



Informatica® Data Integration Hub
10.4.1

Developer ガイド

本ソフトウェアおよびマニュアルは、使用および開示の制限を定めた個別の使用許諾契約のもとでのみ提供されています。本マニュアルのいかなる部分も、いかなる手段（電子的複製、写真複製、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。

米政府の権利プログラム、ソフトウェア、データベース、および関連文書や技術データは、米国政府の顧客に配信され、「商用コンピュータソフトウェア」または「商業技術データ」は、該当する連邦政府の取得規制と代理店固有の補足規定に基づきます。このように、使用、複製、開示、変更、および適応は、適用される政府の契約に規定されている制限およびライセンス条項に従うものとし、政府契約の条項によって適当な範囲において、FAR 52.227-19、商用コンピュータソフトウェアライセンスの追加権利を規定します。

Informatica、Informatica ロゴ、Informatica Cloud、PowerCenter、PowerExchange および Data Engineering Integration は、米国およびその他の国における Informatica LLC の商標または登録商標です。Informatica の商標の最新リストは、Web (<https://www.informatica.com/trademarks.html>) にあります。その他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テーション（あるいはその両方）の一部は、第三者が保有する著作権の対象となります。必要な第三者の通知は、製品に含まれています。

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を提供され、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。

本マニュアルの情報は、予告なしに変更されることがあります。このドキュメントで問題が見つかった場合は、infa_documentation@informatica.com までご報告ください。

Informatica 製品は、それらが提供される契約の条件に従って保証されます。Informatica は、商品性、特定目的への適合性、非侵害性の保証等を含めて、明示的または黙示的ないかなる種類の保証をせず、本マニュアルの情報を「現状のまま」提供するものとします。

発行日: 2020-11-09

目次

序文	9
Informatica のリソース	9
Informatica Network	9
Informatica ナレッジベース	9
Informatica マニュアル	9
Informatica 製品可用性マトリックス	10
Informatica Velocity	10
Informatica Marketplace	10
Informatica グローバルカスタマサポート	10
第 1 章 : Data Integration Hub の概要	11
Data Integration Hub の概要	11
Data Integration Hub アーキテクチャ	14
Data Integration Hub Big Data	15
操作コンソール	16
操作コンソールの言語の変更	16
Data Integration Hub トピック	16
Data Integration Hub のパブリケーションとサブスクリプション	17
パブリケーションプロセス	17
サブスクリプションプロセス	19
開発者ユーザーロール	21
第 2 章 : PowerCenter マッピングおよびワークフロー	22
PowerCenter マッピングおよびワークフローの概要	22
PowerCenter ワークフロータイプ	23
バッチワークフロー	23
リアルタイムワークフロー	23
自動 PowerCenter マッピングおよびワークフロー	24
自動 PowerCenter マッピングおよびワークフローのルールとガイドライン	24
自動 PowerCenter マッピングおよびワークフローのログ	25
カスタム PowerCenter マッピングおよびワークフロー	25
サポートされるデータ型	26
カスタム PowerCenter マッピングおよびワークフローのルールとガイドライン	26
PowerCenter バッチワークフローの開発	29
PowerCenter バッチワークフローの開発のルールとガイドライン	29
パブリケーションおよびサブスクリプションプロセス向けの PowerCenter バッチワークフローの開発	29
手順 1。ソースおよびターゲット定義の作成	30
手順 2。マッピングの作成	32
手順 3。PowerCenter ワークフローとセッションの作成	32

手順 4. PowerCenter ワークフローの保存.	33
PowerCenter リアルタイムワークフローの開発.	33
PowerCenter リアルタイムワークフローの開発のルールとガイドライン.	33
パブリケーションリアルタイムワークフローの開発プロセス.	33
手順 1. ソースおよびターゲット定義の作成.	34
手順 2. マッピングの作成.	35
手順 3. PowerCenter のワークフローおよびセッションの作成と保存.	36
手順 4. トピックの作成.	36
手順 5. ワークフローターゲットの名前の変更.	37
手順 6. パブリケーションリアルタイムワークフローの作成.	37

第 3 章 : Data Engineering Integration および Data Engineering Streaming のマッピングとワークフロー. 38

Data Engineering Integration および Data Engineering Streaming のマッピングとワークフローの概要.	38
Data Integration Hub 内の Data Engineering Integration および Data Engineering Streaming のマッピングおよびワークフロー.	39
作業を開始する前に.	40
パブリケーション用の Data Engineering Integration マッピングの開発.	40
手順 1. ソース接続およびターゲット接続を作成する.	41
手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する.	41
手順 3. ソースとターゲットを使用してマッピングを作成する.	41
手順 4. Data Integration Hub パラメータをマッピングに追加する.	41
手順 5. マッピングへの式トランスフォーメーションの追加.	42
手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する.	42
パブリケーション用 Data Engineering Streaming マッピングの開発.	42
手順 1. ソース接続およびターゲット接続を作成する.	43
手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する.	43
手順 3. ソースとターゲットを使用してマッピングを作成する.	43
手順 4. マッピングへの式トランスフォーメーションの追加.	43
手順 5. Java トランスフォーメーションをマッピングに追加する.	43
手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する.	44
サブスクリプション用の Data Engineering Integration マッピングの開発.	44
手順 1. ソース接続およびターゲット接続を作成する.	45
手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する.	45
手順 3. ソースとターゲットを使用してマッピングを作成する.	45
手順 4. Data Integration Hub パラメータをマッピングに追加する.	46
手順 5. フィルタクエリをリーダーオブジェクトに追加する.	46
手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する.	46
パブリケーション用の Data Engineering Integration ワークフローの開発.	46
手順 1. パブリケーション用の Data Engineering Integration マッピングの開発.	47
手順 2. パブリケーションワークフローを作成する.	47
手順 3. パブリケーションワークフローをデプロイする.	48

サブスクリプション用の Data Engineering Integration ワークフローの開発.	48
手順 1: サブスクリプション用の Data Engineering Integration マッピングの開発.	48
手順 2: サブスクリプションワークフローを作成する.	49
手順 3: サブスクリプションワークフローをデプロイする.	50
第 4 章 : Data Quality マッピングおよびワークフロー.	51
Data Quality マッピングおよびワークフローの概要.	51
Data Integration Hub の Data Quality マッピングおよびワークフロー.	52
作業を開始する前に.	52
パブリケーション用の Data Quality マッピングの開発.	52
手順 1. ソース接続およびターゲット接続を作成する.	53
手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する.	53
手順 3. ソースとターゲットを使用してマッピングを作成する.	53
手順 4. Data Integration Hub パラメータをマッピングに追加する.	53
手順 5. マッピングへの式トランスフォーメーションの追加.	54
手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する.	54
サブスクリプション用の Data Quality マッピングの開発.	54
手順 1. ソース接続およびターゲット接続を作成する.	55
手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する.	55
手順 3. ソースとターゲットを使用してマッピングを作成する.	55
手順 4. Data Integration Hub パラメータをマッピングに追加する.	55
手順 5. フィルタトランスフォーメーションをマッピングに追加する.	56
手順 6. フィルタクエリをリーダーオブジェクトに追加する.	56
手順 7. マッピングのランタイム環境を設定し、アプリケーションを作成する.	56
パブリケーション用の Data Quality ワークフローの開発.	56
手順 1. パブリケーション用の Data Quality マッピングを開発する.	57
手順 2. パブリケーションワークフローを作成する.	57
手順 3. パブリケーションワークフローをデプロイする.	58
サブスクリプション用の Data Quality ワークフローの開発.	58
手順 1. サブスクリプション用の Data Quality マッピングを開発する.	58
手順 2. サブスクリプションワークフローを作成する.	59
手順 3. サブスクリプションワークフローをデプロイする.	60
第 5 章 : Informatica クラウドマッピングおよびタスク.	61
Informatica クラウドマッピングおよびタスクの概要.	61
Data Integration Hub の Informatica Cloud マッピング.	62
Informatica Cloud マッピングのルールとガイドライン.	62
Data Integration Hub の Data Integration タスク.	62
Data Integration タスクのルールとガイドライン.	63
Data Integration Hub でのインテリジェント構造モデルの使用.	64
第 6 章 : Data Integration Hub ワークフロー.	65
Data Integration Hub ワークフローの概要.	65

ワークフロー権限.	66
ワークフロー管理.	66
Data Integration Hub ワークフローの作成.	66
Data Integration Hub ワークフローの編集.	70
Data Integration Hub ワークフローの削除.	70
Data Integration Hub ワークフローのプロパティ.	70
ワークフローの全般プロパティ.	71
ワークフローパラメータのプロパティ.	73
ワークフローのイベント属性のプロパティ.	73
ワークフロー権限のプロパティ.	73
第 7 章 : Data Integration Hub トランスフォーメーション.	74
Data Integration Hub トランスフォーメーションの概要.	74
トランスフォーメーションのインストールと登録.	75
トランスフォーメーションの構成.	75
トランスフォーメーションエラーの処理.	76
Data Integration Hub トランスフォーメーションのルールとガイドライン.	76
DX_Add_Document_To_Event トランスフォーメーション.	77
入力ポート.	77
入出力ポート.	78
Data Integration Hub プロパティ.	78
DX_Event_Attribute トランスフォーメーション.	79
入出力ポート.	79
Data Integration Hub プロパティ.	79
DX_Event_Details トランスフォーメーション.	80
入出力ポート.	80
Data Integration Hub プロパティ.	80
DX_Generate_Temporary_File トランスフォーメーション.	81
入出力ポート.	81
Data Integration Hub プロパティ.	82
DX_Notification トランスフォーメーション.	82
入出力ポート.	83
Data Integration Hub プロパティ.	83
DX_Publication_Parameters.	84
入力ポート.	84
出力ポート.	84
DX_Start_Publication トランスフォーメーション.	85
入出力ポート.	85
Data Integration Hub プロパティ.	86
DX_Throw_Error.	86
入力ポート.	86
入出力ポート.	87
Data Integration Hub プロパティ.	87

第 8 章 : オペレーショナルデータストアダッシュボードとレポート.... 88

オペレーショナルデータストアダッシュボードとレポートの概要.....	88
重要業績評価指標 (KPI)	89
デフォルトの KPI.	89
Logi Info Studio のダッシュボードとレポートの構造.....	90
オペレーショナルデータストアダッシュボードのフィルタ.....	90
オペレーショナルデータストアダッシュボード SQL クエリ要素.....	91
オペレーショナルデータストアダッシュボード JavaScript サポートファイル.....	92
オペレーショナルデータストアダッシュボードテーマ変更子.....	93
カスタムダッシュボードの設定.....	93
Logi Info Studio のカスタムダッシュボードとレポート.....	94
Logi Info Studio のインストールと設定.....	95
カスタムダッシュボードの設定.....	95
カスタムダッシュボードでのレポートの作成アプリケーション.....	96
カスタムダッシュボードへのダッシュボードパネルの追加.....	96
B2B Data Exchange でのカスタムダッシュボードのデプロイとテスト.....	96

第 9 章 : Forms Designer..... 97

Forms Designer の概要.....	97
Forms Designer のユーザーインターフェース.....	97
Forms Designer のアクション.....	98
要素のプロパティ.....	99
グループプロパティ.....	100

第 10 章 : Data Integration Hub のパブリケーションおよびサブスクリプションの API..... 101

パブリケーションおよびサブスクリプション API の概要.....	101
Run Publication Subscription REST API	102
Run Publication Subscription REST API の要求.....	102
Run Publication Subscription REST API アクションの応答.....	104
Run Publication Subscription コマンドライン API	104
Data Integration Hub Run Publication Subscription コマンドライン API のコマンド構文... ..	104
Run Publication Subscription コマンドライン API 通知.....	107
Change Publication Subscription Mode REST API.....	107
Change Publication Subscription Mode REST API アクションの応答.....	108
Reprocess Event REST API.....	109
Reprocess Event REST API アクションの応答.....	109
イベントステータス API.....	110
イベントステータス API の応答.....	110
Change Event Status API.....	112
Change Event Status API の応答.....	113

第 11 章 : データ抽出 API	114
データ抽出 API の概要.	114
カタログ API.	114
Data Integration Hub カatalog API の応答.	114
Data Integration Hub イベントビュー.	119
索引	121

序文

『Data Integration Hub 開発者ガイド』を使用して、PowerCenter ワークフロー、Data Engineering Integration マッピング、Data Quality マッピング、および Informatica Cloud タスクを作成して、Data Integration Hub パブリケーションおよびサブスクリプションを処理する方法について学びます。また、Data Integration Hub で処理されるデータに関するレポートも提供します。

Informatica のリソース

Informatica は、Informatica Network やその他のオンラインポータルを通じてさまざまな製品リソースを提供しています。リソースを使用して Informatica 製品とソリューションを最大限に活用し、その他の Informatica ユーザーや各分野の専門家から知見を得ることができます。

Informatica Network

Informatica Network は、Informatica ナレッジベースや Informatica グローバルカスタマサポートなど、多くのリソースへの入口です。Informatica Network を利用するには、<https://network.informatica.com> にアクセスしてください。

Informatica Network メンバーは、次のオプションを利用できます。

- ナレッジベースで製品リソースを検索できます。
- 製品の提供情報を表示できます。
- サポートケースを作成して確認できます。
- 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

Informatica ナレッジベースを使用して、ハウツー記事、ベストプラクティス、よくある質問に対する回答など、製品リソースを見つけることができます。

ナレッジベースを検索するには、<https://search.informatica.com> に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

Informatica マニュアルポータルでは、最新および最近の製品リリースに関するドキュメントの膨大なライブラリを参照できます。マニュアルポータルを利用するには、<https://docs.informatica.com> にアクセスしてください。

製品マニュアルに関する質問、コメント、ご意見については、Informatica マニュアルチーム (infa_documentation@informatica.com) までご連絡ください。

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 PAM は、<https://network.informatica.com/community/informatica-network/product-availability-matrices> で参照できます。

Informatica Velocity

Informatica Velocity は、Informatica プロフェッショナルサービスが開発したヒントとベストプラクティスのコレクションで、多数のデータ管理プロジェクトから得た実体験に基づいています。Informatica Velocity には、世界中の組織と連携してデータ管理ソリューションを計画、開発、デプロイ、管理する Informatica コンサルタントによる集合知を表しています。

Informatica Velocity リソースには、<http://velocity.informatica.com> からアクセスしてください。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 製品を拡張したり強化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。Marketplace で、Informatica デベロッパーやパートナーからの多数のソリューションを活用すれば、生産性を向上したり、プロジェクトでの実装時間を短縮したりできます。Informatica Marketplace は、<https://marketplace.informatica.com> からアクセスしてください。

Informatica グローバルカスタマサポート

電話または Informatica Network からグローバルサポートセンターに連絡できます。

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 (<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>) を参照してください。

Informatica Network でオンラインサポートリソースを見つけるには、<https://network.informatica.com> にアクセスし、eSupport オプションを選択します。

第 1 章

Data Integration Hub の概要

この章では、以下の項目について説明します。

- [Data Integration Hub の概要, 11 ページ](#)
- [Data Integration Hub アーキテクチャ, 14 ページ](#)
- [Data Integration Hub Big Data, 15 ページ](#)
- [操作コンソール, 16 ページ](#)
- [Data Integration Hub トピック, 16 ページ](#)
- [Data Integration Hub のパブリケーションとサブスクリプション, 17 ページ](#)
- [開発者ユーザーロール, 21 ページ](#)

Data Integration Hub の概要

Data Integration Hub は、組織内の異なるアプリケーション間でデータの共有と同期に使用できるアプリケーション統合ソリューションです。

データを Data Integration Hub にパブリッシュするには、まず、管理するデータセット（販売、顧客、注文など）を定義します。データセットは、トピックを定義することによって定義します。トピックは、Data Integration Hub がパブリケーションリポジトリに格納するデータの構造と、データを格納するパブリケーションリポジトリのタイプを定義します。Data Integration Hub では、さまざまなデータセットを表すトピックを複数管理できます。アプリケーションは、トピックにデータをパブリッシュし、トピックによって表されるデータセットにサブスクライブします。

複数のアプリケーションが同じトピックにパブリッシュできます。例えば、別々の注文アプリケーションが、それぞれの注文を同じ注文トピックにパブリッシュできます。複数のサブスクライバが 1 つのトピックのデータをコンシュームすることができます。サブスクライブするアプリケーションは、定義されているスケジュールに基づき、異なる形式と待ち時間でデータをコンシュームすることができます。

Data Integration Hub は、アプリケーションがトピックにパブリッシュするデータを、Data Integration Hub パブリケーションリポジトリに格納します。Data Integration Hub は、すべてのサブスクライバがデータをコンシュームし、保持期限が切れるまでデータをパブリケーションリポジトリに保持し、その後パブリケーションリポジトリからデータを削除します。

アプリケーションは、PowerExchange (R) アダプタおよび Informatica Intelligent Cloud ServicesSM コネクタを使用して、データベーステーブル、ファイル、Informatica がサポートする任意のソースなどの、さまざまなソースからのデータを共有します。各アプリケーションは、さまざまなトピックのパブリッシャおよびサブスクライバとして機能します。

パブリケーションは、特定トピックに対してパブリッシュします。パブリケーションは、データソースタイプと、アプリケーションがパブリッシュするデータを Data Integration Hub が取得する場所を定義します。サブ

スクリプションは、1つ以上のトピックをサブスクライブします。サブスクリプションは、データターゲットタイプと、パブリッシュされたデータを Data Integration Hub が送信するサブスクライブするアプリケーション内の場所を定義します。

パブリケーションまたはサブスクリプションを作成するときには、自動 Data Integration Hub マッピングまたはカスタム Data Integration Hub マッピングを使用することを選択できます。Data Integration Hub は、トピックで定義されるデータ構造に基づいて自動マッピングを作成します。Data Integration Hub のカスタムマッピングは、開発者がパブリケーションまたはサブスクリプション用に作成および保持する、PowerCenter (R) ワークフロー、Data Engineering Integration マッピング、またはデータ統合タスクに基づいています。

Data Integration Hub 演算子は Enterprise Data Catalog を使用して、既存の Data Integration Hub オブジェクトを検出および利用し、エンタープライズ内の他のエンティティでのリネージュおよび影響を理解します。

例

大手の小売チェーンのためにデータセンターを管理しているとします。メインオフィスには複数のアプリケーションがあります。アプリケーションはオンプレミスまたはクラウドにあります。各小売支店には、POS (point-of-sale) アプリケーションと在庫アプリケーションがあります。アプリケーションおよび支店では、次のデータが必要です。

顧客サービスアプリケーション

最新の顧客注文データが必要です。

販売アプリケーション

最新の製品販売データが必要です。

マーケティングアプリケーション

週次取引レポートが必要です。

アカウンティングアプリケーション

月次取引レポートが必要です。

支店アプリケーション

最新の在庫データと価格データが必要です。

ビジネスインテリジェンス (BI) アプリケーション

販売およびマーケティングデータと、企業の Web サイトのユーザー操作データに関して、直前の 12 か月間の週次レポートが必要です。

Data Integration Hub を使用すると、以下の場合に対処できます。

製品のカatalogと価格を共有する。

販売部署からの製品価格の最新情報を次のように各支店と共有できます。

1. 製品トピックを作成します。
2. Product Information Management (PIM) アプリケーション用に、製品の詳細と価格を製品トピックにパブリッシュするパブリケーションを定義し、毎日データをパブリッシュするようにスケジュールを設定します。
3. 各支店アプリケーション用に、製品トピックに対するサブスクリプションを定義し、パブリッシュされたデータが Data Integration Hub で使用できるようになった時点でそのデータを使用するようにサブスクリプションを設定します。

毎日の販売詳細を共有する。

店舗から受け取る毎日の販売詳細を、主要販売アプリケーションと顧客サービスアプリケーションで次のように共有できます。

1. 販売トピックを作成します。

2. 各支店アプリケーション用に、販売トピックに対するパブリケーションを定義し、毎日公開するようにスケジュールを設定します。
3. 販売アプリケーション用に、販売トピックに対するサブスクリプションを定義し、データが公開される時点でそのデータを使用するようにスケジュールを設定します。
4. 顧客サービスアプリケーション用に、販売トピックに対するサブスクリプションを定義し、週に 1 度データをコンシュームするようにスケジュールを設定します。

Salesforce の取引の詳細を共有します。

Salesforce クラウドアプリケーションの取引の詳細をマーケティングアプリケーションやアカウンティングアプリケーションと次のように共有できます。

1. 取引トピックを作成します。
2. Salesforce アプリケーション用に、取引トピックに対するクラウドパブリケーションを定義し、毎週パブリッシュするようにスケジュールを設定します。
3. マーケティングアプリケーション用に、取引トピックに対するサブスクリプションを定義し、週に 1 度データをコンシュームするようにスケジュールを設定します。
4. アカウンティングアプリケーション用に、取引トピックに対するサブスクリプションを定義し、月に 1 度データをコンシュームするようにスケジュールを設定します。

ビジネスインテリジェンスデータを共有します。

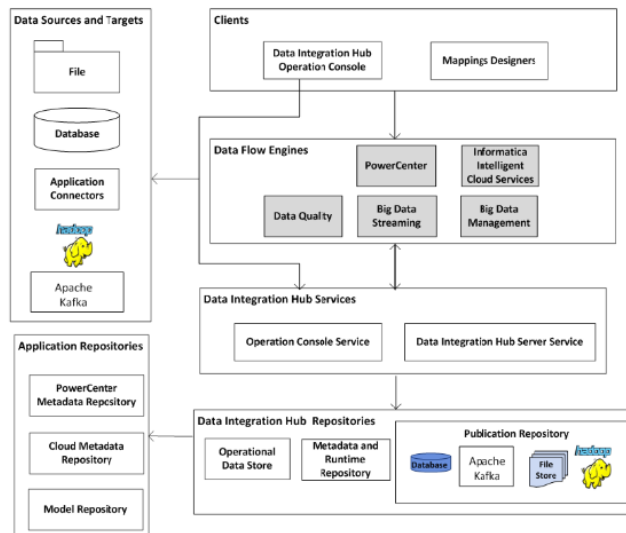
Salesforce の販売データ、マーケティングアプリケーションのマーケティングデータ、および企業の Web サイトのユーザー操作データを、BI アプリケーションを使用して、次のようにして共有できます。

1. Hadoop パブリケーションリポジトリを使用してビジネスインテリジェンストピックを作成し、パブリケーションデータの保持期間を 365 日に設定します。
2. Salesforce アプリケーション用に、ビジネスインテリジェンストピックに対するクラウドパブリケーションを定義し、週に 1 度パブリッシュするようにスケジュールを設定します。
3. マーケティングアプリケーション用に、ビジネスインテリジェンストピックに対するパブリケーションを定義し、週に 1 度パブリッシュするようにスケジュールを設定します。
4. 企業の Web サイトアプリケーション用に、ビジネスインテリジェンストピックに対するクラウドパブリケーションを定義し、週に 1 度パブリッシュするようにスケジュールを設定します。
5. BI アプリケーション用に、ビジネスインテリジェンストピックに対する集計済みサブスクリプションを定義し、週に 1 度データをコンシュームするようにスケジュールを設定します。

Data Integration Hub アーキテクチャ

Data Integration Hub 環境は、ユーザーインタフェースクライアント、データフローエンジン、Data Integration Hub サービスおよびリポジトリ、外部メタデータリポジトリから構成されます。

次の図は、Data Integration Hub コンポーネントを示しています。



Data Integration Hub は、以下のコンポーネントから構成されます。

Data Integration Hub 操作コンソール Web クライアント

アプリケーション、トピック、パブリケーション、およびサブスクリプションの管理と、パブリケーション、サブスクリプション、およびイベントの監視を行うユーザーインタフェース。管理者は、ユーザーとシステム設定の管理にも操作コンソールを使用します。開発者は、操作コンソールを使用して Data Integration Hub ワークフローおよび接続を管理します。

Mapping Designer クライアント

ソースとターゲットの定義、カスタムマッピングの構築、およびワークフローとタスクの作成を行うユーザーインタフェース。カスタムマッピングを使用する場合は、Mapping Designer を使用します。

データフローエンジン

パブリケーションソースからのデータ取得とサブスクリプションターゲットへのデータ送信を行うエンジン。異なる事例には異なるフローエンジンを使用できます。例えば、オンプレミスアプリケーションに対してパブリッシュおよびサブスクライブするには PowerCenter を使用し、クラウドアプリケーションに対してパブリッシュおよびサブスクライブするには Informatica Intelligent Cloud Services を使用します。

Data Integration Hub 操作コンソールサービス

ユーザーが操作コンソールで実行するアクションを処理し、公開されたデータセットの構造をパブリケーションリポジトリに作成するサービス。

Data Integration Hub サーバーサービス

パブリケーションとサブスクリプションの Data Integration Hub ワークフローを開始および監視するサービス。

Data Integration Hub パブリケーションリポジトリ。

サブスクライバがデータを使用するまでの間、公開されたデータを格納するデータベース。データの保持期限に達した後、Data Integration Hub はパブリケーションリポジトリからデータを削除します。

Data Integration Hub **メタデータリポジトリ**

Data Integration Hub アプリケーション、トピック、パブリケーション、サブスクリプション、およびイベントのメタデータを格納するデータベース。

オペレーショナルデータストア

レポート目的のための集計情報が含まれるリポジトリ。Data Integration Hub ダッシュボードと Data Integration Hub のレポートコンポーネントをインストールするときには、提供するデータベース接続詳細に基づき、Data Integration Hub によってオペレーショナルデータストアリポジトリが作成されます。

PowerCenter **メタデータリポジトリ**

PowerCenter マッピング、ワークフロー、およびトランスフォーメーションのメタデータを格納するデータベース。

クラウドメタデータリポジトリ

クラウドマッピングおよびタスクのメタデータを格納するデータベース。

モデルリポジトリサービス

Data Engineering Integration および Data Quality のマッピングおよびトランスフォーメーションのメタデータを格納するデータベース。

データソースとデータターゲット

データの公開と使用に使用するソースとターゲット。以下のタイプのソースとターゲットを使用できます。

- データベース。テーブルとカラム。
- ファイル。バイナリ、テキスト、または非構造化ファイル。
- アプリケーションコネクタ。アプリケーションの接続オブジェクト。カスタムマッピングを使用するときに使用できます。
- Hadoop。Hadoop 分散ファイルシステム (HDFS) および Hive データウェアハウス。

Data Integration Hub Big Data

Data Integration Hub を使用して、大量のデータ、データストリーム、および保存期間が長期間にわたるデータにパブリッシュおよびサブスクライブします。例えば、一定期間にわたって確認する必要があるビジネスインテリジェンスデータを Data Integration Hub Hadoop パブリケーションリポジトリに保存したり、Hadoop 分散ファイルシステム (HDFS) や Hive データウェアハウスに対してパブリッシュおよびサブスクライブを実行したりします。

データがすべてのサブスクライバによって消費された後に、パブリッシュ済みデータを Hadoop パブリケーションリポジトリに保持する場合は、パブリッシュ済みデータをリポジトリから削除しないように、Data Integration Hub を設定できます。

Data Integration Hub でビッグデータをパブリッシュおよび消費する際には、自動マッピングとカスタムマッピングの両方を使用できます。カスタムマッピングによるパブリケーションには、Informatica Data Engineering Integration マッピングおよびワークフローと、Informatica Data Engineering Streaming マッピングを使用できます。カスタムマッピングによるサブスクリプションには、Informatica Data Engineering Integration マッピングおよびワークフローを使用します。

操作コンソール

操作コンソールユーザーインターフェースを使用して、アプリケーション、トピック、パブリケーション、サブスクリプションの管理と、パブリケーション、サブスクリプション、イベントの監視を行います。管理者は、ユーザーとシステム設定の管理にも操作コンソールを使用します。開発者は、ワークフローと接続を管理するために操作コンソールを使用します。

操作コンソールは、英語または日本語で表示できます。表示言語は切り替えることができます。

操作コンソールには、次の 2 つの領域があります。

ナビゲータ

操作コンソールで実行できるタスク間を移動するには、ナビゲータを使用します。ナビゲータは、操作コンソールの左ペインに表示されます。

現在のページ

ナビゲータで選択したタスクを実行するメイン作業領域。現在のページは、操作コンソールの右ペインに表示されます。

操作コンソールの言語の変更

操作コンソールは、英語または日本語で表示できます。表示言語は切り替えることができます。

1. Data Integration Hub にアクセスするブラウザで、言語を目的の言語に設定します。
2. **【ヘルプ】** リンクをクリックすると、オンラインヘルプが英語で表示されます。日本語のオンラインヘルプを表示するには、次の URL にアクセスします。

`http(s)://<host>:<port>/dih-help-ja`

場所：

- <host>は、Data Integration Hub サーバーの IP アドレスです。
- <port>は、Data Integration Hub サーバーのポート番号です。

例：

`https://dih-releases:19443/dih-help-ja/`

Data Integration Hub トピック

Data Integration Hub トピックは、Data Integration Hub で公開および使用されるデータドメインを表すエンティティです。トピックは、標準のデータ構造およびデータ保持期間などの追加のデータ定義を定義します。

例えば、販売データを表す Sales トピックなどがあります。組織内のすべての店舗のアプリケーションは、販売データを Sales トピックにパブリッシュします。経理アプリケーションは Sales トピックにサブスクライブして、すべての店舗からパブリッシュされた販売データをコンシュームします。フィルタが適用されている場合は、特定の店舗からパブリッシュされたデータをコンシュームできます。

Data Integration Hub で公開および消費されるデータのパブリケーションとサブスクリプションを定義する前に、Data Integration Hub パブリケーションリポジトリで Data Integration Hub に公開されるデータを保持する標準構造を定義する必要があります。標準の構造は、トピックを定義するときに定義します。異なるソースのデータセットを表す複数のトピックを定義できます。

Data Integration Hub のパブリケーションとサブスクリプション

パブリケーションおよびサブスクリプションは、アプリケーションが Data Integration Hub にデータを公開する方法、およびアプリケーションが Data Integration Hub のデータを使用する方法を定義するエンティティです。パブリケーションは定義されたトピックにデータを公開し、サブスクリプションはトピックにサブスクライブします。

パブリケーションおよびサブスクリプションはデータパブリケーションまたはデータ使用のデータフローおよびスケジュールを制御します。アプリケーションにはパブリッシャおよびサブスクライバを使用できます。同じトピックに対して複数のアプリケーションからパブリッシュできます。複数のアプリケーションが同じトピックのデータを使用できます。

自動、カスタム、およびモジュラーパブリケーションおよびサブスクリプションを使用して、データのパブリッシュやコンシュームを行うことができます。さまざまなデータソースに対してパブリッシュおよびサブスクライブできます。公開プロセスおよび使用プロセスは完全に分離されているため、公開ソースおよび使用ターゲットに同じデータタイプを指定する必要はありません。例えば、ファイルからデータを公開し、そのデータを使用してデータベースに格納することができます。

自動パブリケーションおよびサブスクリプションでは、パブリッシュやサブスクライブを、リレーショナルデータベース、ファイル、クラウドアプリケーションに対して行ったり、REST API を使用して行ったりすることができます。

カスタムパブリケーションおよびサブスクリプションでは、オンプレミスアプリケーションに対してパブリッシュおよびサブスクライブすることができます。

モジュラーパブリケーションおよびサブスクリプションでは、クラウドアプリケーションに対してパブリッシュおよびサブスクライブすることができます。

パブリケーションプロセス

パブリケーションプロセスでは、パブリッシャからのデータの取得、マッピングやタスクといった関連するマップパーの実行、Data Integration Hub パブリケーションリポジトリ内の関連トピックへのデータの書き込みなどが行われます。パブリケーションプロセスが完了すると、サブスクライバでは、パブリケーションリポジトリからパブリッシュされたデータをコンシュームできます。

パブリケーションプロセスは、パブリケーションタイプによって異なります。

- 自動パブリケーションでは、Data Integration Hub ワークフローを、PowerCenter バッチワークフローに基づいて、または REST API を通じて実行できます。
- カスタムパブリケーションでは、Data Integration Hub ワークフロー（PowerCenter バッチワークフロー、PowerCenter リアルタイムワークフロー、Data Engineering Integration マッピングまたはワークフロー、Data Engineering Streaming マッピング、あるいは Data Quality マッピングまたはワークフローに基づく）か、Informatica Intelligent Cloud Services タスクのいずれかを実行できます。
- モジュラーパブリケーションでは、Informatica Intelligent Cloud Services マッピングを実行します。

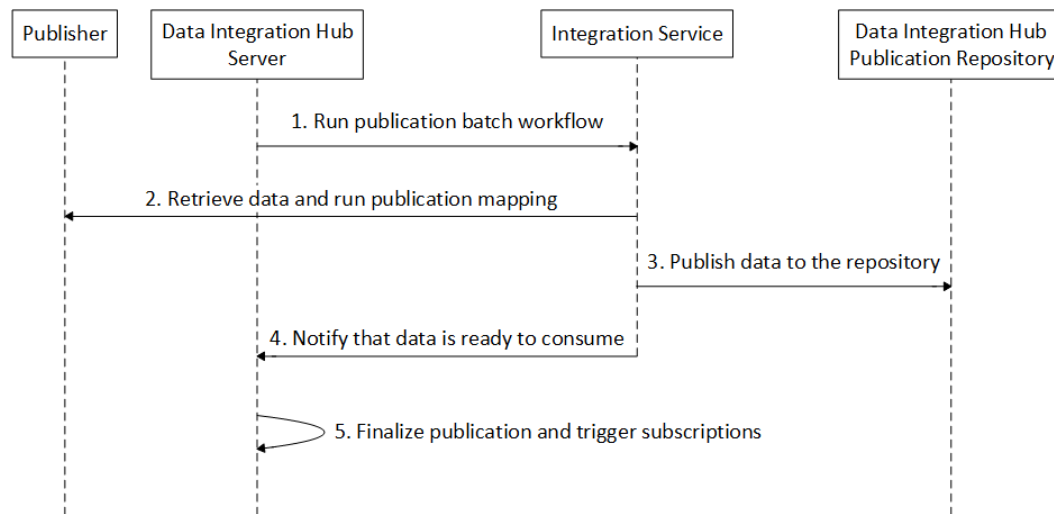
バッチワークフローを使用したパブリケーションプロセス

Data Integration Hub バッチワークフローを実行するパブリケーションのパブリケーションプロセスには、以下の段階があります。

1. パブリッシャがデータをパブリッシュする準備が完了すると、Data Integration Hub サーバーはパブリケーションバッチワークフローを実行し、関連する統合サービス（PowerCenter 統合サービスまたはデータ統合サービス）に要求を送信します。

2. 統合サービスがパブリッシャーからデータを抽出し、そのデータに対して自動マッピングまたはカスタムマッピングを実行します。
3. 統合サービスが、そのデータを Data Integration Hub パブリケーションリポジトリに書き込みます。
4. 統合サービスは、パブリッシュされたデータがサブスクライバに提供できる状態であることを Data Integration Hub サーバーに伝えます。
5. Data Integration Hub サーバーがパブリケーションイベントのステータスを完了に変更し、サブスクリプション処理をトリガします。

次の図は、バッチワークフローを実行するパブリケーションのパブリケーションプロセスの主な段階を示しています。



リアルタイムワークフローを使用したパブリケーションプロセス

Data Integration Hub リアルタイムワークフローを実行するパブリケーションのパブリケーションプロセスには、以下の段階があります。

1. 開発者がリアルタイムワークフローを実行します。ワークフローにより、Data Integration Hub パブリケーションリポジトリ内の関連テーブルにデータが書き込まれます。
2. Data Integration Hub サーバーがスケジュールされたプロセスをトリガし、Data Integration Hub パブリケーションリポジトリ内の関連テーブル内に新しいデータがないかを確認します。
3. 新しいデータが見つかったら、Data Integration Hub がデータのパブリケーション ID とパブリケーション日付を更新してデータの使用準備ができたことを示すと同時に、Data Integration Hub リポジトリ内にパブリケーションイベントを作成します。
4. Data Integration Hub サーバーがパブリケーションイベントのステータスを完了に変更し、サブスクリプション処理をトリガします。

