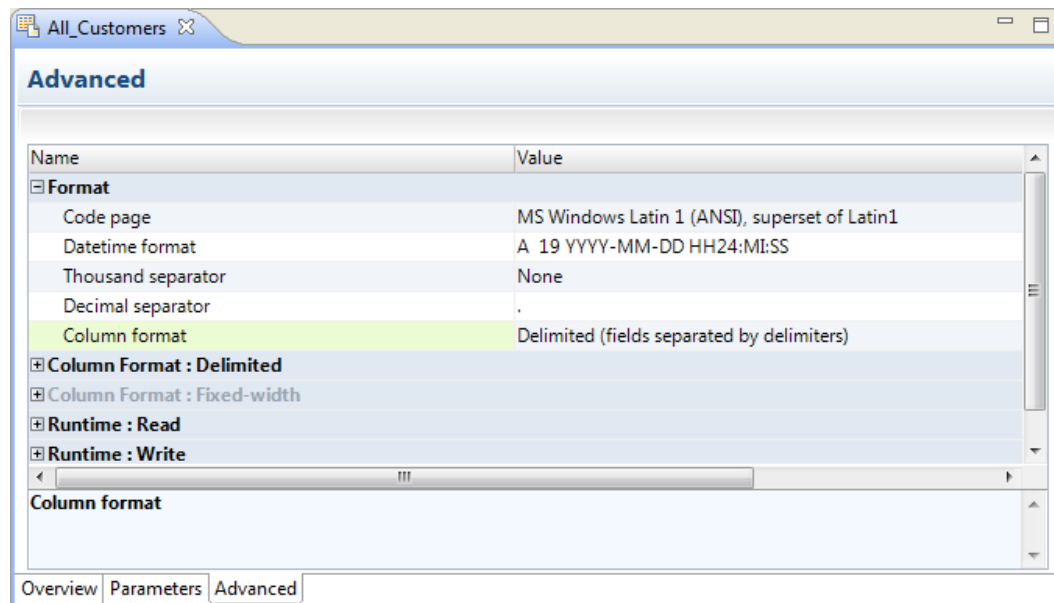
プロパティ	説明
名前	カラムまたはファイル名カラムポートの名前。
ネイティブタイプ	カラムのネイティブタイプ。
処理するバイト数（固定長フラットファイル）	データ統合サービスがカラムに対して読み取りまたは書き込みを行うバイト数。
精度	数値データ型の最大有効桁数、または文字列データ型の最大文字数です。数値データ型の場合、精度にはスケールが含まれます。
スケール	数値の小数点以下の最大桁数です。
形式	数値データ型および日時データ型のカラム形式。 数値データ型の場合、1000 ごとの区切りと小数点記号を定義します。デフォルトは 1000 ごとの区切りなしで、小数点記号はピリオドです (.)。 数値データ型の場合、年、月、日、および時間の表示形式を定義します。また、フィールド幅も定義します。デフォルトは、「A 19 YYYY-MM-DD HH24:MI:SS」です。
可視性	データ統合サービスがカラムに対してデータの読み取りおよび書き込みを実行できるかどうかを決定します。 例えば、可視性が「読み取り」の場合、データ統合サービスはカラムからデータを読み取ることができます。データをカラムに書き込むことはできません。 フラットファイルデータオブジェクトの場合、このプロパティは読み取り専用です。可視性は常に「読み取りおよび書き込み」です。
説明	カラムの説明。

フラットファイルデータオブジェクトの詳細プロパティ

データ統合サービスは、フラットファイルに対してデータの読み取りや書き込みを行う際に詳細プロパティを使用します。詳細プロパティには、フォーマットおよびランタイムプロパティが含まれています。フラットファイルの詳細プロパティは、Developer tool により **【詳細】** ビューに表示されます。

【詳細】 ビューには、折りたたみと展開が可能なプロパティセクションが含まれています。表示されるカラム形式セクションは、区切りカラム形式または固定長カラム形式のどちらを設定したかによって異なります。

次の図は、[詳細] ビューのプロパティセクションを示しています。



フォーマットのプロパティ

Developer tool では、[詳細] ビューの【フォーマット】セクションにフラットファイルデータオブジェクトのフォーマットプロパティが表示されます。

以下の表に、フラットファイルデータオブジェクトに設定するフォーマットプロパティを示します。

プロパティ	説明
コードページ	フラットファイルデータオブジェクトのコードページ。 ソースファイルの場合、ターゲットコードページのサブセットであるソースコードページを指定します。ルックアップファイルの場合、ソースコードページのスーパーセットかつターゲットコードページのサブセットであるコードページを指定します。ターゲットファイルの場合、ソースコードページのスーパーセットであるコードページを指定します。 デフォルトは、「MS Windows Latin 1 (ANSI), superset of Latin 1」です。
日時フォーマット	日時の値の表示形式およびフィールド幅を定義します。 デフォルトは、「A 19 YYYY-MM-DD HH24:MI:SS」です。
1000 ごとの区切り文字	数値に対する 1000 の区切り記号。 デフォルトは「なし」です。
小数位の区切り文字	数値の小数点記号。 デフォルトはピリオド「.」です。
カラム形式	フラットファイルのフォーマット。区切り形式または固定長です。

カラム形式: 区切りプロパティ

フラットファイルが区切り形式であれば、Developer tool は【詳細】ビューの【カラム形式: 区切り記号付き】セクションに区切りプロパティを表示します。

以下の表に、フラットファイルデータオブジェクトに設定する区切りプロパティを示します。

プロパティ	説明
区切り文字	データのカラムを区切るために使用される文字。 区切り文字フィールドをクリックして文字を選択するかパラメータをプロパティに割り当てます。区切り文字は印字可能であり、かつテキスト修飾子やエスケープ文字を選択した場合にはそれとは異なる文字である必要があります。 デフォルトは [カンマ] です。
テキスト修飾子	引用符はテキスト文字列の境界を定義します。引用符を選択すると、引用符で囲まれた文字列内の区切り文字が無視されます。 デフォルトは [引用符なし] です。
次の行からインポートを開始	データ統合サービスがデータのインポートを開始する行。ヘッダ行をスキップする場合には、このオプションを選択します。 デフォルトは 1 です。
行の区切り文字	データ行を区切る文字の 8 進コード。 デフォルトはラインフィード、\012 LF (\n) です。
エスケープ文字	区切り文字がエスケープ文字の直後にある場合に、引用符で囲まれていない文字列の区切り文字をエスケープする文字。エスケープ文字を指定すると、データ統合サービスは、文字列に含まれる通常の文字として区切り文字を読み込みます。 注: ソースファイルに引用符やエスケープ文字が含まれていないと、マッピングのパフォーマンスをある程度向上できます。
データのエスケープ文字を維持	出力文字列にエスケープ文字を含めます。 デフォルトでは無効になっています。
連続した区切り文字を 1 文字として扱う	データ統合サービスで、1 つ以上の連続したカラム区切り文字が 1 つのカラム区切り文字として扱われます。このオプションを選択しない場合、データ統合サービスは 2 つの連続した区切り文字を 1 つの NULL 値として読み取ります。 デフォルトでは無効になっています。

カラム形式: 固定幅プロパティ

フラットファイルが固定幅の場合、Developer tool の【詳細】ビューの【カラム形式: 固定幅】セクションに固定幅プロパティが表示されます。

以下の表に、フラットファイルデータオブジェクトに設定する固定幅プロパティを示します。

プロパティ	説明
NULL 文字タイプ	NULL 文字タイプ。テキストまたはバイナリです。
NULL 文字値	NULL 値を表す文字。NULL 文字には、ファイルコードページの任意の有効な文字、または 0 - 255 の任意のバイナリ値を指定できます。

プロパティ	説明
NULL 文字を繰り返す	<p>ソースファイルの場合、データ統合サービスは繰り返し NULL 文字を 1 つの NULL 値として単一フィールドで読み取ります。</p> <p>ターゲットファイルの場合、データ統合サービスは可能な限り多くの NULL 文字をターゲットフィールドに書き込みます。このオプションを有効にしない場合、データ統合サービスはフィールドの最初に 1 つの NULL 文字を入力して NULL 値を表します。</p> <p>デフォルトでは無効になっています。</p>
次の行からインポートを開始	<p>データ統合サービスがデータのインポートを開始する行。ヘッダ行をスキップする場合には、このオプションを選択します。</p> <p>デフォルトは 1 です。</p>
レコード間でスキップするバイト数	<p>行の最終カラムと次の行の最初のカラムの間のバイト数。データ統合サービスは、指定されたバイト数を行の最後でスキップして、復帰文字や改行文字を読み込まないようにします。</p> <p>UNIX ファイルの場合は「1」、DOS ファイルの場合は「2」を入力します。</p> <p>デフォルトは 2 です。</p>
行順序あり	<p>データ統合サービスは、最後のカラムの改行文字または復帰文字をこのカラムの末尾として読み取ります。ファイルで改行または復帰を使用して各行の最後のカラムを短くしている場合に、このオプションを選択します。</p> <p>デフォルトでは無効になっています。</p>
後続のスペースを削除する	<p>末尾の空白を文字列値から削除します。</p> <p>デフォルトでは無効になっています。</p>
ユーザー定義シフトステート	<p>[カラム] プロパティのソースカラムのシフトステートを選択できるようにします。</p> <p>ソースファイルにマルチバイトとシングルバイトデータの両方が含まれ、シフトインとシフトアウトキーが含まれない場合にこのオプションを選択します。マルチバイトファイルソースにシフトキーが含まれない場合、フラットファイルデータオブジェクトで各カラムにシフトキーを選択する必要があります。データ統合サービスが各文字を正しく読み取れるように、各カラムにシフトキーを選択します。</p> <p>デフォルトでは無効になっています。</p>

ランタイム:読み取りプロパティ

Developer tool では、[詳細] ビューの [ランタイム: 読み取り] セクションに、フラットファイルソースのランタイムプロパティが表示されます。データ統合サービスは、フラットファイルからデータを読み取る際にこの情報を使用します。

以下の表に、フラットファイルデータオブジェクトに設定する読み取りプロパティを示します。

プロパティ	説明
入力タイプ	ソース入力のタイプ。次のソース入力のタイプを選択できます。 <ul style="list-style-type: none">- ファイル。フラットファイルのソースの場合に選択します。- コマンド。シェルコマンドによって生成されたソースデータまたはファイルリスト。
ソースタイプ	同じファイルプロパティを持つファイルのソースタイプを示します。次のいずれかのソースタイプを選択できます。 <ul style="list-style-type: none">- Direct。ソースデータを含むソースファイル。- Indirect。ファイルのリストを含むソースファイル。データ統合サービスはファイルリストを読み取り、ファイル順に読み取ります。- Directory。ディレクトリにあるソースファイル。ソースファイルのディレクトリプロパティでディレクトリの場所を指定する必要があります。データ統合サービスは、アルファベットの昇順にファイルを読み取ります。データ統合サービスはサブディレクトリのファイルを読み取りません。
ソースファイル名	フラットファイルソースのファイル名。
ソースファイルのディレクトリ	フラットファイルのソースが存在するディレクトリ。Informatica サービスをホストするマシンは、このディレクトリにアクセスできる必要があります。デフォルトは SourceDir システムパラメータです。
同時読み込みのパーティション化	複数のパーティションがソースファイルから入力行を読み取る順序。データ統合サービスがマッピングにパーティションを作成しない場合、この値は無視されます。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- スループットの最適化。データ統合サービスが入力行の順序を保持しません。- 相対順序を維持。データ統合サービスが、パーティションごとに読み込まれる行に対し、入力行の順序を保持します。- 絶対順序を維持。データ統合サービスは、すべてのパーティションで読み込まれるすべての行に対して、入力行の順序を保持します。
接続タイプ	接続のタイプ。次のオプションから選択します。 <ul style="list-style-type: none">- なし。ソースファイルは接続を必要としません。- Hadoop ファイルシステム。ソースファイルは HDFS 内に存在します。 デフォルトは [なし] です。

プロパティ	説明
コマンド	<p>ソースファイルデータを生成するために使用するコマンド。コマンドを使用してフラットファイルデータを生成または変換し、コマンドの標準出力をマッピングの実行時にフラットファイルリーダーに送信できます。フラットファイルリーダーは、標準出力をフラットファイルソースデータとして読み取ります。コマンドを使用してソースデータを生成すると、フラットファイルソースをステージングする必要がなくなります。</p> <p>コマンドまたはスクリプトを使用すると、マッピング前のコマンドを使用してフラットファイルソースを生成する代わりに、データ統合サービスに直接ソースデータを送信できます。コマンドを使用してファイルリストを生成することもできます。</p> <p>例えば、ファイルリストとしてディレクトリリストを使用するには、以下のコマンドを使用します。</p> <pre>cd MySourceFiles; ls sales-records-Sep-*-2005.dat</pre>
文字列の NULL の切り捨て	<p>最初の NULL 文字および最初の NULL 文字に続くすべての文字を文字列値から取り除きます。</p> <p>このオプションは、文字列に NULL 文字を含む区切りフラットファイルに対して有効にします。このオプションを有効にしない場合、データ統合サービスは文字列に NULL 文字を含む行に対して行エラーを生成します。</p> <p>デフォルトでは無効になっています。</p>
行順序ありのバッファ長	<p>データ統合サービスが読み取る 1 行のバイト数。</p> <p>データ統合サービスは、このプロパティと合計の行サイズを使用して、行を削除するかどうかを決定します。行が、行順序ありのバッファ長、または合計の行サイズのどちらか大きい方のサイズを超える場合、データ統合サービスはその行を削除してマッピングログファイルに書き込みます。行のサイズを決定するには、カラムの精度と区切り文字を追加して、その合計に各文字の最大バイト数を掛けます。</p> <p>デフォルトは 1024 です。</p>
ランタイムカラム名の生成	<p>実行時にカラムメタデータを生成する方法を決定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - 自動 (Field1、Field2...)。データ統合サービスは、データ内のカラムの位置に基づいて、カラムメタデータを取り込みます。フラットファイルのカラムメタデータは、フラットファイルデータオブジェクトの形式プロパティの 【開始インポート行】 属性に指定された行を開始位置として、実行時に構築されます。 - データファイルヘッダーから (最初の行)。フラットファイルの先頭行にヘッダ行があり、データ統合サービスはこのヘッダ行を使用してカラムメタデータを決定します。 - 制御ファイルから。データ統合サービスは、カラム名、精度、データ型、スケール、処理バイト数など、制御ファイル内のデータに基づいてカラムメタデータを構築します。
制御ファイル名	制御ファイルの名前。制御ファイルからランタイムカラム名を生成する場合は必須。
制御ファイルディレクトリ	制御ファイルが存在するディレクトリ。制御ファイルからランタイムカラム名を生成する場合は必須。
デフォルトのフィールドタイプ	実行時に生成される追加ポートのデータ型。
デフォルトの精度	実行時に生成される追加ポートの精度。

プロパティ	説明
デフォルトのスケール	実行時に生成される追加ポートのスケール。
制約	<p>データ行の各値が満足する必要がある条件式。式エディタを使用して、TRUE に評価される式を入力します。データ統合サービスは、制約を読み取ると、TRUE に評価されない行を削除します。</p> <p>例えば、ソースフラットファイルに AGE カラムがあるとします。このとき、フラットファイルデータオブジェクトに対して「AGE < 70」という制約を設定できます。データ統合サービスは、ソースフラットファイルから制約「AGE < 70」を満たす行を読み取ります。データ統合サービスは「AGE >= 70」の行を読み取ると、その行を削除します。</p>

ランタイム:書き込みプロパティ

Developer tool では、[詳細] ビューの [ランタイム: 書き込み] セクションに、フラットファイルターゲットのランタイムプロパティが表示されます。データ統合サービスは、フラットファイルに対してデータの書き込みを行う際にこの情報を使用します。

以下の表に、フラットファイルデータオブジェクトに設定する書き込みプロパティを示します。

プロパティ	説明
存在する場合は追加	<p>ターゲットファイルおよび拒否ファイルに出力データを追加します。</p> <p>このオプションを選択しない場合、データ統合サービスはターゲットファイルおよび拒否ファイルを切り詰めてからデータを書き込みます。ファイルが存在しない場合、データ統合サービスによって作成されます。</p> <p>デフォルトでは無効になっています。</p>
存在しない場合はディレクトリを作成	<p>ターゲットディレクトリが存在しない場合は作成します。</p> <p>デフォルトでは無効になっています。</p>
ヘッダーのオプション	<p>ファイルターゲットにヘッダ行を作成します。以下のオプションを選択することができます。</p> <ul style="list-style-type: none"> - ヘッダなし。フラットファイルターゲットにヘッダ行を作成しません。 - 出力フィールド名。出力ポート名が指定されたファイルターゲットにヘッダ行を作成します。 - ヘッダコマンド出力を使用。ヘッダ行を生成するには、[ヘッダコマンド] フィールドでこのコマンドを使用します。たとえば、ファイルターゲットのヘッダ行にデータを追加するコマンドを使用できます。 <p>デフォルトは [ヘッダーなし] です。</p>
ヘッダーコマンド	ファイルターゲットにヘッダ行を生成するために使用するコマンド。
フッタのコマンド	ファイルターゲットにフッタ行を生成するために使用するコマンド。
出力タイプ	マッピングのターゲットのタイプ。ターゲットデータをフラットファイルに書き込むためのファイルを選択します。データをコマンドに出力するコマンドを選択します。

プロパティ	説明
コマンド	<p>ターゲットデータの処理に使用するコマンド。</p> <p>UNIX では、有効な任意の UNIX コマンドまたはシェルスクリプトを使用します。例えば、UNIX 上のターゲットデータから圧縮ファイルを生成するには、次のコマンドを使用します。</p> <pre>compress -c - > MyTargetFiles/MyCompressedFile.Z</pre> <p>Windows では、有効な任意の DOS コマンドまたはバッチファイルを使用します。フラットファイルライターは、フラットファイルターゲットではなくコマンドにデータを送信します。例えば、ファイルシステム内のステージングデータの回避やセキュリティ違反の回避には、Windows 上でターゲットコマンドとして cmd を使用します。</p> <p>データ統合サービスではなくコマンドにトランスフォーメーションタスクをプッシュすれば、マッピングのパフォーマンスを向上させることができます。コマンドを使用してターゲットデータのソートや圧縮を行うこともできます。</p>
マージコマンド	<p>すべてのターゲットパーティションのマージデータの処理に使用するマージコマンド。データ統合サービスは、マージデータを処理するためのコマンドに、コンカレントマージタイプを使用する必要があります。コマンドでは、ターゲットデータの順序が維持されない場合があります。</p>
出力ファイルディレクトリ	<p>フラットファイルターゲットの出力ディレクトリ。Informatica サービスをホストするマシンは、このディレクトリにアクセスする必要があります。</p> <p>複数のパーティションがフラットファイルターゲットに書き込む場合、パフォーマンスを向上させるために、セミコロンで区切られた複数のディレクトリを入力します。</p> <p>デフォルトは TargetDir システムパラメータです。</p>
出力ファイル名	<p>フラットファイルターゲットのファイル名。</p> <p>複数のパーティションがフラットファイルのターゲットに書き込む場合でターゲットデータをマージしないことにした場合は、各パーティションが <output_file_name><partition_number>.out という形式で命名される出力ファイルに別々に書き込みを行います。</p>
マージタイプ	<p>データ統合サービスがパーティション化されたターゲットのデータに対して実行するマージのタイプ。データ統合サービスがターゲットにパーティションを作成しない場合、この値は無視されます。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - マージなし。データ統合サービスは、ターゲット出力を各パーティションの個別のファイルに同時に書き込みます。 - 連続。データ統合サービスは各パーティションに対して出力ファイルを作成してから、マッピングの最後にそれらを 1 つのマージファイルにマージします。 - ファイルリスト。データ統合サービスは、各パーティションに対してターゲットファイルを作成し、個々のファイルのパスを含むファイルリストを作成します。 - 同時進行。データ統合サービスは、すべてのターゲットパーティションのデータをマージファイルに同時に書き込みます。データ統合サービスによってすべてのパーティションに対しマージファイルに同時に書き込まれるため、統合ファイル内のデータのソート順は連続していない可能性があります。
マージファイルディレクトリ	<p>すべてのターゲットパーティションに関するマージファイルのディレクトリ。Informatica サービスをホストするマシンは、このディレクトリにアクセスする必要があります。</p> <p>デフォルトは TargetDir システムパラメータです。</p>

プロパティ	説明
マージファイル名	すべてのターゲットパーティションに関するマージファイルの名前。デフォルトは出力ファイル名です。
接続タイプ	接続のタイプ。次のオプションから選択します。 <ul style="list-style-type: none"> - なし。ターゲットファイルは接続を必要としません。ターゲットファイルの場所は、出力ファイルディレクトリによって指定されます。 - Hadoop ファイルシステム。ターゲットファイルは HDFS 内にあります。デフォルトは [なし] です。

制御ファイル

データ統合サービスでは、制御ファイルに基づいてフラットファイルのカラムメタデータを更新できます。

制御ファイルは、フィールド名、データ型、精度、および位取りが含まれる単純なテキストファイルです。制御ファイルの各行には 1 つのカラムのデータが含まれます。

実行時に制御ファイルからカラム名を生成するには、フラットデータオブジェクトの [詳細] プロパティで **[ランタイムカラム名の生成]** プロパティの **[制御ファイルから]** を選択します。読み取りトランスフォーメーション設定して実行時にカラムメタデータを取得する必要もあります。

制御ファイルに次のデータ型を指定できます。

- Bigint
- 日付/時刻
- 10 進型
- 倍精度浮動小数点数型
- 整数型
- 文字列型
- Text
- TimestampwithTZ

制御ファイルを使用してフラットファイルデータオブジェクトに基づく読み取りトランスフォーメーションのランタイムカラム名の生成または、フラットファイルデータオブジェクトの作成ができます。データ統合サービスは制御ファイルに存在する情報を使用して、フラットファイルのカラムメタデータを構築します。カラム名、精度、データ型、およびスケールの各属性はカンマで区切られます。カラム間の情報は改行文字で区切られます。

実行時のカラムの更新

実行時のソースメタデータの変更に对应できるようにフラットファイルデータオブジェクトを設定できます。データ統合サービスは、カラムの数または順序がフラットファイルの物理データオブジェクトのカラムの数または順序と異なる場合に、フラットファイルからデータを読み取ることができます。

例えば、マッピングのソースが別の部門から提供されているとします。ソースの提供元の部門は、ソースファイルのカラムの順序を保証できません。場合により、その部門がファイルのカラムを変更していたり、追加のカラムを含めていたりすることがあります。実行時のソースメタデータの変更を受け付けるようにフラットファイルデータオブジェクトを設定できます。

ランタイムカラム名は、ソースの任意の行、フラットファイルヘッダー、または制御ファイルから自動的に生成できます。

入力ルールを使用すると、フラットファイルデータオブジェクトに基づいて読み取りトランスフォーメーションからマッピング内の次のトランスフォーメーションにランタイムカラムを射影できます。

〔データオブジェクト〕 タブの **〔実行時に、データソースからデータオブジェクトのカラムを取得します〕** を選択するように、読み取りトランスフォーメーションのデータオブジェクトプロパティを設定できます。このプロパティを設定すると、フラットファイルデータオブジェクトのオプションによって、データ統合サービスによるデータの処理方法が決まります。

ランタイムカラム名の自動生成

データ統合サービスでは、カラム位置に基づいて自動的にフラットファイルのカラムメタデータを更新できます。

データ統合サービスは、フラットファイルの形式プロパティにインポート開始のために指定する行に基づいて、フラットファイルのカラムメタデータを動的に更新します。カラム位置を変更せずに、デフォルトのデータ型、精度、およびスケールがランタイムカラムのカラムメタデータに使用されます。

ランタイムに自動でカラム名を生成するには、フラットファイルデータオブジェクトの **〔詳細〕** プロパティの **〔ランタイムカラム名の生成〕** プロパティで **〔自動 (Field1、Field2...)〕** を選択します。読み取りトランスフォーメーション設定して実行時にカラムメタデータを取得する必要もあります。

例えば、フラットファイルデータオブジェクトを設定し、実行時に変更を受け入れます。カラム位置に基づいて、フラットファイルのデータによってカラムメタデータが決まります。カラム Dept、Name、Place を使用して、フラットファイルデータオブジェクトを定義しました。実行時にデータ統合サービスでフラットファイルのカラムメタデータを動的に構築する必要があります。

入力ファイルには、以下の情報が含まれます。

```
HR,Bob,Chicago,US,87675
Finance,Mary,California,US,65437,t567,4200
```

〔形式〕 プロパティでフラットファイルデータオブジェクトを設定して、行 1 からデータのインポートが開始するようにしました。最初の行に 5 つのカラムがあるため、データ統合サービスは 5 つのカラムがあるフラットファイルデータオブジェクトを処理します。データ統合サービスは、位置に基づいて一意の名前 (Field4 とおよび Field5) で、フラットファイルデータオブジェクトに追加のランタイムカラムを作成します。2 番目の行で追加のカラムは処理されません。

マッピングにフラットファイルデータオブジェクトを追加する場合、読み取りトランスフォーメーションを設定して実行時にカラムメタデータを取得します。実行時にデータ統合サービスは、フラットファイルデータオブジェクトを構築し、次の値を使用してフラットファイルの行を処理します。

Dept	Name	Place	Field4	Field5
HR	Bob	Chicago	US	87675
Finance	Mary	California	US	65437

データファイルヘッダからのランタイムカラム名の生成

データ統合サービスでは、ヘッダー行を使用してカラムメタデータを決定できます。フラットファイルの最初の行がヘッダーです。

デフォルトのデータ型、精度、およびスケールがランタイムカラムのカラムメタデータに使用されますが、実行時にカラム位置は変更できます。

実行時にデータファイルヘッダーからカラム名を生成するには、フラットデータオブジェクトの [詳細] プロパティで **ランタイムカラム名の生成** プロパティの **データファイルヘッダーから (最初の行)** を選択します。読み取りトランスフォーメーション設定して実行時にカラムメタデータを取得する必要があります。

例えば、フラットファイルデータオブジェクトを設定し、実行時に変更を受け入れます。フラットファイルのデータで、データファイルヘッダーに基づいてカラムメタデータを決定します。

カラム Name、Age、Dept があるフラットファイルデータオブジェクトを定義したとします。フラットファイルデータオブジェクトの詳細プロパティを設定し、データファイルヘッダーから実行時にカラム名を生成できます。

データフラットファイルには、以下の情報が含まれます。

Dept	Name	Place	Country	Phone_No.
HR	Bob	Chicago	US	87675
Finance	Mary	California	US	65437 t567 4200

マッピングにフラットファイルデータオブジェクトを追加する場合、読み取りトランスフォーメーションを設定して実行時にカラムメタデータを取得します。ランタイム中、データ統合サービスは、Dept、Name、Place、Country および Phone_No のカラムがあるフラットファイルデータオブジェクトを生成します。Age カラムがフラットファイルヘッダー行に存在しないため、実行時に構成されるフラットファイルデータオブジェクトには Age カラムがありません。

データ統合サービスは Country や Phone Number などのランタイムカラムをマッピングルールに基づいてダウンストリームのトランスフォーメーションにプロパゲートします。

制御ファイルからのカラムメタデータの生成

制御ファイルは単純なテキストファイルであり、それを基にしてフラットファイルデータオブジェクトを作成できます。制御ファイルを使用して、実行時にフラットファイルデータオブジェクトに基づいて読み取りトランスフォーメーションのカラムを更新することもできます。

データ統合サービスが実行時にカラムを生成するときは、フラットファイルデータオブジェクトのプロパティに指定されている小数点記号および桁区切り記号を使用します。フラットファイルデータオブジェクトにカラムが存在する場合、制御ファイルで指定する必要があるのは、カラムの名前のみです。

フラットファイルソースの [詳細] ビューのランタイムプロパティでソースタイプを間接として指定した場合、実行時に制御ファイルを使用してカラム名を生成できます。

区切りフラットファイルまたは固定長フラットファイルの制御ファイルを作成できます。

フラットファイルデータオブジェクトのランタイムプロパティの制御ファイル名と制御ファイルディレクトリはパラメータ化できます。

データ統合サービスは、制御ファイルおよび入力データに指定したカラムメタデータに基づいて制御ファイルを処理します。

実行時に制御ファイルからカラム名を生成し、ソースデータオブジェクトにファイル名カラムが含まれている場合、データ統合サービスは検証エラーを重複するファイル名でログに記録します。

制御ファイルの形式

区切りフラットファイルと固定長フラットファイルの制御ファイル形式は異なります。制御ファイルを使用して、実行時にカラムメタデータを生成したり、フラットファイルデータオブジェクトを作成したりできます。

以下のいずれかの形式を使用して、制御ファイルを作成します。

区切り制御ファイル

区切り制御ファイルの各行の形式は以下のとおりです。

```
[column name],<data type>,<precision>,<scale>
```

区切りフラットファイルの制御ファイルの例を以下に示します。

```
Dept,String,10
Name
Place,String,20
Country
Phone number,string,30
```

固定長制御ファイル

固定長制御ファイルの各行の形式は以下のとおりです。

```
[column name],<data type>,<precision>,<scale>,<number of bytes to process>
```

区切りフラットファイルの制御ファイルの例を以下に示します。

```
Dept,String,10
Name
Place,String,20
Country
Phone number,string,30
```

固定長ソースの場合、制御ファイルには、処理バイト数用の追加カラムが含まれます。処理バイト数を指定しない場合、データ統合サービスは、精度に指定された値を処理バイト数として使用します。

ランタイムプロパティのパラメータ化

追加のカラムのデフォルトのスケール、精度、およびデータ型の値は、フラットファイルデータオブジェクトのランタイムプロパティの制御ファイルからパラメータ化または指定できます。フラットファイルデータオブジェクトのランタイムプロパティの制御ファイル名と制御ファイルディレクトリはパラメータ化できます。物理データオブジェクトプロパティの「詳細」タブでパラメータを設定します。物理データオブジェクトからトランスフォーメーションを作成するときに、マッピングパラメータを使用してパラメータのデフォルト値をオーバーライドできます。

制御ファイルのランタイム処理

マッピングを開発する際には、フラットファイルからデータを読み取る方法を決めるデータオブジェクト読み取りプロパティを定義します。データ統合サービスは、制御ファイルに基づくフラットファイルデータオブジェクトのカラムを処理できます。

データ統合サービスは、制御ファイルに基づいてカラムメタデータを構築する際に、以下のデータ処理基準を適用します。

フラットファイルデータオブジェクトに指定されていないカラムプロパティについては、デフォルト値を適用する

データ統合サービスは、制御ファイルにランタイムカラムプロパティが指定されていない場合、フラットファイルデータオブジェクトのデフォルトのランタイムプロパティを適用します。また、制御ファイルの追加カラムにデータ型、精度、スケールが指定されていない場合は、デフォルトのデータ型、精度、スケールを使用します。

制御ファイルには現れないがフラットファイルデータオブジェクトには存在するカラムについては、データを NULL として処理する

制御ファイルにフラットファイルデータオブジェクトのカラムが存在しない場合、データ統合サービスは、データプレビュー時にデータを NULL として処理します。これらのカラムは制御ファイルに存在しないため、データ統合サービスが実行時に処理することはできません。

データ統合サービスは、ソースメタデータが制御ファイルと一致しない場合、マッピングに失敗する

制御ファイルに基づくソースメタデータが入力データと一致しない場合、データ統合サービスはデータの処理に失敗します。例えば、ソースに制御ファイルのデータ型が含まれており、それがソース内のデータ型と互換性がない場合、データ統合サービスはマッピングに失敗します。

制御ファイルに関するルールおよびガイドライン

制御ファイルを使用する場合、以下のガイドラインに従います。

- 制御ファイルは、データ統合サービスが実行されているマシンにアクセスする必要があります。
- 制御ファイルをフラットファイルデータオブジェクトとしてインポート後、データオブジェクトのカラムのデータ型、精度、およびスケールは変更できません。制御ファイルから追加のカラムのカラム位置およびカラムメタデータは変更できません。
- 制御ファイルに基づいて、実行時にカラム名を生成する場合、ルックアップキャッシュに制御ファイルのメタデータが反映されます。マッピングを一度実行した後、追加のカラムがある同じ制御ファイルを更新する場合、ルックアップトランスフォーメーションの「ランタイム」プロパティから「ルックアップソースからの再キャッシュ」を選択する必要があります。
- 制御ファイルで日付形式のカラムを指定することはできません。データ統合サービスは、「詳細」ビューのフラットファイルデータオブジェクトの詳細プロパティで設定されているフラットファイルの日付形式を使用します。フラットファイルソースに複数の日付形式がある場合、データ統合サービスはすべての日付の「詳細」ビューに指定されている 1 つの日付形式のみを使用します。
- 制御ファイルでバイナリデータ型は指定できません。制御ファイルに TimestampwithTZ データ型が含まれる場合、そのデータ型の精度が 36、スケールが 9 に設定されていることを確認します。

フラットファイルデータオブジェクトの作成

フラットファイルデータオブジェクトは、空のデータオブジェクトとして、既存のフラットファイルまたは制御ファイルから作成できます。区切り形式または固定長のフラットファイルデータオブジェクトを任意の方法で作成できます。

フラットファイルデータオブジェクトを作成するには、以下の方法でデータオブジェクトを作成できます。

空のデータオブジェクトとして作成

Developer tool でデータオブジェクトのカラムと行を定義する場合は、空のフラットファイルデータオブジェクトを作成します。

既存のフラットファイルから作成

データオブジェクトに含めるメタデータを定義したフラットファイルが用意されている場合は、その既存のフラットファイルからフラットファイルデータオブジェクトを作成します。

制御ファイルから作成

制御ファイルに基づいてデータオブジェクトのカラムと行を定義する場合は、制御ファイルからフラットファイルデータオブジェクトを作成します。

空のフラットファイルデータオブジェクトの作成

Developer tool でデータオブジェクトのカラムと行を定義する場合は、空のフラットファイルデータオブジェクトを作成します。

1. **【オブジェクトエクスプローラ】** ビューで、プロジェクトまたはフォルダを選択します。
2. **【ファイル】** > **【新規】** > **【データオブジェクト】** をクリックします。
3. **【物理データオブジェクト】** > **【フラットファイルデータオブジェクト】** を選択し、**【次へ】** をクリックします。
【新しいフラットファイルデータオブジェクト】 ダイアログボックスが表示されます。
4. **【空として作成】** を選択します。
5. データオブジェクトの名前を入力します。
6. 必要に応じて、**【参照】** ボタンをクリックして、データオブジェクトのプロジェクトまたはフォルダを選択します。
7. **【次へ】** をクリックします。
8. 処理するデータのコードページに一致するコードページを選択します。
9. **【区切り】** または **【固定長】** を選択します。
10. **【固定長】** を選択した場合、**【完了】** をクリックします。**【区切り】** を選択し、**【次へ】** をクリックします。
11. 次の区切り形式のプロパティを設定します。

プロパティ	説明
区切り文字	データのカラムを区切るために使用される文字。エスケープ文字またはテキスト修飾子と同じ区切り文字を入力すると、予期しない結果が生成されることがあります。
引用符	引用符はテキスト文字列の境界を定義します。引用符を選択すると、引用符で囲まれた文字列内の区切り文字が無視されます。

12. **【完了】** をクリックします。
エディタに空のデータオブジェクトが開きます。 **【概要】** ビューでデータオブジェクトのカラムを定義します。

既存のフラットファイルからのフラットファイルデータオブジェクトの作成

データオブジェクトに含めるメタデータを定義したフラットファイルが用意されている場合は、その既存のフラットファイルからフラットファイルデータオブジェクトを作成します。

1. **【オブジェクトエクスプローラ】** ビューで、プロジェクトまたはフォルダを選択します。
2. **【ファイル】** > **【新規】** > **【データオブジェクト】** をクリックします。
3. **【物理データオブジェクト】** > **【フラットファイルデータオブジェクト】** を選択し、**【次へ】** をクリックします。
【新しいフラットファイルデータオブジェクト】 ダイアログボックスが表示されます。
4. **【既存のフラットファイルから作成】** を選択します。
5. **【参照】** をクリックし、ファイルを含むディレクトリに移動します。
6. **【開く】** をクリックします。

選択したファイルと同じ名前がデータオブジェクトに付けられます。

7. 必要に応じて、データオブジェクト名を編集します。
8. 必要に応じて、**【参照】** ボタンをクリックして、データオブジェクトのプロジェクトまたはフォルダーを選択します。
9. **【次へ】** をクリックします。
10. 処理するデータのコードページに一致するコードページを選択します。
11. **【区切り】** または **【固定長】** を選択します。
12. 必要に応じて、プレビューする行の最大数を編集します。
13. **【次へ】** をクリックします。
14. フラットファイルが区切りまたは固定長のどちらであるかに応じて、フォーマットプロパティを設定します。
 - 区切りフラットファイルの場合は、次のプロパティを設定します。

プロパティ	説明
区切り文字	データのカラムを区切るために使用される文字。エスケープ文字またはテキスト修飾子と同じ区切り文字を入力すると、予期しない結果が生成されることがあります。
引用符	引用符はテキスト文字列の境界を定義します。引用符を選択すると、引用符で囲まれた区切り文字が無視されます。
1 行目からカラム名をインポート	選択すると、Developer tool は最初の行のデータをカラム名として使用します。カラム名が最初の行に含まれている場合に、このオプションを選択します。有効でないフィールド名にはプレフィックス「FIELD_」が付きます。
行区切り文字	改行文字を指定します。リストから選択するか、文字を入力します。8 進コードの前にはバックスラッシュ (\) をつけます。1 文字を使用するには、その文字を入力します。 入力した文字の先頭にバックスラッシュが付いていない場合は、データ統合サービスは最初の文字だけを使用します。文字は 1 バイト文字とします。コードページ内のそれ以外の文字には、そのバイトが含まれてはいけません。デフォルトは、改行、\012 LF (\n) です。
エスケープ文字	引用符で囲まれていない文字列におけるカラム区切り文字の直前の文字、または引用符で囲まれた文字列内における引用符の直前の文字。エスケープ文字を指定すると、データ統合サービスは、区切り文字を普通の文字として読み込みます。
次の行からインポートを開始	データ統合サービスがファイルをインポートする際に、読み込みを開始する行番号。例えば 2 行目から開始するように指定すると、1 行目をスキップしてから読み込みを開始します。
連続した区切り文字を 1 文字として扱う	選択した場合、データ統合サービスは 1 つ以上の連続したカラム区切り文字を 1 つのカラム区切り文字として読み取ります。このオプションを選択しない場合、データ統合サービスは 2 つの連続した区切り文字を 1 つの NULL 値として読み取ります。
データ内のエスケープ文字を維持	出力文字列にエスケープ文字を含めます。

- 固定長フラットファイルの場合は、次のプロパティを設定します。

プロパティ	説明
1 行目からカラム名をインポート	選択すると、Developer tool は最初の行のデータをカラム名として使用します。カラム名が最初の行に含まれている場合に、このオプションを選択します。
次の行からインポートを開始	データ統合サービスがファイルをインポートする際に、読み込みを開始する行番号。例えば 2 行目から開始するように指定すると、1 行目をスキップしてから読み込みを開始します。
カラムの区切り	固定長ファイルのカラムの区切りを設定します。【区切りの編集】をクリックしてカラムの境界線を編集します。または、表示される指示に従い、ファイルのプレビューウィンドウでカラム間の境界線を操作します。カラムの境界線はドラッグして移動できます。境界線を削除するには、これをダブルクリックします。

15. 【次へ】をクリックして、フラットファイルデータオブジェクトをプレビューします。

16. 【完了】をクリックします。

データオブジェクトがエディタに表示されます。

制御ファイルからのフラットファイルデータオブジェクトの作成

制御ファイルに基づいてデータオブジェクトのカラムと行を定義する場合は、制御ファイルからフラットファイルデータオブジェクトを作成します。

1. 【オブジェクトエクスプローラ】ビューで、プロジェクトまたはフォルダを選択します。
2. 【ファイル】 > 【新規】 > 【データオブジェクト】をクリックします。
3. 【物理データオブジェクト】 > 【フラットファイルデータオブジェクト】を選択し、【次へ】をクリックします。
【新しいフラットファイルデータオブジェクト】ダイアログボックスが表示されます。
4. 【制御ファイルから作成】を選択します。
5. 【参照】をクリックし、制御ファイルが含まれるディレクトリに移動します。
6. 【開く】をクリックします。
選択した制御ファイルと同じ名前がデータオブジェクトに付けられます。
7. 必要に応じて、データオブジェクト名を編集します。
8. 必要に応じて、【参照】ボタンをクリックして、データオブジェクトのプロジェクトまたはフォルダを選択します。
9. 【次へ】をクリックします。
10. 処理するデータのコードページに一致するコードページを選択します。
11. 【区切り】または【固定長】を選択します。
12. 【固定長】を選択した場合、【完了】をクリックします。【区切り】を選択し、【次へ】をクリックします。

13. 次の区切り形式のプロパティを設定します。

プロパティ	説明
区切り文字	データのカラムを区切るために使用される文字。エスケープ文字またはテキスト修飾子と同じ区切り文字を入力すると、予期しない結果が生成されることがあります。
引用符	引用符はテキスト文字列の境界を定義します。引用符を選択すると、引用符で囲まれた文字列内の区切り文字が無視されます。

14. **【完了】** をクリックします。
データオブジェクトがエディタに表示されます。

第 7 章

データの論理ビュー

この章では、以下の項目について説明します。

- [データの論理ビューの概要, 109 ページ](#)
- [データの論理ビューの開発, 111 ページ](#)
- [論理データオブジェクトモデル, 111 ページ](#)
- [論理データオブジェクトモデルのプロパティ, 113 ページ](#)
- [論理データオブジェクト, 120 ページ](#)
- [論理データオブジェクトマッピング, 124 ページ](#)

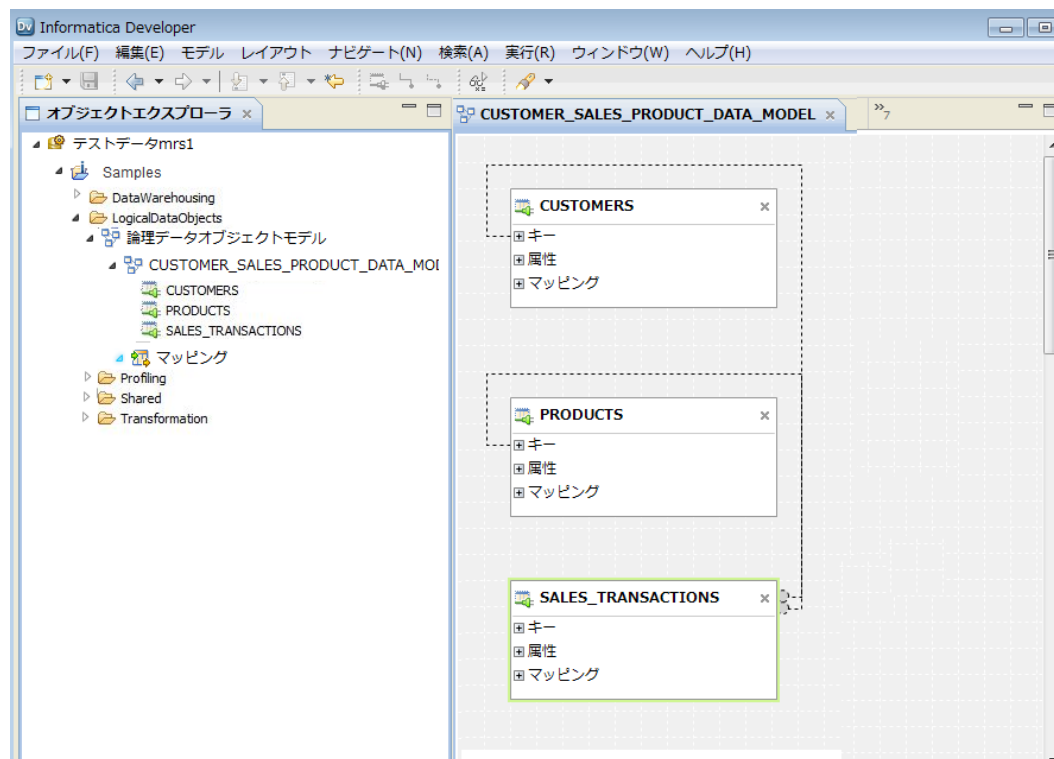
データの論理ビューの概要

データの論理ビューは、企業内に存在するデータを表すビューです。データの論理ビューには、論理データモデル、論理データオブジェクト、および論理データオブジェクトマッピングが含まれます。

データの論理ビューは、以下の目的で使します。

- 個々のビジネスニーズに対応しようとしてデータを定義しなおす手間を省くため、企業全体で共通のデータモデルを使用する。これは、データ属性に変更があったときに変更の適用が 1 回で済むことを意味しています。つまり、1 つのマッピングを使用して、このデータを使用するすべてのデータベースに変更を反映できるのです。
- 関連するデータのソースを見つけて、単一のビューに表示する。データは企業内のさまざまな場所（リレーショナルデータベースやフラットファイルなど）に存在しますが、1 つのビューからすべてのデータソースにアクセスして表示することができます。
- 論理データをリレーショナルテーブルとして公開し、再利用を促進する。

以下の図に、関連する論理データオブジェクト例を示します。



論理データオブジェクトモデルの例

論理データオブジェクトモデルを作成して、企業内の論理エンティティを示します。例えば、論理データオブジェクトモデルを作成して、異なるソースからのアカウントデータを単一のビューに表示します。

American Bank が California Bank を買収しました。買収後、American Bank は以下の操作が必要になりました。

- 両行のデータを、上位 10 位の顧客レポートなどのビジネスインテリジェンスレポートに表示する。
- 両行のデータを中央のデータウェアハウスに統合する。

従来、American Bank は、開発環境でデータを中央のデータウェアハウスに統合し、データを検証して、そのデータウェアハウスをプロダクション環境に移動していました。このプロセスには数か月以上を要していました。また、American Bank では、ビジネスインテリジェンスレポートはプロダクション環境のデータウェアハウスに対して実行していました。

American Bank の開発者は Developer ツールを使用して、顧客、口座、支店、およびその他の企業内データのモデルを作成できます。American Bank と California Bank のリレーショナルソースを、顧客の単一ビューにリンクします。その結果、中央のデータウェアハウスを作成する前に、ビジネスインテリジェンスレポートに使用可能なデータを作成できます。

データの論理ビューの開発

企業におけるデータへのアクセス方法とデータの使用方法を表す、データの論理ビューを開発します。

データの論理ビューを開発した後、データサービスに追加して、仮想データをエンドユーザーが使えるようにします。

データの論理ビューを開発する前に、論理データオブジェクトマッピングで使用する物理データオブジェクトを定義できます。データ品質を分析するために物理データソースをプロファイルすることもできます。

