



Informatica® Mass Ingestion  
April 2024

# 一括取り込みアプリケーション

© 著作権 Informatica LLC 2021, 2024

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 Cloud、Informatica Intelligent Cloud Services、PowerCenter、PowerExchange、および Informatica ロゴは、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。

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](mailto: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: 2024-05-20

# 目次

<b>序文</b>	<b>6</b>
Informatica のリソース	6
Informatica マニュアル	6
Informatica Intelligent Cloud Services Web サイト	6
Informatica Intelligent Cloud Services コミュニティ	6
Informatica Intelligent Cloud Services マーケットプレイス	7
データ統合のコネクタのドキュメント	7
Informatica ナレッジベース	7
Informatica Intelligent Cloud Services Trust Center	7
Informatica グローバルカスタマサポート	7
<b>第 1 章 : 一括取り込みアプリケーション</b>	<b>8</b>
使用例	8
システム要件	9
一括取り込みアプリケーションアーキテクチャ	9
サポートされるソース	10
Google Analytics ソースのガイドライン	11
Marketo ソースのガイドライン	11
Microsoft Dynamics 365 ソースのガイドライン	12
NetSuite ソースのガイドライン	13
Oracle Fusion Cloud ソースのガイドライン	14
Salesforce ソースのガイドライン	15
Salesforce Marketing Cloud ソースのガイドライン	16
SAP ECC および SAP S4/HANA ソースのガイドライン SAP ODP Extractor コネクタの使用	16
SAP Mass Ingestion コネクタを使用した SAP ソースのガイドライン	19
ServiceNow ソースのガイドライン	22
Workday ソースのガイドライン	23
Zendesk ソースのガイドライン	25
サポートされるターゲット	27
Amazon Redshift ターゲットのガイドライン	28
Amazon S3、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、 Microsoft Fabric OneLake、および Oracle Cloud Object Storage ターゲットのガイドライン	28
Apache Kafka ターゲットのガイドライン	39
Databricks Delta ターゲットのガイドライン	39
Google BigQuery ターゲットのガイドライン	41
Microsoft Azure Synapse Analytics ターゲットのガイドライン	42
Microsoft Azure SQL Database ターゲットのガイドライン	43
Oracle ターゲットのガイドライン	44
PostgreSQL ターゲットのガイドライン	45
Snowflake ターゲットのガイドライン	45

Avro データ型	48
ソーススキーマの変更の処理	49
ターゲットに対する削除を論理削除として適用する機能	50
すべてのソーステーブル変更操作の監査履歴のターゲットテーブルへの適用	51
アプリケーション取り込みタスクの設定	52
始める前に	52
基本的なタスク情報の定義	52
ソースの設定	53
ターゲットの設定	98
スケジュールとランタイムオプションの設定	132
アプリケーション取り込みタスクのデプロイ	135
アプリケーション取り込みジョブの実行	136
アプリケーション取り込みジョブの停止	137
アプリケーション取り込みジョブの再開	137
増分ロードジョブの再開とリカバリ	137
アプリケーション取り込みジョブの再開時のスキーマドリフトオプションのオーバーライド	138
アプリケーション取り込みジョブの再デプロイ	139
アプリケーション取り込みジョブのデプロイ解除	140
アプリケーション取り込みジョブの強制終了	140
ソースオブジェクトとターゲットオブジェクトの再同期	141
デフォルトデータ型のマッピング	142
Adobe Analytics ソースと Amazon Redshift ターゲット	142
Adobe Analytics ソースと Google BigQuery ターゲット	143
Adobe Analytics ソースと Microsoft Azure Synapse Analytics ターゲット	143
Adobe Analytics ソースと Oracle ターゲット	144
Adobe Analytics ソースと Snowflake ターゲット	144
NetSuite ソースと Amazon Redshift ターゲット	145
NetSuite ソースと Databricks Delta ターゲット	145
NetSuite ソースと Google BigQuery ターゲット	146
NetSuite ソースと Microsoft Azure Synapse Analytics ターゲット	147
NetSuite ソースと Oracle ターゲット	147
NetSuite ソースと Snowflake ターゲット	148
Salesforce ソースと Amazon Redshift ターゲット	148
Salesforce ソースと Databricks Delta ターゲット	149
Salesforce ソースと Google BigQuery ターゲット	150
Salesforce ソースと Microsoft Azure Synapse Analytics ターゲット	152
Salesforce ソースと Microsoft Azure SQL Database ターゲット	153
Salesforce ソースと Oracle ターゲット	154
Salesforce ソースと PostgreSQL ターゲット	155
Salesforce ソースと Snowflake ターゲット	156
Salesforce Marketing Cloud ソースと Amazon Redshift ターゲット	157
Salesforce Marketing Cloud ソースと Databricks Delta ターゲット	157