Data Integration タスクによるパブリケーションプロセス

Data Integration タスクを実行するパブリケーションのパブリケーションプロセスには、以下の段階があります。

1. スケジュールに従って、または外部 API によってパブリケーションがトリガされると、Data Integration Hub サーバーは、Informatica Intelligent Cloud Services REST API によってパブリケーション用に定義された Data Integration タスクをトリガします。

2. パブリケーションプロセスは、Data Integration Hub クラウドコネクタを使用してデータを Data Integration Hub に書き込みます。
3. Data Integration Hub サーバーがパブリケーションイベントのステータスを完了に変更し、サブスクリプション処理をトリガします。

データドリブンパブリケーションのパブリケーションプロセス

データドリブンパブリケーションのパブリケーションプロセスには、次のステージが含まれます。

1. データドリブンパブリケーションを作成した後、パブリケーションを実行する POST 要求を作成します。
2. 要求を送信すると、Data Integration Hub はパブリッシュ済みデータを要求から直接 Data Integration Hub パブリケーションリポジトリ（パブリケーションで定義したトピック）に転送します。
3. Data Integration Hub は、パブリケーションに定義されたイベントのグループ化に基づいて、データドリブンパブリケーションイベントを作成します。
 - グループ化時間がゼロに設定されている場合（つまり、パブリケーションにグループ化が定義されていない場合）、Data Integration Hub は、データがパブリケーションリポジトリにパブリッシュされるたびにイベントを作成します。
 - グループ化時間を定義すると、Data Integration Hub は、パブリケーションを含む各グループ化期間の最後にイベントを作成します。例えば、パブリケーションを 10 秒ごとにグループ化するようにパブリケーションを設定すると、10 秒間にパブリケーションリポジトリにデータがパブリッシュされた場合、Data Integration Hub は 10 秒ごとにイベントを作成します。

サブスクリプションプロセス

サブスクリプションプロセスでは、Data Integration Hub サブスクリプションリポジトリからの必要なデータの取得、マッピングやタスクといった関連するマッパーの実行、1 つ以上のサブスクライバターゲットへのデータの書き込みなどが行われます。Data Integration Hub は、トピックの保持期限が切れるまでそのデータをサブスクリプションリポジトリに保持します。

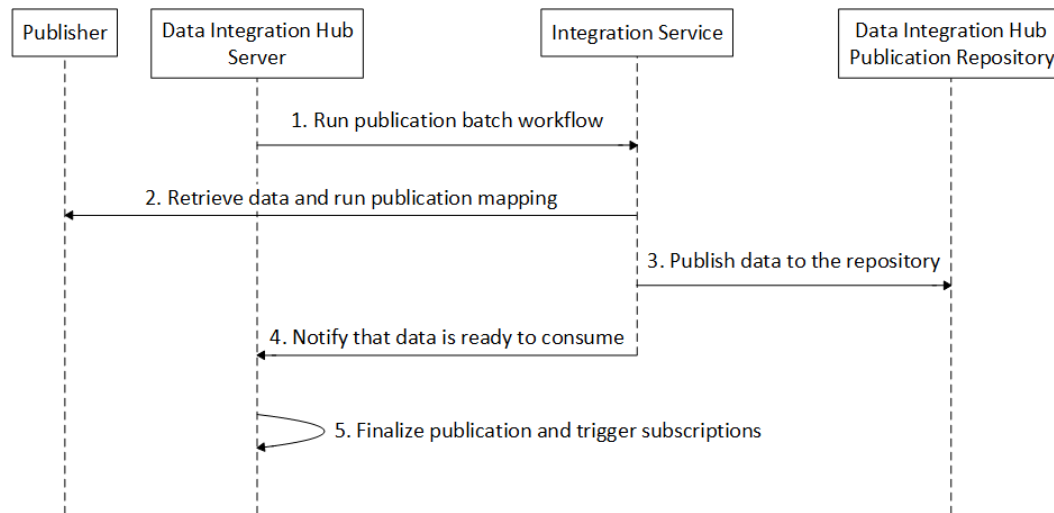
- 自動サブスクリプションでは、Data Integration Hub ワークフローを、PowerCenter バッチワークフローに基づいて、または REST API を通じて実行できます。
- カスタムサブスクリプションでは、Data Integration Hub ワークフロー（PowerCenter バッチワークフロー、Data Engineering Integration マッピングまたはワークフロー、Data Engineering Streaming マッピング、あるいは Data Quality マッピングまたはワークフローに基づく）か、Informatica Intelligent Cloud Services タスクのいずれかを実行できます。
- モジュラーサブスクリプションでは、Informatica Intelligent Cloud Services マッピングを実行します。

バッチワークフローを使用したサブスクリプションプロセス

Data Integration Hub バッチワークフローを実行するサブスクリプションのサブスクリプションプロセスには、以下の段階があります。

1. サブスクライバ向けにパブリケーションの準備ができると、Data Integration Hub サーバーはサブスクリプションバッチワークフローを実行し、関連する統合サービス（PowerCenter 統合サービスまたはデータ統合サービス）に要求を送信します。
2. 統合サービスが Data Integration Hub パブリケーションリポジトリからデータを抽出し、そのデータに対して自動マッピングまたはカスタムマッピングを実行します。
3. 統合サービスが、必要データをサブスクライバに送信します。

4. 必要なパブリッシュ済みデータをサブスクライバがコンシュームすると、統合サービスが Data Integration Hub サーバーに通知します。
 5. Data Integration Hub サーバーが、サブスクリプションイベントのステータスを完了に変更します。
- 次の図は、各サブスクリプションのサブスクリプションプロセスの主な段階を示しています。



Data Integration タスクによるサブスクリプションプロセス

Data Integration タスクを実行するサブスクリプションのサブスクリプションプロセスには、以下の段階があります。

1. サブスクライバ向けにパブリケーションの準備ができると、Data Integration Hub サーバーは、Informatica Intelligent Cloud Services REST API によってサブスクリプション用に定義された Data Integration タスクをトリガします。
2. サブスクリプションプロセスは、Data Integration Hub クラウドコネクタを使用してデータを Data Integration Hub から読み取ります。
3. Data Integration タスクは、Data Integration Hub からデータを読み取り、そのデータをクラウドアプリケーションに書き込みます。
4. Data Integration Hub サーバーが、サブスクリプションイベントのステータスを完了に変更します。

データドリブンサブスクリプションのサブスクリプションプロセス

データドリブンサブスクリプションのサブスクリプションプロセスには、次のステージが含まれます。

1. データドリブンサブスクリプションのプロパティを設定する場合、Data Integration Hub パブリケーションリポジトリのサブスクリプションで定義したトピックからデータをコンシュームできる状態になった際の Data Integration Hub による通知の送信先 URL を入力します。
2. サブスクリプションを実行する POST 要求を作成し、Data Integration Hub パブリケーションリポジトリ（サブスクリプションで定義したトピック）からデータを取得します。
3. Data Integration Hub からデータがトピックからコンシュームできる状態になったという通知が送信されると、サブスクリプションを実行しデータを取得するという要求を送信します。

開発者ユーザーロール

Data Integration Hub の開発者は、カスタムパブリケーションおよびサブスクリプションや監視ルール用に、ワークフロー、マッピング、およびタスクの開発を担当します。

カスタムパブリケーションおよびサブスクリプション

開発者は、PowerCenter ワークフロー、Data Engineering Integration マッピング、Data Quality マッピング、または Informatica Cloud タスクを使用して、カスタムパブリケーションおよびサブスクリプションのマッピングを定義できます。パブリケーションでは、ワークフロー、マッピング、またはタスクが、パブリッシュ元のアプリケーションから Data Integration Hub パブリケーションリポジトリへのフローを定義します。サブスクリプションでは、ワークフロー、マッピング、またはタスクが、Data Integration Hub パブリケーションリポジトリからサブスクライブ元アプリケーションへのフローを定義します。

PowerCenter ワークフロー

開発者は、PowerCenter Designer でワークフローを作成します。次に、Data Integration Hub ワークフローを操作コンソールで作成し、PowerCenter ワークフローを Data Integration Hub ワークフローに割り当てます。ワークフローのパラメータのカスタマイズには、Data Integration Hub Forms Designer を使用できます。

次に、オペレータがオンプレミスアプリケーション用にパブリケーションまたはサブスクリプションを作成するときに、Data Integration Hub ワークフローを選択します。

Data Engineering Integration マッピングおよび Data Engineering Streaming マッピング

開発者は、Developer tool でマッピングを作成します。次に、Data Integration Hub ワークフローを操作コンソールで作成し、Data Engineering Integration または Data Engineering Streaming マッピングを Data Integration Hub ワークフローに割り当てます。ワークフローパラメータは、Forms Designer でカスタマイズできます。

次にオペレータが、ビッグデータのパブリケーションまたはサブスクリプションを作成するときに、Data Integration Hub ワークフローを選択します。

Data Quality マッピング

開発者は、Developer tool でマッピングを作成します。次に、Data Integration Hub ワークフローを操作コンソールで作成し、Data Quality マッピングを Data Integration Hub ワークフローに割り当てます。ワークフローパラメータは、Forms Designer でカスタマイズできます。

次に、オペレータがオンプレミスアプリケーション用にパブリケーションまたはサブスクリプションを作成するときに、Data Integration Hub ワークフローを選択します。

Informatica Cloud タスク

開発者は、Informatica Cloud でタスクを作成します。次に、オペレータが、クラウドアプリケーション用にパブリケーションまたはサブスクリプションを作成するときに、タスクを選択します。

監視ルール

開発者は、PowerCenter Designer でワークフローを作成します。次に、Data Integration Hub ワークフローを操作コンソールで作成し、PowerCenter ワークフローを Data Integration Hub ワークフローに割り当てます。ワークフローパラメータは、Forms Designer でカスタマイズできます。

次に、オペレータが、監視ルールを作成するときに、Data Integration Hub ワークフローを選択します。

開発者は、操作コンソールにおけるパブリケーションとサブスクリプションのソース接続とターゲット接続も管理します。

第 2 章

PowerCenter マッピングおよびワークフロー

この章では、以下の項目について説明します。

- [PowerCenter マッピングおよびワークフローの概要, 22 ページ](#)
- [PowerCenter ワークフロータイプ, 23 ページ](#)
- [自動 PowerCenter マッピングおよびワークフロー, 24 ページ](#)
- [カスタム PowerCenter マッピングおよびワークフロー, 25 ページ](#)
- [PowerCenter バッチワークフローの開発, 29 ページ](#)
- [PowerCenter リアルタイムワークフローの開発, 33 ページ](#)

PowerCenter マッピングおよびワークフローの概要

PowerCenter は、オンプレミスアプリケーション間での Data Integration Hub パブリケーションおよびサブスクリプションの実行、パブリケーションの前処理およびサブスクリプションの後処理の実行、ワークフローを起動する監視ルール用のアクションの実行に Data Integration Hub が使用する処理エンジンです。

自動マッピングパブリケーションおよびサブスクリプションの場合、トピックで定義したデータ構造に基づいてパブリケーションおよびサブスクリプションを処理する PowerCenter マッピングが Data Integration Hub によって作成されます。カスタムマッピングパブリケーションおよびサブスクリプションの場合、PowerCenter Client ツールを使用して、パブリケーションおよびサブスクリプションを処理する PowerCenter マッピングおよびワークフローを開発します。また、パブリケーションの前処理、サブスクリプションの後処理、PowerCenter ワークフローを起動する監視ルールで使用する PowerCenter マッピングおよびワークフローも開発します。

PowerCenter でワークフローを開発した後、Data Integration Hub 操作コンソールを使用して PowerCenter ワークフローを Data Integration Hub ワークフローへインポートします。詳細については、[「Data Integration Hub ワークフローの作成」 \(ページ 66\)](#)を参照してください。

次に、Data Integration Hub のオペレータは、Data Integration Hub 操作コンソールでパブリケーション、サブスクリプション、または監視ルールを作成し、PowerCenter ワークフローに基づいた Data Integration Hub ワークフローを選択します。詳細については、『*Data Integration Hub オペレータガイド*』を参照してください。

次のディレクトリにサンプルワークフローがあります。<DIHInstallationDir>/samples 各サンプルワークフローには、そのサンプルワークフローについての説明と使用方法が述べられた関連する readme ファイルがあります。

PowerCenter ワークフロータイプ

Data Integration Hub は、自動マッピングによるパブリケーションおよびサブスクリプションに PowerCenter バッチワークフローを使用します。

PowerCenter ワークフローは、カスタムマッピングを使用したパブリケーションとサブスクリプション、パブリケーションの前処理、サブスクリプションの後処理、および PowerCenter ワークフローを起動する監視ルールで使用できます。使用できる PowerCenter ワークフローのタイプは、ワークフローを使用するプロセスのタイプによって異なります。

PowerCenter バッチワークフロー

すべてのプロセスでバッチワークフローを使用できます。

PowerCenter リアルタイムワークフロー

リアルタイムワークフローは、カスタムパブリケーションに使用できます。

バッチワークフロー

バッチワークフローはトリガに応じて一度だけ実行されます。Data Integration Hub は、バッチワークフローを使用して、パブリケーションとサブスクリプションの処理、パブリケーションの前処理とサブスクリプションの後処理の実行、ワークフローを起動する監視ルール用のアクションの実行を行います。バッチワークフローでは、PowerCenter Web サービス Hub を使用して、PowerCenter と通信します。

次の操作によりバッチワークフローをトリガできます。

- スケジュールされたパブリケーションまたはサブスクリプションの開始。
- オペレータによる手動のワークフロー実行。
- DX_Start_Publication トランスフォーメーションの Data Integration Hub サーバーへのリクエストの送信。
- 完了したパブリケーションのサブスクリプションワークフローを開始するリクエスト、またはサブスクリプションの後処理ワークフローを開始するリクエストの、DX_Notification トランスフォーメーションから Data Integration Hub サーバーへの送信。
- ワークフローを起動するように設定された監視ルールの条件への、パブリケーションプロセスまたはサブスクリプションプロセスの一致。

パブリケーションまたはサブスクリプション操作によりバッチワークフローがトリガされると、Data Integration Hub は、処理中のステータスでイベントを作成します。スケジュールされたサブスクリプションに対してトリガが発生した場合、サブスクリプションのイベントは、データをコンシュームする準備ができたときからスケジュールされたサブスクリプションの開始時間まで遅延のステータスになります。ワークフローが監視ルールによって起動された場合は、Data Integration Hub はイベントを作成しません。

リアルタイムワークフロー

Data Integration Hub は、リアルタイムワークフローを使用してパブリケーションを処理します。リアルタイムワークフローでは、リアルタイムソースからのデータがオンデマンドで処理されます。リアルタイムソースには、Web サービスプロバイダ、Apache Kafka、Java Message Service (JMS) キューを使用できます。

リアルタイムワークフローは Data Integration Hub によっては起動されません。ワークフローの実行および管理は、Data Integration Hub のデベロッパーが行います。リアルタイムワークフローは継続的またはオンデマンドで実行できます。

リアルタイムワークフローを使用してパブリッシュする場合は、あらかじめ定義された間隔で、パブリッシュされたデータを単一のパブリケーションにグループ化します。スケジュールされたパブリケーション時に、Data Integration Hub は *処理中* ステータスのイベントを作成します。

自動 PowerCenter マッピングおよびワークフロー

自動マッピングを使用したパブリケーションまたはサブスクリプションを作成するときには、パブリケーションウィザードまたはサブスクリプションウィザードで設定された定義に従ってそのパブリケーションまたはサブスクリプションを処理する PowerCenter コンポーネントが Data Integration Hub によって作成されます。

自動マッピングによるパブリケーションまたはサブスクリプションには、Data Integration Hub によって以下の PowerCenter エンティティが作成されます。

- バッチワークフロー。データを処理するために Data Integration Hub がトリガする、スケジュールされたバッチワークフロー。
- データ処理セッション。パブリケーションソースからのデータ抽出またはサブスクリプションターゲットへのデータ書き込みを行うトランスフォーメーションが含まれるセッション。このセッションには、マッピング、ソースオブジェクト、およびターゲットオブジェクトも含まれます。
- レポートセッション。パブリケーションプロセスまたはサブスクリプションプロセスについての通知を送受信するセッション。このセッションは、処理後、パブリケーションイベントまたはサブスクリプションイベントを更新します。このセッションには、マッピング、ソースオブジェクト、およびターゲットオブジェクトも含まれます。
- PowerCenter リポジトリ内のメタデータフォルダ。これらのフォルダには、ワークフロー、マッピング、およびセッション情報が格納されます。分離したフォルダの 1 つに、パブリケーションまたはサブスクリプションのメタデータが格納されます。自動パブリケーションマッピングを作成するときには、パブリケーションターゲットに基づいて PowerCenter によってサブスクリプションソースショートカットが作成されます。

自動 PowerCenter マッピングおよびワークフローのルールとガイドライン

自動マッピングによってデータセットをパブリッシュまたはコンシュームする場合は、以下のルールとガイドラインを考慮します。

- Data Integration Hub は、自動マッピングごとに PowerCenter エンティティを作成します。各 PowerCenter エンティティ名は、プレフィックス **DIH__** で始まります。PowerCenter エンティティは手動で変更または削除してはいけません。
- PowerCenter の高精度処理を使用して自動マッピングを実行することはできません。高精度処理を使用するにはカスタムマッピングを使用します。
- 自動マッピングによってパブリケーションからのデータセットを格納するパブリケーションリポジトリ内のテーブルは手動で変更または削除してはいけません。
- 自動マッピングは、DIH__STAGING と DIH__REPO PowerCenter 接続で実行されます。パブリケーションまたはサブスクリプションのエラーを防止するため、接続を変更または削除してはいけません。
- Microsoft SQL Server データベースソースから varbinary または nvarchar データ型をパブリッシュする場合、PowerCenter はデータサイズ制限を 100MB の固定サイズに変更します。パブリッシュされたデータを Microsoft SQL Server データベースターゲットに書き込む自動マッピングを使用してサブスクリプションを作成する場合は、ターゲットテーブルを作成する時点でサブスクリプションワークフローを実行する前にデータサイズ制限を「最大」に変更する必要があります。
- リレーショナルデータベースからのパブリッシュおよびリレーショナルデータベースへのサブスクリプションにデフォルトのスキーマのテーブルを使用する場合、スキーマ名は維持されません。
- リレーショナルデータベースからのパブリッシュおよびリレーショナルデータベースへのサブスクリプションにデフォルト以外のスキーマのテーブルを使用する場合、スキーマ名は維持されます。

- スキーマ名は、PowerCenter セッションの [マッピング] タブの以下の場所に保存されます。
 - パブリケーションの場合、スキーマ名は **【所有者名】** フィールドの [ソースのプロパティ] に保存されます。
 - サブスクリプションの場合、スキーマ名は **【ターゲットテーブル名プレフィックス】** フィールドの [ターゲットのプロパティ] に保存されます。

自動 PowerCenter マッピングおよびワークフローのログ

自動マッピングによってデータセットをパブリッシュまたはコンシュームすると、PowerCenter 統合サービスは次のログを作成します。

パブリケーションワークフローのログ

PowerCenter 統合サービスでは、次の設定でパブリケーションワークフローが作成されます。

パラメータ	値
ワークフローログの保存元	実行数
これらの実行のワークフローログファイルの保存	\$PMWorkflowLogCount

PowerCenter 統合サービスの *\$PMWorkflowLogCount* の値によって、統合サービスがワークフローの実行ごとに作成するログの数が決まります。

パブリケーションセッションのログ

PowerCenter 統合サービスでは、次の設定でパブリケーションワークフローセッションが作成されます。

パラメータ	値
セッションログの保存元	実行数
これらの実行のセッションログの保存	\$PMSessionLogCount

PowerCenter 統合サービスの *\$PMSessionLogCount* の値によって、統合サービスがセッションの実行ごとに作成するログの数が決まります。

サブスクリプションのログ

PowerCenter 統合サービスは、同時実行セットでサブスクリプションワークフローを作成します。ログにはタイムスタンプが付加され、生成数には制限がありません。

カスタム PowerCenter マッピングおよびワークフロー

次の場合に、PowerCenter ワークフローを開発して Data Integration Hub にインポートできます。

- オンプレミスアプリケーションからパブリッシュする前に、データを準備する必要がある場合。
- データセットをオンプレミスアプリケーションからパブリッシュするため、またはオンプレミスアプリケーションにコンシュームするために、追加の処理やトランスフォーメーションが必要な場合。

- オンプレミスのアプリケーションまたはクラウドアプリケーションのいずれかで、サブスクライブするデータにロジックを追加する必要がある場合。
- パブリケーションプロセスまたはサブスクリプションプロセスが条件に一致した場合にワークフローを起動する、監視ルールを作成する場合。

このような場合のワークフローの開発手順は、他の PowerCenter ワークフローの開発手順と類似しています。次にオペレータは、カスタムマッピングを使用したパブリケーションまたはサブスクリプション、パブリケーションの前処理、サブスクリプションの後処理、または監視ルールで使用するワークフローを、必要に応じて選択します。

サポートされるデータ型

PowerCenter ワークフローを使用するパブリケーションリポジトリにデータセットをパブリッシュするときには、パブリケーションリポジトリのデータベースタイプに基づいてデータ型をパブリッシュできます。

次の表に、パブリケーションリポジトリにパブリッシュできる、サポートされているデータ型を示します。

データベースタイプ	データ型
Oracle	<ul style="list-style-type: none"> - blob - clob - 数字 - 数字(\$p, \$s) - 29 ビット以下のタイムスタンプ - varchar2(\$l char)
Microsoft SQL Server	<ul style="list-style-type: none"> - 長整数 - 日時 - 10 進数(\$p,\$s) - 浮動小数点数型 - 整数 - nvarchar(\$l) - nvarchar(MAX) - varbinary(MAX)

カスタム PowerCenter マッピングおよびワークフローのルールとガイドライン

カスタムマッピングを使用したパブリケーションとサブスクリプション、パブリケーションの前処理、サブスクリプションの後処理、および監視ルールで使用する PowerCenter ワークフローを開発する場合は、次のルールとガイドラインを考慮します。

一般的なルールとガイドライン

- PowerCenter で Data Integration Hub のワークフローを開発する前に、Data Integration Hub PowerCenter クライアントおよびサーバープラグインが PowerCenter リポジトリにインストールされ登録されていることを確認します。詳細については、『*Data Integration Hub インストール&環境設定ガイド*』を参照します。
- カスタムマッピング用の PowerCenter エンティティの命名規則とは異なる命名規則で、別のフォルダにカスタムマッピング用の PowerCenter エンティティを名前を付けて保存します。
- Data Integration Hub は、自動マッピングを使用するパブリケーションイベントおよびサブスクリプションイベントのステータスの更新に、個別のレポートセッションを使用します。自動マッピングと同様に、データ処理とレポートには個別のセッションを作成することをお勧めします。自動マッピングのワークフローは参照として使用できます。

- ユーザー定義のセッションパラメータをカスタムワークフローで使用し、それらの値を Data Integration Hub またはパラメータファイルで定義できます。ユーザー定義のセッションパラメータの値は、Forms Designer の Data Integration Hub で管理できます。組み込みセッションパラメータの値は、Data Integration Hub では管理できません。セッションパラメータの詳細については、『PowerCenter 上級ワークフローガイド』の「セッションパラメータに関する作業」セクションを参照します。

注: Data Integration Hub では、\$InputFile_\$\$CustomVariable という形式のセッションパラメータはサポートされません。

- PowerCenter マッピングを使用するカスタムのパブリケーションおよびサブスクリプションでは、Data Integration Hub で自動マッピングに割り当てられるものと同じイベントのステータスおよびタイプを使用することをお勧めします。詳細については、『Data Integration Hub オペレータガイド』を参照してください。イベントのステータスおよびタイプは、DX_Event_Details トランスフォーメーションで割り当てます。
- 名前の競合を防ぐため、パラメータ名に **_DIH_** を使用せず、Data Integration Hub が自動マッピングでのパブリケーション用とサブスクリプション用のワークフローに使用するワークフローおよびマッピングパラメータと同じ名前のワークフローおよびマッピングパラメータを使用しません。
- データベースタイプがパブリケーションリポジトリデータベースタイプとは異なるデータベースターゲットにデータベースソースから公開または書き込みを行うと、Data Integration Hub はそのデータをパブリケーションリポジトリデータベースがサポートするデータ型に変換します。このため、公開されたデータをパブリケーションリポジトリから異なるターゲットデータベースに使用する場合は、データ型変換によって処理中にランタイムエラーが発生しないことを確認します。例えば、Microsoft SQL Server データベースソースから Oracle パブリケーションリポジトリにデータを公開し、続いてその公開されたデータを Microsoft SQL Server データベースターゲットに使用する場合は、MIN 値または MAX 値が、Microsoft SQL Server データベースターゲットがサポートする値よりも大きい値または小さい値に変換される可能性があります。
- Data Integration Hub のワークフローをパブリケーションワークフローまたはサブスクリプションワークフローとして使用するには、PowerCenter リポジトリ内のワークフローを選択するか、またはエクスポートされたワークフロー定義ファイルを選択することにより、Data Integration Hub 操作コンソールで Data Integration Hub ワークフローを作成します。詳細については、『[Data Integration Hub ワークフローの作成](#)』（ページ 66）を参照してください。
- バッチワークフローに適用されるその他のルールとガイドラインについては、『[PowerCenter バッチワークフローの開発のルールとガイドライン](#)』（ページ 29）を参照してください。
- リアルタイムワークフローに適用されるその他のルールとガイドラインについては、『[PowerCenter リアルタイムワークフローの開発のルールとガイドライン](#)』（ページ 33）を参照してください。

パブリケーションのルールとガイドライン

- データベースソースから公開する場合、パブリケーションターゲットのテーブル名とカラム名にスペース ()、ダッシュ (-)、およびピリオド (.) は使用できません。これらの文字は、パブリケーションプロセスでアンダースコア (_) に置き換えられます。
- パブリケーション前処理ワークフローを開発する場合は、前処理ワークフローの最後に、たとえば別のマッピング内で、DX_Start_Publication トランスフォーメーションを呼び出します。このトランスフォーメーションは、Data Integration Hub サーバーにパブリケーションプロセスをトリガするように指示します。DX_Start_Publication トランスフォーメーションを設定する場合は、次の点を考慮してください。

- パブリケーションの前処理で単一のパブリケーションを開始する場合は、DXEventId ポートを使用します。イベント ID を使用することで、Data Integration Hub ではパブリケーションの前処理ワークフローとパブリケーションワークフローで同じイベントが使用され、イベントステータスが適宜変更されます。

注: DXEventId ポートを定義しない場合は、DXPublicationName ポートを定義する必要があります。

- パブリケーションの前処理で複数のパブリケーションを開始する場合は、DX_Start_Publication トランスフォーメーション内でイベント ID を使用しないでください。PowerCenter トランスフォーメーションのイベントの詳細を使用してイベントステータスを変更できます。

注: DX_Start_Publication トランスフォーメーションは、ワークフロー内で 2 回以上呼び出さないでください。2 回以上呼び出した場合、Data Integration Hub はパブリケーションを複数回開始します。

- ファイルソースを使用したパブリケーションのワークフローを開発する際にソースファイルのパスをパラメータ化している場合、Data Integration Hub がそのファイルを取得して Data Integration Hub ドキュメントストアに移動します。ソースファイルのパスがハードコードされている場合、PowerCenter ソースがファイルを取得して処理します。ファイルパスがパラメータ化されているソースファイルの場合は、次のルールが適用されます。
 - フラットファイルソースの場合、ソースファイルのタイプは間接でなければなりません。
 - パススルーファイルソースの場合、ソースファイルのタイプは直接でなければなりません。
- カスタムマッピングによるパブリケーションで使用する PowerCenter に基づく Data Integration Hub ワークフローを選択すると、そのワークフローのターゲット定義に基づいて、パブリッシュされたデータセットの構造が Data Integration Hub によってパブリケーションリポジトリに作成されます。

サブスクリプションのルールとガイドライン

- サブスクリプションの後処理ワークフローを開発する場合は、そのワークフローの最後で DX_Notification トランスフォーメーションを呼び出します。後処理ワークフローのサンプルは、ディレクトリ <DIHInstallationDir>/samples/post_processing_workflow で確認できます。
- 複合サブスクリプション用のワークフローを開発する場合は、複合サブスクリプションをすべての公開されたデータセットの使用準備が整う前に手動で開始するかどうかを定義します。例えば、ワークフローを失敗させたり空のテーブルを無視したりするようにマッピングを設定できます。使用の準備ができていない公開されたデータセットは、パブリケーションインスタンス ID が 0 です。
- ファイルターゲットを使用したサブスクリプションのワークフローを開発する場合は、ターゲットファイルパスをパラメータ化できます。ファイルパスをパラメータ化する場合は、次のルールおよびガイドラインが適用されます。
 - フラットファイルターゲットの場合、ターゲットファイルのパラメータは\$OutputFile で開始する必要があります。
 - パススルーファイルターゲットの場合、ターゲットファイルのパラメータは\$OutputFile_DIHRepoFile_ で開始する必要があります。
 - Data Integration Hub オペレータが Data Integration Hub 操作コンソールでサブスクリプションを作成する場合は、出力ファイルパラメータの値としてターゲット出力ファイルの名前を指定する必要があります。
 - 出力ファイルパラメータの値には、(\$sequence)など、ファイル名を一意にするためのパターンを含めることができます。
- 差分検出が適用されているトピックテーブルからのデータを使用するサブスクリプションのワークフローを開発する場合は、マッピングに更新ストラテジトランスフォーメーションを追加し、ターゲットアプリケーションに存在するデータの更新ストラテジを定義します。差分検出が適用されているトピックテーブルの各行に対して、次のフラグのいずれかを追加します。

フラグ	操作
I	行の挿入
U	行の更新

フラグ	操作
D	行の削除
N	操作なし

PowerCenter バッチワークフローの開発

PowerCenter バッチワークフローを Data Integration Hub カスタムマッピング向けに開発する場合、他の PowerCenter ワークフローと同じ方法で開発します。Data Integration Hub トランスフォーメーションは、Data Integration Hub と直接相互作用します。

PowerCenter バッチワークフローの開発のルールとガイドライン

カスタムマッピングを使用したパブリケーションおよびサブスクリプションや、監視ルールで使用する PowerCenter バッチワークフローを開発する場合は、次のルールとガイドラインを考慮します。

- PowerCenter トランスフォーメーションを PowerCenter マッピングに追加すると、対応する Data Integration Hub ワークフローが使用できるワークフローパラメータを追加できます。Data Integration Hub 操作コンソールで Forms Designer を使用して、カスタムマッピングによるパブリケーションまたはサブスクリプションをオペレータが作成または編集するときに表示されるワークフローパラメータのレイアウトと動作をカスタマイズできます。
- PowerCenter のワークフローまたはマッピングを編集する場合、Data Integration Hub 操作コンソールで PowerCenter ワークフローまたは XML ファイルを再選択することにより、関連付けられている Data Integration Hub ワークフローを更新する必要があります。ワークフローのパラメータとパラメータタイプの変更は、更新処理中に解決できます。ワークフローのパラメータタイプを変更すると、インポートされたパラメータの値はインポートプロセスによって削除されます。それらのパラメータ値は手動で入力する必要があります。

Data Integration Hub で使用する PowerCenter ワークフローの開発についての一般的なルールとガイドラインについては、[カスタム PowerCenter マッピングおよびワークフローのルールとガイドライン](#)を参照してください。

パブリケーションおよびサブスクリプションプロセス向けの PowerCenter バッチワークフローの開発

Data Integration Hub データを処理するバッチワークフローを PowerCenter で開発するには、次の手順を実行します。

1. ソースおよびターゲット定義を作成します。
2. マッピングを作成し、トランスフォーメーションを追加します。
3. ワークフローとセッションを作成します。
4. PowerCenter ワークフローを PowerCenter リポジトリに保存します。

注: このプロセスは、監視ルールワークフローには該当しません。監視ルール向けのワークフローの定義方法には、制限はありません。

手順 1. ソースおよびターゲット定義の作成

パブリケーションワークフローを開発する場合は、パブリッシュするデータベースまたはファイルの構造に基づいてソースを作成し、ターゲットをパブリケーションリポジトリに設定します。

データベースソースからパブリッシュする場合は、パブリケーションリポジトリが格納できるデータ型をパブリッシュすることを確認します。

サブスクリプションワークフローを開発する場合は、パブリケーションリポジトリからトピック構造に基づいてソースを作成します。PowerCenter リポジトリのパブリケーションメタデータフォルダからソースをコピーできます。

他のマッピング用のソースおよびターゲットの定義を作成する方法と同じ方法で PowerCenter Designer にソースおよびターゲットの定義を作成できます。ソースおよびターゲットの定義の全般的な情報については、『*PowerCenter Designer ガイド*』を参照します。

サブスクリプションワークフローのソース定義の作成

PowerCenter Designer で、ソースを作成し、サブスクリプションワークフローのソースプロパティを定義します。パブリケーションワークフローを開発する場合は、データをパブリッシュする元アプリケーションに基づいてソースを定義します。

ソース定義プロセスには以下の手順が含まれます。

1. ソースオブジェクトを作成します。ソース接続を DIH__STAGING に設定します。ソースは、パブリケーションリポジトリからトピック構造に基づいて作成されます。PowerCenter リポジトリのパブリケーションメタデータフォルダからソースをコピーできます。
2. 使用するパブリッシュされたデータをフィルタリングするには、変数を追加します。例えば、パブリケーションリポジトリ内の特定のテーブルからデータを使用することを選択できます。
3. 使用するデータセットを決定する必須フィールドを追加します。自動マッピングによる集計サブスクリプションまたは複合サブスクリプションと同様に、使用するデータセットを複数定義できます。

次の表に、サブスクリプションワークフローのソースオブジェクトに追加するフィールドを示します。

フィールド	説明
DIH__PUBLICATION_INSTANCE_ID	<p>必須。1 つ以上のパブリッシュされたデータセットの識別子（カンマ区切りリスト形式）。アプリケーションがパブリッシュする各データセットには一意の識別子があります。コンSUMするデータをフィルタリングするには、\$ \$<topicName>__PublicationInstanceIDs ワークフローパラメータの値を使用します。</p> <p>パラメータのデータ型は、Oracle データベースターゲットに書き込む場合は number(19)、Microsoft SQL Server データベースターゲットに書き込む場合は number(19,0)でなければなりません。</p>
DIH__PUBLICATION_INSTANCE_DATE	<p>各アプリケーションがデータセットのパブリッシュを開始した日時（カンマ区切りのリスト形式）。データベースパーティションを使用する場合は、\$ \$<topic_name>__PublicationInstanceDatesSQL ワークフローパラメータの値を使用して、CONSUMするデータをフィルタリングできます。値の形式は、パブリケーションリポジトリデータベースのタイプによって異なります。</p> <p>Oracle データベースでは、データ型は date にし、値は次の形式でなければなりません。</p> <p>YYYY-MM-DD HH24:MI:SS</p> <p>Microsoft SQL Server データベースでは、データ型は datetime にし、値は次の形式でなければなりません。</p> <p>yyyy-mm-dd hh:mi:ss (24h)</p> <p>注: 異なるトランスフォーメーションで使用するデータをフィルタリングする場合は、代わりに\$ \$<topic_name>__PublicationInstanceDates パラメータを使用できます。</p>

使用するデータは、サブスクリプションワークフロー内のソース修飾子トランスフォーメーションのソースフィルタ属性でフィルタリングできます。次の例は、1 つの行内の ID と日付範囲によってフィルタリングするフィールド構文を示しています。

```
MY_TABLE.DIH__PUBLICATION_INSTANCE_ID in ($$myTopic__PublicationInstanceIDs)
AND MY_TABLE.DIH__PUBLICATION_INSTANCE_DATE in ($$myTopic__PublicationInstanceDatesSQL)
```

ターゲット定義の作成

PowerCenter Designer で、ターゲットを作成し、ワークフローを実行するためのプロパティを格納するポートを追加します。PowerCenter から返送する他の Data Integration Hub 値を保存するためにプロパティを追加することもできます。