1. 論理データモデルを作成またはインポートします。
2. 必要に応じて、論理データオブジェクトを論理データオブジェクトモデルに追加して、オブジェクト間のリレーションを定義します。
3. 論理データオブジェクトに対してデータの読み取りまたは書き込みを行うための論理データオブジェクトマッピングを作成します。論理データオブジェクトマッピングには、データを変換するためのトランスフォーマーメソッドロジックを含めることができます。このトランスフォーメーションには、データを検証およびクレンジングするためのデータ品質トランスフォーメーションを含めることができます。
4. 論理データオブジェクトマッピングの出力を表示します。

論理データオブジェクトモデル

論理データオブジェクトモデルは、企業内のデータの構造および使用方法を表します。論理データオブジェクトモデルは論理データオブジェクトを含み、オブジェクト間のリレーションを定義します。

論理データオブジェクトモデルを定義して、企業内のデータの統合モデルを作成します。企業のデータは、リレーショナルデータベースやフラットファイルなど、複数の異なるソースシステム内に存在します。論理データオブジェクトモデルは、ソースシステムにかかわらず、ビジネスの観点からデータを表します。論理データオブジェクトモデルを作成して、データの検証、データ属性の記述、および属性間のリレーションの定義を行います。

例えば、American Bank の顧客口座データが Oracle データベースに存在し、California Bank のデータが IBM DB2 データベースに存在する場合に、顧客と口座のリレーションを定義する、顧客口座の統合モデルを作成するには、このリレーションを定義する論理データオブジェクトモデルを作成します。

モデリングツールから論理データオブジェクトモデルをインポートすることができます。モデリングツールで作成した XSD ファイルから論理データオブジェクトモデルをインポートすることもできます。または、Developer ツールで論理データオブジェクトモデルを手動で作成することもできます。

論理データオブジェクトモデルはプロジェクトまたはフォルダーに追加して、モデルリポジトリ内に格納します。

エンドユーザーが論理データオブジェクトに対して SQL クエリを実行できるようにするには、論理データオブジェクトを SQL データサービスに含めます。論理データオブジェクトを仮想テーブルのソースにします。エンドユーザーが Web 経由で論理データオブジェクトにアクセスできるようにするには、論理データオブジェクトを Web サービスに含めます。論理データオブジェクトを操作のソースにします。

論理データオブジェクトモデルの作成

論理データオブジェクトモデルを作成して、企業内のデータの構造および使用方法を定義します。論理データオブジェクトモデルを作成する際には、論理データオブジェクトを追加できます。物理データオブジェクトを

各論理データオブジェクトに関連付けます。Developer ツールで、モデルの各論理データオブジェクトに対する論理データオブジェクト読み取りマッピングを作成します。

1. **[Object Explorer]** ビューで、プロジェクトまたはフォルダを選択します。
2. **[ファイル] > [新規] > [論理データオブジェクトモデル]** をクリックします。
[新規] ダイアログボックスが表示されます。
3. **[論理データオブジェクトモデル]** を選択し、**[次へ]** をクリックします。
[新しい論理データオブジェクトモデル] ダイアログボックスが表示されます。
4. 論理データオブジェクトモデルの名前を入力します。
5. 論理データオブジェクトを作成するには、**[次へ]** をクリックします。空の論理データオブジェクトモデルを作成するには、**[完了]** をクリックします。
[次へ] をクリックすると、論理データオブジェクトをモデルに追加するように求められます。
6. 論理データオブジェクトを作成するには、**[新規]** ボタンをクリックします。
論理データオブジェクトがリストに追加されます。
7. **[名前]** カラムに名前を入力します。
8. 必要に応じて、**[データオブジェクト]** カラムの **[オープン]** ボタンをクリックして、物理データオブジェクトを論理データオブジェクトに関連付けます。
[データオブジェクトの選択] ダイアログボックスが表示されます。
9. 物理データオブジェクトを選択し、**[OK]** をクリックします。
10. 手順 6～9 を繰り返して、論理データオブジェクトを追加します。
11. **[完了]** をクリックします。
論理データオブジェクトモデルがエディタに表示されます。

モデリングツールからの論理データオブジェクトモデルのインポート

モデリングツールまたは XSD ファイルから論理データオブジェクトモデルをインポートすることができます。企業内の既存の構造のモデルおよびデータを利用するには、論理データオブジェクトモデルをインポートします。

1. 論理データオブジェクトモデルのインポート先のプロジェクトまたはフォルダを選択します。
2. **[ファイル] > [新規] > [論理データオブジェクトモデル]** をクリックします。
[新しい論理データオブジェクトモデル] ダイアログボックスが表示されます。
3. **[データモデルから論理データオブジェクトモデル]** を選択します。
4. **[次へ]** をクリックします。
5. **[モデルタイプ]** フィールドで、論理データオブジェクトモデルのインポート元のモデリングツールを選択します。
6. 論理データオブジェクトモデルの名前を入力します。
7. **[参照]** をクリックして、論理データオブジェクトモデルの場所を選択します。
8. **[次へ]** をクリックします。
9. インポートするファイルを参照し、選択して **[オープン]** をクリックします。
10. インポートプロパティを設定します。
11. **[次へ]** をクリックします。

12. 論理データオブジェクトを論理データオブジェクトモデルに追加します。
13. **【完了】** をクリックします。
論理データオブジェクトがエディタに表示されます。

論理データオブジェクトモデルのプロパティ

論理データオブジェクトモデルをモデリングツールからインポートする場合、このツールに関連付けられたプロパティを指定します。

CA ERwin Data Modeler のインポートプロパティ

論理データオブジェクトモデルを CA ERwin Data Modeler からインポートする場合、インポートプロパティを設定します。

以下の表に、モデルを CA ERwin Data Modeler からインポートする場合に設定するプロパティを示します。

プロパティ	説明
UDP のインポート	ユーザー定義プロパティのインポート方法を指定します。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- メタデータとして: 明示的な値をプロパティ値オブジェクトとしてインポートします。明示的な値はエクスポートされません。- メタデータとして既定値を移行: 明示的な値および黙示的な値をプロパティ値オブジェクトとしてインポートします。- 説明に既定値を移行: 黙示的な場合にも、プロパティ名と値をオブジェクトの説明プロパティに追加します。- 両方で既定値を移行: 黙示的な場合にも、UDP 値をメタデータとして、またオブジェクトの説明にインポートします。 デフォルトは [メタデータとして] です。
リレーション名のインポート	ERwin からのリレーション名のインポート方法を指定します。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- リレーション名から- リレーション説明から デフォルトは [リレーション名から] です。
ID のインポート	オブジェクトの一意の ID をネイティブ ID プロパティとして設定するかどうかを指定します。
サブジェクト領域のインポート	ERwin からのサブジェクト領域のインポート方法を指定します。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- 図としてインポート- パッケージおよび図としてインポート- パッケージおよび図としてインポート、各エンティティに対して 1 つのサブジェクト領域を想定しています- サブジェクト領域をインポートしない デフォルトは [図としてインポート] です。

プロパティ	説明
カラム順序のインポート形式	<p>テーブル内のカラムの位置のインポート方法を指定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - カラム順序: ERwin 物理ビューに表示されるカラムの順序。 - 物理的順序: SQL DDL で生成されたときの、データベース内のカラムの順序。 <p>デフォルトは [物理的順序] です。</p>
所有者スキーマのインポート	所有者スキーマをインポートするかどうかを指定します。

IBM Cognos Business Intelligence Reporting - Framework Manager のインポートプロパティ

IBM Cognos Business Intelligence Reporting - Framework Manager から論理データオブジェクトモデルをインポートする場合、インポートプロパティを設定します。

以下の表に、モデルを IBM Cognos Business Intelligence Reporting - Framework Manager からインポートする場合に設定するプロパティを示します。

プロパティ	説明
フォルダの表示	<p>Framework Manager のフォルダの表示方法を指定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - 無視: フォルダを無視します。 - フラット: フォルダを図として表しますが、階層は維持しません。 - 階層: フォルダを図として表し、階層も維持します。 <p>デフォルトは [無視] です。</p>
パッケージの表示	<p>Cognos Framework Manager のパッケージの表示方法を指定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - 無視: サブジェクト領域を無視します。 - サブジェクト領域: パッケージをサブジェクト領域として表します。 - モデル: パッケージをモデルとして表します。 <p>デフォルトは [無視] です。</p>
リバースエンジニアリングのリレーション	2 つの dbQueries 間のリレーションを参照整合性制約として計算するかどうかを指定します。
テーブルデザインレベル	<p>インポートしたテーブルのデザインレベルの制御方法を指定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - 論理および物理: テーブルは、モデルの論理ビューと物理ビューの両方で表示されます。 - 物理: テーブルはモデルの物理ビューで表示されます。 <p>デフォルトは [物理] です。</p>
使用方法プロパティを無視	queryItem の使用方法プロパティを使用するかどうかを指定します。

SAP BusinessObjects Designer のインポートプロパティ

論理データオブジェクトモデルを SAP BusinessObjects Designer からインポートする場合、インポートプロパティを設定します。

以下の表に、モデルを SAP BusinessObjects Designer からインポートする場合に設定するプロパティを示します。

プロパティ	説明
System	BusinessObjects リポジトリの名前。 BusinessObjects versions 11.x および 12.x (XI)の場合、Central Management Server の名前を入力します。 BusinessObjects version 5.x および 6.x の場合、Supervisor アプリケーションによって定義されたリポジトリの名前を入力します。
認証モード	ログイン認証モード。 このパラメータは、SAP BusinessObjects Designer 11.0 以降に適用されます。 次のいずれかの認証モードを選択します。 <ul style="list-style-type: none">- エンタープライズ: Business Objects Enterprise ログイン- LDAP: LDAP サーバー認証- Windows AD: Windows Active Directory サーバー認証- Windows NT: Windows NT ドメインサーバー認証- スタンドアロン: スタンドアロン認証 デフォルトは [エンタープライズ] です。
ユーザー名	BusinessObjects サーバーのユーザー名。 バージョン 11.x および 12.x (XI)の場合、BusinessObjects グループのメンバーであることが必要です。
パスワード	BusinessObjects サーバーのパスワード。
サイレント実行	対話式モードで実行するか、サイレントモードで実行するかを指定します。 デフォルトは [サイレント] です。
実行後に閉じる	Developer Tool がモデルのインポートを完了した後に BusinessObjects を閉じるかどうかを指定します。
テーブルデザインレベル	インポートしたテーブルのデザインレベルを指定します。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- 論理および物理: テーブルは、モデルの論理ビューと物理ビューの両方で表示されます。- 物理: テーブルはモデルの物理ビューで表示されます。 デフォルトは [物理] です。
結合を外部キーに変換	モデル内の単純な SQL 結合を外部キーリレーションに変換します。 データベースデザインツールなど、構造化リレーショナルメタデータのみをサポートするツールにモデルをエクスポートする場合、このパラメータを選択します。

プロパティ	説明
クラス表示	<p>クラスおよびサブクラスのツリー構造をインポートする方法を指定します。Developer ツールは、CWM OLAP 標準の定義どおりに各クラスをディメンションとしてインポートします。また、Developer ツールは、クラスおよびサブクラスを CWM 標準および UML 標準の定義どおりにパッケージのツリーとしてインポートします。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - フラット構造: Developer ツールはパッケージを作成しません。 - 単純ツリー構造: Developer ツールは、サブクラスを持つ各クラスのパッケージを作成します。 - 完全ツリー構造: Developer ツールは、各クラスのパッケージを作成します。 <p>デフォルトは [フラット構造] です。</p>
値のリストを含める	<p>オブジェクトに関連付けられている値のリストをインポートする方法を制御します。</p>
次元プロパティのトランスフォーメーション	<p>次元名、説明、およびロールを、基本テーブルに変換する方法、および属性名、説明、データ型を基本カラムに変換する方法を指定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - 無効: プロパティ変換は発生しません。 - 有効: 次元オブジェクトとリレーションオブジェクト間に直接一致がある場合にプロパティ変換が発生します。Developer ツールは、次元名をリレーション名に移行します。 - 有効 (名前を保持): 次元オブジェクトとリレーションオブジェクト間に直接一致がある場合にプロパティ変換が発生します。Developer ツールはリレーション名を保持します。 <p>デフォルトは [無効] です。</p>

SAP PowerDesigner CDM のインポートプロパティ

論理データオブジェクトモデルを SAP PowerDesigner CDM からインポートする場合、インポートプロパティを設定します。

次の表に、SAP PowerDesigner CDM からモデルをインポートする場合に設定するプロパティを示します。

プロパティ	説明
UDP のインポート	<p>ユーザー定義プロパティのインポート方法を指定します。</p> <p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - メタデータとして: 明示的な値をプロパティ値オブジェクトとしてインポートします。明示的な値はエクスポートされません。 - メタデータとして既定値を移行: 明示的な値および黙示的な値をプロパティ値オブジェクトとしてインポートします。 - 説明に既定値を移行: 黙示的な場合にも、プロパティ名と値をオブジェクトの説明プロパティに追加します。 - 両方で既定値を移行: 黙示的な場合にも、UDP 値をメタデータとして、またオブジェクトの説明にインポートします。 <p>デフォルトは [メタデータとして] です。</p>
関連クラスのインポート	<p>関連クラスをインポートするかどうかを指定します。</p>
ID のインポート	<p>オブジェクトの一意の ID をネイティブ ID プロパティとして設定するかどうかを指定します。</p>

プロパティ	説明
説明フィールドに容量情報を追加する	オカレンス数の情報をインポートして説明プロパティに追加します。
テキストの書式の削除	リッチテキスト書式を削除するか保持するかを指定します。 モデルが PowerDesigner 7.0 または 7.5 で生成された場合は、このオプションを選択します。 モデルが PowerDesigner 8.0 以上で生成された場合は、このオプションをクリアします。

SAP PowerDesigner OOM (9.x - 15.x) のインポートプロパティ

論理データオブジェクトモデルを SAP PowerDesigner OOM (9.x - 15.x) からインポートする場合、インポートプロパティを設定します。

論理データオブジェクトモデルを SAP PowerDesigner OOM からインポートする場合、Developer tool はクラスと属性をインポートしますが、他のエンティティはインポートしません。論理データオブジェクトモデルをインポートするには、SAP PowerDesigner から UML 1.3 - XMI 1.0 XML 形式でモデルをエクスポートします。

次の表に、SAP PowerDesigner OOM からモデルをインポートする場合に設定するプロパティを示します。

プロパティ	説明
ターゲットツール	<p>インポートするモデルを生成したツールを指定します。 次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - 自動検出: ファイルを生成したツールを Developer ツールが検出します。 - OMG XMI: ファイルは OMG XMI 1.0 標準 DTD に準拠しています。 - Argo/UML 0.7: ファイルは 0.7.0 以前のバージョン Argo/UML によって生成されました。 - Argo/UML 0.8: ファイルは 0.7.1 以降のバージョンの Argo/UML によって生成されました。 - XMI Toolkit: ファイルは IBM XMI Toolkit によって生成されました。 - XMI Interchange: ファイルは Unisys Rose XMI Interchange によって生成されました。 - Rose UML: ファイルは Unisys Rose UML によって生成されました。 - Visio UML: ファイルは、UML to XMI Export を使用して、Microsoft Visio Professional 2002 および Visio for Enterprise Architects によって生成されました。 - PowerDesigner UML: ファイルは、XMI Export を使用して、Sybase PowerDesigner によって生成されました。 - Component Modeler: ファイルは、XMI Export を使用して、CA AllFusion Component Modeler によって生成されました。 - Netbeans XMI Writer: ファイルは、Poseidon などの Netbeans XMI Writer を使用するアプリケーションによって生成されました。 - Embarcadero Describe: ファイルは Embarcadero Describe によって生成されました。 <p>デフォルトは [自動検出] です。</p>
自動修正	不完全または不正確なモデルを修正して XML ファイルにインポートします。
モデルフィルタ	XML ファイルに複数のモデルが含まれている場合にインポートするモデル。複数のモデルはカンマで区切って指定します。

プロパティ	説明
最上位パッケージ	モデルの最上位パッケージ。
UUID のインポート	UUID をネイティブ ID としてインポートします。

SAP PowerDesigner PDM のインポートプロパティ

SAP PowerDesigner PDM から論理データオブジェクトモデルをインポートする場合、インポートプロパティを設定します。

次の表に、SAP PowerDesigner PDM からモデルをインポートする場合に設定するプロパティを示します。

プロパティ	説明
UDP のインポート	<p>ユーザー定義プロパティのインポート方法を指定します。 次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - メタデータとして: 明示的な値をプロパティ値オブジェクトとしてインポートします。明示的な値はエクスポートされません。 - メタデータとして既定値を移行: 明示的な値および黙示的な値をプロパティ値オブジェクトとしてインポートします。 - 説明に既定値を移行: 黙示的な場合にも、プロパティ名と値をオブジェクトの説明プロパティに追加します。 - 両方で既定値を移行: 黙示的な場合にも、UDP 値をメタデータとして、またオブジェクトの説明にインポートします。 <p>デフォルトは [メタデータとして] です。</p>
ID のインポート	オブジェクトの一意の ID をネイティブ ID プロパティとして設定するかどうかを指定します。
説明フィールドに容量情報を追加する	オカレンス数の情報をインポートして説明プロパティに追加します。
テキストの書式の削除	<p>リッチテキスト書式を削除するか保持するかを指定します。 モデルが PowerDesigner 7.0 または 7.5 で生成された場合は、このオプションを選択します。 モデルが PowerDesigner 8.0 以上で生成された場合は、このオプションをクリアします。</p>

XSD のインポートプロパティ

モデリングツールでエクスポートした XSD ファイルから論理データオブジェクトモデルをインポートすることができます。

以下の表に、モデルを XSD ファイルからインポートする場合に設定するプロパティを示します。

プロパティ	説明
要素のコンテンツ名	XSD ファイル内の#PCDATA などのテキストコンテンツを保持するための属性。 デフォルトは [メタデータとして] です。
縮小レベル	クラスを縮小する状況を指定します。選択した値によって、XSD ファイルの要素と属性のすべてがインポートされるか、または一部がインポートされるかが決まります。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- なし: すべての XSD 要素はクラスになり、すべての XSD 属性は属性になります。- 空: 空のクラスのみが、親クラスに縮小されます。- 単一属性: 属性が 1 つで、子がない XSD 要素のみが親クラスに縮小されます。- 子なし: 子がない XSD 要素のみが親クラスに縮小されます。- すべて: すべての縮小可能な XSD 要素が親クラスに縮小されます。 デフォルトは [すべて] です。
星印を縮小	入力 xlink を持つ XML 要素を親クラスに縮小するかどうかを指定します。
クラスタイプ	要素を親要素に縮小する場合にクラスタイプを作成するかどうかを指定します。
Any	「xs:any」疑似要素に対するクラスまたはエンティティを作成するかどうかを指定します。
ID の生成	プライマリキーと外部キーを作成するための追加の属性を生成するかどうかを指定します。デフォルトでは、追加の属性は生成されません。
置換グループのインポート方法	継承の表示方法を指定します。 次のいずれかのオプションを選択します。 <ul style="list-style-type: none">- 汎化: 継承を汎化として表します。- ロールダウン: サブクラスの継承された属性を複製します。 デフォルトは [ロールダウン] です。
インクルードパス	インクルードされたスキーマファイルを含むディレクトリのパス（該当する場合）。
UDP 名前空間	ユーザー定義プロパティとしてインポートされる属性を含む名前空間。

論理データオブジェクト

論理データオブジェクトは論理データオブジェクトモデル内のオブジェクトであり、企業の論理エンティティを表します。論理データオブジェクトには属性とキーがあり、属性間のリレーションを表します。

相互に関係する論理データオブジェクトをデータオブジェクトモデルに含めます。例えば、国立銀行の論理データオブジェクトモデルに、論理データオブジェクト「Customer（顧客）」と「Account（口座）」があるとします。論理データオブジェクトモデルは、顧客と口座のリレーションを表します。

このモデルでは、論理データオブジェクト Account は、属性 Account_Number（口座番号）を含みます。口座を一意に識別する Account_Number がプライマリキーです。Customer データオブジェクトは各顧客の口座を参照する必要があるため、Account には論理データオブジェクト Customer とのリレーションがあります。

物理データオブジェクトを論理データオブジェクトモデルエディタにドラッグして、論理データオブジェクトを作成できます。または、論理データオブジェクトを作成して、属性とキーを定義します。

論理データオブジェクトのプロパティ

論理データオブジェクトには、データオブジェクトを定義するプロパティと、論理データオブジェクトモデル内の他の論理データオブジェクトへのリレーションが含まれます。

論理データオブジェクトのプロパティは、エディタ内のエディタタブにまとめられています。以下の図に、論理データオブジェクトエディタを示します。

Overview

General

Name:

customer

Description:

Read Mapping:

customer_Read_Mapping

Add...

Delete

Write Mapping:

Add...

Delete

customer

Name	Type
CUSTID	decimal
Status	string
LastName	string
FirstName	string
CUSTID1	decimal

Attributes

	Name	Type	Primary...	Precision	Scale	Nullable	Lower	Upper	Queryable	Description	>>
1	CUSTID	decimal	✓	2	0		1	1			
2	Status	string		8	0	✓	1	1			
3	LastName	string		15	0	✓	1	1			
4	FirstName	string		9	0	✓	1	1			
5	CUSTID1	decimal	✓	2	0		1	1			

Overview

Keys

Relationships

Access

Read Mapping

Advanced

120 第7章: データの論理ビュー

次の表では、論理データオブジェクトエディタに表示される情報について説明します。

タブ名	説明
概要	[全般] 領域には、論理データオブジェクトの名前、説明、読み取りおよび書き込みマッピング（該当する場合）が含まれます。 [属性] 領域には、論理データオブジェクト内のデータの構造が表示されます。
キー	論理データオブジェクト内の 1 つ以上の属性をプライマリーキーまたは一意キーに指定できます。
リレーション	論理データオブジェクト間の関連付け。
アクセス	論理データオブジェクトとデータオブジェクトの各属性に対するアクセスの種類。
読み取りマッピング	論理データオブジェクトに関連付けられている論理データオブジェクト読み取りマッピング。論理データオブジェクトに読み取りマッピングが含まれていない場合は、[読み取りマッピング] タブは表示されません。
書き込みマッピング	論理データオブジェクトに関連付けられている論理データオブジェクト書き込みマッピング。論理データオブジェクトに書き込みマッピングが含まれていない場合は、[書き込みマッピング] タブは表示されません。

属性のリレーション

リレーションは、1 つ以上の論理データオブジェクトのプライマリー属性または外部キー属性間の関連付けです。

属性間には以下のタイプのリレーションを定義できます。

識別

ある属性が他の属性との関連付けによって識別される、2 つの属性間のリレーション。

例えば、論理データオブジェクト Branch（支店）の Branch_ID（支店 ID）属性と、論理データオブジェクト Customer（顧客）の Branch_Location（支店の場所）属性間のリレーションは「識別」です。支店 ID は、支店の場所に対して一意だからです。

非識別

一方の属性を他方の属性に依存せずに識別する、2 つの属性間のリレーション。

例えば、論理データオブジェクト Account（口座）の Account_Type（口座種別）属性と、論理データオブジェクト Customer（顧客）の Account_Number（口座番号）属性間のリレーションは「非識別」です。口座種別は、口座番号と関連付けなくても識別可能です。

リレーションを定義する際、論理データオブジェクトモデルでは、識別リレーションが実線で表されます。非識別リレーションは属性間の点線で表されます。

論理データオブジェクトの作成

論理データオブジェクトを論理データオブジェクトモデル内に作成し、企業の論理エンティティを定義できます。

1. [ファイル] > [新規] > [論理データオブジェクト] をクリックします。
2. 論理データオブジェクト名を入力します。

3. 論理データオブジェクトの論理データオブジェクトモデルを選択し、**【完了】** をクリックします。
論理データオブジェクトが論理データオブジェクトモデルエディタに表示されます。
4. 論理データオブジェクトを選択し、**【プロパティ】** ビューをクリックします。
5. [全般] 領域の **【概要】** タブで、必要に応じて論理データオブジェクトの名前と説明を編集できます。
6. [属性] 領域の **【概要】** タブで、属性を作成し、そのデータ型と精度を指定できます。

Overview

General
Name:
Description:
Read Mapping:
Write Mapping:

customer

Name	Type
CUSTID	decimal
Status	string
LastName	string
FirstName	string

Attributes

	Name	Type	Primary...	Precision	Scale	Nullable	Lower	Upper	Querya...	Description	>>
1	CUSTID	decimal	<input type="checkbox"/>	2	0	<input checked="" type="checkbox"/>	1	1			
2	Status	string	<input type="checkbox"/>	8	0	<input checked="" type="checkbox"/>	1	1			
3	LastName	string	<input type="checkbox"/>	15	0	<input checked="" type="checkbox"/>	1	1			
4	FirstName	string	<input type="checkbox"/>	9	0	<input checked="" type="checkbox"/>	1	1			

Overview | Keys | Relationships | Access | Read Mapping | Advanced

7. **【キー】** タブで、必要に応じてデータオブジェクトのプライマリキーと一意キーを指定します。

Keys

Keys:

Key

Add...

Remove

Name:
Description:

Available Attributes:

Status
LastName
FirstName

>
<

Selected Attributes:

CUSTID

↑
↓

Overview | Keys | Relationships | Access | Read Mapping | Advanced

8. **【リレーション】** タブで、必要に応じて論理データオブジェクト間のリレーションを作成します。

Relationships

Relationships: Relationship

Add... Remove

Name: Relationship

Description:

Relationship Type

☒ Identifying ☐ Non-Identifying

Referenced Key

Key: customer.Key Browse...

Attributes:

CUSTID

Logical Data Object

Available Attributes:

CUSTID
Status
LastName
FirstName

Selected Attributes:

CUSTID1

Overview | Keys | Relationships | Access | Read Mapping | Advanced

9. **【アクセス】** タブで、必要に応じて論理データオブジェクトとデータオブジェクトの各属性のアクセスの種類を編集します。

デフォルトは「読み取り専用」です。

Access

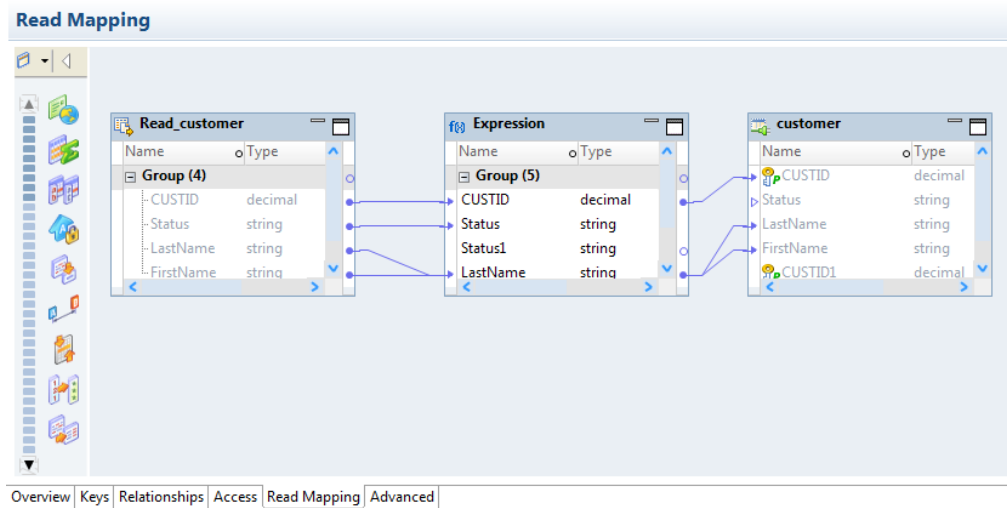
Write Access: Read Only

Attributes:

	Name	Type	Write Access
1	CUSTID	decimal	Read Only
2	Status	string	Read Only
3	LastName	string	Read Only
4	FirstName	string	Read Only
5	CUSTID1	decimal	Read Only

Overview | Keys | Relationships | Access | Read Mapping | Advanced

10. 【読み取りマッピング】タブで、必要に応じて、論理データオブジェクト読み取りマッピングを作成します。



11. 【書き込みマッピング】タブで、必要に応じて、論理データオブジェクト書き込みマッピングを作成します。
12. 論理データオブジェクトを保存します。

論理データオブジェクトマッピング

論理データオブジェクトマッピングは、論理データオブジェクトを1つ以上の物理データオブジェクトにリンクするマッピングです。論理データオブジェクトマッピングには、トランスフォーメーションロジックを含めることができます。

論理データオブジェクトマッピングのタイプは以下のいずれかです。

- 読み取り
- 書き込み

各論理データオブジェクトを、1つの論理データオブジェクト読み取りマッピングまたは1つの論理データオブジェクト書き込みマッピングに関連付けることができます。

論理データオブジェクト読み取りマッピング

論理データオブジェクト読み取りマッピングには、入力として1つ以上の物理データオブジェクトが、また出力として1つの論理データオブジェクトが含まれます。マッピングには、データを変換するためのトランスフォーメーションロジックを含めることができます。

論理データオブジェクト読み取りマッピングは、基のデータソースにアクセスせずに、データにアクセスする方法を提供します。また、複数のソースのデータを1つのビューにまとめて表示することを可能にします。

例えば、American Bankに顧客の口座の論理データオブジェクトモデルがあり、この論理データオブジェクトモデルにはCustomers（顧客）論理データオブジェクトが含まれているとします。

Customers 論理データオブジェクトにある2つのリレーショナルデータベースから顧客データを表示する場合、論理データオブジェクト読み取りマッピングを使用してこのタスクを実行し、出力を【データビュー】ビューで表示することができます。

論理データオブジェクト書き込みマッピング

論理データオブジェクト書き込みマッピングは、入力として論理データオブジェクトを含みます。このマッピングにより、論理データオブジェクトからターゲットへの書き込みが可能になります。

マッピングには、データを変換するためのトランスフォーメーションロジックを含めることができます。このマッピングは基になるデータターゲットにアクセスすることなく実行されます。ターゲットへ書き込むことなく、変換されたデータの単一ビューを提供します。

論理データオブジェクトマッピングの作成

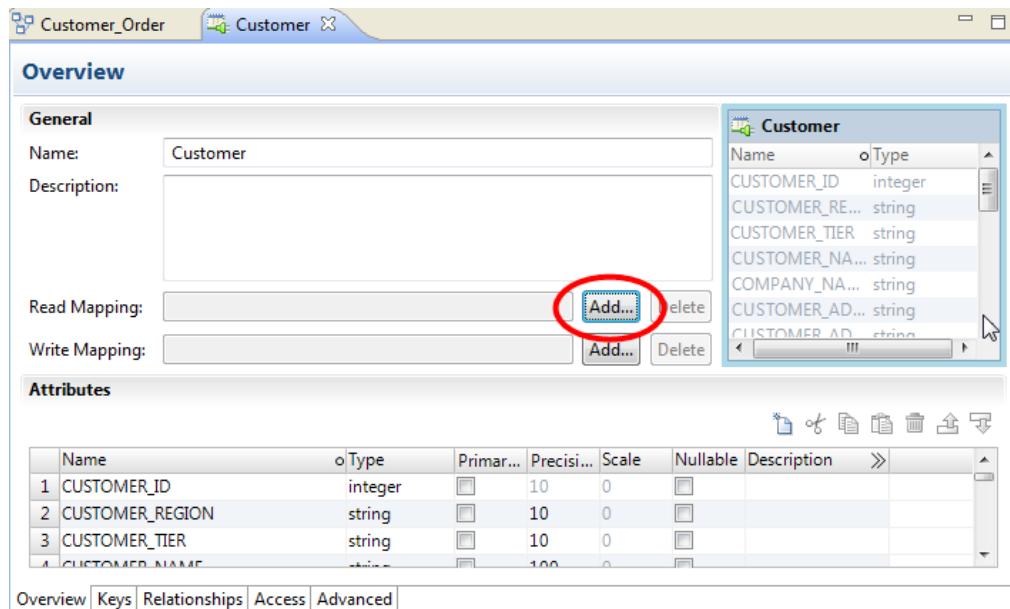
論理データオブジェクトマッピングを作成して、物理データオブジェクトから論理データオブジェクトにデータをリンクし、データを変換することができます。

1. **【オブジェクトエクスプローラ】** ビューで、マッピングを追加する論理データオブジェクトをダブルクリックします。

論理データオブジェクトエディタが開きます。

2. **【全般】** 領域の **【概要】** タブで、**【追加】** をクリックして読み取りマッピングまたは書き込みマッピングを追加します。

次の図に、**【追加】** ボタンを示します。



3. マッピングの名前を入力して **【完了】** をクリックします。
マッピングが読み取りマッピングか書き込みマッピングかに応じて、論理データオブジェクトがマッピング入力または出力としてエディタに表示されます。

4. 物理データオブジェクトから論理データオブジェクトにデータをリンクします。

- a. **【読み取りマッピング】** タブまたは **【書き込みマッピング】** タブをクリックしてマッピングを編集します。

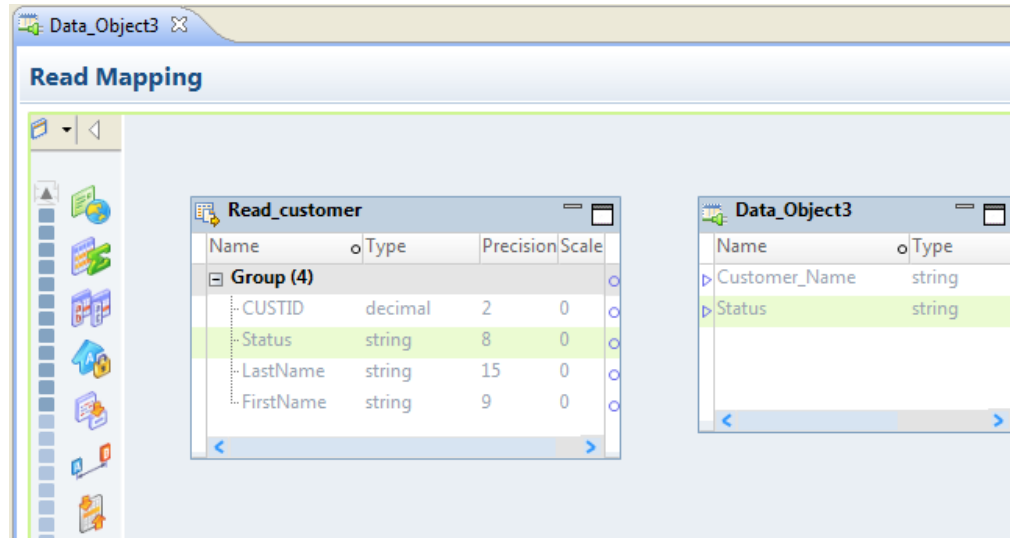
マッピングエディタパレットが開きます。

- b. **【オブジェクトエクスプローラ】** ビューで、リンクする物理データオブジェクトを参照し、物理データオブジェクトをマッピングエディタパレットにドラッグします。

【マッピングに追加】 ダイアログボックスが開きます。

- c. 読み取りマッピング、書き込みマッピング、またはルックアップマッピングの作成を選択します。
- d. 読み取りマッピングの作成を選択する場合は、読み取りマッピングのオブジェクトアクセスを関連データオブジェクトまたは独立したデータオブジェクトとして指定します。
- e. **【OK】** をクリックします。

以下の画像は、**【顧客】** フラットファイルデータオブジェクトをマッピングエディタパレットにドラッグし、読み取りマッピングとして指定した後の状態です。



5. 必要に応じて、マッピングに再利用可能なオブジェクトを追加します。
論理データオブジェクトまたは他のモデルリポジトリオブジェクトを追加できます。
 - a. マッピングエディタを右クリックして **【再利用可能なオブジェクトの追加】** を選択します。
 - b. モデルリポジトリからオブジェクトを 1 つ選択して **【OK】** をクリックします。
 - c. 再利用可能なオブジェクトを読み取りマッピング、書き込みマッピング、またはルックアップマッピングとして指定します。
 - d. 再利用可能なオブジェクトを読み取りマッピングとして指定する場合は、読み取りマッピングのオブジェクトアクセスを関連データオブジェクトまたは独立したデータオブジェクトとして指定します。
 - e. **【OK】** をクリックします。
6. 必要に応じて、追加オブジェクトとトランスフォーメーションをマッピングに追加し、マッピングオブジェクト間にリンクを作成してから、**【データビューア】** ビューをクリックしてマッピングを実行します。
結果が **【出力】** セクションに表示されます。

第 8 章

データの表示

この章では、以下の項目について説明します。

- [データの表示の概要, 127 ページ](#)
- [構成, 128 ページ](#)
- [データのエクスポート, 135 ページ](#)
- [オブジェクトの依存関係, 135 ページ](#)
- [ログ, 137 ページ](#)
- [検証設定, 138 ページ](#)
- [Developer tool からのジョブの監視, 139 ページ](#)

データの表示の概要

ユーザーは、マッピングの実行、プロファイル結果の表示、ソースデータの表示、トランスフォーメーションのデータのプレビュー、SQL クエリの実行、Web サービスのメッセージのプレビュー、またはオブジェクトに関する依存関係の表示を行うことができます。

マッピングを実行し、出力をソースからターゲットに移動してデータを変換します。コマンドラインまたは
【実行】ダイアログボックスからマッピングを実行できます。エディタにプロファイル結果を表示します。

【データビューア】ビューでソースデータの表示、トランスフォーメーションのデータのプレビュー、SQL クエリの実行、または Web サービスメッセージのプレビューを行います。

注: データビューアでプレビューできる最大行数は 100,000 行です。

データを表示する前に、デフォルトのデータ統合サービスを選択する必要があります。他のデータ統合サービスを追加して、データを表示するときに使用することもできます。設定を作成して、データを表示するときに Developer ツールで適用される設定を制御できます。

データを【データビューア】ビューで表示する場合、データをファイルにエクスポートできます。ログイベントを表示するログにアクセスすることもできます。

また、モデルリポジトリのオブジェクトを表示、変更、削除するときに、オブジェクトの依存性の表示することもできます。【オブジェクトの依存関係】ビューで、オブジェクトの依存関係を表示することができます。

構成

構成とは、マッピングの実行、データのプレビュー、SQL クエリの実行、または Web サービスメッセージのプレビュー時に、Developer ツールにより適用されるひとまとまりの設定です。

構成は、デフォルトの Data Integration Service、ソースから読み取る行数、デフォルトの日付/時刻フォーマット、および最適化レベルなどの設定を制御します。作成した構成は、Developer ツールのインストールに適用されます。

以下の構成を作成できます。

- データビューアの構成。出力を **【データビューア】** ビューでプレビューする場合に Developer ツールが適用する設定を制御します。
- マッピング構成。マッピングを **【実行設定】** ダイアログボックスから、またはコマンドラインから実行する場合に Developer ツールが適用する設定を制御します。
- Web サービス設定。**【データビューア】** ビューで Web サービスの出力をプレビューする場合に、Developer ツールにより適用される設定を制御します。

構成プロパティ

Developer ツールは、出力のプレビューまたはマッピングの実行時に構成プロパティを適用します。 **【データビューア】** ビューまたはマッピングの構成プロパティは、**【実行】** ダイアログボックスで設定します。

Data Integration Service のプロパティ

Developer ツールでは、データビューア、マッピング、および Web サービス設定用の **【Data Integration Service】** タブが表示されます。

以下の表に、Data Integration Service に対して構成するプロパティを示します。

プロパティ	説明
デフォルトのデータ統合サービスを使用	マッピングの実行にデフォルトのデータ統合サービスを使用します。 デフォルトでは有効になっています。
データ統合サービス	デフォルトのデータ統合サービスを使用しない場合に、マッピングを実行するデータ統合サービスを指定します。
使用可能な OS プロファイル	データ統合サービスでオペレーティングシステムのプロファイルを使用できる場合、マッピングを実行するためのオペレーティングシステムのプロファイルを指定します。 Developer tool でこのプロパティが表示されるのは、管理者がオペレーティングシステムのプロファイルを少なくとも 1 つユーザーに割り当てた場合のみです。データ統合サービスはユーザーに割り当てられたデフォルトのオペレーティングシステムのプロファイルを使用してマッピングを実行します。オペレーティングシステムのプロファイルは、使用可能なオペレーティングシステムのプロファイルのリストから変更できます。

ソースのプロパティ

Developer ツールでは、データビューア、マッピング、および Web サービス設定用の【ソース】タブが表示されます。

以下の表に、ソースに設定するプロパティを示します。

プロパティ	説明
すべての行の読み取り	ソースからすべての行を読み取る。 デフォルトでは有効になっています。
次の行数まで読み取る	すべての行を読み取らない場合にソースから読み取る最大行数を指定します。 注: カスタマイズデータオブジェクトに書き込むマッピングにこのオプションを指定すると、Data Integration Service はターゲットに書き込む前にターゲットテーブルを切り詰めません。 デフォルトは 1000 です。
すべての文字を読み取る	カラム内のすべての文字を読み取ります。 デフォルトでは無効になっています。
次の文字数まで読み取る	すべての文字を読み取らない場合に各カラムに読み取る最大文字数を指定します。SAP ソースの場合、このプロパティは無視されます。 デフォルトは 4000 です。

結果のプロパティ

Developer ツールでは、データビューアおよび Web サービス設定用の【結果】タブが表示されます。

以下の表に、【データビューア】ビューの結果に対して構成するプロパティを示します。

プロパティ	説明
すべての行を表示	【データビューア】ビューにすべての行を表示します。 デフォルトでは無効になっています。
次の行数まで表示	すべての行を表示しない場合に表示する、最大行数を指定します。 デフォルトは 1000 です。
すべての文字を表示	カラム内のすべての文字を表示します。 デフォルトでは無効になっています。
次の文字数まで表示	すべての文字を表示しない場合に各カラムで表示する、最大文字数を指定します。 デフォルトは 4000 です。

メッセージのプロパティ

Developer ツールでは、Web サービス設定用の【メッセージ】タブが表示されます。

以下の表に、メッセージに設定するプロパティを示します。

プロパティ	説明
要求メッセージで読み取られる最大文字数	入力メッセージで処理する最大文字数を指定します。
要求メッセージで表示される最大文字数	出力メッセージで表示する最大文字数を指定します。

詳細プロパティ

Developer ツールでは、データビューア、マッピング、および Web サービス設定用の【詳細】タブが表示されます。

以下の表に詳細プロパティを示します。

プロパティ	説明
デフォルトの日時形式	マッピングで文字列が日付に変換される場合に、データ統合サービスが使用する日付/時刻形式。 デフォルトは MM/DD/YYYY HH24:MI:SS です。
トレースレベルのオーバーライド	マッピングの各トランスフォーメーションのトレースレベルをオーバーライドします。トレースレベルは、データ統合サービスがマッピングログファイルに送信する情報量を決定します。 以下のいずれかのトレースレベルを選択します。 <ul style="list-style-type: none">- なし。データ統合サービスは、マッピングに設定されたトレースレベルを使用します。- Terse。データ統合サービスは、初期化情報、エラーメッセージ、および却下されたデータの通知をログに記録します。- ノーマル。データ統合サービスは、初期化情報、ステータス情報、検出されたエラー、およびトランスフォーメーション行エラーが原因でスキップされた行をログに記録します。マッピングの結果を要約しますが、個別行のレベルでは行いません。- Verbose initialization。ノーマルトレースに加え、データ統合サービスにより、追加の初期化の詳細、インデックスおよび使用されたデータファイルの名前、および詳細なトランスフォーメーション統計もログに記録されます。- 冗長データ。データ統合サービスにより、Verbose 初期化トレースに加え、マッピングに渡された各行がログに記録されます。また、カラムの精度に合わせて文字列データを切り詰めた場所と、詳細なトランスフォーメーション統計も記録します。 デフォルトは [なし] です。
ソート順	データ統合サービスがマッピングで文字データをソートする順番。 デフォルトは [バイナリ] です。

プロパティ	説明
最適化レベル	<p>データ統合サービスがマッピングに適用する最適化の方法を制御します。</p> <p>なし</p> <p>データ統合サービスは最適化は適用されません。</p> <p>最小</p> <p>データ統合サービスは初期プロジェクション最適化方式を適用します。</p> <p>ノーマル</p> <p>データ統合サービスは、初期プロジェクション、初期選択、ブランチ刈り込み、プッシュイン、グローバル述部、述部の最適化方式を適用します。ノーマルがデフォルトの最適化レベルです。</p> <p>完全</p> <p>データ統合サービスは、コストベース、初期プロジェクション、初期選択、ブランチ刈り込み、述部、プッシュイン、準結合、データシップ結合の最適化方式を適用します。</p> <p>デフォルトは「通常」です。</p>
高精度	<p>マッピングを高精度で実行します。</p> <p>[高精度] を選択すると、データ値の精度が高くなります。15 桁以上の精度など、マッピングが大きな数値を生成する場合や、正確な値を必要とする場合に [高精度] を有効にします。[高精度] を有効にすると、桁数の多い数値で精度の損失を防ぐことができます。</p> <p>デフォルトでは有効になっています。</p>
クライアントにログを送信	<p>Developer ツールでログファイルを表示できるようにします。このオプションを無効にした場合、ログファイルは Administrator ツールで表示する必要があります。</p> <p>デフォルトでは有効になっています。</p>

データビューアの構成

データビューア構成は、出力を【データビューア】ビューでプレビューする場合に Developer ツールが適用する設定を制御します。

以下のオブジェクトに対する出力をプレビューする場合に、データビューア構成を選択できます。

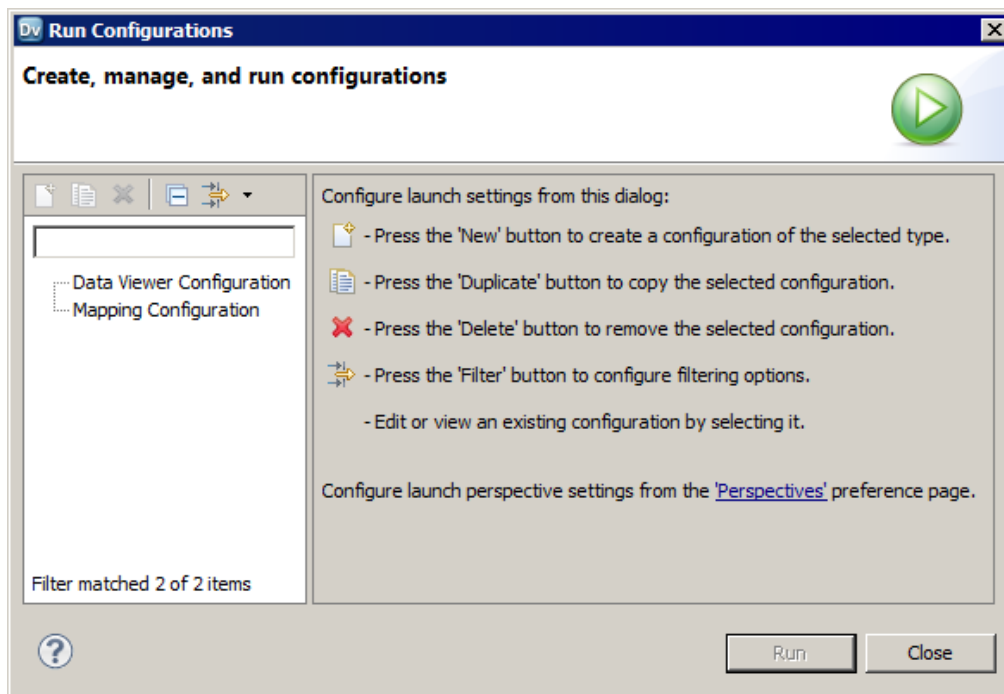
- カスタムデータオブジェクト
- 論理データオブジェクト
- 論理データオブジェクト読み取りマッピング
- 物理データオブジェクト
- マッピング内のソースとトランスフォーメーション
- 仮想ストアドプロシージャ
- 仮想テーブル
- 仮想テーブルマッピング