Salesforce Marketing Cloud ソースと Google BigQuery ターゲット. . . . .	158
Salesforce Marketing Cloud ソースと Microsoft Azure Synapse Analytics ターゲット. . . . .	158
Salesforce Marketing Cloud ソースと Oracle ターゲット. . . . .	159
Salesforce Marketing Cloud ソースと Snowflake ターゲット. . . . .	159
SAP ソースと Amazon Redshift ターゲット. . . . .	160
SAP ソースと Databricks Delta ターゲット. . . . .	160
SAP ソースと Google BigQuery ターゲット. . . . .	161
SAP ソースと Microsoft Azure Synapse Analytics ターゲット. . . . .	162
SAP ソースと Oracle ターゲット. . . . .	162
SAP ソースと Snowflake ターゲット. . . . .	163
<b>索引. . . . .</b>	<b>165</b>

# 序文

一括取り込み一括取り込みアプリケーションアプリケーション取り込みアプリケーション取り込み一括取り込み」をお読みください。

## 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 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](mailto:infa_documentation@informatica.com)) までご連絡ください。

### Informatica Intelligent Cloud Services Web サイト

Informatica Intelligent Cloud Services Web サイト (<http://www.informatica.com/cloud>) にアクセスできます。このサイトには、Informatica Cloud 統合サービスに関する情報が含まれます。

### Informatica Intelligent Cloud Services コミュニティ

Informatica Intelligent Cloud Services コミュニティを使用して、技術的な問題について議論し、解決します。また、技術的なヒント、マニュアルの更新情報、FAQ（よくある質問）への答えを得ることもできます。

次の Informatica Intelligent Cloud Services コミュニティにアクセスします。

<https://network.informatica.com/community/informatica-network/products/cloud-integration>

開発者は、次の Cloud 開発者コミュニティで詳細情報を確認したり、ヒントを共有したりできます。

<https://network.informatica.com/community/informatica-network/products/cloud-integration/cloud-developers>

## Informatica Intelligent Cloud Services マーケットプレイス

Informatica マーケットプレイスにアクセスすると、データ統合コネクタ、テンプレート、およびマップレットを試用したり購入したりできます。

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

## データ統合のコネクタのドキュメント

データ統合のコネクタのドキュメントには、マニュアルポータルからアクセスできます。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。

## 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](mailto:KB_Feedback@informatica.com)) です。

## Informatica Intelligent Cloud Services Trust Center

Informatica Intelligent Cloud Services Trust Center は、Informatica のセキュリティポリシーおよびリアルタイムでのシステムの可用性について情報を提供します。

Trust Center (<https://www.informatica.com/trust-center.html>) にアクセスします。

Informatica Intelligent Cloud Services Trust Center にサブスクライブして、アップグレード、メンテナンス、およびインシデントの通知を受信します。[Informatica Intelligent Cloud Services Status](#) ページには、すべての Informatica Cloud 製品の実稼働ステータスが表示されます。メンテナンスの更新はすべてこのページに送信され、停止中は最新の情報が表示されます。更新と停止の通知がされるようにするには、Informatica Intelligent Cloud Services の 1 つのコンポーネントまたはすべてのコンポーネントについて更新の受信をサブスクライブします。すべてのコンポーネントにサブスクライブするのが、更新を逃さないようにするための最良の方法です。

サブスクライブするには、[Informatica Intelligent Cloud Services Status](#) ページで **【サブスクライブして更新】** をクリックします。電子メール、SMS テキストメッセージ、Webhook、RSS フィード、またはこの 4 つの任意に組み合わせとして送信される通知を受信するという選択ができます。

## Informatica グローバルカスタマサポート

グローバルサポートセンターには、Informatica Network または電話でお問い合わせください。

Informatica Network でオンラインサポートリソースを検索するには、Informatica Intelligent Cloud Services のヘルプメニューで **【サポートにお問い合わせください】** をクリックして、**Cloud Support** ページに移動します。**Cloud Support** ページには、システムステータス情報とコミュニティディスカッションが記載されています。追加のリソースを検索する場合や電子メールで Informatica グローバルカスタマサポートに問い合わせる場合は、Informatica Network にログインし、**【サポートが必要な場合】** をクリックしてください。

Informatica グローバルカスタマサポートの電話番号は、Informatica の Web サイト <https://www.informatica.com/services-and-training/support-services/contact-us.html> に掲載されています。

## 第 1 章

# 一括取り込みアプリケーション

一括取り込みアプリケーションは、Software-as-a-Service (SaaS) およびオンプレミスアプリケーションからクラウドベースのデータレイク、データウェアハウス、およびイベントストリーミングプラットフォームにデータを転送できます。

ビジネスまたは組織で使用される SaaS とオンプレミスのアプリケーションは、ビジネスに不可欠な大量のデータを毎日保存しています。一括取り込みアプリケーションを使用すると、アプリケーションによって保存されたデータを、大量のデータを処理できるターゲットに転送できます。データをターゲットに転送した後、そのデータを統合して、高度なデータ分析やデータウェアハウジングなどのさまざまな目的に使用できます。