ターゲット定義プロセスには以下の手順が含まれます。

1. ターゲットオブジェクトを作成します。パブリケーションワークフローを開発する場合は、ターゲット定義で DIH__STAGING 接続を使用します。データベースターゲットタイプは、パブリケーションリポジトリデータベースタイプと一致する必要があります。
2. パブリケーションワークフローを開発する場合は、パブリッシュするデータセットを決定する必須フィールドを追加します。

次の表に、パブリケーションのターゲットオブジェクトに追加する必須フィールドを示します。

パラメータ	説明
DIH__PUBLICATION_INSTANCE_ID	必須。パブリッシュされたデータセットの識別子。アプリケーションがパブリッシュする各データセットには一意の識別子があります。このフィールドは、 <code>\$\$publicationInstanceId</code> ワークフローパラメータの値を使用します。
DIH__PUBLICATION_INSTANCE_DATE	必須。アプリケーションがデータセットのパブリッシュを開始した日時。このフィールドは、 <code>\$\$publicationInstanceDate</code> ワークフローパラメータの値を使用します。値の形式は、パブリケーションリポジトリデータベースのタイプによって異なります。 Oracle データベースでは、この値は次の形式にする必要があります。 <code>to_date('<DATE>', 'YYYY-MM-DD HH24:MI:SS')</code> Microsoft SQL Server データベースでは、この値は次の形式にする必要があります。 <code>convert(datetime, '<DATE>', 120)</code>

手順 2. マッピングの作成

Data Integration Hub のデータの処理に使用するソース定義、ターゲット定義、およびトランスフォーメーションを含んだマッピングを作成します。

Data Integration Hub のマッピングは、他の PowerCenter マッピングを構築する方法と同じ方法で作成します。マッピングに製品固有の機能を追加するには、Data Integration Hub トランスフォーメーションと構造化されていないデータトランスフォーメーションを使用します。さらに、他の PowerCenter トランスフォーメーションもマッピングに追加します。

手順 3. PowerCenter ワークフローとセッションの作成

マッピングを実行するワークフローとセッションを作成します。ワークフローは、他の PowerCenter ワークフローを作成する方法と同じ方法で作成します。

- Workflow Designer でワークフローおよびセッションオブジェクトを作成します。ワークフローおよびセッションの作成の全般的な情報については、『*PowerCenter 上級ワークフローガイド*』を参照してください。
- パブリケーションプロセスまたはサブスクリプションプロセスが完了したことを Data Integration Hub サーバーに知らせるトランスフォーメーションを追加します。
 - パブリッシュワークフローを開発する場合は、サブスクリプションワークフローをトリガする `DX_Notification` トランスフォーメーションを追加し、パブリケーションイベントステータスを変更します。
 - サブスクリプションワークフローを開発する場合は、サブスクリプションイベントステータスを変更する `DX_Event_Details` を追加します。
- セッションオブジェクトを保存します。
- ワークフローをテストして、正しく機能することを確認します。

手順 4. PowerCenter ワークフローの保存

ワークフローを PowerCenter リポジトリに保存します。ワークフローを保存した後で、それを PowerCenter から XML ファイルにエクスポートすることができ、その XML ファイルはワークフロー定義ファイルとして使用されます。ワークフローをエクスポートするには、Repository Manager を使用する必要があります。

注: フォルダ名の先頭が DIH__pub または DIH__sub であるフォルダにはワークフローを保存しないでください。

PowerCenter リアルタイムワークフローの開発

Data Integration Hub 用の PowerCenter ワークフローは、他の PowerCenter ワークフローの開発と同じ方法で開発します。Data Integration Hub トランスフォーメーションは、Data Integration Hub と直接相互作用します。

PowerCenter リアルタイムワークフローの開発のルールとガイドライン

PowerCenter リアルタイムワークフローを開発してカスタムマッピングによるパブリケーションで使用する場合、次の規則およびガイドラインを考慮します。

- 必ず、自動マッピング用の PowerCenter エンティティの命名規則とは異なる命名規則で、別のフォルダにカスタムマッピング用の PowerCenter エンティティを名前を付けて保存します。
- プロパティ **【高精度を有効にする】** を、Data Integration Hub パブリケーションリポジトリにデータを書き込むセッションで有効にする必要があります。
- 名前の競合を防ぐため、パラメータ名に「**_DIH__**」は使用せず、Data Integration Hub が自動マッピングでのパブリケーション用のワークフローに使用するワークフローおよびマッピングパラメータと同じ名前のワークフローおよびマッピングパラメータを使用しません。
- ワークフローはエラーメッセージを生成しません。ワークフローの管理は Data Integration Hub のデベロッパーが行います。
- 1 つのトピック内に複数のテーブルがあり、これらのテーブルすべてのレコードが同じパブリケーションに含まれることを確認する場合は、システムプロパティ **dih.realtime.time.window** の値を変更します。詳細については、『*Data Integration Hub 管理者ガイド*』を参照してください。
- PowerCenter のワークフローまたはマッピングを編集する場合、Data Integration Hub 操作コンソールで PowerCenter ワークフローまたは XML ファイルを再選択することにより、関連付けられている Data Integration Hub ワークフローを更新する必要があります。

Data Integration Hub で使用する PowerCenter ワークフローの開発についての一般的なルールとガイドラインについては、[カスタム PowerCenter マッピングおよびワークフローのルールとガイドライン](#)を参照してください。

パブリケーションリアルタイムワークフローの開発プロセス

Data Integration Hub のリアルタイムパブリケーションを処理する PowerCenter でワークフローを開発するには、次のタスクを実行します。

1. ソースおよびターゲット定義を作成します。
2. マッピングを作成し、トランスフォーメーションを追加します。
3. ワークフローとセッションを作成し、ワークフローを保存します。

4. Data Integration Hub 操作コンソールで、アプリケーションがリアルタイムデータを公開するトピックを作成します。トピック構造は、PowerCenter ワークフローまたはワークロー定義ファイルに基づくようにします。Data Integration Hub が PowerCenter にトピックを作成し、Data Integration Hub の命名規則に従ってターゲットの名前を付けます。
5. 手順 3 で作成したワークフローで、ターゲットの名前を PowerCenter のトピックターゲットの名前に変更し、ワークフローを保存します。必要に応じて、ワークフローを XML ワークフロー定義ファイルにエクスポートします。
6. Data Integration Hub 操作コンソールで、パブリケーションリアルタイムワークフローを作成し、手順 5 で変更した PowerCenter ワークフローを選択します。

手順 1. ソースおよびターゲット定義の作成

パブリケーションリアルタイムワークフローを開発する際には、パブリッシュするリアルタイムデータに基づいてソースを作成します。パブリケーションリアルタイムワークフローのターゲットは、Data Integration Hub パブリケーションリポジトリに設定します。PowerCenter から送信する他の Data Integration Hub 値を保存するためにプロパティを追加することもできます。

パブリケーションリアルタイムワークフローのソースは、次のリアルタイムデータに基づいて定義します。

- Apache Kafka ストリームのデータ
- JMS の構造
- Web サービスのソース

他のマッピング用のソースおよびターゲットの定義を作成する方法と同じ方法で PowerCenter Designer にソースおよびターゲットの定義を作成出来ます。ソースおよびターゲットの定義の全般的な情報については、『*PowerCenter Designer ガイド*』を参照します。

ソースおよびターゲットの定義を作成する際、以下のガイドラインを考慮します。

- ターゲット定義の DIH__STAGING 接続を使用します。
- データベースのターゲットタイプは、Data Integration Hub パブリケーションリポジトリのデータベースタイプと一致している必要があります。
- ターゲットには、DIH__PUBLICATION_INSTANCE_ID カラムおよび DIH__PUBLICATION_INSTANCE_DATE カラムが含まれている必要があります。

次の図は、DIH__PUBLICATION_INSTANCE_ID カラムおよび DIH__PUBLICATION_INSTANCE_DATE カラムのあるターゲットの例を示しています。

Target Designer

K	Name	Datatype	L...
	Notification_ID	number(p,s)	15
	Notification_Type	varchar2	60
	OrderID	number	15
	ORDER_DATE	timestamp	29
	CUSTOMER_ID	number	15
	PARTY_ID	number	15
	NAME	varchar2	60
	ADDRESS	varchar2	60
	ADDRESS2	varchar2	60
	CITY	varchar2	30
	STATE	varchar2	20
	ZIPCODE	varchar2	20
	COUNTRY	varchar2	20
	PHONE	varchar2	40
	EMAIL	varchar2	80
	CC_TYPE	varchar2	20
	SHIP_DATE	timestamp	29
	FREIGHT	number(p,s)	5
	DIH__PUBLICATION_INSTANCE_ID	number(p,s)	19
	DIH__PUBLICATION_INSTANCE_DATE	date	19

K	Name	Datatype	L...
	Notification_ID	number(p,s)	15
	Notification_Type	varchar2	100
	DetailID	varchar2	20
	ORDER_ID	number	15
	PRODUCT_ID	number	15
	QUANTITY	number	15
	UNIT_PRICE	number(p,s)	5
	DISCOUNT	number	15
	PACKAGING	varchar2	100
	DIH__PUBLICATION_INSTANCE_ID	number(p,s)	19
	DIH__PUBLICATION_INSTANCE_DATE	date	19

手順 2。マッピングの作成

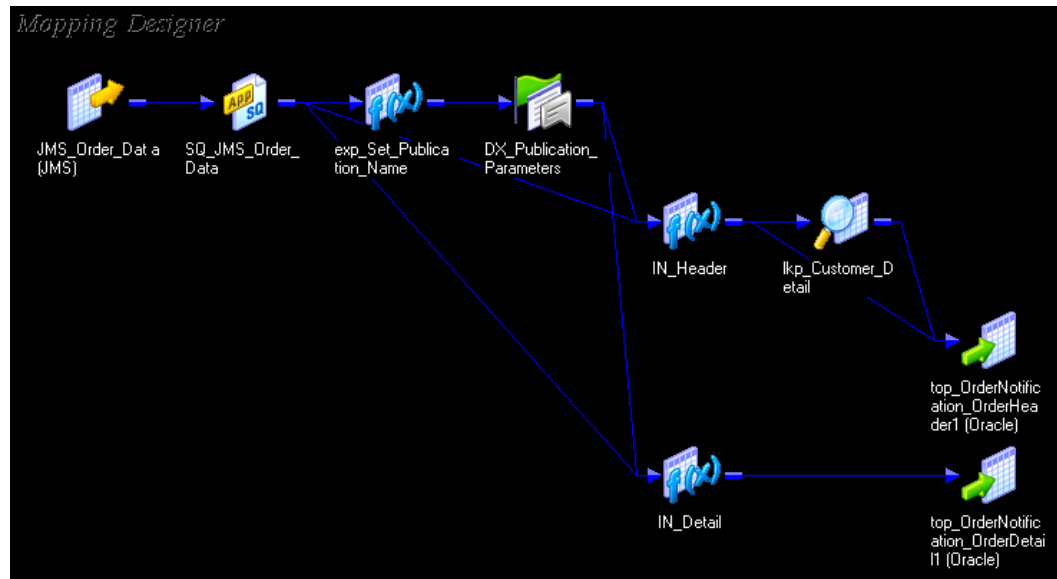
リアルタイムパブリケーションワークフローで使用するソース定義、ターゲット定義、およびトランスフォーメーションを含むマッピングを作成します。

Data Integration Hub のマッピングは、他の PowerCenter のマッピングを構築する方法と同じ方法で作成します。

ソースおよびマッピングの作成時には、次のガイドラインを考慮します。

- DX_Publication_Parameters トランスフォーメーションをマッピングに追加します。さらに、他の PowerCenter トランスフォーメーションもマッピングに追加します。
- DX_Publication_Parameters トランスフォーメーションの DXPublicationName ポートに、ワークフローをパブリッシュするパブリケーションの名前を入力します。DX_Publication_Parameters トランスフォーメーションは、パブリケーションインスタンス ID 用およびパブリケーション日用のプレースホルダを Data Integration Hub パブリケーションリポジトリに追加します。
- ポート DXPublicationInstanceId を DX_Publication_Parameters トランスフォーメーションからターゲットの DIH__PUBLICATION_INSTANCE_ID カラムに接続します。
- ポート DXPublicationInstanceDate を DX_Publication_Parameters トランスフォーメーションからターゲットの DIH__PUBLICATION_INSTANCE_DATE カラムに接続します。

次の図は DX_Publication_Parameters トランスフォーメーションを含むマッピング例を示しています。マッピングにはパブリケーション名を設定する式も含まれます。



手順 3. PowerCenter のワークフローおよびセッションの作成と保存

マッピングを実行するリアルタイムワークフローおよびセッションを作成します。ワークフローは、他の PowerCenter ワークフローを作成する方法と同じ方法で作成します。

1. Workflow Designer でワークフローおよびセッションオブジェクトを作成します。セッションはリアルタイムセッションとして設定する必要があります。ワークフローおよびセッションの作成の全般的な情報については、『PowerCenter 上級ワークフローガイド』を参照してください。
2. セッションオブジェクトを保存します。
3. ワークフローをテストして、正しく機能することを確認します。
4. ワークフローを PowerCenter リポジトリに保存します。必要に応じて、Repository Manager を使用してワークフローを PowerCenter から XML ファイルにエクスポートします。

注: フォルダ名の先頭が DIH__pub または DIH__sub であるフォルダにはワークフローを保存しないでください。

手順 4. トピックの作成

アプリケーションがリアルタイムデータを公開するトピックを作成します。トピックは、データ構造およびデータ保持期間などの追加のデータ定義を定義します。

トピックは、Data Integration Hub 操作コンソールで作成します。

トピックの作成時には、次のガイドラインを考慮します。

- トピック構造は、[トピック] ウィザードの [構造] ページで定義します。
- [テーブルの追加] リストから、[PowerCenter から] を選択します。

- **[PowerCenter ワークフローからのテーブルの追加]** ダイアログボックスで、次のいずれかのオプションを選択します。
 - リアルタイムワークフローからテーブルを追加する場合は、**[PowerCenter リポジトリからのワークフローの選択]** を選択します。
 - リアルタイムワークフローからエクスポートした XML ファイルからテーブルを追加する場合は、**[ワークフロー定義ファイル (.xml) の選択]** を選択します。

詳細については、『*Data Integration Hub Operator ガイド*』を参照してください。

手順 5. ワークフローターゲットの名前の変更

PowerCenter リアルタイムワークフローのターゲットの名前を PowerCenter のトピックターゲットの名前に変更します。ターゲットの名前は PowerCenter Designer で変更します。

1. ナビゲーションペインで、作成したトピックのフォルダにアクセスし、フォルダツリーを展開します。
2. Targets サブフォルダを展開し、クリップボードにターゲット名をコピーします。
3. ワークフローのフォルダにアクセスし、フォルダツリーを展開します。
4. Targets サブフォルダを展開し、ターゲット名をトピックのフォルダからコピーした名前に置き換えます。
5. セッションオブジェクトを保存します。
6. ワークフローをテストして、正しく機能することを確認します。
7. PowerCenter ワークフローを PowerCenter リポジトリに保存します。必要に応じて、Repository Manager を使用してワークフローを PowerCenter から XML ファイルにエクスポートします。

注: フォルダ名の先頭が DIH__pub または DIH__sub であるフォルダにはワークフローを保存しないでください。

手順 6. パブリケーションリアルタイムワークフローの作成

リアルタイムデータを公開する Data Integration Hub リアルタイムワークフローを作成します。Data Integration Hub のオペレータは、このワークフローを使用してリアルタイムパブリケーションを作成できます。

ワークフローは Data Integration Hub 操作コンソールの **[ワークフロー]** ページで作成します。

ワークフローの作成時には、次のガイドラインを考慮します。

- トピック構造は、[トピック] ウィザードの **[構造]** ページで定義します。
- **[フロータイプ]** で、**[PowerCenter リアルタイムワークフロー]** を選択します。
- **[タイプ]** で **[パブリケーション]** を選択します。

Data Integration Hub オペレータは、**[カスタムリアルタイムパブリケーションの作成]** の **[マッピング]** ページでワークフローを選択します。

第 3 章

Data Engineering Integration および Data Engineering Streaming のマッピングとワークフロー

この章では、以下の項目について説明します。

- [Data Engineering Integration および Data Engineering Streaming のマッピングとワークフローの概要, 38 ページ](#)
- [Data Integration Hub 内の Data Engineering Integration および Data Engineering Streaming のマッピングおよびワークフロー, 39 ページ](#)
- [作業を開始する前に, 40 ページ](#)
- [パブリケーション用の Data Engineering Integration マッピングの開発, 40 ページ](#)
- [パブリケーション用 Data Engineering Streaming マッピングの開発, 42 ページ](#)
- [サブスクリプション用の Data Engineering Integration マッピングの開発, 44 ページ](#)
- [パブリケーション用の Data Engineering Integration ワークフローの開発, 46 ページ](#)
- [サブスクリプション用の Data Engineering Integration ワークフローの開発, 48 ページ](#)

Data Engineering Integration および Data Engineering Streaming のマッピングとワークフローの概要

Data Integration Hub は、Data Engineering Integration および Data Engineering Streaming を使用して、Data Integration Hub のビッグデータパブリケーションおよびサブスクリプションを実行します。

大規模かつ多種多様で急速に変化するデータセットをパブリッシュして消費するカスタムのバッチパブリケーションおよびサブスクリプションを実行するには、Data Engineering Integration マッピングを使用します。ワークフローで Data Engineering Integration ワークフローと複数のマッピングを使用して、大規模かつ多種多様で急速に変化するデータセットをパブリッシュおよび消費する複数のカスタムバッチパブリケーションおよびサブスクリプションを実行します。データのストリームをリアルタイムでパブリッシュ

するカスタムのマルチレイテンシーパブリケーションを実行するには、Data Engineering Streaming マッピングを使用します。

Data Engineering Integration ワークフローを、複数の Data Engineering Integration マッピングを使用して作成します。

データ統合サービスは、Data Engineering Integration マッピング、Data Engineering Streaming マッピング、Data Engineering Integration ワークフローを Hadoop 環境で実行します。

Developer tool を使用して、パブリケーションおよびサブスクリプションを処理する Data Engineering Integration マッピング、Data Engineering Streaming マッピング、および Data Engineering Integration ワークフローを開発します。開発したマッピングは、Data Integration Hub 操作コンソールを使用して、Data Integration Hub ワークフローにインポートします。詳細については、[「Data Integration Hub ワークフローの作成」 \(ページ 66\)](#)を参照してください。

Data Integration Hub のオペレータは、Data Integration Hub 操作コンソールでパブリケーションまたはサブスクリプションを作成し、Data Engineering Integration マッピング、Data Engineering Streaming マッピング、または Data Engineering Integration ワークフローに基づいた Data Integration Hub ワークフローを選択します。詳細については、『*Data Integration Hub オペレータガイド*』を参照してください。

サンプルマッピング

次の場所にサンプルマッピングがあります。

- Data Engineering Integration マッピング: <DIHInstallationDir>/samples/bdm_mappings。各サンプルマッピングには、独自のマッピングを作成する基礎として、そのサンプルマッピングについての説明とマッピングを使用するためのガイドラインが記載された関連 readme ファイルがあります。
- Data Engineering Streaming マッピング: <DIHInstallationDir>/samples/bds_mappings。このフォルダの下には、Oracle パブリケーションリポジトリと Microsoft SQL Server パブリケーションリポジトリのサンプルマッピングを含むサブフォルダがあります。このフォルダにある readme ファイルには、独自のマッピングを作成する基礎として、そのサンプルマッピングについての説明とマッピングを使用するためのガイドラインが記載されています。
- Data Engineering Integration ワークフロー: <DIHInstallationDir>/samples/bdm_workflows。このフォルダの下には、Oracle パブリケーションリポジトリと Microsoft SQL Server パブリケーションリポジトリのサンプルワークフローを含むサブフォルダがあります。このフォルダにある readme ファイルには、独自のワークフローを作成する基礎として、そのサンプルワークフローについての説明とワークフローを使用するためのガイドラインが記載されています。

Data Integration Hub 内の Data Engineering Integration および Data Engineering Streaming のマッピングおよびワークフロー

Data Engineering Integration マッピングおよびワークフローは、カスタムバッチマッピングを使用する Data Integration Hub のパブリケーションおよびサブスクリプションで使用します。Data Engineering Streaming マッピングおよびワークフローは、カスタムマルチレイテンシーマッピングを使用する Data Integration Hub のパブリケーションで使用します。

パブリケーションマッピングでは、ソースはパブリッシュ元のアプリケーション、ターゲットは Data Integration Hub パブリケーションリポジトリです。サブスクリプションマッピングでは、ソースは Data Integration Hub パブリケーションリポジトリ、ターゲットはサブスクライブ元のアプリケーションです。

ユーザー定義のマッピングパラメータをカスタムバッチマッピングで使用し、それらの値を Data Integration Hub またはパラメータファイルで定義できます。マッピングパラメータの値は、Developer tool のマッピング

に基づいて管理できます。組み込みマッピングパラメータの値は、Data Integration Hub では管理できません。マッピングパラメータの詳細については、『*Informatica Developer Tool ガイド*』を参照してください。

注: Data Integration Hub では、\$InputFile_、\$OutputFile_の形式のセッションパラメータはサポートされません。

作業を開始する前に

Data Integration Hub で使用する Data Engineering Integration マッピング、Data Engineering Streaming マッピング、または Data Engineering Integration ワークフローを開発する前に、次の条件が満たされていることを確認します。

- 次の Data Integration Hub コンポーネントがインストールされている。
 - Data Integration Hub Hadoop サービス。
 - Data Integration Hub Data Engineering Integration。
- データのパブリッシュ先およびコンシューム元のトピックが Data Integration Hub で設定されている。トピックは Hadoop ベースのトピックである必要があります。
- 次の接続が Data Integration Hub で設定されている。
 - パブリッシュ元のアプリケーションへの接続。
 - サブスクライブ元のアプリケーションへの接続。
- マッピングを実行する場所への Hadoop 接続が、使用している環境に存在する。接続はクラスタ接続である必要があります。また、接続でマッピングロジックを Hadoop クラスタにプッシュする必要があります。

Data Engineering Streaming マッピングを開発する前に、<DIHInstallationDir>/powercenter/lib のランタイム.jar ファイルを次の場所にコピーします。

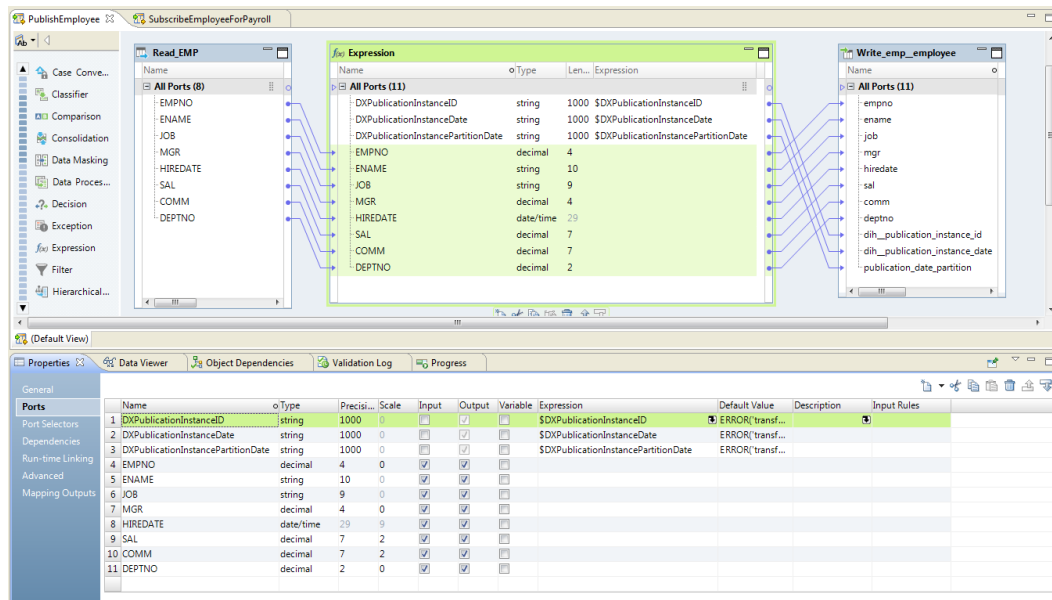
BDS_HOME/services/shared/hadoop/\$Distribution/extras/spark-auxjars/

パブリケーション用の Data Engineering Integration マッピングの開発

パブリケーション用の Data Engineering Integration マッピングを開発するには、Developer tool で次の手順を実行します。

1. ソース接続とターゲット接続を作成する。ソース接続はパブリッシュ元のアプリケーションへの接続、ターゲット接続は Data Integration Hub パブリケーションリポジトリへの Hive 接続です。
2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成します。
3. マッピングを作成し、ソースオブジェクトおよびターゲットオブジェクトをマッピングに追加します。
4. Data Integration Hub パラメータをマッピングに追加します。
5. 式トランスフォーメーションをマッピングに追加し、トランスフォーメーションのポートを設定して、ソース、トランスフォーメーション、およびターゲット間でポートを接続します。
6. マッピングのランタイム環境を設定し、マッピングからアプリケーションを作成します。

次の図に、パブリケーションマッピングの例を示します。



手順 1. ソース接続およびターゲット接続を作成する

1. パブリッシュ元のアプリケーションへの Streaming ソース接続を作成します。
2. Data Integration Hub パブリケーションリポジトリへのターゲット接続を作成します。

手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する

【物理データオブジェクト】でデータオブジェクトを作成します。

1. ソースデータオブジェクトを作成し、パブリッシュするソース接続内のコラムプロジェクションを定義します。
2. ターゲットデータオブジェクトを作成し、ソースのデータのパブリッシュ先となるターゲット接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトである必要があります。

手順 3. ソースとターゲットを使用してマッピングを作成する

1. 新しいマッピングを作成して、名前を付けます。
2. ソースの物理データオブジェクトを、リーダーとしてマッピングに追加します。
3. ターゲットの物理データオブジェクトを、ライターとしてマッピングに追加します。

手順 4. Data Integration Hub パラメータをマッピングに追加する

- ▶ 次のパラメータをマッピングに追加します。

DXPublicationInstanceID
DXPublicationInstanceDate
DXPublicationInstancePartitionDate

手順 5. マッピングへの式トランスフォーメーションの追加

1. 式トランスフォーメーションを、ソースオブジェクトとターゲットオブジェクト間のマッピングに追加します。
2. ソースオブジェクトのすべてのポートを式トランスフォーメーションの同一ポートにリンクします。例えば、トピックテーブルにカラム EMPNO が含まれている場合、ソースオブジェクトの EMPNO ポートを式トランスフォーメーションの EMPNO ポートにリンクします。
3. 式トランスフォーメーションで次の追加ポートを設定します。
DXPublicationInstanceID
DXPublicationInstanceDate
DXPublicationInstancePartitionDate
すべてのポートのデータ型は文字列で、最小精度は 200 です。
4. 式トランスフォーメーションのすべてのポートをターゲットオブジェクトの同一ポートにリンクします。例えば、式トランスフォーメーションの EMPNO ポートをターゲットオブジェクトの EMPNO ポートにリンクし、式トランスフォーメーションの DXPublicationInstanceID ポートをターゲットオブジェクトの dih_PublicationInstanceID ポートにリンクします。
5. マッピングを保存する。

手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する

1. **【プロパティ】** ペインで **【ランタイム】** を選択し、**【検証環境】** で **【Hadoop】** と **【Hive】** を順に選択します。**【ネイティブ】** が選択されていないことを確認します。
2. マッピングからアプリケーションを作成します。
マッピングはデータ統合サービスの Hadoop 環境にデプロイされます。

パブリケーション用 Data Engineering Streaming マッピングの開発

パブリケーション用の Data Engineering Streaming マッピングを開発するには、Developer tool で次の手順を実行します。

1. ソース接続とターゲット接続を作成する。ソース接続はパブリッシュ元のアプリケーションへの接続、ターゲット接続は Data Integration Hub パブリケーションリポジトリへの接続です。
2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成します。
3. マッピングを作成し、ソースオブジェクトおよびターゲットオブジェクトをマッピングに追加します。
4. 式トランスフォーメーションをマッピングに追加し、トランスフォーメーションのポートを設定して、ソースとトランスフォーメーションとの間でポートを接続します。
5. Java トランスフォーメーションをマッピングに追加し、フィールドを式トランスフォーメーションから Java トランスフォーメーションにマッピングした後、Java トランスフォーメーションからターゲットにマッピングします。
6. マッピングのランタイム環境を設定し、マッピングからアプリケーションを作成します。

手順 1. ソース接続およびターゲット接続を作成する

1. パブリッシュ元のアプリケーションへの Streaming ソース接続を作成します。
2. Data Integration Hub パブリケーションリポジトリへのターゲット接続を作成します。

手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する

【物理データオブジェクト】でデータオブジェクトを作成します。

1. ソースデータオブジェクトを作成し、パブリッシュするソース接続内のカラムプロジェクションを定義します。
2. ターゲットデータオブジェクトを作成し、ソースのデータのパブリッシュ先となるターゲット接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトである必要があります。

手順 3. ソースとターゲットを使用してマッピングを作成する

1. 新しいマッピングを作成して、名前を付けます。
2. ソースの物理データオブジェクトを、リーダーとしてマッピングに追加します。
3. ターゲットの物理データオブジェクトを、ライターとしてマッピングに追加します。

手順 4. マッピングへの式トランスフォーメーションの追加

1. 式トランスフォーメーションを、ソースオブジェクトとターゲットオブジェクト間のマッピングに追加します。
2. ソースオブジェクトのすべてのポートを式トランスフォーメーションの同一ポートにリンクします。
3. 式トランスフォーメーションで次の追加ポートを設定します。

ポート	説明
PublicationName	パブリケーションの名前。
DX_SERVER_URL	有効な Data Integration Hub RMI URL。以下に例を示します。rmi://localhost:18095

手順 5. Java トランスフォーメーションをマッピングに追加する

1. サンプルマッピングから Java トランスフォーメーションをコピーします。次の場所にサンプルマッピングがあります。<DIHInstallationDir>/samples/bds_mappings このフォルダの下には、Oracle パブリケーションリポジトリと Microsoft SQL Server パブリケーションリポジトリのサンプルマッピングを含むサブフォルダがあります。
2. 次のフィールドを式トランスフォーメーションから Java トランスフォーメーションにマッピングします。
 - PublicationName
 - DX_SERVER_URL

3. 次のフィールドを Java トランスフォーメーションからターゲットオブジェクトのトピックテーブルにマッピングします。

Java トランスフォーメーションフィールド	トピックテーブルフィールド
DXPublicationInstanceID	DIH__PUBLICATION_INSTANCE_ID
DXPublicationInstanceDate	DIH__PUBLICATION_INSTANCE_DATE

4. ターゲットトランスフォーメーションで、ターゲットデータオブジェクトを開き、データ型 `DIH__PUBLICATION_INSTANCE_ID` を `decimal` から `bigint` に変更します。
5. マッピングを保存します。

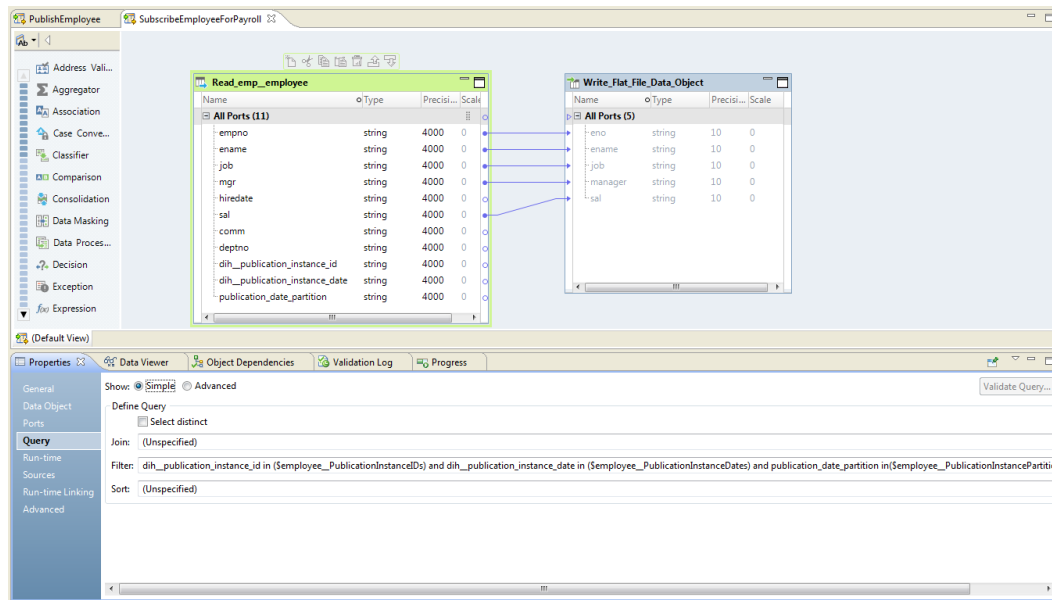
手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する

1. **【プロパティ】** ペインで **【ランタイム】** を選択し、**【検証環境】** で **【Hadoop】** と **【Spark】** を順に選択します。**【ネイティブ】** が選択されていないことを確認します。
2. マッピングからアプリケーションを作成します。
マッピングはデータ統合サービスの Hadoop 環境にデプロイされます。

サブスクリプション用の Data Engineering Integration マッピングの開発

サブスクリプション用の Data Engineering Integration マッピングを開発するには、Developer tool で次の手順を実行します。

1. ソース接続とターゲット接続を作成する。ソース接続は Data Integration Hub パブリケーションリポジトリへの Hive 接続、ターゲット接続はサブスクライバ元のアプリケーションへの接続です。
2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成します。
3. マッピングを作成し、ソースオブジェクトおよびターゲットオブジェクトをそのマッピングに追加して、ソースとターゲット間でポートを接続します。
4. Data Integration Hub パラメータをマッピングに追加します。
5. フィルタクエリをソースに追加します。パブリケーションのインスタンス ID、パブリケーションの日付、およびパブリケーションの日付パーティションに基づいて、サブスクリプションをフィルタリングできます。
注: バインドされていないサブスクリプションのマッピングを作成している場合は、マッピングにフィルタクエリを追加する必要はありません。
6. マッピングのランタイム環境を設定し、マッピングからアプリケーションを作成します。
次の図に、サブスクリプションマッピングの例を示します。



手順 1. ソース接続およびターゲット接続を作成する

1. Data Integration Hub パブリケーションリポジトリへの Hive ソース接続を作成します。Hive は、マッピングが実行されるクラスタに属している必要があります。
2. サブスクライプ元のアプリケーションへの Oracle またはフラットファイルターゲット接続を作成します。

手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する

【物理データオブジェクト】でデータオブジェクトを作成します。

1. ソースデータオブジェクトを作成し、コンSUMするソース接続内のテーブルを選択します。オブジェクトは、リレーショナル Hive データオブジェクトである必要があります。
2. ターゲットデータオブジェクトを作成し、データをサブスクライブするターゲット接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトまたはフラットファイルデータオブジェクトのいずれかにできます。

手順 3. ソースとターゲットを使用してマッピングを作成する

1. 新しいマッピングを作成して、名前を付けます。
2. ソースの物理データオブジェクトを、リーダーとしてマッピングに追加します。
3. ターゲットの物理データオブジェクトを、ライターとしてマッピングに追加します。
4. ソースオブジェクトのすべてのポートを、ターゲットオブジェクトの同一ポートにリンクします。例えば、トピックテーブルにカラム ENAME が含まれている場合、ソースオブジェクトの ENAME ポートをターゲットオブジェクトの ENAME ポートにリンクします。

手順 4.Data Integration Hub パラメータをマッピングに追加する

- ▶ 次のパラメータをマッピングに追加します。

```
<TOPIC_NAME>_DXPublicationInstanceIDs  
<TOPIC_NAME>_DXPublicationInstanceDates  
<TOPIC_NAME>_DXPublicationInstancePartitionDate
```