データビューア構成の作成

データビューア構成を作成し、出力を【データビューア】ビューでプレビューする場合に Developer ツールが適用する設定を制御します。

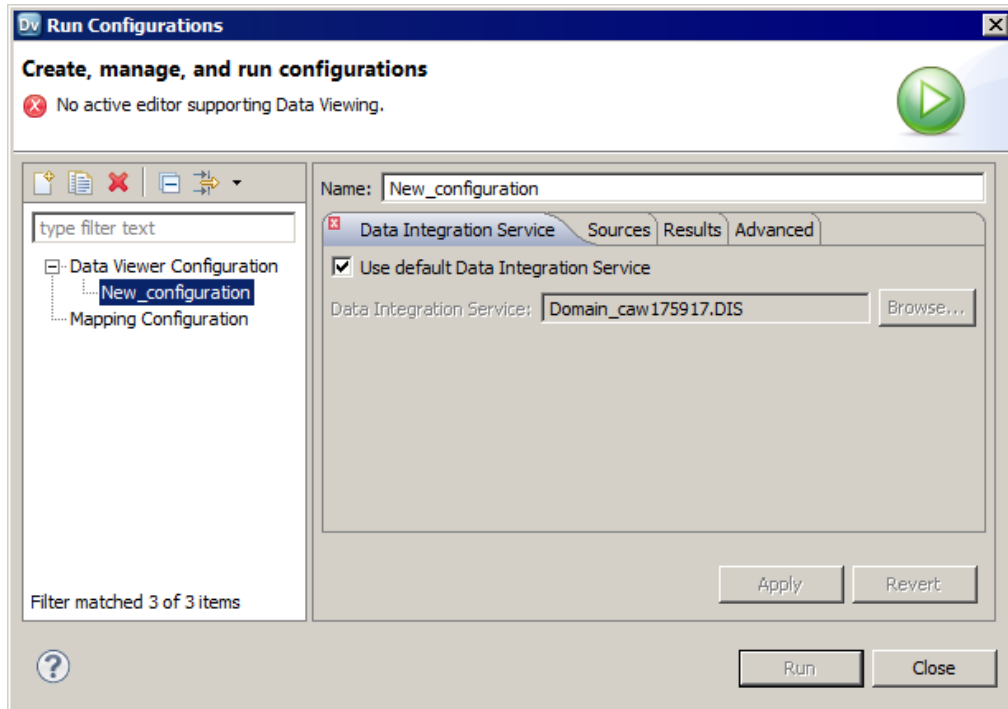
1. 【実行】 > 【実行ダイアログを開く】 をクリックします。

【実行設定】 ダイアログボックスが表示されます。



2. 【実行】 をクリックします。
3. 【新規】 ボタンをクリックします ()。

【実行設定】ダイアログボックスの右側のパネルには、データビューアの設定プロパティが表示されます。



4. データビューアの設定の名前を入力します。
5. データビューアの設定プロパティを設定します。
6. 【適用】をクリックします。
7. 【閉じる】をクリックします。

データビューア構成が作成されます。

マッピング構成

マッピング設定は、マッピングを【実行設定】ダイアログボックスまたはコマンドラインから実行する場合に Developer ツールが使用するマッピングのデプロイメントのプロパティを制御します。

Developer ツールから実行するマッピングにマッピング設定を適用するには、【実行設定】ダイアログボックスからマッピングを実行する必要があります。【実行】メニューまたはマッピングエディタから実行すると、Developer ツールは、デフォルトのマッピングのデプロイメントのプロパティを使用してマッピングを実行します。

コマンドラインから実行するマッピングにマッピングのデプロイメントのプロパティを適用するには、マッピングをアプリケーションに追加するときにマッピング構成を選択します。選択したマッピング構成は、アプリケーション内のすべてのマッピングに適用されます。

マッピングのデプロイメントのプロパティは、アプリケーションを編集するときに変更できます。管理者は Administrator ツールを使用してマッピングのデプロイメントのプロパティを変更することもできます。変更内容を有効にするには、アプリケーションを再デプロイする必要があります。

マッピング構成の作成

マッピング構成を作成し、マッピングを【実行】ダイアログボックスまたはコマンドラインから実行する場合に Developer ツールが使用するマッピングのデプロイメントのプロパティを制御します。

1. 【実行】 > 【実行ダイアログを開く】をクリックします。
【実行設定】ダイアログボックスが表示されます。
2. 【マッピング設定】をクリックします。
3. 【新規】ボタンをクリックします。
【実行設定】ダイアログボックスの右側のパネルには、マッピング設定のプロパティが表示されます。
4. マッピング設定の名前を入力します。
5. マッピング設定のプロパティを構成します。
6. 【適用】をクリックします。
7. 【閉じる】をクリックします。
マッピング構成が作成されます。

Web サービスの設定

Web サービス設定では、【データビューア】ビューで Web サービスの出力をプレビューする場合に Developer ツールによって適用される設定を制御します。

Web サービス設定を作成して、特定の Web サービスに使用する設定を制御します。操作マッピングまたは操作マッピングのトランスフォーメーションの出力をプレビューする際に、Web サービス設定を選択できます。

注: プレビューするすべての Web サービスに適用する Web サービス設定を作成するには、【設定】ダイアログボックスを使用してデフォルトの Web サービス設定を更新します。

Web サービス設定の作成

Web サービス設定を作成して、【データビューア】ビューで Web サービスの出力をプレビューする場合に Developer ツールで適用される設定を制御します。

1. 【実行】 > 【実行ダイアログを開く】をクリックします。
【実行】ダイアログボックスが表示されます。
2. 【Web サービスの設定】をクリックします。
3. 【新規】をクリックします。
4. Web サービス設定の名前を入力します。
5. Web サービス設定のプロパティを設定します。
6. 【適用】をクリックします。
7. 【閉じる】をクリックします。

デフォルトの構成プロパティの更新

データビューア、マッピング、および Web サービスのデフォルトの設定プロパティを更新できます。

1. 【ウィンドウ】 > 【設定】をクリックします。
【設定】ダイアログボックスが表示されます。
2. 【Informatica】 > 【実行設定】をクリックします。

3. **【データビューア】**、**【マッピング】**、または **【Web サービス】** 設定を選択します。
4. データビューア、マッピング、または Web サービスのデフォルトの設定プロパティを設定します。
5. **【OK】** をクリックします。
デフォルトの構成プロパティが更新されます。

構成のトラブルシューティング

同じ名前で大文字と小文字が異なる 2 つの設定を作成したが、Developer ツールを閉じて再度開くと、1 つの設定が見つからない。

データビューア構成名とマッピング構成名では大文字と小文字が区別されません。同じ名前で大文字と小文字が異なる複数の構成を作成すると、Developer ツールはユーザーがセッションを終了した時点で構成の 1 つを削除します。Developer ツールはこの 2 つの構成名を一意とは見なしません。

長い名前の構成を作成すると、ファイルを書き込めないというエラーメッセージが表示される。

Developer ツールは、ビューア構成とマッピング構成を、Developer ツールを実行するマシン上のファイルに格納します。例えば 100 文字以上の長い名前の構成を作成すると、このファイルはハードウェアドライブに保存されません。

この問題を回避するには、構成名を短くします。

データのエクスポート

【データビューア】 ビューに表示するデータを、TXT ファイルや CSV ファイルなどのタブ区切りのフラットファイルにエクスポートできます。データのローカルコピーを作成するには、データをエクスポートします。

1. **【データビューア】** ビューで、結果を右クリックして **【データのエクスポート】** 選択します。
2. ファイル名と拡張子を入力します。
3. ファイルの保存先を選択します。
4. **【OK】** をクリックします。

オブジェクトの依存関係

オブジェクトを変更または削除する前に、モデルリポジトリ内でそのオブジェクトの影響を受けるすべてのオブジェクトに関して、オブジェクトの依存関係を確認することができます。選択したオブジェクトに依存しているすべてのオブジェクトと、そのオブジェクトが依存しているオブジェクトに関して、オブジェクトの依存関係を確認することができます。

例えば、複数のアプリケーションでデプロイされたデータオブジェクトを削除する必要があります。しかし、このデータオブジェクトを削除することでデプロイ済みのアプリケーションに影響が及ぶかが分かっていません。このような場合に、オブジェクトの依存関係を表示することで、このデータオブジェクトがデプロイ済みのアプリケーションに影響を及ぼすかどうかを判定することができます。データオブジェクトを削除したら、アプリケーションをデプロイしなおすことができます。

【オブジェクトの依存関係】 ビューで、オブジェクトの依存関係を表示することができます。読み取り権限のないプロジェクト内に依存オブジェクトがある場合、Developer ツールでは**【オブジェクトの依存関係】** ビューにそのオブジェクトが表示されません。Developer ツールに、Developer ツールがオブジェクトにアクセスできないことを伝えるメッセージが表示されます。

オブジェクトの依存関係の表示

【オブジェクトの依存関係】 ビューの中のオブジェクトに関して、オブジェクトの依存関係を表示することができます。ユーザーは、**【Object Explorer】** ビューから選択するオブジェクトに関して、または**【接続エクスプローラ】** ビューから選択した接続に関して、そのオブジェクトの依存性を表示できます。

次のタスクを実行すると、オブジェクトの依存性を表示することができます。

循環依存の関係の表示。

2 つのオブジェクトが相互依存している場合、循環依存性が生じます。例えば、オブジェクト A はオブジェクト B に依存し、オブジェクト B はオブジェクト A に依存するとします。オブジェクトの依存関係ツリーで同じオブジェクトが 2 回出てくることを検知した場合、Developer ツールはオブジェクトの 2 回目以降のインスタンスを表示しません。その代わりに、Developer ツールは循環依存が発生していることを示す丸のアイコンをオブジェクトに追加します。

ダウンストリームまたはアップストリームの依存関係の表示。

選択されたオブジェクトに依存するオブジェクトを確認するには、ダウンストリームの依存関係を表示します。デフォルトでは Developer ツールにダウンストリームの依存関係が表示されます。

選択されたオブジェクトが依存するオブジェクトを確認するには、アップストリームの依存関係を表示します。

オブジェクトの依存関係のフィルタリング。

オブジェクトの依存関係にフィルタリングを実行して依存するオブジェクトのリストを絞り込むことができます。オブジェクトタイプでのフィルタリングと、プロジェクトでのフィルタリングを選ぶことができます。例えば、特定のオブジェクトがワークフローに影響を及ぼすかを知る必要があるとします。この場合は、オブジェクトタイプでフィルタリングしてワークフローを選択します。

オブジェクトの依存関係の履歴の表示。

オブジェクトの依存関係のうち直近の 10 件に関する履歴を表示します。

オブジェクトの依存関係の表示

依存関係を表示する際、フィルタを適用したり、履歴をクリアしたり、特定の依存性に関する詳細を選択したりすることができます。

1. **【Object Explorer】** ビューまたは**【接続エクスプローラ】** ビューで、オブジェクトまたは接続を右クリックしてから**【依存関係の検索】** をクリックします。

【オブジェクトの依存関係】 ビューには、オブジェクトの依存関係ツリーの中で選択されたオブジェクトまたは接続に関するオブジェクトの依存関係がリスト表示されます。

2. オブジェクトの依存関係ツリー内のオブジェクトまたは接続を選択すると、オブジェクトまたは接続のプロパティを表示できます。
3. 必要に応じて依存性をフィルタリングする場合は、**【フィルタ】** アイコンをクリックして、オブジェクトまたはプロジェクトのタイプでフィルタリングするように選択します。
4. 次のタイプの依存関係を表示できます。
 - 循環。Developer ツールは循環依存が発生していることを示す丸のアイコンをオブジェクトに追加します。

- アップストリーム。アップストリームの依存関係を表示する場合は、**[アップストリームの依存関係]** アイコンをクリックします。
 - ダウンストリーム。ダウンストリームの依存関係を表示する場合は、**[ダウンストリームの依存関係]** アイコンをクリックします。
5. 必要に応じて依存関係の履歴を表示する場合は、**[履歴]** アイコンをクリックして、オブジェクトの依存関係の履歴を表示します。必要に応じて、**[履歴のクリア]** をクリックして履歴をクリアします。

オブジェクトの依存関係のフィルタリング

オブジェクトに依存関係ツリーを **[タイプ]** および **[プロジェクト]** でフィルタリングできます。フィルタされた結果で、オブジェクトの一部が親オブジェクトに直接的に依存しないものであることがあります。これらのオブジェクトにはプラス (+) アイコンが表示され、オブジェクトの依存関係ツリー内の親オブジェクトへの絶対パスが非表示になります。このパスを拡大して親オブジェクトまでのすべてのオブジェクトレベルを挿入することもできます。

1. **[オブジェクトの依存関係]** ビューで、**[フィルタ]** アイコンをクリックします。
[フィルタ] ダイアログボックスが開きます。
2. **[タイプ]** タブでのデフォルトの選択を受け入れて、オブジェクトタイプでフィルタ表示します。
3. 必要があれば、**[プロジェクト]** タブをクリックして、モデルリポジトリ内のすべてのプロジェクトでフィルタリングするか、または各プロジェクトでフィルタリングするかを選択します。
 - モデルリポジトリ内のすべてのプロジェクトでフィルタリングする場合は、すべてのプロジェクトを含めるオプションを選択します。
 - 各プロジェクトでフィルタリングする場合は、開いたプロジェクトを含めるオプションを選択します。
4. オブジェクトタイプまたはプロジェクトを個別に選択するのか、すべてのオブジェクトタイプまたはプロジェクトを選択するのか、あるいは選択してあったオブジェクトタイプとプロジェクトをクリアするのかを指定します。
 - オブジェクトタイプまたはプロジェクトを個別に選択する場合は、オブジェクトタイプまたはプロジェクトをそれぞれ選択します。
 - すべてのオブジェクトタイプまたはプロジェクトを選択する場合は、**[すべて選択]** をクリックします。
 - 選択してあったオブジェクトタイプとプロジェクトをクリアする場合は、**[すべて選択解除]** をクリックします。
5. **[OK]** をクリックします。
フィルタリングされたオブジェクトの依存関係ツリーに **[オブジェクトの依存関係]** ビューが表示されます。
6. 必要に応じて、オブジェクトの名前に対してプラス (+) のアイコンが表示された場合に、そのオブジェクトを右クリックし、**[パスを拡大]** をクリックして、親のオブジェクトまでのすべてのオブジェクトレベルを挿入します。

ログ

Data Integration Service は、マッピングやプロファイルの実行、データのプレビュー、または SQL クエリの実行時に、ログイベントを生成します。ログイベントには Data Integration Service によって実行されるタスク、エラー、ロード要約についての情報とトランスフォーメーション統計が含まれます。

Developer ツールで生成されたログを表示して、ローカルディレクトリに保存できます。

【データビューア】ビューの【ログの表示】ボタンからログイベントを表示できます。

【実行】 > 【マッピングの実行】 からマッピングを実行する場合、【進捗状況】ビューでログイベントを表示できます。Developer ツールでログイベントを開くには、マッピング実行のリンクをクリックして、【ログに移動】を選択します。

プロファイルを実行する場合、Monitoring ツールからログイベントを表示できます。

ログをファイルに保存するには、【ファイル】 > 【名前を付けてコピーを保存】をクリックして、ディレクトリを選択します。デフォルトでは、ログファイルは次のディレクトリに保存されます。c:\[TEMP]\AppData\Local\Temp

ログファイル形式

ログファイルの情報は、実行時のイベントのシーケンスによって異なります。ログに送信される情報量は、トレースレベルによって決まります。

Data Integration Service は、マッピングやプロファイルの実行、データのプレビュー、または SQL クエリの実行時に、以下の情報でログファイルを更新します。

論理 DTM メッセージ

マッピングのコンパイル、最適化、変換の準備に関する情報が含まれます。ログイベントおよび情報量は、設定されている構成のプロパティによって異なります。

データ変換マネージャ (DTM) メッセージ

ソースへの接続の確立、データの読み取り、データのトランスフォーム、ターゲットへのデータのロードに関する情報が含まれます。

ロード要約およびトランスフォーメーション統計メッセージ

ソースから読み取られた行数、ターゲットに出力された行数、拒否された行数、および実行時間に関する情報が含まれます。

検証設定

【検証ログ】ビューに表示されるエラーメッセージの数を制限できます。【検証ログ】ビューでオブジェクトまたはオブジェクトタイプ別にエラーメッセージをグループ化することもできます。

エラーメッセージのグループ化

【検証ログ】ビューのエラーメッセージをグループ化して、メッセージをオブジェクトまたはオブジェクトタイプ別に分類します。これを行わなかった場合、メッセージはアルファベット順に表示されます。

【検証ログ】ビューのエラーメッセージをグループ化するには、【メニュー】 > 【グループ別】を選択し、【オブジェクト】または【オブジェクトタイプ】を選択します。

エラーメッセージのグループを削除するには、【メニュー】 > 【グループ別】 > 【なし】を選択します。エラーメッセージはグループ化が解除され、【検証ログ】ビューでアルファベット順に一覧表示されます。

エラーメッセージの制限

【検証ログ】ビューに表示されるエラーメッセージの数を制限できます。この制限により、グループに表示されるメッセージ数や【検証ログ】ビューに表示されるメッセージの合計数が決まります。エラーメッセージはアルファベット順に一覧表示され、制限を適用すると、下から上に向かって削除されます。

1. 【ウィンドウ】 > 【設定】 をクリックします。
【設定】 ダイアログボックスが表示されます。
2. 【Informatica】 > 【検証】 を選択します。
3. 必要に応じて、エラーの上限を設定し、表示される項目数を設定します。
デフォルトは 100 です。
4. デフォルト値に戻すには、【デフォルトの復元】 をクリックします。
5. 【適用】 をクリックします。
6. 【OK】 をクリックします。

Developer tool からのジョブの監視

Developer tool から Monitoring ツールにアクセスし、アプリケーションとジョブのステータスを監視できます。管理者は、Administrator ツールでアプリケーションとジョブを監視することもできます。

アプリケーションとジョブを監視して、オブジェクトのプロパティ、ランタイム統計、およびランタイムレポートを表示します。

ジョブを監視する場合、ジョブのサマリ統計または実行統計を表示できます。【サマリ統計】ビューにドメイン内のジョブのステータス概要が図で表示されます。

【実行統計】ビューにジョブの全般のプロパティおよびステータス情報が表示されます。例えば、ジョブの開始者やジョブの完了に要した時間を表示できます。マッピングジョブを監視する場合、ジョブ実行のスループットおよびリソースの使用状況統計も表示できます。

Developer tool からアプリケーションとジョブを監視するには、【進行状況】ビューを開き、【表示】メニュー > 【ジョブの監視】 をクリックします。アプリケーションを実行するデータ統合サービスを選択して、【OK】 をクリックします。Monitoring ツールが開きます。

第 9 章

アプリケーションデプロイメント

この章では、以下の項目について説明します。

- [アプリケーションデプロイメントの概要, 140 ページ](#)
- [n アプリケーションの作成, 141 ページ](#)
- [アプリケーションのプロパティ, 141 ページ](#)
- [アプリケーションデプロイメント, 143 ページ](#)
- [オブジェクトのデプロイメント, 143 ページ](#)
- [アプリケーションアーカイブファイルへのデプロイメント, 144 ページ](#)
- [アプリケーションの再デプロイメント, 145 ページ](#)
- [アプリケーションの作成、デプロイ、更新の方法, 146 ページ](#)

アプリケーションデプロイメントの概要

アプリケーションは、物理データオブジェクト、論理データオブジェクト、データサービス、マッピング、マップレット、トランスフォーメーション、Web サービス、ワークフローを含むことができるデプロイ可能なオブジェクトです。

Developer tool の外部からアプリケーションオブジェクトにアクセスできるようにするには、そのオブジェクトを含むアプリケーションをデプロイします。アプリケーションを初めから作成するか、オブジェクトのデプロイに合わせてアプリケーションを作成することができます。

オブジェクトをデプロイすると、そのオブジェクトにはデータ構造の変更は反映されなくなります。オブジェクトはデータ統合サービスまたはネットワークファイルシステムにデプロイできます。

アプリケーションをデータ統合サービスにデプロイすると、アプリケーションが実行され、エンドユーザーがアプリケーションに接続できるようになります。アプリケーションのオブジェクトのタイプに応じて、適切な権限のあるエンドユーザーはオブジェクトに対してクエリを実行したり、Web サービスにアクセスしたり、コマンドラインからマッピングやワークフローを実行したりすることができます。また、ユーザーがサードパーティのクライアントツールでオブジェクトをクエリできるようにオブジェクトをデプロイすることもできます。

オブジェクトをネットワークファイルシステムにデプロイすると、Developer tool によってアプリケーションアーカイブファイルが作成されます。バージョン管理システムにオブジェクトをアーカイブする場合は、オブジェクトをアーカイブファイルにデプロイします。管理者がデータ統合サービスにオブジェクトをデプロイする必要がある場合、Administrator ツールを使用してアプリケーションアーカイブファイルをデプロイできます。また、アプリケーションアーカイブからモデルリポジトリ内のプロジェクトまたはフォルダへオブジェクトをインポートすることもできます。

オブジェクトに変更を加えた場合、変更を有効にするにはこのオブジェクトを含むアプリケーションを再デプロイする必要があります。

例

検索機能を実行するマッピングと、ショッピングカートに選択した検索結果を入れる別のマッピングを作成します。この2つのマッピングを含むアプリケーションを作成し、データ統合サービスにアプリケーションをデプロイします。アプリケーションオブジェクトの出力をテストしたら、オブジェクトに変更を加えてアプリケーションを再デプロイします。アプリケーションはアプリケーションアーカイブファイルにもデプロイします。管理者はアーカイブファイルをバージョン管理システムにチェックインします。

n アプリケーションの作成

アプリケーションを作成してデプロイし、アプリケーションに含まれるマッピングおよびその他のオブジェクトを実行します。

アプリケーションはいちから作成するか、オブジェクトのデプロイ時に作成することができます。アプリケーションを初めから作成する場合は、アプリケーションに含めるオブジェクトを選択します。マッピングや物理データオブジェクトなどのオブジェクトを追加できます。

論理データオブジェクトをアプリケーションに直接追加することはできません。しかし、論理データオブジェクトを含む Web サービスおよび SQL データサービス用のアプリケーションは作成できます。

アプリケーションは検証することができます。アプリケーションが有効かどうかは、含まれるオブジェクトが有効かどうかによって決まります。例えば、マッピングを含むアプリケーションが有効であるためには、マッピング内のオブジェクト間のリンク設定が有効である必要があります。アプリケーションが有効でない場合は、[検証ログ] ビューまたはエラーダイアログボックスにエラーが表示されます。

アプリケーションのプロパティ

アプリケーションを作成すると、**アプリケーション**エディタでプロパティを編集できます。

アプリケーション全般のプロパティ

以下の表に、**[概要]** タブで設定できるアプリケーション全般のプロパティを示します。

プロパティ	説明
名前	アプリケーションの名前。
説明	オプション。アプリケーションの説明。

マッピングのデプロイメントのプロパティ

以下の表では、アプリケーションがマッピングを含む場合に【詳細】タブで設定できるマッピングのデプロイメントのプロパティを示します。

プロパティ	説明
デフォルトの日時形式	マッピングで文字列が日付に変換される場合に、データ統合サービスが使用する日付/時刻形式。 デフォルトは MM/DD/YYYY HH24:MI:SS です。
トレースレベルのオーバーライド	マッピングの各トランスフォーメーションのトレースレベルをオーバーライドします。トレースレベルは、データ統合サービスがマッピングログファイルに送信する情報量を決定します。 以下のいずれかのトレースレベルを選択します。 <ul style="list-style-type: none">- なし。トランスフォーメーションごとに設定したトレースレベルはオーバーライドされません。- 簡易。データ統合サービスは、初期化、エラーメッセージ、および拒否されたデータの通知に関する情報にログに記録します。- ノーマル。データ統合サービスは、初期化、ステータス、検出されたエラー、およびトランスフォーメーション行エラーが原因でスキップされた行に関する情報をログに記録します。データ統合サービスは、個別の行レベルではなくマッピング結果を要約します。- 詳細 - 初期化。ノーマルトレースに加え、データ統合サービスにより、追加の初期化の詳細、インデックスおよび使用されたデータファイルの名前、および詳細なトランスフォーメーション統計もログに記録されます。- 詳細 - データ。データ統合サービスにより、Verbose 初期化トレースに加え、マッピングに渡された各行がログに記録されます。ログには、データ統合サービスがカラムの精度に合わせて文字列データを切り詰めた場所を記録します。このログには、詳細なトランスフォーメーション統計が含まれます。データ統合サービスではトランスフォーメーションを処理する際にブロック中のすべての行の行データが書き込まれます。 デフォルトは [なし] です。
ソート順	データ統合サービスがマッピングで文字データをソートする順番。 デフォルトは [バイナリ] です。
最適化レベル	データ統合サービスがマッピングに適用する最適化の方法を制御します。 <ul style="list-style-type: none">- なし。データ統合サービスは最適化は適用されません。- 最小。データ統合サービスは初期プロジェクション最適化方式を適用します。- ノーマル。データ統合サービスは、初期プロジェクション、初期選択、ブランチ刈り込み、プッシュイン、プッシュダウン、述部の最適化方式を適用します。ノーマルがデフォルトの最適化レベルです。- 完全。データ統合サービスは、コストベース、初期プロジェクション、初期選択、ブランチ刈り込み、述部、プッシュイン、プッシュダウン、準結合の最適化方式を適用します。 デフォルトは [通常] です。
高精度	マッピングを高精度で実行します。 [高精度] を選択すると、データ値の精度が高くなります。15 桁以上の精度など、マッピングが大きな数値を生成する場合や、正確な値を必要とする場合に [高精度] を有効にします。[高精度] を有効にすると、桁数の多い数値で精度の損失を防ぐことができます。 デフォルトでは有効になっています。

アプリケーションデプロイメント

データ統合サービスにアプリケーションをデプロイすると、ユーザーは、マッピング、ワークフロー、およびアプリケーションに含まれる他のオブジェクトにアクセスできるようになります。

アプリケーションをデプロイすると、データ統合サービスによってアプリケーションオブジェクトが実行されます。

デプロイするアプリケーションにパラメータセットを追加することができます。パラメータセットはモデルリポジトリ内のオブジェクトであり、マッピングおよびワークフローで使用するパラメータとパラメータ値を含みます。データ統合サービスでワークフローまたはマッピングを実行するときにパラメータ値を使用するには、ワークフローまたはマッピングと一緒にパラメータセットをデプロイします。アプリケーションに複数のパラメータセットを追加すると、マッピングまたはワークフローの実行ごとに異なるパラメータセットを使用できます。

同じ名前の完全なアプリケーションを同じデータ統合サービスにデプロイすると、データ統合サービスは、デプロイされたアプリケーションとアプリケーション内のすべてのオブジェクトを上書きします。

オブジェクトのデプロイメント

アプリケーションとして、またはアプリケーションの一部であるデータサービスとしてオブジェクトをデプロイすることができます。

Web サービスまたは SQL データサービスとしてデプロイできるオブジェクトもあります。最初に、アプリケーションを作成してオブジェクトを追加します。次に、アプリケーションをデプロイすると、Developer tool によってオブジェクトに基づいてサービスを作成するように求められます。サービスがアプリケーションに追加されます。

また、ユーザーがサードパーティのクライアントツールでオブジェクトをクエリできるようにオブジェクトをデプロイすることもできます。

以下のオブジェクトをデプロイすると、Developer tool によってアプリケーションを作成するように求められます。

- マッピング
- ワークフロー

Web サービスとしてのオブジェクトのデプロイ

以下のオブジェクトを Web サービスとしてデプロイできます。

- マップレット
- Web サービスコンシューマトランスフォーメーションを除くトランスフォーメーション
- フラットファイルデータオブジェクト
- リレーショナルデータオブジェクト
- 論理データオブジェクト

Web サービスとしてオブジェクトをデプロイすると、Developer tool によって、オブジェクトに基づいて Web サービスを作成し、この Web サービスを含むアプリケーションを作成するように求められます。

Web サービスとしてオブジェクトをデプロイする場合は、以下の情報を入力します。

プロパティ	説明
名前	Web サービスの名前。
場所	アプリケーションを格納するモデルリポジトリプロジェクトフォルダ。
名前空間	ユーザーがアクセスできる Web サービスの URL。データ統合サービスは、WSDL ファイルで定義されている URL をオーバーライドします。
プレフィックス	ターゲット名前空間のプレフィックス。

SQL データサービスとしてのオブジェクトのデプロイ

以下のデータオブジェクトを SQL データサービスとしてデプロイできます。

- 物理データオブジェクト
- 論理データオブジェクト

SQL データサービスとしてデータオブジェクトをデプロイすると、Developer tool によって、オブジェクトに基づいて SQL データサービスを作成し、このサービスを含むアプリケーションを作成するように求められます。

SQL データサービスとしてオブジェクトをデプロイする場合は、以下の情報を入力します。

プロパティ	説明
名前	Web サービスの名前。
場所	アプリケーションを格納するモデルリポジトリプロジェクトフォルダ。

アプリケーションアーカイブファイルへのデプロイメント

アプリケーションアーカイブファイルには、アプリケーションのオブジェクトおよびメタデータが XML 形式で含まれています。

アプリケーションをこの形式にデプロイすると、アプリケーションに関するすべての情報が XML 形式で保存されます。ファイルの拡張子は.iar です。

以下の理由により、アプリケーションアーカイブファイルを作成する場合があります。

- アプリケーションをデプロイします。組織でデータ統合サービスへのアプリケーションのデプロイを管理者に限定した場合、管理者はアーカイブファイルからデータ統合サービスにアプリケーションをデプロイすることができます。管理者は管理ツールまたは infacmd dis deployapplication を使用します。
- モデルリポジトリにアプリケーションをインポートします。管理者は、infacmd oie importobjects を使用してアーカイブファイルからモデルリポジトリにアプリケーションをインポートできます。モデルリポジトリ内にすでに保存されたオブジェクトを含むアプリケーションアーカイブをインポートする場合、Developer tool によって新しいプロジェクトフォルダが作成されます。

注: アプリケーションをインポートするモデルリポジトリのバージョンには、そのアプリケーションをアーカイブファイルにエクスポートした元のバージョンとの互換性が必要です。

- 別のシステムにアプリケーションアーカイブファイルをアーカイブします。例えば、モデルリポジトリがバージョン管理システムに統合されていない場合、管理者はアーカイブファイルをバージョン管理システムにチェックインできます。

アプリケーションの再デプロイメント

アプリケーションの内容を変更したら、変更を適用するためにアプリケーションを再デプロイする必要があります。

アプリケーションオブジェクトの編集、追加、または削除により、アプリケーションを更新することができます。

更新したアプリケーションをデプロイする場合は、アプリケーションの状態情報の保持または破棄を選択します。状態情報は、マッピングプロパティ、マッピング出力、およびランタイムオブジェクトのプロパティを参照しています。

実行中のターゲットアプリケーションを更新する場合は、ターゲットアプリケーションの停止を選択する必要があります。実行中のオブジェクトを強制終了する代わりに、アプリケーションの名前を変更するか、アプリケーションを別のサービスにデプロイすることもできます。

アプリケーションを更新してネットワークファイルシステムにエクスポートする場合、アプリケーションアーカイブファイルを置換するか、デプロイメントを取り消すことができます。アプリケーションアーカイブファイルを上書きする場合は、アプリケーション内のオブジェクトが置換され、オブジェクトのプロパティがリセットされます。

アプリケーションの状態情報

アプリケーションを再デプロイする場合は、デプロイしたアプリケーションの状態情報の保持または破棄を選択できます。

状態情報は、マッピングプロパティと、マッピング出力やシーケンスジェネレータトランスフォーメーションなどのランタイムオブジェクトのプロパティを参照しています。状態情報を保持すると、デプロイしたアプリケーションでこれらの設定およびプロパティが保持されます。状態情報を破棄すると、デプロイしたアプリケーションでこれらの設定およびプロパティの状態が破棄されます。

例: 設定可能なプロパティの保持または破棄

アプリケーションには、設定可能なランタイムプロパティを使用したマッピングが含まれます。[高精度] プロパティを True に設定します。アプリケーションをデプロイしたら、マッピングを編集して [高精度] プロパティを False に変更します。アプリケーションを再デプロイし、**【状態情報を保持する】** にチェックを入れると、データ統合サービスはデプロイしたアプリケーションの状態情報を保持し、ランタイムプロパティへの変更が認識されません。プロパティへの変更を有効にするには、**【状態情報を保持する】** のチェックを解除する必要があります。

例: シーケンスの保持または破棄

マッピングには、ターゲットテーブルの行に対して一意のキーを生成するシーケンスジェネレータトランスフォーメーションが含まれます。マッピングを含むアプリケーションをデプロイしたら、マッピングが実行され、シーケンスジェネレータトランスフォーメーションは 1~2000 の行に対するキーを生成します。次にマッピングが実行されると、シーケンスジェネレータトランスフォーメーションは 2001 以降のキーを生成しようとしていますが、シーケンスが 10000 から開始されるように設定します。新しい開始値を指定できるようにシーケンスジェネレータトランスフォーメーションを編集します。アプリケーションを再デプロイし、**【状態情報を保持する】**

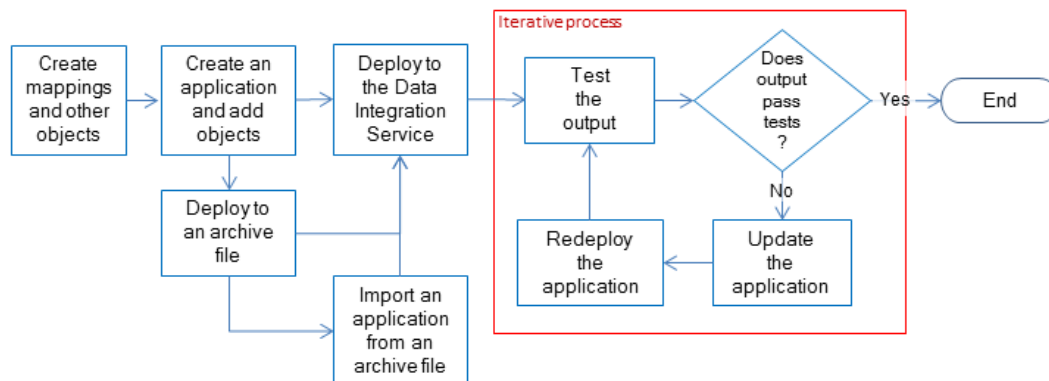
る] にチェックを入れると、データ統合サービスはデプロイしたアプリケーションの状態情報を保持し、設定への変更が認識されません。マッピングが再度実行されると、シーケンスは 2001 から再開します。設定への変更を有効にするには、[状態情報を保持する] のチェックを解除する必要があります。

注: アプリケーションの状態を保持または破棄しても、シーケンスジェネレータトランスフォーメーションの [リセット] プロパティには影響しません。

アプリケーションの作成、デプロイ、更新の方法

マッピングと他のオブジェクトを作成および編集し、アプリケーションにデプロイしてエンドユーザーが使用できるようにします。

以下の画像に、アプリケーションの開発およびデプロイのプロセスを示します。



1. マッピング、ワークフロー、トランスフォーメーション、および他のオブジェクトを作成し、データにアクセスしてデータを変換します。
2. アプリケーションを作成し、そのアプリケーションにオブジェクトを追加します。
3. データ統合サービスまたはアプリケーションアーカイブファイルへのアプリケーションのデプロイを選択します。
 - データ統合サービスでオブジェクトを実行できるように、データ統合サービスにアプリケーションをデプロイします。
 - 管理者がファイルをアーカイブしたり、アーカイブファイルからアプリケーションをデプロイしたり、またはアプリケーションアーカイブファイルからモデルリポジトリにアプリケーションをインポートしたりできるように、アプリケーションアーカイブファイルにアプリケーションをデプロイします。
4. 要件を満たすためにオブジェクトをさらに開発する必要がある場合、オブジェクトを編集してアプリケーションを更新し、データ統合サービスにそのアプリケーションを再デプロイすることができます。

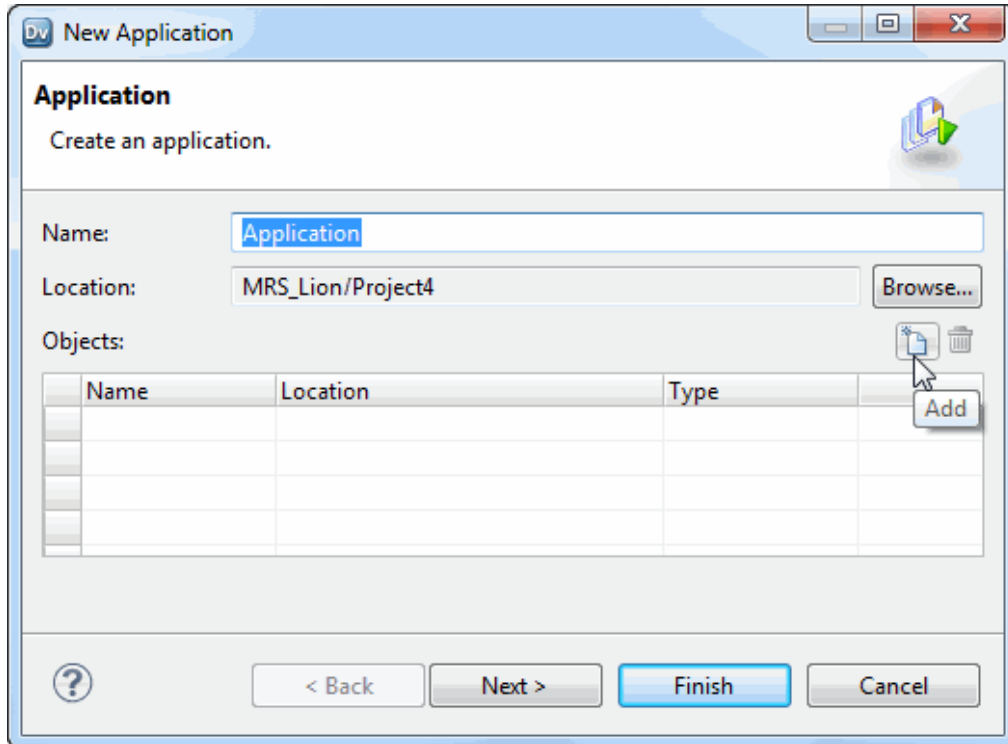
管理者はアプリケーションをデプロイし、コマンドラインまたは Administrator ツールを使用して他の管理タスクを実行できます。これらのタスクの詳細については、『*Informatica アプリケーションサービスガイド*』を参照してください。

アプリケーションの作成

アプリケーションの作成時に、アプリケーションに含めるオブジェクトを選択します。1 つ以上のオブジェクトをデプロイして、エンドユーザーがサードパーティのツールを使用してデータにアクセスできるように、アプリケーションを作成します。

1. **【オブジェクトエクスプローラ】** ビューで、プロジェクトまたはフォルダを選択します。
2. **【ファイル】 > 【新規】 > 【アプリケーション】** をクリックします。

【新しいアプリケーション】 ダイアログボックスが表示されます。



3. アプリケーションの名前を入力します。
4. **【参照】** をクリックし、アプリケーションの場所を選択します。
アプリケーションはプロジェクトまたはフォルダ内に作成する必要があります。
5. **【追加】** ボタンをクリックし、アプリケーションにオブジェクトを追加します。
【オブジェクトの追加】 ダイアログボックスが表示されます。
6. 1 つ以上のオブジェクトを選択し、**【OK】** をクリックします。
選択したオブジェクトが **【新しいアプリケーション】** ダイアログボックスに一覧表示されます。
7. **【パラメータセット】** ビューを選択し、アプリケーションにパラメータセットを追加します。**【完了】** をクリックすると、パラメータセットの追加をスキップします。
パラメータセットの追加を選択すると、**【パラメータセットの追加】** ダイアログボックスが表示されます。**【完了】** をクリックすると、Developer tool はプロジェクトまたはフォルダにアプリケーションを追加します。
8. アプリケーションにパラメータセットを追加するには、**【追加】** ボタンをクリックします。
【パラメータセットの追加】 ダイアログボックスが表示されます。
9. パラメータセットを選択してアプリケーションに追加し、**【OK】** をクリックします。
10. **【完了】** をクリックし、アプリケーションを作成します。

アプリケーションを作成したら、データ統合サービスにすぐにデプロイするか、アプリケーションアーカイブにデプロイして後で実行します。

データ統合サービスへのアプリケーションのデプロイ

アプリケーションを作成したら、データ統合サービスにそのアプリケーションをデプロイして、アプリケーションオブジェクトを実行します。

1. **【オブジェクトエクスプローラ】** ビューでアプリケーションを右クリックし、**【デプロイメント】** を選択します。

【デプロイメント】 ダイアログボックスが表示されます。

Deploy

Deployment Method

Choose how to deploy the application.

Application: ApplicationDemo

☒ Deploy to Service

Domain: caw177748 Browse...

Available Services:

<input checked="" type="checkbox"/>	Service	Target Name	Action
<input checked="" type="checkbox"/>	dsB480	ApplicationDemo	Add
<input type="checkbox"/>			
<input type="checkbox"/>			

☐ Export as application archive file

Location: Browse...