一括取り込みアプリケーションは、次のタイプのロード操作を実行できます。

- **初期ロード。** ある時点で読み取られたソースデータをターゲットにロードします。データがロードされると、取り込みジョブは終了します。このロードタイプを使用して、後で増分変更が送信されるターゲットをマテリアライズできます。
- **増分ロード。** データの変更を継続的に、または取り込みジョブが停止または終了するまでロードします。取り込みジョブは、最後に実行されてから、または特定の開始点から発生した変更をロードします。このロードタイプを使用して、ターゲットのデータを最新の状態に保つことができるため、最新のデータに基づいてビジネスまたは組織の意思決定を行うことができます。
- **初期ロードと増分ロード。** ターゲットへのポイントインタイムデータの初期ロードを実行してから、同じソースオブジェクトに対して継続的に行われた増分データ変更のプロパゲートに自動的に切り替わります。

各ロードタイプでサポートされるソースとターゲットの詳細については、「[「サポートされるソース」 \(ページ 10\)](#)」および「[「サポートされるターゲット」 \(ページ 27\)](#)」を参照してください。

## 使用例

一括取り込みアプリケーションを使用して、さまざまなビジネス上の問題を解決できます。

次に、一括取り込みアプリケーションの使用例をいくつか示します。

- **データウェアハウジング:** SaaS とオンプレミスのアプリケーションのデータをクラウドベースのデータウェアハウスシステムに転送して整理します。データウェアハウスへのデータの初期バッチロードが完了すると、一括取り込みアプリケーションは、ソースアプリケーションからデータの変更を継続的にレプリケートして、データウェアハウス内のデータを最新の状態に保つことができます。
- **高度なデータ分析:** データレイクおよびデータウェアハウス内のアプリケーションデータを統合して、情報に基づいたビジネス上の意思決定に役立つ広範な分析を行います。
- **他のデータ処理アプリケーションでのアプリケーションデータの利用:** データレイクおよびデータウェアハウスを SaaS およびオンプレミスのソースと同期し続けて、最新のデータを他のアプリケーションに提供して処理します。



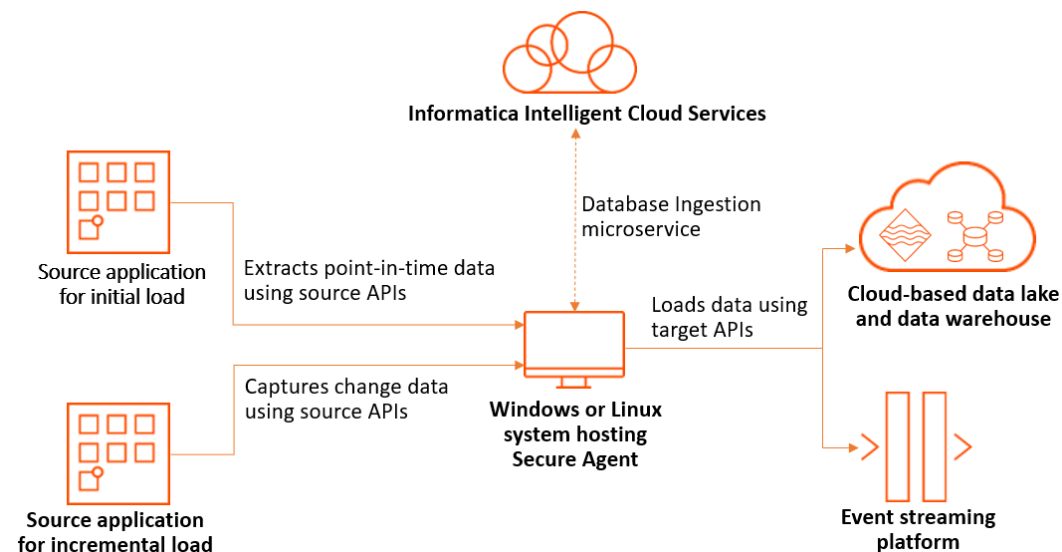
# システム要件

次の表に、Secure Agent に対する一括取り込みアプリケーションの最小システム要件を示します。

コンポーネント	最小要件
CPU あたりのコア数	最小 8 個、初期ロードで多数のソースオブジェクトを処理する必要がある場合は 16 個を推奨
メモリ	32GB
ディスク容量	2 KB の行サイズに基づいて、ジョブごとに 5 GB