<TOPIC_NAME>は、サブスクライバがデータをコンシュームするトピックの名前です。

手順 5.フィルタクエリをリーダーオブジェクトに追加する

バインドされていないサブスクリプションのマッピングを作成している場合は、マッピングにフィルタクエリを追加しないでください。

1. 次のマッピングパラメータで、ソースのフィルタクエリを設定します。

```
<TOPIC_NAME>_DXPublicationInstanceIDs  
<TOPIC_NAME>_DXPublicationInstanceDates  
<TOPIC_NAME>_DXPublicationInstancePartitionDate
```

<TOPIC_NAME>は、サブスクライバがデータをコンシュームするトピックの名前です。

フィルタクエリのパラメータは引用符で囲まないでください。

例えば、パーティションの日付パラメータのフィルタ条件を使用して、フィルタクエリの次の形式を使用します。

```
dih_publication_instance_id in ($MY_TOPIC_PublicationInstanceIDs) and dih_publication_instance_date in  
($MY_TOPIC_PublicationInstanceDates) and publication_date_partition in  
($MY_TOPIC_PublicationInstancePartitionDate)
```

2. マッピングを保存する。

手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する

1. **【プロパティ】** ペインで **【ランタイム】** を選択し、**【検証環境】** で **【Hadoop】** と **【Hive】** を順に選択します。**【ネイティブ】** が選択されていないことを確認します。
2. マッピングからアプリケーションを作成します。
マッピングはデータ統合サービスの Hadoop 環境にデプロイされます。

パブリケーション用の Data Engineering Integration ワークフローの開発

パブリケーション用の Data Engineering Integration ワークフローを開発するには、Developer tool で次の手順を実行します。

1. パブリケーション用に Data Engineering Integration マッピングを開発します。
2. パブリケーション用のワークフローを作成し、パブリケーションマッピングまたは同じトピックに属するマッピングを含めます。Start_Event および End_Event を特徴とする 1 つのパイプラインに複数のマッピングを含めることができます。
3. マッピングおよびワークフローを検証し、データ統合サービス内のアプリケーションにワークフローをデプロイします。

4. Data Integration Hub コンソールでパブリケーション用の Data Engineering Integration ワークフローを作成します。

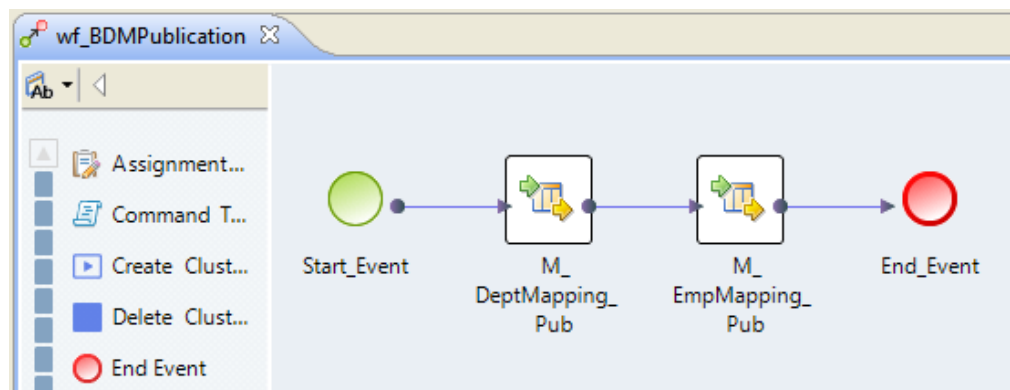
手順 1.パブリケーション用の Data Engineering Integration マッピングの開発

[「パブリケーション用の Data Engineering Integration マッピングの開発」 \(ページ 40\)](#)で説明するように、パブリケーション用の Data Engineering Integration マッピングを開発します。

手順 2.パブリケーションワークフローを作成する

パブリケーションワークフローの作成手順は、次のとおりです。

1. パブリケーション用のワークフローを作成します。Start_Event および End_Event 内の 1 つのパイプラインに複数のマッピングを含めることができます。

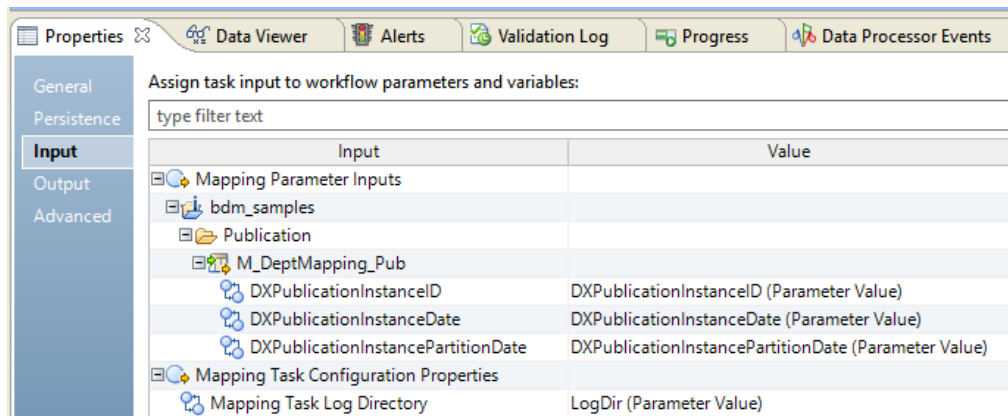


2. **【プロパティ】 > 【パラメータ】** メニューで、次のワークフローパラメータを作成します。

- DXPublicationInstanceID
- DXPublicationInstanceDate
- DXPublicationInstancePartitionDate

Properties					
Data Viewer Alerts Validation Log Progress Data Pro					
General					
Parameters	Name	Type	Precision	Scale	Default Value
Variables	1 DXPublicationInstanceID	String	1000	0	Default
Advanced	2 DXPublicationInstanceDate	String	1000	0	Default
	3 DXPublicationInstancePartitionDate	String	1000	0	Default

3. ワークフロー内に含まれるマッピングごとに、対応するマッピングパラメータにワークフローパラメータを割り当てます。次のワークフローメニューを使用します。**【ワークフロー】 > 【マッピング】 > 【プロパティ】 > 【入力】 > 【パラメータのマッピング】**。



4. このワークフローを保存します。

手順 3.パブリケーションワークフローをデプロイする

パブリケーションワークフローを選択し、Data Integration Hub コンソールのデータ統合サービスにデプロイします。これで、Data Integration Hub 管理者は、ワークフローにアクセスしてパブリケーション用の Data Engineering Integration ワークフローを作成できます。

サブスクリプション用の Data Engineering Integration ワークフローの開発

サブスクリプション用の Data Engineering Integration ワークフローを開発するには、Developer tool で次の手順を実行します。

1. サブスクリプション用の Data Engineering Integration マッピングを開発します。
2. サブスクリプション用のワークフローを作成し、サブスクリプションマッピングまたは同じトピックに属するマッピングを含めます。Start_Event および End_Event を特徴とする 1 つのパイプラインに複数のマッピングを含めることができます。
3. マッピングおよびワークフローを検証し、データ統合サービス内のアプリケーションにワークフローをデプロイします。
4. Data Integration Hub コンソールでサブスクリプション用の Data Engineering Integration ワークフローを作成します。

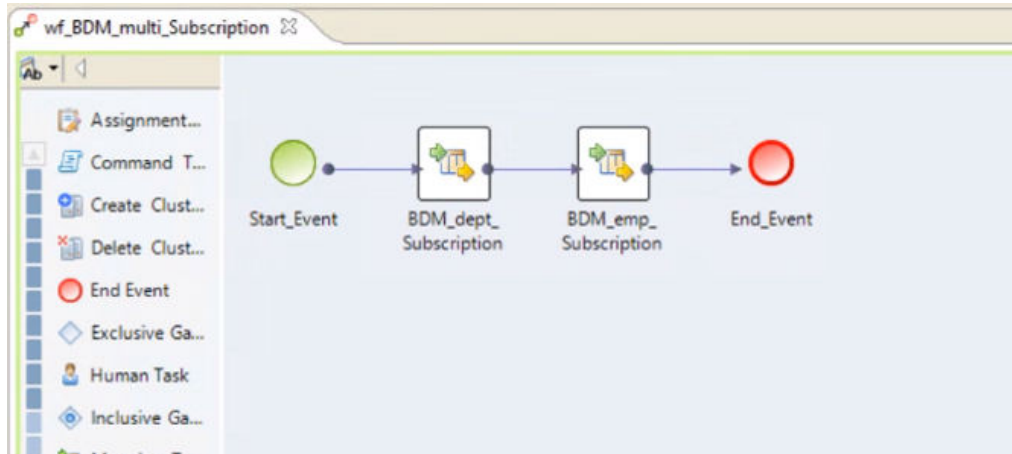
手順 1: サブスクリプション用の Data Engineering Integration マッピングの開発

[「サブスクリプション用の Data Engineering Integration マッピングの開発」 \(ページ 44\)](#)で説明するように、サブスクリプション用の Data Engineering Integration マッピングを開発します。

手順 2. サブスクリプションワークフローを作成する

サブスクリプションワークフローの作成手順は、次のとおりです。

1. サブスクリプション用のワークフローを作成します。Start_Event および End_Event 内に含まれる 1 つのパイプラインに複数のマッピングを含めることができます。



2. [プロパティ] > [パラメータ] メニューで、次のワークフローパラメータを作成します。

- bdm_multiple_topic_PublicationInstanceIDs
- bdm_multiple_topic_PublicationInstanceDates
- bdm_multiple_topic_PublicationInstancePartition_Date

	Name	Type	Precision	Scale	Default Value	Description
1	bdm_multitable_topic_PublicationInstanceIDs	String	1000	0	Default	
2	bdm_multitable_topic_PublicationInstanceDates	String	1000	0	Default	
3	bdm_multitable_topic_PublicationInstancePartition_Date	String	1000	0	Default	

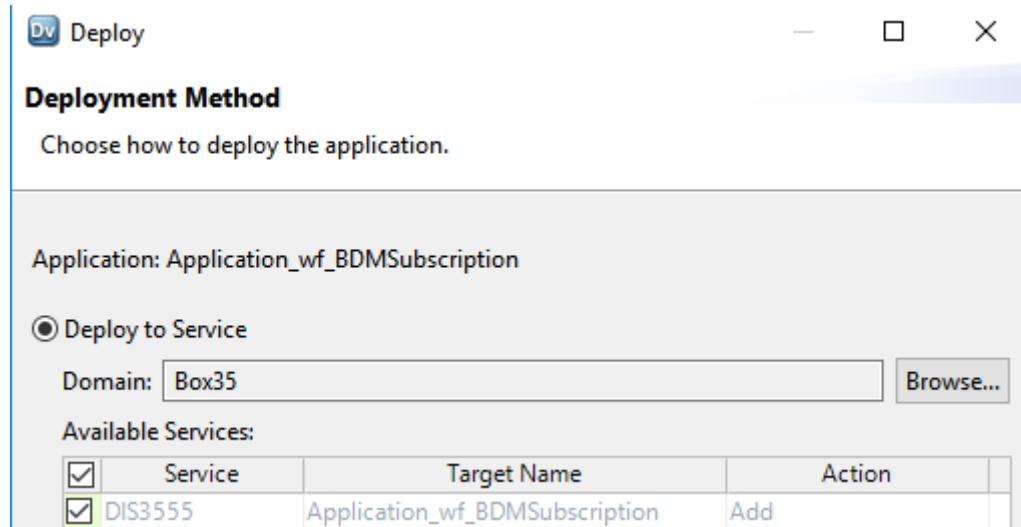
3. ワークフロー内に含まれるマッピングごとに、対応するマッピングパラメータにワークフローパラメータを割り当てます。次のワークフローメニューを使用します。[ワークフロー] > [マッピング] > [プロパティ] > [入力] > [パラメータのマッピング]。

Input	Value
Mapping Parameter Inputs	
bdm_samples	
Publication	
M_DeptMapping_Pub	
DXPublicationInstanceID	DXPublicationInstanceID (Parameter Value)
DXPublicationInstanceDate	DXPublicationInstanceDate (Parameter Value)
DXPublicationInstancePartitionDate	DXPublicationInstancePartitionDate (Parameter Value)
Mapping Task Configuration Properties	
Mapping Task Log Directory	LogDir (Parameter Value)

4. このワークフローを保存します。

手順 3. サブスクリプションワークフローをデプロイする

サブスクリプションワークフローを選択し、Data Integration Hub コンソールのデータ統合サービスにデプロイします。これで、Data Integration Hub 管理者は、ワークフローにアクセスしてサブスクリプション用の Data Engineering Integration ワークフローを作成できます。



Deploy

Deployment Method
Choose how to deploy the application.

Application: Application_wf_BDMSubscription

☒ Deploy to Service

Domain:

Available Services:

<input checked="" type="checkbox"/>	Service	Target Name	Action
<input checked="" type="checkbox"/>	DIS3555	Application_wf_BDMSubscription	Add

第 4 章

Data Quality マッピングおよびワークフロー

この章では、以下の項目について説明します。

- [Data Quality マッピングおよびワークフローの概要, 51 ページ](#)
- [Data Integration Hub の Data Quality マッピングおよびワークフロー, 52 ページ](#)
- [作業を開始する前に, 52 ページ](#)
- [パブリケーション用の Data Quality マッピングの開発, 52 ページ](#)
- [サブスクリプション用の Data Quality マッピングの開発, 54 ページ](#)
- [パブリケーション用の Data Quality ワークフローの開発, 56 ページ](#)
- [サブスクリプション用の Data Quality ワークフローの開発, 58 ページ](#)

Data Quality マッピングおよびワークフローの概要

Data Quality とは、オンプレミスアプリケーションの Data Integration Hub のカスタムパブリケーションおよびサブスクリプションを実行するために Data Integration Hub が使用する処理エンジンです。データ統合サービスはネイティブ環境で Data Quality マッピングおよびワークフローを実行します。

Developer tool を使用して、パブリケーションおよびサブスクリプションを処理する Data Quality マッピングを開発します。開発した Data Quality マッピングは、Data Integration Hub 操作コンソールを使用して、Data Integration Hub ワークフローにインポートします。詳細については、「[Data Integration Hub ワークフローの作成](#)」(ページ 66)を参照してください。

複数の Data Quality マッピングを使用して、Data Quality ワークフローを作成します。

Data Integration Hub のオペレータは、Data Integration Hub 操作コンソールでパブリケーションまたはサブスクリプションを作成し、Data Quality マッピングや Data Quality ワークフローに基づいた Data Integration Hub ワークフローを選択します。詳細については、『*Data Integration Hub オペレータガイド*』を参照してください。

次のディレクトリにサンプルマッピングがあります。<DIHInstallationDir>/samples/idq_mappings 各サンプルマッピングには、そのサンプルマッピングについての説明と使用方法が述べられた関連する readme ファイルがあります。

Data Integration Hub の Data Quality マッピングおよびワークフロー

Data Quality マッピングおよびワークフローは、カスタムマッピングを使用する Data Integration Hub のパブリケーションおよびサブスクリプションで使⽤します。

パブリケーションマッピングでは、ソースはパブリッシュ元のアプリケーション、ターゲットは Data Integration Hub パブリケーションリポジトリです。サブスクリプションマッピングでは、ソースは Data Integration Hub パブリケーションリポジトリ、ターゲットはサブスクライブ元のアプリケーションです。

ユーザー定義のセッションパラメータをカスタムワークフローで使⽤し、それらの値を Data Integration Hub またはパラメータファイルで定義できます。ユーザー定義のセッションパラメータの値は、Forms Designer の Data Integration Hub で管理できます。組み込みセッションパラメータの値は、Data Integration Hub では管理できません。セッションパラメータの詳細については、『PowerCenter 上級ワークフローガイド』の「セッションパラメータに関する作業」セクションを参照します。

注: Data Quality マッピングおよびワークフローを使⽤してファイル転送でファイルをパブリッシュするには、セッションパラメータ名の先頭に \$InputFile_ が付いている必要があります。

作業を開始する前に

Data Integration Hub のカスタムビッグデータのパブリケーションおよびサブスクリプションで使⽤する Data Quality マッピングと Data Quality ワークフローを開発する前に、次の条件が満たされていることを確認します。

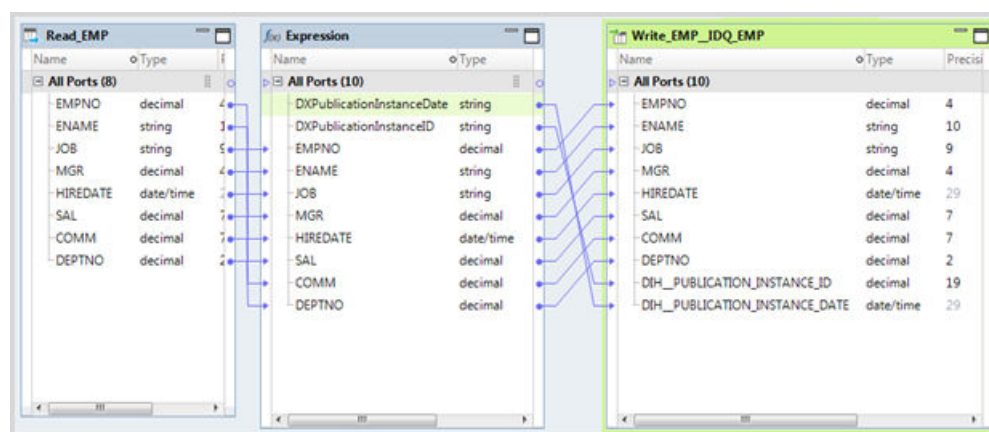
- Informatica プラットフォームが組織にインストールされている。
- Data Integration Hub の Data Quality コンポーネントがインストールされている。
- データのパブリッシュ先およびコンシューム元のトピックが Data Integration Hub で設定されている。
- 次の接続が Data Integration Hub で設定されている。
 - パブリッシュ元のアプリケーションへの接続。
 - サブスクライブ元のアプリケーションへの接続。

パブリケーション用の Data Quality マッピングの開発

パブリケーション用の Data Quality マッピングを開発するには、Developer tool で次の手順を実行します。

1. ソース接続とターゲット接続を作成する。ソース接続はパブリッシュ元のアプリケーションへの接続、ターゲット接続は Data Integration Hub パブリケーションリポジトリへの接続です。
2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成します。
3. マッピングを作成し、ソースオブジェクトおよびターゲットオブジェクトをマッピングに追加します。
4. Data Integration Hub パラメータをマッピングに追加します。

5. 式トランスフォーメーションをマッピングに追加し、トランスフォーメーションのポートを設定して、ソース、トランスフォーメーション、およびターゲット間でポートを接続します。
 6. マッピングのランタイム環境を設定し、マッピングからアプリケーションを作成します。
- 次の図に、パブリケーションマッピングの例を示します。



手順 1. ソース接続およびターゲット接続を作成する

1. パブリッシュ元へのアプリケーションへの Oracle またはフラットファイルソース接続を作成します。
2. Data Integration Hub パブリケーションリポジトリへのリレーショナルターゲット接続を作成します。

手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する

【物理データオブジェクト】でデータオブジェクトを作成します。

1. ソースデータオブジェクトを作成し、パブリッシュするソース接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトまたはフラットファイルデータオブジェクトのいずれかにできます。
2. ターゲットデータオブジェクトを作成し、ソースのデータのパブリッシュ先となるターゲット接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトである必要があります。

手順 3. ソースとターゲットを使用してマッピングを作成する

1. 新しいマッピングを作成して、名前を付けます。
2. ソースの物理データオブジェクトを、リーダーとしてマッピングに追加します。
3. ターゲットの物理データオブジェクトを、ライターとしてマッピングに追加します。

手順 4. Data Integration Hub パラメータをマッピングに追加する

- ▶ 次のパラメータをマッピングに追加します。

DXPublicationInstanceID
DXPublicationInstanceDate

手順 5. マッピングへの式トランスフォーメーションの追加

1. 式トランスフォーメーションを、ソースオブジェクトとターゲットオブジェクト間のマッピングに追加します。
2. ソースオブジェクトのすべてのポートを式トランスフォーメーションの同一ポートにリンクします。例えば、トピックテーブルにカラム EMPNO が含まれている場合、ソースオブジェクトの EMPNO ポートを式トランスフォーメーションの EMPNO ポートにリンクします。
3. 式トランスフォーメーションで次の追加ポートを設定します。
DXPublicationInstanceID
DXPublicationInstanceDate
すべてのポートのデータ型は文字列で、最小精度は 200 です。
4. 式トランスフォーメーションのすべてのポートをターゲットオブジェクトの同一ポートにリンクします。例えば、式トランスフォーメーションの EMPNO ポートをターゲットオブジェクトの EMPNO ポートにリンクし、式トランスフォーメーションの DXPublicationInstanceID ポートをターゲットオブジェクトの dih_PublicationInstanceID ポートにリンクします。
5. マッピングを保存する。

手順 6. マッピングのランタイム環境を設定し、アプリケーションを作成する

1. **【プロパティ】** ペインで **【ランタイム】** を選択し、**【検証環境】** で **【ネイティブ】** を選択します。
2. マッピングからアプリケーションを作成します。
マッピングはデータ統合サービスのネイティブ環境にデプロイされます。

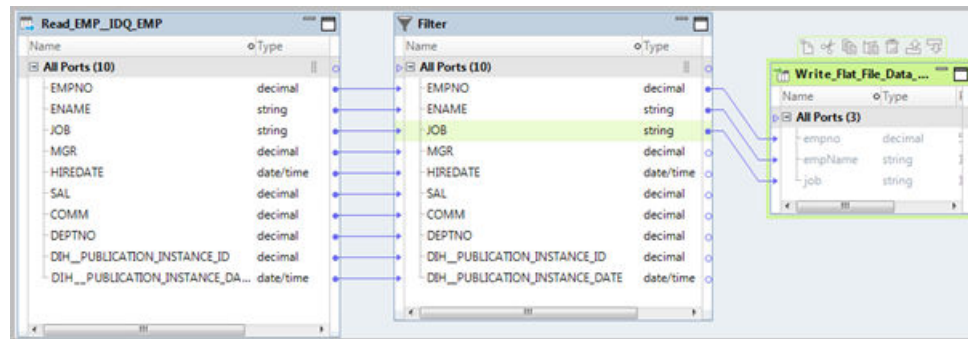
サブスクリプション用の Data Quality マッピングの開発

サブスクリプション用の Data Quality マッピングを開発するには、Developer tool で次の手順を実行します。

1. ソース接続とターゲット接続を作成する。ソース接続は Data Integration Hub パブリケーションリポジトリへの接続、ターゲット接続はサブスクライブ元のアプリケーションへの接続です。
2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成します。
3. マッピングを作成し、ソースオブジェクトおよびターゲットオブジェクトをマッピングに追加します。
4. Data Integration Hub パラメータをマッピングに追加します。
5. フィルタトランスフォーメーションをマッピングに追加し、トランスフォーメーションフィルタを設定して、ソース、トランスフォーメーション、およびターゲット間でポートを接続します。
6. フィルタクエリをリーダーオブジェクトに追加します。
7. マッピングのランタイム環境を設定し、マッピングからアプリケーションを作成します。

注: バインドされていないサブスクリプションのマッピングを作成している場合は、マッピングにフィルタトランスフォーメーションとフィルタクエリを追加する必要はありません。

次の図に、サブスクリプションマッピングの例を示します。



手順 1. ソース接続およびターゲット接続を作成する

1. Data Integration Hub パブリケーションリポジトリへのリレーショナルソース接続を作成します。
2. サブスクライバ元のアプリケーションへの Oracle またはフラットファイルターゲット接続を作成します。

手順 2. ソースデータオブジェクトおよびターゲットデータオブジェクトを作成する

【物理データオブジェクト】でデータオブジェクトを作成します。

1. ソースデータオブジェクトを作成し、コンSUMするソース接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトである必要があります。
2. ターゲットデータオブジェクトを作成し、データをサブスクライブするターゲット接続内のテーブルを選択します。オブジェクトは、リレーショナルデータオブジェクトまたはフラットファイルデータオブジェクトのいずれかにできます。

手順 3. ソースとターゲットを使用してマッピングを作成する

1. 新しいマッピングを作成して、名前を付けます。
2. ソースの物理データオブジェクトを、リーダーとしてマッピングに追加します。
3. ターゲットの物理データオブジェクトを、ライターとしてマッピングに追加します。

手順 4. Data Integration Hub パラメータをマッピングに追加する

- ▶ 次のパラメータをマッピングに追加します。
<TOPIC_NAME>_DXPublicationInstanceIDs
<TOPIC_NAME>は、サブスクライバがデータをコンSUMするトピックの名前です。

手順 5. フィルタトランスフォーメーションをマッピングに追加する

パブリケーション日にフィルタ条件を追加します。バインドされていないサブスクリプションのマッピングを作成している場合は、マッピングにフィルタトランスフォーメーションを追加しないでください。

1. フィルタトランスフォーメーションを、ソースオブジェクトとターゲットオブジェクト間のマッピングに追加します。
2. ソースオブジェクトのすべてのポートをフィルタトランスフォーメーションの同一ポートにリンクします。例えば、ソースオブジェクトの EMPNO ポートをフィルタトランスフォーメーションの EMPNO ポートにリンクし、ソースオブジェクトの DXPublicationInstanceID ポートをフィルタトランスフォーメーションの DXPublicationInstanceID ポートにリンクします。
3. フィルタトランスフォーメーションの [フィルタ] タブで、次のパラメータを設定します。
<TOPIC_NAME>__PublicationInstanceDates
<TOPIC_NAME>は、サブスクライバがデータをコンSUMするトピックの名前です。
[デフォルト値] が TRUE に設定されていることを確認します。
4. マッピングを保存する。

手順 6. フィルタクエリをリーダーオブジェクトに追加する

パブリケーション ID パラメータにフィルタ条件を追加します。バインドされていないサブスクリプションのマッピングを作成している場合は、マッピングにフィルタクエリを追加しないでください。

1. 次のマッピングパラメータで、ソースのフィルタクエリを設定します。
<TOPIC_NAME>__DXPublicationInstanceIDs
<TOPIC_NAME>は、サブスクライバがデータをコンSUMするトピックの名前です。
フィルタクエリのパラメータは引用符で囲まないのでください。
例:
dih_publication_instance_id in (\$MY_TOPIC__PublicationInstanceIDs)
2. マッピングを保存する。

手順 7. マッピングのランタイム環境を設定し、アプリケーションを作成する

1. [プロパティ] ペインで [ランタイム] を選択し、[検証環境] で [ネイティブ] を選択します。
2. マッピングからアプリケーションを作成します。
マッピングはデータ統合サービスのネイティブ環境にデプロイされます。

パブリケーション用の Data Quality ワークフローの開発

パブリケーション用の Data Quality マッピングを開発するには、Developer tool で次の手順を実行します。

1. パブリケーション用の Data Quality マッピングを開発します。

2. パブリケーション用のワークフローを作成し、パブリケーションマッピングまたは同じトピックに属するマッピングを含めます。Start_Event および End_Event を特徴とする 1 つのパイプラインに複数のマッピングを含めることができます。
3. マッピングおよびワークフローを検証し、データ統合サービス内のアプリケーションにワークフローをデプロイします。
4. Data Integration Hub コンソールでパブリケーション用の Data Quality ワークフローを作成します。

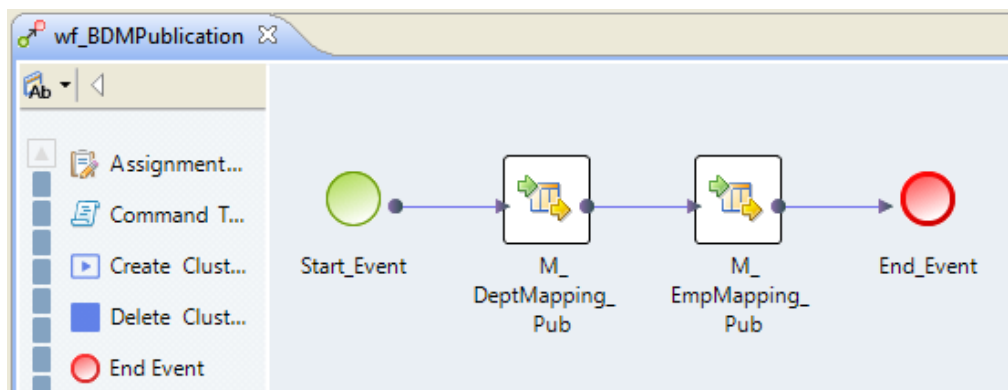
手順 1.パブリケーション用の Data Quality マッピングを開発する

[「パブリケーション用の Data Quality マッピングの開発」 \(ページ 52\)](#)に記載された説明に従って、パブリケーション用の Data Quality マッピングを開発します。

手順 2.パブリケーションワークフローを作成する

パブリケーションワークフローの作成手順は、次のとおりです。

1. パブリケーション用のワークフローを作成します。Start_Event および End_Event 内の 1 つのパイプラインに複数のマッピングを含めることができます。

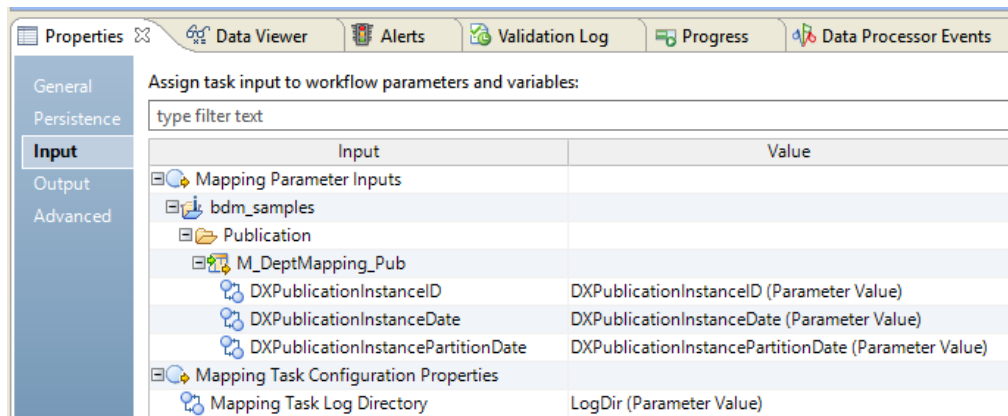


2. **【プロパティ】 > 【パラメータ】** メニューで、次のワークフローパラメータを作成します。

- DXPublicationInstanceID
- DXPublicationInstanceDate
- DXPublicationInstancePartitionDate

Properties						
Data Viewer						
Alerts						
Validation Log						
Progress						
Data Proc						
General						
Parameters						
Variables						
Advanced						
Name	Type	Precision	Scale	Default Value		
1 DXPublicationInstanceID	String	1000	0	Default		
2 DXPublicationInstanceDate	String	1000	0	Default		
3 DXPublicationInstancePartitionDate	String	1000	0	Default		

3. ワークフロー内に含まれるマッピングごとに、対応するマッピングパラメータにワークフローパラメータを割り当てます。次のワークフローメニューを使用します。**【ワークフロー】 > 【マッピング】 > 【プロパティ】 > 【入力】 > 【パラメータのマッピング】**。



4. このワークフローを保存します。

手順 3.パブリケーションワークフローをデプロイする

- ▶ パブリケーションワークフローを選択し、Data Integration Hub コンソールのデータ統合サービスにデプロイします。これで、Data Integration Hub 管理者は、ワークフローにアクセスしてパブリケーション用の Data Quality ワークフローを作成できます。

サブスクリプション用の Data Quality ワークフローの開発

サブスクリプション用の Data Quality ワークフローを開発するには、Developer tool で次の手順を実行します。

1. サブスクリプション用の Data Quality マッピングを開発します。
2. サブスクリプション用のワークフローを作成し、サブスクリプションマッピングまたは同じトピックに属するマッピングを含めます。Start_Event および End_Event を特徴とする 1 つのパイプラインに複数のマッピングを含めることができます。
3. マッピングおよびワークフローを検証し、データ統合サービス内のアプリケーションにワークフローをデプロイします。
4. Data Integration Hub コンソールでサブスクリプション用の Data Quality ワークフローを作成します。

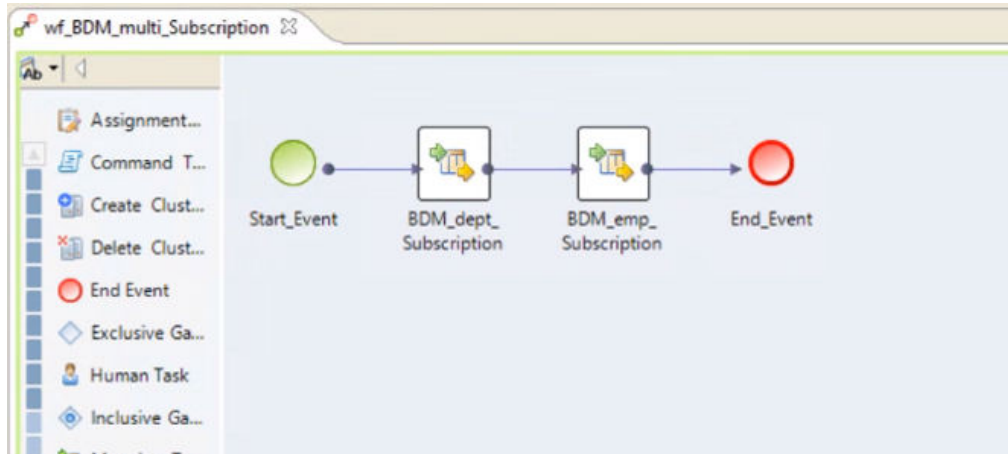
手順 1.サブスクリプション用の Data Quality マッピングを開発する

[「サブスクリプション用の Data Quality マッピングの開発」 \(ページ 54\)](#)に記載された説明に従って、サブスクリプション用の Data Quality マッピングを開発します。

手順 2. サブスクリプションワークフローを作成する

サブスクリプションワークフローの作成手順は、次のとおりです。

1. サブスクリプション用のワークフローを作成します。Start_Event および End_Event 内に含まれる 1 つのパイプラインに複数のマッピングを含めることができます。



2. [プロパティ] > [パラメータ] メニューで、次のワークフローパラメータを作成します。

- bdm_multiple_topic_PublicationInstanceIDs
- bdm_multiple_topic_PublicationInstanceDates
- bdm_multiple_topic_PublicationInstancePartition_Date

Name	oType	Precision	Scale	Default Value	Description
bdm_multitable_topic_PublicationInstanceIDs	String	1000	0	Default	
bdm_multitable_topic_PublicationInstanceDates	String	1000	0	Default	
bdm_multitable_topic_PublicationInstancePartition_Date	String	1000	0	Default	

3. ワークフロー内に含まれるマッピングごとに、対応するマッピングパラメータにワークフローパラメータを割り当てます。次のワークフローメニューを使用します。[ワークフロー] > [マッピング] > [プロパティ] > [入力] > [パラメータのマッピング]。

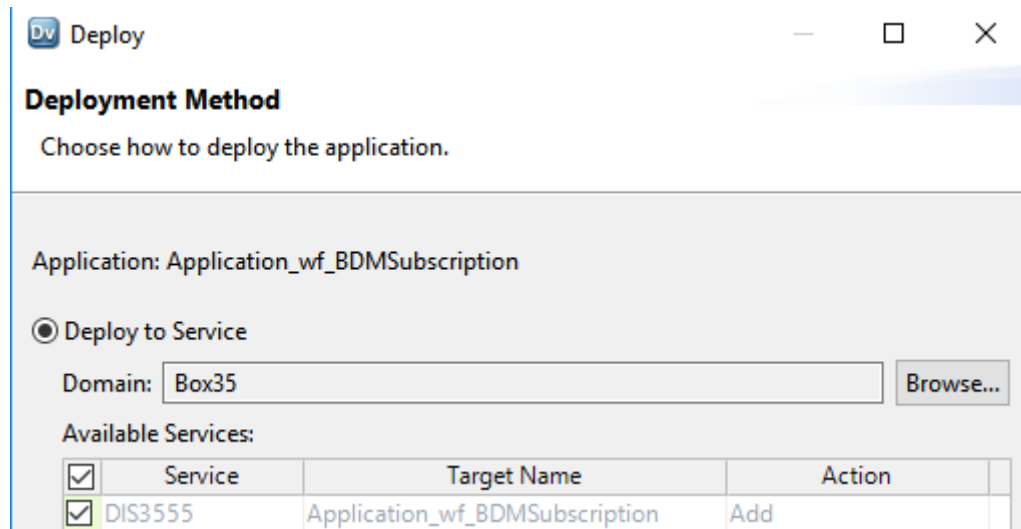
Input	Value
Mapping Parameter Inputs	
bdm_samples	
Publication	
M_DeptMapping_Pub	
DXPublicationInstanceID	DXPublicationInstanceID (Parameter Value)
DXPublicationInstanceDate	DXPublicationInstanceDate (Parameter Value)
DXPublicationInstancePartitionDate	DXPublicationInstancePartitionDate (Parameter Value)
Mapping Task Configuration Properties	
Mapping Task Log Directory	LogDir (Parameter Value)

4. このワークフローを保存します。

手順 3. サブスクリプションワークフローをデプロイする

- ▶ サブスクリプションワークフローを選択し、Data Integration Hub コンソールのデータ統合サービスにデプロイします。

これで、Data Integration Hub 管理者は、ワークフローにアクセスしてサブスクリプション用の Data Quality ワークフローを作成できます。



Deploy

Deployment Method
Choose how to deploy the application.