? < Back Next > Finish Cancel

2. **【サービスにデプロイ】** を選択します。
3. **【参照】** をクリックし、ドメインを選択します。
 - a. **【ドメインの選択】** ダイアログボックスでドメインを選択して、**【OK】** をクリックします。
 - b. アプリケーションをデプロイするデータ統合サービスを選択します。
4. **【完了】** をクリックします。

Developer tool によって、アプリケーションがデータ統合サービスにデプロイされます。データ統合サービスは、アプリケーションおよびアプリケーションに含まれるオブジェクトを実行します。

アプリケーションオブジェクトの出力をテストし、要件に対する出力を検証できます。必要に応じて、アプリケーションを更新して再デプロイすることができます。

データ統合サービスへのオブジェクトのデプロイ

データ統合サービスにオブジェクトをデプロイしてオブジェクトを実行できるようにする場合は、Developer tool から、オブジェクトをアプリケーションに含めるように求めるメッセージが表示されます。その後、データ統合サービスにアプリケーションをデプロイし、アプリケーションオブジェクトを実行できます。

1 つ以上のオブジェクトをデプロイする場合、ビジネスおよび機能上の要件に基づいてアプリケーションを作成します。また、デプロイしたアプリケーションを更新する場合、アプリケーションを作成することもできます。

1. 実行可能なオブジェクトまたはフラットファイルデータソースのデプロイを選択します。
 - マッピング、ワークフロー、または他の実行可能なオブジェクトをデプロイするには、オブジェクトを右クリックして **【デプロイメント】** を選択します。
 - フラットファイルデータソースをデプロイするには、オブジェクトを右クリックして **【Web サービスとしてデプロイします】** または **【SQL データサービスとしてデプロイします】** を選択します。
アプリケーションを作成するように求められます。
2. アプリケーション名を入力します。
3. アプリケーションにオブジェクトを追加するには、**【追加】** ボタンをクリックしてオブジェクトを選択します。
4. パラメータセットを設定したアプリケーションをデプロイするには、**【次へ】** をクリックします。
 - a. **【追加】** ボタンをクリックして、アプリケーションに含めるパラメータセットを選択します。
 - b. リストから含めるパラメータセットをそれぞれ選択します。すべてのパラメータセットを選択することができます。
5. **【次へ】** をクリックします。
6. アプリケーションのデプロイ先としてデータ統合サービスを選択します。
7. Developer tool に複数ドメインに関する接続情報が含まれている場合、**【参照】** をクリックしてドメインを選択し、次の手順を実行します。
 - a. **【ドメインの選択】** ダイアログボックスでドメインを選択して、**【OK】** をクリックします。
Developer ツールが、ドメインに関連付けられているデータ統合サービスを **【デプロイ】** ダイアログボックスの **【使用できるサービス】** セクションに一覧表示します。
 - b. アプリケーションをデプロイするデータ統合サービスを選択します。
 - c. **【次へ】** をクリックします。
8. Web サービスにオブジェクトをデプロイするには、以下の手順を実行します。
 - a. Web サービスのプロパティを設定します。
 - b. Web サービスに操作を追加するには、**【次へ】** をクリックします。
デフォルトでは、Web サービスとしてデプロイするオブジェクトごとに操作が作成されます。
 - c. プロパティを表示および設定する各操作、操作入力、および操作出力を選択します。
 - d. **【完了】** をクリックします。
9. SQL データサービスにオブジェクトをデプロイするには、以下の手順を実行します。
 - a. SQL データサービスの名前を入力します。
 - b. デフォルトの場所を受け入れるか、または **【参照】** をクリックし、SQL データサービスのモデルリポジトリおよびプロジェクトフォルダの場所を選択します。
 - c. **【次へ】** をクリックします。
【SQL データサービスに仮想テーブルを追加】 ダイアログボックスが表示されます。

- d. [追加] ボタンをクリックします。
- e. 仮想テーブルの名前を入力します。
- f. [データオブジェクト] カラムの [開く] ボタンをクリックします。
[データオブジェクトの選択] ダイアログボックスが表示されます。
- g. 物理データオブジェクトを選択し、[OK] をクリックします。
- h. [仮想スキーマ] カラムに仮想スキーマの名前を入力します。
- i. [データアクセス] カラムで [読み取り] を選択し、仮想テーブルとデータオブジェクトをリンクします。仮想テーブルとデータオブジェクトをリンクしない場合は、[なし] を選択します。

10. [完了] をクリックします。

Developer tool によって、アプリケーションがデータ統合サービスにデプロイされます。データ統合サービスは、アプリケーションおよびアプリケーションに含まれるオブジェクトを実行します。

アプリケーションオブジェクトの出力をテストし、要件に対する出力を検証できます。必要に応じて、アプリケーションを更新して再デプロイすることができます。

アーカイブファイルへのオブジェクトのデプロイ

アプリケーションアーカイブファイルにオブジェクトをエクスポートすることができます。ファイルを保存したり、データ統合サービスにデプロイするために管理者が使用できるようにしたりできます。

ファイルにオブジェクトをエクスポートする場合、有効なアプリケーションを作成し、そのアプリケーションにオブジェクトを追加する必要があります。次に、アプリケーションアーカイブファイルにアプリケーションをエクスポートします。

注: アプリケーションは、アプリケーションアーカイブファイルにエクスポートする前に有効にしておく必要があります。無効なアプリケーションをエクスポートするには、オブジェクトを右クリックして [エクスポート] を選択し、エクスポート形式を選択します。

1. 実行可能なオブジェクトまたはフラットファイルデータソースのエクスポートを選択します。
 - マッピング、ワークフロー、または他の実行可能なオブジェクトをエクスポートするには、オブジェクトを右クリックして [デプロイメント] を選択します。
 - フラットファイルデータソースをエクスポートするには、オブジェクトを右クリックして [Web サービスとしてデプロイします] または [SQL データサービスとしてデプロイします] を選択します。

アプリケーションを作成するように求められます。

2. アプリケーション名を入力します。
3. アプリケーションの場所を選択するには、デフォルトの場所を受け入れるか、[参照] をクリックして別の場所を選択します。
4. アプリケーションにオブジェクトを追加するには、[追加] ボタンをクリックしてオブジェクトを選択します。
5. パラメータセットを設定したアプリケーションをデプロイするには、以下の手順を実行します。
 - a. [追加] ボタンをクリックして、アプリケーションに含めるパラメータセットを選択します。
 - b. リストから含めるパラメータセットをそれぞれ選択します。すべてのパラメータセットを選択することができます。
6. [次へ] をクリックします。
7. [アプリケーションアーカイブファイルとしてエクスポート] を選択します。
8. [参照] をクリックし、ファイルの保存先となるディレクトリを選択します。
[ディレクトリの選択] ダイアログボックスが表示されます。

9. ディレクトリを選択して、**[OK]** をクリックします。
10. SQL データサービスの詳細を設定するには、以下の手順を実行します。
 - a. SQL データサービスの場所を選択するには、デフォルトの場所を受け入れるか、**[参照]** をクリックして別の場所を選択します。
 - b. SQL データサービスに仮想テーブルを追加するには、**[次へ]** をクリックします。
 - c. SQL データサービスに追加する仮想テーブルを設定するには、**[追加]** ボタンをクリックします。
11. Web サービスの詳細を設定するには、以下の手順を実行します。
 - a. Web サービスの名前を入力します。
 - b. モデルリポジトリのデフォルトの場所を受け入れるか、**[参照]** をクリックして別の場所を選択します。
 - c. 名前空間を入力します。
 - d. プレフィックスを入力します。
 - e. Web サービスに操作を追加するには、**[次へ]** をクリックします。
 - f. **[追加]** ボタンをクリックして新しい操作を作成します。
 - g. 各操作に対する入力と出力を設定します。

12. **[完了]** をクリックします。

Developer Tool はアプリケーションを検証し、アプリケーションアーカイブファイルにそのアプリケーションをエクスポートします。

注: アプリケーションが無効の場合は、デプロイプロセスが失敗します。

アプリケーションアーカイブファイルは別のシステムに保存することができます。例えば、バージョン管理システムにアーカイブできます。管理者は、`infacmd dis DeployApplication` コマンドを使用して、ファイル内のアプリケーションをデータ統合サービスにデプロイできます。ファイル内のアプリケーションをモデルリポジトリにインポートすることもできます。

アーカイブファイルへのアプリケーションのデプロイ

データ統合サービスにアプリケーションを直接デプロイしない場合は、アプリケーションアーカイブファイルにアプリケーションをデプロイします。ファイルを保存したり、別のシステムにインポートするために管理者が使用できるようにしたり、ファイルからデータ統合サービスにアプリケーションをデプロイしたりできます。

1. **[オブジェクトエクスプローラ]** でアプリケーションを右クリックし、**[デプロイメント]** を選択します。
2. **[ファイルシステムにデプロイ]** を選択します。
3. **[参照]** ボタンをクリックし、ファイルの保存先となるディレクトリを選択します。

[ディレクトリの選択] ダイアログボックスが表示されます。
4. ディレクトリを選択して、**[OK]** をクリックします。
5. **[完了]** をクリックします。

Developer tool はアプリケーションを検証し、アプリケーションアーカイブファイルにそのアプリケーションをエクスポートします。

アプリケーションアーカイブファイルは別のシステムに保存することができます。例えば、バージョン管理システムにアーカイブできます。管理者は、`infacmd dis DeployApplication` を使用して、ファイル内のアプリケーションをデータ統合サービスにデプロイできます。

アプリケーションアーカイブのインポート

アプリケーションアーカイブファイルにデプロイしたアプリケーションをインポートすることができます。アプリケーションと依存オブジェクトをモデルリポジトリにインポートします。アーカイブファイルを異なるモデルリポジトリからアーカイブした場合、またはバージョン管理システムから取得した場合は、この操作を実行することができます。

1. **【ファイル】 > 【インポート】** をクリックします。
インポートウィザードが表示されます。
2. **【Informatica】 > 【アプリケーションアーカイブ】** を選択します。
3. **【次へ】** をクリックします。
4. **【参照】** をクリックし、アプリケーションアーカイブファイルを選択します。
アプリケーションアーカイブファイルの内容が一覧表示されます。
5. アプリケーションのインポート先のモデルリポジトリを選択します。
6. **【完了】** をクリックします。
アプリケーションがリポジトリにインポートされます。

アプリケーションの更新

アプリケーションのオブジェクトを更新する場合は、アプリケーションを更新して再デプロイし、エンドユーザーが変更内容にアクセスできるようにする必要があります。アプリケーションを更新する場合、オブジェクトを追加または削除し、プロパティを編集することもできます。

アプリケーションを更新して再デプロイする前に、データ統合サービス上でアプリケーションを停止します。

1. アプリケーションオブジェクトを編集するには、**【オブジェクトエクスプローラ】** からオブジェクトを開きます。
2. **【オブジェクトエクスプローラ】** でアプリケーションを右クリックし、**【開く】** を選択します。
【アプリケーションエディタ】 が開きます。
3. **【追加】** ボタンをクリックし、アプリケーションにオブジェクトを追加します。
【オブジェクトの追加】 ダイアログボックスが表示されます。
4. 1つ以上のオブジェクトを選択し、**【OK】** をクリックします。
5. 削除するオブジェクトを1つ以上選択し、**【削除】** ボタンをクリックします。
6. アプリケーションを保存します。

アプリケーションを更新したら、データ統合サービスにアプリケーションを再デプロイします。

データ統合サービスへのアプリケーションの再デプロイ

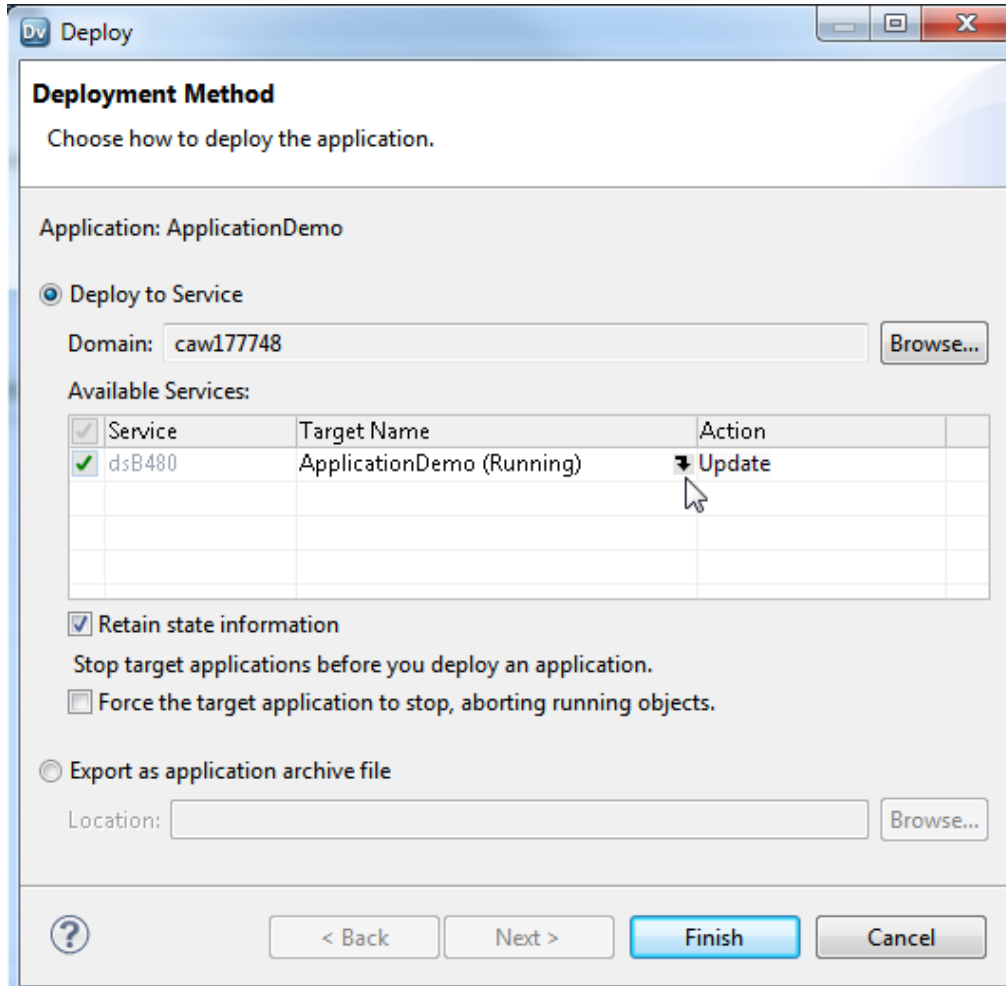
アプリケーションに含まれるオブジェクトを更新したら、データ統合サービスにアプリケーションを再デプロイします。

1. **【オブジェクトエクスプローラ】** ビューでアプリケーションを右クリックし、**【デプロイ】** をクリックします。
【デプロイメント】 ダイアログボックスが表示されます。
2. **【サービスにデプロイ】** を選択します。
3. デフォルトのデータ統合サービスが設定されていない場合、**【参照】** をクリックしてドメインを選択します。

【ドメインの選択】ダイアログボックスが表示されます。

4. ドメインを選択して【OK】をクリックし、データ統合サービスを選択します。

【ターゲット名】 カラムには、同名のアプリケーションがデフォルトで表示されます。例えば、以下の図に、アプリケーション ApplicationDemo のデプロイメントを示します。



5. 【ターゲット名】 カラムをクリックし、データ統合サービス上で異なるターゲットアプリケーションを選択します。
6. アプリケーションの一部である実行中オブジェクトの状態を保持するには、【状態情報を保持する】を選択します。
7. デプロイしたオブジェクトを置き換えるには、【ターゲットアプリケーションを停止して、実行中のオブジェクトを強制終了する】を選択します。
8. 【完了】をクリックします。

アプリケーションを再デプロイしたら、要件と比較して検証することができます。

第 10 章

オブジェクトのインポートとエクスポート

この章では、以下の項目について説明します。

- [オブジェクトのインポートとエクスポートの概要, 154 ページ](#)
- [オブジェクトのインポートとエクスポート, 155 ページ](#)
- [オブジェクトエクスポート, 156 ページ](#)
- [オブジェクトインポート, 157 ページ](#)

オブジェクトのインポートとエクスポートの概要

複数のオブジェクトをプロジェクトから 1 つの XML ファイルにエクスポートできます。オブジェクトをインポートするときに、XML ファイルで個々のオブジェクトまたはすべてのオブジェクトを選択できます。

オブジェクトを XML ファイルにエクスポートしたり、その XML ファイルからオブジェクトをインポートしたりできます。オブジェクトをエクスポートすると、そのオブジェクトのメタデータが格納された XML ファイルが作成されます。この XML ファイルを使用して、オブジェクトをプロジェクトまたはフォルダーにインポートします。オブジェクトのインポートとエクスポートは、`infacmd` コマンドを使用して行うこともできます。

オブジェクトのエクスポートとインポートで、次の作業を行えます。

メタデータのプロダクション環境へのデプロイメント

開発環境のリポジトリでマッピングのテストを終えた後で、そのマッピングを XML ファイルにエクスポートし、そのファイルからプロダクション環境のリポジトリへマッピングをインポートできます。

メタデータのアーカイブ

不要になったオブジェクトをリポジトリから削除する前に、XML ファイルへエクスポートできます。

メタデータの共有

第三者とメタデータを共有できます。例えば、テストまたは分析のためにマッピングを他の人に送信できます。

リポジトリ間でのメタデータのコピー

同じクライアントからでは接続できないリポジトリの間でオブジェクトをコピーできます。このような場合には、オブジェクトを XML ファイルにエクスポートし、その XML ファイルをコピー先のマシンに転送します。そして、XML ファイルからコピー先のリポジトリにオブジェクトをインポートします。同じバー

ジョンのリポジトリ間でオブジェクトをエクスポートおよびインポートできます。オブジェクトにタグが含まれている場合は、Developer ツールによって自動的にリポジトリにインポートされます。

infacmd を使用して、読み取り可能な XML ファイルをエクスポートファイルから生成できます。オブジェクトをリポジトリにインポートする前に、読み取り可能な XML でオブジェクト名を編集したりエクスポート XML を更新したりすることもできます。

オブジェクトのインポートとエクスポート

プロジェクトやプロジェクト内のオブジェクトをインポートおよびエクスポートできます。リポジトリ内のアプリケーションアーカイブファイルをインポートおよびエクスポートすることもできます。

オブジェクトをエクスポートすると、依存オブジェクトもエクスポートされます。依存オブジェクトは、別のオブジェクトによって使用されるオブジェクトです。例えば、マッピング入力として使用される物理データオブジェクトは、そのマッピングの依存オブジェクトです。オブジェクトをインポートすると、すべての依存オブジェクトがインポートされます。

プロジェクトまたはフォルダー内のオブジェクトをエクスポートまたはインポートするときは、モデルリポジトリサービスによってオブジェクト階層が保持されます。

以下の表に、エクスポートできるオブジェクトと依存オブジェクトを示します。

オブジェクト	依存関係
アプリケーション	- SQL データサービス、マッピング、またはワークフローおよびその依存オブジェクト
プロジェクト	- プロジェクトには他のオブジェクトは含まれますが、依存オブジェクトは含まれません
フォルダー	- フォルダーには他のオブジェクトは含まれますが、依存オブジェクトは含まれません
参照テーブル	- 参照テーブルには依存オブジェクトは含まれません
コンテンツセット	- コンテンツセットには依存オブジェクトは含まれません
物理データオブジェクト (カスタマイズデータオブジェクト以外)	- 物理データオブジェクトには依存オブジェクトは含まれません
カスタマイズデータオブジェクト	- 物理データオブジェクト
論理データオブジェクト モデル	- 論理データオブジェクト - 物理データオブジェクト - 再利用可能なトランスフォーメーションとその依存オブジェクト - マップレットとその依存オブジェクト
トランスフォーメーション	- 物理データオブジェクト - 参照テーブル - コンテンツセット

オブジェクト	依存関係
マップレット	<ul style="list-style-type: none"> - 論理データオブジェクト - 物理データオブジェクト - 再利用可能なトランスフォーメーションとその依存オブジェクト - マップレットとその依存オブジェクト
マッピング	<ul style="list-style-type: none"> - 論理データオブジェクト - 物理データオブジェクト - 再利用可能なトランスフォーメーションとその依存オブジェクト - マップレットとその依存オブジェクト
SQL データサービス	<ul style="list-style-type: none"> - 論理データオブジェクト - 物理データオブジェクト - 再利用可能なトランスフォーメーションとその依存オブジェクト - マップレットとその依存オブジェクト
プロファイル	<ul style="list-style-type: none"> - 論理データオブジェクト - 物理データオブジェクト
スコアカード	<ul style="list-style-type: none"> - プロファイルとその依存オブジェクト
Web サービス	<ul style="list-style-type: none"> - 操作マッピング
ワークフロー	<ul style="list-style-type: none"> - マッピングとその依存オブジェクト

オブジェクトエクスポート

オブジェクトをエクスポートすると、そのオブジェクトのメタデータが格納された XML ファイルが作成されます。

エクスポートするオブジェクトを選択できます。依存オブジェクトもすべてエクスポートするように選択する必要があります。Developer ツールによってオブジェクトと依存オブジェクトがエクスポートされます。エクスポートされるのは最後に保存されたバージョンのオブジェクトです。Developer ツールは、XML ファイルの要素に Cyclic Redundancy Checking Value (CRCVALUE) コードを付加します。CRCVALUE コードが含まれる要素の属性を変更すると、オブジェクトをインポートできなくなります。属性を変更する場合は、infacmd xrf コマンドを使用します。

オブジェクトのエクスポートは、infacmd oie ExportObjects コマンドを使用して行うこともできます。

オブジェクトのエクスポート

別のリポジトリのモデルリポジトリオブジェクトを使用するために、オブジェクトを XML メタデータファイルとしてエクスポートできます。

1. **【ファイル】 > 【エクスポート】** をクリックします。
【エクスポート】 ウィザードが開きます。
2. **【Informatica】 > 【オブジェクトメタデータファイルのエクスポート】** を選択します。

- [次へ] をクリックします。
3. [参照] をクリックします。エクスポートするオブジェクトを含むリポジトリプロジェクトを選択します。
[次へ] をクリックします。
 4. エクスポートするオブジェクトを1つ以上選択してください。エクスポートプロセスを開始する前にリポジトリオブジェクトをハイライトした場合は、ウィザードがオブジェクトを選択します。
 5. ファイル名とXMLメタデータファイルの場所を入力します。Developer ツールは、選択したオブジェクトをすべて1つのファイルにエクスポートします。
[次へ] をクリックします。
 6. ウィザードはメタデータオブジェクトが使用する依存オブジェクトを表示します。
[次へ] をクリックし、依存オブジェクトを受け入れます。
 7. 選択したオブジェクトに参照データオブジェクトが含まれる場合は、[コンテンツのエクスポート] オプションを選択し、エクスポートの設定を確認します。
 - エクスポートする参照データファイルの名前と場所を確認します。Developer ツールサービスは参照データファイルを1つのZIPファイルにエクスポートします。デフォルトの場合、ウィザードはZIPファイルとXMLメタデータファイルを同じディレクトリにエクスポートします。
 - 参照データが使用するコードページを確認します。デフォルトのコードページはUTF-8です。参照テーブルデータをエクスポートする場合は、デフォルトのコードページを受け入れます。
 - エクスポートする確率的なモデルデータを確認します。デフォルトの場合、ウィザードはすべてのモデルデータをエクスポートします。選択したオブジェクトが確率的なモデルを含まない場合、エクスポートプロセスはオプションを無視します。
 8. [完了] をクリックし、選択したオブジェクトをエクスポートします。
Developer ツールはオブジェクトメタデータをXMLファイルにエクスポートし、任意の依存参照データファイルをZIPファイルにエクスポートします。

オブジェクトインポート

プロジェクトまたはプロジェクト内のオブジェクトをエクスポートファイルからインポートできます。オブジェクトと依存オブジェクトをプロジェクトまたはフォルダにインポートできます。

プロジェクトまたは個別のオブジェクトをインポートできます。プロジェクト内のすべてのオブジェクトを再利用する場合は、プロジェクトをインポートします。複数のプロジェクトにまたがってオブジェクトを再利用する場合は、個々のオブジェクトをインポートします。

オブジェクトをインポートするときに、すべての依存オブジェクトが一覧表示されます。オブジェクトをインポートする前に、各依存オブジェクトをターゲットに追加する必要があります。

オブジェクトをインポートするときに、エクスポートファイル内のオブジェクトの名前がターゲットプロジェクトまたはフォルダ内のオブジェクトと同じである場合があります。名前の競合を解決する方法を選択できます。

オブジェクトのインポートは、`infacmd oie Import0bjects` コマンドを使用して行うこともできます。

以前のバージョンで作成したエクスポートファイルからオブジェクトをインポートすることはできません。

プロジェクトのインポート

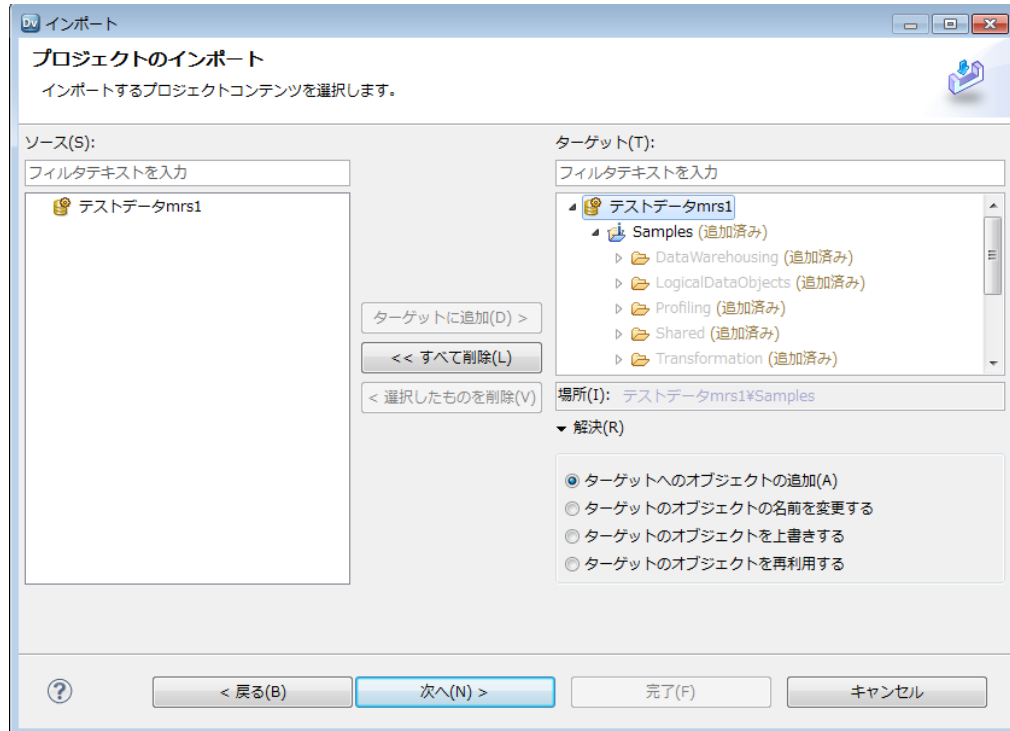
XML ファイルからターゲットリポジトリにプロジェクトをインポートできます。また、プロジェクトのコンテンツをターゲットリポジトリ内のプロジェクトにインポートすることもできます。

1. **【ファイル】** > **【インポート】** をクリックします。
2. **【Informatica】** > **【オブジェクトメタデータファイルのインポート（基本）】** を選択します。
3. **【次へ】** をクリックします。
4. **【参照】** をクリックして、インポートするエクスポートファイルを選択します。
5. **【次へ】** をクリックします。
6. **【ソース】** ペインでプロジェクトまたは **[<project name>プロジェクトコンテンツ]** を選択します。
 - **【ソース】** ペインでプロジェクトを選択した場合は、**【ターゲット】** ペインでプロジェクトをインポートするモデルリポジトリサービスを選択します。
 - **【ソース】** ペインでプロジェクトコンテンツを選択した場合は、**【ターゲット】** ペインでプロジェクトコンテンツをインポートするプロジェクトを選択します。
7. **【ターゲットに追加】** をクリックして、プロジェクトをターゲットに追加します。

ヒント: プロジェクトを **【ソース】** ペインから **【ターゲット】** ペインのリポジトリにドラッグすることもできます。また、**【ソース】** ペインのプロジェクトコンテンツを **【ターゲット】** ペインのプロジェクトにドラッグすることもできます。

8. **【解決】** をクリックして、重複するオブジェクトの処理方法を指定します。

インポートするオブジェクトの名前を変更するか、インポートするオブジェクトで既存のオブジェクトを上書きするか、既存のオブジェクトを再利用することができます。デフォルトでは、重複するすべてのオブジェクトの名前が変更されます。



9. **【次へ】** をクリックします。
インポート中の参照テーブルデータが一覧表示されます。追加の参照テーブル設定を指定します。

10. **【次へ】** をクリックします。

インポート対象オブジェクトの概要が表示されます。**【ソースとターゲットのオブジェクトのリンク】** をクリックし、ソースとターゲットのどちらかのオブジェクトを選択すると、**【ソース】** および **【ターゲット】** の表示ペインにあるオブジェクトがリンクされます。例えば、このオプションを選択してから **【ソース】** ペインでオブジェクトを選択すると、**【ターゲット】** ペインにある同じオブジェクトが選択されます。

11. **【追加のインポート設定】** ペインで、インポートファイルからターゲットドメインへの接続をマッピングします。オブジェクトの既存のタグを上書きするかどうかを選択することもできます。
12. **【完了】** をクリックします。

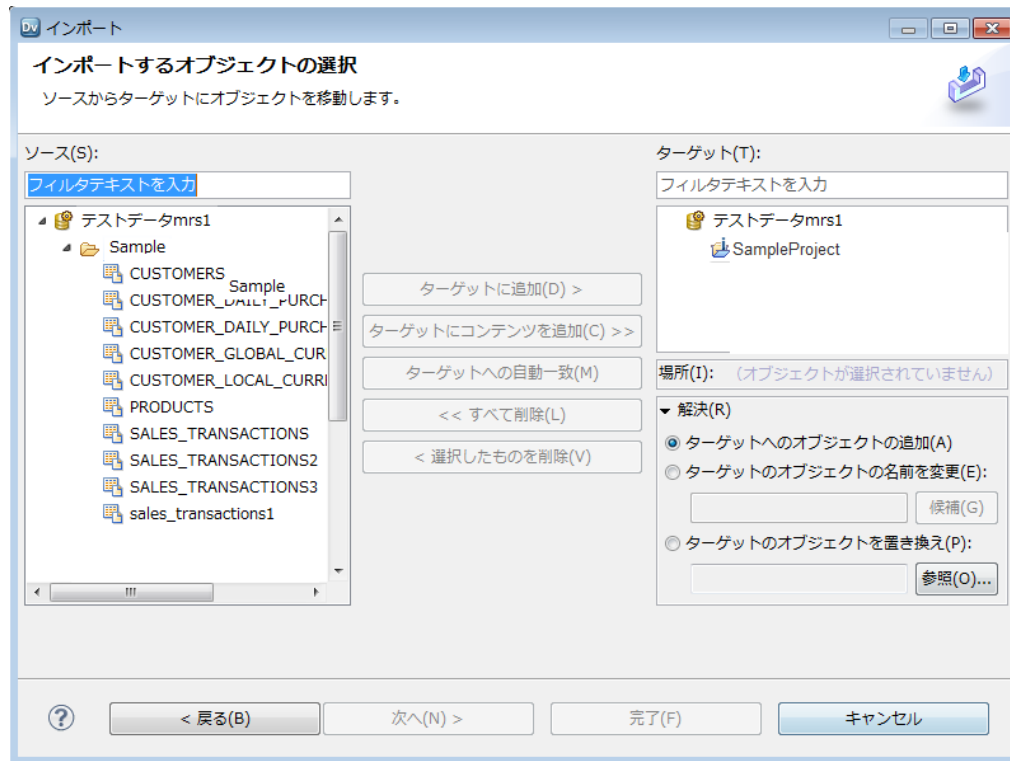
重複するプロジェクトの名前を変更した場合は、モデルリポジトリサービスによってオブジェクト名に番号が付加されます。プロジェクトの名前はインポート後に変更できます。

オブジェクトのインポート

XML ファイルまたはアプリケーションアーカイブファイルからオブジェクトをインポートできます。オブジェクトと依存オブジェクトをプロジェクトにインポートします。

1. **【ファイル】** > **【インポート】** をクリックします。
2. **【Informatica】** > **【オブジェクトメタデータファイルのインポート（詳細）】** を選択します。
3. **【次へ】** をクリックします。
4. **【参照】** をクリックして、インポートするエクスポートファイルを選択します。
5. **【次へ】** をクリックします。
6. **【ソース】** ペインで、インポートするオブジェクトを選択します。
7. **【ターゲット】** ペインで、オブジェクトのインポート先のプロジェクトを選択します。

8. **【ターゲットに追加】** をクリックして、オブジェクトをターゲットに追加します。



【ターゲットへの自動一致】 をクリックすると、現在のソース選択の下位オブジェクトがターゲット選択の名前、タイプ、親階層を基準にして個別に照合され、一致するオブジェクトが追加されます。

フォルダーまたはプロジェクトのすべてのオブジェクトをインポートする場合は、インポート先のフォルダーまたはプロジェクトを選択して **【ターゲットにコンテンツを追加】** をクリックします。

ヒント: オブジェクトを [ソース] ペインから [ターゲット] ペインの必要なプロジェクトにドラッグすることもできます。ソースとターゲットでオブジェクト階層を維持するには、Ctrl キーを押しながらドラッグします。

9. クリックして、重複するオブジェクトの処理方法を指定します。

インポートするオブジェクトの名前を変更するか、インポートするオブジェクトで既存のオブジェクトを上書きするか、既存のオブジェクトを再利用することができます。デフォルトでは、重複するすべてのオブジェクトの名前が変更されます。

10. **【次へ】** をクリックします。

インポートファイルの依存オブジェクトが一覧表示されます。

11. 依存オブジェクトをターゲットフォルダーまたはプロジェクトに追加します。

12. **【次へ】** をクリックします。

インポート中の参照テーブルデータが一覧表示されます。追加の参照テーブル設定を指定します。

13. **【次へ】** をクリックします。

インポート対象オブジェクトの概要が表示されます。**【ソースとターゲットのオブジェクトのリンク】** をクリックし、ソースおよびターゲットのどちらかのオブジェクトを選択すると、[ソース] および [ターゲット] の表示ペインにあるオブジェクトがリンクされます。例えば、このオプションを選択してから [ソース] ペインでオブジェクトを選択すると、[ターゲット] ペインにある同じオブジェクトが選択されます。

14. **【追加のインポート設定】** ペインで、インポートファイルからターゲットドメインへの接続をマッピングします。オブジェクトの既存のタグを上書きするかどうかを選択することもできます。

15. **【完了】** をクリックします。

重複するプロジェクトの名前を変更するように選択した場合は、**インポートウィザード**によって、インポートするプロジェクトに「<Original Name>_<number of the copy>」という名前が付けられます。プロジェクトの名前はインポート後に変更できます。

付録 A

データ型リファレンス

この付録では、以下の項目について説明します。

- [データ型リファレンスの概要, 162 ページ](#)
- [トランスフォーメーションのデータ型, 163 ページ](#)
- [DB2 for i5/OS および DB2 for z/OS のデータ型とトランスフォーメーションのデータ型, 170 ページ](#)
- [フラットファイルとトランスフォーメーションのデータ型, 172 ページ](#)
- [DB2 for LUW とトランスフォーメーションデータ型, 172 ページ](#)
- [JDBC のデータ型とトランスフォーメーションのデータ型, 174 ページ](#)
- [Microsoft SQL Server のデータ型とトランスフォーメーションデータ型, 175 ページ](#)
- [非リレーショナルのデータ型とトランスフォーメーションのデータ型, 178 ページ](#)
- [ODBC データ型とトランスフォーメーションデータ型, 180 ページ](#)
- [Oracle のデータ型およびトランスフォーメーションデータ型, 181 ページ](#)
- [SAP HANA およびトランスフォーメーションデータ型, 184 ページ](#)
- [XML データ型とトランスフォーメーションデータ型, 186 ページ](#)
- [データの変換, 188 ページ](#)

データ型リファレンスの概要

マッピングを作成する場合は、ソースからのデータ読み込み、変換、ターゲットへの書き込みを Data Integration Service に実行させるための指示セットを作成します。Data Integration Service は、マッピング内の最初のトランスフォーメーションから始まるマッピング内のデータフロー、およびマッピング内の各ポートに割り当てられているデータ型に基づいて、データを変換します。

Developer ツールは、以下の 2 種類のデータ型を扱います。

- **ネイティブデータ型。**リレーショナルテーブルまたはフラットファイル固有の型で、物理データオブジェクトとして使用されます。ネイティブデータ型は、物理データオブジェクトカラムのプロパティに表示されます。
- **トランスフォーメーションデータ型。**トランスフォーメーションで扱われるデータ型のセットです。これは ANSI SQL-92 汎用データタイプに基づく内部データタイプで、Data Integration Service がプラットフォーム間でデータを移動するために使用します。トランスフォーメーションデータタイプはマッピングのすべてのトランスフォーメーション内に表示されます。

Data Integration Service は、ソースデータからの読み込み時に、ネイティブデータ型を同等のトランスフォーメーションデータ型に変換してからデータのトランスフォームを実行します。Data Integration Service は、

ターゲットへの書き込み時に、トランスフォーメーションデータタイプを同等のネイティブデータ型に変換します。

マルチバイト文字セットを指定した場合、データタイプは3バイトまで文字を格納する追加スペースをデータベース内に割り当てます。

トランスフォーメーションのデータ型

以下の表に、トランスフォーメーションのデータ型を示します。

データ型	サイズ (バイト数)	説明
Bigint	8 バイト	-9,223,372,036,854,775,808～ 9,223,372,036,854,775,807 精度 19、位取り 0。 整数値。
Binary	精度	1～104,857,600 バイト フラットファイルソースでバイナリデータは使用できません。
Date/Time	16 バイト	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 精度 29、位取り 9 (精度はナノ秒まで) 日付値と時刻値が結合されたデータ型。
Decimal	8 バイト (高精度がオフまたは 38 より大きい場合) 16 バイト (精度が 18 以下で高精度がオンの場合) 20 バイト (精度が 18 より大きく、28 以下の場合) 24 バイト (精度が 28 より大きく、38 以下の場合)	宣言された精度と位取りを持つ 10 進型の値。位取りは、精度以下にする必要があります。 最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 最大 28 桁の精度をサポートするトランスフォーメーションでは、精度が 1～28 桁、スケールが 0～28 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
Double	8 バイト	精度 15 倍精度浮動小数点数型の値。
Integer	4 バイト	-2,147,483,648～2,147,483,647 精度 10、位取り 0 整数値。
String	Unicode モード: (精度+1) * 2 ASCII モード: 精度+1	1～104,857,600 文字 固定長または可変長文字列。

データ型	サイズ (バイト数)	説明
Text	Unicode モード: (精度+1) * 2 ASCII モード: 精度+1	1~104,857,600 文字 固定長または可変長文字列。
timestampWithTZ	40 バイト	西暦 1947 年 8 月 1 日~西暦 2040 年 12 月 31 日 -12:00~+14:00 精度 36、スケール 9。 (精度はナノ秒まで) Timestamp with Time Zone データ型では、次のタイムゾーン地域はサポートされません。 <ul style="list-style-type: none"> - AFRICA_CAIRO - AFRICA_MONROVIA - エジプト - AMERICA_MONTREAL

Integer データタイプ

ソースからターゲットに整数データを渡して、整数データでトランスフォーメーションを実行できます。トランスフォーメーション言語は、Bigint および Integer データタイプをサポートします。

トランスフォーメーションの Integer データタイプは、厳密値を表します。

計算における整数値

計算に整数値を使用する場合、データ統合サービスでは、計算の前に整数値が浮動小数点数に変換されることがあります。例えば、MOD(12.00, 5)を評価する場合、データ統合サービスは、除算を実行する前に整数値「5」を浮動小数点数に変換します。データ統合サービスでは、高精度が有効にされているかどうかに応じて、整数値が Double 値または Decimal 値に変換されます。

データ統合サービスは、次の算術演算の整数値を変換します。

算術演算	高精度の無効	高精度の有効
小数点を使用できない関数と計算。 たとえば、整数の加算、減算、乗算、および CUME、MOVINGSUM、SUM などの関数。	変換なし ¹	Decimal
小数点を使用できる非科学関数と計算。 たとえば、整数の除算、および AVG、MEDIAN、PERCENTILE などの関数。	Double	Decimal
すべての科学関数、EXP 関数、LN 関数、LOG 関数、POWER 関数、および SQRT 関数。	Double	ダブル

¹ 計算が範囲外の結果を算出した場合、統合サービスは行エラーを書き込みます。

トランスフォーメーションの Double データ型は最大 15 桁の精度をサポートし、Bigint データ型は最大 19 桁の精度をサポートします。このため、精度が 15 桁より大きい Bigint 値となる計算では、精度の損失が発生することがあります。

例えば、式トランスフォーメーションには以下の計算が含まれています。

POWER(BIGINTVAL, EXPVAL)

データ統合サービスでは、計算を開始する前に、POWER 関数への入力を Double 値に変換します。BIGINTVAL ポートに Bigint 値 9223372036854775807 が含まれている場合、データ統合サービスはこの値を 9.22337203685478e+18 に変換し、最後の 4 桁の精度が失われます。EXPVAL ポートに値 1.0 が含まれており、結果ポートが Bigint の場合、計算結果の 9223372036854780000 が最大 Bigint 値を超えているため、行エラーが作成されます。

結果が Decimal 値となる計算で Integer データ型を使用し、高精度を有効にする場合、データ統合サービスでは、整数値が Decimal 値に変換されます。

最大 28 桁精度の Decimal データ型をサポートするトランスフォーメーションの場合、結果で高精度モードの 28 桁を超える値が出ない限り計算の精度は失われません。この場合、データ統合サービスは結果を Double 型として保存します。ポート精度が 28 桁以下で、結果が高精度モードの 28 桁より大きい場合、データ統合サービスは行を拒否します。

最大 38 桁精度の Decimal データ型をサポートするトランスフォーメーションの場合、結果で高精度モードの 38 桁を超える値が出ない限り計算の精度は失われません。この場合、データ統合サービスは結果を Double 型として保存します。ポート精度が 38 桁以下で、結果が高精度モードの 38 桁より大きい場合、データ統合サービスは行を拒否します。

式の整数定数

Integration Service では、計算結果が整数の場合でも、式の定数を浮動小数点数として解釈します。例えば、式 INTVALUE+1000 では、高精度が有効でない場合、Integration Service は整数値「1000」を Double 値に変換します。高精度が有効な場合は、値「1000」を 10 進数値に変換します。値 1000 を整数値として処理するには、Integer データタイプを使用して変数ポートを作成して定数を保持し、式を変更して 2 つのポートを追加します。

NaN 値

NaN（非数）とは、通常、特に浮動小数点数の計算で、無効な入力オペランドでの演算の結果として返される値のことです。例えば、ゼロをゼロで除算する演算の場合、結果として NaN が返されます。

NaN の表示は、オペレーティングシステムおよびプログラミング言語によって異なります。例えば、以下のリストのような NaN の表示があります。

```
nan
NaN
NaN%
NAN
NaNQ
NaNS
qNaN
sNaN
1.#SNAN
1.#QNAN
```

Integration Service は、Win64EMT プラットフォームでは QNAN 値を 1.#QNAN に変換します。1.#QNAN は、NaN の有効な表示です。

フラットファイルへの整数値の書き込み

整数値を固定長のフラットファイルに書き込む場合、ファイルライターではデータが範囲内であることが確認されません。例えば、ターゲットカラムのフィールド長が少なくとも 13 の場合、ファイルライターによりターゲットの Integer カラムに結果 3,000,000,000 が書き込まれます。ファイル writer では、結果が整数値の有効範囲外であるため、行は拒否されません。

バイナリデータタイプ

マッピングにバイナリデータが含まれている場合、Integration Service がデータをソースからターゲットに移動するために必要なメモリを割り当てることができるように、トランスフォーメーションのバイナリデータタイプの精度を設定します。

フラットファイルソースでバイナリデータタイプは使用できません。

日付/時刻データ型

日付/時刻データ型は、グレゴリオ暦の西暦 1 年から西暦 9999 年までの年を扱います。西暦 9999 より先の年はエラーになります。

日付/時刻データ型は、ナノ秒までの精度をサポートします。データ型の精度は 29 で、位取りは 9 です。一部のネイティブデータ型には精度が小さいものがありますが、日付の値を含むソースをインポートする際、インポートプロセスはソースカラムから正しい精度をインポートします。例えば、Microsoft SQL Server の Datetime データ型の精度は 23 で、位取りは 3 です。日時の値を含む Microsoft SQL Server ソースをインポートする場合、マッピングソースの日付カラムは精度が 23 で、位取りが 3 になります。

Integration Service は、マッピングソースに指定されている精度で、ソースから日時の値を読み込みます。Integration Service は、日時の値を変換する際、29 桁までの精度をサポートします。例えば、ミリ秒の精度で日時の値をインポートする場合、式トランスフォーメーションの ADD_TO_DATE 関数を使用して日付にナノ秒を追加することができます。

より短い精度をサポートするターゲットカラムに日付/時刻値を書き込むと、Integration Service はターゲットカラムの精度まで値を切り詰めます。より長い精度をサポートするターゲットカラムに日付/時刻値を書き込むと、Integration Service は日時の値でサポートされていない部分にゼロを挿入します。

Timestamp with Time Zone

Timestamp with Time Zone は、タイムゾーンのオフセット (TIME_ZONE_HOUR: TIME_ZONE_MINUTE) が含まれる Timestamp データ型のバリエーションです。夏時間やタイムゾーンの地域名がこれに含まれる場合があります。タイムゾーンのオフセットは、ローカルタイムゾーンと UTC (Coordinated Universal Time: 協定世界時) の差 (時分単位) です。

例えば、「16-JUN-08 07.01.25.376000 PM -06:00」のようなタイムスタンプがあります。この例では、「-06:00」がタイムゾーンのオフセットです。

また、「05-JUN-2008 07:01:25.376000 PM America/Los_Angeles」のようなタイムスタンプがあります。この例では、「America/Los_Angeles」がタイムゾーンの地域名です。

Timestamp with Time Zone データ型を Developer tool にインポートすると、関連付けられたトランスフォーメーションデータ型は timestampWithTZ になります。

timestampWithTZ の精度は 36 で、スケールは 9 です。Timestamp with Time Zone の置換値の範囲は、-12:00 < UTC < +14:00 です。

データ統合サービスは、Timestamp with Time Zone データ型を Oracle およびフラットファイルデータオブジェクト内で処理できます。Timestamp with Time Zone データ型は、1947 年 8 月～2040 年 12 月の日付にのみ適用されます。

Timestamp with Time Zone データ型は、Developer tool 内では、以下のタイムゾーン地域をサポートしません。

- AFRICA_CAIRO
- AFRICA_MONROVIA
- エジプト

- AMERICA_MONTREAL

Timestamp with Time Zone データ型は、タイムゾーンデータベースバージョン 2015b の IANA 標準を使用します。Timestamp with Time Zone データ型を使用する場合は、データの破損を防ぐため、Oracle サーバーとクライアントに Oracle DST パッチ 24 が適用されていることを確認します。データオブジェクトのキャッシュを有効にする場合、データオブジェクトのキャッシュで使用するデータベースサーバーに Oracle DST パッチ 24 が適用されていることを確認します。

パッチが適用されていない場合は、Oracle Web サイトから Oracle DST パッチ 24 をダウンロードしてください。Oracle サーバーで Oracle DST パッチ 24 が使用されていることを確認するには、Oracle サーバーのコマンドプロンプトで以下のコマンドを実行します。

```
SELECT VERSION FROM v$timezone_file;
```

バージョンが 24 として表示されることを確認できます。Oracle サーバーで以下のコマンドを実行してもバージョンを確認できます。

```
SELECT TZ_VERSION FROM registry$database;
```

TZ_VERSION が 24 として表示されることを確認できます。

Timestamp with Local Time Zone

Timestamp with Local Time Zone は、タイムスタンプデータがデータベースタイムゾーンに正規化される場合の Timestamp データ型のバリエーションです。タイムゾーンの置換はカラムデータの一部ではありません。データ統合サービスがデータを読み取る場合、Oracle はデータ統合サービスのタイムゾーンのデータを返します。

Timestamp with Local Time Zone データ型を Developer tool にインポートする場合、関連付けられるトランスフォーメーションのデータ型は日付/時刻です。

例えば、「04-APR-10 10.27.451 AM」などです。

Timestamp (9) with Local Time Zone の精度は 29、スケールは 9 です。これは、日付/時刻 (29,9) トランスフォーメーションデータ型にマッピングされます。

データ統合サービスが Timestamp with Local Time Zone データを読み取る、または書き込む場合にデフォルトのセッションタイムゾーンを設定するには、ORA_SDTZ 環境変数を指定します。ORA_SDTZ 環境変数を次の値に設定できます。

- オペレーティングシステムローカルタイムゾーン (「OS_TZ」)
- データベースタイムゾーン (「DB_TZ」)
- UTC からの絶対オフセット (例: 「-05:00」)
- タイムゾーン地域名 (例: 「America/Los_Angeles」)

Oracle でサポートされているタイムゾーン

次のインストール場所で、タイムゾーンファイルに Oracle でサポートされているすべてのタイムゾーン (AFRICA_CAIRO、AFRICA_MONROVIA、EGYPT、および AMERICA_MONTREAL の 4 つのタイムゾーンを除く) のリストが格納されていることを確認できます。

```
<Informatica インストールディレクトリ>/services/shared/timezones/timezones.txt
```

後に Oracle でタイムゾーンが追加サポートされたときに、それに準じてタイムゾーンを追加する場合は、新しいタイムゾーンを含む新しいファイルを同じインストール場所に配置します。

10 進型と倍精度浮動小数点数型のデータ型

10 進型データおよび倍精度浮動小数点数型データをソースからターゲットに渡して、10 進型データおよび倍精度浮動小数点数型データに対するトランスフォーメーションを実行することができます。

トランスフォーメーション言語では以下のデータ型をサポートしています。

Decimal

最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1~38 桁、スケールが 0~38 になります。最大 28 桁の精度をサポートするトランスフォーメーションでは、精度が 1~28 桁、スケールが 0~28 になります。精度より大きなスケールや負の精度を持つ 10 進型の値は使用できません。トランスフォーメーションには 10 進型のデータ型に割り当てた任意の範囲が表示されますが、データ統合サービスでは、トランスフォーメーションに応じて、最大 38 桁または 28 桁の精度のみサポートします。

高精度を有効にし、ポート精度が 38 桁または 28 桁（トランスフォーメーションに依存）より高い場合、データ統合サービスは結果を Double 型として保存します。

Double

精度 15。

計算における 10 進型の値と倍精度浮動小数点数型の値

計算結果が最大精度を超える値になる場合、10 進型と倍精度浮動小数点数型のデータ型を計算に使用しても、精度の損失が発生する可能性があります。

高精度を無効にすると、データ統合サービスは 10 進値を倍精度浮動小数点数に変換します。10 進値が 15 桁を超える精度を持つ場合には、精度の損失が発生します。例えば、Decimal (20,0) を使って数値 40012030304957666903 を渡すマッピングがあるとします。高精度を無効にすると、データ統合サービスは 10 進値を倍精度浮動小数点数に変換し、 $4.00120303049577 \times 10^{19}$ を渡します。

最大 38 桁の精度の Decimal データ型をサポートするトランスフォーメーションでは、Decimal データ型を使用し、高精度を有効にして、最大 38 桁の精度を保証します。

最大 28 桁の精度の Decimal データ型をサポートするトランスフォーメーションでは、Decimal データ型を使用し、高精度を有効にして、最大 28 桁の精度を保証します。