## 一括取り込みアプリケーションアーキテクチャ

次の図は、一括取り込みアプリケーションアーキテクチャを示しています。



一括取り込みアプリケーションでアプリケーション取り込みジョブを実行するには、次のコンポーネントが必要です。

- **Secure Agent:** Secure Agent は、タスクを実行し、組織と Informatica Intelligent Cloud Services 間での安全な通信を可能にするプログラムです。一括取り込みアプリケーションを使用するすべてのシステムに Secure Agent をダウンロードする必要があります。アプリケーション取り込みジョブを実行すると、取り込みタスクのメタデータが Secure Agent にプッシュされ、取り込みジョブがデータを処理できるようになります。
- **データベース取り込みサービス:** データベース取り込みサービスは、Secure Agent がアプリケーション取り込みタスクとデータベース統合タスクの両方を実行するために使用するマイクロサービスです。データベース取り込みサービスは、Secure Agent をダウンロードするとシステムに自動的にダウンロードされます。
- **Informatica Intelligent Cloud Services インタフェース:** Informatica Intelligent Cloud Services には、アプリケーション取り込みタスクを作成してデプロイするための Web ベースのインタフェースが用意されています。タスクをデプロイすると、タスクの実行可能な取り込みジョブが作成されます。

- ソース API とターゲット API:** アプリケーション取り込みジョブは、ソース API を使用してソースオブジェクトからデータを取得し、ターゲット API を使用してデータをターゲットにロードします。増分ロード操作の場合、取り込みジョブは、特定の日時以降にソースオブジェクトに加えられた変更を識別し、関連付けられた取り込みタスクで定義されている間隔で変更を取得します。

## サポートされるソース

一括取り込みアプリケーションがサポートするソースは、アプリケーション取り込みタスクが、バッチ初期ロード操作でデータのポイントインタイムスナップショットを転送するか、特定の開始ポイントから増分変更データをロードするかによって異なります。

次の表に、一括取り込みアプリケーションがサポートしているソースタイプと、各ソースタイプでサポートされているロード操作のタイプを示します。

ソースタイプ	サポートされているロード操作
Adobe Analytics	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Google Analytics	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Marketo	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Microsoft Dynamics 365	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
NetSuite	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Oracle Fusion Cloud Applications	REST: 初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ BICC: 初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Salesforce	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Salesforce Marketing Cloud	初期ロード
SAP ERP Central Component (SAP ECC)	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ（接続タイプによって異なります。）
SAP ODP	SAP ECC: 初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ SAP S4/HANA: 初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
SAP Mass Ingestion	SAP ECC: Oracle データベースの初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ。Snowflake ターゲットで動作確認済みです。 SAP S4/HANA: HANA データベースの初期ロードと増分ロード。Snowflake ターゲットで動作確認済みです。
ServiceNow	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ

ソースタイプ	サポートされているロード操作
Workday	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ
Zendesk	初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせ

ソースタイプに使用するコネクタを判断するには、「コネクタと接続」 > 「一括取り込みアプリケーションコネクタ」を参照してください。

## Google Analytics ソースのガイドライン

Google Analytics ソースを使用する場合は、次のガイドラインを考慮してください。

- NetSuite ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- アプリケーション取り込みの初期ロードジョブでは、Google Analytics レポートの説明が入力データになります。各レポートには、ターゲットにレプリケートされるディメンションとメトリックのリストが含まれています。ディメンションとメトリックを含め、535 列あります。ただし、Google Analytics で取得できるのは、1 レポートにつき最大 10 個のメトリックと 9 個のディメンションです。複数のレポートを設定して、すべてのディメンションとメトリックを表示できます。
- アプリケーション取り込みタスクを設定する前に、レポートの説明ドキュメントを作成し、Secure Agent がアクセスできる場所に保存する必要があります。ファイルの読み取り権限が必要です。
- Google Analytics は、タイムスタンプで表される特定の間隔ではなく、日単位のデータのみを取得します。
- アプリケーションの取り込みに関するレポートの説明ドキュメントを更新する際は、次のガイドラインを考慮してください。
  - レポート内で名前を変更する際は、フィールドの順序を変更しないでください。新しいフィールド名は上書きする必要があります。
  - レポートに新しいフィールドを追加する際は、レポートのフィールドのリストの最後にフィールドを追加します。
  - フィールドを削除するには、レポートのフィールドのリストから対応するフィールド名を削除します。

## Marketo ソースのガイドライン

Marketo ソースを使用する場合は、次のガイドラインを考慮してください。

- Marketo ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- Marketo ソース用に設定された増分ロードジョブは、ソースオブジェクトに対して実行された挿入操作と更新操作のみをレプリケートできます。
- 一括取り込みアプリケーションは、Marketo オブジェクト間で定義されているリレーションをレプリケートしません。ターゲットには、ソース上のオブジェクト間に存在するリレーションは含まれません。
- デフォルトでは、Marketo インスタンスは 1 日あたりにダウンロード可能なデータが 500 MB に制限されます。さらに、Bulk API ダウンロードは期間が 31 日間に制限されます。