Application: Application_wf_BDMSubscription

☒ Deploy to Service

Domain:

Available Services:

<input checked="" type="checkbox"/>	Service	Target Name	Action
<input checked="" type="checkbox"/>	DIS3555	Application_wf_BDMSubscription	Add

第 5 章

Informatica クラウドマッピング およびタスク

この章では、以下の項目について説明します。

- [Informatica クラウドマッピングおよびタスクの概要, 61 ページ](#)
- [Data Integration Hub の Informatica Cloud マッピング, 62 ページ](#)
- [Data Integration Hub の Data Integration タスク, 62 ページ](#)

Informatica クラウドマッピングおよびタスクの概要

Data Integration Hub は、Informatica Cloud マッピングおよびタスクを使用して、ソースのクラウドアプリケーションから Data Integration Hub パブリケーションリポジトリにデータをパブリッシュします。このパブリケーションリポジトリのデータは、ターゲットのクラウドアプリケーションによって消費されます。

自動マッピングを使用するパブリケーションおよびサブスクリプションでは、Informatica Cloud マッピングを使用してデータ処理を定義します。

自動マッピングを使用するパブリケーションおよびサブスクリプションでは、データ同期タスクとマッピング設定タスクを使用してデータ処理を定義します。

Data Integration Hub 用の Informatica Cloud マッピングおよびタスクは、他の Informatica Cloud マッピングおよびタスクと同じ方法で開発します。Data Integration Hub 接続を、パブリケーションマッピングおよびタスクのターゲットと、サブスクリプションマッピングおよびタスクのソースとして使用します。

Informatica Cloud で Data Integration Hub 用のマッピングおよびタスクを開発する前に、クラウド Data Integration Hub コネクタがインストールされていて、そのコネクタを使用する接続が存在することを確認します。詳細については、『*Data Integration Hub 管理者ガイド*』を参照してください。

Data Integration Hub の Informatica Cloud マッピング

Informatica Cloud マッピングは、自動マッピングを使用する Data Integration Hub のパブリケーションおよびサブスクリプションで使⽤します。

パブリケーションマッピングでは、パブリッシュ元のアプリケーションがマッピングのソースで、Data Integration Hub パブリケーションリポジトリがマッピングのターゲットです。サブスクリプションマッピングでは、Data Integration Hub パブリケーションリポジトリがマッピングのソースで、サブスクライブ元のアプリケーションがマッピングのターゲットです。

汎用のパラメータ化されたマッピングを作成し、複数のパブリケーションおよびサブスクリプションに使用できます。マッピングの任意の要素（ソースおよびターゲット接続、フィールドマッピング、トランスフォーメーションなど）をパラメータ化できます。オペレータは、Data Integration Hub 操作コンソールでパブリケーションまたはサブスクリプションを作成するときにパラメータ値を入力します。

Informatica Cloud マッピングのルールとガイドライン

自動マッピングを使用する Data Integration Hub パブリケーションおよびサブスクリプションで使用する Informatica Cloud マッピングを開発する際は、次のルールとガイドラインを考慮してください。

- Data Integration Hub 用に作成したマッピングは、Informatica Cloud 内からは実行しないでください。Data Integration Hub マッピングは、このマッピングが関連付けられているパブリケーションまたはサブスクリプションを実行することによって、Data Integration Hub から実行する必要があります。
- Data Integration Hub 接続を使用するときにマッピングがパラメータ化されていない場合は、パブリケーションマッピングに含まれるターゲットオブジェクトと、サブスクリプションマッピングに含まれるソースオブジェクトが、Data Integration Hub に定義されているトピックのリストを表します。このリストの形式は、TopicName.tableName です。
- マッピングがパラメータ化されている場合は、パブリケーションマッピングとサブスクリプションマッピングの両方として使用できます。
- マッピングがパラメータ化されておらず、パブリケーションマッピングとサブスクリプションマッピングを区別する場合は、マッピング名にマッピングの種類を指定します。例えば、パブリケーションマッピングの名前は Pub_<MappingName>とし、サブスクリプションマッピングの名前は Sub_<MappingName>とします。
- マッピングに式とフィルタトランスフォーメーションを追加できます。
- マッピングにパラメータが含まれている場合、オペレータがパブリケーションまたはサブスクリプションを作成すると、パブリケーションまたはサブスクリプションウィザードの **【入力パラメータ】** タブにパラメータが表示されます。

Data Integration Hub の Data Integration タスク

Data Integration Hub のカスタムマッピングを使用するパブリケーションおよびサブスクリプションで、Data Integration タスクを使用します。

パブリケーションタスクでは、パブリッシュ元のアプリケーションがタスクのソースで、Data Integration Hub パブリケーションリポジトリがタスクのターゲットです。サブスクリプションタスクでは、Data Integration Hub パブリケーションリポジトリがタスクのソースで、サブスクライブ元のアプリケーションがタスクのターゲットです。

Data Integration Hub パブリケーション用の Data Integration タスクを開発する場合は、ターゲットオブジェクトとして Data Integration Hub 接続を選択します。Data Integration Hub サブスクリプション用の Data Integration タスクを開発する場合は、ソースオブジェクトとして Data Integration Hub 接続を選択します。

注: Data Integration タスクをデータドリブンのパブリケーションおよびサブスクリプションで使用しないでください。

Data Integration タスクのルールとガイドライン

カスタムマッピングによる Data Integration Hub パブリケーションおよびサブスクリプションで使用する Informatica Cloud のマッピングおよびタスクを開発する際には、次のルールとガイドラインを考慮します。

一般的なルールとガイドライン

Data Integration マッピングおよびタスクを作成する際には、次のルールとガイドラインを考慮します。

- Data Integration Hub 用に作成したタスクは、Informatica Intelligent Cloud Services 内からは実行しないでください。タスクは、このタスクが関連付けられているパブリケーションまたはサブスクリプションを実行することによって、Data Integration Hub から実行する必要があります。
- Data Integration Hub 接続を使用する場合は、パブリケーションマッピングまたはタスクに含まれるターゲットオブジェクトと、サブスクリプションマッピングまたはタスクに含まれるソースオブジェクトが、Data Integration Hub に定義されているトピックのリストを表します。このリストの形式は、TopicName.tableName です。
- Data Integration Hub では、オペレータがパブリケーションウィザードまたはサブスクリプションで定義した設定に基づいて、パブリケーションまたはサブスクリプションのスケジューリングが決定されます。Data Integration タスクを作成する際には、タスクウィザードの **【スケジュール】** ページで、**【このタスクはスケジュールを使用しない】** オプションが選択されていることを確認します。
- パブリケーションタスクとサブスクリプションタスクを区別するために、タスク名にタスクのタイプを示すようにします。パブリケーション用またはサブスクリプション用にタスクを選択するときに、適切なタスクを簡単に選択できます。
例えば、パブリケーションタスクの名前は Pub_<TaskName>とし、サブスクリプションタスクの名前は Sub_<TaskName>とします。

同期タスクのルールとガイドライン

同期タスクおよびマッピングを作成する場合は、次のルールとガイドラインを考慮します。

- パブリケーションタスクのタスク操作は、挿入操作です。
- パブリケーションタスクを作成する場合は、ターゲットとして、Data Integration Hub 接続を選択します。サブスクリプションタスクを作成する場合は、ソースとして、Data Integration Hub 接続を選択します。
- 同期タスクは複数ソースをサポートしていません。そのため、パブリケーションまたはサブスクリプションの同期タスクを複数のソースを使用して作成する場合、次のユースケースに対してソース間の関係を作成します。
 - パブリケーション: 複数のテーブルからパブリッシュする場合。
 - サブスクリプション: 複数のテーブルへサブスクライブする場合、またはサブスクリプションが複合サブスクリプションの場合。

マッピングタスクのルールとガイドライン

マッピングタスクおよびマッピングを作成する場合は、次のルールとガイドラインを考慮します。

- マッピング操作は、パブリケーションマッピングとサブスクリプションマッピングの両方に対する挿入操作です。
- パブリケーションマッピングを作成する場合は、ターゲットプロパティを設定するときに、Data Integration Hub 接続を選択します。サブスクリプションマッピングを作成する場合は、ソースプロパティを設定するときに、Data Integration Hub 接続を選択します。

Informatica Intelligent Cloud Services マッピングおよびタスクを Data Integration Hub 用に作成するための手順については、『*Data Integration Hub* コネクタガイド』を参照してください。

Data Integration Hub でのインテリジェント構造モデルの使用

Data Integration Hub は、非構造化フラットファイルによるデータのパブリッシュをサポートしています。インテリジェント構造モデルを使用して非構造化ファイルや半構造化ファイルのデータを構造化し、インテリジェント構造モデルを使用して Data Integration Hub に非構造化データをパブリッシュするカスタムクラウドマッピングを作成できます。

Data Integration Hub でインテリジェント構造モデルを使用して非構造化データや半構造化データをパブリッシュする方法の詳細については、「H2L ...」を参照してください。

第 6 章

Data Integration Hub ワークフロー

この章では、以下の項目について説明します。

- [Data Integration Hub ワークフローの概要, 65 ページ](#)
- [ワークフロー権限, 66 ページ](#)
- [ワークフロー管理, 66 ページ](#)
- [Data Integration Hub ワークフローのプロパティ, 70 ページ](#)

Data Integration Hub ワークフローの概要

Data Integration Hub ワークフローは、次のアクションのいずれかを定義する指示のセットです。

- オンプレミスアプリケーションが、Data Integration Hub パブリケーションリポジトリとの間でパブリッシュおよびコンシュームするデータをカスタムのパブリケーションおよびサブスクリプションによって処理する方法。カスタムパブリケーションおよびサブスクリプションの Data Integration Hub ワークフローは、PowerCenter ワークフロー、Data Engineering Integration マッピングおよびワークフロー、Data Engineering Streaming マッピング、または Data Quality マッピングおよびワークフローに基づくことができます。
- データを前処理してからパブリッシュする方法。Data Integration Hub の前処理ワークフローは、PowerCenter ワークフローに基づいている必要があります。
- サブスクライブするデータを後処理する方法。Data Integration Hub の後処理ワークフローは、PowerCenter ワークフローに基づいている必要があります。
- パブリケーションまたはサブスクリプションイベントが、定義されたステータスになったときにどのようなアクションを実施するか。このステータスは、PowerCenter ワークフローを起動するイベント監視ルールに定義します。監視ルール用の Data Integration Hub ワークフローは、PowerCenter ワークフローに基づいている必要があります。

Data Integration Hub でワークフローを作成する前に、関連する PowerCenter ワークフロー、Data Engineering Integration マッピングまたはワークフロー、Data Engineering Integration ストリーミング、Data Quality マッピングまたはワークフローを作成する必要があります。詳細については、以下のセクションを参照してください。

- [第 2 章, 「PowerCenter マッピングおよびワークフロー」 \(ページ 22\)](#)
- [第 3 章, 「Data Engineering Integration および Data Engineering Streaming のマッピングとワークフロー」 \(ページ 38\)](#)

- [第4章, 「Data Quality マッピングおよびワークフロー」 \(ページ 51\)](#)

その後、Data Integration Hub ワークフローを作成するときに、ワークフローまたはマッピングを選択します。詳細については、[「Data Integration Hub ワークフローの作成」 \(ページ 66\)](#)を参照してください。

オペレータは、パブリケーション、サブスクリプション、または監視ルールを作成するときに、使用する Data Integration Hub ワークフローを必要に応じて選択します。

以下のパブリケーションおよびサブスクリプションのタイプでは、Data Integration Hub ワークフローを作成する必要はありません。

- 自動マッピングを使用するパブリケーションおよびサブスクリプション。Data Integration Hub は、PowerCenter のワークフローおよびマッピングをパブリケーションまたはサブスクリプションから直接作成します。
- クラウドアプリケーション用のパブリケーションおよびサブスクリプション。パブリケーションまたはサブスクリプションを作成するときに、オペレータが Informatica Intelligent Cloud Services タスクを選択します。

ワークフロー権限

ワークフロー権限は、ワークフローにアクセスできるユーザーを管理します。

Data Integration Hub 管理者は、カテゴリを作成し、ユーザーグループにカテゴリを割り当てて、ワークフローの表示、編集、および削除を行う権限を持つユーザーを決定します。ワークフローを作成または編集するときに、このワークフローのカテゴリを選択できます。ワークフローにカテゴリを選択した場合、選択したカテゴリの少なくとも1つを持つユーザーグループに属しているユーザーのみが、ワークフローへの権限を持ちます。ワークフローにカテゴリを選択しない場合は、すべてのユーザーがワークフローに対する権限を持ちます。

ワークフロー管理

【ワークフローの作成】 ページを使用して、Data Integration Hub ワークフローを作成、編集、および削除します。

Data Integration Hub オペレータは、カスタムパブリケーションおよびサブスクリプションを含むワークフローを使用して、パブリッシュ前のデータの前処理や、サブスクライブするデータの後処理を行い、パブリケーションイベントまたはサブスクリプションイベントが定義されたステータスになったときに実行するアクションを定義します。

Data Integration Hub ワークフローの作成

ナビゲータを使用して、Data Integration Hub ワークフローを作成します。

ワークフローを作成したら、Data Integration Hub は、必要に応じて、パブリケーションウィザードまたはサブスクリプションウィザードでパブリケーションまたはサブスクリプションにワークフローを関連付けるか、監視ルールウィザードで監視ルールにワークフローを関連付けることができます。

Data Integration Hub ワークフローを作成するには、次のタスクを実行します。

1. ワークフローの全般的なパラメータを定義します。

2. ワークフローにユーザー定義のパラメータが含まれる場合は、必要に応じてパラメータをカスタマイズします。
3. 必要に応じて、イベント属性をワークフローに追加します。
4. 必要に応じて、ワークフロー権限を定義します。

手順 1.ワークフローの全般プロパティの定義

[ワークフローの作成] ページの **[全般]** タブで、ワークフローの全般プロパティを定義します。

1. ナビゲータで、**[ハブの管理]** > **[ワークフロー]** > **[新しいワークフロー]** をクリックします。
2. ワークフロー名を入力します。名前には最大 64 文字を含めることができ、スペースおよび特殊文字を含めることができます。
3. 必要に応じて、ワークフローの説明を入力します。
4. ワークフローを実行するプロセッサの種類を選択します。以下のいずれかのプロセッサタイプを選択できます。
 - Data Integration Hub ワークフローを Data Engineering Integration マッピングに関連付けるには、**[Data Engineering Integration マッピング]** を選択します。このマッピングは一度実行し、完了後に停止します。Data Engineering Integration マッピングの割り当て先は、ビッグデータをオンプレミスアプリケーションから Hadoop ベースのトピックにパブリッシュするカスタムバッチパブリケーションと、Hadoop ベースのトピックのビッグデータをオンプレミスアプリケーションにコンSUMするカスタムバッチサブスクリプションに設定できます。
 - Data Integration Hub ワークフローを Data Engineering Integration ワークフローに関連付けるには、**[Data Engineering Integration ワークフロー]** を選択します。Data Engineering Integration ワークフローの割り当て先は、ビッグデータをオンプレミスアプリケーションから Hadoop ベースのトピックにパブリッシュするカスタムバッチパブリケーションと、Hadoop ベースのトピックのビッグデータをオンプレミスアプリケーションにコンSUMするカスタムバッチサブスクリプションに設定できます。

注: Data Engineering Integration ワークフローは、パブリケーションとサブスクリプションの使用のみをサポートします。

 - Data Integration Hub ワークフローを Data Engineering Streaming マッピングに関連付けるには、**[Data Engineering Streaming Real-time Mapping]** を選択します。マッピングは継続的に実行され、ストリーミングソースから読み取りを行います。Data Engineering Streaming マッピングの割り当て先は、ビッグデータストリームをオンプレミスアプリケーションから Hadoop ベースのトピックにパブリッシュするカスタムマルチレイテンシーパブリケーションに設定できます。
 - Data Integration Hub ワークフローを Data Engineering Quality マッピングに関連付けるには、**[Data Quality マッピング]** を選択します。このマッピングは一度実行し、完了後に停止します。Data Engineering Quality マッピングは、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーション、およびデータベースまたはファイルベースのトピックからオンプレミスアプリケーションにデータをコンSUMするカスタムサブスクリプションに割り当てることができます。
 - Data Integration Hub ワークフローを Data Engineering Quality ワークフローに関連付けるには、**[Data Quality ワークフロー]** を選択します。このワークフローは一度実行し、完了後に停止します。Data Engineering Quality ワークフローは、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーション、およびデータベースまたはファイルベースのトピックからオンプレミスアプリケーションに日付をコンSUMするカスタムサブスクリプションに割り当てることができます。

注: Data Engineering Quality ワークフローは、パブリケーションおよびサブスクリプション用途のみをサポートしています。

- **[PowerCenter バッチワークフロー]** を選択すると、Data Integration Hub ワークフローが PowerCenter バッチワークフローに関連付けられます。PowerCenter バッチワークフローがファイル、データベース、または他のソースから読み取りを行います。このワークフローは一度実行し、完了後に停止します。PowerCenter バッチワークフローの割り当て先は、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーション、データベースまたはファイルベースのトピックからオンプレミスアプリケーションにデータをコンシュームするカスタムサブスクリプション、パブリケーションの前処理、サブスクリプションの後処理、PowerCenter ワークフローを起動するイベント監視ルールに設定できます。
 - **[PowerCenter リアルタイムワークフロー]** を選択すると、Data Integration Hub ワークフローをリアルタイムワークフローに関連付けます。PowerCenter リアルタイムワークフローは継続的に実行され、リアルタイムソースから読み取りを行います。PowerCenter リアルタイムワークフローの割り当て先は、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーションに設定できます。リアルタイムワークフローを使用すると、リアルタイムソースからデータを読み取り、そのデータを継続的に Data Integration Hub に書き込むことができます。例えば、Web サービスプロバイダまたは Java Message Service (JMS) キューを使用してデータをパブリッシュする場合に使用します。リアルタイムワークフローによりパブリッシュする際、パブリッシュされたデータを定義済みの時間間隔で単一のパブリケーションにグループ化できます。
5. Data Integration Hub ワークフローを使用するプロセッサのタイプを選択します。次のいずれかのオプションを選択します。
- パブリケーションの前処理。パブリケーションの前処理を実行します。
 - パブリケーション。パブリケーション処理を実行します。
 - サブスクリプション。サブスクリプション処理を実行します。
 - サブスクリプションの後処理。サブスクリプションの後処理を実行します。
 - 監視ルール。パブリケーションまたはサブスクリプションイベントが、イベント監視ルールに定義されたステータスになると、ワークフローが実行されます。
6. Data Integration Hub ワークフローに関連付けるワークフローまたはマッピングを選択します。**[プロセッサタイプ]** で選択したワークフローまたはマッピングのタイプに基づいて選択します。
- マッピングタイプが **[Data Engineering Integration マッピング]** の場合、**[Data Engineering Integration マッピングの追加]** ダイアログボックスが表示されます。Hadoop 環境用にデータ統合サービスにデプロイされているすべての Data Engineering Integration マッピングを表示するには、**[すべて表示]** をクリックします。アプリケーション名でマッピングを検索するには、**[アプリケーション名]** テキストボックスに文字列を入力し、**[検索]** をクリックします。マッピングを選択して **[追加]** をクリックします。
 - マッピングタイプが **[Data Engineering Streaming Real-time Mapping]** の場合、**[Data Engineering Streaming Real-time Mapping の追加]** ダイアログボックスが表示されます。Hadoop 環境用にデータ統合サービスにデプロイされているすべての Data Engineering Streaming マッピングを表示するには、**[すべて表示]** をクリックします。アプリケーション名でマッピングを検索するには、**[アプリケーション名]** テキストボックスに文字列を入力し、**[検索]** をクリックします。マッピングを選択して **[追加]** をクリックします。
 - マッピングタイプが **[Data Engineering Quality マッピング]** の場合、**[Data Engineering Quality マッピングの追加]** ダイアログボックスが表示されます。ネイティブ環境用にデータ統合サービスにデプロイされているすべてのマッピングを表示するには、**[すべて表示]** をクリックします。アプリケーション名でマッピングを検索するには、**[アプリケーション名]** テキストボックスに文字列を入力し、**[検索]** をクリックします。マッピングを選択して **[追加]** をクリックします。

- マッピングタイプが PowerCenter ワークフローの場合、バッチワークフローでもリアルタイムワークフローでも、**[PowerCenter ワークフローの追加]** ダイアログボックスが表示されます。次のいずれかのオプションを選択します。
 - [PowerCenter リポジトリからのワークフローの選択]**。PowerCenter リポジトリに保存した PowerCenter ワークフローを選択して、**[追加]** をクリックします。リポジトリ内のすべてのワークフローを表示するには、**[すべて表示]** をクリックします。名前でワークフローを検索するには、**[フォルダ名]** テキストボックスに文字列を入力し、**[検索]** をクリックします。
 - [ワークフロー定義ファイル (XML) の選択]**。PowerCenter リポジトリからエクスポートした XML PowerCenter ワークフロー定義ファイルを選択して、**[追加]** をクリックします。ファイルを選択するには、参照ボタンをクリックし、必要なファイルの場所を参照し、ファイルを選択し、**[アップロード]** をクリックします。

7. ワークフローパラメータ、イベント属性、または権限を定義しない場合は、**[保存]** をクリックします。

手順 2.ワークフローパラメータのカスタマイズ

ワークフローにパラメータが含まれる場合は、Forms Designer を使用してパラメータに値を追加して、パラメータのレイアウトや動作をカスタマイズします。**[ワークフローの作成]** ページの **[ワークフローパラメータ]** タブで Forms Designer にアクセスします。

注: ワークフローにセッションパラメータが含まれているときに、Data Integration Hub のオペレータがパブリケーションウィザードまたはサブスクリプションウィザードでセッションパラメータを設定しないようにする場合、Forms Designer を使用してセッションパラメータを非表示にします。

1. **[ワークフローパラメータ]** タブをクリックします。

Data Integration Hub ワークフローに関連付けられているワークフローまたはマッピングにユーザー定義のパラメータが含まれる場合は、**[プレビュー]** ペインにワークフローパラメータリストのデフォルトビューが表示されます。

注: **[プレビュー]** ペインではパラメータ値を編集しないでください。

2. **[デザイン]** をクリックします。

[デザイン] ペインにパラメータリストがアルファベット順に表示されます。

3. パラメータおよびグループの位置を変更するには、該当する要素またはグループをドラッグします。

4. パラメータ値を編集して、パラメータ要素の動作および外観を変更するには、パラメータをクリックし、**[アクション]** > **[編集]** をクリックします。

[詳細] ダイアログボックスが表示されます。

5. パラメータのプロパティを変更し、**[保存]** をクリックします。

6. グループを追加するには、**[アクション]** メニューをクリックし、タブ、カラム、ヘッダ、またはディスクロージャを追加する選択をします。

7. カスタマイズしたフォームをプレビューするには、**[プレビュー]** をクリックし、フォームの外観および定義したフィールドの依存性を確認します。

8. ワークフローのイベント属性または権限を定義しない場合は、**[保存]** をクリックします。

手順 3.イベント属性のワークフローへの追加

[ワークフローの作成] ページの **[イベント属性]** タブで、ワークフローにイベント属性を追加します。

イベント属性の詳細については、『*Data Integration Hub 管理者ガイド*』を参照してください。

1. **[イベント属性]** タブをクリックします。

2. **【使用可能な属性キー】** リストで、エクスポートする属性を選択して、**【選択】** アイコンをクリックします。単一の属性または複数の属性を選択できます。
選択した属性は、**【選択した属性キー】** リストに表示されます。
3. ワークフローの権限を定義しない場合は、**【保存】** をクリックします。

手順 4. ワークフロー権限の定義

【ワークフローの作成】 ページの **【権限】** タブで、ワークフローへのアクセスを制御します。

1. **【権限】** タブをクリックします。
2. **【使用可能なカテゴリ】** リストで、ワークフローに追加するカテゴリを選択して、**【選択】** アイコンをクリックします。単一のカテゴリまたは複数のカテゴリを選択できます。
選択した属性は、**【選択されたカテゴリ】** リストに表示されます。
3. **【保存】** をクリックします。

Data Integration Hub ワークフローの編集

ナビゲータを使用して、Data Integration Hub ワークフローを編集します。

1. ナビゲータで **【ハブの管理】** > **【ワークフロー】** をクリックします。
【ワークフロー】 ページが表示されます。
2. ワークフローの横にある **【編集】** アイコンをクリックして、必要に応じてワークフロープロパティを更新します。
3. **【保存】** をクリックします。

Data Integration Hub ワークフローの削除

ナビゲータを使用して、Data Integration Hub ワークフローを削除します。

ワークフローを削除すると、そのワークフローの依存オブジェクトもすべて削除されます。

1. ナビゲータで **【ハブの管理】** > **【ワークフロー】** をクリックします。
【ワークフロー】 ページが表示されます。
2. ワークフローの横にある **【削除】** アイコンをクリックして、削除を確定します。

Data Integration Hub ワークフローのプロパティ

【ワークフローの作成】 ページには、次のタブがあります。

【全般】 タブ

基本的なワークフロープロパティを定義して、プロセッサタイプ、ワークフローの使用方法、およびマッピングを選択します。

【ワークフローパラメータ】 タブ

ワークフローにユーザー定義のパラメータが含まれる場合、パラメータに値を追加してカスタマイズできます。

【イベント属性】 タブ

イベント属性をワークフローに関連付けます。

【権限】 タブ

カテゴリを選択してワークフローにアクセスする権限をユーザーに与えることにより、操作コンソールでワークフローへのアクセスを制御します。選択したカテゴリの少なくとも 1 つを持つユーザーグループに属しているユーザーのみが、ワークフローに対する権限を持ちます。カテゴリを選択しない場合、すべてのユーザーがワークフローに対する権限を持ちます。

ワークフローの全般プロパティ

【ワークフローの作成】 ページの **【全般】** タブには、次のプロパティが含まれます。

ワークフロー名

ワークフロー名。名前には最大 64 文字を含めることができ、スペースおよび特殊文字を含めることができます。

説明

ワークフローの説明（オプション）。

プロセッサタイプ

プロセッサタイプを選択する際には、次のガイドラインを考慮します。

- Data Integration Hub ワークフローを Data Engineering Integration マッピングに関連付けるには、**【Data Engineering Integration マッピング】** を選択します。このマッピングは一度実行し、完了後に停止します。Data Engineering Integration マッピングの割り当て先は、ビッグデータをオンプレミスアプリケーションから Hadoop ベースのトピックにパブリッシュするカスタムバッチパブリケーションと、Hadoop ベースのトピックのビッグデータをオンプレミスアプリケーションにコンSUMするカスタムバッチサブスクリプションに設定できます。
- Data Integration Hub ワークフローを Data Engineering Integration ワークフローに関連付けるには、**【Data Engineering Integration ワークフロー】** を選択します。Data Engineering Integration ワークフローの割り当て先は、ビッグデータをオンプレミスアプリケーションから Hadoop ベースのトピックにパブリッシュするカスタムバッチパブリケーションと、Hadoop ベースのトピックのビッグデータをオンプレミスアプリケーションにコンSUMするカスタムバッチサブスクリプションに設定できます。

注: Data Engineering Integration ワークフローは、パブリケーションとサブスクリプションの使用のみをサポートします。

- Data Integration Hub ワークフローを Data Engineering Streaming マッピングに関連付けるには、**【Data Engineering Streaming Real-time Mapping】** を選択します。マッピングは継続的に実行され、ストリーミングソースから読み取りを行います。Data Engineering Streaming マッピングの割り当て先は、ビッグデータストリームをオンプレミスアプリケーションから Hadoop ベースのトピックにパブリッシュするカスタムマルチレイテンシーパブリケーションに設定できます。
- Data Integration Hub ワークフローを Data Engineering Quality マッピングに関連付けるには、**【Data Quality マッピング】** を選択します。このマッピングは一度実行し、完了後に停止します。Data Engineering Quality マッピングは、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーション、およびデータベースまたはファイルベースのトピックからオンプレミスアプリケーションにデータをコンSUMするカスタムサブスクリプションに割り当てることができます。
- Data Integration Hub ワークフローを Data Engineering Quality ワークフローに関連付けるには、**【Data Quality ワークフロー】** を選択します。このワークフローは一度実行し、完了後に停止します。Data Engineering Quality ワークフローは、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーション、およびデータベースまたは

ファイルベースのトピックからオンプレミスアプリケーションに日付をコンシュームするカスタムサブスクリプションに割り当てることができます。

注: Data Engineering Quality ワークフローは、パブリケーションおよびサブスクリプション用途のみをサポートしています。

- **[PowerCenter バッチワークフロー]** を選択すると、Data Integration Hub ワークフローが PowerCenter バッチワークフローに関連付けられます。PowerCenter バッチワークフローがファイル、データベース、または他のソースから読み取りを行います。このワークフローは一度実行し、完了後に停止します。PowerCenter バッチワークフローの割り当て先は、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーション、データベースまたはファイルベースのトピックからオンプレミスアプリケーションにデータをコンシュームするカスタムサブスクリプション、パブリケーションの前処理、サブスクリプションの後処理、PowerCenter ワークフローを起動するイベント監視ルールに設定できます。
- **[PowerCenter リアルタイムワークフロー]** を選択すると、Data Integration Hub ワークフローをリアルタイムワークフローに関連付けます。PowerCenter リアルタイムワークフローは継続的に実行され、リアルタイムソースから読み取りを行います。PowerCenter リアルタイムワークフローの割り当て先は、オンプレミスアプリケーションからデータベースまたはファイルベースのトピックにパブリッシュするカスタムパブリケーションに設定できます。リアルタイムワークフローを使用すると、リアルタイムソースからデータを読み取り、そのデータを継続的に Data Integration Hub に書き込むことができます。例えば、Web サービスプロバイダまたは Java Message Service (JMS) キューを使用してデータをパブリッシュする場合に使用します。リアルタイムワークフローによりパブリッシュする際、パブリッシュされたデータを定義済みの時間間隔で単一のパブリケーションにグループ化できます。

使用方法

Data Integration Hub ワークフローを使用するプロセッサのタイプ。次のいずれかのオプションを選択します。

- パブリケーションの前処理。パブリケーションの前処理を実行します。
- パブリケーション。パブリケーション処理を実行します。
- サブスクリプション。サブスクリプション処理を実行します。
- サブスクリプションの後処理。サブスクリプションの後処理を実行します。
- 監視ルール。パブリケーションまたはサブスクリプションイベントが、イベント監視ルールに定義されたステータスになると、ワークフローが実行されます。

マッピング

Data Integration Hub に追加するワークフローまたはマッピング。表示されるダイアログボックスは、**[プロセッサタイプ]** で選択したワークフローまたはマッピングのタイプによって異なります。

- マッピングタイプが **[Data Engineering Integration マッピング]** の場合、**[Data Engineering Integration マッピングの追加]** ダイアログボックスが表示されます。Hadoop 環境用にデータ統合サービスにデプロイされているすべての Data Engineering Integration マッピングを表示するには、**[すべて表示]** をクリックします。アプリケーション名でマッピングを検索するには、**[アプリケーション名]** テキストボックスに文字列を入力し、**[検索]** をクリックします。マッピングを選択して **[追加]** をクリックします。
- マッピングタイプが **[Data Engineering Streaming Real-time Mapping]** の場合、**[Data Engineering Streaming Real-time Mapping の追加]** ダイアログボックスが表示されます。Hadoop 環境用にデータ統合サービスにデプロイされているすべての Data Engineering Streaming マッピングを表示するには、**[すべて表示]** をクリックします。アプリケーション名でマッピングを検索するには、**[アプリケーション名]** テキストボックスに文字列を入力し、**[検索]** をクリックします。マッピングを選択して **[追加]** をクリックします。
- マッピングタイプが **[Data Engineering Quality マッピング]** の場合、**[Data Engineering Quality マッピングの追加]** ダイアログボックスが表示されます。ネイティブ環境用にデータ統合サービスにデ

プロイされているすべてのマッピングを表示するには、[すべて表示] をクリックします。アプリケーション名でマッピングを検索するには、[アプリケーション名] テキストボックスに文字列を入力し、[検索] をクリックします。マッピングを選択して [追加] をクリックします。

- マッピングタイプが PowerCenter ワークフローの場合、バッチワークフローでもリアルタイムワークフローでも、[PowerCenter ワークフローの追加] ダイアログボックスが表示されます。次のいずれかのオプションを選択します。
 - [PowerCenter リポジトリからのワークフローの選択]。PowerCenter リポジトリに保存した PowerCenter ワークフローを選択して、[追加] をクリックします。リポジトリ内のすべてのワークフローを表示するには、[すべて表示] をクリックします。名前でワークフローを検索するには、[フォルダ名] テキストボックスに文字列を入力し、[検索] をクリックします。
 - [ワークフロー定義ファイル (XML) の選択]。PowerCenter リポジトリからエクスポートした XML PowerCenter ワークフロー定義ファイルを選択して、[追加] をクリックします。ファイルを選択するには、参照ボタンをクリックし、必要なファイルの場所を参照し、ファイルを選択し、[アップロード] をクリックします。

ワークフローパラメータのプロパティ

[ワークフローの作成] ページの [ワークフローパラメータ] タブには、次のビューが含まれます。

プレビュー

ワークフローパラメータのプレビュー。

デザイナー

Forms Designer を使用すると、これらのパラメータのレイアウトおよび動作をカスタマイズできます。

注: ワークフローにセッションパラメータが含まれているときに、Data Integration Hub のオペレータがパブリケーションウィザードまたはサブスクリプションウィザードでセッションパラメータを設定しないようにする場合、Forms Designer を使用してセッションパラメータを非表示にします。

ワークフローのイベント属性のプロパティ

[ワークフローの作成] ページの [イベント属性] タブには、次のプロパティが含まれます。

使用可能な属性キー

ワークフローに割り当てることができる属性キーのリスト。

選択した属性キー

ワークフローに割り当てられた属性キーのリスト。

ワークフロー権限のプロパティ

[ワークフローの作成] ページの [権限] タブには、次のプロパティが含まれます。

使用可能なカテゴリ

ワークフローに割り当てることができるカテゴリのリスト。

選択したカテゴリ

ワークフローに割り当てられたカテゴリのリスト。

第 7 章

Data Integration Hub トランスフォーメーション

この章では、以下の項目について説明します。

- [Data Integration Hub トランスフォーメーションの概要, 74 ページ](#)
- [DX_Add_Document_To_Event トランスフォーメーション, 77 ページ](#)
- [DX_Event_Attribute トランスフォーメーション, 79 ページ](#)
- [DX_Event_Details トランスフォーメーション, 80 ページ](#)
- [DX_Generate_Temporary_File トランスフォーメーション, 81 ページ](#)
- [DX_Notification トランスフォーメーション, 82 ページ](#)
- [DX_Publication_Parameters, 84 ページ](#)
- [DX_Start_Publication トランスフォーメーション, 85 ページ](#)
- [DX_Throw_Error, 86 ページ](#)