計算結果が、最大許容桁より大きい精度を持つ値にならない限り、精度の損失は発生しません。この場合、データ統合サービスは結果を倍精度浮動小数点数として保存します。

ルックアップ条件や結合条件など、等価条件で使用するデータに倍精度浮動小数点数型のデータ型を使用しないでください。

以下の表に、データ統合サービスで高精度設定に基づいて 10 進値がどのように処理されるかを示します。

ポートのデータ型	精度	高精度オフ	高精度オン
Decimal	0～15	Decimal	Decimal
Decimal	最大 38 桁の精度を持つ Decimal データ型をサポートするトランスフォーメーションの場合、15～38。 最大 28 桁の精度を持つ Decimal データ型をサポートするトランスフォーメーションの場合、15～28。	Double	Decimal
Decimal	最大 38 桁の精度を持つ Decimal データ型をサポートするトランスフォーメーションの場合、38 より大きい。 最大 28 桁の精度を持つ Decimal データ型をサポートするトランスフォーメーションの場合、28 より大きい。	Double	Double

高精度を有効にすると、データ統合サービスは式関数の数値定数を Decimal 型に変換します。高精度を有効にしないと、データ統合サービスは数値定数を Double 型に変換します。

数値の最大精度を、トランスフォーメーションに応じて、28 または 38 桁よりも大きくすることができます。トランスフォーメーション関数を使用して計算またはトランスフォーメーションを実行する前に、大きい数字を切り詰めるか、丸めます。

Double 値の丸め処理方式

システムの実行時ライブラリと、データベースが double データタイプ計算を行うコンピュータシステムの違いにより、予想外の結果になる場合があります。double データタイプは IEEE 794 標準に準拠します。データベースクライアントライブラリへの変更、データベースの異なるバージョン、システム実行時ライブラリへの変更は、算術的に同等な値のバイナリ表現に影響します。また、多くのシステム実行時ライブラリは偶数丸め方法や対称算術法を実装しています。偶数丸め方法は、数字が次の上下の数の中間にある場合、偶数の最小ビットの最も近い値に丸められます。例えば、偶数丸め方法では、0.125 は 0.12 に丸められます。対称算術法では末尾桁が 5 以上の場合は次の桁に丸められます。例えば、対称算術法では 0.125 は 0.13 に丸められ、0.124 は 0.12 に丸められます。

プラットフォームの違いによる計算結果への影響を少なくするため、Integration Service は有効桁数 15 桁の Double データタイプの値を格納しています。例えば、Windows で計算すると、数値 1234567890.1234567890 が返され、同じ計算を UNIX で実行すると、1234567890.1234569999 が返される場合、Integration Service はこの数値を 1234567890.1234600000 に変換します。

文字列データタイプ

トランスフォーメーションデータタイプには次の文字列データタイプがあります。

- 文字列
- テキスト

String、および Text データタイプは 104,857,600 文字までの同精度をサポートしますが、Integration Service は、ソースからターゲットに文字列データを移動する場合には String を、テキストデータを移動する場合には Text を使用します。一部のデータベースではテキストデータと文字列データの格納方法が異なるため、Integration Service ではこの 2 つの文字データタイプを区別する必要があります。通常、Char や Varchar などのサイズの小さい文字列データタイプは、トランスフォーメーション内で String として表示されますが、Text、Long、および Long Varchar などのサイズの大きな文字列データタイプは Text として表示されます。

トランスフォーメーション内では、String および Text を相互に入れ替えて使用できます。ただし、ルックアップトランスフォーメーションでは、ターゲットのデータタイプを一致させる必要があります。データベースドライバは、データが正確に渡されるように文字列データタイプとトランスフォーメーションデータタイプを一致させる必要があります。例えば、ルックアップテーブルの Varchar は、ルックアップトランスフォーメーションの String と一致しなければなりません。

DB2 for i5/OS および DB2 for z/OS のデータ型とトランスフォーメーションのデータ型

DB2 for i5/OS および DB2 for z/OS のデータ型は、IBM DB2 のデータ型と同じ方法で、トランスフォーメーションのデータ型に対応します。Data Integration Service は、トランスフォーメーションのデータ型を使用して、プラットフォーム間のデータ移動を行います。

以下の表に、DB2 for i5/OS および DB2 for z/OS のデータ型と、トランスフォーメーションのデータ型の比較を示します。

データ型	範囲	トランスフォーメーション	範囲
Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 精度 19、位取り 0
Char	1～254 文字	String	1～104,857,600 文字
ビットデータの Char	1～254 バイト	Binary	1～104,857,600 バイト
Date	西暦 0001 年～西暦 9999 年 精度 19、位取り 0（精度は日付まで）	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 （精度はナノ秒まで）
Decimal	精度 1～31、位取り 0～31	Decimal	精度 1～31、位取り 0～31
Float	精度 1～15	Double	精度 15

データ型	範囲	トランスフォーメーション	範囲
Integer	-2,147,483,648～2,147,483,647	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Smallint	-32,768～32,767	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
時間	24 時間制の時刻 精度 19、位取り 0 (精度は秒まで)	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Timestamp ¹	26 バイト 精度 26、位取り 6 (精度はマイクロ秒まで)	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Varchar	4,000 文字まで	String	1～104,857,600 文字
ビットデータの Varchar	4,000 バイトまで	Binary	1～104,857,600 バイト
1. DB2 for z/OS バージョン 10 の拡張精度タイムスタンプはトランスフォーメーションのデータ型に以下の ように対応します。 - 位取りが 6 の場合、精度は 26 でトランスフォーメーションのデータ型は date/time です。 - 位取りが 0 の場合、精度は 19 でトランスフォーメーションのデータ型は string です。 - 位取りが 1～5 または 7～12 の場合、精度は 20 以上でトランスフォーメーションのデータ型は string で す。			

DB2 for i5/OS および DB2 for z/OS のサポートされていないデータ型

Developer ツールは DB2 for i5/OS および DB2 for z/OS の一部のデータ型をサポートしていません。

Developer ツールは DB2 for i5/OS および DB2 for z/OS の large object (LOB) データ型はサポートしていません。LOB カラムはリレーショナルテーブルオブジェクト内ではサポートされないものとして、ネイティブタイプである varchar、および精度と位取りが 0 で表示されます。カラムは、カスタマイズされたデータオブジェクトまたは出力にマッピングには投影されません。

フラットファイルとトランスフォーメーションのデータ型

フラットファイルのデータ型は、データ統合サービスがプラットフォーム間でデータを移動するために使用するトランスフォーメーションのデータ型にマップされます。

次の表で、フラットファイルのデータ型とトランスフォーメーションのデータ型を比較します。

フラットファイル	トランスフォーメーション	範囲
Bigint	Bigint	精度 19、位取り 0
Datetime	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日（精度はナノ秒まで）
Double	Double	精度 15
Int	Integer	-2,147,483,648～2,147,483,647
Nstring	String	1～104,857,600 文字
Number	Decimal	最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
String	String	1～104,857,600 文字

フラットファイルの数値カラムから数値以外のデータを読み込むと、データ統合サービスはその行を削除し、ログにメッセージを書き込みます。また、フラットファイルの日時カラムから日時以外のデータを読み込むと、データ統合サービスはその行を削除し、ログにメッセージを書き込みます。

DB2 for LUW とトランスフォーメーションデータ型

DB2 for LUW のデータ型は、データ統合サービスがプラットフォーム間でデータを移動するために使用するトランスフォーメーションデータ型にマップされます。

次の表に、DB2 for LUW のデータ型とトランスフォーメーションデータ型の比較を示します。

データ型	範囲	トランスフォーメーション	範囲
Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 精度 19、位取り 0
Blob	1～2,147,483,647 バイト	Binary	1～104,857,600 バイト

データ型	範囲	トランスフォーメーション	範囲
文字	1～254 文字	String	1～104,857,600 文字
ビットデータの Char	1～254 バイト	Binary	1～104,857,600 バイト
Clob	1～2,447,483,647 バイト	Text	1～104,857,600 文字
Date	西暦 0001 年～西暦 9999 年 精度 19、位取り 0（精度は日付まで）	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 （精度はナノ秒まで）
Dblob	1～1073741823 バイト	Dblob	1～104,857,600 バイト
10 進型	精度 1～31、位取り 0～31	10 進型	精度 1～31、位取り 0～31
Float	精度 1～15	Double	精度 15
Graphic	1～127 バイト	Graphic	1～127 バイト
Integer	-2,147,483,648～2,147,483,647	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Long Varchar	32700 バイト	Long Varchar	32700 バイト
Long Vargraphic	163350 バイト	Long Vargraphic	163350 バイト
数値	精度 1～31、位取り 0～31	10 進型	精度 1～28、位取り 0～28
Smallint	-32,768～32,767	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
時間	24 時間制の時刻 精度 19、位取り 0 （精度は秒まで）	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 （精度はナノ秒まで）
タイムスタンプ	26 バイト 精度 26、位取り 6 （精度はマイクロ秒まで）	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 （精度はナノ秒まで）
Varchar	4,000 文字まで	String	1～104,857,600 文字
ビットデータの Varchar	4,000 バイトまで	Binary	1～104,857,600 バイト
Vargraphic	1～16336 バイト	Vargraphic	1～16336 バイト

JDBC のデータ型とトランスフォーメーションのデータ型

データ統合サービスでデータを JDBC ソースから読み取るときは、ネイティブデータタイプが対応する JDBC データ型に変換され、その後、トランスフォーメーションデータ型に変換されます。トランスフォーメーションデータ型は、プラットフォーム間でのデータの移動に使用されます。

以下の表は、JDBC データ型とトランスフォーメーションデータ型を比較します。

JDBC データ型	トランスフォーメーション	範囲
Bigint	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 精度 19、位取り 0
Binary*	バイナリ	1～104,857,600 バイト
Bit	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Blob*	バイナリ	1～104,857,600 バイト
ブール	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Char*	String	1～104,857,600 文字
Clob*	テキスト	1～104,857,600 文字
日付	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Decimal	Decimal	精度 1～28、位取り 0～28
ダブル	ダブル	精度 15
Float	Double	精度 15
Integer	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Long VarBinary*	バイナリ	1～104,857,600 バイト
Long Varchar*	テキスト	1～104,857,600 文字
Numeric	Decimal	精度 1～28、位取り 0～28
Real	ダブル	精度 15
Smallint	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0

JDBC データ型	トランスフォーメーション	範囲
Time	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
タイムスタンプ	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Tinyint	Integer	-2,147,483,648~2,147,483,647 精度 10、位取り 0
Varchar*	String	1~104,857,600 文字
Varbinary*	バイナリ	1~104,857,600 バイト
*ポートのデータサイズが 100 MB を超えている場合、デフォルトで Developer ツールはポート精度を 4000 に設定します。より大きいサイズのデータを処理するには、ポート精度を大きくします。		

Microsoft SQL Server のデータ型とトランスフォーメーションデータ型

Microsoft SQL Server のデータ型は、データ統合サービスがプラットフォーム間のデータ移動に使用するトランスフォーメーションデータ型に対応します。

次の表で、Microsoft SQL Server のデータ型とトランスフォーメーションデータ型を比較します。

Microsoft SQL Server	範囲	トランスフォーメーション	範囲
Binary	1~8,000 バイト	Binary	1~104,857,600 バイト
bigint	-9,223,372,036,854,775,808 - 9,223,372,036,854,775,807	bigint	-9,223,372,036,854,775,808~9,223,372,036,854,775,807 精度 19、位取り 0
Bit	1 ビット	String	1~104,857,600 文字
Char	1~8,000 文字	String	1~104,857,600 文字
Datetime	西暦 1753 年 1 月 1 日~西暦 9999 年 12 月 31 日 精度 23、位取り 3 (精度は 3.33 ミリ秒まで)	Date/Time	西暦 0001 年 1 月 1 日~西暦 9999 年 12 月 31 日 (精度はナノ秒まで)

Microsoft SQL Server	範囲	トランスフォーメーション	範囲
日時 2	西暦 0001 年 1 月 1 日 00:00:00 ～ 西暦 9999 年 12 月 31 日 23:59:59.9999999	タイムスタンプ	精度 22～27
Decimal	精度 1～38、位取り 0～38	Decimal	最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
Float	-1.79E+308～1.79E+308	Double	精度 15
Image	1～2,147,483,647 バイト	Binary	1～104,857,600 バイト
Int	-2,147,483,648～2,147,483,647	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
通貨	-922,337,203,685,477.5807～ 922,337,203,685,477.5807	Decimal	精度 1～28、位取り 0～28
nchar	1 - 4000 文字	String	1～104,857,600 文字
ntext	1 - 1,073,741,823 バイト	Text	1～104,857,600 文字
Numeric	精度 1～38、位取り 0～38	Decimal	最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 最大 28 桁の精度をサポートするトランスフォーメーションでは、精度が 1～28 桁、スケールが 0～28 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
nvarchar	1 - 4000 文字	String	1～104,857,600 文字
Real	-3.40E+38～3.40E+38	Double	精度 15
Smalldatetime	西暦 1900 年 1 月 1 日～西暦 2079 年 6 月 6 日 精度 19、位取り 0 (精度は分まで)	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)

Microsoft SQL Server	範囲	トランスフォーメーション	範囲
Smallint	-32,768～32,768	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Smallmoney	-214,748.3648～214,748.3647	Decimal	精度 1～28、位取り 0～28
Sysname	1～128 文字	String	1～104,857,600 文字
Text	1～2,147,483,647 文字	Text	1～104,857,600 文字
タイムスタンプ	8 バイト	Binary	1～104,857,600 バイト
Tinyint	0～255	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Uniqueidentifier	精度 38、位取り 0	String	Uniqueidentifier データを正常に移動または変更するために、データが次の形式であることを確認します。 XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX x は、0～9 の範囲の 16 進数または a～f です。
Varbinary	1～8,000 バイト	Binary	1～104,857,600 バイト
Varchar	1～8,000 文字	String	1～104,857,600 文字

Uniqueidentifier データタイプ

Uniqueidentifier は、グローバル一意識別子（GUID）を格納するための Microsoft SQL Server のデータタイプです。16 バイトのデータを格納できます。

Developer ツールは Uniqueidentifier データタイプを文字列として扱います。Uniqueidentifier データを移動または変更するには、Uniqueidentifier カラムを文字列カラムに接続します。Uniqueidentifier データを正常に移動または変更するために、データが次の形式であることを確認します。

XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX

x は、0～9 の範囲の 16 進数または a～f です。

例えば、6F9619FF-8B86-D011-B42D-00C04FC964FF は有効な Uniqueidentifier 値です。

Developer ツールでは、36 文字に 16 バイトの Uniqueidentifier データを格納できます。ただし、Uniqueidentifier データは 2 つの中括弧内に表示できるため、Developer ツールでは 2 つの追加文字を精度に割り当てて、中括弧に対応しています。Uniqueidentifier カラムを文字列カラムに接続する際、Uniqueidentifier データを正常に移動または変更するために、文字列カラムの精度を 38 に設定します。

非リレーショナルのデータ型とトランスフォーメーションのデータ型

非リレーショナルのデータ型は、Data Integration Service がプラットフォーム間のデータ移動に使用するトランスフォーメーションのデータ型に対応します。

非リレーショナルのデータ型は、以下のタイプのデータ型に適用されます。

- Adabas
- IMS
- シーケンシャル
- VSAM

以下の表に、非リレーショナルのデータ型とトランスフォーメーションのデータ型の比較を示します。

非リレーショナル	精度	トランスフォーメーション	範囲
BIN	10	バイナリ	1～104,857,600 バイト
CHAR	10	String	1～104,857,600 文字 固定長または可変長文字列。
DATE	10	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 結合された日時の値、ナノ秒単位の精度。
DOUBLE	18	Double	精度 15 桁 倍精度浮動小数点数値。
FLOAT	7	Double	精度 15 桁 倍精度浮動小数点数値。
NUM8	3	Integer	精度 10、位取り 0 整数値。
NUM8U	3	Integer	精度 10、位取り 0 整数値。
NUM16	5	Integer	精度 10、位取り 0 整数値。
NUM16U	5	Integer	精度 10、位取り 0 整数値。
NUM32	10	Integer	精度 10、位取り 0 整数値。
NUM32U	10	Double	精度 15 桁 倍精度浮動小数点数値。

非リレーショナル	精度	トランスフォーメーション	範囲
NUM64	19	Decimal	精度 1～28 桁、位取り 0～28 宣言された精度と位取りを持つ 10 進値。位取りは、精度以下にする必要があります。負の位取りを持つ値または 28 より大きい精度が宣言された値を渡した場合、倍精度に変換されます。
NUM64U	19	Decimal	精度 1～28 桁、位取り 0～28 宣言された精度と位取りを持つ 10 進値。位取りは、精度以下にする必要があります。負の位取りを持つ値または 28 より大きい精度が宣言された値を渡した場合、倍精度に変換されます。
NUMCHAR	100	String	1～104,857,600 文字 固定長または可変長文字列。
PACKED	15	Decimal	精度 1～28 桁、位取り 0～28 宣言された精度と位取りを持つ 10 進値。位取りは、精度以下にする必要があります。負の位取りを持つ値または 28 より大きい精度が宣言された値を渡した場合、倍精度に変換されます。
TIME	5	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 結合された日時の値、ナノ秒単位の精度。
TIMESTAMP	5	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 結合された日時の値、ナノ秒単位の精度。
UNPACKED	15	Decimal	精度 1～28 桁、位取り 0～28 宣言された精度と位取りを持つ 10 進値。位取りは、精度以下にする必要があります。負の位取りを持つ値または 28 より大きい精度が宣言された値を渡した場合、倍精度に変換されます。
UZONED	15	Decimal	精度 1～28 桁、位取り 0～28 宣言された精度と位取りを持つ 10 進値。位取りは、精度以下にする必要があります。負の位取りを持つ値または 28 より大きい精度が宣言された値を渡した場合、倍精度に変換されます。
VARBIN	10	バイナリ	1～104,857,600 バイト
VARCHAR	10	String	1～104,857,600 文字 固定長または可変長文字列。
ZONED	15	Decimal	精度 1～28 桁、位取り 0～28 宣言された精度と位取りを持つ 10 進値。位取りは、精度以下にする必要があります。負の位取りを持つ値または 28 より大きい精度が宣言された値を渡した場合、倍精度に変換されます。

ODBC データ型とトランスフォーメーションデータ型

ODBC データ型は、データ統合サービスがプラットフォーム間のデータ移動に使用するトランスフォーメーションデータ型にマッピングされます。

次の表では、Microsoft Access、Excel などの ODBC データ型をトランスフォーメーションデータ型と比較します。

データ型	トランスフォーメーション	範囲
Bigint	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 精度 19、位取り 0
バイナリ	Binary	1～104,857,600 バイト
Bit	String	1～104,857,600 文字
文字	String	1～104,857,600 文字
日付	日付/時刻	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Decimal	Decimal	最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
Double	Double	精度 15
Float	Double	精度 15
Integer	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Long Varbinary	Binary	1～104,857,600 バイト
Nchar	String	1～104,857,600 文字
Nvarchar	String	1～104,857,600 文字
Ntext	Text	1～104,857,600 文字
Numeric	Decimal	最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
Real	Double	精度 15

データ型	トランスフォーメーション	範囲
Smallint	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
テキスト	テキスト	1～104,857,600 文字
Time	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
タイムスタンプ	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Tinyint	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
Varbinary	Binary	1～104,857,600 バイト
Varchar	String	1～104,857,600 文字

Oracle のデータ型およびトランスフォーメーションデータ型

Oracle のデータ型は、データ統合サービスがプラットフォーム間のデータ移動に使用するトランスフォーメーションのデータ型に対応します。

次の表で、Oracle のデータ型とトランスフォーメーションデータ型を比較します。

Oracle	範囲	トランスフォーメーション	範囲
Blob	4GB まで	Binary	1～104,857,600 バイト
Char (L)	1～2,000 バイト	String	1～104,857,600 文字
Clob	4GB まで	Text	1～104,857,600 文字
Date	紀元前 4712 年 1 月 1 日～ 西暦 4712 年 12 月 31 日 精度 19、位取り 0	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Float	精度 1～15、位取り 0	Double	精度 15、位取り 0

Oracle	範囲	トランスフォーメーション	範囲
Long	2GB まで	Text	1～104,857,600 文字 マッピングに Long データが含まれている場合、統合サービスはそれをトランスフォーメーションの文字列データ型に変換し、104,857,600 文字に切り詰めます。
Long Raw	2GB まで	Binary	1～104,857,600 バイト
Nchar	1～2,000 バイト	String	1～104,857,600 文字
Nclob	4GB まで	Text	1～104,857,600 文字
Number	精度 1～38	Double	精度 15
Number (P,S)	精度 1～38、 位取り 0～38	Decimal	最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1～38 桁、スケールが 0～38 になります。 この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。
Nvarchar2	1～4,000 バイト	String	1～104,857,600 文字
RAW	1～2,000 バイト	Binary	1～104,857,600 バイト
タイムスタンプ	紀元前 4712 年 1 月 1 日～ 西暦 9999 年 12 月 31 日 精度 19～29、位取り 0～9 (精度はナノ秒まで)	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)

Oracle	範囲	トランスフォーメーション	範囲
Timestamp with Time Zone	紀元前 4712 年 1 月 1 日～ 西暦 9999 年 12 月 31 日 -12:00～+14:00 精度 26 - 36、位取り 0 - 9 (精度はナノ秒まで)	timestampWithTZ	西暦 1947 年 8 月 1 日～西暦 2040 年 12 月 31 日 -12:00～+14:00 精度 36、スケール 9。 (精度はナノ秒まで) Timestamp with Time Zone データ型では、次のタイムゾーン地域はサポートされません。 - AFRICA_CAIRO - AFRICA_MONROVIA - エジプト - AMERICA_MONTR EAL
Timestamp with Local Time Zone	紀元前 4712 年 1 月 1 日～ 西暦 9999 年 12 月 31 日 精度 19～29、位取り 0～9 (精度はナノ秒まで)	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Varchar	1～4,000 バイト	String	1～104,857,600 文字
Varchar2	1～4,000 バイト	String	1～104,857,600 文字
XMLType	4GB まで	Text	1～104,857,600 文字

Number (P,S) データ型

Developer tool は、負の位取りを持つ Oracle Number(P,S)値をサポートしています。

ただし、精度 38 より大きい位取り、または負の精度を持つ Number (P,S) 値はサポートしていません。

負の位取りの Oracle Number をインポートした場合、Developer tool には Decimal データ型として表示されます。ただし、データ統合サービスにより倍精度に変換されます。

Char、Varchar、Clob データタイプ

データ統合サービスでは、Unicode データ移動モードを使用する場合、Oracle データベースのカラムに設定した長さのセマンティクスに応じて、Char カラム、Varchar カラム、および Clob カラムの精度が読み込まれます。

バイトセマンティクスを使用してカラム長を決定する場合は、データ統合サービスは精度をバイト数として読み込みます。Char セマンティクスを使用する場合は、データ統合サービスは精度を文字数として読み込みます。

サポートされていない Oracle のデータ型

Developer tool は Oracle の一部のデータ型をサポートしていません。

Developer tool でサポートされていない Oracle のデータ型は以下のとおりです。

- Bfile
- Interval Day to Second
- Interval Year to Month
- Mslabel
- Raw Mslabel
- Rowid

SAP HANA およびトランスフォーメーションデータ型

SAP HANA データ型は、Data Integration Service がプラットフォーム間のデータ移動に使用するトランスフォーメーションのデータ型に対応します。

以下の表に、SAP HANA データ型およびトランスフォーメーションデータ型の比較を示します。

SAP HANA データ型	範囲	トランスフォーメーションデータ型	範囲
英数字	精度 1 - 127	Nstring	1～104,857,600 文字
Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 精度 19、位取り 0
バイナリ	バイナリデータのバイトの格納に使用	Binary	1～104,857,600 バイト
Blob	2GB まで	Binary	1～104,857,600 バイト
Clob	2GB まで	Text	1～104,857,600 文字
Date	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 精度 10、位取り 0	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)

SAP HANA データ型	範囲	トランスフォーメーションデータ型	範囲
Decimal (精度, 位取り) または Dec (p, s)	精度 1 - 34	Decimal	<p>最大 38 桁の精度をサポートするトランスフォーメーションでは、精度が 1 ～38 桁、スケールが 0～38 になります。</p> <p>最大 28 桁の精度をサポートするトランスフォーメーションでは、精度が 1 ～28 桁、スケールが 0～28 になります。</p> <p>この最大桁数を上回る精度を指定した場合、データ統合サービスは 10 進値を高精度モードの倍精度浮動小数点数に変換します。</p>
ダブル	単精度 64 ビット浮動小数点数を指定	Double	精度 15
Float	精度 1 - 53	Double	精度 15
Integer	-2,147,483,648～2,147,483,647	Integer	-2,147,483,648～2,147,483,647 精度 10、位取り 0
NClob	2GB まで	Ntext	1～104,857,600 文字
Nvarchar	精度 1 - 5000	Nstring	1～104,857,600 文字
Real	単精度 32 ビット浮動小数点数を指定	Real	精度 7、位取り 0
Seconddate	0001-01-01 00:00:01 - 9999-12-31 24:00:00	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
Shorttext	可変長文字列を指定、このデータ型ではテキスト検索および文字列検索機能をサポート	Nstring	1～104,857,600 文字
Smalldecimal	精度 1 - 16	Decimal	精度 1～28、位取り 0～28
Smallint	-32,768～32,767	Small Integer	精度 5、位取り 0
Text	可変長文字列を指定、このデータ型ではテキスト検索機能をサポート	Text	1～104,857,600 文字
時間	24 時間制の時刻	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
タイムスタンプ	0001-01-01 00:00:00.0000000 - 9999-12-31 23:59:59.9999999	Date/Time	西暦 0001 年 1 月 1 日～西暦 9999 年 12 月 31 日 (精度はナノ秒まで)

SAP HANA データ型	範囲	トランスフォーメーションデータ型	範囲
Tinyint	0～255	Small Integer	精度 5、位取り 0
Varchar	精度 1 - 5000	String	1～104,857,600 文字
Varbinary	1 - 5000 バイト	Binary	1～104,857,600 バイト

XML データ型とトランスフォーメーションデータ型

XML のデータ型は、データ統合サービスがデータのプラットフォーム間の移動に使用するトランスフォーメーションデータ型にマッピングされます。

データ統合サービスは、W3C が 2001 年 5 月 2 日の勧告で指定した XML データ型をすべてサポートしています。しかし、データ統合サービスは XML の値の範囲全体には対応していない場合があります。XML データタイプの詳細については、次のサイトで XML データタイプの W3C 仕様を参照してください:

<http://www.w3.org/TR/xmlschema-2>

以下の表に、XML データ型とトランスフォーメーションデータ型の比較を示します。

データ型	トランスフォーメーション	範囲
anyURI	String	1～104,857,600 文字
base64Binary	バイナリ	1～104,857,600 バイト
boolean	String	1～104,857,600 文字
byte	Integer	-2,147,483,648～2,147,483,647
日付	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
dateTime	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日 (精度はナノ秒まで)
decimal	Decimal	精度 1～28、位取り 0～28
double	Double	15 桁精度
duration	String	1～104,857,600 文字
ENTITIES	String	1～104,857,600 文字
ENTITY	String	1～104,857,600 文字
float	Double	15 桁精度

データ型	トランスフォーメーション	範囲
gDay	String	1～104,857,600 文字
gMonth	String	1～104,857,600 文字
gMonthDay	String	1～104,857,600 文字
gYear	String	1～104,857,600 文字
gYearMonth	String	1～104,857,600 文字
hexBinary	バイナリ	1～104,857,600 バイト
ID	String	1～104,857,600 文字
IDREF	String	1～104,857,600 文字
IDREFS	String	1～104,857,600 文字
整数型	Integer	-2,147,483,648～2,147,483,647
Integer	Integer	-2,147,483,648～2,147,483,647
language	String	1～104,857,600 文字
長整数型	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
名前	String	1～104,857,600 文字
NCName	String	1～104,857,600 文字
negativeInteger	Integer	-2,147,483,648～2,147,483,647
NMTOKEN	String	1～104,857,600 文字
NMTOKENS	String	1～104,857,600 文字
nonNegativeInteger	Integer	-2,147,483,648～2,147,483,647
nonPositiveInteger	Integer	-2,147,483,648～2,147,483,647
normalizedString	String	1～104,857,600 文字
NOTATION	String	1～104,857,600 文字
positiveInteger	Integer	-2,147,483,648～2,147,483,647
QName	String	1～104,857,600 文字
short	Integer	-2,147,483,648～2,147,483,647
string	String	1～104,857,600 文字

データ型	トランスフォーメーション	範囲
時間	日付/時刻	西暦 0001 年 1 月 1 日 - 西暦 9999 年 12 月 31 日（精度はナノ秒まで）
トークン	String	1～104,857,600 文字
unsignedByte	Integer	-2,147,483,648～2,147,483,647
unsignedInt	Integer	-2,147,483,648～2,147,483,647
unsignedLong	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
unsignedShort	Integer	-2,147,483,648～2,147,483,647

データの変換

あるデータ型から別のデータ型にデータを変換できます。

別のデータ型に変換するには、以下のいずれかの方法を使用します。

- ポート間でデータを渡す（ポート対ポートのデータ変換）。
- トランスフォーメーション関数を使ってデータを変換する。
- トランスフォーメーション算術演算子を使ってデータを変換する。

ポート対ポートのデータ変換

データ統合サービスは、ポートのデータ型に基づいてデータを変換します。データがポートを通過するたびに、データ統合サービスはポートに割り当てられているデータ型を識別し、必要に応じてサポートされているデータ型に基づいてデータを変換します。

同じデータ型のポート間でデータを渡す

同じデータ型のポート間でデータを渡すためのマッピングを作成できます。同じデータ型のポート間でデータを渡す場合、データ統合サービスはデータまたはデータ型を変更せずにデータを渡します。

同じ数値データ型のポート間でデータを渡し、データがトランスフォーメーション間で転送される場合、データ統合サービスは必ずしもデータの転送先のポートの精度と位取りにデータを変換しません。

以下の方法を使用すると、異なる精度、位取り、および精度モードの 10 進型データをポート間で転送できます。

10 進型データを低精度モードで渡す

低精度モードでデータを 10 進型ポートに渡す場合、データ統合サービスは精度 15 および位取り 0 ですべての 10 進型ポートを倍精度浮動小数点数に変換します。例えば、低精度モードで 10 進型 (14, 3) から 10 進型 (9, 5) に値を渡すとします。データ統合サービスは、値を内部的に格納し、低精度モードでのデータの切り詰めは行いません。

高精度モードで位取りを下げて 10 進型データを渡す

高精度モードで位取りを下げて 10 進型ポート間でデータを 10 進型ポートに渡す場合は、位取りが切り詰められます。例えば、10 進型 (18, 5) から 10 進型 (18, 2) に値を渡すとします。18.01234 を渡す場合、データ統合サービスはデータの位取りを切り詰めて、出力データは 18.01 になります。

高精度モードで精度を下げて 10 進型データを渡す

高精度モードで精度を下げてデータを 10 進型ポートに渡すことができます。例えば、高精度モードで 10 進型 (19, 5) から 10 進型 (17, 2) に値を渡すとします。出力フィールドに 17 桁を超える値が含まれる場合、データ統合サービスはその行を拒否します。

異なるデータ型のポート間でデータを渡す

異なるデータ型のポート間でデータを渡す場合、データ統合サービスはトランスフォーメーション言語の変換関数を使用してデータ型間の変換を行います。

例えば、文字列のポートを整数のポートに接続するとします。データ統合サービスがマッピングを実行する場合は、TO_INTEGER 関数を使用して入力データを文字列データ型から整数データ型に変換します。

データ統合サービスがポート間の変換を実行する場合は、渡すデータが変換先のデータ型に対して有効である必要があります。データ統合サービスが変換できない値があると、トランスフォーメーションの行エラーが発生します。例えば、「9,000,000,000,000,000,000.777」という値を含む文字列のポートを Bigint 型のポートに接続するとします。この場合、データ統合サービスは String の値を Bigint の値に変換できずに、エラーを返します。

データ統合サービスは、トランスフォーメーション間、およびパイプライン内の最後のトランスフォーメーションとターゲットとの間でポート対ポート変換を行います。

以下の表に、データ統合サービスで実行されるポート対ポート変換を示します。

データ型	Bigint	整数型	10 進型	Double	String、Text	Date/Time	Binary	Timestamp with Time Zone
Bigint	いいえ	はい	はい	はい	はい	いいえ	いいえ	いいえ
Integer	はい	いいえ	はい	はい	はい	いいえ	いいえ	いいえ
Decimal	はい	はい	いいえ	はい	はい	いいえ	いいえ	いいえ
Double	はい	はい	はい	いいえ	はい	いいえ	いいえ	いいえ
String、Text	はい	はい	はい	はい	はい	はい	いいえ	はい
Date/Time	いいえ	いいえ	いいえ	いいえ	はい	はい	いいえ	いいえ
Binary	いいえ	いいえ	いいえ	いいえ	いいえ	いいえ	はい	いいえ
Timestamp with Time Zone	いいえ	いいえ	いいえ	いいえ	はい	いいえ	いいえ	はい

付録 B

キーボードショートカット

この付録では、以下の項目について説明します。

- [オブジェクトのキーボードショートカット, 190](#) ページ
- [ポートのキーボードショートカット, 191](#) ページ
- [\[トランスフォーメーション\] パレットのキーボードショートカット, 192](#) ページ
- [ワークベンチのキーボードショートカット, 192](#) ページ

オブジェクトのキーボードショートカット

キーボードショートカットを使用するとエディタ内のオブジェクトで作業ができます。

エディタのオブジェクトを選択した場合、オブジェクトの外観を変更したり、オブジェクトを移動できます。オブジェクトが選択されているときには、緑色の境界線がオブジェクトの周囲に表示されます。オブジェクトがフォーカスされていることを示す点線も選択したオブジェクトの周囲に表示されます。その他のオブジェクトが選択できる状態でオブジェクトをフォーカスできます。エディタ内で複数のオブジェクトを選択する場合に選択したオブジェクトから他のオブジェクトにフォーカスを移動します。

注: エディタ内のオブジェクトの移動順序は上から下、左から右です。

キーボードショートカットで次のタスクを実行できます。

オブジェクトを選択します。

エディタが選択されているときに、**Tab** を押します。このオブジェクトもフォーカスされています。**Tab** を再度押して次のオブジェクトを選択します。

前のオブジェクトを選択します。

Shift+Tab を押します。

特定のオブジェクトを検索します。

Ctrl+O を押します。

次のオブジェクトをフォーカスします。

オブジェクトがフォーカスされているときに **Ctrl+Tab** を押すと、フォーカスを次のオブジェクトに移動できます。前のオブジェクトは選択されたままです。**Ctrl+Tab** を再度押して次のオブジェクトをフォーカスします。

複数のオブジェクトを個別に選択します。

オブジェクトがフォーカスされているときに **Ctrl+Tab** を押すと、フォーカスを次に選択するオブジェクトに移動できます。次に **Ctrl+Spacebar** を押してフォーカスしたオブジェクトを選択します。この手順を繰り返して複数のオブジェクトを個別に選択します。

選択したオブジェクトをアイコン化します。

Ctrl+I を押します。

選択したオブジェクトをリストアします。

Ctrl+R を押します。

選択したオブジェクトをサイズ変更します。

Ctrl+Shift を押し続けながら、**Up**、**Down**、**Left**、**Right** 方向キーを使用します。

選択したオブジェクトを移動します。

方向キーを使用すると、1 つまたは複数のオブジェクトをどの方向にも 1 ピクセルずつ移動できます。

オブジェクトからエディタに移動します。

1 つまたは複数のオブジェクトが選択されているかフォーカスされているときに、**Esc** を押します。

ポートのキーボードショートカット

キーボードショートカットを使用してポートで作業ができます。

ポートを選択した場合、ポートを編集してそのポートを他のポートにリンクできます。ポートが選択された場合、ポート全体が緑色で表示されます。ポートがフォーカスされていることを示す点線も選択したポートのポート値の周囲に表示されます。その他のポートが選択できる状態でポートをフォーカスできます。複数のポートを個別に選択する場合に 1 つのポートから他のポートにフォーカスを移動します。

キーボードショートカットで次のタスクを実行できます。

ポートを選択します。

オブジェクトが選択され、フォーカスされているときに、**Ctrl+G** を押します。**Up** および **Down** 方向キーを押して異なるポートを選択します。

複数のポートを選択します。

Shift+Up または **Shift+Down** を押します。

複数のポートを個別に選択します。

Ctrl キーを押し続けながら、**Up** および **Down** 方向キーを使用して選択するポートをフォーカスします。

Ctrl+Spacebar を押してポートを選択します。この手順を繰り返して複数のポートを個別に選択します。

次のポート値をフォーカスして編集します。

Tab を押します。

前のポート値をフォーカスして編集します。

Shift+Tab を押します。

ポートからオブジェクトに移動します。

Esc キーを押します。

2 つのオブジェクトのポートをリンクします。

最初のオブジェクトでリンクするポートを選択し、**Ctrl+L** を押します。選択したポートは強調表示されます。2 番目のオブジェクトに移動してリンクするポートを選択します。**Ctrl+L** を押してポートをリンクします。

リンクモードを終了します。

Ctrl+Alt+L を押して最初のオブジェクトで選択したポートをクリアします。

リンクを選択します。

選択したポートで **Ctrl+G** を押して、そのポートからの出力リンクを選択します。

次のリンクを選択します。

Tab を押します。

前のリンクを選択します。

Shift+Tab を押します。

【トランスフォーメーション】パレットのキーボードショートカット

キーボードショートカットを使用して【トランスフォーメーション】パレットを操作できます。

キーボードショートカットで次のタスクを実行できます。

エディタから【トランスフォーメーション】パレットに移動する。

エディタが選択されているときに、**Ctrl+Shift+P** を押します。

【トランスフォーメーション】パレットで移動する。

Tab キーを押すか、矢印キーの上向きまたは下向きを押します。

【トランスフォーメーション】パレットからエディタに移動する。

Esc キーを押します。

ワークベンチのキーボードショートカット

キーボードショートカットを使用してワークベンチ内のビューやエディタに移動できます。

エディタを選択した場合、エディタ内を移動できます。選択された場合、緑色の境界線がエディタの周囲に表示されます。

エディタおよびビューをフォーカスできます。フォーカスされている場合、エディタまたはビューの周囲に青色の境界線が表示されます。点線がフォーカスされているビューのタブの周囲に表示されます。ビューがフォーカスされている場合、そのビューを選択したり、他のビューをフォーカスできます。

キーボードショートカットで次のタスクを実行できます。

ビューをフォーカスします。

エディタが選択されているときに、**Shift+Tab** を押します。**Left** および **Right** 方向キーを使用して同じエディタ内の他のビューをフォーカスします。また、**Shift+Tab** をワークベンチの別のビューがフォーカスされるまで押します。

ビューで最初の領域を選択します。

ビューがエディタ内でフォーカスされた場合、**Ctrl+Tab** を押します。**Ctrl+Tab** を再度押して次の領域を選択します。

ビューで前の領域を選択します。

Ctrl+Shift+Tab を押します。

マッピングの [プロパティ] ビューで、説明を選択して参照します。

Ctrl+A を押します。

エディタを最小化します。

ビューがフォーカスされた場合、**Shift+Tab** を押して [最小化] コントロールを選択し、次に Space キーを押します。

エディタを最大化します。

ビューがフォーカスされている場合、**Shift+Tab** を押して [最小化] コントロールを選択します。**Right** 方向キーを使用して [最大化] コントロールを選択し、Space キーを押します。

ワークベンチの領域を選択します。

エディタが選択されている場合、**Ctrl+Tab** を押して、ビューなど、ワークベンチのその他の領域を選択します。**Ctrl+Tab** を再度押して次の領域を選択します。

ワークベンチの前の領域を選択します。

Ctrl+Shift+Tab を押します。

ダイアログボックスのエラーメッセージに移動します。

Ctrl+M を押します。

付録 C

接続プロパティ

この付録では、以下の項目について説明します。

- [接続プロパティの概要, 195 ページ](#)
- [Adabas 接続のプロパティ, 195 ページ](#)
- [DataSift 接続プロパティ, 198 ページ](#)
- [Facebook 接続プロパティ, 199 ページ](#)
- [Greenplum 接続プロパティ, 199 ページ](#)
- [HBase 接続プロパティ, 201 ページ](#)
- [HDFS 接続プロパティ, 202 ページ](#)
- [Hive 接続のプロパティ, 203 ページ](#)
- [HTTP 接続プロパティ, 209 ページ](#)
- [IBM DB2 接続のプロパティ, 211 ページ](#)
- [IBM DB2 for i5/OS 接続プロパティ, 214 ページ](#)
- [IBM DB2 for z/OS 接続のプロパティ, 218 ページ](#)
- [IMS 接続のプロパティ, 221 ページ](#)
- [JDBC 接続のプロパティ, 224 ページ](#)
- [JD Edwards EnterpriseOne 接続プロパティ, 228 ページ](#)
- [LDAP 接続プロパティ, 229 ページ](#)
- [LinkedIn 接続プロパティ, 230 ページ](#)
- [Microsoft Azure Blob ストレージ接続のプロパティ, 231 ページ](#)
- [Microsoft Azure SQL Data Warehouse 接続プロパティ, 232 ページ](#)
- [MS SQL Server 接続のプロパティ, 233 ページ](#)
- [Netezza 接続のプロパティ, 237 ページ](#)
- [OData 接続のプロパティ, 238 ページ](#)
- [ODBC 接続のプロパティ, 239 ページ](#)
- [Oracle 接続のプロパティ, 241 ページ](#)
- [Salesforce 接続プロパティ, 243 ページ](#)
- [SAP 接続のプロパティ, 244 ページ](#)
- [シーケンシャル接続のプロパティ, 247 ページ](#)
- [Teradata Parallel Transporter 接続のプロパティ, 249 ページ](#)
- [Twitter 接続プロパティ, 251 ページ](#)
- [Twitter Streaming 接続プロパティ, 252 ページ](#)

- [VSAM 接続のプロパティ, 253 ページ](#)
- [Web Content-Kapow Katalyst 接続プロパティ, 255 ページ](#)
- [Web サービス接続のプロパティ, 257 ページ](#)
- [データベース接続における識別子のプロパティ, 258 ページ](#)

接続プロパティの概要

接続プロパティを使用すると、Informatica クライアントでデータソースに接続できます。

この章では、Informatica クライアントで作成および管理できる各接続の接続プロパティについて説明します。

Adabas 接続のプロパティ

Adabas データベースにアクセスするには Adabas 接続を使用します。Adabas 接続は、メインフレームデータベースタイプの接続です。Adabas 接続は Developer ツールで作成します。Adabas 接続は、Administrator ツールまたは Developer ツールで管理できます。

以下の表に、Adabas 接続のプロパティを示します。

オプション	説明
場所	Adabas に接続する PowerExchange リスナの場所のノード名。ノード名は、PowerExchange のコンフィギュレーションファイル (dbmover.cfg) 内の NODE 文の最初のパラメータで定義されます。
ユーザー名	データベースユーザー名。 サポートされている Linux または UNIX システムのデータベースで、PowerExchange LDAP ユーザー認証を有効にした場合、ユーザー名はエンタープライズユーザー名になります。詳細については、『 <i>PowerExchange リファレンスマニュアル</i> 』を参照してください。

オプション	説明
パスワード	<p>データベースユーザー名のパスワードまたは有効な PowerExchange のパスフレーズ。</p> <p>PowerExchange パスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p> <p>IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: 有効な RACF のパスフレーズは最長 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p>
コードページ	<p>必須。データソースの読み取りや書き込みに使用するコードページの名前。通常この値は ISO コードページの名前です（ISO-8859-6 など）。</p>
パススルーセキュリティ有効	<p>接続のパススルーセキュリティを有効にする。接続のパススルーセキュリティを有効にすると、ドメインは、接続オブジェクトに定義されている資格情報ではなく、クライアントのユーザー名とパスワードを使用して対応するデータベースにログインします。</p>
暗号化タイプ	<p>データ統合サービスが使用する暗号化のタイプ。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - RC2 - DES <p>デフォルトは [なし] です。</p> <p>注:</p> <ul style="list-style-type: none"> - Informatica では、暗号化タイプおよびレベル接続プロパティを設定したり、DBMOVER 構成ファイルに ENCRYPT 文と ENCRYPTLEVEL 文を設定する代わりに、Secure Sockets Layer（SSL）認証を使用することを推奨します。SSL 認証は複数の Informatica 製品で使用されており、より厳格なセキュリティが提供されます。 <p>PowerExchange ネットワークへの SSL 認証の実装の詳細については、『PowerExchange リファレンスマニュアル』を参照してください。</p> <ul style="list-style-type: none"> - ENCRYPT および ENCRYPTLEVEL 文の値が統合サービスマシンの DBMOVER 構成ファイルで定義されている場合、暗号化タイプおよびレベル接続属性に選択する値によって、これらの値はオーバーライドされます。マッピングの暗号化を有効にするには、必ず該当する接続属性を選択します。

オプション	説明
[暗号化] レベル	<p>[暗号タイプ] で [RC2] または [DES] を選択した場合は、データ統合サービスが使用する暗号化レベルを指定するために、次に挙げるオプションのいずれかを選択してください。</p> <ul style="list-style-type: none"> - 1. DES および RC2 には 56 ビットの暗号化キーを使用します。 - 2. DES には 168 ビットのトリプル暗号化キーを、RC2 には 64 ビットの暗号化キーを使用します。 - 3. DES には 168 ビットのトリプル暗号化キーを、RC2 には 128 ビットの暗号化キーを使用します。 <p>暗号化タイプを選択しない場合、このオプションは無視されます。 デフォルトは 1 です。</p>
ペーシングサイズ	<p>オプション。ソースシステムが PowerExchange リスナに渡すことができるデータ量。ペーシングサイズは、外部アプリケーション、データベース、またはデータ統合サービスのノードがボトルネックになっている場合に設定します。低い値を使用するとパフォーマンスが向上します。</p> <p>最小値およびデフォルト値は 0 です。値を 0 にするとパフォーマンスが最大になります。</p>
行として解釈	<p>オプション。このオプションは、ペーシングサイズを行数で表示する場合に選択します。このオプションをクリアすると、ペーシングサイズがキロバイト単位で表示されます。デフォルトではこのオプションが選択されていないため、ペーシングサイズはキロバイトで表示されています。</p>
圧縮	<p>オプション。このオプションは、ソースデータの圧縮を有効にする場合に選択します。データを圧縮することで、Informatica アプリケーションがネットワークで送信するデータ量を減少させることができます。デフォルトではこのオプションが選択されていないため、圧縮は無効になっています。</p>
オフロード処理	<p>オプション。バルクデータの処理をソースマシンからデータ統合サービスのマシンにオフロードするかどうかを制御します。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - AUTO。オフロード処理を使用するかどうかデータ統合サービスによって決定されます。 - はい。オフロード処理を使用します。 - いいえ。オフロード処理は使用しません。 <p>デフォルトは [AUTO] です。</p>
ワーカースレッド	<p>オプション。オフロード処理の有効時にデータ統合サービスがバルクデータを処理するために使用するスレッドの数。最適なパフォーマンスを得るためには、データ統合サービスマシンで使用可能なプロセッサの数を超えないようにこの値を設定します。有効な値は 1~64 です。デフォルトは 0 です。マルチスレッド処理は無効になります。</p>