- 一部の Marketo オブジェクトでは、API がそれらに保存されているデータを直接取得することを許可していません。API がこのようなオブジェクトからデータを取得するには、入力パラメーターとして特定のフィルタが必要になります。API が使用するフィルタは、他の Marketo オブジェクトのフィールドです。一括取り込みアプリケーションは、他のオブジェクトに依存しているその Marketo オブジェクトに保存されているデータをレプリケートしません。
- アプリケーション取り込みジョブは、リードオブジェクトにリンクされていないカスタムオブジェクトに格納されているデータをプロパゲートしません。さらに、アプリケーション取り込みジョブは、表示名 ID を持つカスタムオブジェクトフィールドに格納されているデータをプロパゲートしません。
- デフォルトでは、一括取り込みアプリケーションは、次の Marketo オブジェクトの変更データをキャプチャできます。
  - キャンペーン
  - 電子メールテンプレート
  - フォルダ
  - ランディングページ
  - ランディングページテンプレート
  - プログラム
  - セグメンテーション
  - スマートキャンペーン
  - スマートリスト
  - スニペット
  - 静的リスト

**注:** キャンペーン、電子メールテンプレート、フォルダ、ランディングページ、ランディングページテンプレート、セグメンテーション、およびスニペットの各オブジェクトの変更データをキャプチャするには、`marketo.include.additional.cdc.objects` カスタムプロパティを TRUE に設定する必要があります。カスタムプロパティはアプリケーション取り込みタスクウィザードの【ソース】タブで設定できます。

## Microsoft Dynamics 365 ソースのガイドライン

Microsoft Dynamics 365 ソースを使用する場合は、次のガイドラインを考慮してください。

- Microsoft Dynamics 365 ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- Microsoft Dynamics 365 ソースの増分ロードジョブは、`modifiedon` フィールドのないオブジェクトの変更データをキャプチャしません。
- Microsoft Dynamics 365 ソースの増分ロードジョブ、および初期ロードと増分ロードの組み合わせジョブは、ソーススキーマに加えられた変更を検出およびレプリケートしません。
- Microsoft Dynamics 365 ソースの増分ロードジョブと、初期ロードと増分ロードの組み合わせジョブは、CDC サイクルの DELETE データ操作をキャプチャしません。
- ソースオブジェクトにロケール ID とタイムゾーンコードの値が含まれている場合、一括取り込みアプリケーションは、それらの値を実際の値に変換せずにターゲットにプロパゲートします。
- 一括取り込みアプリケーションは、次の データ型のフィールドをプロパゲートしません。
  - CUSTOMERTYPE
  - LOOKUPTYPE

- OWNERTYPE
- PARTYLISTTYPE
- VIRTUALTYPE
- CALENDARRULESTYPE
- UNKNOWN

## NetSuite ソースのガイドライン

NetSuite ソースを使用する場合は、次のガイドラインを考慮してください。

- NetSuite ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- NetSuite ソース用に設定された増分ロードジョブは、プライマリーキーを含まないテーブルの変更データを取得しません。
- 増分ロード ジョブは、次のタイムスタンプの少なくとも 1 つを含んでいないテーブルの変更データをキャプチャしません。
  - そのレコードが作成されたときのタイムスタンプ。
  - そのレコードが最後に変更されたときのタイムスタンプ。

**注:** レコードが作成されたときのタイムスタンプのみを含んだテーブルの場合、増分ロードジョブは、レコードに対して実行された挿入操作のみをキャプチャします。
- アプリケーション取り込みジョブがソースからドキュメントまたはイメージを取得する場合、ジョブは、ファイルのバイナリコンテンツをプロパゲートする代わりに、ドキュメントまたはイメージを含むファイルキャビネットの一意的識別子をプロパゲートします。
- 一括取り込みアプリケーションはソーステーブルからのレコードの削除は取得しません。
- NetSuite ソース用に設定された増分ロードジョブ、および初期ロードジョブと増分ロードジョブの組み合わせは、ソースフィールドの削除と名前変更を検出およびレプリケートしません。
- NetSuite.com データソース用に設定された NetSuite Mass Ingestion 接続がアプリケーション取り込みタスクですでに使用されており、デプロイ解除以外のステータスである場合は、接続を変更して NetSuite2.com データソースにアクセスしないでください。既存の接続を変更する代わりに、NetSuite2.com データソースに新しい接続を作成してから、この新しい接続を使用して新しいアプリケーション取り込みタスクを設定します。
- 一括取り込みアプリケーションは、次の NetSuite テーブルに格納されているデータをプロパゲートしません。
  - Account\_activity
  - Account\_period\_activity
  - Case\_types
  - Case\_origins
  - CompanyAddressbook
  - CompanyAddressbookEntityAddress
  - Deleted\_records
  - MAP\_wave\_custbody17
  - MAP\_wave\_custbody\_cseg4

- MAP\_wave\_custbody\_csegcs\_multsel

## Oracle Fusion Cloud ソースのガイドライン

Oracle Fusion Cloud ソースを使用する場合は、次のガイドラインを考慮してください。

- 一括取り込みアプリケーションは、次の Oracle Fusion Cloud Applications Suite のモジュールおよびアプリケーションのデータをレプリケートできます。