Data Integration Hub トランスフォーメーションの概要

Data Integration Hub トランスフォーメーションは、PowerCenter において Data Integration Hub のデータを処理する関数のセットです。

Data Integration Hub のインストール時に、Data Integration Hub のデータを処理するために、PowerCenter ワークフローで使用するトランスフォーメーションのセットをインストールできます。マッピングを作成する際に、トランスフォーメーションを使用して、他の PowerCenter トランスフォーメーションを使用する方法と同じ方法で必要な関数を実行できます。

Data Integration Hub トランスフォーメーションは、Java コードを記述せずに Data Integration Hub API にアクセスするために使用するカスタムの Java トランスフォーメーションです。これらのトランスフォーメーション、構造化されていないデータのトランスフォーメーション、および他のトランスフォーメーションを使用してパブリケーションおよびサブスクリプションを処理できます。

以下の表に、Data Integration Hub トランスフォーメーションを示します。

トランスフォーメーション	説明
DX_Add_Document_To_Event	イベントにドキュメントを添付します。
DX_Event_Attribute	イベント属性の値を取得または設定します。
DX_Event_Details	イベントのプロパティの値を取得または設定します。
DX_Generate_Temporary_File	Data Integration Hub のドキュメントストア内のファイルに対してファイル名を生成します。
DX_Notification	処理が成功したことを Data Integration Hub サーバーに通知します。
DX_Publication_Parameters	パブリケーションインスタンス ID 用およびパブリケーション日用のプレースホルダを Data Integration Hub パブリケーションリポジトリに追加します。
DX_Start_Publication	パブリケーション処理を開始します。
DX_Throw_Error	トランスフォーメーションが失敗した場合に、イベントステータスをエラーに設定します。

トランスフォーメーションのインストールと登録

Data Integration Hub トランスフォーメーションを PowerCenter に追加するには、Data Integration Hub インストーラを使用します。

以下のコンポーネントをインストールします。

- Data Integration Hub PowerCenter サーバープラグイン。このプラグインを登録するには Administrator ツールを使用します。
- Data Integration Hub PowerCenter Client プラグイン。インストール処理中にインストーラがプラグインを登録します。

プラグインのインストールおよび登録後、トランスフォーメーションが PowerCenter Designer で有効になっていることを確認します。プラグインのインストールおよび設定についての詳細は、『*Data Integration Hub インストール&環境設定ガイド*』を参照します。

トランスフォーメーションの構成

Data Integration Hub トランスフォーメーションをマッピングに追加した後、ドキュメントの処理要件に基づいてこのトランスフォーメーションを編集および設定します。

Data Integration Hub トランスフォーメーションを編集する際、以下のタブでトランスフォーメーションコンポーネントを設定できます。

- [トランスフォーメーション] タブ。トランスフォーメーションの名前を変更し、説明を追加します。
- ポート。ポートを追加、編集、または削除します。ポートを入力または出力ポートとして指定することができます。
- [プロパティ] タブ。モジュールと関数の識別子、トランザクションプロパティ、実行時位置などのトランスフォーメーションのプロパティを設定します。このタブに表示されるプロパティは、他の PowerCenter Custom トランスフォーメーションのプロパティと同じです。[Properties (プロパティ)] タブの詳細については、『*PowerCenter トランスフォーメーションガイド*』を参照します。

- [DX Properties (DX プロパティ)] タブ。トランスフォーメーションでのポートのデフォルト値を設定します。他の Data Integration Hub トランスフォーメーションプロパティのデフォルト値も設定できます。

トランスフォーメーションエラーの処理

特定のトランスフォーメーションポートを使用して、ワークフローの実行中に発生するエラーの対処方法を定義できます。

各 Data Integration Hub トランスフォーメーションは、次のポートを使用してエラーに対処します。

- DXErrorCode。トランスフォーメーションが失敗すると、そのトランスフォーメーションは DXErrorCode を 0 よりも大きい値に設定します。
- DXErrorMessage。トランスフォーメーションが失敗すると、そのトランスフォーメーションはその失敗を説明するエラーメッセージを DXErrorMessage ポートに送信します。

トランスフォーメーションでエラーが発生すると、そのトランスフォーメーションは PowerCenter セッションログにエラーを書き込みます。このエラーログには、例外クラス、説明、原因、およびスタックトレースが含まれます。ロギングレベルは、PowerCenter の設定に基づきます。ログには、エラーに関連する最大 1K のドキュメントが含まれます。

トランスフォーメーションが失敗したときに、イベントステータスをエラーに設定するオプションが true に設定された場合にも、トランスフォーメーションはイベントのステータスをエラーに設定します。

エラー処理ポートを入力ポートとして設定して、入力エラーが発生した場合にトランスフォーメーションが実行しないようにできます。

Data Integration Hub トランスフォーメーションのルールとガイドライン

PowerCenter において Data Integration Hub トランスフォーメーションを使用する際には、ルールとガイドラインに従い、パフォーマンスを最適化し、エラーを防止します。

以下に、Data Integration Hub トランスフォーメーションを使用するためのルールとガイドラインを一覧して説明します。

- Data Integration Hub トランスフォーメーションは、PowerCenter Custom トランスフォーメーションに基づき、他のカスタムのトランスフォーメーションと同じ設定オプションを提供します。これらは、他の PowerCenter トランスフォーメーションを使用するように使用できます。
- Data Integration Hub トランスフォーメーションは接続されたトランスフォーメーションです。接続されたトランスフォーメーションは、他のトランスフォーメーションに対してデータを受け渡します。
- Data Integration Hub トランスフォーメーションは、パッシブトランスフォーメーションです。
- トランスフォーメーションポートに、対応する Data Integration Hub プロパティがある場合、ポートの値が実行時のプロパティの値よりも優先されます。セッションの実行時、ポートの値が NULL ではない場合、PowerCenter 統合サービスはそのポートの値を処理に使用します。ポートの値が NULL の場合、PowerCenter 統合サービスは Data Integration Hub のプロパティの値を処理に使用します。
- ポート名は大文字と小文字は区別されず、プレフィックスも区別されません。DXEventID、dxEVENTid、および eventid はすべて同じポートとして扱われます。
- Data Integration Hub トランスフォーメーションを使用する PowerCenter ワークフローを実行する際には、PowerCenter はイベントのステータスおよびタイプのリストを取得してトランスフォーメーションで使用するために、Data Integration Hub リポジトリへの接続を試行します。エラーメッセージは、接続が失敗したことを示します。接続が失敗した場合、PowerCenter はクライアントのプラグイン設定ファイル

からイベントタイプの値を取得します。接続エラーを解決するには、dxplugin.ini ファイルの接続セクションが以下の設定を含んでいることを確認します。

```
[DX_REPOSITORY]
; ODBC connection string to the DX repository
; CONNECTION_STRING=DRIVER={DataDirect 7.0 Oracle Wire Protocol};
; UID=%1;PWD=%2;Host=localhost;Port=1521;SN=orcl
; CUSTOM_CONNECTION_STRING
; ODBC DSN to the DX repository
; DSN_NAME=dxOdbcResourceName
USER_NAME=DX
USER_PASSWD=DX
EVENT_TYPE_NAME=SELECT event_type_name FROM dx_event_type ORDER BY 1
EVENT_STATUS_NAME=SELECT event_status_name FROM dx_event_status ORDER BY 1
```

DX_Add_Document_To_Event トランスフォーメーション

DX_Add_Document_To_Event トランスフォーメーションは、イベントにドキュメントを添付します。

このトランスフォーメーションを使用して、以前のトランスフォーメーションで作成したドキュメントを追加できます。例えば、このトランスフォーメーションを使用してイベントにログファイルを添付できます。

入力ポート

DX_Add_Document_To_Event トランスフォーメーションの入力ポートは、**[Ports (ポート)]** タブで設定します。

以下の表に、DX_Add_Document_To_Event の入力ポートを示します。

ポート	タイプ	説明
DXDescription	文字列	イベントに添付するドキュメントの説明。
DXMimeType	文字列	イベントに添付するドキュメントの MIME タイプ。
DXEncoding	文字列	イベントに添付するドキュメントの文字エンコード。これは、文字列のバイト配列への変換に使用する文字セットです。
DXTemporaryFilePath	文字列	オプション。DX_Generate_Temporary_File トランスフォーメーションにより生成されるパスおよびファイル名で、ワークフローはここに新しいファイルを格納します。DX_Add_Document_To_Event トランスフォーメーションは、このファイルをドキュメントストアに新しいドキュメント参照として格納し、ファイル参照をイベントに添付します。 このポートを設定するか、DXData および DXDataByReference ポートを設定できます。このポートと、DXData および DXDataByReference ポートが設定されない場合、トランスフォーメーションは空のドキュメントを作成し、それをイベントに追加します。

入出力ポート

DX_Add_Document_To_Event トランスフォーメーションの入出力ポートは、**[Ports (ポート)]** タブで設定します。

以下の表で、DX_Add_Document_To_Event の入出力ポートについて説明します。

ポート	タイプ	説明
DxEventId	文字列	必須。ドキュメントを添付するイベントの識別子。
DXDataByReference	文字列	DXData ポートにドキュメントデータが含まれるか、ドキュメント参照が含まれるかを示します。値が true の場合、DXData ポートにはドキュメント参照が含まれます。値が NULL または false の場合、DXData ポートにはドキュメントデータが含まれます。
DXDocumentId	文字列	イベントに添付するドキュメントの識別子。
DXErrorMessage	文字列	トランスフォーメーションにより生成されるエラーメッセージ。
DXErrorCode	文字列	トランスフォーメーションにより生成されるエラーコード。トランスフォーメーションが失敗すると、DXErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Add_Document_To_Event トランスフォーメーションのプロパティは、**[DX Properties (DX プロパティ)]** タブで設定します。

以下の表で、DX_Add_Document_To_Event トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
説明	イベントに添付するドキュメントの説明。
Document Role (ドキュメントロール)	ドキュメントがイベントに添付されたときのドキュメントのロール。次のいずれかのロールを選択します。 <ul style="list-style-type: none">- SOURCE。ドキュメントはソースドキュメントとしてイベントに添付されます。イベントを操作コンソールに表示する場合、[Input (入力)] セクションに添付されたドキュメントを表示できます。- TARGET。ドキュメントはターゲットドキュメントとしてイベントに添付されます。イベントを操作コンソールに表示する場合、[Output (出力)] セクションに添付されたドキュメントを表示できます。- LOG。ドキュメントはログドキュメントとしてイベントに添付されます。イベントを操作コンソールに表示する場合、[Logging Information (ロギング情報)] セクションに添付されたドキュメントを表示できます。
Generate an error in case a failure occurs in this transformation (このトランスフォーメーションで失敗が発生した場合エラーを生成する)	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。 デフォルトで選択されています。

DX_Event_Attribute トランスフォーメーション

DX_Event_Attribute は、イベント属性値を取得または設定します。

このトランスフォーメーションを使用するには、最初に操作コンソールでイベント属性を設定します。次に、ポートをトランスフォーメーションに追加します。各ポートは、取得または設定するイベント属性を示します。

ポート名は、イベント属性名と一致する必要があります。ただし、属性名には大文字小文字の区別はありません。例えば、event_att1 属性と EVENT_ATT1 は同じ属性として扱われます。

イベント名はプレフィックスを区別します。例えば、DX_event_att1 属性と event_att1 属性は同じ属性として扱われません。

ポートに対して値を設定した場合、トランスフォーメーションはこの値にイベント属性を設定します。イベント属性の値を取得するには、空のポートを作成します。

入出力ポート

DX_Event_Attribute トランスフォーメーションの入出力ポートは、**[Ports (ポート)]** タブで設定します。

以下の表で、DX_Event_Attribute の入出力ポートについて説明します。

ポート	タイプ	説明
DxEventId	文字列	必須。取得または更新する属性に関連付けられたイベントの識別子。
DXErrorMessage	文字列	トランスフォーメーションにより生成されるエラーメッセージ。
DXErrorCode	文字列	トランスフォーメーションにより生成されるエラーコード。トランスフォーメーションが失敗すると、DXErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Event_Attribute トランスフォーメーションのプロパティは、**[DX Properties (DX プロパティ)]** タブで設定します。

以下の表で、DX_Event_Attribute トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
Event attribute name (イベント属性名)	更新するイベント属性の名前。プロパティは、操作コンソールで設定します。値が NULL の場合、トランスフォーメーションはエラーを生成します。 注: オプションのプロパティ。かわりに【<Attribute> (<属性>)] ポートを使用することを推奨します。このプロパティは、後方互換性のために保持されます。
Generate an error in case a failure occurs in this transformation (このトランスフォーメーションで失敗が発生した場合エラーを生成する)	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。デフォルトで選択されています。

DX_Event_Details トランスフォーメーション

DX_Event_Details トランスフォーメーションは、イベントの複数のプロパティを取得または設定します。

トランスフォーメーションポートは、イベントのプロパティを表します。このトランスフォーメーションは、入力ポートがトランスフォーメーションに渡す値に基づいてイベントのプロパティを設定します。このトランスフォーメーションはプロパティの値を出力ポートを介してイベントに渡します。

入出力ポート

DX_Event_Details トランスフォーメーションの入出力ポートは、**[Ports (ポート)]** タブで設定します。

以下の表で、DX_Event_Details の入出力ポートについて説明します。

ポート	タイプ	説明
DXStatusName	文字列	イベントのステータスです。このステータスは、イベントが処理中に渡すステージを示します。
DxEventId	文字列	必須。取得または更新するプロパティに関連付けられたイベントの識別子。
DXTypeName	文字列	イベントのタイプ。
DXSubject	文字列	イベントの件名。
DXErrorMessage	文字列	トランスフォーメーションにより生成されるエラーメッセージ。
DXErrorCode	文字列	トランスフォーメーションにより生成されるエラーコード。トランスフォーメーションが失敗すると、DXErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Event_Details トランスフォーメーションのプロパティは **[DX Properties (DX プロパティ)]** タブで設定します。

以下の表で、DX_Event_Details トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
イベントのステータス	イベントのステータスです。
イベントのタイプ	イベントのタイプ。
Generate an error in case a failure occurs in this transformation (このトランスフォーメーションで失敗が発生した場合エラーを生成する)	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。 デフォルトで選択されています。

DX_Generate_Temporary_File トランスフォーメーション

DX_Generate_Temporary_File トランスフォーメーションは、Data Integration Hub ドキュメントストアに格納するドキュメントのパスおよびファイル名を生成します。

このトランスフォーメーションを使用して、ドキュメントストア内の一時ディレクトリに一意のファイル名を生成します。例えば、構造化されていない Data transformation を使用して一時ファイルにデータを書き込む場合、DX_Generate_Temporary_File トランスフォーメーションを使用してドキュメントストアのファイルのファイル名を生成できます。次にファイルにファイル名およびパスを割り当てます。

一時ディレクトリにファイルを作成した後、それを他のトランスフォーメーションで使用できます。例えば、DX_Add_Document_To_Event トランスフォーメーションを使用してファイルをイベントに添付できます。PowerCenter でこのファイルを一時ディレクトリから、イベントに追加するすべてのドキュメントを含むディレクトリにコピーします。

入出力ポート

DX_Generate_Temporary_File トランスフォーメーションの入出力ポートは **【Ports (ポート)】** タブで設定します。

以下の表で、DX_Generate_Temporary_File の入出力ポートについて説明します。

ポート	タイプ	説明
DxEventId	文字列	必須。生成されたファイルに関連付けるイベントの識別子。
DxTemporaryFilePath	文字列	一時ファイルのパスおよびファイル名。
DxErrorMessage	文字列	トランスフォーメーションにより生成されるエラーメッセージ。
DxErrorCode	文字列	トランスフォーメーションにより生成されるエラーコード。トランスフォーメーションが失敗すると、DxErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Generate_Temporary_File トランスフォーメーションのプロパティは **DX プロパティ** タブで設定します。

以下の表で、DX_Generate_Temporary_File トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
生成されたファイルパスで一時フォルダを使用しない	Data Integration Hub ドキュメントストアの一時ディレクトリ内にファイルパスを生成するか、または標準のドキュメントディレクトリにファイルパスを直接生成するかを示します。このオプションは、PowerCenter が参照によって渡すドキュメントに対するパフォーマンスを向上させ、一時ディレクトリが提供する追加のバックアップをなくすために選択します。 デフォルトでは選択されていません。
トランスフォーメーションが失敗した場合にイベントステータスをエラーに設定	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。 デフォルトで選択されています。

DX_Notification トランスフォーメーション

DX_Notification トランスフォーメーションでイベントのステータスを設定し、必要に応じて Data Integration Hub サーバーに処理が終了したことを通知します。

パブリケーションワークフローでこのトランスフォーメーションを使用する際には、Data Integration Hub サーバーにパブリケーションの処理が完了し、発行済みデータを使用する準備ができたことを通知できます。次に Data Integration Hub サーバーでサブスクリプション用の子イベントを作成し、データをパブリッシュする際に使用するサブスクリプション用のサブスクリプションバッチワークフローをトリガします。スケジュール済みサブスクリプションイベントは、スケジュール済みサブスクライバがデータのコンSUMEを開始するまで保留されます。トランスフォーメーションが送信する通知はオプションです。そのため、パブリケーション処理がエラーで終了した場合、サブスクリプションをトリガせずに、このトランスフォーメーションを使用してイベントステータスをエラーに変更できます。

サブスクリプションの後処理ワークフローでトランスフォーメーションを使用する場合は、サブスクリプションの後処理が完了したことを Data Integration Hub サーバーに通知し、後処理ステータスをサーバーに送信する必要があります。その後 Data Integration Hub サーバーは、後処理ワークフローを取り消したサブスクリプションの子イベントを後処理のステータスで作成します。

入出力ポート

DX_Notification トランスフォーメーションの入出力ポートは **【Ports (ポート)】** タブで設定します。

以下の表で、DX_Notification の入出力ポートについて説明します。

ポート	タイプ	説明
DXEventId	文字列	必須。ステータスを設定するイベントの識別子。\$\$DxeventId パラメータから値を取得できます。
DXStatusName	文字列	設定するステータスの名前。値が NULL の場合、このトランスフォーメーションでトランスフォーメーションプロパティに基づいた値を設定します。
DXNotifySubscribers	文字列	必須。Data Integration Hub サーバーに通知を送信するかどうかを示します。可能な値: <ul style="list-style-type: none">- True。トランスフォーメーションで Data Integration Hub サーバーに通知を送信します。Data Integration Hub サーバーでサブスクリプションバッチワークフローをトリガし、各サブスクリプションに対して子イベントを作成します。- False。Data Integration Hub サーバーへの通知なしで、トランスフォーメーションでイベントステータスを設定します。
DXErrorMessage	文字列	トランスフォーメーションにより生成されるエラーメッセージ。
DXErrorCode	文字列	トランスフォーメーションにより生成されるエラーコード。トランスフォーメーションが失敗すると、DXErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Notification トランスフォーメーションのプロパティは **【DX Properties (DX プロパティ)】** タブで設定します。

以下の表で、DX_Notification トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
イベントのステータス	作成するイベントのステータス。 デフォルトは Complete です。
Notify the hub that the publication is ready for subscribers (パブリケーションがサブスクライバの準備ができたことを Hub に通知する)	ワークフローが正常に終了したことについて Data Integration Hub サーバーに通知を送信するかどうかを示します。選択された場合、Data Integration Hub サーバーは各サブスクリプションに対して子イベントを作成し、パブリッシュされた際にデータを使用するように定義したサブスクリプション用のサブスクリプションバッチワークフローをトリガします。 デフォルトで選択されています。
Generate an error in case a failure occurs in this transformation (このトランスフォーメーションで失敗が発生した場合エラーを生成する)	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。 デフォルトで選択されています。

DX_Publication_Parameters

DX_Publication_Parameters トランスフォーメーションは、パブリケーションインスタンス ID 用およびパブリケーション日用のプレースホルダを Data Integration Hub パブリケーションリポジトリに追加します。

このトランスフォーメーションを使用して、リアルタイムワークフローでのカスタムマッピングを使用するパブリケーション用のプレースホルダを追加します。

入力ポート

DX_Publication_Parameters トランスフォーメーションの入力ポートは **【Ports (ポート)】** タブで設定します。

以下の表で、DX_Publication_Parameters の入力ポートについて説明します。

ポート	タイプ	説明
DXPublicationName	文字列	パブリケーションの名前。この名前は、トランスフォーメーションを実行するワークフローでパブリッシュするパブリケーションの名前と同一である必要があります。 名前は大文字小文字の区別はせず、最大 64 文字の英数字とアンダースコアを含むことができます。

出力ポート

DX_Publication_Parameters トランスフォーメーションの出力ポートは **【Ports (ポート)】** タブで設定します。

以下の表で、DX_Publication_Parameters の出力ポートについて説明します。

ポート	説明
DXPublicationInstanceId	DIH__PUBLICATION_INSTANCE_ID のプレースホルダを Data Integration Hub のパブリケーションリポジトリに追加します。 DIH__PUBLICATION_INSTANCE_ID で、アプリケーションがパブリッシュしたデータセットを識別します。 パブリケーションリアルタイムワークフローを作成する際、このポートをターゲットの DIH__PUBLICATION_INSTANCE_ID カラムに接続します。
DXPublicationInstanceDate	DIH__PUBLICATION_INSTANCE_DATE のプレースホルダを Data Integration Hub のパブリケーションリポジトリに追加します。 DIH__PUBLICATION_INSTANCE_DATE は、アプリケーションでデータセットの発行を開始した日時です。 パブリケーションリアルタイムワークフローを作成する際、このポートをターゲットの DIH__PUBLICATION_INSTANCE_DATE カラムに接続します。

DX_Start_Publication トランスフォーメーション

DX_Start_Publication トランスフォーメーションで、指定されたパブリケーションのパブリケーションワークフローをトリガするよう Data Integration Hub の PowerCenter 統合サービスに指示します。

このトランスフォーメーションを使用して、PowerCenter からパブリケーションを開始します。例えば、パブリケーションの処理が開始する前にデータを準備するパブリケーションの前処理ワークフローを実行する場合、DX_Start_Publication トランスフォーメーションをワークフローの末尾に追加します。PowerCenter は Data Integration Hub サーバーにリクエストを送信し、Data Integration Hub サーバーがパブリケーションリポジトリにデータを公開するパブリケーションワークフローをトリガします。

注: DX_Start_Publication トランスフォーメーションは、ワークフロー内で 2 回以上呼び出さないでください。2 回以上呼び出した場合、Data Integration Hub はパブリケーションを複数回開始します。

PowerCenter パラメータファイルを使用する場合、このファイルの内容を DXParameterFileContents ポートに渡し、パラメータをパブリケーションワークフローで使用できます。

入出力ポート

DX_Start_Publication トランスフォーメーションの入出力ポートは【ポート】タブで設定します。

注: ワークフローに渡す特定のパラメータでポートを追加できます。パラメータの型は文字列である必要があります。トランスフォーメーションにパススルー入力ポートを追加できません。

以下の表で、DX_Start_Publication の入出力ポートについて説明します。

ポート	タイプ	説明
DXPublicationName	文字列	オプション。開始するパブリケーションの名前。この名前は操作コンソールでのパブリケーションの名前と一致する必要があります。 注: DXPublicationName ポートを定義しない場合は、DXEventId ポートを定義する必要があります。
DXEventId	文字列	パブリケーションの前処理およびパブリケーションと関連付けられたイベントの識別子。\$\$DXEventId パラメータから値を取得できます。パブリケーションの前処理で単一のパブリケーションを開始する場合は、使用することを推奨します。イベント ID を使用すると、Data Integration Hub ではパブリケーションの前処理ワークフローとパブリケーションワークフローで同じイベントが使用され、イベントステータスが適宜変更されます。 パブリケーションの前処理で複数のパブリケーションを開始する場合は、DX_Start_Publication トランスフォーメーション内でイベント ID を使用しないでください。PowerCenter トランスフォーメーションのイベントの詳細を使用してイベントステータスを変更できます。 注: DXEventId ポートを定義しない場合は、DXPublicationName ポートを定義する必要があります。
DXParameterFileContents	文字列	PowerCenter パラメータファイルからのパラメータの名前および値。ポートの値にファイルパスを使用できません。
DXErrorMessage	文字列	トランスフォーメーションによって生成されるエラーメッセージ。
DXErrorCode	文字列	トランスフォーメーションによって生成されるエラーコード。トランスフォーメーションが失敗すると、DXErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Start_Publication トランスフォーメーションのプロパティは **[DX Properties (DX プロパティ)]** タブで設定します。

以下の表で、DX_Start_Publication トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
Generate an error in case a failure occurs in this transformation (このトランスフォーメーションで失敗が発生した場合エラーを生成する)	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。 デフォルトで選択されています。

DX_Throw_Error

ワークフロー実行の際にエラーが発生した場合、DX_Throw_Error トランスフォーメーションでエラーを生成します。

このトランスフォーメーションを使用して次のアクションを実行します。

- 関連付けられたイベントのステータスをエラーに設定します。
- DXDescription ポートの値からエラーメッセージを作成します。
- 関連付けられたイベントにエラーメッセージを添付します。
- セッションログにエラーを送信します。

入力ポート

DX_Throw_Error トランスフォーメーションの入力ポートは **[Ports (ポート)]** タブで設定します。

以下の表で、DX_Throw_Error の入力ポートについて説明します。

ポート	タイプ	説明
DXDescription	文字列	セッションログに送信するエラーの説明。このエラーは、イベントに添付されるログドキュメントの説明でもあります。
DXMessageType	文字列	オプション。エラーイベントのタイプ。イベントと関連付けられた英数字の値。任意の値が有効です。
DXMIMEType	文字列	イベントに添付するドキュメントの MIME タイプ。

入出力ポート

DX_Throw_Error トランスフォーメーションの入出力ポートは **【Ports (ポート)】** タブで設定します。

以下の表で、DX_Throw_Error の入出力ポートについて説明します。

ポート	タイプ	説明
DXEventId	文字列	必須。エラーに関連付けられているイベントの識別子。
DXData	文字列 バイナリ テキスト	イベントに添付するログドキュメント。このポートには、ドキュメントのデータまたはドキュメントへのファイルパスを含めることができます。このパラメータの値が NULL の場合、トランスフォーメーションは空のドキュメントを作成し、そのドキュメントをイベントに追加します。 テキストデータのあるドキュメントを添付するには、ポートのデータ型を文字列またはテキストに設定します。バイナリデータのあるドキュメントを添付するには、ポートのデータ型をバイナリに変更します。
DXDataByReference	文字列	DXData ポートにドキュメントデータが含まれるか、ドキュメント参照が含まれるかを示します。可能な値: - True。DXData ポートにはドキュメント参照が含まれます。 - False。DXData ポートにはドキュメントデータが含まれます。 - Null。DXData ポートにはドキュメントデータが含まれます。
DXErrorMessage	文字列	トランスフォーメーションにより生成されるエラーメッセージ。
DXErrorCode	文字列	トランスフォーメーションにより生成されるエラーコード。トランスフォーメーションが失敗すると、DXErrorCode ポートの値は 0 よりも大きくなります。

Data Integration Hub プロパティ

DX_Throw_Error トランスフォーメーションのプロパティは **【DX Properties (DX プロパティ)】** タブで設定します。

以下の表で、DX_Throw_Error トランスフォーメーションの Data Integration Hub プロパティについて説明します。

プロパティ	説明
Error log document description (エラーログドキュメントの説明。)	Description of the error log document that the transformation attaches to the event. (トランスフォーメーションでイベントに添付するエラーログドキュメントの説明。)
Message type (メッセージタイプ)	イベントと関連付けられた英数字の値。任意の値を入力できます。
Generate an error in case a failure occurs in this transformation (このトランスフォーメーションで失敗が発生した場合エラーを生成する)	トランスフォーメーションでエラーが生成されたときに、イベントのステータスをエラーに設定するかどうかを示します。デフォルトで選択されています。

第 8 章

オペレーショナルデータストアダッシュボードとレポート

この章では、以下の項目について説明します。

- [オペレーショナルデータストアダッシュボードとレポートの概要, 88 ページ](#)
- [重要業績評価指標 \(KPI\) , 89 ページ](#)
- [Logi Info Studio のダッシュボードとレポートの構造, 90 ページ](#)
- [Logi Info Studio のカスタムダッシュボードとレポート, 94 ページ](#)

オペレーショナルデータストアダッシュボードとレポートの概要

ダッシュボードとは、Data Integration Hub が処理する情報に関するレポートが含まれるパネルの集合です。ダッシュボードを使用して、特定のパートナーのイベント数や特定のアカウントのエラー率など、Data Integration Hub イベントおよびドキュメント処理に関するパーソナライズされたサマリ情報を視覚的に表示します。

ダッシュボードを Data Integration Hub で使用可能にするには、2 つの方法があります。Data Integration Hub をインストールすると、オペレーショナルデータストアを使用するダッシュボードを有効にすることができます。または、デフォルトで使用可能なメタデータリポジトリを使用するダッシュボードを使用できます。このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。

オペレーショナルデータストアを使用するダッシュボードのほとんどのレポートは、イベント、エラー、トランザクション、値など、Data Integration Hub がオペレーショナルデータストアから取得する重要業績評価指標 (KPI) に基づいています。オペレーショナルデータストアは、レポート用途に特化された集計情報を含むリポジトリです。ダッシュボードはオペレーショナルデータストアから集計されたイベントを取得し、パネルに情報を表示します。デフォルトの KPI に加えて、カスタム KPI としてオペレーショナルデータストアにロードするイベント属性を操作コンソールで選択できます。

Logi Analytics を使用して、デフォルトの KPI またはカスタム KPI に基づいてカスタムレポートを作成します。Logi Analytics で、カスタムダッシュボードをセットアップし、表示する情報に基づいて KPI が含まれるカスタムダッシュボードパネルを作成します。Data Integration Hub でカスタムダッシュボードをデプロイし、作成したカスタムダッシュボードパネルを取得するように接続プロパティを変更します。

注: Logi Analytics の操作に関する全般的な情報については、Logi Analytics ドキュメントを参照してください。

参照リンクでダウンロード可能なソフトウェアは、Informatica LLC ではなく、サードパーティに属しています。ダウンロード元リンクは、誤りがある可能性や、削除または変更される可能性があります。Informatica LLC は、そのようなリンクやソフトウェアに対し、明示的、黙示的にかかわらず、商品性、特定目的への適合性、権原、および非侵害についてのいかなる保証についても、その責任を一切負わないものとします。

Data Integration Hub 管理者はメインの Data Integration Hub とともにオペレーショナルデータストアをインストールし、Data Integration Hub システムプロパティを変更して、ユーザーが操作コンソールにログインしたときにダッシュボードを表示するかどうかや、[エラー率ゲージ] パネルのエラー率の範囲など、特定のダッシュボードの動作を決定します。オペレータはダッシュボードをパーソナライズして、表示するパネルを変更し、確認したい情報に基づいてフィルタを適用できます。

Data Integration Hub 管理者はダッシュボードとレポートコンポーネントをインストールしたら、PowerCenter Workflow Manager を使用してオペレーショナルデータストアのイベントローダーを設定します。オペレーショナルデータストアのイベントローダーは、指定されたパラメータに基づいて Data Integration Hub リポジトリから KPI を収集し、集計されたイベントをオペレーショナルデータストアにロードする PowerCenter ワークフローです。オペレーショナルデータストアのイベントローダーで、Data Integration Hub 管理者はエラーが発生した場合の各イベントロードプロセスの再試行回数やイベントロードプロセス間で待機する分数などのパラメータを定義します。

重要業績評価指標（KPI）

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。重要業績評価指標（KPI）を使用して、Data Integration Hub が生成するイベントに関する情報を収集します。オペレーショナルデータストアのイベントローダーは、メインの Data Integration Hub リポジトリから KPI を収集し、集計された KPI をダッシュボードに転送します。

オペレーショナルデータストアを使用するダッシュボードには、Data Integration Hub がオペレーショナルデータストアから取得するデフォルトの KPI に基づいてほとんどのレポートが表示されます。デフォルトの KPI に加えて、カスタム KPI としてオペレーショナルデータストアにロードするイベント属性を選択できます。選択したカスタム KPI に基づいてカスタムダッシュボードパネルを作成するには、Logi Info Studio を使用します。カスタムダッシュボードパネルでは数値イベント属性を使用します。

オペレーショナルデータストアからのレポートに加えて、Data Integration Hub はランタイム Data Integration Hub から未解決のエラーイベント情報を直接取得します。パフォーマンスを維持するには、未解決のエラーイベントパネルをカスタマイズしたり、ランタイム Data Integration Hub リポジトリからのイベント情報に基づいてカスタムダッシュボードパネルを作成したりしないでください。

サービスレベル契約（SLA）の要件を追跡する場合、デフォルトの KPI またはカスタム KPI に基づいてルールを作成し、ダッシュボードの [SLA 違反] パネルで違反を表示できます。

デフォルトの KPI

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。オペレーショナルデータストアはレポートでデフォルトの KPI を使用します。オペレーショナルデータストアのイベントローダーは、KPI を収集して KPI をオペレーショナルデータストアにロードします。

操作コンソールのユーザーは、集計されたイベント情報をダッシュボードパネルのレポートとして表示できます。デフォルトの KPI に基づいて Logi Info Studio でカスタムダッシュボードパネルを作成できます。カスタムダッシュボードパネルを作成するときに、合計、カウント、最小、最大、平均などの測定結果をデフォルトの KPI に適用できます。

以下の表に、オペレーショナルデータストアを使用するダッシュボードで表示されるデフォルトの KPI について説明します。

KPI	説明
メッセージ処理時間	イベントが最初に最終状態に達するまでに要する分数。
イベント数	Data Integration Hub が処理するイベント数。
エラーイベントの数	最終状態に到達したエラーイベントの数。

Logi Info Studio のダッシュボードとレポートの構造

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。ダッシュボードには、オペレーショナルデータストアから取得した集計データに基づいてイベント情報が表示されます。Logi Info Studio を使用し、デフォルトのダッシュボードパネルを表示して、デフォルトの KPI およびオペレーショナルデータストアから取得したカスタム KPI に基づいてカスタムダッシュボードパネルを作成します。

Logi Info Studio のデフォルトのダッシュボードには以下のグローバル要素が含まれています。

- ダッシュボードが各レポートに適用できるフィルタ。
- ダッシュボードパネルの外観を制御するテーマ変数。
- オペレーショナルデータストアで実行する共通クエリの SQL 定数。
- JavaScript は、Data Integration Hub がレポートを処理してダッシュボードパネルを表示するために使用するファイルをサポートしています。

オペレーショナルデータストアダッシュボードのフィルタ

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。ダッシュボードは、Logi Info Studio で生成した各レポートにフィルタを使用します。フィルタは、操作コンソールのダッシュボードページのグローバルフィルタに対応します。

次の表に、デフォルトのダッシュボードのフィルタの説明を示します。

フィルタ	説明
@Request.idBegin~	yyyy-mm-dd 形式の date/time フィルタの開始日。
@Request.itBegin~	hh:mm 形式の date/time フィルタの開始時刻。
@Request.idEnd~	yyyy-mm-dd 形式の date/time フィルタの終了日。
@Request.itEnd~	hh:mm 形式の date/time フィルタの終了時刻。
@Request.iPartnerId~	選択したパートナーのパートナーデータベース識別子。
@Request.iAccountId~	選択したアカウントのアカウントデータベース識別子。

フィルタ	説明
@Request.eventTypeId~	選択したイベントタイプのイベントタイプデータベース識別子。
@Request.eventStatusId~	選択したイベントステータスのイベントステータスデータベース識別子。