オプション	説明
配列サイズ	オプション。ワーカースレッドのためのストレージアレイのレコード数。このオプションは、 [ワーカースレッド] オプションを 0 より大きな値に設定したときに適用されます。有効な値は 1~100000 です。デフォルトは 25 です。
書き込みモード	<p>オプション。データ統合サービスが PowerExchange リスナにデータを送信する際のモード。次のうちいずれかの書き込みモードを選択します。</p> <ul style="list-style-type: none"> - CONFIRMWRITEON。PowerExchange リスナにデータを送信し、応答を待ってから以降のデータを送信します。このオプションはエラーリカバリを優先する場合に選択します。ただし、このオプションを使用するとパフォーマンスが落ちる可能性があります。 - CONFIRMWRITEOFF。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションは、エラーが発生したときにターゲットテーブルを再ロードしてもよい場合に使用します。 - ASYNCHRONOUSWITHFAULTTOLERANCE。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションはエラー検出も有効にします。このオプションは、CONFIRMWRITEOFF のスピードと CONFIRMWRITEON のデータの完全性を結合します。 <p>デフォルトは [CONFIRMWRITEON] です。</p>

DataSift 接続プロパティ

DataSift 接続を使用し、DataSift ストリームからデータを抽出します。DataSift 接続は、ソーシャルメディア接続です。DataSift 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、DataSift 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。DataSift を選択します。
ユーザー名	DataSift アカウントのユーザー名。
API キー	API キー。Developer API キーは、DataSift アカウントの [ダッシュボード] ページまたは [設定] ページに表示されます。

Facebook 接続プロパティ

Facebook 接続を使用して、Facebook Web サイトからデータにアクセスします。Facebook 接続は、ソーシャルメディア接続です。Facebook 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Facebook 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。[Facebook] を選択します。
OAuth の詳細の有無	OAuth を設定するかどうかを示します。次のうちいずれかの値を選択します。 - [はい]。アクセストークンがあることを示します。 - [いいえ]。OAuth Utility を起動します。
コンシューマキー	Facebook にアプリケーションを作成すると支給される App ID。Facebook はこのキーを使用してアプリケーションを識別します。
コンシューマシークレット	Facebook にアプリケーションを作成すると支給されるアプリケーションシークレット。Facebook はこのシークレットを使用してコンシューマキーの所有権を確立します。
アクセストークン	OAuth Utility から返されるアクセストークン。Facebook は、ユーザークレデンシャルではなくこのトークンを使用して保護されたリソースにアクセスします。
アクセスシークレット	アクセスシークレットは Facebook 接続に必要ありません。
スコープ	アプリケーションの権限。OAuth の設定に使用した権限を入力します。

Greenplum 接続プロパティ

Greenplum 接続を使用して、Greenplum データベースに接続します。Greenplum 接続は、リレーショナルタイプの接続です。Greenplum 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

Greenplum 接続の作成時には、メタデータおよびデータアクセスの情報を入力します。

以下の表に、Greenplum 接続のプロパティを示します。

プロパティ	説明
名前	Greenplum リレーショナル接続の名前。
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。

メタデータをインポートするには、ユーザー名、パスワード、ドライバ名、接続文字列が必要です。以下の表に、メタデータアクセスのプロパティを示します。

プロパティ	説明
ユーザー名	Greenplum データベースにアクセスする権限を持つユーザー名。
パスワード	Greenplum データベースに接続するためのパスワード。
ドライバ名	Greenplum JDBC ドライバの名前。 例: com.pivotal.jdbc.GreenplumDriver ドライバの詳細については、Greenplum のドキュメントを参照してください。
接続文字列	以下の接続 URL を使用します。 jdbc:pivotal:greenplum://<hostname>:<port>;DatabaseName=<database_name> 接続 URL の詳細については、Greenplum のドキュメントを参照してください。

PowerExchange for Greenplum では、ホスト名、ポート番号、データベース名を使用して制御ファイルを作成し、ロード仕様を Greenplum gpload バルクロードユーティリティに提供します。これは [SSL の有効化] オプションと証明書パスを使用して、SSL を介した Greenplum サーバーへの安全な通信を確立します。

以下の表に、データアクセスの接続プロパティを示します。

プロパティ	説明
ホスト名	Greenplum サーバーのホスト名または IP アドレス。
ポート番号	Greenplum サーバーのポート番号。0 を入力すると、gpload ユーティリティは環境変数 \$PGPORT から読み取ります。デフォルトは 5432 です。
データベース名	データベースの名前。
SSL の有効化	SSL を介した gpload ユーティリティと Greenplum サーバー間の安全な通信を確立するには、このオプションを選択します。
証明書パス	Greenplum サーバーの SSL 証明書が保存されている場所へのパス。 証明書パスに存在している必要があるファイルの詳細については、gpload のマニュアルを参照してください。

HBase 接続プロパティ

HBase 接続を使用して HBase にアクセスします。HBase 接続は NoSQL 接続です。HBase 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。Hbase 接続プロパティは、特に明記されている場合を除き、大文字と小文字が区別されます。

以下の表に、HBase 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。このプロパティは、接続を作成した後に変更できます。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、4,000 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。HBase を選択します。
ZooKeeper ホスト	ZooKeeper サーバーをホストするマシンの名前。この名前は、大文字と小文字が区別されます。 ZooKeeper をレプリケートモードで実行する場合は、ZooKeeper quorum サーバーの中のカンマで区切られたサーバーリストを指定します。サーバーへの TCP 接続が切断された場合、クライアントは quorum の中の別のサーバーに接続します。
ZooKeeper ポート	ZooKeeper サーバーをホストするマシンのポート番号。
Kerberos 接続の有効化	Kerberos 認証を使用する HBase Master サーバーまたは Region サーバーと通信するために、Informatica ドメインを有効にします。

プロパティ	説明
HBase Master プリンシパル	<p>HBase Master サーバーのサービスプリンシパル名 (SPN)。Kerberos 認証を使用する HBase Master サーバーと通信するために、ZooKeeper サーバーを有効にします。</p> <p>文字列は次の形式で入力します。</p> <p><code>hbase/<domain.name>@<YOUR-REALM></code></p> <p>説明:</p> <ul style="list-style-type: none"> - domain.name は、HBase Master サーバーをホストするマシンのドメイン名です。 - YOUR-REALM は、Kerberos レalm です。
HBase Region サーバープリンシパル	<p>HBase Region サーバーのサービスプリンシパル名 (SPN)。Kerberos 認証を使用する HBase Region サーバーと通信するために、ZooKeeper サーバーを有効にします。</p> <p>文字列は次の形式で入力します。</p> <p><code>hbase_rs/<domain.name>@<YOUR-REALM></code></p> <p>説明:</p> <ul style="list-style-type: none"> - domain.name は、HBase Master サーバーをホストするマシンのドメイン名です。 - YOUR-REALM は、Kerberos レalm です。

HDFS 接続プロパティ

Hadoop ファイルシステム (HDFS) 接続を使用して、Hadoop クラスタ内のデータにアクセスします。HDFS 接続は、ファイルシステムタイプの接続です。HDFS 接続は、Administrator ツール、Analyst ツール、または Developer ツールで作成および管理できます。HDFS 接続プロパティは、特に明記されている場合を除き、大文字と小文字が区別されます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、HDFS 接続のプロパティを示します。

プロパティ	説明
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p><code>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</code></p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、765 文字を超えることはできません。</p>
場所	<p>接続を作成するドメイン。Analyst ツールでは使用できません。</p>
タイプ	<p>接続タイプ。デフォルトは Hadoop ファイルシステムです。</p>

プロパティ	説明
ユーザー名	HDFS にアクセスするためのユーザー名。
NameNode の URI	<p>HDFS にアクセスするための URI。</p> <p>Cloudera および Hortonworks ディストリビューションでは、以下の形式を使用して NameNode URI を指定します。</p> <pre>hdfs://<namenode>:<port></pre> <p>説明</p> <ul style="list-style-type: none"> - <namenode>は、NameNode のホスト名または IP アドレスです。 - <port>は、NameNode がリモートプロシージャコール（RPC）をリスンするポートです。 <p>MapR ディストリビューションの場合は、次のいずれかの形式を使用して NameNode URI を指定します。</p> <ul style="list-style-type: none"> - maprfs:/// - maprfs:///mapr/my.cluster.com/ <p>my.cluster.com は、mapr-clusters.conf ファイルに指定したクラスタ名です。</p>

Hive 接続のプロパティ

Hive 接続を利用して、Hive データにアクセスします。Hive 接続は、データベースタイプの接続です。Hive 接続は、Administrator ツール、Analyst ツール、または Developer tool で作成および管理できます。Hive 接続プロパティは、特に明記されている場合を除き、大文字と小文字が区別されます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Hive 接続のプロパティを示します。

プロパティ	説明
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。このプロパティは、接続を作成した後に変更できます。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <pre>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</pre>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、4000 文字を超えることはできません。</p>
場所	<p>接続を作成するドメイン。Analyst ツールでは使用できません。</p>
タイプ	<p>接続タイプ。Hive を選択します。</p>

プロパティ	説明
接続モード	<p>Hive 接続モード。以下のオプションのうちの 1 つ以上を選択します。</p> <ul style="list-style-type: none"> - HiveServer2 にアクセスしてマッピングを実行。このオプションは、Hive データウェアハウスにアクセスするために接続を使用する場合に選択します。Hive をターゲットとして使用する場合は、同じ接続または別の Hive 接続を有効にして Hadoop クラスタでマッピングを実行する必要があります。 - Hive CLI にアクセスしてマッピングを実行。このオプションは、Hadoop クラスタでマッピングを実行するために Hive CLI を使用する場合に選択します。 <p>両方のオプションを選択できます。デフォルトは、[ソースまたはターゲットとして、Hive にアクセス] です。</p>

プロパティ	説明
ユーザー名	<p>Hadoop クラスタでマッピングを実行するためにデータ統合サービスが偽装するユーザーのユーザー名。ユーザー名は、ネイティブ環境の [メタデータ接続文字列] または [データアクセス接続文字列] に指定する JDBC 接続文字列によって決まります。</p> <p>Hadoop クラスタで Hortonworks HDP を実行する場合、ユーザー名を指定する必要があります。Tez を使用してマッピングを実行する場合、データ統合サービスのユーザーアカウントを指定する必要があります。Tez を使用してマッピングを実行しない場合、偽装ユーザーアカウントを使用できます。</p> <p>Hadoop クラスタが Kerberos 認証を使用する場合、JDBC 接続文字列のプリンシパル名とこのユーザー名が同じである必要があります。Hadoop クラスタが Kerberos 認証を使用しない場合、ユーザー名は JDBC ドライバの動作によって異なります。Hive JDBC ドライバを使用すると、さまざまな方法でユーザー名を指定することができ、そのユーザー名を JDBC URL の一部にすることができます。</p> <p>Hadoop クラスタが Kerberos 認証を使用しない場合、ユーザー名は JDBC ドライバの動作によって決まります。</p> <p>ユーザー名を指定しないと、Hadoop クラスタは次の基準に基づいてジョブを認証します。</p> <ul style="list-style-type: none"> - Hadoop クラスタが Kerberos 認証を使用しない。データ統合サービスを実行するマシンのオペレーティングシステムのプロファイルユーザー名に基づいてジョブを認証します。 - Hadoop クラスタが Kerberos 認証を使用する。ジョブの認証は、データ統合サービスの SPN に基づいて行います。
両方のモードに共通する属性: 環境 SQL	<p>Hadoop 環境を設定する SQL コマンド。ネイティブ環境タイプでは、データ統合サービスは Hive メタストアへの接続を作成するたびに環境 SQL を実行します。Hadoop クラスタでのプロファイルの実行に Hive 接続を使用すると、データ統合サービスは各 Hive セッションの最初に環境 SQL を実行します。</p> <p>どちらの接続モードでも、環境 SQL の使用には以下のルールとガイドラインが適用されます。</p> <ul style="list-style-type: none"> - 環境 SQL は、Hive クエリを指定するために使用します。 - 環境 SQL を使用して Hive ユーザー定義関数のクラスパスを設定し、続いて環境 SQL または PreSQL を使用して Hive ユーザー定義関数を指定します。データオブジェクトプロパティで PreSQL を使用してクラスパスを指定することはできません。パスは、ユーザー定義関数に使用される JAR ファイルの完全修飾パスでなければなりません。 infapdo.aux.jars.path にすべてのエントリを指定したパラメータ hive.aux.jars.path およびパスをユーザー定義関数の JAR ファイルに設定します。 - 環境 SQL は、PreSQL コマンド内またはカスタムクエリ内で使用する Hadoop パラメータまたは Hive パラメータを定義するために使用できません。 - 環境 SQL プロパティに複数の値を使用する場合は、値の間にスペースを含めないようにしてください。次のサンプルテキストは、環境 SQL プロパティに使用できる 2 つの値を示しています。 set hive.execution.engine='tez';set hive.exec.dynamic.partition.mode='nonstrict'; <p>Hadoop クラスタでのプロファイルの実行に Hive 接続を使用すると、データ統合サービスは Hive 接続の環境 SQL のみを実行します。Hive ソースとターゲットが別のクラスタにある場合、データ統合サービスは Hive ソースまたはターゲットの接続に対して異なる環境 SQL コマンドを実行しません。</p>

ソースまたはターゲットとして Hive にアクセスするためのプロパティ

以下の表に、ソースまたはターゲットとして Hive にアクセスするために設定する接続プロパティを示します。

プロパティ	説明
メタデータ 接続文字列	<p>Hadoop サーバーからメタデータにアクセスするために使用される JDBC 接続 URI。 PowerExchange for Hive を使用して、HiveServer サービスまたは HiveServer2 サービスと通信できます。</p> <p>HiveServer に接続するには、次の形式で接続文字列を指定します。 <code>jdbc:hive2://<hostname>:<port>/<db></code></p> <p>ここで</p> <ul style="list-style-type: none">- <hostname>は、HiveServer2 が実行されているマシンの名前または IP アドレスです。- <port>は、HiveServer2 がリスンするポート番号です。- <db>は、接続先のデータベース名です。データベース名を指定しないと、データ統合サービスはデフォルトのデータベースの詳細を使用します。 <p>HiveServer 2 に接続するには、その特定の Hadoop ディストリビューション用に Apache Hive が実装する接続文字列形式を使用します。Apache Hive 接続文字列形式の詳細については、Apache Hive のドキュメントを参照してください。</p>
Hive JDBC サーバーを バイパス	<p>JDBC ドライバモード。埋め込み JDBC ドライバモードを使用するには、チェックボックスを選択します。</p> <p>JDBC 埋め込みモードを使用するには、以下のタスクを実行します。</p> <ul style="list-style-type: none">- Hive クライアントと Informatica サービスが同一のマシンにインストールされていることを確認します。- Hadoop クラスタでマッピングを実行するように Hive 接続プロパティを設定します。 <p>非埋め込みモードを選択する場合は、データアクセス接続文字列を設定する必要があります。</p> <p>JDBC 埋め込みモードを使用することをお勧めします。</p>
データアクセス接続文字列	<p>Hadoop データストアからデータにアクセスするための接続文字列。</p> <p>HiveServer に接続するには、次の形式で非埋め込み JDBC モードの接続文字列を指定します。 <code>jdbc:hive2://<hostname>:<port>/<db></code></p> <p>ここで</p> <ul style="list-style-type: none">- <hostname>は、HiveServer2 が実行されているマシンの名前または IP アドレスです。- <port>は、HiveServer2 がリスンするポート番号です。- <db>は、接続先のデータベースです。データベース名を指定しないと、データ統合サービスはデフォルトのデータベースの詳細を使用します。 <p>HiveServer 2 に接続するには、特定の Hadoop ディストリビューション用に Apache Hive が実装する接続文字列形式を使用します。Apache Hive 接続文字列形式の詳細については、Apache Hive のドキュメントを参照してください。</p>

Hadoop クラスタでマッピングを実行するためのプロパティ

以下の表に、Hive 接続を使用して Hadoop クラスタで Informatica マッピングを実行する場合に設定する Hive 接続プロパティを示します。

プロパティ	説明
データベース名	テーブルの名前空間。指定されたデータベース名を持たないテーブルには、名前として default を使用してください。
デフォルト FS の URI	<p>デフォルトの Hadoop 分散ファイルシステムにアクセスするための URI。次の接続 URI を使用します。 hdfs://<node name>:<port></p> <p>ここで</p> <ul style="list-style-type: none">- <node name>は、NameNode のホスト名または IP アドレスです。- <port>は、NameNode がリモートプロシージャコール (RPC) をリスンするポートです。 <p>Hadoop クラスタが MapR を実行する場合、次の URI を使用して MapR のファイルシステムにアクセスします。maprfs:///</p>
JobTracker/Yarn リソースマネージャ URI	<p>クラスタ内の特定のノードに MapReduce タスクを送信する、Hadoop 内のサービス。</p> <p>次の形式を使用します。 <hostname>:<port></p> <p>ここで</p> <ul style="list-style-type: none">- <hostname>は、JobTracker または Yarn リソースマネージャのホスト名または IP アドレスです。- <port>は、JobTracker または Yarn リソースマネージャがリモートプロシージャコール (RPC) をリスンするポートです。 <p>クラスタユーザーが YARN で MapR を使用する場合、yarn-site.xml の yarn.resourcemanager.address で指定された値を使用します。クラスタの NameNode の次のディレクトリに yarn-site.xml があります。/opt/mapr/hadoop/hadoop-2.5.1/etc/hadoop</p> <p>MapR with MapReduce 1 では可用性の高い JobTracker をサポートします。MapR 配布を使用している場合は、JobTracker URI を maprfs:///形式で定義します。</p>
HDFS 上の Hive ウェアハウスディレクトリ	<p>クラスタに対してローカルなウェアハウスの、デフォルトデータベースの HDFS ファイルの絶対パスです。例えば、以下のファイルパスはローカルウェアハウスを指定します。 /user/hive/warehouse</p> <p>Cloudera CDH で、Metastore Execution Mode がリモートの場合は、ファイルパスが、Hadoop クラスタの Hive メタストアサービスによって指定されたファイルパスと一致する必要があります。</p> <p>MapR の場合、hive-site.xml の hive.metastore.warehouse.dir プロパティに指定された値を使用します。HiveServer2 が実行されるノード上の次のディレクトリに hive-site.xml があります。/opt/mapr/hive/hive-0.13/conf</p>

プロパティ	説明
Hive/Hadoop の詳細プロパティ	<p>データ統合サービスが実行されているマシンで、Hive または Hadoop クラスタのプロパティを設定またはオーバーライドします。複数のプロパティを指定できます。</p> <p>【編集】を選択してプロパティの名前と値を指定します。プロパティは次の形式で表示されます。</p> <pre><property1>=<value></pre> <p>ここで</p> <ul style="list-style-type: none"> - <property1>は、hive-site.xml での Hive または Hadoop プロパティです。 - <value>は、Hive または Hadoop プロパティの値です。 <p>複数のプロパティを指定する場合、&:がプロパティのセパレータとして表示されます。</p> <p>この形式の最大長は 1MB です。</p> <p>Hive 接続の必須プロパティを入力すると、このプロパティによって Hive/Hadoop の詳細プロパティで設定したプロパティがオーバーライドされます</p> <p>データ統合サービスは、map-reduce ジョブごとにこれらのプロパティを追加または設定します。これらのプロパティは、各 mapper および reducer ジョブの JobConf で確認できます。各ジョブの JobConf には、各 map-reduce ジョブの Jobtracker URL からアクセスできます。</p> <p>データ統合サービスは、これらのプロパティのメッセージをデータ統合サービスのログに書き込みます。データ統合サービスで、ログのトレースレベルを各行をログに記録するように設定するか、ログのトレースレベルを Verbose Initialization トレースに設定する必要があります。</p> <p>例えば、マッピングジョブを実行する reducer の数を制御および制限するには、次のプロパティを指定します。</p> <pre>mapred.reduce.tasks=2&:hive.exec.reducers.max=10</pre>
一時テーブル圧縮コーデック	圧縮コーデッククラス名の Hadoop 圧縮ライブラリ。
コーデッククラス名	データ圧縮を有効にし、一時ステージングテーブルのパフォーマンスを改善するコーデッククラス名。
メタストア実行モード	リモートメタストアとローカルメタストアのどちらに接続するかを制御します。デフォルトではローカルが選択されます。ローカルメタストアの場合、メタストアデータベースの URI、ドライバ、ユーザー名、およびパスワードを指定する必要があります。リモートメタストアの場合、指定する必要があるのはリモートメタストアの URI だけです。
メタストアデータベースの URI	<p>ローカルメタストアセットアップ内のデータストアにアクセスするために使用される JDBC 接続の URI。次の接続 URI を使用します。</p> <pre>jdbc:<datastore type>://<node name>:<port>/<database name></pre> <p>ここで</p> <ul style="list-style-type: none"> - <node name>は、データストアのホスト名または IP アドレスです。 - <data store type>は、データストアのタイプです。 - <port>は、データストアがリモートプロシージャコール (RPC) をリスンするポートです。 - <database name>は、データベースの名前です。 <p>例えば、以下の URI はデータストアとして MySQL を使用するローカルメタストアを指定します。</p> <pre>jdbc:mysql://hostname23:3306/metastore</pre> <p>MapR の場合、hive-site.xml の javax.jdo.option.ConnectionURL プロパティに設定されている値を使用します。HiveServer 2 が実行されるノード上の次のディレクトリに hive-site.xml はあります。/opt/mapr/hive/hive-0.13/conf</p>

プロパティ	説明
メタストアデータベースドライバ	JDBC データストアのドライバクラス名。例えば、以下のクラス名は MySQL ドライバを指定します。 com.mysql.jdbc.Driver MapR の場合、hive-site.xml の javax.jdo.option.ConnectionDriverName プロパティに指定された値を使用します。HiveServer 2 が実行されるノード上の次のディレクトリに hive-site.xml はあります。/opt/mapr/hive/hive-0.13/conf
メタストアデータベースユーザー名	メタストアデータベースユーザー名。 MapR の場合、hive-site.xml の javax.jdo.option.ConnectionUserName プロパティに指定された値を使用します。HiveServer 2 が実行されるノード上の次のディレクトリに hive-site.xml はあります。/opt/mapr/hive/hive-0.13/conf
メタストアデータベースパスワード	メタストアユーザー名のパスワード。 MapR の場合、hive-site.xml の javax.jdo.option.ConnectionPassword プロパティに指定された値を使用します。HiveServer 2 が実行されるノード上の次のディレクトリに hive-site.xml はあります。/opt/mapr/hive/hive-0.13/conf
リモートメタストアの URI	リモートメタストアセットアップ内のメタデータにアクセスするために使用されるメタストアの URI。リモートメタストアの場合、Thrift サーバーの詳細を指定する必要があります。 次の接続 URI を使用します。 thrift://<hostname>:<port> ここで - <hostname>は、Thrift メタストアサーバーの名前または IP アドレスです。 - <port>は、Thrift サーバーがリスンしているポートです。 MapR の場合、hive-site.xml の hive.metastore.uris プロパティに指定された値を使用します。HiveServer 2 が実行されるノード上の次のディレクトリに hive-site.xml はあります。/opt/mapr/hive/hive-0.13/conf

HTTP 接続プロパティ

REST Web サービスコンシューマトランスフォーメーションを Web サービスに接続するには、HTTP 接続を使用します。HTTP 接続は、Web タイプの接続です。HTTP 接続は Developer ツールで作成します。HTTP 接続は、Administrator ツールまたは Developer ツールで管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、HTTP 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
ユーザー名	Web サービスに接続するユーザー名。HTTP 認証または WS-Security を有効にした場合は、ユーザー名を入力します。 Web サービスコンシューマトランスフォーメーションに WS-Security ポートが含まれている場合、Web サービスコンシューマトランスフォーメーションは入力ポートを介して動的ユーザー名を受け取ります。データ統合サービスは、接続で定義されているユーザー名をオーバーライドします。
パスワード	ユーザー名に対するパスワード。HTTP 認証または WS-Security を有効にした場合は、パスワードを入力します。 Web サービスコンシューマトランスフォーメーションに WS-Security ポートが含まれている場合、Web サービスコンシューマトランスフォーメーションは入力ポートを介して動的パスワードを受け取ります。データ統合サービスは、接続で定義されているパスワードをオーバーライドします。
エンドポイント URL	アクセス先の Web サービスの URL。データ統合サービスは、WSDL ファイルで定義されている URL をオーバーライドします。 Web サービスコンシューマトランスフォーメーションにエンドポイント URL ポートが含まれている場合、Web サービスコンシューマトランスフォーメーションは入力ポートを介して URL を動的に受け取ります。データ統合サービスは、接続で定義されている URL をオーバーライドします。
タイムアウト	データ統合サービスが、接続を閉じる前に、Web サービスプロバイダからの応答を待機する秒数。
HTTP 認証タイプ	HTTP 経由のユーザー認証のタイプ。次のいずれかの値を選択します。 <ul style="list-style-type: none"> - なし: 認証しません。 - 自動: データ統合サービスが Web サービスプロバイダの認証タイプを選択します。 - 基本: Web サービスプロバイダのドメインに対するユーザー名およびパスワードの入力が要求されます。データ統合サービスは、認証のためにユーザー名とパスワードを Web サービスプロバイダに送ります。 - ダイジェスト: Web サービスプロバイダのドメインに対するユーザー名およびパスワードの入力が要求されます。データ統合サービスは、ユーザー名とパスワードから暗号化されたメッセージダイジェストを生成し、それを Web サービスプロバイダに送ります。プロバイダはユーザー名とパスワードの一時的な値を生成し、それをドメインコントローラの Active Directory に格納します。その値とメッセージダイジェストが比較されます。これらが一致する場合、Web サービスプロバイダはユーザを認証します。 - NTLM: ドメイン名、サーバー名、またはデフォルトのユーザー名とパスワードの入力が要求されます。Web サービスプロバイダは、接続先のドメインに基づいてユーザを認証します。Windows Domain Controller からユーザー名とパスワードが取得され、ユーザが入力したユーザー名とパスワードと比較されます。これらが一致する場合、Web サービスプロバイダはユーザを認証します。NTLM 認証では、暗号化されたパスワードがドメインコントローラのアクティブディレクトリに保存されることはありません。

プロパティ	説明
トラストサーティフィケーションファイル	Web サービスの SSL 証明書の認証時にデータ統合サービスが使用するトラストサーティフィケーションのバンドルを含むファイル。ファイル名と完全ディレクトリパスを入力します。 デフォルトは、<Informatica installation directory>/services/shared/bin/ca-bundle.crt です。
クライアント証明書ファイル名	クライアントの認証時に Web サービスで使用されるクライアント証明書。Web サービスがデータ統合サービスを認証する必要がある場合にクライアントサーティフィケーションファイルを指定します。
クライアント証明書のパスワード	クライアントサーティフィケーションのパスワード。Web サービスがデータ統合サービスを認証する必要がある場合にクライアント証明書のパスワードを指定します。
クライアント証明書のタイプ	クライアントサーティフィケーションファイルの形式。次のいずれかの値を選択します。 - PEM: 「.pem」 拡張子のファイルです。 - DER: 「.cer」 または 「.der」 拡張子のファイルです。 Web サービスがデータ統合サービスを認証する必要がある場合にクライアント証明書のタイプを指定します。
プライベートキーファイル名	クライアントサーティフィケーションのプライベートキーファイル。Web サービスがデータ統合サービスを認証する必要がある場合にプライベートキーファイルを指定します。
プライベートキーのパスワード	クライアントサーティフィケーションのプライベートキーのパスワード。Web サービスがデータ統合サービスを認証する必要がある場合にプライベートキーのパスワードを指定します。
プライベートキーのタイプ	プライベートキーのタイプ。サポートされているタイプは PEM です。

IBM DB2 接続のプロパティ

IBM DB2 にアクセスするには、IBM DB2 接続を使用します。IBM DB2 接続は、リレーショナルデータベース接続です。IBM DB2 接続は、Administrator ツール、Developer ツール、または Analyst ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、DB2 接続のプロパティを示します。

プロパティ	説明
データベースタイプ	データベースのタイプ。
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /

プロパティ	説明
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
ユーザー名	データベースユーザー名。
パスワード	データベースユーザー名のパスワード。
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。接続のパススルーセキュリティを有効にすると、ドメインは、接続オブジェクトに定義されている資格情報ではなく、クライアントのユーザー名とパスワードを使用して対応するデータベースにログインします。
データアクセス接続文字列	データベースのメタデータへのアクセスに使用する DB2 接続 URL。 dbname dbname は、DB2 クライアントで設定されているエイリアスです。
メタデータアクセスプロパティ: 接続文字列	<p>次のメタデータ接続文字列 URL を使用します。</p> <pre>jdbc:informatica:db2://<host name>:<port>;DatabaseName=<database name></pre> <p>テーブルをインポートすると、デフォルトでは、デフォルトスキーマ名の下にすべてのテーブルが表示されます。デフォルトスキーマではなく特定のスキーマの下にテーブルを表示するには、テーブルをインポートするスキーマ名を指定します。スキーマ名を指定するには、URL に ischemaname パラメータを追加します。例えば、特定のスキーマからテーブルをインポートするには、次の構文を使用します。</p> <pre>jdbc:informatica:db2://<host name>:<port>;DatabaseName=<database name>;ischemaname=<schema_name></pre> <p>複数のスキーマからテーブルを検索してインポートするには、ischemaname パラメータに複数のスキーマ名を指定します。スキーマ名では大文字小文字が区別されます。複数のスキーマ名を指定する場合、特殊文字は使用できません。複数のスキーマ名を区切るには、パイプ () 文字を使用します。例えば、3 つのスキーマからテーブルを検索してインポートするには、次の構文を使用します。</p> <pre>jdbc:informatica:db2://<host name>:<port>;DatabaseName=<database name>;ischemaname=<schema_name1> <schema_name2> <schema_name3></pre> <p>複数のスキーマ名を指定する場合、指定したスキーマ名の下にテーブルを表示するには 【デフォルトのスキーマだけを表示する】 オプションの選択を解除する必要があります。</p>

プロパティ	説明
AdvancedJDBCSecurityOptions	<p>セキュアデータベースへのメタデータアクセスのためのデータベースパラメータ。Informatica は、AdvancedJDBCSecurityOptions フィールドの値を機密データとして扱い、パラメータ文字列を暗号化して格納します。</p> <p>セキュアデータベースに接続するには、以下のパラメータを含めます。</p> <ul style="list-style-type: none"> - EncryptionMethod。必須。ネットワーク上で送信される際にデータが暗号化されるかどうかを示します。このパラメータは SSL に設定する必要があります。 - ValidateServerCertificate。オプション。データベースサーバーによって送信される証明書を Informatica が検証するかどうかを示します。 このパラメータを True に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証します。HostNameInCertificate パラメータを指定すると、Informatica は証明書内のホスト名も検証します。 このパラメータを false に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証しません。指定するトラストストア情報がすべて無視されます。 - HostNameInCertificate。オプション。セキュアデータベースをホストするマシンのホスト名。ホスト名を指定すると、Informatica は接続文字列に含められたそのホスト名を SSL 証明書内のホスト名と照らして検証します。 - cryptoProtocolVersion。オプション。IBM DB2 インスタンスに対して TLS を有効にする場合は、次のように cryptoProtocolVersion パラメータを設定します。 cryptoProtocolVersion=TLSv<version number> 例: cryptoProtocolVersion=TLSv1.2 注: バージョン番号はサーバーに設定した TLS バージョンと同じにする必要があります。 - TrustStore。必須。トラストストアファイルのパスおよびファイル名です。 注: SSL または TLS を設定してファイル名のみを指定する場合は、トラストストアファイルを Informatica のインストールディレクトリにコピーする必要があります。接続をテストしメタデータをインポートするには、トラストストアファイルを次のディレクトリにコピーします。 <Informatica client installation directory>/clients/DeveloperClient マッピングを実行するには、トラストストアファイルを次のディレクトリにコピーします。 <Informatica server installation directory>/tomcat/bin 注: SSL または TLS を設定してファイル名のみを指定する場合は、トラストストアファイルを次のディレクトリにコピーして接続をテストする必要があります。<Informatica server installation directory>/tomcat/bin - TrustStorePassword。必須。セキュアデータベースに対するトラストストアファイルのパスワード。 注: セキュア JDBC パラメータは接続文字列に自動的に追加されます。セキュア JDBC パラメータを接続文字列に直接指定する場合は、AdvancedJDBCSecurityOptions フィールドにはパラメータを入力しないでください。
データアクセスプロパティ: 接続文字列	<p>データベースのデータへのアクセスに使用する接続文字列。</p> <p>IBM DB2 の場合、これは<database name>です</p>
コードページ	<p>ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。</p>

プロパティ	説明
環境 SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。
トランザクション SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、トランザクション環境 SQL を各トランザクションの開始時に実行します。
リトライ期限	このプロパティは、将来使用するために予約されています。
テーブルスペース	データベースのテーブルスペース名。
SQL 識別子文字	データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。 データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。 データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。
大文字と小文字が混在する識別子をサポート	データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を 【SQL 識別子文字】 プロパティに対して選択された文字で囲みます。 【SQL 識別子文字】 プロパティが [なし] に設定されている場合、 【大文字と小文字が混在する識別子をサポート】 プロパティは無効になっています。
ODBC プロバイダ	ODBC。ODBC を接続するデータベースのタイプ。プッシュダウンの最適化用に、データベースタイプを指定して、データ統合サービスでネイティブデータベース SQL を生成できるようにします。次のオプションがあります。 - その他 - Sybase - Microsoft_SQL_Server デフォルトは 【その他】 です。

IBM DB2 for i5/OS 接続プロパティ

IBM DB2 for i5/OS 接続を使用し、IBM DB2 for i5/OS 内のテーブルにアクセスします。IBM DB2 for i5/OS 接続はリレーショナルデータベース接続です。IBM DB2 for i5/OS 接続は、Administrator ツールまたは Developer ツールで管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、DB2 for i5/OS 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、255 文字を超えることはできません。
接続タイプ	接続タイプ (DB2I)。
ユーザー名	データベースユーザー名。
パスワード	指定したユーザー名のパスワードまたは有効な PowerExchange のパスフレーズ。 PowerExchange のパスフレーズは 9~31 文字の範囲で作成します。使用できる文字は次のとおりです。 <ul style="list-style-type: none"> - 英字 (大文字と小文字) - 数字 (0~9) - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符 (')、二重引用符 (")、または通貨記号を含めることはできません。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p>
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。
データベース名	データベースインスタンス名。
場所	DB2 に接続する PowerExchange リスナの場所のノード名。ノード名は、PowerExchange の設定ファイル (dbmover.cfg) 内の NODE 文の最初のパラメータで定義されます。
環境 SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。

プロパティ	説明
データベースファイルのオーバーライド	<p>i5/OS のデータベースファイルオーバーライドを次の形式で指定します。</p> <p>from_file/to_library/to_file/to_member</p> <p>説明：</p> <ul style="list-style-type: none"> - from_file はオーバーライドされるファイルです。 - to_library は新しく使用するライブラリです。 - to_file は新しく使用するライブラリ内のファイルです。 - to_member はオプションで、新しく使用するライブラリおよびファイル内のメンバです。何も指定されない場合、*FIRST が使用されます。 <p>単一の接続において 8 回まで、一意のファイルオーバーライドを指定することができます。1 つのオーバーライドは 1 つのソースまたはターゲットに適用されます。複数のファイルオーバーライドを指定する場合は、ファイルオーバーライドの文字列を二重引用符 (") で囲み、各ファイルオーバーライドの間にスペースを入れます。</p> <p>注: 【ライブラリリスト】 と 【データベースファイルのオーバーライド】 の両方を指定していて、テーブルにその両方が存在する場合、【データベースファイルのオーバーライド】 の値が優先されます。</p>
ライブラリリスト	<p>Select、Insert、Delete、または Update 文のテーブル名を修飾するために PowerExchange が検索するライブラリのリスト。テーブル名が修飾されていない場合、PowerExchange はリストを検索します。</p> <p>ライブラリはセミコロンで区切ります。</p> <p>注: 【ライブラリリスト】 と 【データベースファイルのオーバーライド】 の両方を指定していて、テーブルにその両方が存在する場合、【データベースファイルのオーバーライド】 の値が優先されます。</p>
コードページ	<p>ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。</p>
使用する SQL 識別子文字	<p>データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。</p> <p>データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。</p> <p>データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。</p>
大文字と小文字が混在する識別子をサポート	<p>データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を 【SQL 識別子文字】 プロパティに対して選択された文字で囲みます。</p> <p>【SQL 識別子文字】 プロパティが [なし] に設定されている場合、【大文字と小文字が混在する識別子をサポート】 プロパティは無効になっています。</p>
分離レベル	<p>トランザクションのコミット範囲。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - CS。カーソルの安定性。 - RR。繰り返し可能な読み込み。 - CHG。変更。 - ALL <p>デフォルトは CS です。</p>

プロパティ	説明
暗号化タイプ	<p>オプション。データ統合サービスが使用する暗号化のタイプ。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - RC2 - DES <p>デフォルトは [なし] です。</p> <p>注:</p> <ul style="list-style-type: none"> - Informatica では、暗号化タイプおよびレベル接続プロパティを設定したり、DBMOVER 構成ファイルに ENCRYPT 文と ENCRYPTLEVEL 文を設定する代わりに、Secure Sockets Layer (SSL) 認証を使用することを推奨します。SSL 認証は複数の Informatica 製品で使用されており、より厳格なセキュリティが提供されます。 <p>PowerExchange ネットワークへの SSL 認証の実装の詳細については、『<i>PowerExchange リファレンスマニュアル</i>』を参照してください。</p> <ul style="list-style-type: none"> - ENCRYPT および ENCRYPTLEVEL 文の値が統合サービスマシンの DBMOVER 構成ファイルで定義されている場合、暗号化タイプおよびレベル接続属性に選択する値によって、これらの値はオーバーライドされます。マッピングの暗号化を有効にするには、必ず該当する接続属性を選択します。
暗号化レベル	<p>[暗号タイプ] で [RC2] または [DES] を選択した場合は、データ統合サービスが使用する暗号化レベルを指定するために、次に挙げるオプションのいずれかを選択してください。</p> <ul style="list-style-type: none"> - 1. DES および RC2 には 56 ビットの暗号化キーを使用します。 - 2. DES には 168 ビットのトリプル暗号化キーを、RC2 には 64 ビットの暗号化キーを使用します。 - 3. DES には 168 ビットのトリプル暗号化キーを、RC2 には 128 ビットの暗号化キーを使用します。 <p>暗号化タイプを選択しない場合、このオプションは無視されます。</p> <p>デフォルトは 1 です。</p>
ページングサイズ	<p>オプション。ソースシステムが PowerExchange リスナに渡すことができるデータ量。ページングサイズは、外部アプリケーション、データベース、またはデータ統合サービスのノードがボトルネックになっている場合に設定します。低い値を使用するとパフォーマンスが向上します。最小値およびデフォルト値は 0 です。値を 0 にするとパフォーマンスが最大になります。</p>
行として解釈	<p>オプション。このオプションは、ページングサイズを行数で表示する場合に選択します。このオプションをクリアすると、ページングサイズがキロバイト単位で表示されます。デフォルトではこのオプションが選択されていないため、ページングサイズはキロバイトで表示されています。</p>
圧縮	<p>オプション。このオプションは、ソースデータの圧縮を有効にする場合に選択します。データを圧縮することで、Informatica アプリケーションがネットワークで送信するデータ量を減少させることができます。デフォルトではこのオプションが選択されていないため、圧縮は無効になっています。</p>
配列サイズ	<p>オプション。ワーカースレッドのためのストレージアレイのレコード数。このオプションは、[ワーカースレッド] オプションを 0 より大きな値に設定したときに適用されます。有効な値は 25~100000 です。デフォルトは 25 です。</p>

プロパティ	説明
書き込みモード	<p>オプション。データ統合サービスが PowerExchange リスナにデータを送信するモード。次のうちのいずれかの書き込みモードを選択します。</p> <ul style="list-style-type: none"> - CONFIRMWRITEON。PowerExchange リスナにデータを送信し、応答を待ってから以降のデータを送信します。このオプションはエラーリカバリを優先する場合に選択します。ただし、このオプションを使用するとパフォーマンスが落ちる可能性があります。 - CONFIRMWRITEOFF。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションは、エラーが発生したときにターゲットテーブルを再ロードしてもよい場合に使用します。 - ASYNCHRONOUSWITHFAULTTOLERANCE。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションはエラー検出も有効にします。このオプションは、CONFIRMWRITEOFF のスピードと CONFIRMWRITEON のデータの完全性を結合します。 <p>デフォルトは [CONFIRMWRITEON] です。</p>
拒否ファイル	<p>拒否ファイルの PWXR のデフォルトプレフィックスをオーバーライドします。書き込みモードが ASYNCHRONOUSWITHFAULTTOLERANCE のときは、PowerExchange は拒否ファイルをターゲットマシンに作成します。拒否ファイルを作成しないようにするには、PWXDISABLE と入力します。</p>

IBM DB2 for z/OS 接続のプロパティ

IBM DB2 for z/OS 接続を使用し、IBM DB2 for z/OS 内のテーブルにアクセスします。IBM DB2 for z/OS 接続はリレーショナルデータベース接続です。IBM DB2 for z/OS 接続は、Administrator ツールまたは Developer ツールで管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、DB2 for z/OS 接続のプロパティを示します。

プロパティ	説明
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、255 文字を超えることはできません。</p>
接続タイプ	<p>接続タイプ (DB2Z)。</p>
ユーザー名	<p>データベースユーザー名。</p>

プロパティ	説明
パスワード	<p>指定したユーザー名のパスワードまたは有効な PowerExchange のパスフレーズ。 PowerExchange パスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次を示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。 パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p> <p>IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: 有効な RACF のパスフレーズは最長 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p>
パススルーセキュリティ有効	<p>接続のパススルーセキュリティを有効にする。</p>
DB2 サブシステム ID	<p>DB2 サブシステムの名前。</p>
場所	<p>DB2 に接続する PowerExchange リスナの場所のノード名。ノード名は、PowerExchange の設定ファイル（dbmover.cfg）内の NODE 文の最初のパラメータで定義されます。</p>
環境 SQL	<p>データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。</p>
相関 ID	<p>DB2 要求用の DB2 相関 ID を形成するために、PWX のプレフィックスと連結される値。</p>
コードページ	<p>ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。</p>
使用する SQL 識別子文字	<p>データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。</p> <p>データベースで通常識別子を使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。</p> <p>データベースで区切り識別子を使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。</p>
大文字と小文字が混在する識別子をサポート	<p>データベースで大文字と小文字を区別する識別子を使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を【SQL 識別子文字】プロパティに対して選択された文字で囲みます。</p> <p>【SQL 識別子文字】プロパティが [なし] に設定されている場合、【大文字と小文字が混在する識別子をサポート】プロパティは無効になっています。</p>

プロパティ	説明
暗号化タイプ	<p>オプション。データ統合サービスが使用する暗号化のタイプ。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - RC2 - DES <p>デフォルトは [なし] です。</p> <p>注:</p> <ul style="list-style-type: none"> - Informatica では、暗号化タイプおよびレベル接続プロパティを設定したり、DBMOVER 構成ファイルに ENCRYPT 文と ENCRYPTLEVEL 文を設定する代わりに、Secure Sockets Layer (SSL) 認証を使用することを推奨します。SSL 認証は複数の Informatica 製品で使用されており、より厳格なセキュリティが提供されます。 <p>PowerExchange ネットワークへの SSL 認証の実装の詳細については、『<i>PowerExchange リファレンスマニュアル</i>』を参照してください。</p> <ul style="list-style-type: none"> - ENCRYPT および ENCRYPTLEVEL 文の値が統合サービスマシンの DBMOVER 構成ファイルで定義されている場合、暗号化タイプおよびレベル接続属性に選択する値によって、これらの値はオーバーライドされます。マッピングの暗号化を有効にするには、必ず該当する接続属性を選択します。
暗号化レベル	<p>[暗号タイプ] で [RC2] または [DES] を選択した場合は、データ統合サービスが使用する暗号化レベルを指定するために、次に挙げるオプションのいずれかを選択してください。</p> <ul style="list-style-type: none"> - 1. DES および RC2 には 56 ビットの暗号化キーを使用します。 - 2. DES には 168 ビットのトリプル暗号化キーを、RC2 には 64 ビットの暗号化キーを使用します。 - 3. DES には 168 ビットのトリプル暗号化キーを、RC2 には 128 ビットの暗号化キーを使用します。 <p>暗号化タイプを選択しない場合、このオプションは無視されます。</p> <p>デフォルトは 1 です。</p>
ページングサイズ	<p>オプション。ソースシステムが PowerExchange リスナに渡すことができるデータ量。ページングサイズは、外部アプリケーション、データベース、またはデータ統合サービスのノードがボトルネックになっている場合に設定します。低い値を使用するとパフォーマンスが向上します。</p> <p>最小値およびデフォルト値は 0 です。値を 0 にするとパフォーマンスが最大になります。</p>
行として解釈	<p>オプション。このオプションは、ページングサイズを行数で表示する場合に選択します。このオプションをクリアすると、ページングサイズがキロバイト単位で表示されます。デフォルトではこのオプションが選択されていないため、ページングサイズはキロバイトで表示されています。</p>
圧縮	<p>オプション。このオプションは、ソースデータの圧縮を有効にする場合に選択します。データを圧縮することで、Informatica アプリケーションがネットワークで送信するデータ量を減少させることができます。デフォルトではこのオプションが選択されていないため、圧縮は無効になっています。</p>
オフロード処理	<p>オプション。バルクデータの処理をソースマシンからデータ統合サービスのマシンにオフロードするかどうかを制御します。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - AUTO。オフロード処理を使用するかどうかデータ統合サービスによって決定されます。 - はい。オフロード処理を使用します。 - いいえ。オフロード処理は使用しません。 <p>デフォルトは [いいえ] です。</p>
ワーカースレッド	<p>オプション。オフロード処理の有効時にデータ統合サービスがバルクデータを処理するために使用するスレッドの数。最適なパフォーマンスを得るためには、データ統合サービスマシンで使用可能なプロセッサの数を超えないようにこの値を設定します。有効な値は 1~64 です。デフォルトは 0 です。マルチスレッド処理は無効になります。</p>