モジュール	アプリケーション
Enterprise Resource Planning (ERP)	Cloud Financials
	Procurement
	Project Management
	Risk Management and Compliance
Human Capital Management (HCM)	Human Resources
Industry	Higher Education
	Public Sector
	Digital Experience for Communications
	CX for Consumer Goods
	CX for Financial Services
Sales	Loyalty
	Partner Relationship Management
	Customer Data Management
	Sales Automation
	Subscription Management
Service	B2B Service
Supply Chain & Manufacturing (SCM)	AI Apps
	Cloud Service Logistics
	Fusion Cloud Inventory Management
	Maintenance
	Manufacturing
	Order Management

モジュール	アプリケーション
	Product Lifecycle Management
	Supply Chain Collaboration
	Supply Chain Planning

- 一括取り込みアプリケーションは、Business Intelligence Cloud Connector (BICC) または REST アプローチを使用して、ソースからターゲットにデータを転送します。
- Oracle Fusion Cloud ソース用に設定された増分ロードジョブおよび初期ロードと増分ロードの組み合わせジョブは、ソーススキーマへの変更をキャプチャおよびレプリケートしません。
- 増分ロード、および初期ロードと増分ロードの組み合わせタイプをサポートするには、データストアの BICC コンソールの **【増分サポート】** オプションを有効にします。**【増分サポート】** オプションを有効にするには、データストア内のカラムの少なくとも 1 つで、**TIMESTAMP** データ型の **【増分フィルタ】** がマークされている必要があります。
- デフォルトでは、増分ロードジョブは、LASTUPDATEDATE または CREATIONDATE フィールドを含んだオブジェクトの変更データのみをキャプチャします。ただし、Oracle Fusion Cloud (REST) の一部のオブジェクトには、LASTUPDATEDATE フィールドではなく CREATIONDATE フィールドのみが含まれます。CREATIONDATE フィールドのみを含むそのようなオブジェクトの変更データをキャプチャするには、アプリケーション取り込みタスクウィザードの **【ソース】** タブで、oraclefusion.cdc.allow.creationdate.objects カスタムプロパティを **true** に設定します。デプロイされたタスクのカスタムプロパティを **true** に設定した場合は、変更を反映するため、ジョブをデプロイ解除してからタスクを再度デプロイしてください。オブジェクトに CREATIONDATE フィールドしかない場合、LASTUPDATEDATE フィールドは使用できないため、挿入のみが記録され、更新と削除は無視されます。

## Salesforce ソースのガイドライン

Salesforce ソースを使用する場合は、次のガイドラインを考慮してください。

- Salesforce ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- デフォルトでは、増分ロードジョブは、lastmodifieddate または systemModstamp フィールドを含んだオブジェクトの変更データのみをキャプチャします。ただし、Salesforce の一部のオブジェクトには、lastmodifieddate フィールドと systemModstamp フィールドはなく、CreatedDate フィールドしかありません。CreatedDate フィールドのみを含むそのようなオブジェクトの変更データをキャプチャするには、アプリケーション取り込みタスクウィザードの **【ソース】** タブで、salesforce.cdc.allow.createddate.objects カスタムプロパティを true に設定します。デプロイされたタスクのカスタムプロパティを true に設定した場合は、変更を反映するため、ジョブをデプロイ解除してからタスクを再度デプロイしてください。
- Data.com オブジェクトをレプリケートする場合は、Salesforce アカウントに適切な Data.com ライセンスを割り当てる必要があります。
- 一括取り込みアプリケーションは、Salesforce オブジェクト間で定義されているリレーションをキャプチャしません。ターゲットには、ソースオブジェクト間のリレーションは含まれません。
- オブジェクトに複合フィールドが含まれている場合、一括取り込みアプリケーションは、複合フィールドをプロパゲートするのではなく、複合フィールドを構成するコンポーネントフィールドをプロパゲートします。例えば、複合フィールド geolocation がフィールド geolocation\_longitude と geolocation\_latitude の組み合わせである場合、一括取り込みアプリケーションは、複合フィールド geolocation をプロパゲートするの



ではなく、geolocation\_longitude フィールドと geolocation\_latitude フィールドに保存されているデータをターゲットに個別にロードします。

- ソースフィールドのマスキング設定は、ターゲットで保持されます。
- すべての増分ロードアクティビティは、標準の REST API のみを使用します。
- ルックアップフィールドは、ターゲット上で一意の英数字の識別子によって表されます。
- Function データ型のデフォルト値を持つ新しいフィールドが既存のソースオブジェクトに追加されると、一括取り込みアプリケーションは、デフォルト値をターゲットの既存の行にプロパゲートしません。ただし、新しい行がオブジェクトに追加されると、増分ジョブは新しく追加された行に対してフィールドに保存されている値をプロパゲートします。
- アプリケーション取り込みジョブは、Salesforce プラットフォームでインデックス処理されていない Big Object に保存されているデータをレプリケートしません。Salesforce ソースのアプリケーション取り込みタスクをデプロイする前に、タスクのオブジェクト選択ルールで選択されたすべての Big Object をインデックス処理してください。
- Decimal データ型のソースフィールドに、そのフィールドに定義された上限を超えるスケールの値が含まれている場合、アプリケーション取り込みジョブは、そのようなフィールドのプロパゲートに失敗する場合があります。ただし、フィールドのプロパゲート中に、スケールの値を定義された制限値にトリミングすることは可能です。スケールの値のトリミングは、カスタムプロパティ salesforce.decimal.trim.scale を true に設定することで行えます。