オペレーショナルデータストアダッシュボード SQL クエリ要素

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。オペレーショナルデータストアを使用するデフォルトのダッシュボードの各レポートは、オペレーショナルデータストアおよびクエリパラメータに対して共通クエリの SQL 定数を使用します。これらの SQL 定数は、オペレーショナルデータストアにロードする KPI に基づいてカスタムダッシュボードパネルを作成するときに再利用できます。

注: SQL サーバーでは、同じ定数に末尾に `_SQLSERVER` を追加して使用します。例えば、`DX_ODS_FILTER_ACCOUNT` は `DX_ODS_FILTER_ACCOUNT_SQLSERVER` になります。

注: SQL クエリを使用して Logi Info Studio でランタイム Data Integration Hub リポジトリに対するレポートを作成することはできません。

次の表に、デフォルトのダッシュボード SQL 定数の説明を示します。

SQL 定数名	説明
<code>DX_ODS_FILTER_ACCOUNT</code>	<p>関連するアカウントに基づいてデータベース行をフィルタリングする。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - <code>requestAccountId</code>
<code>DX_ODS_FILTER_ERROR_STATE</code>	<p>エラーイベントのみを表示するようにデータベース行をフィルタリングする。</p>
<code>DX_ODS_FILTER_EVENT_STATUS</code>	<p>イベントステータスに基づいてデータベース行をフィルタリングする。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - <code>requestEventStatusId</code>
<code>DX_ODS_FILTER_EVENT_STATUS_STATE</code>	<p>イベントの状態に基づいてデータベース行をフィルタリングする。以下の値が使用できます。</p> <ul style="list-style-type: none"> - 1. 最終状態にあるエラーイベントを表示する。 - 2. 最終状態にあるエラー以外のイベントを表示する。 - -1. 最終状態にあるすべてのイベントを表示する。 <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - <code>requestEventStatusState</code>
<code>DX_ODS_FILTER_EVENT_TYPE</code>	<p>イベントタイプに基づいてデータベース行をフィルタリングする。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - <code>requestEventTypeId</code>
<code>DX_ODS_FILTER_PARTNER</code>	<p>関連パートナーに基づいてデータベース行をフィルタリングする。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - <code>requestPartnerId</code>

SQL 定数名	説明
DX_ODS_FILTER_TIME_ORACLE	<p>選択したタイムフレームに基づいてデータベース行をフィルタリングする。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - requestDateBegin - requestTimeBegin - requestDateEnd - requestTimeEnd
DX_ODS_USER_RESTRICTION_BY_ACCESS_ID	<p>関連したユーザーアクセス識別子に基づいてデータベース行をフィルタリングする。</p> <p>注: ユーザーアクセス制御を最大化するには、カスタムダッシュボードパネルで DX_ODS_USER_RESTRICTION_BY_ACCESS_ID、 DX_ODS_USER_RESTRICTION_BY_EVENT_TYPE、および DX_ODS_USER_RESTRICTION_BY_PARTNER を使用します。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - requestUserId
DX_ODS_USER_RESTRICTION_BY_EVENT_TYPE	<p>ユーザーが表示特権を持つイベントタイプのみを表示するようにデータベース行をフィルタリングする。</p> <p>注: ユーザーアクセス制御を最大化するには、カスタムダッシュボードパネルで DX_ODS_USER_RESTRICTION_BY_ACCESS_ID、 DX_ODS_USER_RESTRICTION_BY_EVENT_TYPE、および DX_ODS_USER_RESTRICTION_BY_PARTNER を使用します。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - requestUserId
DX_ODS_USER_RESTRICTION_BY_PARTNER	<p>ユーザーが表示特権を持つ関連パートナーのみを表示するようにデータベース行をフィルタリングする。</p> <p>注: ユーザーアクセス制御を最大化するには、カスタムダッシュボードパネルで DX_ODS_USER_RESTRICTION_BY_ACCESS_ID、 DX_ODS_USER_RESTRICTION_BY_EVENT_TYPE、および DX_ODS_USER_RESTRICTION_BY_PARTNER を使用します。</p> <p>次のクエリパラメータが含まれます。</p> <ul style="list-style-type: none"> - requestUserId

オペレーショナルデータストアダッシュボード JavaScript サポートファイル

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。操作コンソールでダッシュボードを表示すると、Data Integration Hub は JavaScript ファイルと jQuery ファイルを使用してイベント情報を処理します。

ファイルは以下の場所にあります。

<DIHInstallationDir>\DataIntegrationHub\tomcat\webapps\dih-dashboard_SupportFiles

次の表に、デフォルトのダッシュボードが使用する JavaScript ファイルと jQuery ファイルの説明を示します。

ファイル	説明
dx.js	Data Integration Hub サーバーが LogiXML エンジンを使用してレポートを処理するときに使用する関数。
dxbrowser.js	操作コンソールでダッシュボードを表示したときにブラウザクライアントが使用する関数。
jquery-*.*	パートナーやアカウントなど、グローバルフィルタセクションのドロップダウンリストの動的な入力を可能にする jQuery ファイルの集合。

オペレーショナルデータストアダッシュボードテーマ変更子

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合のみ適用されます。テーマ変更子は、ダッシュボードパネルの外観を制御します。Data Integration Hub は、作成したすべてのカスタムダッシュボードパネルにテーマ変更子のスタイルおよび外観の設定を適用します。

テーマ変更子ファイルは、以下の場所にあります。

<DIHInstallationDir>\DataIntegrationHub\tomcat\webapps\dih-dashboard_Themes\DIHDashboard\Informatica

次の表に、デフォルトのダッシュボードのテーマ変更子コンポーネントの説明を示します。

ファイル	説明
Theme.css	ダッシュボードパネルのグラフィックおよびテキストのフォーマットを決定するカスケードスタイルシート。
ThemeModifier.xml	操作コンソールにログオンしたときに Data Integration Hub が使用するダッシュボードパネルの動作およびシングルサインオンメカニズムのグローバル設定。

カスタムダッシュボードの設定

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Data Integration Hub 用のカスタムダッシュボードを作成する場合は、LogiXML の _Settings.lgx ファイルで一部のパラメータを変更します。

以下の表に、変更するプロパティを示します。

プロパティ	説明
DX_CONSOLE_URL	次の形式の操作コンソールの場所。 http://<DIH host>:<DIH port>/DIH-console
DASHBOARD_SAVEFOLDER	各ダッシュボードユーザーの個人向けレイアウト設定が格納される場所。 デフォルト: @Function.AppPhysicalPath~\dx\saved_dashboards
DX_ODS_ORACLE_CONNECTION	オペレーショナルデータストアの接続文字列。
DX_ORACLE_CONNECTION	メインの Data Integration Hub リポジトリの接続文字列。

プロパティ	説明
LagonFailPage	ユーザーが操作コンソールからログアウトした後に表示する URL。 デフォルトの URL は Data Integration Hub ログアウトページ: <code>http://<DIH host>:<DIH port>/DIH-console/logout.jsp</code>
AuthenticationClientAddresses	LogiXML サーバーから認証トークンを要求できるサーバーの IP アドレス (セミコロン区切り)。各サーバーについて IP4 アドレスと IP6 アドレスを設定します。
caption	カスタムダッシュボードの説明。
COOKIE_PATH	カスタムダッシュボードのクッキーファイルへのパス。Cookie Path プロパティの値と一致する必要があります。
WEBAPP_NAME	カスタムダッシュボードが含まれるフォルダの名前。 デフォルト: <code>dxdashboard</code>
Redirect Error URL	ダッシュボードでエラーが発生した場合に表示するページのリンク。

Logi Info Studio のカスタムダッシュボードとレポート

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Logi Info Studio を使用して、デフォルトの KPI またはカスタム KPI に基づいてカスタムダッシュボードパネルを作成します。

Logi Info Studio で、レポートを作成してオペレーショナルデータストアから数値イベント属性を取得する SQL クエリを追加します。次に、カスタムレポートを表示して Data Integration Hub でカスタムダッシュボードをデプロイするようにダッシュボードパネルを設定します。

以下のレポートに基づいてカスタムダッシュボードパネルを作成することはできません。

- パートナー別の未解決のエラーイベント (eventsbypartnerErrorRepository.lgx)
- アカウント別の未解決のエラーイベント (eventsbyaccountErrorRepository.lgx)
- イベントタイプ別の未解決のエラーイベント (eventsbyeventstatusErrorRepository.lgx)
- イベントステータス別の未解決のエラーイベント (eventsbyeventstatusErrorRepository.lgx)
- タスク (dxtasks.lgx)

Logi Info Studio でカスタムレポートを作成する場合、SQL 定数をデフォルトのダッシュボードからコピーして、ダッシュボードパネルに表示する KPI に基づいて変更することができます。

SQL 定数は LogiXML 定数トークンを付けて使用できます。Logi Info Studio はカスタムレポートを処理するときに、定数トークンが SQL 定数を拡張します。例えば、次の SQL クエリを使用します。

```
select facts.ACCOUNT_ID, sum(EVENT_COUNT) as SUM_COUNT from DX_ODS_EVENT_FACTS facts
where @Constant.DX_ODS_FILTER_TIME_ORACLE~
```

LogiXML 定数トークンはこの SQL クエリを以下の SQL クエリに拡張します。

```
select facts.ACCOUNT_ID, sum(EVENT_COUNT) as SUM_COUNT from DX_ODS_EVENT_FACTS facts
where to_timestamp(:requestDateBegin || :requestTimeBegin, 'YYYY-MM-DD HH24:MI:SS') <= facts.TIMESLICE
and facts.TIMESLICE <= to_timestamp(:requestDateEnd || :requestTimeEnd, 'YYYY-MM-DD HH24:MI:SS')
```

カスタムレポート用のパラメータを新規作成する場合、各パラメータに一意の名前を指定する必要があります。名前が一意でないと、ダッシュボードは同じ名前の既存のパラメータを上書きするか、パラメータ値を誤って処理します。データレイヤリンクを作成する場合、すべてのレポートで各データレイヤリンクに一意の名前を指定する必要があります。名前が一意でないと、日付やステータスが間違っているなど、作成したダッシュボードに不正なデータが表示される場合があります。

Logi Info Studio のインストールと設定

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Data Integration Hub 用にダッシュボードパネルを強化および作成するには、Logi Info Studio をインストールします。

Logi Info Studio をインストールする前に、Data Integration Hub のダッシュボードとレポートのコンポーネントが Data Integration Hub マシンにインストールされていて、システムが Logi Info Studio の最小要件を満たしていることを確認してください。Logi Info Studio のシステム要件の詳細については、Logi Analytics ドキュメントを参照してください。

1. Logi Info Studio インストーラで、カスタムインストールを選択して **Studio** コンポーネントのみのインストールを選択します。
2. Informatica の発送部署に問い合わせて、Logi Info Studio ライセンスファイルを受け取ります。
3. Logi Info Studio ライセンスファイルを<LogiXMLInstallationDir>\LogiStudio フォルダにコピーします。

カスタムダッシュボードの設定

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Logi Info Studio でカスタムダッシュボードをアプリケーションとして作成し、デフォルトのダッシュボードファイルを Logi Info Studio にインポートします。

1. Logi Info Studio で、Java アプリケーションを作成します。
2. **【新しいアプリケーションの準備】** ウィザードで、ダッシュボードアプリケーションの名前と場所を設定します。
 - アプリケーション名に、カスタムダッシュボードの名前を入力します。
 - デプロイメントフォルダの場所に、次のパスを入力します。
<DIHInstallationDir>\DataIntegrationHub\tomcat\webapps
3. **【新しいアプリケーションの準備】** ウィザードの残りの画面はスキップします。
4. <DIHInstallationDir>\DataIntegrationHub\tomcat\webapps\DIH-dashboard フォルダの内容を、新規作成した <DIHInstallationDir>\DataIntegrationHub\tomcat\webapps\<custom_dashboard> フォルダにコピーします。
5. LogiXML ライセンスファイルを <DIHInstallationDir>\DataIntegrationHub\tomcat\webapps\DIH-dashboard フォルダから <DIHInstallationDir>\DataIntegrationHub\tomcat\webapps\<custom_dashboard> フォルダにコピーします。
6. LogiXML の _Settings.lgx ファイルで、カスタムダッシュボードの名前および場所に基づいてプロパティを変更します。
7. Data Integration Hub サーバーで、dx.dashboard.url システムプロパティの値を次の形式でカスタムダッシュボードの URL の値に置換します: http://<host>:<port>/<custom_dashboard>

カスタムダッシュボードでのレポートの作成アプリケーション

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Logi Info Studio でカスタムダッシュボードをセットアップしたら、レポートを作成してカスタムダッシュボードパネルで使用するよう設定します。必要に応じて既存のレポートをコピーして変更できます。

1. Logi Info Studio でレポート定義を作成します。
2. ユーザーに提供するレポートフィルタを選択します。
3. オペレーショナルデータストアで実行する SQL クエリの SQL パラメータを定義します。各パラメータの名前とタイプを定義する必要があります。

カスタムダッシュボードへのダッシュボードパネルの追加

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Logi Info Studio でレポートを作成したら、カスタムダッシュボードパネルとその要素をメインのダッシュボードレポートファイルに追加します。

1. Logi Info Studio でカスタムダッシュボードレポートファイルを開きます。
2. dxdashboard 要素で、キャプションと一意の識別子を付けて新しい **Panel** 要素を追加します。
3. 追加した Panel 要素で、**Panel Content** 要素を追加します。
4. 追加した Panel Content 要素で、一意の識別子を付けて **SubReport** 要素を追加し、**Frame Border** プロパティを **false** に設定します。
5. 追加した SubReport 要素で、一意の識別子を付けて **Target.Report** 要素を追加し、新しいレポートの名前を **Report Definition File** プロパティに入力します。
6. **Link Parameters** 要素を既存の SubReport 要素から新しい SubReport 要素にコピーします。
7. **dxdashboard** レポートファイルを保存します。

B2B Data Exchange でのカスタムダッシュボードのデプロイとテスト

このセクションは、オペレーショナルデータストアを使用したダッシュボードを使用する場合にのみ適用されます。Logi Info Studio でカスタムダッシュボードを変更または作成してから、カスタムダッシュボードアプリケーションフォルダの内容を Data Integration Hub にコピーします。次に、カスタムダッシュボードが正しく表示されるかテストします。

1. カスタムダッシュボードアプリケーションフォルダをそのまま<DIHInstallationDir>\DataIntegrationHub\tomcat\webapps にコピーします。
2. _Setting.lgx ファイルをバックアップして次の設定を変更します。
 - **Redirect Error URL** プロパティの値を削除して、LogiXML エラーレポートにアクセスできるようにします。
 - **Debugger Style** プロパティの値を **DebuggerLinks** に設定し、リンクを LogiXML 進行状況レポートに追加します。ダッシュボードパネルレイアウトの表示は異なる場合があります。
3. 操作コンソールにログオンして、新しいダッシュボードパネルが開き、正しい値が入力されているかテストします。
4. カスタムダッシュボードをテストして、_Setting.lgx ファイルでプロパティを復元します。

第 9 章

Forms Designer

この章では、以下の項目について説明します。

- [Forms Designer の概要, 97 ページ](#)
- [Forms Designer のユーザーインターフェース, 97 ページ](#)
- [要素のプロパティ, 99 ページ](#)
- [グループプロパティ, 100 ページ](#)

Forms Designer の概要

Forms Designer は、動的なインターフェースを備えた操作コンソールページにおいて、開発者および管理者がユーザーインターフェースおよびフィールドの動作をカスタマイズできるツールです。Forms Designer を使用して、ワークフローパラメータを表示するページをカスタマイズできます。

Forms Designer では、[Designer (デザイナー)] タブを使用してフィールドの動作および外観をカスタマイズします。例えば、タブ内でワークフローパラメータの大きいリストを整列したり、別のパラメータに値がある場合に、特定のパラメータのみが表示されるようにフィールド間の依存性を作成したりできます。また、フィールドを必須または読み取り専用としてマークしたり、フィールドのタイプを決定することもできます。例えば、フィールドをテキストフィールドまたは数値フィールドとして定義できます。カスタマイズしたレイアウトは [Preview (プレビュー)] タブでプレビューできます。

Forms Designer のユーザーインターフェース

[Designer (デザイナー)] タブには、カスタマイズする要素が表示されます。**[Actions (アクション)]** メニューを使用すると、タブ、カラム、ヘッダ、またはディスクロジャなどのグループを追加できます。また、要素をグループにドラッグアンドドロップし、フィールドを最大 4 レベルまでネストできます。

[Designer (デザイナー)] タブには、ラベルやタイプなどの要素についての情報が表示されます。この情報は、読み取り専用モードで表示されます。

以下の表で、[Designer (デザイナ)] タブに表示される要素プロパティについて説明します。

カラム	説明
ID	要素の一意の識別子。例えば、PowerCenter で定義されたワークフローパラメータの名前です。
ラベル	フォームに表示される要素のラベル。ラベルは要素の ID とは異なる場合があります。
タイプ	テキストや日付などの要素のタイプ。
表示	フォームでの要素の表示。例えば、要素は入力フィールド、ラジオボタン、またはドロップダウンリストなどです。
必須	要素が必須であるかどうかを示します。
読み取り専用	要素が読み取り専用であるかどうかを示します。
依存対象	パラメータが依存する要素またはグループの名前。

Forms Designer のアクション

[Designer (デザイナ)] タブでは、タブ、カラム、ヘッダ、またはディスクロジャを追加できます。要素を選択すると、要素のプロパティを編集できます。

次の表で、[Designer (デザイナ)] タブで実行できるアクションについて説明します。

アクション	説明
編集	要素プロパティの編集に使用するダイアログボックスを開きます。 要素の選択時に使用可能です。
タブの作成	タブを作成します。要素をタブにドラッグアンドドロップし、タブを要素リストで上下に移動してタブの順序を決定できます。最上部のタブはデフォルトで表示されます。 タブは最大 2 レベルまでネストできます。
カラムの作成	カラムを作成します。2 つ以上のカラムを追加すると、カラムにドラッグアンドドロップした要素はフォーム上で並んで表示されます。 カラムは最大 2 レベルまでネストできます。
ヘッダの作成	ヘッダを作成します。ヘッダは、要素をフォームで非表示にせずにグループ化するために使用するタイトルの付いたセパレータです。 ヘッダは最大 2 レベルまでネストできます。
ディスクロジャの作成	ディスクロジャを作成します。ディスクロジャはタイトルと展開矢印の付いたヘッダです。矢印をクリックすると、要素グループが表示されます。デフォルトでは、ディスクロジャはフォームで最小化されて表示されます。 ディスクロジャは最大 2 レベルまでネストできます。

アクション	説明
グループの削除	タブ、カラム、ヘッダ、またはディスクロジャを削除します。グループを削除してもフォームから要素は削除されません。 グループの選択時に使用可能です。
グループへ移動	要素を 1 つのタブ、カラム、ヘッダ、またはディスクロジャグループから別のグループへ移動します。 要素の選択時に使用可能です。

要素のプロパティ

【Details（詳細）】ダイアログボックスで要素を編集する際、ラベル、タイプ、およびデフォルト値などの要素プロパティを変更できます。

要素を選択すると、ダイアログボックスが表示されます。グループを選択すると、そのグループのラベル、タイプ、およびフィールドの依存性のみを編集できます。

以下の表に、要素プロパティを示します。

プロパティ	説明
ラベル	必須。フォームに表示する要素の名前。ラベルは ID とは異なる場合があります。
説明	パラメータのテキストによる説明。説明は、フォーム内で要素上にマウスポインタを重ねるとツールチップに表示されます。
必須	ユーザーは、フォームの要素に値を入力することを求められます。
読み込み専用	要素を読み取り専用として定義し、ユーザーは要素の値を変更できません。
非表示	フォームから要素を非表示にします。オペレータが編集できないワークフローパラメータにはこのチェックボックスを選択します。
タイプ	要素の値のデータ型。次のいずれかのオプションを選択します。 <ul style="list-style-type: none"> - ブール。このオプションを選択した場合、【Check box（チェックボックス）】の表示はデフォルトで選択されます。 - 日付。このオプションを選択した場合、【Date picker（日付選択画面）】の表示はデフォルトで選択されます。 - 数値。このオプションを選択した場合、【Entry field（入力フィールド）】の表示はデフォルトで選択されます。 - テキスト
表示	ユーザーが要素の値に対して入力する入力のタイプ。次のいずれかのオプションを選択します。 <ul style="list-style-type: none"> - 入力フィールド - ドロップダウンリスト - ラジオボタン - リスト - チェックボックス

プロパティ	説明
デフォルト値	フォームが表示されるときに表示される値。
値	パラメータに対する有効な値のリスト。リストまたはドロップダウンリスト表示を選択した場合、1つ以上の値を入力します。値リストでは大文字と小文字は区別されません。 注: 外部ソースからリストの値を読み込んだ場合、リストの値は変更しないでください。外部ソースからの値は、リストへの手動の変更を上書きします。
最小	要素に対する最小の有効値。文字列要素では、このプロパティに入力した値により最小文字数が決まります。数値要素では、このプロパティに入力した値により最小数値が決まります。 文字列または数値の要素タイプの選択時に使用可能です。
最大	要素に対する最大の有効値。文字列要素では、このプロパティに入力した値により最大文字数が決まります。数値要素では、このプロパティに入力した値により最大数値が決まります。 文字列または数値の要素タイプの選択時に使用可能です。
フィールドの依存性	要素をフォーム内の別の要素またはグループにリンクするかどうかを決定する条件のセット。例えば、ユーザーが別の要素に値を入力した後のみ、要素を有効にするように選択できます。

グループプロパティ

【Details（詳細）】 ダイアログボックスでタブ、カラム、ヘッダ、ディスクロジャを作成または編集する際、ラベル、タイプ、またはフィールドの依存性のプロパティを変更できます。

以下の表に、グループプロパティを示します。

プロパティ	説明
ラベル	必須。フォームに表示するグループの名前。
タイプ	必須。グループのタイプ。次のいずれかのオプションを選択します。 <ul style="list-style-type: none"> - タブ - カラム - ヘッダ - ディスクロジャ
フィールドの依存性	グループをフォーム内の別の要素またはグループにリンクするかどうかを決定する条件のセット。例えば、ユーザーが別の要素に特定の値を入力した後のみ、タブを有効にするように選択できます。

第 10 章

Data Integration Hub のパブリケーションおよびサブスクリプションの API

この章では、以下の項目について説明します。

- [パブリケーションおよびサブスクリプション API の概要, 101 ページ](#)
- [Run Publication Subscription REST API, 102 ページ](#)
- [Run Publication Subscription コマンドライン API, 104 ページ](#)
- [Change Publication Subscription Mode REST API, 107 ページ](#)
- [Reprocess Event REST API, 109 ページ](#)
- [イベントステータス API, 110 ページ](#)
- [Change Event Status API, 112 ページ](#)

パブリケーションおよびサブスクリプション API の概要

Data Integration Hub のパブリケーションおよびサブスクリプションの API を使用してパブリケーションおよびサブスクリプションを実行し、パブリケーションおよびサブスクリプションを有効化および無効化したり、パブリケーションおよびサブスクリプションイベントを再処理したり、パブリケーションおよびサブスクリプションイベントのステータスをクエリしたり、Data Integration Hub カタログからデータを抽出したりします。

次のパブリケーションおよびサブスクリプションの API を使用できます。

- Run Publication Subscription REST API。無効化されたパブリケーションおよびサブスクリプションを含むパブリケーションまたはサブスクリプションを開始し、Data Integration Hub が生成するパブリケーションまたはサブスクリプションイベントのイベント ID を返します。
注: Run Publication Subscription REST API を使用して、Data Integration タスクをトリガするパブリケーションおよびサブスクリプションを使用してデータをパブリッシュおよびデータにサブスクライブできません。データドリブンパブリケーションおよびサブスクリプションを使用したデータのパブリケーションやサブスクリプションには、この API を使用できません。
- Run Publication Subscription コマンドライン API パブリケーションまたはサブスクリプションをコマンドラインユーティリティから開始し、Data Integration Hub が生成するパブリケーションまたはサブス

リプションイベントのイベント ID を返します。コマンドライン API では、無効化されたパブリケーションおよびサブスクリプションは実行できません。コマンドライン API をリモートサーバーから呼び出すことはできません。

ヒント: Run Publication Subscription REST API を使用して、リモートサーバーからパブリケーションとサブスクリプションの実行を開始します。

- Change Publication Subscription Mode REST API。パブリケーションまたはサブスクリプションのモードを変更します。つまり、無効化されたパブリケーションまたはサブスクリプションを有効化し、有効化されたパブリケーションまたはサブスクリプションを無効化します。
- Reprocess Event REST API。無効化されたパブリケーションおよびサブスクリプションのイベントを含む、パブリケーションまたはサブスクリプションのイベントを再処理します。
- Event Status REST API。パブリケーションまたはサブスクリプションのイベントのステータスを返します。

Run Publication Subscription REST API

Data Integration Hub Run Publication Subscription REST API を使用して、特定のパブリケーションまたは特定のサブスクリプションを実行します。パブリケーションまたはサブスクリプションは、そのモードに関係なく実行できます。つまり、有効化されたパブリケーションおよびサブスクリプションと、無効化されたパブリケーションおよびサブスクリプションの両方を実行できます。

注: Run Publication Subscription REST API を使用して、Data Integration タスクをトリガするパブリケーションおよびサブスクリプションを使用してデータをパブリッシュおよびデータにサブスクライブできます。データドリブンパブリケーションおよびサブスクリプションを使用したデータのパブリケーションやサブスクリプションには、この API を使用できません。

Run Publication Subscription API は、実行したアクションの応答コードを返します。パブリケーションまたはサブスクリプションの実行に成功した場合、この API は Data Integration Hub が生成したパブリケーションまたはサブスクリプションイベントのイベント ID を返します。Data Integration Hub イベントステータス API を実行すると、パブリケーションまたはサブスクリプションのステータスをクエリできます。

API からパブリケーションまたはサブスクリプションを実行するには、実行するパブリケーションまたはサブスクリプションに対して権限と特権を持った Data Integration Hub ユーザーを使用する必要があります。詳細については、『*Data Integration Hub 管理者ガイド*』を参照してください。

サンプルファイル

<DIH_InstallationDir>\samples\rest_api_samples フォルダに、サンプルのクライアントアプリケーションと、アプリケーションについて説明した readme ファイルが含まれています。

Run Publication Subscription REST API の要求

Data Integration Hub は、パブリケーションやサブスクリプションを実行するために、さまざまな REST URL を使用します。

パブリケーションを実行するには、次の REST URL を使用します。

`http://<DIH_console_host_port>/dih-console/api/v1/publication/start`

例:

`http://localhost:18080/dih-console/api/v1/publication/start`

サブスクリプションを実行するには、次の REST URL を使用します。

`http://<DIH_console_host_port>/dih-console/api/v1/subscription/start`

以下に例を示します。

http://localhost:18080/dih-console/api/v1/subscription/start

注: Start subscription REST API では、同じサブスクリプションの同時トリガはサポートされていません。数秒後にサブスクリプションをトリガすることをお勧めします。

パブリケーション実行の要求構文

パブリケーションを実行するには、次の要求構文を使用します。

```
{
  "publicationName": "<publicationName>",
  "parameters": [
    {"name": "param1", "value": "value"},
    {"name": "param2", "value": "value"}
  ],
  "paramFileContent": "<content of parameter file>",
  "runPreprocess": "<true/false>",
  "runDisabled": "<true/false>"
}
```

次のリストに、要求の要素を示します。

- publicationName。実行するパブリケーションの名前。
- parameters。オプションの要求パラメータ。
- paramFileContent。オプションのパラメータファイルのコンテンツ。
- runPreprocess。パブリケーションの実行前に、パブリケーションの前処理を実行するかどうか。
- runDisabled。ステータスが無効になっているパブリケーションを実行するかどうか。

例:

```
{
  "publicationName": "daily_sales",
  "parameters": [
    {"name": "SRC_SAP", "value": "sapbw.etl-tools.info"},
    {"name": "TGT_ORA", "value": "oracle_sap_staging"}
  ],
  "paramFileContent": "[DWH_PROJECT.WF:wkf_daily_loading] \n $$platform=rh7",
  "runPreprocess": "false",
  "runDisabled": "true"
}
```

サブスクリプション実行の要求構文

サブスクリプションを実行するには、次の要求構文を使用します。

```
{
  "subscriptionName": "<subscriptionName>",
  "parameters": [
    {"name": "param1", "value": "value"},
    {"name": "param2", "value": "value"}
  ],
  "paramFileContent": "<content of parameter file>",
  "runDisabled": "<true/false>"
}
```

次のリストに、要求の要素を示します。

- subscriptionName。実行するサブスクリプションの名前。
- parameters。オプションの要求パラメータ。
- paramFileContent。オプションのパラメータファイルのコンテンツ。
- runDisabled。ステータスが無効になっているサブスクリプションを実行するかどうか。

例:

```
{
  "subscriptionName": "daily_report",
  "parameters": [
    {"name": "SRC_ORA", "value": "oracle_sap_staging"},
    {"name": "SuccessEmail", "value": "customer@informatica.com"}
  ],
  "paramFileContent": "[DWH_PROJECT.WF:wkf_daily_loading] \n $$platform=rh7",
  "runDisabled": "true"
}
```

Run Publication Subscription REST API アクションの応答

Data Integration Hub Run Publication Subscription REST API を使用してパブリケーションまたはサブスクリプションの実行を開始すると、Data Integration Hub により、REST API の応答で実行したアクションの応答コードが返されます。

パブリケーションまたはサブスクリプションを REST API から実行すると、次の応答コードのいずれかが返されます。

- SUCCESS。Data Integration Hub はパブリケーションまたはサブスクリプションを正常にトリガしました。このステータスメッセージには、Data Integration Hub が生成するパブリケーションまたはサブスクリプションイベントのイベント ID が含まれます。
- FAILED。Data Integration Hub はパブリケーションまたはサブスクリプションをトリガできませんでした。応答には失敗した理由が示されます。例: サブスクリプションでコンシュームできるパブリケーションがないため、Data Integration Hub はサブスクリプションを実行しませんでした。

Run Publication Subscription コマンドライン API

コマンドラインユーティリティから特定のパブリケーションおよび特定のサブスクリプションの実行を開始するには、Data Integration Hub Run Publication Subscription コマンドライン API を使用します。この API は Windows および UNIX オペレーティングシステムで使用できます。

Run Publication Subscription コマンドライン API は、Data Integration Hub サーバーから実行できます。

ヒント: Run Publication Subscription REST API を使用して、リモートサーバーからパブリケーションとサブスクリプションの実行を開始します。

Run Publication Subscription API は、実行したアクションの応答コードを返します。パブリケーションまたはサブスクリプションの実行に成功した場合、この API は Data Integration Hub が生成したパブリケーションまたはサブスクリプションイベントのイベント ID を返します。Data Integration Hub イベントステータス API を実行すると、パブリケーションまたはサブスクリプションのステータスをクエリできます。

API からパブリケーションまたはサブスクリプションを実行するには、実行するパブリケーションまたはサブスクリプションに対して権限と特権を持った Data Integration Hub ユーザーを使用する必要があります。詳細については、『*Data Integration Hub 管理者ガイド*』を参照してください。

Data Integration Hub Run Publication Subscription コマンドライン API のコマンド構文

Data Integration Hub Run Publication Subscription コマンドライン API では、次の構文を使用します。

```
runsubscriptionpublication
<-c|--command> publication|subscription
<-n|--name> name
```



```
<-u|--user> user
<-p|--password> password
[-r|--preprocess]
[-f|--file "<parameters file path>"]
[-v|--parameter "<key>=<value>"...]
[--server "<hostname:port>"]
```

注: パブリケーションまたはサブスクリプションを--file および--parameter の両オプションにより実行すると、Data Integration Hub は、次の順序でパラメータを並べ替えます。

1. --file で指定するパラメータファイルのワークフロー固有のセクションで定義されるパラメータ。
2. --parameter で定義されるパラメータ。
3. --file で指定するパラメータファイルの [グローバル] セクションで定義されるパラメータ。

コマンドライン API は次の場所にあります。<DIHInstallationDir>/dx-tools

次の表に、Data Integration Hub Run Publication Subscription コマンドライン API のオプションおよび引数を示します。

オプション	引数	説明
-c --command	コマンド	必須。パブリケーションを実行するため、またはサブスクリプションを実行するためのコマンド。次のいずれかのコマンドを入力します。 - パブリケーション。Data Integration Hub に、--name オプションで定義するパブリケーションのパブリケーションワークフローをトリガするよう指示します。 - サブスクリプション。Data Integration Hub に、--name オプションで定義するサブスクリプションのサブスクリプションワークフローをトリガするよう指示します。
-n --name	名前	必須。実行するパブリケーションまたは実行するサブスクリプションの名前。Data Integration Hub により、このユーザーに対し、パブリケーションまたはサブスクリプションを実行する権限および特権が確認されます。
-u --user	ユーザー名	オプション。runsubscriptionpublication コマンドを実行するデータの管理権限を持つ操作コンソールユーザーアカウントのユーザー名。このコマンドでサブスクリプションまたはパブリケーションを実行するには、ユーザーアカウントにサブスクリプションの実行またはパブリケーションの実行権限が必要です。 Informatica ドメイン認証または Informatica ドメインに設定した Kerberos 認証を使用している場合、ユーザー名では@記号で区切った Informatica セキュリティドメインを指定する必要があります。以下に例を示します。 Administrator@SecurityDomain

オプション	引数	説明
-U	環境変数	<p>オプション。ユーザー名を含む環境変数。</p> <p>runsubscriptionpublication コマンドを実行するデータの管理権限を持つ操作コンソールユーザーアカウントのユーザー名。このコマンドでサブスクリプションまたはパブリケーションを実行するには、ユーザーアカウントにサブスクリプションの実行またはパブリケーションの実行権限が必要です。</p> <p>Informatica ドメイン認証または Informatica ドメインに設定した Kerberos 認証を使用している場合、ユーザー名では@記号で区切った Informatica セキュリティドメインを指定する必要があります。以下に例を示します。</p> <p>Administrator@SecurityDomain</p> <p>注: 1 つ以上のユーザー名オプション (-u または-U) を指定する必要があります。</p>
-p --password	password	<p>オプション。runsubscriptionpublication コマンドを実行する操作コンソールユーザーのパスワード。</p>
-P	環境変数	<p>オプション。パスワードを含む環境変数。</p> <p>runsubscriptionpublication コマンドを実行する操作コンソールユーザーのパスワード。パスワードは暗号化される必要があります。暗号化値に dxencrypt を使用します。</p> <p>注: 1 つ以上のパスワードオプション (-p または-P) を指定してください。</p>
-r --preprocess	-	<p>publication コマンドでは省略可能です。前処理が定義されているパブリケーションの場合、このコマンドにより前処理が実行されます。</p>
-f --file	parameters_file_path	<p>オプション。ワークフローパラメータファイルへのパス。</p> <p>Data Integration Hub により、ワークフローパラメータファイルのコンテンツとしてこのファイルのコンテンツが使用されます。</p>

オプション	引数	説明
-v --parameter	<key>=<value>	<p>オプション。カスタムワークフローパラメータとしてのキーと値のペアの渡し。</p> <ul style="list-style-type: none"> - API がパブリケーションを実行している場合、Data Integration Hub はこの値をパブリケーションワークフローに渡します。 - --preprocess オプションを使用して API がパブリケーション前処理を実行している場合、Data Integration Hub はこのパラメータを前処理ワークフローに渡します。 - API がサブスクリプションを実行している場合、Data Integration Hub はこの値をサブスクリプションワークフローに渡します。 <p>複数のパラメータを入力することができます。以下に例を示します。</p> <p>-c subscription -v "employee_ID=148" -v "department=sales"</p> <p>注: API が実行するエンティティのワークフローには、単数または複数のこのパラメータの定義が必要です。Data Integration Hub は、定義されていないパラメータを PowerCenter に渡さないため、エラーは表示されません。</p>
--server	hostname:port	<p>オプション。Data Integration Hub サーバーのホスト名およびポート番号。値を入力しない場合、Run Publication Subscription API は、デフォルトのポート 18095 経由で localhost サーバーに接続します。</p> <p>この値は引用符で囲む必要があります。以下に例を示します。</p> <p>-c subscription --server "localhost:18096"</p>