プロパティ	説明
配列サイズ	オプション。ワーカースレッドのためのストレージアレイのレコード数。このオプションは、 【ワーカースレッド】 オプションを 0 より大きな値に設定したときに適用されます。有効な値は 1~100000 です。デフォルトは 25 です。
書き込みモード	<p>データ統合サービスが PowerExchange リスナにデータを送信するモード。次のうちいずれかの書き込みモードを設定します。</p> <ul style="list-style-type: none"> - CONFIRMWRITEON。PowerExchange リスナにデータを送信し、応答を待ってから以降のデータを送信します。エラーのリカバリを優先する場合に選択します。このオプションを選択すると、パフォーマンスが低下する場合があります。 - CONFIRMWRITEOFF。応答を待たずに、PowerExchange リスナにデータを送信します。エラーが発生した際にターゲットテーブルをリロードできる場合には、このオプションを使用します。 - ASYNCHRONOUSWITHFAULTTOLERANCE。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションには、エラーを検出する機能も用意されています。このオプションは書き込み確認オンのデータ整合性で、書き込み確認オフの速度を上げる。デフォルトは CONFIRMWRITEON です。
拒否ファイル	拒否ファイルの PWXR のデフォルトプレフィックスをオーバーライドします。書き込みモードが ASYNCHRONOUSWITHFAULTTOLERANCE のときは、PowerExchange は拒否ファイルをターゲットマシンに作成します。拒否ファイルを作成しないようにするには、PWXDISABLE と入力します。

IMS 接続のプロパティ

IMS データベースにアクセスするには IMS 接続を使用します。IMS 接続は、非リレーショナルメインフレームデータベースタイプの接続です。データ統合サービスは PowerExchange 経由で IMS に接続します。IMS 接続は Developer ツールで作成します。IMS 接続は、Administrator ツールまたは Developer ツールで管理できます。

以下の表に、IMS 接続のプロパティを示します。

オプション	説明
場所	IMS に接続する PowerExchange リスナの場所のノード名。ノード名は、PowerExchange のコンフィギュレーションファイル (dbmover.cfg) 内の NODE 文の最初のパラメータで定義されます。
ユーザー名	データベースユーザー名。

オプション	説明
パスワード	<p>指定したデータベースユーザー名のパスワードまたは有効な PowerExchange のパスフレーズ。</p> <p>PowerExchange パスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。</p> <p>IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: 有効な RACF のパスフレーズは最長 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>IMS への接続のパスフレーズを使用するときは、次の要件を確実に満足させてください。</p> <ul style="list-style-type: none"> - PowerExchange リスナは、DBMOVER メンバでセキュリティ設定を SECURITY=(1,N) かそれ以上に設定したうえで実行する必要があります。詳細は、『PowerExchange リファレンスマニュアル』の「SECURITY ステートメント」を参照してください。 - ODBA の IMS へのアクセスは、『PowerExchange Navigator ユーザーガイド』で説明されているとおりに設定する必要があります。 - 必ずアクセスメソッドに IMS ODBA を指定している IMS データマップを使用してください。DL/1 BATCH アクセスメソッドを指定するデータマップは使用しないでください。これは、パスフレーズをサポートしていない netport ジョブの使用がこのアクセスメソッドで必須になっているためです。 - IMS データベースは、ODBA の IMS へのアクセスを使用するために、IMS 制御領域でオンラインにする必要があります。
コードページ	<p>必須。データソースの読み取りや書き込みに使用するコードページの名前。通常この値は ISO コードページの名前です（ISO-8859-6 など）。</p>
パススルーセキュリティ有効	<p>接続のパススルーセキュリティを有効にする。</p>

オプション	説明
暗号化タイプ	<p>データ統合サービスが使用する暗号化のタイプ。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - RC2 - DES <p>デフォルトは [なし] です。</p> <p>注:</p> <ul style="list-style-type: none"> - Informatica では、暗号化タイプおよびレベル接続プロパティを設定したり、DBMOVER 構成ファイルに ENCRYPT 文と ENCRYPTLEVEL 文を設定する代わりに、Secure Sockets Layer (SSL) 認証を使用することを推奨します。SSL 認証は複数の Informatica 製品で使用されており、より厳格なセキュリティが提供されます。 <p>PowerExchange ネットワークへの SSL 認証の実装の詳細については、『<i>PowerExchange リファレンスマニュアル</i>』を参照してください。</p> <ul style="list-style-type: none"> - ENCRYPT および ENCRYPTLEVEL 文の値が統合サービスマシンの DBMOVER 構成ファイルで定義されている場合、暗号化タイプおよびレベル接続属性に選択する値によって、これらの値はオーバーライドされます。マッピングの暗号化を有効にするには、必ず該当する接続属性を選択します。
[暗号化] レベル	<p>[暗号タイプ] で [RC2] または [DES] を選択した場合は、データ統合サービスが使用する暗号化レベルを指定するために、次に挙げるオプションのいずれかを選択してください。</p> <ul style="list-style-type: none"> - 1. DES および RC2 には 56 ビットの暗号化キーを使用します。 - 2. DES には 168 ビットのトリプル暗号化キーを、RC2 には 64 ビットの暗号化キーを使用します。 - 3. DES には 168 ビットのトリプル暗号化キーを、RC2 には 128 ビットの暗号化キーを使用します。 <p>暗号化タイプを選択しない場合、このオプションは無視されます。</p> <p>デフォルトは 1 です。</p>
ページングサイズ	<p>オプション。ソースシステムが PowerExchange リスナに渡すことができるデータ量。ページングサイズは、外部アプリケーション、データベース、またはデータ統合サービスのノードがボトルネックになっている場合に設定します。低い値を使用するとパフォーマンスが向上します。</p> <p>最小値およびデフォルト値は 0 です。値を 0 にするとパフォーマンスが最大になります。</p>
行として解釈	<p>オプション。このオプションは、ページングサイズを行数で表示する場合に選択します。このオプションをクリアすると、ページングサイズがキロバイト単位で表示されます。デフォルトではこのオプションが選択されていないため、ページングサイズはキロバイトで表示されています。</p>
圧縮	<p>オプション。このオプションは、ソースデータの圧縮を有効にする場合に選択します。データを圧縮することで、Informatica アプリケーションがネットワークで送信するデータ量を減少させることができます。デフォルトではこのオプションが選択されていないため、圧縮は無効になっています。</p>
オフロード処理	<p>オプション。バルクデータの処理をソースマシンからデータ統合サービスのマシンにオフロードするかどうかを制御します。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - AUTO。オフロード処理を使用するかどうかデータ統合サービスによって決定されます。 - はい。オフロード処理を使用します。 - いいえ。オフロード処理は使用しません。 <p>デフォルトは [AUTO] です。</p>

オプション	説明
ワーカースレッド	オプション。オフロード処理の有効時にデータ統合サービスがバルクデータを処理するために使用するスレッドの数。最適なパフォーマンスを得るためには、データ統合サービスマシンで使用可能なプロセッサの数を超えないようにこの値を設定します。有効な値は 1～64 です。デフォルトは 0 です。マルチスレッド処理は無効になります。
配列サイズ	オプション。ワーカースレッドのためのストレージアレイのレコード数。このオプションは、[ワーカースレッド] オプションを 0 より大きな値に設定したときに適用されます。有効な値は 1～100000 です。デフォルトは 25 です。
書き込みモード	<p>オプション。データ統合サービスが PowerExchange リスナにデータを送信する際のモード。次のうちいずれかの書き込みモードを選択します。</p> <ul style="list-style-type: none"> - CONFIRMWRITEON。PowerExchange リスナにデータを送信し、応答を待ってから以降のデータを送信します。このオプションはエラーリカバリを優先する場合に選択します。ただし、このオプションを使用するとパフォーマンスが落ちる可能性があります。 - CONFIRMWRITEOFF。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションは、エラーが発生したときにターゲットテーブルを再ロードしてもよい場合に使用します。 - ASYNCHRONOUSWITHFAULTTOLERANCE。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションはエラー検出も有効にします。このオプションは、CONFIRMWRITEOFF のスピードと CONFIRMWRITEON のデータの完全性を結合します。 <p>デフォルトは [CONFIRMWRITEON] です。</p>

JDBC 接続のプロパティ

JDBC 接続を使用してデータベースのテーブルにアクセスすることができます。JDBC 接続は、Administrator ツール、Developer ツール、または Analyst ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、JDBC 接続のプロパティを示します。

プロパティ	説明
データベースタイプ	データベースのタイプ。
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。</p>
説明	接続の説明。説明は、765 文字を超えることはできません。
ユーザー名	データベースユーザー名。

プロパティ	説明
パスワード	データベースユーザー名のパスワード。
JDBC ドライバクラス名	<p>JDBC ドライバクラスの名前。</p> <p>以下の表に、該当するデータベースタイプに入力できる、ドライバクラス名を示します。</p> <ul style="list-style-type: none"> - Oracle 用 DataDirect JDBC ドライバクラス名: com.informatica.jdbc.oracle.OracleDriver - IBM DB2 用 DataDirect JDBC ドライバクラス名: com.informatica.jdbc.db2.DB2Driver - Microsoft SQL Server 用 DataDirect JDBC ドライバクラス名: com.informatica.jdbc.sqlserver.SQLServerDriver - Sybase ASE 用 DataDirect JDBC ドライバクラス名: com.informatica.jdbc.sybase.SybaseDriver - Informix 用 DataDirect JDBC ドライバクラス名: com.informatica.jdbc.informix.InformixDriver - MySQL 用 DataDirect JDBC ドライバクラス名: com.informatica.jdbc.mysql.MySQLDriver <p>特定のデータベースで使用するドライバクラスの詳細については、ベンダ提供のドキュメントを参照してください。</p>
接続文字列	<p>データベースへの接続に使用する接続文字列。以下の接続文字列を使用します。</p> <p>jdbc:<subprotocol>:<subname></p>
環境 SQL	オプション。データベースに接続するとき、SQL コマンドを入力してデータベースの環境設定を行います。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。
トランザクション SQL	オプション。データベースに接続するとき、SQL コマンドを入力してデータベースの環境設定を行います。データ統合サービスにより、トランザクション環境 SQL が各トランザクションの開始時に実行されます。
SQL 識別子文字	<p>データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。</p> <p>データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。</p> <p>データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。</p>
大文字と小文字が混在する識別子をサポート	<p>データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を [SQL 識別子文字] プロパティに対して選択された文字で囲みます。</p> <p>[SQL 識別子文字] プロパティが [なし] に設定されている場合、[大文字と小文字が混在する識別子をサポート] プロパティは無効になっています。</p>
パススルーセキュリティ有効	<p>接続のパススルーセキュリティを有効にする。接続のパススルーセキュリティを有効にすると、ドメインは、接続オブジェクトに定義されている資格情報ではなく、クライアントのユーザー名とパスワードを使用して対応するデータベースにログインします。</p>

プロパティ	説明
メタデータアクセスプロパティ: 接続文字列	<p>データベースのメタデータへのアクセスに使用する JDBC 接続 URL。次のリストに、該当するデータベースタイプに入力できる接続文字列を示します。</p> <ul style="list-style-type: none"> - Oracle 用 DataDirect JDBC ドライバ: jdbc:informatica:oracle://<hostname>:<port>;SID=<sid> - IBM DB2 用 DataDirect JDBC ドライバ: jdbc:informatica:db2://<hostname>:<port>;DatabaseName=<database name> - Microsoft SQL Server 用 DataDirect JDBC ドライバ: jdbc:informatica:sqlserver://<host>:<port>;DatabaseName=<database name> - Sybase ASE 用 DataDirect JDBC ドライバ: jdbc:informatica:sybase://<host>:<port>;DatabaseName=<database name> - Informix 用 DataDirect JDBC ドライバ: jdbc:informatica:informix://<host>:<port>;informixServer=<informix server name>;DatabaseName=<database name> - MySQL 用 DataDirect JDBC ドライバ: jdbc:informatica:mysql://<host>:<port>;DatabaseName=<database name> <p>特定のデータベースに使用する接続文字列の詳細については、ベンダ提供のドキュメントで「URL 構文」を参照してください。</p>
AdvancedJDBCSecurityOptions	<p>セキュアデータベースへのメタデータアクセスのためのデータベースパラメータ。Informatica は、AdvancedJDBCSecurityOptions フィールドの値を機密データとして扱い、パラメータ文字列を暗号化して格納します。</p> <p>セキュアデータベースに接続するには、以下のパラメータを含めます。</p> <ul style="list-style-type: none"> - EncryptionMethod。必須。ネットワーク上で送信される際にデータが暗号化されるかどうかを示します。このパラメータは SSL に設定する必要があります。 - ValidateServerCertificate。オプション。データベースサーバーによって送信される証明書を Informatica が検証するかどうかを示します。 <p>このパラメータを True に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証します。 HostNameInCertificate パラメータを指定すると、Informatica は証明書内のホスト名も検証します。</p> <p>このパラメータを false に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証しません。指定するトラストストア情報がすべて無視されます。</p> <ul style="list-style-type: none"> - HostNameInCertificate。オプション。セキュアデータベースをホストするマシンのホスト名。ホスト名を指定すると、Informatica は接続文字列に含められたそのホスト名を SSL 証明書内のホスト名と照らして検証します。 - TrustStore。必須。データベースの SSL 証明書が含まれるトラストストアファイルのパスとファイル名。 - TrustStorePassword。必須。セキュアデータベースに対するトラストストアファイルのパスワード。 <p>ODBC には適用されません。 注: セキュア JDBC パラメータは接続文字列に自動的に追加されます。セキュア JDBC パラメータを接続文字列に直接指定する場合は、AdvancedJDBCSecurityOptions フィールドにはパラメータを入力しないでください。</p>
コードページ	ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。

プロパティ	説明
環境 SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。
トランザクション SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、トランザクション環境 SQL を各トランザクションの開始時に実行します。
リトライ期限	このプロパティは、将来使用するために予約されています。
SQL 識別子文字	<p>特殊文字と SQL の予約済み SQL キーワード（WHERE など）の識別に使用される文字のタイプ。データ統合サービスは特殊文字と予約済み SQL キーワードを選択した文字で囲みます。データ統合サービスはこの文字を、[大文字と小文字が混在する識別子をサポート] プロパティにも使用します。</p> <p>接続内のデータベースに対応する引用符を選択します。</p>
大文字と小文字が混在する識別子をサポート	有効にすると、接続内でテーブル、ビュー、スキーマ、シノニム、およびカラムに対して SQL を生成および実行するときに、データ統合サービスはこれらのオブジェクトの名前を識別子文字で囲みます。オブジェクト名が、大文字と小文字が混在している場合または小文字の場合に使用します。デフォルトでは、このオプションは選択されていません。
Sqoop コネクタを使用	<p>JDBC 接続を使用するデータオブジェクトの Sqoop 接続を可能にします。データ統合サービスは、Sqoop プログラムをトリガして、Hadoop ランタイム環境でマッピングを実行します。</p> <p>次のデータベースに基づくリレーショナルデータオブジェクト、カスタマイズされたデータオブジェクト、および論理データオブジェクトに Sqoop 接続を設定できます。</p> <ul style="list-style-type: none"> - Aurora - Greenplum - IBM DB2 - IBM DB2 for z/OS - Microsoft SQL Server - Netezza - Oracle - Teradata <p>Sqoop 接続を有効にするには [Sqoop v1.x] を選択します。</p> <p>【なし】 を選択すると、データ統合サービスは Hadoop ランタイム環境で Sqoop プログラムがマッピングを実行するようにトリガしません。マッピングの実行には JDBC が使用されます。</p>
Sqoop 引数	<p>データベースに接続するために Sqoop プログラムが使用する必要がある引数を入力します。複数の引数を入力するには、オプションをスペースで区切ります。</p> <p>Sqoop 引数が入力されなかった場合、データ統合サービスは、JDBC 接続プロパティに基づいて、Sqoop コマンドを作成します。</p>

JD Edwards EnterpriseOne 接続プロパティ

JD Edwards EnterpriseOne 接続を使用して、JD Edwards EnterpriseOne オブジェクトに接続します。

次の表に、JD Edwards EnterpriseOne 接続プロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成する Informatica ドメイン。
タイプ	接続タイプ。JD Edwards EnterpriseOne を選択します。
ホスト名	JD Edwards EnterpriseOne サーバーのホスト名。
エンタープライズポート	JD Edwards EnterpriseOne サーバーのポート番号。デフォルトは 6016 です。
ユーザー名	JD Edwards EnterpriseOne データベースユーザーの名前。
パスワード	JD Edwards EnterpriseOne データベースユーザーのパスワード。
環境	接続先の JD Edwards EnterpriseOne 環境の名前。
ロール	JD Edwards EnterpriseOne ユーザーのロール。デフォルトは [*すべて] です。
ユーザー名	JD Edwards EnterpriseOne データベースユーザーの名前。
パスワード	データベースユーザーのパスワード。

プロパティ	説明
ドライバクラス名	<p>以下の表に、該当するデータベースタイプに入力できる、ドライバクラス名を示します。</p> <ul style="list-style-type: none"> - Oracle 用 DataDirect JDBC ドライバクラス名: <code>com.informatica.jdbc.oracle.OracleDriver</code> - IBM DB2 用 DataDirect JDBC ドライバクラス名: <code>com.informatica.jdbc.db2.DB2Driver</code> - Microsoft SQL Server 用 DataDirect JDBC ドライバクラス名: <code>com.informatica.jdbc.sqlserver.SQLServerDriver</code> <p>特定のデータベースで使用するドライバクラスの詳細については、ベンダ提供のドキュメントを参照してください。</p>
接続文字列	<p>データベースへの接続に使用する接続文字列。以下の接続文字列を使用します。</p> <p>JDBC 接続文字列では、次の構文を使用します。</p> <ul style="list-style-type: none"> - Oracle の場合: <code>jdbc:informatica:oracle://<host name>:<port>,ServiceName=<db service name></code> - DB2 の場合: <code>jdbc:informatica:db2://<host name>:<port>;databaseName=<db name></code> - Microsoft SQL の場合: <code>jdbc:informatica:sqlserver://<host name>:<port>;databaseName=<db name></code>

LDAP 接続プロパティ

LDAP 接続を使用して、LDAP オブジェクトに接続します。

次の表に、LDAP 接続のプロパティを示します。

プロパティ	説明
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、765 文字を超えることはできません。</p>
場所	<p>接続を作成する Informatica ドメイン。</p>
タイプ	<p>接続タイプ。LDAP を選択します。</p>
ホスト名	<p>LDAP ディレクトリサーバーのホスト名。デフォルトは localhost です。</p>
ポート	<p>LDAP ディレクトリサーバーのポート番号。デフォルトは 389 です。</p>

プロパティ	説明
匿名接続	LDAP ディレクトリサーバーとの匿名接続を確立します。匿名接続を選択し、認証不要の匿名ユーザーとしてディレクトリサーバーにアクセスします。 注: アクティブディレクトリとは匿名接続を確立できません。
ユーザー名	LDAP ディレクトリサーバーに接続するための LDAP ユーザー名。
パスワード	LDAP ディレクトリサーバーに接続するためのパスワード。
セキュアな接続	TLS プロトコル経由で LDAP ディレクトリサーバーとのセキュアな接続を確立します。
TrustStore のファイル名	LDAP ディレクトリサーバーとのセキュアな接続を確立するための TLS 証明書を含むトラストストアのファイル名。 デフォルトは、infa_truststore.jks です。 セキュアな接続を選択した場合は必須です。トラストストアのファイル名とパスワードについては、LDAP 管理者にお問い合わせください。
TrustStore のパスワード	SSL 証明書を含むトラストストアファイルのパスワード。
KeyStore のファイル名	LDAP ディレクトリサーバーとのセキュアな通信を確立するために必要なキーと証明書を含むキーストアのファイル名。 セキュアな接続を選択した場合は必須です。キーストアのファイル名とパスワードについては、LDAP 管理者にお問い合わせください。
KeyStore のパスワード	通信を安全に行うために必要なキーストアファイルのパスワードです。

LinkedIn 接続プロパティ

LinkedIn Web サイトからデータを抽出するには、LinkedIn 接続を使用します。LinkedIn 接続は、ソーシャルメディアタイプの接続です。LinkedIn 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、LinkedIn 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。

プロパティ	説明
場所	接続を作成するドメイン。
タイプ	接続タイプ。LinkedIn を選択します。
OAuth の詳細の有無	OAuth を設定するかどうかを示します。 次のうちいずれかの値を選択します。 - [はい]。アクセストークンおよびシークレットがあることを示します。 - [いいえ]。OAuth Utility を起動します。
コンシューマキー	LinkedIn でアプリケーションを作成するときに取得する API キー。 LinkedIn はこのキーを使用してアプリケーションを識別します。
コンシューマシークレット	LinkedIn でアプリケーションを作成するときに取得するシークレットキー。 LinkedIn はこのシークレットを使用してコンシューマキーの所有権を確立します。
アクセストークン	OAuth Utility から返されるアクセストークン。 LinkedIn アプリケーションはユーザークレデンシャルの代わりにこのトークンを使用して、保護されたリソースにアクセスします。
アクセスシークレット	OAuth Utility が返すアクセスシークレット。 このシークレットがトークンの所有権を確立します。
スコープ	オプション。アプリケーションの権限。 OAuth の設定に使用した権限を入力します。

Microsoft Azure Blob ストレージ接続のプロパティ

Microsoft Azure Blob ストレージ接続をセットアップする際には、接続プロパティを設定する必要があります。

次の表に、Microsoft Azure Blob ストレージ接続のプロパティを示します。

プロパティ	説明
名前	Microsoft Azure Blob ストレージ接続の名前。
ID	データ統合サービスが接続の識別に使用する文字列。 ID では大文字と小文字が区別されません。 255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。
場所	接続を作成するドメイン。
タイプ	接続タイプ。AzureBlob を選択します。

【接続の詳細】タブには、Microsoft Azure Blob ストレージ接続の接続属性が含まれています。以下の表に、接続属性を示します。

プロパティ	説明
アカウント名	Microsoft Azure ストレージアカウントの名前。
アカウントキー	Microsoft Azure ストレージアクセスキー。
コンテナ名	Microsoft Azure Blob ストレージコンテナ名。 注: PowerExchange for Microsoft Azure Blob Storage はサブコンテナをサポートしていません。

Microsoft Azure SQL Data Warehouse 接続プロパティ

Microsoft Azure SQL Data Warehouse 接続をセットアップするときは、接続プロパティを設定する必要があります。

次の表に、Microsoft Azure SQL Data Warehouse 接続プロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。このプロパティは、接続を作成した後に変更できます。名前は 128 文字以内で指定し、空白および次の特殊文字は使用できません。~`!\$%^&*()-+={} \;\:'<,>./
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、4,000 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。Microsoft Azure SQL Data Warehouse を選択します。

次の表に、メタデータアクセスのプロパティを示します。

プロパティ	説明
Azure DW JDBC URL	Microsoft Azure Data Warehouse JDBC 接続文字列。例えば、次の接続文字列を入力できます: jdbc:sqlserver://<Server>.database.windows.net:1433;database=<Database>。管理者は Microsoft Azure ポータルから URL をダウンロードできます。
Azure DW JDBC ユーザー名	Microsoft Azure SQL Data Warehouse アカウントに接続するためのユーザー名。Microsoft Azure SQL Data Warehouse のデータを読み取り、書き込み、切り詰めるための権限が必要です。

プロパティ	説明
Azure DW JDBC パスワード	Microsoft Azure SQL Data Warehouse アカウントに接続するためのパスワード。
Azure DW スキーマ名	Microsoft Azure SQL Data Warehouse 内のスキーマの名前。
Azure Blob アカウント名	ファイルをステージングする Microsoft Azure ストレージアカウントの名前。
Azure Blob アカウントキー	Blob ストレージアカウントへのアクセスを認証するキー。

MS SQL Server 接続のプロパティ

Microsoft SQL Server に接続するには、Microsoft SQL Server 接続を使用します。Microsoft SQL Server 接続は、Microsoft SQL Server リレーショナルデータベースへの接続です。Microsoft SQL Server 接続は、Administrator ツールまたは Developer tool で作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、MS SQL Server 接続のプロパティを示します。

プロパティ	説明
データベースタイプ	データベースタイプ。
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
信頼関係接続の使用	アプリケーションサービスで Windows 認証を使用してデータベースにアクセスできるようにします。アプリケーションサービスを起動するユーザー名は、データベースにアクセスできる Windows ユーザーである必要があります。デフォルトでは、このオプションはクリアされています。
ユーザー名	データベースユーザー名。Microsoft SQL Server が NTLMv1 または NTLMv2 認証を使用する場合に必須です。
パスワード	データベースユーザー名のパスワード。Microsoft SQL Server が NTLMv1 または NTLMv2 認証を使用する場合に必須です。

プロパティ	説明
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。接続のパススルーセキュリティを有効にすると、ドメインは、接続オブジェクトに定義されている資格情報ではなく、クライアントのユーザー名とパスワードを使用して対応するデータベースにログインします。
メタデータアクセスプロパティ: 接続文字列	<p>データベースからメタデータにアクセスするために使用する接続文字列。以下の接続文字列を使用します。</p> <pre>jdbc:informatica:sqlserver://<host name>:<port>;DatabaseName=<database name></pre> <p>NTLM 認証を使用した接続をテストするには、接続文字列に次のパラメータを追加します。</p> <ul style="list-style-type: none"> - AuthenticationMethod。使用する NTLM 認証のバージョン。 <p>注: UNIX では、NTLMv1 と NTLMv2 はサポートされますが、NTLM はサポートされません。</p> <ul style="list-style-type: none"> - ドメイン。SQL サーバーが属するドメイン。 <p>次の例では、Informatica.com という名前の NT ドメインで NTLMv2 認証を使用する SQL サーバーへの接続文字列を示します。</p> <pre>jdbc:informatica:sqlserver://host01:1433;DatabaseName=SQL1;AuthenticationMethod=ntlm2java;Domain=Informatica.com</pre> <p>NTLM 認証で接続する場合、MS SQL Server 接続プロパティの【信頼関係接続の使用】オプションを有効にできます。NTLMv1 または NTLMv2 認証で接続する場合、接続プロパティにユーザー名とパスワードを入力する必要があります。</p>

プロパティ	説明
AdvancedJDBCSecurityOptions	<p>セキュアデータベースへのメタデータアクセスのためのデータベースパラメータ。Informatica は、AdvancedJDBCSecurityOptions フィールドの値を機密データとして扱い、パラメータ文字列を暗号化して格納します。</p> <p>セキュアデータベースに接続するには、以下のパラメータを含めます。</p> <ul style="list-style-type: none"> - EncryptionMethod。必須。ネットワーク上で送信される際にデータが暗号化されるかどうかを示します。このパラメータは SSL に設定する必要があります。 - ValidateServerCertificate。オプション。データベースサーバーによって送信される証明書を Informatica が検証するかどうかを示します。 このパラメータを True に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証します。HostNameInCertificate パラメータを指定すると、Informatica は証明書内のホスト名も検証します。 このパラメータを false に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証しません。指定するトラストストア情報がすべて無視されます。 - HostNameInCertificate。オプション。セキュアデータベースをホストするマシンのホスト名。ホスト名を指定すると、Informatica は接続文字列に含められたそのホスト名を SSL 証明書内のホスト名と照らして検証します。 - cryptoProtocolVersion。オプション。Microsoft SQL Server インスタンスに対して TLS を有効にする場合は、次のように cryptoProtocolVersion パラメータを設定します。 cryptoProtocolVersion=TLsv<version number> 例: cryptoProtocolVersion=TLsv1.2 注: バージョン番号はサーバーに設定した TLS バージョンと同じにする必要があります。 - TrustStore。必須。トラストストアファイルのパスおよびファイル名です。 注: SSL または TLS を設定してファイル名のみを指定する場合は、トラストストアファイルを Informatica のインストールディレクトリにコピーする必要があります。接続をテストしメタデータをインポートするには、トラストストアファイルを次のディレクトリにコピーします。 <Informatica client installation directory>/clients/DeveloperClient マッピングを実行するには、トラストストアファイルを次のディレクトリにコピーします。 <Informatica server installation directory>/tomcat/bin 注: SSL または TLS を設定してファイル名のみを指定する場合は、トラストストアファイルを次のディレクトリにコピーして接続をテストする必要があります。 <Informatica server installation directory>/tomcat/bin - TrustStorePassword。必須。セキュアデータベースに対するトラストストアファイルのパスワード。 <p>ODBC には適用されません。 注: セキュア JDBC パラメータは接続文字列に自動的に追加されます。セキュア JDBC パラメータを接続文字列に直接指定する場合は、AdvancedJDBCSecurityOptions フィールドにはパラメータを入力しないでください。</p>

プロパティ	説明
データアクセスプロパティ: プロバイダタイプ	<p>Microsoft SQL Server データベースへの接続に使用する接続プロバイダ。次のプロバイダタイプを選択することができます。</p> <ul style="list-style-type: none"> - ODBC - Oledb (廃止予定) <p>デフォルトは ODBC です。</p> <p>OLEDDB は廃止予定のプロバイダタイプです。今後のリリースで OLEDB プロバイダタイプのサポートは終了する予定です。</p>
DSN の使用	<p>データ統合サービスで接続のデータソース名を使用できるようになります。</p> <p>[DSN の使用] オプションを使用した場合、データ統合サービスはデータベース名とサーバー名を DSN から取得します。</p> <p>[DSN の使用] オプションを選択しない場合、データベース名とサーバー名を指定する必要があります。</p>
接続文字列	<p>DSN モードを有効にしない場合、次の接続文字列を使用します。</p> <p><server name>@<database name></p> <p>DSN モードを有効にする場合、次の接続文字列を使用します。</p> <p><DSN Name></p>
コードページ	ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。
ドメイン名	ドメインの名前。
パケットサイズ	データ転送で使用するパケットサイズ。Microsoft SQL Server のネイティブドライバの最適化に使用されます。
所有者名	<p>スキーマの所有者の名前。</p> <p>注: 動的マッピングまたは [DDL の生成と実行] オプションを使用してテーブル DDL を生成すると、DDL メタデータにスキーマ名と所有者名のプロパティが含まれません。</p>
スキーマ名	<p>データベースのスキーマの名前。スキーマ名がデータベースユーザー名と異なる場合は、プロファイリングウェアハウスのスキーマ名を指定する必要があります。スキーマ名がデータベースのユーザー名と異なり、ユーザーが管理するキャッシュテーブルを設定する場合は、データオブジェクトキャッシュデータベースのスキーマ名を指定する必要があります。</p> <p>注: 動的マッピングまたは [DDL の生成と実行] オプションを使用してテーブル DDL を生成すると、DDL メタデータにスキーマ名と所有者名のプロパティが含まれません。</p>
環境 SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。
トランザクション SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、トランザクション環境 SQL を各トランザクションの開始時に実行します。
リトライ期限	このプロパティは、将来使用するために予約されています。

プロパティ	説明
SQL 識別子文字	<p>データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。</p> <p>データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。</p> <p>データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。</p>
大文字と小文字が混在する識別子をサポート	<p>データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を [SQL 識別子文字] プロパティに対して選択された文字で囲みます。</p> <p>[SQL 識別子文字] プロパティが [なし] に設定されている場合、[大文字と小文字が混在する識別子をサポート] プロパティは無効になっています。</p>
ODBC プロバイダ	<p>ODBC。ODBC の接続先となるデータベースのタイプ。プッシュダウンの最適化用に、データベースタイプを指定して、データ統合サービスでネイティブデータベース SQL を生成できるようにします。次のオプションがあります。</p> <ul style="list-style-type: none"> - その他 - Sybase - Microsoft_SQL_Server <p>デフォルトは [その他] です。</p>

Netezza 接続のプロパティ

Netezza 接続を使用して、Netezza データベースにアクセスします。Netezza 接続は、データベース接続です。Netezza 接続は、Administrator ツールまたは Developer tool で作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Netezza 接続のプロパティを示します。

プロパティ	説明
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、765 文字を超えることはできません。</p>
場所	<p>接続を作成するドメイン。</p>
タイプ	<p>接続タイプ。[Netezza] を選択します。</p>
ユーザー名	<p>Netezza データベースにアクセスするための適切な権限を持つユーザー名。</p>

プロパティ	説明
パスワード	上記データベースユーザー名のパスワード。
JDBC Url	Netezza データベースへの接続時に Developer tool で使用する必要がある JDBC URL。 次の形式を使用します。 jdbc:netezza://<ホスト名>:<ポート>/<データベース名>
接続文字列	Netezza データベースへの接続のために使用する ODBC データソースの名前。
タイムアウト	Developer tool が接続を閉じる前に Netezza データベースからの応答を待機する秒数。

OData 接続のプロパティ

OData 接続を使用して、OData URL にアクセスします。OData 接続は、Web 接続です。OData 接続は、Administrator ツールまたは Developer tool で作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、OData 接続プロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、4,000 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。[OData] を選択します。
ユーザー名	オプション。OData リソースからデータを読み取るための適切な権限を持つユーザー名。
パスワード	オプション。OData URL ユーザー名のパスワード。
URL	読み取る必要があるデータを公開している OData サービスのルート URL。

プロパティ	説明
セキュリティタイプ	<p>オプション。OData サーバーへの安全な接続を確立するために Developer tool で使用する必要があるセキュリティプロトコル。</p> <p>次のいずれかの値を選択します。</p> <ul style="list-style-type: none"> - なし - SSL - TLS <p>デフォルトは「なし」です。</p>
トラストストアファイル名	<p>セキュリティタイプを選択する場合、必須です。</p> <p>OData サーバーの公開証明書が含まれるトラストストアファイルの名前。</p> <p>デフォルトは、infa_truststore.jks です。</p>
パスワード	<p>セキュリティタイプを選択する場合、必須です。</p> <p>OData サーバーの公開証明書が含まれるトラストストアファイルのパスワード。</p>
キーストアファイル名	<p>セキュリティタイプを選択する場合、必須です。</p> <p>OData サーバーのプライベートキーが含まれるキーストアファイルの名前。</p> <p>デフォルトは、infa_truststore.jks です。</p>
パスワード	<p>セキュリティタイプを選択する場合、必須です。</p> <p>OData サーバーのプライベートキーが含まれるキーストアファイルのパスワード。</p>

ODBC 接続のプロパティ

ODBC データにアクセスするには、ODBC 接続を使用します。ODBC 接続はリレーショナルデータベース接続です。ODBC 接続は、Administrator ツール、Developer ツール、または Analyst ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、ODBC 接続のプロパティを示します。

プロパティ	説明
データベースタイプ	データベースのタイプ。
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、765 文字を超えることはできません。</p>
ユーザー名	データベースユーザー名。

プロパティ	説明
パスワード	データベースユーザー名のパスワード。
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。接続のパススルーセキュリティを有効にすると、ドメインは、接続オブジェクトに定義されている資格情報ではなく、クライアントのユーザー名とパスワードを使用して対応するデータベースにログインします。
データアクセスプロパティ: 接続文字列	データベースのメタデータへのアクセスに使用する ODBC 接続 URL。 <data source name>
コードページ	ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。
環境 SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。
トランザクション SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、トランザクション環境 SQL を各トランザクションの開始時に実行します。
リトライ期限	このプロパティは、将来使用するために予約されています。
SQL 識別子文字	データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。 データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。 データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。
大文字と小文字が混在する識別子をサポート	データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を【SQL 識別子文字】プロパティに対して選択された文字で囲みます。 【SQL 識別子文字】プロパティが [なし] に設定されている場合、【大文字と小文字が混在する識別子をサポート】プロパティは無効になっています。
ODBC プロバイダ	ODBC を接続するデータベースのタイプ。プッシュダウンの最適化用に、データベースタイプを指定して、データ統合サービスでネイティブデータベース SQL を生成できるようにします。次のオプションがあります。 - その他 - Sybase - Microsoft_SQL_Server デフォルトは【その他】です。

注: UNIX または Linux 上でデータ統合サービスを実行する場合、ODBC 接続を使用して Microsoft SQL Server に接続します。Windows 上でデータ統合サービスを実行する場合、ネイティブ接続を使用して Microsoft SQL Server に接続します。

Oracle 接続のプロパティ

Oracle データベースに接続するには、Oracle 接続を使用します。Oracle 接続は、リレーショナル接続タイプです。Oracle 接続は、Administrator ツール、Developer ツール、または Analyst ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Oracle 接続のプロパティを示します。

プロパティ	説明
データベースタイプ	データベースタイプ。
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~`!\$%^&*()-+=[{] \:;"'<, > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
ユーザー名	データベースユーザー名。
パスワード	データベースユーザー名のパスワード。
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。接続のパススルーセキュリティを有効にすると、ドメインは、接続オブジェクトに定義されている資格情報ではなく、クライアントのユーザー名とパスワードを使用して対応するデータベースにログインします。
メタデータアクセスプロパティ: 接続文字列	データベースからメタデータにアクセスするために使用する接続文字列。 以下の接続文字列を使用します。 <code>jdbc:informatica:oracle://<host_name>:<port>;SID=<database name></code>

プロパティ	説明
AdvancedJDBCSecurityOptions	<p>セキュアデータベースへのメタデータアクセスのためのデータベースパラメータ。Informatica は、AdvancedJDBCSecurityOptions フィールドの値を機密データとして扱い、パラメータ文字列を暗号化して格納します。</p> <p>セキュアデータベースに接続するには、以下のパラメータを含めます。</p> <ul style="list-style-type: none"> - EncryptionMethod。必須。ネットワーク上で送信される際にデータが暗号化されるかどうかを示します。このパラメータは SSL に設定する必要があります。 - ValidateServerCertificate。オプション。データベースサーバーによって送信される証明書を Informatica が検証するかどうかを示します。 このパラメータを True に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証します。HostNameInCertificate パラメータを指定すると、Informatica は証明書内のホスト名も検証します。 このパラメータを false に設定すると、Informatica はデータベースサーバーによって送信される証明書を検証しません。指定するトラストストア情報がすべて無視されます。 - HostNameInCertificate。オプション。セキュアデータベースをホストするマシンのホスト名。ホスト名を指定すると、Informatica は接続文字列に含められたそのホスト名を SSL 証明書内のホスト名と照らして検証します。 - cryptoProtocolVersion。オプション。Oracle インスタンスに対して TLS を有効にする場合は、次のように cryptoProtocolVersion パラメータを設定します。 cryptoProtocolVersion=TLSv<version number> 例: cryptoProtocolVersion=TLSv1.2 注: バージョン番号はサーバーに設定した TLS バージョンと同じにする必要があります。 - TrustStore。必須。トラストストアファイルのパスおよびファイル名です。 注: SSL または TLS を設定してファイル名のみを指定する場合は、トラストストアファイルを Informatica のインストールディレクトリにコピーする必要があります。接続をテストしメタデータをインポートするには、トラストストアファイルを次のディレクトリにコピーします。 <Informatica client installation directory>/clients/DeveloperClient マッピングを実行するには、トラストストアファイルを次のディレクトリにコピーします。 <Informatica server installation directory>/tomcat/bin 注: SSL または TLS を設定してファイル名のみを指定する場合は、トラストストアファイルを次のディレクトリにコピーして接続をテストする必要があります。<Informatica server installation directory>/tomcat/bin - TrustStorePassword。必須。セキュアデータベースに対するトラストストアファイルのパスワード。 - KeyStore。必須。キーストアファイルのパスとファイル名。 - KeyStorePassword。必須。セキュアデータベースに対するキーストアファイルのパスワード。 <p>注: セキュア JDBC パラメータは接続文字列に自動的に追加されます。セキュア JDBC パラメータを接続文字列に直接指定する場合は、AdvancedJDBCSecurityOptions フィールドにはパラメータを入力しないでください。</p>
データアクセスプロパティ: 接続文字列	<p>以下の接続文字列を使用します。</p> <p><database name>.world</p>

プロパティ	説明
コードページ	ソースデータベースからの読み取りや、ターゲットデータベースまたはファイルへの書き込みに使用されるコードページ。
環境 SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、データベースに接続するたびに、接続環境 SQL を実行します。
トランザクション SQL	データベースに接続する場合に、データベース環境を設定する SQL コマンド。データ統合サービスは、トランザクション環境 SQL を各トランザクションの開始時に実行します。
リトライ期限	このプロパティは、将来使用するために予約されています。
パラレルモードを有効にする	データを一括モードでテーブルにロードする場合の並列処理を可能にします。デフォルトでは、このオプションはクリアされています。
SQL 識別子文字	データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。 データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。 データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。
大文字と小文字が混在する識別子をサポート	データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を 【SQL 識別子文字】 プロパティに対して選択された文字で囲みます。 【SQL 識別子文字】 プロパティが 【なし】 に設定されている場合、 【大文字と小文字が混在する識別子をサポート】 プロパティは無効になっています。

Salesforce 接続プロパティ

Salesforce 接続を使用して Salesforce オブジェクトに接続します。Salesforce 接続はアプリケーション接続タイプです。Salesforce 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Salesforce 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。128 文字を超えることはできません。また、空白文字および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID では大文字と小文字が区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成してから変更できません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成する Informatica ドメイン。
タイプ	接続タイプ。Salesforce を選択。
ユーザー名	Salesforce のユーザ名。
ユーザパスワード	Salesforce のユーザ名に対するパスワード。 所属する組織の信頼済みネットワークの外部で Salesforce にアクセスするには、セキュリティトークンをパスワードに追加して、API またはデスクトップクライアントにログインする必要があります。 セキュリティトークンを受信またはリセットするには、Salesforce にログインして、[設定] > [個人情報] > [セキュリティトークンのリセット] の順にクリックします。 パスワードでは、大文字と小文字が区別されます。
サービスの URL	アクセスする Salesforce サービスの URL。テストまたは開発環境で、Salesforce Sandbox テスト環境にアクセスできます。Salesforce Sandbox の詳細については、Salesforce のマニュアルを参照してください。

SAP 接続のプロパティ

SAP テーブルまたは SAP BW オブジェクトにアクセスするには、SAP 接続を使用します。SAP 接続は、エンタープライズアプリケーション接続の一種です。SAP 接続は、Administrator のコンソールまたは Developer tool で作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、SAP 接続のプロパティを示します。

プロパティ	説明
ユーザー名	必須。アクセス先の SAP ソースシステムのユーザー名。
パスワード	必須。ユーザー名に対するパスワード。

プロパティ	説明
接続タイプ	<p>必須。作成する接続のタイプ。 次のいずれかの値を選択します。</p> <ul style="list-style-type: none"> - アプリケーション。特定の SAP アプリケーションサーバーに接続する際にアプリケーション接続を作成します。 - 負荷分散。SAP 負荷分散を使用する場合は、負荷分散接続を作成します。 <p>デフォルトは [アプリケーション] です。</p> <p>選択した接続タイプに応じて、対応する接続プロパティフィールドが【接続の詳細】ダイアログボックスで使用可能になります。Developer tool では、特定の接続タイプで使用できない接続プロパティフィールドが灰色で表示されます。</p>
ホスト名	<p>SAP アプリケーション接続を作成する場合は必須。 接続先の SAP サーバーのホスト名または IP アドレス。</p>
システム番号	<p>SAP アプリケーション接続を作成する場合は必須。 SAP システム番号。</p>
メッセージホスト名	<p>SAP 負荷分散接続を作成する場合は必須。 SAP メッセージサーバーのホスト名。</p>
R3 名/SysID	<p>SAP 負荷分散接続を作成する場合は必須。 SAP システムの名前。</p>
グループ	<p>SAP 負荷分散接続を作成する場合は必須。 SAP アプリケーションサーバーのグループ名。</p>
クライアント	<p>必須。SAP クライアント番号。</p>
言語	<p>オプション。マッピングおよびワークフローで使用する言語。 Developer tool のコードページと互換性のある言語でなければなりません。 このオプションを空白のままにすると、Developer tool は、SAP システムのデフォルトの言語を使用します。</p>
トレース	<p>オプション。このオプションは、SAP システムによる JCo 呼び出しを追跡する場合に使用します。SAP では、JCo 呼び出しについての情報をトレースファイルに保存しています。 次のいずれかの値を指定します。</p> <ul style="list-style-type: none"> - 0. オフ - 1. フル <p>デフォルトは 0 です。</p> <p>以下のディレクトリからトレースファイルにアクセスできます。</p> <ul style="list-style-type: none"> - Informatica サービスをインストールしたマシン上のディレクトリ<Informatica インストールディレクトリ>/tomcat/bin - Developer tool をインストールしたマシン上のディレクトリ<Informatica インストールディレクトリ>/clients/DeveloperClient
追加パラメータ	<p>オプション。使用するその他の接続パラメータを入力します。 次の形式を使用します。</p> <p><parameter name>=<value></p>
ステージングディレクトリ	<p>ステージファイルが作成される SAP システムのパス。</p>

プロパティ	説明
ソースディレクトリ	ソースファイルが保存されるパス。データ統合サービスに、このパスへのアクセス許可を与える必要があります。
FTP を使用	FTP で SAP にアクセスできるようにします。
FTP ユーザー名	FTP を使用する場合は必須。 FTP サーバーに接続するユーザー名。
FTP パスワード	FTP を使用する場合は必須。 FTP ユーザーのパスワード。
FTP ホスト	FTP を使用する場合は必須。 FTP サーバーのホスト名または IP アドレス。 オプションで、1 から 65535 までのポート番号を指定できます。FTP のデフォルトは 21 です。 次のいずれかの構文でホスト名を指定します。 - hostname:port_number - IP address:port_number ポート番号を指定する場合は、ホストマシン上でそのポート番号に対して FTP を使用可能にしてください。 SFTP を有効にする場合、SFTP サーバ用のホスト名またはポート番号を指定する必要があります。SFTP のデフォルトは 22 です。
リトライ期限	FTP ホストへの接続が失敗した場合に、データ統合サービスによって再接続が試行される秒数。 リトライ期限までに FTP ホストに再接続できなかった場合、マッピングまたはワークフローは失敗します。 デフォルトは 0 です。値 0 は、リトライ期限を無期限にすることを意味します。
SFTP の使用	SFTP が SAP にアクセスできるようにします。
パブリックキーファイル名	SFTP が有効で、SFTP サーバーがパブリックキー認証を使用している場合は必須。 パブリックキーファイルのパスおよびファイル名。
プライベートキーファイル名	SFTP が有効で、SFTP サーバーがパブリックキー認証を使用している場合は必須。 プライベートキーファイルのパスおよびファイル名。
プライベートキーファイル名パスワード	SFTP が有効で、SFTP サーバーがパブリックキー認証を使用しており、プライベートキーを暗号化する場合に必須。 プライベートキーファイルを復号化するためのパスワード。
HTTP ポート範囲	データ統合サービスが、SAP サーバーからストリーミングモードでデータを読み取る際に使用する必要がある HTTP ポート範囲。 最小ポート番号と最大ポート番号をハイフンでつないで入力します。最小ポート番号と最大ポート番号は、10000 - 65535 の範囲内で指定します。組織に応じたポート範囲を指定することもできます。 デフォルトは 10000～65535 です。

シーケンシャル接続のプロパティ

ソースデータセットにアクセスするにはシーケンシャル接続を使用します。シーケンシャル接続は Developer ツールで作成します。シーケンシャル接続は、Administrator ツールまたは Developer ツールで管理できます。

シーケンシャルデータソースは、PowerExchange が SEQ のアクセス方式で定義されたデータマップを使用してアクセスできるデータソースです。データ統合サービスは PowerExchange 経由でデータソースに接続します。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、シーケンシャル接続のプロパティを示します。