## Salesforce Marketing Cloud ソースのガイドライン

Salesforce Marketing Cloud ソースを使用する場合は、次のガイドラインを考慮してください。

- 一括取り込みアプリケーションは、Salesforce Marketing Cloud のデータエクステンションオブジェクトに保存されているデータを取り込むことができます。
- Salesforce Marketing Cloud ソースに対しては初期ロードジョブのみを設定できます。
- アプリケーション取り込みジョブは、データエクステンションオブジェクトをターゲットにレプリケートするときにオブジェクト名の先頭に DATA\_EXTENSION\_ を追加します。例えば、アプリケーション取り込みジョブがデータエクステンションオブジェクト Address をレプリケートした場合、ターゲットでのオブジェクト名は DATA\_EXTENSION\_Address になります。
- Salesforce Marketing Cloud ソースと Microsoft Azure Synapse Analytics ターゲットを使用したアプリケーション取り込み初期ロードジョブで、一部のオブジェクトのデプロイに失敗する場合があります。

## SAP ECC および SAP S4/HANA ソースのガイドライン SAP ODP Extractor コネクタの使用

SAP ODP Extractor コネクタで SAP ECC または SAP S4/HANA ソースを使用する場合は、次のガイドラインを考慮してください。

- SAP ソースを初めて使用する場合は、ソースに対してアプリケーション取り込みタスクを設定する前に、次の手順を実行します。
  1. 適切な SAP Notes が SAP サーバーで使用可能であることを確認します。詳細については、『SAP コネクタガイド』を参照してください。
  2. SAP Java Connector (SAP JCo) ライブラリを [SAP](#) Web サイトからダウンロードします。
  3. ダウンロードした SAP JCo ライブラリからネイティブファイルと JAR ファイルを次のディレクトリにコピーします。

```
<Secure_Agent>\ext\connectors\thirdparty\infa.odp\
```



4. sap-adapter-common.jar ファイルを<Secure\_Agent>\downloads\package-ICSAgent\_<バージョン>\package\ICS\main\bin\rdtm\javalib\sap から次のディレクトリにコピーします。  
<Secure\_Agent>\ext\connectors\thirdparty\infa.odp\
  5. Secure Agent を再起動します。
- SAP ソースに対してアプリケーション取り込みタスクを設定する前に、データの転送元のデータソースを必ず解放してください。次の関数を使用して、必要なデータソースを解放することができます。
    - BS\_ANLY\_DS\_RELEASE\_ODP
    - RODPS\_OS\_EXPOSE
  - SAP ソースの変更データを取得する増分ロードジョブを初めて実行するとき、ジョブは、ソースデータを含むデータストリーム内の最新の位置から変更レコードを取得してロードします。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
  - SAP ソースの増分ロードジョブは、プライマリキーを持たないデータソースの変更データをキャプチャしません。また、増分ロードジョブは、次の SAP データソースの変更データをキャプチャしません。
    - SAP が差分をサポートしていないデータソース。
    - 変更データを付加イメージの形式で提供するように設定されたデータソース。
  - アプリケーション取り込みタスクで SAP テーブルソースカラムに NULL 値が検出された場合、空白データを使用して NULL がターゲットテーブルにレプリケートされることはありません。
  - 一括取り込みアプリケーションは、SAP ソースのスキーマ変更を検出およびプロパゲートしません。
  - 一括取り込みアプリケーションは、次の廃止された SAP データ型のフィールドをプロパゲートしません。
    - D16S
    - D34S
    - PREC
    - VARC
- さらに、アプリケーション取り込みジョブは、RSTR データ型のフィールドをプロパゲートしません。

## SAP ユーザー権限の設定

SAP テーブルデータを処理する SAP ユーザーアカウントを設定します。

次の表に、SAP テーブルから読み取るために必要な承認を示します。

読み取りオブジェクト名	必要な承認
S_BTCH_JOB	DELE、LIST、PLAN、SHOW Job Operation を RELE に設定します。
S_PROGRAM	BTCSUBMIT、SUBMIT
S_RFC	SYST、SDTX、SDIFRUNTIME、/INFADI/TBLRDR、RFC1
S_TABU_DIS/ S_TABU_NUM	データを読み取る SAP テーブル名を指定します。

## SAP ODP コネクタのトランスポートファイルのインストール

アクセスする SAP マシンに SAP ODP Extractor のトランスポートファイルをインストールします。本番システムにトランスポートをインストールする前に、開発システムにトランスポートをインストールしてテストしてください。