Run Publication Subscription コマンドライン API 通知

Data Integration Hub コマンドライン API を使用してパブリケーションまたはサブスクリプションの実行を開始すると、Data Integration Hub はコマンドに成功したかどうかを通知します。

コマンドに成功した場合、通知には、Data Integration Hub が生成するパブリケーションまたはサブスクリプションイベントのイベント ID が含まれます。

コマンドに失敗した場合、通知にはエラーメッセージとエラーの原因が含まれます。失敗の原因は次のいずれかです。

- パブリケーションまたはサブスクリプションが無効。
- パブリケーションが進行中。リレーショナルデータベースパブリケーションにのみ該当します。パブリケーションが現在実行中であるため、Data Integration Hub はパブリケーションを実行できません。
- 使用するパブリケーションがない。サブスクリプションにのみ該当します。サブスクリプションで使用できるパブリケーションがないため、Data Integration Hub はサブスクリプションを実行しませんでした。

Change Publication Subscription Mode REST API

Data Integration Hub Change Publication Subscription Mode REST API を使用して、パブリケーションまたはサブスクリプションのモードを変更します。モード変更を行うと、無効化されたパブリケーションまたはサブスクリプションは有効になり、有効化されたパブリケーションまたはサブスクリプションは無効になります。

パブリケーションのモードを変更するには、次の REST URL を使用します。

`http://<DIH_console_host_port>/dih-console/api/v1/publication/changemode`

例:

`http://localhost:18080/dih-console/api/v1/publication/changemode`

サブスクリプションのモードを変更するには、次の REST URL を使用します。

`http://<DIH_console_host_port>/dih-console/api/v1/subscription/changemode`

以下に例を示します。

`http://localhost:18080/dih-console/api/v1/subscription/changemode`

パブリケーションのモード変更の要求構文

パブリケーションのモードを変更するには、次の要求構文を使用します。

```
{
  "publicationName": "<publicationName>",
  "mode": "<enable/disable>"
}
```

例:

```
{
  "publicationName": "daily_sales",
  "mode": "enable"
}
```

サブスクリプションのモード変更の要求構文

サブスクリプションのモードを変更するには、次の要求構文を使用します。

```
{
  "subscriptionName": "<subscriptionName>",
  "mode": "<enable/paused/disable>"
}
```

例:

```
{
  "subscriptionName": "daily_reports",
  "mode": "disable"
}
```

サンプルファイル

<DIH_InstallationDir>\samples\rest_api_samples フォルダに、サンプルのクライアントアプリケーションと、アプリケーションについて説明した readme ファイルが含まれています。

Change Publication Subscription Mode REST API アクションの応答

Data Integration Hub REST API を使用してパブリケーションまたはサブスクリプションのモードを変更すると、Data Integration Hub は、REST API の応答で、実行したアクションの応答コードを返します。

パブリケーションまたはサブスクリプションのモードを REST API から変更すると、次の応答コードのいずれかが返されます。

- Data Integration Hub がパブリケーションまたはサブスクリプションのモードの変更に成功した場合は、API から SUCCESS の応答が返されます。
- Data Integration Hub がパブリケーションまたはサブスクリプションのモードの変更に失敗した場合は、応答に失敗の理由が返されます。例えば、操作を実行するための十分な特権がない場合などです。

Reprocess Event REST API

Data Integration Hub Reprocess Event REST API を使用すると、無効化されたパブリケーションおよびサブスクリプションを含む、パブリケーションおよびサブスクリプションのイベントを再処理できます。

パブリケーションまたはサブスクリプションのイベントを再処理するには、次の REST URL を使用します。

`http://<DIH_console_host_port>/dih-console/api/v1/event/reprocess`

例:

`http://localhost:18080/dih-console/api/v1/event/reprocess`

次の構文を使用して、イベントを再処理します。

```
{
    "eventId" : "<eventId>"
}
```

例:

```
{
    "eventId" : "40558"
}
```

サンプルファイル

<DIH_InstallationDir>\samples\rest_api_samples フォルダに、サンプルのクライアントアプリケーションと、アプリケーションについて説明した readme ファイルが含まれています。

Reprocess Event REST API アクションの応答

Data Integration Hub Reprocess Event REST API を使用してパブリケーションまたはサブスクリプションイベントを再処理すると、Data Integration Hub は、REST API の応答で実行したアクションの応答コードを返します。

応答には次の情報が含まれています。

プロパティ	説明
responseCode	アクションのレスポンスは次のとおりです。 - 0: 成功 - 0 より大きい任意の値: エラー
reprocessEventId	Data Integration Hub が既存のイベントを再処理するときに、パブリケーション用またはサブスクリプション用に生成する新しいイベント ID。
message	エラーメッセージ。応答コードが 0（成功）の場合、API はメッセージ null を返します。

イベントステータス API

Data Integration Hub Run Publication Subscription API を使用してパブリケーションまたはサブスクリプションの実行を開始して成功した場合、Data Integration Hub は生成したパブリケーションまたはサブスクリプションイベントのイベント ID を返します。

Data Integration Hub でイベント ID が返される方法は、パブリケーションまたはサブスクリプションの実行に使用されている API によって異なります。

- REST API を実行した場合、Data Integration Hub は REST API の応答でイベント ID を返します。
- コマンドライン API を実行した場合、Data Integration Hub はコマンドライン通知でイベント ID を返します。

Data Integration Hub イベントステータス REST API を使用すると、イベント ID に基づいてパブリケーションまたはサブスクリプションイベントのステータスをクエリできます。パブリケーションまたはサブスクリプションのプロセスが実行中であるかどうか、およびプロセス完了後は、プロセスが正常に完了したかどうかを確認できます。プロセスに失敗した場合、クエリへの応答には失敗の原因が含まれています。

イベントのステータスをクエリするには、次の REST URL で GET コマンドを使用します。

`http://<DIH_console_host_port>/dih-console/api/v1/event/<eventId>`

以下に例を示します。

`http://localhost:18080/dih-console/api/v1/event/2435`

サンプルファイル

`<DIH_InstallationDir>\samples\api\restapi\examples\event` フォルダには次のファイルが含まれています。

- `EventResponse.java`: イベント応答モデルクラス。
- `HttpRestClientEventApi.java`: イベントステータスをクエリする、Java API に基づくサンプル HTTP クライアント。このクライアントは `org.codehaus.jackson.map.ObjectMapper` を使用してイベントステータスを解析し、`EventResponse` オブジェクトに json 応答を送信します。詳細については、次の Web サイトを参照してください。
 - <http://wiki.fasterxml.com/JacksonDownload>
 - <http://mvnrepository.com/artifact/org.codehaus.jackson>
- `SprintRestTemplateClientEventApi.java`: `SpringRest` テンプレートを使用してイベントステータスをクエリするサンプルクライアント。クライアントアプリケーションが `Spring` コンテナに格納されている場合は、このクライアントを使用します。

イベントステータス API の応答

Data Integration Hub イベントステータス API を使用してパブリケーションまたはサブスクリプションイベントのステータスをクエリした場合、API は `EventResponse.java` モデルクラス内でイベント応答を返します。

応答には次の情報が含まれています。

プロパティ	説明
<code>responseCode</code>	Run Publication Subscription API アクションの応答。
<code>eventId</code>	Data Integration Hub が生成するパブリケーションまたはサブスクリプションイベントの ID。

プロパティ	説明
eventType	Data Integration Hub が生成するパブリケーションまたはサブスクリプションイベントのタイプ。
topicName	パブリケーションまたはサブスクリプションに関連付けられたトピックの名前。
publicationName または subscriptionName	パブリケーションまたはサブスクリプションの名前。
applicationName	公開元またはサブスクライブ元のアプリケーションの名前。
eventStatus	Data Integration Hub が生成するパブリケーションまたはサブスクリプションイベントのステータス。
eventStartTimeLong	パブリケーションイベントまたはサブスクリプションイベントの開始時間。Java API <code>java.lang.System.currentTimeMillis</code> が返したシステム時間（ミリ秒）です。
eventEndTimeLong	パブリケーションイベントまたはサブスクリプションイベントの終了時間。Java API <code>java.lang.System.currentTimeMillis</code> が返したシステム時間（ミリ秒）です。
referencedEventsList	ファイルパブリケーションイベント、集計済みサブスクリプションイベント、および複合サブスクリプションイベントに該当します。ファイルパブリケーション、集計済みサブスクリプション、または複合サブスクリプションイベントに関連するイベント ID のリストです。 例えば、ファイルパブリケーションイベントの <code>referencedEventsList</code> には、パブリケーションイベントで公開されるファイルのファイルイベントが含まれます。
isFinal	イベントが最終状態かどうか。
isError	イベントがエラーステータスかどうか。
sourceSuccessRows	Data Integration Hub が正常に読み取ったソース行の数。
sourceFailedRows	Data Integration Hub が読み取りに失敗したソース行の数。
targetFailedRows	Data Integration Hub が書き込みに失敗したターゲット行の数。
targetSuccessRows	Data Integration Hub が正常に書き込んだターゲット行の数。
detailedMessage	エラーステータスのイベントに該当します。エラーの原因が Data Integration Hub の場合、 <code>detailedMessage</code> は Data Integration Hub イベントのエラーメッセージを返します。認証エラーや REST URL 要求が正しくない場合などのその他のエラーの場合、 <code>detailedMessage</code> にはエラーの原因を示すメッセージが含まれます。

イベントステータス API の応答例

パブリケーションイベント 4003 のステータスをクエリする要求への応答:

```
{
  "responseCode": "SUCCESS",
  "eventId": 4003,
  "eventType": "Publication",
  "topicName": "top_120",
  "publicationName": "ng_pub_120_1",
  "applicationName": "appl",
  "eventStatus": "Complete",
```

```

"eventStartTimeLong": 1431078308560,
"eventEndTimeLong": 1431078313780,
"isFinal": true,
"isError": false,
"sourceSuccessRows": 10,
"sourceFailedRows": 0,
"targetFailedRows": 0,
"targetSuccessRows": 10}

```

サブスクリプションイベント 3008 および 3007 を含む集計済みサブスクリプションイベント 3009 のステータスをクエリする要求への応答

```

{
  "responseCode": "SUCCESS",
  "eventId": 3009,
  "eventType": "Aggregated Subscription",
  "topicName": "topic1",
  "subscriptionName": "sub1",
  "applicationName": "appl",
  "eventStatus": "Complete",
  "eventStartTimeLong": 1431065700088,
  "eventEndTimeLong": 1431065704372,
  "referencedEventsList": "3008,3007"
  "isFinal": true,
  "isError": false,
  "sourceSuccessRows": 15,
  "sourceFailedRows": 0,
  "targetFailedRows": 0,
  "targetSuccessRows": 15
}

```

パブリケーションプロセスが失敗したパブリケーションイベント 3016 のステータスをクエリする要求への応答。

Response:

```

{
  "responseCode": "SUCCESS",
  "eventId": 3016,
  "eventType": "Publication",
  "topicName": "top_120",
  "publicationName": "ng_pub_120_1",
  "applicationName": "appl",
  "eventStatus": "Error",
  "eventStartTimeLong": 1431066353202,
  "eventEndTimeLong": 1431066357162,
  "isFinal": true,
  "isError": true,
  "sourceSuccessRows": 2,
  "sourceFailedRows": 1,
  "targetFailedRows": 1,
  "targetSuccessRows": 2,
  "detailedMessage": "Error while copying several rows :\nSrcFailedRows: 1\nTgtFailedRows: 1\nSrcSuccessRows: 2\nTgtSuccessRows: 2\nPowerCenter workflow: s__DIH_pub_ng_pub_120_1\nPowerCenter session: s__DIH_pub_ng_pub_120_1\nCheck the PowerCenter session log for more information."
}

```

Change Event Status API

Data Integration Hub Change Event Status API を使用して、既存イベントのステータスを変更します。

イベントのステータスを変更するには、次の API URL を使用します。

http://<DIH_console_host_port>/dih-console/api/v1/event/changestatus

以下に例を示します。

http://localhost:18080/dih-console/api/v1/event/changestatus

必要なヘッダー、本文、認証詳細を入力する必要があります。

イベントのステータス変更の要求構文

既存イベントのステータスを変更するには、次の要求本文構文を使用します。

```
{
  "eventId": "<event ID>",
  "eventStatusName": "<name of the event status>",
  "comment": "<Specify a comment to be displayed.>"
}
```

例:

```
{
  "eventId": "270003",
  "eventStatusName": "Complete",
  "comment": "From Rest API"
}
```

サンプルファイル

<DIH_InstallationDir>\samples\rest_api_samples フォルダに、サンプルのクライアントアプリケーションと、アプリケーションについて説明した readme ファイルが含まれています。

Change Event Status API の応答

Data Integration Hub API を使用して既存イベントのステータスを変更すると、Data Integration Hub は、API の応答で、実行したアクションの応答コードを返します。

REST API を使用してイベントのステータスを変更すると、以下のいずれかの応答コードが返されます。

- Data Integration Hub がイベントのイベントステータスを正常に変更すると、API は SUCCESS 応答を返します。

以下に例を示します。

```
SUCCESS
{
  "responseCode": "SUCCESS"
}
```

- Data Integration Hub がイベントのイベントステータスの変更失敗すると、応答に失敗の理由が提供されます。例えば、操作を実行するための十分な特権がない場合などです。

以下に例を示します。

```
INSUFFICIENT_PRIVILEGES
{
  "responseCode": "INSUFFICIENT_PRIVILEGES",
  "message": "The user does not have sufficient privileges or permissions to perform this operation on the event."
}
```

第 11 章

データ抽出 API

この章では、以下の項目について説明します。

- [データ抽出 API の概要, 114 ページ](#)
- [カタログ API, 114 ページ](#)
- [Data Integration Hub イベントビュー, 119 ページ](#)

データ抽出 API の概要

Data Integration Hub からデータを抽出するには、データ抽出 API を使用します。

次の Data Integration Hub API を使用して、データを抽出できます。

- カタログ API。Data Integration Hub カタログからデータ（トピック、パブリケーション、サブスクリプションメタデータなど）を抽出します。
- イベント API。Data Integration Hub イベントのデータベースビューを提供します。

カタログ API

Data Integration Hub カタログから、トピックのメタデータや各トピックに関連付けられているパブリケーションおよびサブスクリプションに関するメタデータなどのデータを抽出するには、カタログ API を使用します。

表示特権と読み取りアクセス許可のあるトピック、パブリケーション、およびサブスクリプションに関連するメタデータを抽出できます。

カタログからデータを抽出するには、次の REST URL を使用します。

`http://localhost:18080/dih-console/api/v1/catalog/topics`

Data Integration Hub カタログ API の応答

Data Integration Hub カタログ API を使用して、Data Integration Hub カタログからデータを抽出する場合、必要な特権のあるすべてのトピックに関するメタデータを含む JSON 文字列が返されます。

文字列には、応答の各トピックの次のデータが含まれます。

`topicName`

トピックの名前。

topicDesc

トピックの説明。

topicType

トピックのタイプ: 差分または完全。

topicTables

トピック内の各テーブルについて、テーブル名とその各フィールドの詳細情報が含まれるエントリ。

パブリケーション

トピックに関連付けられているパブリケーションごとに、次のデータが提供されます。

publicationName

パブリケーションの名前。

publicationDesc

パブリケーションの説明。

applicationName

アプリケーション。パブリケーションは、このアプリケーションからデータまたはファイルを公開します。

publicationSourceType

パブリケーションソースのタイプ。

publicationConnectionName

リレーショナルデータベースパブリケーションおよび HDFS パブリケーションの場合: 接続の名前。パブリケーションワークフローは、公開されるデータまたはファイルをこの接続から読み取ります。

publicationDBType

リレーショナルデータベースパブリケーションの場合: データベースのタイプ。

サブスクリプション

トピックに関連付けられているサブスクリプションごとに、次のデータが提供されます。

subscriptionName

サブスクリプションの名前。

subscriptionDesc

サブスクリプションの説明。

applicationName

トピックのデータまたはファイルを使用するアプリケーション。

subscriptionTargetType

サブスクリプションターゲットのタイプ。

subscriptionConnectionName

リレーショナルデータベースサブスクリプションおよび HDFS サブスクリプションの場合: 接続の名前。サブスクリプションワークフローは、アプリケーションが使用するデータまたはファイルをこの接続に書き込みます。

subscriptionDBType

リレーショナルデータベースサブスクリプションの場合: データベースのタイプ。

Data Integration Hub カタログ API の応答例

次の例は、Data Integration Hub カタログからデータを抽出する要求に対する応答を示しています。

```
{
  "responseCode": "SUCCESS",
  "catalogTopics": [
    {
      "topicName": "FileTopic",
      "topicDesc": null,
      "topicType": "Delta"
      "topicTables": [
        {
          "tableName": "Orders"
          "tableFields": [
            {
              "name": "_Name_"
              "fieldType": "STRING"
              "nullable": false
              "scale": -1
              "precision": 255
              "length": 255
              "primaryKey": false
              "filterAccelerator": false
              "encrypted": true
            },
            {
              "name": "_Type_"
              "fieldType": "STRING"
              "nullable": false
              "scale": -1
              "precision": 255
              "length": 255
              "primaryKey": false
              "filterAccelerator": false
              "encrypted": true
            },
            {
              "name": "_ParentId_"
              "fieldType": "STRING"
              "nullable": false
              "scale": -1
              "precision": 255
              "length": 255
              "primaryKey": false
              "filterAccelerator": false
              "encrypted": true
            }
          ]
        }
      ]
    },
    {
      "publicationName": "FilePub",
      "publicationDesc": null,
      "applicationName": "FileApp",
      "publicationSourceType": "FLAT_FILE",
      "publicationConnectionName": null,
      "publicationDBType": null
    },
    {
      "subscriptionName": "FileSub",
      "subscriptionDesc": null,
      "applicationName": "FileApp",
      "subscriptionTargetType": "FLAT_FILE",
      "subscriptionConnectionName": null,
      "subscriptionDBType": null
    }
  ]
}
```

```

{
  "topicName": "OrderTopic",
  "topicDesc": null,
  "topicType": "Delta",
  "topicTables": [
    {
      "tableName": "OrderTable"
      "tableFields": [
        {
          "name": "_Name_"
          "fieldType": "STRING"
          "nullable": false
          "scale": -1
          "precision": 255
          "length": 255
          "primaryKey": false
          "filterAccelerator": false
          "encrypted": true
        },
        {
          "name": "_Type_"
          "fieldType": "STRING"
          "nullable": false
          "scale": -1
          "precision": 255
          "length": 255
          "primaryKey": false
          "filterAccelerator": false
          "encrypted": true
        },
        {
          "name": "_ParentId_"
          "fieldType": "STRING"
          "nullable": false
          "scale": -1
          "precision": 255
          "length": 255
          "primaryKey": false
          "filterAccelerator": false
          "encrypted": true
        },
        {
          "name": "_StartDate_"
          "fieldType": "STRING"
          "nullable": false
          "scale": -1
          "precision": 255
          "length": 255
          "primaryKey": false
          "filterAccelerator": false
          "encrypted": true
        },
        {
          "name": "_EndDate_"
          "fieldType": "STRING"
          "nullable": false
          "scale": -1
          "precision": 255
          "length": 255
          "primaryKey": false
          "filterAccelerator": false
          "encrypted": true
        }
      ]
    },
    {
      "tableName": "CustomerTable"
      "tableFields": [
        {
          "name": "_Name_"
          "fieldType": "STRING"
          "nullable": false

```

```

        "scale":-1
        "precision":255
        "length":255
        "primaryKey":false
        "filterAccelerator":false
        "encrypted":true
    },
    {
        "name": "_Type_"
        "fieldType": "STRING"
        "nullable": false
        "scale": -1
        "precision": 255
        "length": 255
        "primaryKey": false
        "filterAccelerator": false
        "encrypted": true
    },
    {
        "name": "_ParentId_"
        "fieldType": "STRING"
        "nullable": false
        "scale": -1
        "precision": 255
        "length": 255
        "primaryKey": false
        "filterAccelerator": false
        "encrypted": true
    },
    {
        "name": "_ExpectedRevenue_"
        "fieldType": "STRING"
        "nullable": false
        "scale": -1
        "precision": 255
        "length": 255
        "primaryKey": false
        "filterAccelerator": false
        "encrypted": true
    },
    {
        "name": "_IsActive_"
        "fieldType": "STRING"
        "nullable": false
        "scale": -1
        "precision": 255
        "length": 255
        "primaryKey": false
        "filterAccelerator": false
        "encrypted": true
    }
    },
    {
    },
    ],
    "publications": [
    {
        "publicationName": "OrdersPublication",
        "publicationDesc": null,
        "applicationName": "OrderPublications",
        "publicationSourceType": "RDBMS",
        "publicationConnectionName": "OrderSource",
        "publicationDBType": "SQLSERVER"
    }
    ],
    "subscriptions": [
    {
        "subscriptionName": "OrdersSubscription",
        "subscriptionDesc": null,
        "applicationName": "OrderSubscriptions",

```

```

        "subscriptionTargetType": "FLAT_FILE",
        "subscriptionConnectionName": null,
        "subscriptionDBType": null
    },
    {
        "subscriptionName": "OrderSubs",
        "subscriptionDesc": null,
        "applicationName": "OrderSubscriptions",
        "subscriptionTargetType": "RDBMS",
        "subscriptionConnectionName": "OrderTarget",
        "subscriptionDBType": "SQLSERVER"
    }
]
}

```

Data Integration Hub イベントビュー

Data Integration Hub リポジトリの Data Integration Hub イベントを表示するには、SQL クライアントを使用します。

イベントビューは、SQL クライアントの **DX_VIEW_PUB_SUB_EVENT** の Data Integration Hub データベースに存在します。

ビューには、次のカラムが表示されます。

EVENT_ID

イベントの ID。

EVENT_TYPE

イベントのタイプ（パブリケーション、サブスクリプション、ファイル、およびシステムイベントなど）。

PUBLICATION_SUBSCRIPTION_NAME

イベントを生成したパブリケーションまたはサブスクリプションの名前。

APPLICATION_NAME

イベントを生成したパブリケーションまたはサブスクリプションに関連付けられているアプリケーションの名前。

TOPIC_NAME

イベントを生成したパブリケーションまたはサブスクリプションに関連付けられているトピックの名前。

EVENT_STATUS

イベントステータス（エラー、完了、および保留中など）。

EVENT_START_TIME

イベントの開始日時。

EVENT_END_TIME

イベントの終了日時。

EVENT_DURATION

イベントの期間。Data Integration Hub は、EVENT_START_TIME および EVENT_END_TIME に基づいて、期間を計算します。

SOURCE_SUCCESS_ROWS

Data Integration Hub によるソースからの読み取りが成功した行の数。

SOURCE_FAILED_ROWS

Data Integration Hub によるソースからの読み取りが失敗した行の数。

TARGET_SUCCESS_ROWS

Data Integration Hub によるターゲットからの読み取りが成功した行の数。

TARGET_FAILED_ROWS

Data Integration Hub によるターゲットからの読み取りが失敗した行の数。

注:

- パブリケーションの場合、ソースは公開元のアプリケーションを参照し、ターゲットは Data Integration Hub パブリケーションリポジトリを参照します。
- サブスクリプションの場合、ソースは Data Integration Hub パブリケーションリポジトリを参照し、ターゲットはサブスクライブ元のアプリケーションを参照します。
- 成功したソースおよびターゲット行の数と、失敗したソースおよびターゲット行の数は、Data Integration Hub バージョン 9.6.2 をインストールした後、または古いバージョンの Data Integration Hub をバージョン 9.6.2 にアップグレードした後に Data Integration Hub リポジトリで作成されたイベントの場合にのみ表示されます。

索引

A

API

Run Publication Subscription [102](#), [104](#)
イベントの再処理 [109](#)
パブリケーションサブスクリプションモード [107](#)

B

Big Data

説明 [15](#)

Big Data Management

マッピングの作成 [41](#), [43](#), [45](#)

Big Data Management マッピング

create application [42](#), [46](#)

作成 [41](#), [43](#), [45](#)

サブスクリプション接続 [45](#)

サブスクリプションデータオブジェクト [45](#)

式 [42](#)

パブリケーションデータオブジェクト [41](#), [43](#)

パブリケーション接続 [41](#), [43](#)

パラメータ [41](#), [46](#)

フィルタ [46](#)

ランタイム環境 [42](#), [46](#)

Big Data Streaming

マッピングの作成 [41](#), [43](#)

Big Data Streaming マッピング

Java トランスフォーメーション [43](#)

アプリケーションの作成 [44](#)

パブリケーションデータオブジェクト [41](#), [43](#)

パブリケーション接続 [41](#), [43](#)

ランタイム環境 [44](#)

作成 [41](#), [43](#)

式 [43](#)

C

change event status API

プロセスのステータス [112](#)

create application

Big Data Management マッピング [42](#), [46](#)

Data Quality マッピング [54](#), [56](#)

D

Data Engineering Integration

サブスクリプションマッピング [44](#)

パブリケーションのマッピング [40](#)

マッピング [38-40](#), [44](#)

Data Engineering Integration マッピング

開発 [40](#), [44](#)

作業を開始する前に [40](#)

説明 [38](#)

Data Engineering Streaming

パブリケーションのマッピング [42](#)

マッピング [38-40](#), [42](#)

Data Engineering Streaming マッピング

開発 [42](#)

作業を開始する前に [40](#)

説明 [38](#)

Data Integration Hub ワークフロー

イベント属性の追加 [69](#)

ワークフローパラメータのカスタマイズ [69](#)

権限 [66](#), [70](#)

作成 [66](#)

削除 [70](#)

全般的なパラメータの定義 [67](#)

定義 [65](#)

編集集中 [70](#)

Data Quality

サブスクリプションマッピング [54](#)

パブリケーションのマッピング [52](#)

マッピング [51](#), [52](#), [54](#)

マッピングの作成 [53](#), [55](#)

Data Quality マッピング

create application [54](#), [56](#)

作成 [53](#), [55](#)

サブスクリプションデータオブジェクト [55](#)

式 [54](#)

パブリケーション接続 [53](#)

パブリケーションデータオブジェクト [53](#)

パラメータ [53](#), [55](#)

フィルタ [56](#)

ランタイム環境 [54](#), [56](#)

開発 [52](#), [54](#)

作業を開始する前に [52](#)

説明 [51](#)

DX_Add_Document_To_Event

定義 [77](#)

プロパティ [78](#)

ポート [77](#), [78](#)

DX_Event_Attribute

定義 [79](#)

プロパティ [79](#)

ポート [79](#)

DX_Event_Details

定義 [80](#)

プロパティ [80](#)

ポート [80](#)

DX_Generate_Temporary_File

定義 [81](#)

プロパティ [82](#)

ポート [81](#)

DX_Notification

定義 [82](#)

プロパティ [83](#)

ポート [83](#)

DX_Publication_Parameters

定義 [84](#)

DX_Publication_Parameters (続く)

ポート [84](#)

DX_Start_Publication

定義 [85](#)

プロパティ [86](#)

ポート [85](#)

DX_Throw_Error

定義 [86](#)

プロパティ [87](#)

ポート [86, 87](#)

F

forms designer

[Designer (デザイナー)] タブ [97](#)

アクション [98](#)

グループプロパティ [100](#)

定義 [97](#)

要素のプロパティ [99](#)

I

Informatica Cloud

マッピング [62](#)

Informatica Intelligent Cloud Services

タスク [63](#)

マッピング [63](#)

J

Java トランスフォーメーション

Big Data Streaming マッピング [43](#)

P

PowerCenter

ソース定義の作成 [30](#)

ソースとターゲット [30](#)

ターゲット定義の作成 [31](#)

マッピング [22](#)

マッピングの作成 [32](#)

ワークフロー [22, 23](#)

ワークフローのエクスポート [33, 36](#)

ワークフローの作成 [32](#)

ワークフローの保存 [33, 36](#)

PowerCenter バッチワークフロー

ガイドライン [29](#)

PowerCenter マッピング

カスタム [25](#)

自動 [24](#)

PowerCenter ワークフロー

カスタム [25](#)

カスタムマッピングの開発 [29](#)

自動 [24](#)

タイプ [23](#)

バッチ [29](#)

R

REST API

Run Publication Subscription [102](#)

イベントの再処理 [109](#)

パブリケーションサブスクリプションモード [107](#)

Run Publication Subscription API

REST API [102, 104](#)

コマンドライン [107](#)

コマンドライン API [104](#)

プロセスのステータス [110, 111](#)

Run Publication Subscription REST API

アクションステータス [104](#)

要求 [102](#)

Run Publication Subscription REST API アクション

ステータス [104](#)

Run Publication Subscription コマンドライン API

コマンドの構文 [104](#)

通知 [107](#)

Run Publication Subscription のステータス

イベントステータス API [110, 111](#)

Run Publication Subscription プロセス

ステータス [110, 111](#)

あ

アーキテクチャ

コンポーネント [14](#)

アプリケーションの作成

Big Data Streaming マッピング [44](#)

い

イベントステータス API

プロセスのステータス [110, 111](#)

応答 [110, 111](#)

イベントの再処理 API

REST API [109](#)

イベントの再処理 API REST API

アクションステータス [109](#)

イベントの再処理 API REST API アクション

ステータス [109](#)

イベントの再処理

REST API [109](#)

イベントビュー

説明 [119](#)

イベント属性

Data Integration Hub ワークフロー [69](#)

か

ガイドライン

PowerCenter バッチワークフロー [29](#)

カスタム PowerCenter マッピングおよびワークフロー [26](#)

自動 PowerCenter マッピングおよびワークフロー [24](#)

カスタム PowerCenter マッピング

ガイドライン [26](#)

説明 [25](#)

カスタム PowerCenter ワークフロー

ガイドライン [26](#)

説明 [25](#)

カスタムマッピング

開発 [29](#)

タスク [63](#)

タスクのガイドライン [63](#)

マッピング [63](#)

マッピングのガイドライン [63](#)

カタログ API

応答 [114](#)

説明 [114](#)

カタログ API の応答
例 [116](#)

く

クラウド
mapping [61](#)
タスク [61](#), [62](#)
マッピング [62](#)
クラウドタスク
サブスクリプションプロセス [20](#)
パブリケーションプロセス [18](#)
定義 [61](#)
クラウドマッピング
定義 [61](#)

こ

コマンドライン API
Run Publication Subscription [104](#)

さ

サブスクリプション
プロセス [19](#)
定義 [17](#)
サブスクリプション接続
Big Data Management [45](#)
サブスクリプションソース
Big Data Management [45](#)
Data Quality [55](#)
サブスクリプションターゲット
Big Data Management [45](#)
Data Quality [55](#)
サブスクリプションデータオブジェクト
Big Data Management [45](#)
Data Quality [55](#)
サブスクリプションプロセス
クラウドタスク [20](#)
データドリブン [20](#)
バッチワークフロー [19](#)
サブスクリプションマッピング
Data Engineering Integration [44](#)
Data Quality [54](#)

し

式
Big Data Management マッピング [42](#)
Big Data Streaming マッピング [43](#)
Data Quality マッピング [54](#)
自動 PowerCenter マッピング
ガイドライン [24](#)
説明 [24](#)
ログ [25](#)
自動 PowerCenter ワークフロー
ガイドライン [24](#)
説明 [24](#)
ログ [25](#)

そ

操作コンソール
説明 [16](#)

た

タスク
ガイドライン [63](#)
ダッシュボードとレポート
KPI の定義 [89](#)
Logi Info Studio でのカスタマイズ [94](#)
Logi Info Studio でのカスタムアプリケーションのセットアップ [95](#)
Logi Info Studio での構造 [90](#)
Logi Info Studio のインストール [95](#)
カスタム LogiXML プロパティ [93](#)
カスタムパネルの追加 [96](#)
カスタムレポートの作成 [96](#)
デフォルトの JavaScript サポートファイル [92](#)
デフォルトの KPI [89](#)
デフォルトの SQL 定数 [91](#)
デフォルトのテーマ変数 [93](#)
デフォルトのフィルタ [90](#)
デブロイとテスト [96](#)
概要 [88](#)

て

データ抽出 API
イベントビュー [119](#)
カタログ API [114](#)
説明 [114](#)
データドリブン
サブスクリプションプロセス [20](#)
パブリケーションプロセス [19](#)

と

トピック
プロパティ [73](#)
概要 [16](#)
トピックプロパティ
イベント属性 [73](#)
権限 [73](#)
トランスフォーメーション
DX_Add_Document_To_Event [77](#)
DX_Event_Attribute [79](#)
DX_Event_Details [80](#)
DX_Generate_Temporary_File [81](#)
DX_Notification [82](#)
DX_Publication_Parameters [84](#)
DX_Start_Publication [85](#)
DX_Throw_Error [86](#)
エラー処理 [76](#)
ガイドライン [76](#)
構成 [75](#)
定義 [74](#)
プラグインのインストール [75](#)

は

バッチワークフロー
ガイドライン [29](#)
開発 [29](#)

バッチワークフロー (続く)
サブスクリプションプロセス [19](#)
定義 [23](#)
パブリケーションプロセス [17](#)
バッチワークフローの開発
プロセス [29](#)
パブリケーション
プロセス [17](#)
定義 [17](#)
パブリケーションおよびサブスクリプションの API
説明 [101](#)
パブリケーションサブスクリプションモード
REST API [107](#)
パブリケーションソース
Big Data Management [41, 43](#)
Big Data Streaming [41, 43](#)
Data Quality [53](#)
パブリケーションターゲット
Big Data Management [41, 43](#)
Big Data Streaming [41, 43](#)
Data Quality [53](#)
パブリケーションデータオブジェクト
Big Data Management [41, 43](#)
Big Data Streaming [41, 43](#)
Data Quality [53](#)
パブリケーションのマッピング
Data Engineering Integration [40](#)
Data Engineering Streaming [42](#)
Data Quality [52](#)
パブリケーションプロセス
クラウドタスク [18](#)
データドリブン [19](#)
バッチワークフロー [17](#)
リアルタイム [18](#)
パブリケーションリポジトリ
データ型 [26](#)
パブリケーション接続
Big Data Management [41, 43](#)
Big Data Streaming [41, 43](#)
Data Quality [53](#)
パラメータ
Big Data Management マッピング [41, 46](#)
Data Quality マッピング [53, 55](#)

ふ

フィルタ
Big Data Management マッピング [46](#)
Data Quality マッピング [56](#)

ま

マッピング
Data Engineering Integration [38-40, 44](#)
Data Engineering Streaming [38-40, 42](#)
Data Quality [51, 52, 54](#)
PowerCenter [22](#)
ガイドライン [62, 63](#)
マッピングの作成
Big Data Management [41, 43, 45](#)
Big Data Streaming [41, 43](#)
Data Quality [53, 55](#)

も

モード変更の API
REST API [108](#)
モード変更の REST API
アクションステータス [108](#)
モード変更の REST API アクション
ステータス [108](#)

ゆ

ユーザーロール
デベロッパ [21](#)

ら

ランタイム環境
Big Data Management マッピング [42, 46](#)
Big Data Streaming マッピング [44](#)
Data Quality マッピング [54, 56](#)

り

リアルタイムワークフロー
ガイドライン [33](#)
開発 [33](#)
ソースとターゲット [34](#)
トピックの作成 [36, 37](#)
パブリケーションプロセス [18](#)
マッピングの作成 [35](#)
ワークフローターゲットの名前の変更 [37](#)
ワークフローの作成 [36](#)
定義 [23](#)
リアルタイムワークフローの開発
プロセス [33](#)

れ

例
カタログ API の応答 [116](#)

ろ

ログ
自動 PowerCenter マッピングおよびワークフロー [25](#)

わ

ワークフロー
Data Integration Hub [65](#)
PowerCenter [22, 23](#)
タイプ [23](#)
パラメータプロパティ [73](#)
プロパティ [70](#)
リアルタイムワークフロー [33](#)
全般プロパティ [71](#)
ワークフロープロパティ
ワークフローパラメータ [73](#)
説明 [70](#)
全般 [71](#)