オプション	説明
場所	シーケンシャルなデータセットに接続する PowerExchange リスナの場所のノード名ノード名は、PowerExchange のコンフィギュレーションファイル (dbmover.cfg) 内の NODE 文の最初のパラメータで定義されます。
ユーザー名	シーケンシャルデータセットにアクセスする権限のあるユーザー名
パスワード	<p>指定したユーザー名のパスワードまたは有効な PowerExchange のパスフレーズ。PowerExchange のパスフレーズは 9～128 文字の範囲で作成します。使用できる文字は次のとおりです。</p> <ul style="list-style-type: none">- 大文字と小文字- 0～9 の数字- スペース- 次の特殊文字: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに単一引用符 (')、二重引用符 (")、通貨記号を含めることはできません。</p> <p>パスフレーズを使用するには、DBMOVER メンバでセキュリティ設定を SECURITY=(1,N) かそれ以上に設定したうえで PowerExchange リスナが実行されていることを確認してください。詳細は、『PowerExchange リファレンスマニュアル』の「SECURITY ステートメント」を参照してください。</p> <p>IBM IRRPHREX の終了で許可される文字は、PowerExchange のパスフレーズで許可されている文字に影響を与えません。</p> <p>注: 有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、パスフレーズを検証のため RACF に渡す場合、100 文字より長いパスフレーズを切り捨てます。</p>
コードページ	必須。シーケンシャルなデータセットの読み取りや書き込みに使用するコードページの名前。通常この値は ISO コードページの名前です (ISO-8859-6 など)。
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。

オプション	説明
暗号化タイプ	<p>オプション。データ統合サービスが使用する暗号化のタイプ。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - RC2 - DES <p>デフォルトは [なし] です。</p> <p>注:</p> <ul style="list-style-type: none"> - Informatica では、暗号化タイプおよびレベル接続プロパティを設定したり、DBMOVER 構成ファイルに ENCRYPT 文と ENCRYPTLEVEL 文を設定する代わりに、Secure Sockets Layer (SSL) 認証を使用することを推奨します。SSL 認証は複数の Informatica 製品で使用されており、より厳格なセキュリティが提供されます。 - PowerExchange ネットワークへの SSL 認証の実装の詳細については、『PowerExchange リファレンスマニュアル』を参照してください。 - ENCRYPT および ENCRYPTLEVEL 文の値が統合サービスマシンの DBMOVER 構成ファイルで定義されている場合、暗号化タイプおよびレベル接続属性に選択する値によって、これらの値はオーバーライドされます。マッピングの暗号化を有効にするには、必ず該当する接続属性を選択します。
[暗号化] レベル	<p>[暗号タイプ] で [RC2] または [DES] を選択していた場合、データ統合サービスの使用する暗号化レベルを指定するために、次に挙げるオプションのいずれかを選択してください。</p> <ul style="list-style-type: none"> - 1. DES および RC2 には 56 ビットの暗号化キーを使用します。 - 2. DES には 168 ビットのトリプル暗号化キーを、RC2 には 64 ビットの暗号化キーを使用します。 - 3. DES には 168 ビットのトリプル暗号化キーを、RC2 には 128 ビットの暗号化キーを使用します。 <p>暗号化タイプを選択しない場合、このオプションは無視されます。</p> <p>デフォルトは 1 です。</p>
ページングサイズ	<p>オプション。ソースシステムが PowerExchange リスナに渡すことができるデータの量。外部アプリケーション、データベース、またはデータ統合サービスのノードがボトルネックになっている場合に、ページングサイズを設定します。低い値を使用するとパフォーマンスが向上します。</p> <p>最小値およびデフォルト値は 0 です。値を 0 にするとパフォーマンスが最大になります。</p>
行として解釈	<p>(オプション) このオプションはページングサイズを行数で表示する場合に選択します。このオプションをクリアすると、ページングサイズがキロバイト単位で表示されます。デフォルトでは、このオプションが選択されていないため、ページングサイズはキロバイトで表示されています。</p>
圧縮	<p>オプション。このオプションはソースデータの圧縮を有効にする場合に選択します。データを圧縮することで、Informatica アプリケーションがネットワークで送信するデータ量を減少させることができます。デフォルトではこのオプションが選択されていないため、圧縮は無効になっています。</p>
オフロード処理	<p>オプション。バルクデータの処理をソースマシンからデータ統合サービスのマシンにオフロードするかどうかを制御します。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - AUTO。オフロード処理を使用するかどうかデータ統合サービスによって決定されます。 - はい。オフロード処理を使用します。 - いいえ。オフロード処理は使用しません。 <p>デフォルトは [AUTO] です。</p>

オプション	説明
ワーカースレッド	オプション。オフロード処理が有効なときにデータ統合サービスがバルクデータを処理するために使用するスレッドの数。最適なパフォーマンスを得るためには、データ統合サービスのマシンで使用可能なプロセッサの数を超えないように、この値を設定します。有効な値は 1~64 です。デフォルトは 0 です。マルチスレッド処理は無効になります。
配列サイズ	(オプション) ワーカースレッドのためのストレージアレイのレコード数。このオプションは、[ワーカースレッド] オプションを 0 より大きな値に設定したときに適用されます。有効な値は 25~100000 です。デフォルトは 25 です。
書き込みモード	<p>オプション。データ統合サービスが PowerExchange リスナにデータを送信するモード。次のうちのいずれかの書き込みモードを選択します。</p> <ul style="list-style-type: none"> - CONFIRMWRITEON。PowerExchange リスナにデータを送信し、応答を待ってから以降のデータを送信します。このオプションはエラーリカバリを優先する場合に選択します。ただし、このオプションを使用するとパフォーマンスが落ちる可能性があります。 - CONFIRMWRITEOFF。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションは、エラーが発生したときにターゲットテーブルを再ロードしてもよい場合に使用します。 - ASYNCHRONOUSWITHFAULTTOLERANCE。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションはエラー検出も有効にします。このオプションは、CONFIRMWRITEOFF のスピードと CONFIRMWRITEON のデータの完全性を結合します。 <p>デフォルトは [CONFIRMWRITEON] です。</p>

Teradata Parallel Transporter 接続のプロパティ

Teradata PT 接続を使用して Teradata テーブルにアクセスします。Teradata PT 接続は、データベースタイプの接続です。Teradata PT 接続は、Administrator ツールまたは Developer tool で作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Teradata PT 接続のプロパティを示します。

プロパティ	説明
名前	<p>接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。</p> <p>~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /</p>
ID	<p>データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。</p>
説明	<p>接続の説明。説明は、765 文字を超えることはできません。</p>
ユーザー名	<p>データベースにアクセスするための適切な書き込み権限を持つ Teradata データベースのユーザー名。</p>

プロパティ	説明
パスワード	Teradata データベースユーザー名のパスワード。
ドライバ名	Teradata JDBC ドライバの名前。
接続文字列	データベースからメタデータにアクセスするために使用する接続文字列。 以下の接続文字列を使用します。 jdbc:teradata://<hostname>/database=<database name>, tmode=ANSI, charset=UTF8

以下の表に、データアクセスのプロパティを示します。

プロパティ	説明
TDPID	Teradata データベースマシンの名前、または IP アドレス。
データベース名	Teradata データベース名。 データベース名を入力しない場合、Teradata PT API はデフォルトのログインデータベース名を使用します。
データコードページ	データベースに関連付けられているコードページ。 Teradata ターゲットへのロードを行うマッピングを実行する場合、Teradata PT 接続のコードページは Teradata ターゲットのコードページと同じでなければなりません。 デフォルトは UTF-8。
Tenacity	Teradata データベース上で最大数の操作が実行されている場合に、Teradata PT API がログインの試行を続ける時間数（時間単位）。 ゼロ以外の正の整数である必要があります。デフォルト値は 4 です。
最大セッション数	Teradata PT API が Teradata データベースとの間で確立するセッションの最大数。 ゼロ以外の正の整数である必要があります。デフォルト値は 4 です。
最小セッション数	Teradata PT API ジョブを継続するために必要な Teradata PT API セッションの最大数。 1～最大セッション数の値までの正の整数である必要があります。デフォルトは 1 です。
スリープ	Teradata データベース上で最大数の操作が実行されている場合に、Teradata PT API がログインを再試行する前に休止する時間数（分単位）。 ゼロ以外の正の整数である必要があります。デフォルト値は 6 です。
TDCH にメタデータ JDBC URL を使用	メタデータアクセスプロパティの接続文字列に指定した JDBC URL を、Teradata Connector for Hadoop（TDCH）が使用する必要があることを示しています。 デフォルトは選択されています。 TDCH がマッピングを実行するときに使用する必要のある別の JDBC URL を入力するには、このオプションをクリアします。

プロパティ	説明
データ暗号化	Windows で、SQL の要求、応答、およびデータの完全なセキュリティ暗号化を有効にします。 UNIX でデータの暗号化を有効にするには、odbc.ini ファイルの DSN にコマンド UseDataEncryption=Yes を追加します。 デフォルトでは無効になっています。
TDCH JDBC URL	TDCH が Teradata マッピングを実行するときに使用する必要のある JDBC URL を入力します。次の形式を使用します。 <code>jdbc:teradata://<hostname>/database=<database name>, tmode=ANSI, charset=UTF8</code> このフィールドは、【TDCH にメタデータ JDBC URL を使用】オプションをクリアにした場合にのみ使用できます。

Twitter 接続プロパティ

Twitter Web サイトからデータを抽出するには、Twitter Web 接続を使用します。Twitter 接続はソーシャルメディアへの接続です。Twitter 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Twitter 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。Twitter を選択します。
OAuth の詳細の有無	OAuth を設定するかどうかを示します。次のうちいずれかの値を選択します。 - [はい]。アクセストークンおよびシークレットがあることを示します。 - [いいえ]。OAuth Utility を起動します。
コンシューマキー	Twitter でアプリケーションを作成するときに取得するコンシューマキー。Twitter はこのキーを使用してアプリケーションを識別します。
コンシューマシークレット	Twitter アプリケーションを作成するときに取得するコンシューマシークレット。Twitter はこのシークレットを使用してコンシューマキーの所有権を確立します。

プロパティ	説明
アクセストークン	OAuth Utility から返されるアクセストークン。Twitter はユーザークレデンシャルの代わりにこのトークンを使用して、保護されたリソースにアクセスします。
アクセスシークレット	OAuth Utility が返すアクセスシークレット。このシークレットがトークンの所有権を確立します。

Twitter Streaming 接続プロパティ

Twitter Web サイトからほぼリアルタイムでデータにアクセスするには、Twitter ストリーミング接続を使用します。Twitter ストリーミング接続は、ソーシャルメディア企業のストリーミング API への接続です。Twitter ストリーミング接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Twitter ストリーミング接続の一般的なプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成するドメイン。
タイプ	接続タイプ。Twitter ストリーミングを選択します。

以下の表に、ホースタイプおよび OAuth 認証のプロパティを示します。

プロパティ	説明
ホースタイプ	ストリーミング API のメソッド。以下のいずれかの方式を指定できます。 - フィルタ。Twitter statuses/filter メソッドは検索条件に一致するパブリックステータスを返します。 - サンプル。Twitter statuses/sample メソッドはすべてのパブリックステータスのランダムサンプルを返します。
コンシューマキー	Twitter でアプリケーションを作成するときに取得するコンシューマキー。Twitter はこのキーを使用してアプリケーションを識別します。
コンシューマシークレット	Twitter アプリケーションを作成するときに取得するコンシューマシークレット。Twitter はこのシークレットを使用してコンシューマキーの所有権を確立します。

プロパティ	説明
OAuth の詳細の有無	OAuth を設定するかどうかを示します。次のうちいずれかの値を選択します。 - [はい]。アクセストークンおよびシークレットがあることを示します。 - [いいえ]。OAuth Utility を起動します。
アクセストークン	OAuth Utility から返されるアクセストークン。Twitter はユーザークレデンシャルの代わりにこのトークンを使用して、保護されたリソースにアクセスします。
アクセスシークレット	OAuth Utility が返すアクセスシークレット。このシークレットがトークンの所有権を確立します。

VSAM 接続のプロパティ

VSAM 接続を使用して VSAM データテーブルにアクセスします。VSAM 接続はフラットファイル接続タイプです。VSAM 接続は Developer ツールで作成します。VSAM 接続は、Administrator ツールまたは Developer ツールで管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、VSAM 接続のプロパティを示します。

オプション	説明
場所	VSAM のデータセットに接続する PowerExchange リスナの場所のノード名ノード名は、PowerExchange のコンフィギュレーションファイル (dbmover.cfg) 内の NODE 文の最初のパラメータで定義されます。
ユーザー名	VSAM データセットに接続する権限のあるユーザー名
パスワード	<p>指定したユーザーのパスワードまたは有効な PowerExchange のパスフレーズ。PowerExchange のパスフレーズは 9～128 文字の範囲で作成します。使用できる文字は次のとおりです。</p> <ul style="list-style-type: none"> - 大文字と小文字 - 0～9 の数字 - スペース - 次の特殊文字: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに単一引用符 (')、二重引用符 (")、通貨記号を含めることはできません。</p> <p>パスフレーズを使用するには、DBMOVER メンバでセキュリティ設定を SECURITY=(1,N)かそれ以上に設定したうえで PowerExchange リスナが実行されていることを確認してください。詳細は、『<i>PowerExchange リファレンスマニュアル</i>』の「SECURITY ステートメント」を参照してください。</p> <p>IBM IRRPHREX の終了で許可される文字は、PowerExchange のパスフレーズで許可されている文字に影響を与えません。</p> <p>注: 有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、パスフレーズを検証のため RACF に渡す場合、100 文字より長いパスフレーズを切り捨てます。</p>

オプション	説明
コードページ	必須。VSAM データセットの読み取りや書き込みに使用するコードページの名前。通常この値は ISO コードページの名前です (ISO-8859-6 など)。
パススルーセキュリティ有効	接続のパススルーセキュリティを有効にする。
暗号化タイプ	<p>オプション。データ統合サービスが使用する暗号化のタイプ。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - なし - RC2 - DES <p>デフォルトは [なし] です。</p> <p>注:</p> <ul style="list-style-type: none"> - Informatica では、暗号化タイプおよびレベル接続プロパティを設定したり、DBMOVER 構成ファイルに ENCRYPT 文と ENCRYPTLEVEL 文を設定する代わりに、Secure Sockets Layer (SSL) 認証を使用することを推奨します。SSL 認証は複数の Informatica 製品で使用されており、より厳格なセキュリティが提供されます。 - PowerExchange ネットワークへの SSL 認証の実装の詳細については、『<i>PowerExchange</i> リファレンスマニュアル』を参照してください。 - ENCRYPT および ENCRYPTLEVEL 文の値が統合サービスマシンの DBMOVER 構成ファイルで定義されている場合、暗号化タイプおよびレベル接続属性に選択する値によって、これらの値はオーバーライドされます。マッピングの暗号化を有効にするには、必ず該当する接続属性を選択します。
[暗号化] レベル	<p>[暗号タイプ] で [RC2] または [DES] を選択していた場合、データ統合サービスの使用する暗号化レベルを指定するために、次に挙げるオプションのいずれかを選択してください。</p> <ul style="list-style-type: none"> - 1. DES および RC2 には 56 ビットの暗号化キーを使用します。 - 2. DES には 168 ビットのトリプル暗号化キーを、RC2 には 64 ビットの暗号化キーを使用します。 - 3. DES には 168 ビットのトリプル暗号化キーを、RC2 には 128 ビットの暗号化キーを使用します。 <p>暗号化タイプを選択しない場合、このオプションは無視されます。</p> <p>デフォルトは 1 です。</p>
ページングサイズ	<p>オプション。ソースシステムが PowerExchange リスナに渡すことができるデータの量。外部アプリケーション、データベース、またはデータ統合サービスのノードがボトルネックになっている場合に、ページングサイズを設定します。低い値を使用するとパフォーマンスが向上します。</p> <p>最小値およびデフォルト値は 0 です。値を 0 にするとパフォーマンスが最大になります。</p>
行として解釈	<p>オプション。このオプションはページングサイズを行数で表示する場合に選択します。このオプションをクリアすると、ページングサイズがキロバイト単位で表示されます。デフォルトでは、このオプションが選択されていないため、ページングサイズはキロバイトで表示されています。</p>
圧縮	<p>オプション。このオプションはソースデータの圧縮を有効にする場合に選択します。データを圧縮することで、Informatica アプリケーションがネットワークで送信するデータ量を減少させることができます。デフォルトではこのオプションが選択されていないため、圧縮は無効になっています。</p>

オプション	説明
オフロード処理	<p>オプション。バルクデータの処理をソースマシンからデータ統合サービスのマシンにオフロードするかどうかを制御します。次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> - AUTO。オフロード処理を使用するかどうかデータ統合サービスによって決定されます。 - はい。オフロード処理を使用します。 - いいえ。オフロード処理は使用しません。 <p>デフォルトは [AUTO] です。</p>
ワーカースレッド	<p>オプション。オフロード処理が有効なときにデータ統合サービスがバルクデータを処理するために使用するスレッドの数。最適なパフォーマンスを得るためには、データ統合サービスのマシンで使用可能なプロセッサの数を超えないように、この値を設定します。有効な値は 1～64 です。デフォルトは 0 です。マルチスレッド処理は無効になります。</p>
配列サイズ	<p>オプション。ワーカースレッドのためのストレージアレイのレコード数。このオプションは、[ワーカースレッド] オプションを 0 より大きな値に設定したときに適用されます。有効な値は 25～100000 です。デフォルトは 25 です。</p>
書き込みモード	<p>オプション。データ統合サービスが PowerExchange リスナにデータを送信する際のモード。次のうちいずれかの書き込みモードを選択します。</p> <ul style="list-style-type: none"> - CONFIRMWRITEON。PowerExchange リスナにデータを送信し、応答を待ってから以降のデータを送信します。このオプションはエラーリカバリを優先する場合に選択します。ただし、このオプションを使用するとパフォーマンスが落ちる可能性があります。 - CONFIRMWRITEOFF。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションは、エラーが発生したときにターゲットテーブルを再ロードしてもよい場合に使用します。 - ASYNCHRONOUSWITHFAULTTOLERANCE。応答を待たずに、PowerExchange リスナにデータを送信します。このオプションはエラー検出も有効にします。このオプションは、CONFIRMWRITEOFF のスピードと CONFIRMWRITEON のデータの完全性を結合します。 <p>デフォルトは [CONFIRMWRITEON] です。</p>

Web Content-Kapow Katalyst 接続プロパティ

Web Content-Kapow Katalyst 接続を使用して Kapow Katalyst のロボットにアクセスします。これは、ソーシャルメディアタイプの接続です。Web Content-Kapow Katalyst 接続は、Administrator ツールまたは Developer ツールで作成および管理できます。

注: 接続プロパティの順序は、表示するツールによって異なる場合があります。

以下の表に、Web content-Kapow Katalyst 接続のプロパティを示します。

プロパティ	説明
名前	接続の名前。この名前では、大文字と小文字が区別されず、ドメイン内で一意にする必要があります。名前は 128 文字以内で指定し、空白および以下の特殊文字は使用できません。 ~ ` ! \$ % ^ & * () - + = { [] } \ : ; " ' < , > . ? /
ID	データ統合サービスが接続の識別に使用する文字列。ID の大文字と小文字は区別されません。255 文字以下で、ドメイン内で一意である必要があります。このプロパティは、接続を作成した後で変更することはできません。デフォルト値は接続名です。
説明	接続の説明。説明は、765 文字を超えることはできません。
場所	接続を作成する Informatica ドメイン。
タイプ	接続タイプ。Web Content-Kapow Katalyst を選択します。
管理コンソールの URL	ロボットがアップロードされる管理コンソールの URL。 URL アドレスの先頭は「http:」または「https:」である必要があります。以下に例を示します。http://localhost:50080
RQL Service のポート	ソケットサービスが RQL Service 用にリッスンするポート番号。 1〜65535 の値を入力します。デフォルトは 50000 です。
ユーザー名	ローカル管理コンソールへのアクセスに必要なユーザー名。
パスワード	ローカル管理コンソールにアクセスするためのパスワード。

Web サービス接続のプロパティ

Web サービスコンシューマトランスフォーメーションを Web サービスに接続するには、Web サービス接続を使用します。

以下の表に、Web サービス接続のプロパティを示します。

プロパティ	説明
ユーザー名	Web サービスに接続するユーザー名。HTTP 認証または WS-Security を有効にした場合は、ユーザー名を入力します。 Web サービスコンシューマトランスフォーメーションに WS-Security ポートが含まれている場合、Web サービスコンシューマトランスフォーメーションは入力ポートを介して動的ユーザー名を受け取ります。データ統合サービスは、接続で定義されているユーザー名をオーバーライドします。
パスワード	ユーザー名に対するパスワード。HTTP 認証または WS-Security を有効にした場合は、パスワードを入力します。 Web サービスコンシューマトランスフォーメーションに WS-Security ポートが含まれている場合、Web サービスコンシューマトランスフォーメーションは入力ポートを介して動的パスワードを受け取ります。データ統合サービスは、接続で定義されているパスワードをオーバーライドします。
エンドポイント URL	アクセス先の Web サービスの URL。データ統合サービスは、WSDL ファイルで定義されている URL をオーバーライドします。 Web サービスコンシューマトランスフォーメーションにエンドポイント URL ポートが含まれている場合、Web サービスコンシューマトランスフォーメーションは入力ポートを介して URL を動的に受け取ります。データ統合サービスは、接続で定義されている URL をオーバーライドします。
タイムアウト	データ統合サービスが、接続を閉じる前に、Web サービスプロバイダからの応答を待機する秒数。
HTTP 認証タイプ	HTTP 経由のユーザー認証のタイプ。次のいずれかの値を選択します。 <ul style="list-style-type: none">- なし。認証しません。- 自動: データ統合サービスが Web サービスプロバイダの認証タイプを選択します。- 基本: Web サービスプロバイダのドメインに対するユーザー名およびパスワードの入力が要求されます。データ統合サービスは、認証のためにユーザー名とパスワードを Web サービスプロバイダに送ります。- ダイジェスト: Web サービスプロバイダのドメインに対するユーザー名およびパスワードの入力が要求されます。データ統合サービスは、ユーザー名とパスワードから暗号化されたメッセージダイジェストを生成し、それを Web サービスプロバイダに送ります。プロバイダはユーザー名とパスワードの一時的な値を生成し、それをドメインコントローラの Active Directory に格納します。その値とメッセージダイジェストが比較されます。これらが一致する場合、Web サービスプロバイダはユーザを認証します。- NTLM: ドメイン名、サーバー名、またはデフォルトのユーザー名とパスワードの入力が要求されます。Web サービスプロバイダは、接続先のドメインに基づいてユーザを認証します。Windows Domain Controller からユーザー名とパスワードが取得され、ユーザが入力したユーザー名とパスワードと比較されます。これらが一致する場合、Web サービスプロバイダはユーザを認証します。NTLM 認証では、暗号化されたパスワードがドメインコントローラのアクティブディレクトリに保存されることはありません。

プロパティ	説明
WS Security タイプ	<p>使用する WS-Security のタイプ。次のいずれかの値を選択します。</p> <ul style="list-style-type: none"> - なし。データ統合サービスは、生成した SOAP リクエストに Web サービスセキュリティヘッダを追加しません。 - PasswordText。データ統合サービスは、生成した SOAP リクエストに Web サービスセキュリティヘッダを追加します。パスワードはクリアテキスト形式で保存されます。 - PasswordDigest。データ統合サービスは、生成した SOAP リクエストに Web サービスセキュリティヘッダを追加します。パスワードはダイジェスト形式で保存され、ネットワーク上の再生攻撃に対する有効な保護を提供します。データ統合サービスが、ナンス値とタイムスタンプをパスワードに組み合わせます。データ統合サービスでは、そのパスワードに SHA ハッシュを適用して base64 エンコーディングでエンコードし、エンコードしたパスワードを SOAP ヘッダー内で使用します。
トラストサーティフィケーションファイル	<p>Web サービスの SSL 証明書の認証時にデータ統合サービスが使用するトラストサーティフィケーションのバンドルを含むファイル。ファイル名と完全ディレクトリパスを入力します。</p> <p>デフォルトは、<Informatica installation directory>/services/shared/bin/ca-bundle.crt です。</p>
クライアント証明書ファイル名	<p>クライアントの認証時に Web サービスで使用するクライアント証明書。Web サービスがデータ統合サービスを認証する必要がある場合にクライアントサーティフィケーションファイルを指定します。</p>
クライアント証明書のパスワード	<p>クライアントサーティフィケーションのパスワード。Web サービスがデータ統合サービスを認証する必要がある場合にクライアント証明書のパスワードを指定します。</p>
クライアント証明書のタイプ	<p>クライアントサーティフィケーションファイルの形式。次のいずれかの値を選択します。</p> <ul style="list-style-type: none"> - PEM: 「.pem」 拡張子のファイルです。 - DER: 「.cer」 または 「.der」 拡張子のファイルです。 <p>Web サービスがデータ統合サービスを認証する必要がある場合にクライアント証明書のタイプを指定します。</p>
プライベートキーファイル名	<p>クライアントサーティフィケーションのプライベートキーファイル。Web サービスがデータ統合サービスを認証する必要がある場合にプライベートキーファイルを指定します。</p>
プライベートキーのパスワード	<p>クライアントサーティフィケーションのプライベートキーのパスワード。Web サービスがデータ統合サービスを認証する必要がある場合にプライベートキーのパスワードを指定します。</p>
プライベートキーのタイプ	<p>プライベートキーのタイプ。サポートされているタイプは PEM です。</p>

データベース接続における識別子のプロパティ

多数のリレーショナルデータベース接続を作成するときには、データベースの識別子のプロパティを設定する必要があります。識別子のプロパティは、データベースにアクセスするためにデータ統合サービスが SQL クエリを生成するときにこのサービスが識別子を区切り文字で囲むかどうかを決定します。

データベース識別子は、データベースオブジェクト名です。テーブル、ビュー、カラム、インデックス、トリガ、プロシージャ、制約、ルールは、識別子を持つことができます。その識別子を使用して、SQL クエリ内でオブジェクトを参照します。データベースは、通常の識別子を持つことも、区切り文字で囲む必要がある区切り記号付き識別子を持つこともできます。

通常の識別子

通常の識別子とは、識別子の形式ルールに従った識別子のことです。通常の識別子は、SQL クエリで使用するときに区切り文字を必要としません。

例えば、以下の SQL 文では通常の識別子 *MYTABLE* と *MYCOLUMN* が使用されています。

```
SELECT * FROM MYTABLE
WHERE MYCOLUMN = 10
```

区切り記号付き識別子

区切り記号付き識別子は識別子の形式ルールに従いません。このため、区切り記号付き識別子は区切り文字で囲む必要があります。

データベースは、以下のタイプの区切り記号付き識別子を使用できます。

予約キーワードを使用する識別子

識別子が予約キーワードを使用する場合、SQL クエリでは区切り文字でその識別子を囲む必要があります。例えば、以下の SQL 文は *ORDER* という名前のテーブルにアクセスします。

```
SELECT * FROM "ORDER"
WHERE MYCOLUMN = 10
```

特殊文字を使用する識別子

識別子が特殊文字を使用する場合、SQL クエリでは区切り文字でその識別子を囲む必要があります。例えば、以下の SQL 文は *MYTABLE\$@* という名前のテーブルにアクセスします。

```
SELECT * FROM "MYTABLE$@"
WHERE MYCOLUMN = 10
```

大文字小文字を区別する識別子

デフォルトでは、IBM DB2、Microsoft SQL Server、および Oracle データベースにおける識別子は、大文字小文字を区別しません。データベースオブジェクト名は大文字で格納されますが、SQL クエリではそれらを参照するのに大文字と小文字のどちらでも使用できます。例えば、以下の SQL 文は *MYTABLE* という名前のテーブルにアクセスします。

```
SELECT * FROM mytable
SELECT * FROM MyTable
SELECT * FROM MYTABLE
```

大文字小文字を区別する識別子を使用するには、SQL クエリでは区切り文字でその識別子を囲む必要があります。例えば、以下の SQL 文は *MyTable* という名前のテーブルにアクセスします。

```
SELECT * FROM "MyTable"
WHERE MYCOLUMN = 10
```

識別子のプロパティ

多数のデータベース接続を作成するときには、データベース識別子のプロパティを設定する必要があります。設定する識別子のプロパティは、データベースが通常の識別子を使用するか、識別子内でキーワードまたは特殊文字を使用するか、それとも大文字小文字を区別する識別子を使用するかによって異なります。

データベース接続で以下の識別子のプロパティを設定します。

SQL 識別子文字

データベースが、SQL クエリで区切り識別子を囲むのに使用する文字のタイプ。使用できる文字は、データベースタイプによって異なります。

データベースで通常識別子が使用される場合、[なし] を選択します。データ統合サービスで SQL クエリが生成される際、識別子の周りに区切り文字は配置されません。

データベースで区切り識別子が使用される場合、文字を選択します。データ統合サービスで SQL クエリが生成される際、区切り識別子はこの文字で囲まれます。

大文字と小文字が混在する識別子をサポート

データベースで大文字と小文字を区別する識別子が使用される場合、有効にします。有効にした場合、データ統合サービスは、すべての識別子を **[SQL 識別子文字]** プロパティに対して選択された文字で囲みます。

Informatica クライアントツールでは、大文字と小文字を正しく使用して識別子を参照する必要があります。例えば、データベース接続を作成する際、大文字と小文字を正しく使用してデータベースユーザー名を入力する必要があります。

[SQL 識別子文字] プロパティが **[なし]** に設定されている場合、**[大文字と小文字が混在する識別子をサポート]** プロパティは無効になっています。

例: データベースが通常の識別子を使用する

この例では、データベースが通常の識別子を使用しています。どの識別子にも予約キーワードや特殊文字は含まれていません。データベースは、大文字小文字を区別しない識別子を使用します。

データベース接続で、**[SQL 識別子文字]** プロパティを **[(なし)]** に設定します。**[SQL 識別子文字]** が **[なし]** に設定されている場合、**[大文字と小文字が混在する識別子をサポート]** プロパティは無効になっています。

データ統合サービスで SQL クエリを生成する場合は、識別子を区切り文字で囲みません。

例: データベースが識別子内でキーワードまたは特殊文字を使用する

この例では、データベースは一部の識別子内でキーワードまたは特殊文字を使用しています。データベースは、大文字小文字を区別しない識別子を使用します。

データベース接続で、識別子のプロパティを以下のように設定します。

1. **[SQL 識別子文字]** プロパティを、データベースが区切り記号付き識別子に使用する文字に設定します。

この例は、プロパティを `"""` (引用符) に設定します。

2. **[大文字と小文字が混在する識別子をサポート]** プロパティをクリアします。

データ統合サービスは、SQL クエリを生成するときに、予約キーワードを使用する識別子または特殊文字を使用する識別子を、選択した文字で囲みます。例えば、データ統合サービスは以下のようなクエリを生成します。

```
SELECT * FROM "MYTABLE$@" /* identifier with special characters enclosed within delimited
                           character */
WHERE MYCOLUMN = 10 /* regular identifier not enclosed within delimited character */
```

例: 大文字小文字を区別する識別子をデータベースが使用する

この例では、大文字小文字を区別する識別子をデータベースが使用しています。このデータベースは、いくつかの識別子でキーワードまたは特殊文字を使う可能性もあれば、これらを使わない可能性もあります。

データベース接続で、識別子のプロパティを以下のように設定します。

1. **[SQL 識別子文字]** プロパティを、データベースが区切り記号付き識別子に使用する文字に設定します。

この例は、プロパティを `"""` (引用符) に設定します。

2. **[大文字と小文字が混在する識別子をサポート]** プロパティを選択します。

データ統合サービスは、SQL クエリを生成するときに、すべての識別子を選択した文字で囲みます。例えば、データ統合サービスは以下のようなクエリを生成します。

```
SELECT * FROM "MyTable"      /* case-sensitive identifier enclosed within delimited character */
WHERE "MYCOLUMN" = 10        /* regular identifier enclosed within delimited character */
```

索引

数字

10 進型のデータ型
概要 [168](#)

A

Adabas 接続
プロパティ [195](#)

B

bigint
計算での使用 [164](#)
高精度処理 [164](#)
フラットファイルへの書き込み [165](#)
式の定数 [165](#)

D

Data Integration Service
選択 [22](#)
DataSift 接続
プロパティ [198](#)
DDL の生成
DDL の生成エラー [70](#)
DDL の生成と実行 [67](#)
decimal
高精度処理 [164](#), [168](#)
Decimal
データ型の変換 [188](#)
Developer ツール
ワークスペースディレクトリ [16](#)
double
高精度処理 [168](#)

F

Facebook 接続
プロパティ [199](#)

G

Greenplum 接続
プロパティ [199](#)

H

HBase 接続
プロパティ [201](#)

HDFS 接続
プロパティ [202](#)
Hive 接続
プロパティ [203](#)
HTTP 接続
プロパティ [209](#)

I

IBM DB2 接続
プロパティ [211](#)
IBM DB2 for i5/OS 接続
プロパティ [214](#)
IBM DB2 for z/OS 接続
プロパティ [218](#)
IMS 接続
プロパティ [221](#)
Informatica Data Services
概要 [15](#)
Informatica Developer
概要 [14](#)
検索 [42](#)
設定 [21](#)
起動 [16](#)
Informatica Marketplace
説明 [21](#)

J

JD Edwards EnterpriseOne 接続
プロパティ [228](#)
JDBC 接続
プロパティ [224](#)

L

LDAP 接続
プロパティ [229](#)
LinkedIn 接続
プロパティ [230](#)

M

Microsoft Azure SQL Data Warehouse 接続
プロパティ [232](#)
MS SQL Server 接続
プロパティ [233](#)

N

NaN
説明 [165](#)
Netezza 接続
プロパティ [237](#)

O

OData 接続
プロパティ [238](#)
ODBC 接続
プロパティ [239](#)
Oracle 接続
プロパティ [241](#)

Q

QNaN
1.への変換#QNaN [165](#)

S

SAP 接続
プロパティ [244](#)
SQL ヒント
Developer ツールでの入力 [73](#)

T

Teradata Parallel Transporter 接続
プロパティ [249](#)
Twitter ストリーミング接続
プロパティ [252](#)
Twitter 接続
プロパティ [251](#)

V

VSAM 接続
プロパティ [253](#)

W

Web content-Kapow Katalyst 接続
プロパティ [255](#)
Web サービス
構成 [134](#)
構成の作成 [134](#)
構成プロパティ [128](#)
Web 接続
プロパティ [209](#)
WSDL データオブジェクト
インポート [84](#)
概要ビュー [85](#)
作成 [85](#)
詳細ビュー [85](#)
スキーマビュー [84](#)
同期 [86](#)

あ

アプリケーション
アプリケーションの検証 [141](#)
アプリケーションの作成 [141](#)
概要 [140](#)
更新 [145](#), [152](#)
再デプロイ [152](#)
作成 [141](#), [147](#)
作成、デプロイ、更新の方法 [146](#)
置換 [145](#)
データ統合サービスへのデプロイ [143](#)
ファイルへのエクスポート [144](#)
ファイルへのデプロイ [144](#)
プロパティ [141](#)
マッピングのデプロイメントのプロパティ [141](#)
アプリケーションアーカイブファイル [144](#)

い

インポート
XML ファイル [156](#)
アプリケーションアーカイブ [152](#)
依存オブジェクト [155](#)
オブジェクト [159](#)
概要 [154](#)

え

エクスポート
XML ファイル [156](#)
アプリケーションアーカイブファイルへ [148](#), [149](#)
依存オブジェクト [155](#)
オブジェクト [156](#)
概要 [154](#)
ファイルへ [150](#)
エラーメッセージ
グループ化 [138](#)
制限 [139](#)

お

オブジェクト
コピー [28](#)
バージョン履歴 [40](#)
ロックされたオブジェクト [34](#)
オブジェクトの依存関係
オブジェクトの依存関係の表示 [136](#)
オブジェクトの依存関係の表示
オブジェクトの依存関係 [136](#)
オブジェクトの依存関係のフィルタリング [137](#)
オブジェクトの依存関係のフィルタリング
オブジェクトの依存関係 [137](#)
[オブジェクトの依存関係] ビュー
データの表示 [135](#)
オブジェクトのチェックアウトとチェックイン [40](#)
オブジェクトのチェックインとチェックアウト [40](#)

か

外部結合のサポート
カスタマイズデータオブジェクト [77](#)
カスタマイズデータオブジェクト
[個別に選択] の使用 [74](#)

カスタマイズデータオブジェクト (続く)

- 書き込みプロパティ [64](#)
- カスタム SQL クエリ [70](#)
- カスタムクエリの作成 [82](#)
- キーの作成 [66](#)
- キーリレーション [63](#)
- キーリレーションの作成 [67](#)
- 個別に選択 [73](#)
- 作成 [65](#)
- 詳細クエリ [71](#)
- 説明 [62](#)
- ソースフィルタの入力 [74](#)
- ソート済みポート [74](#)
- ソート済みポートの使用 [75](#)
- 単純クエリ [71](#)
- デフォルトクエリ [71](#)
- トラブルシューティング [89](#)
- マッピング実行前/実行後 SQL コマンド [81](#)
- マッピング実行前/実行後 SQL コマンドの追加 [82](#)
- ユーザー定義ジョイン [76](#)
- ユーザー定義ジョインの入力 [76](#)
- 予約語ファイル [71](#)
- リレーショナルデータオブジェクトの追加 [66](#)
- リレーショナルリソースの追加 [65](#)
- カスタム SQL クエリ
- カスタマイズデータオブジェクト [70](#)
- 作成 [82](#)
- カスタムクエリ
- Informatica ジョイン構文 [77](#)
- Normal ジョイン構文 [78](#)
- 外部結合のサポート [77](#)
- 右外部結合構文 [81](#)
- レフトアウタージョイン構文 [79](#)
- 監視
- 説明 [139](#)

き

- キーリレーション
- カスタマイズデータオブジェクト [63](#)
- カスタマイズデータオブジェクトの作成 [67](#)
- リレーショナルデータオブジェクト [60](#)
- リレーショナルデータオブジェクト間に作成 [61](#)

く

- クエリビュー
- ヒントの設定 [73](#)
- 区切り記号付き識別子
- データベース接続 [259](#)
- 区切りプロパティ
- フラットファイルデータオブジェクト [94](#)

け

- 検索
- エディタ [45](#)
- 概要 [42](#)
- ビジネス用語集 [43](#)
- モデルリポジトリ [42](#)
- 検証
- エラーメッセージのグループ化 [138](#)
- エラーメッセージの制限 [139](#)
- プレファレンスの設定 [138](#)

こ

- 構成
- トラブルシューティング [135](#)
- 高精度
- Bigint データ型 [164](#)
- Decimal データ型 [164](#)
- 固定幅プロパティ
- フラットファイルデータオブジェクト [94](#)
- コピー
- オブジェクト [28](#)
- 説明 [28](#)
- 個別に選択
- カスタマイズデータオブジェクト [73](#)
- カスタマイズデータオブジェクトでの使用 [74](#)

し

- 識別子
- 区切り記号付き [259](#)
- 通常の [259](#)
- 識別リレーション
- 説明 [121](#)
- シーケンシャル接続
- プロパティ [247](#)
- 自己結合
- カスタム SQL クエリ [70](#)
- ジョイン構文
- Informatica 構文 [77](#)
- Normal ジョイン構文 [78](#)
- カスタマイズデータオブジェクト [77](#)
- 右外部結合構文 [81](#)
- レフトアウタージョイン構文 [79](#)
- 証明書
- 証明書の管理 [87](#)
- 証明書のプロパティ [87](#)
- 信頼されていない証明書 [87](#)
- 信頼されていない証明書の追加 [87](#)

せ

- 整数
- 計算での使用 [164](#)
- フラットファイルへの書き込み [165](#)
- 式の定数 [165](#)
- 接続
- Web サービスのプロパティ [257](#)
- 概要 [47](#)
- 切り替え [47](#)
- 削除 [47](#)
- 作成 [50](#)
- [接続エクスプローラ] ビュー [49](#)
- データベースの識別子のプロパティ [258](#)
- 名前の変更 [47](#)
- 編集 [47](#)

そ

- ソースフィルタ
- 入力 [74](#)
- ソート済みポート
- カスタマイズデータオブジェクト [74](#)
- カスタマイズデータオブジェクトでの使用 [75](#)
- 属性
- リレーション [121](#)

ち

チートシート

説明 [20](#)

チームベース開発

トラブルシューティング [40](#)

つ

通常の識別子

データベース接続 [259](#)

て

データ型

DB2 for i5/OS [170](#)

DB2 for z/OS [170](#)

decimal [168](#)

double [168](#)

IBM DB2 [172](#)

Microsoft SQL Server [175](#)

ODBC [180](#)

Oracle [181](#)

SAP HANA [184](#)

XML [186](#)

暗黙の変換 [188](#)

概要 [162](#)

トランスフォーメーション [163](#)

日付/時刻 [166](#)

非リレーショナル [178](#)

フラットファイル [172](#)

ポート対ポートのデータ変換 [188](#)

データタイプ

Bigint [164](#)

Integer [164](#)

JDBC [174](#)

バイナリ [166](#)

文字列 [170](#)

データの論理ビュー

開発 [111](#)

概要 [109](#)

データビューア

構成 [128](#), [131](#)

構成の作成 [132](#)

構成のトラブルシューティング [135](#)

構成プロパティ [128](#)

データベース接続

識別子のプロパティ [258](#)

データベースのヒント

Developer ツールでの入力 [73](#)

デフォルトの SQL クエリ

表示 [82](#)

デプロイメント

アプリケーションアーカイブファイルへ [148](#), [149](#)

アプリケーションの更新 [145](#)

アプリケーションの置換 [145](#)

概要 [140](#)

データ統合サービスへ [148](#), [149](#)

ファイルへ [150](#)

マッピングプロパティ [141](#)

と

同期

カスタマイズデータオブジェクト [88](#)

同期 (続く)

物理データオブジェクト [88](#)

ドメイン

説明 [23](#)

追加 [21](#)

トラブルシューティング

バージョンニング [40](#)

トランスフォーメーションのデータ型

リスト [163](#)

は

バージョンニング

トラブルシューティング [40](#)

倍精度浮動小数点数型のデータ型

概要 [168](#)

バイナリデータタイプ

概要 [166](#)

パフォーマンスのチューニング

Web サービス設定 [134](#)

データビューア構成の作成 [132](#)

データビューアの構成 [131](#)

マッピング構成 [133](#)

マッピング構成の作成 [134](#)

ひ

非識別リレーション

説明 [121](#)

ビジネス用語

ホットキーのカスタマイズ [44](#)

ルックアップ [44](#)

日付/日時データ型

概要 [166](#)

ビュー

〔接続エクスプローラ〕ビュー [49](#)

説明 [17](#)

非リレーショナルデータ操作

読み取り、書き込み、およびルックアップトランスフォーメーションの作成 [83](#)

ヒント

クエリビュー [73](#)

ふ

フォーマットのプロパティ

フラットファイルデータオブジェクト [93](#)

フォルダ

説明 [27](#)

フォルダー

作成 [27](#)

物理データオブジェクト

カスタマイズデータオブジェクト [62](#)

説明 [57](#)

同期 [88](#)

トラブルシューティング [89](#)

フラットファイルデータオブジェクト [90](#)

リレーショナルデータオブジェクト [58](#)

非リレーショナルデータオブジェクト [82](#)

プライマリキー

カスタマイズデータオブジェクトの作成 [66](#)

リレーショナルデータオブジェクトでの作成 [61](#)

フラットファイルデータオブジェクト

空として作成 [105](#)

カラムのプロパティ [91](#)

フラットファイルデータオブジェクト (続く)

既存のフラットファイルからの作成 [105](#)

区切りプロパティ [94](#)

固定幅プロパティ [94](#)

詳細プロパティ [92](#)

制御ファイルからの作成 [107](#)

説明 [90](#)

全般プロパティ [91](#)

フォーマットのプロパティ [93](#)

ランタイム書き込みプロパティ [98](#)

ランタイム読み取りプロパティ [96](#)

プロジェクト

共有 [23](#)

権限 [24](#)

権限の割り当て [26](#)

作成 [23](#)

説明 [23](#)

フィルタリング [24](#)

プロジェクトの権限

親オブジェクトのアクセスの許可 [26](#)

外部オブジェクトの権限 [25](#)

書き込み権限 [24](#)

権限の付与 [24](#)

セキュリティの詳細の表示 [26](#)

読み取り権限 [24](#)

割り当て [26](#)

依存オブジェクトインスタンス [25](#)

ほ

方法

アプリケーションの作成、デプロイ、更新 [146](#)

ま

マッピング

構成 [128](#), [133](#)

構成の作成 [134](#)

構成のトラブルシューティング [135](#)

構成プロパティ [128](#)

デプロイメントのプロパティ [141](#)

マッピング実行前/実行後 SQL コマンド

カスタマイズデータオブジェクト [81](#)

リレーショナルデータオブジェクトへの追加 [82](#)

も

モデルリポジトリ

オブジェクト [32](#)

オブジェクトとプロパティの検索 [43](#)

オブジェクトのチェックアウトとチェックイン [40](#)

検索 [42](#)

接続 [41](#)

説明 [31](#)

チームベース開発 [36](#)

バージョン管理された [36](#)

バージョン管理されていない [36](#)

バージョンされたオブジェクト [37](#)

追加 [22](#)

モデルリポジトリサービス

更新 [41](#)

ゆ

ユーザー定義ジョイン

Informatica 構文 [77](#)

Normal ジョイン構文 [78](#)

外部結合のサポート [77](#)

カスタマイズデータオブジェクト [76](#)

入力 [76](#)

右外部結合構文 [81](#)

レフトアウタージョイン構文 [79](#)

よ

ようこそページ

説明 [20](#)

予約語ファイル

カスタマイズデータオブジェクト [71](#)

作成 [71](#)

ら

ランタイム書き込みプロパティ

フラットファイルデータオブジェクト [98](#)

ランタイム読み取りプロパティ

フラットファイルデータオブジェクト [96](#)

り

リレーショナル接続

カスタマイズデータオブジェクトへの追加 [65](#)

リレーショナルデータオブジェクト

インポート [60](#)

カスタマイズデータオブジェクトへの追加 [66](#)

キーの作成 [61](#)

キーリレーション [60](#)

キーリレーションの作成 [61](#)

説明 [58](#)

トラブルシューティング [89](#)

る

ルックアップ

ビジネス用語集デスクトップ [44](#)

ろ

ローカルワークスペースディレクトリ

構成 [16](#)

ログ

説明 [137](#)

論理データオブジェクト

作成 [121](#)

説明 [120](#)

属性のリレーション [121](#)

プロパティ [120](#)

例 [110](#)

論理データオブジェクトマッピング

書き込みマッピング [125](#)

作成 [125](#)

タイプ [124](#)

読み取りマッピング [124](#)

論理データオブジェクトモデル

インポート [112](#)

論理データオブジェクトモデル (続く)

作成 [112](#)

説明 [111](#)

例 [110](#)

ワークベンチ

説明 [17](#)

わ

ワークスペースディレクトリ

構成 [16](#)