SAP ODP オブジェクトからデータを抽出するための最新のトランスポートファイルをインストールしていることを確認します。

次のデータファイルと cofile をインストールして、SAP ODP オブジェクトからデータを読み取ります。

データおよび Co ファイル名	トランスポート要求	機能
- K900426.IN7 - R900426.IN7	IN7K900426	階層をサポートする SAP ODP オブジェクトから読み取る場合にのみ、トランスポートをインストールします。 階層をサポートしないオブジェクトについては、SAP ODP Extractor トランスポートファイルをインストールせずに SAP ODP Extractor コネクタを使用できます。

## トランスポートファイルのインストール

Unicode SAP システムから読み取るために、Secure Agent ディレクトリの最新のトランスポートファイルをインストールします。トランスポートファイルは、SAP バージョン ERP 6.0 EHP7 システム以降に対応しています。

- Secure Agent マシンの次のディレクトリに、トランスポートファイルがあります：  
<Informatica Secure Agent installation directory>\downloads\package-SAPODP.<Latest version>\package\sapodp\sap-transport
- アクセスする各 SAP マシンの SAP トランスポート管理ディレクトリにある Cofile ディレクトリに、cofile トランスポートファイルをコピーします。  
cofile トランスポートファイルでは、次の命名規則を使用します: <number>.<sap system>
- アクセスする各 SAP マシンの SAP トランスポート管理ディレクトリにある Data ディレクトリにデータトランスポートファイルをコピーします。  
データトランスポートファイルでは、次の命名規則を使用します: <number>.<sap-system>
- STMS でトランスポートを SAP にインポートするには、**【補足】** > **【その他の依頼】** > **【追加】** をクリックし、トランスポート要求をシステムキューに追加します。
- 【インポートキューに移送依頼追加】** ダイアログボックスに、cofile トランスポートの要求番号を入力します。  
要求番号は、名前を変更した cofile を次のような順序に置き替えたものです: <sap-system><number>
- インポートキューの要求領域で、追加したトランスポート要求番号を選択して、**【インポート】** をクリックします。
- Informatica Transports の以前のバージョンからアップグレードする場合、**【オリジナルを上書き】** オプションを選択します。

# SAP Mass Ingestion コネクタを使用した SAP ソースのガイドライン

SAP Mass Ingestion コネクタは、次のデータベースをサポートしています。

- SAP ECC (Oracle データベース)
- SAP S/4 HANA (トリガベース)

## SAP Mass Ingestion コネクタを使用した SAP ECC (Oracle データベース) ソースのガイドライン

SAP Mass Ingestion コネクタを使用した SAP ECC ソースを使用する場合は、次のガイドラインを考慮してください。

- SAP Mass Ingestion コネクタを使用した SAP ECC (Oracle データベース) ソースは、データベースビュー (初期ロード)、透過テーブル、プールテーブル、およびクラスターテーブルをサポートします。
- SAP Mass Ingestion コネクタを使用した SAP ECC ソースを初めて使用する場合は、ソースに対してアプリケーション取り込みタスクを設定する前に、次の手順を実行します。

1. アクセスする SAP マシンに、最新バージョンの SAP トランスポートファイルをインストールする必要があります。  
SAP トランスポートファイルへのアクセスの詳細については、Informatica グローバルカスタマサポートにお問い合わせください。

次の表は、SAP 開発システムでのトランスポートについて説明しています。

データおよび Co ファイル名	データおよび Co ファイル名	トランスポート要求	機能
R<number>.EP6 K<number>.EP6	必須	EP6K<number>	SAP 透過テーブル、クラスターテーブル、プールテーブル、およびビューから読み取るには、トランスポートをインストールします。

トランスポートファイルのインストールに関する詳細については、[「トランスポートファイルのインストール」 \(ページ 22\)](#)を参照してください。

2. 適切な SAP Notes が SAP サーバーで使用可能であることを確認します。詳細については、『SAP コネクタガイド』を参照してください。
  3. SAP Java Connector (SAP JCo) ライブラリを [SAP](#) Web サイトからダウンロードします。
  4. ダウンロードした SAP JCo ライブラリからネイティブファイルと JAR ファイルを次のディレクトリにコピーします。  
<Secure\_Agent>\ext\connectors\thirdparty\informatica.sapami\  
5. Secure Agent を再起動します。
- 一括取り込みアプリケーションは、次の廃止された SAP データ型のフィールドをプロパゲートしません。
    - D16S
    - D34S
    - PREC
    - VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型のフィールドをプロパゲートしません。

- SAP Mass Ingestion コネクタを使用した SAP ECC ソースは、スキーマドリフトをサポートしません。

## SAP Mass Ingestion コネクタを使用した SAP S/4 HANA ソースのガイドライン

SAP Mass Ingestion コネクタを使用した SAP S/4 HANA ソースを使用する場合は、次のガイドラインを考慮してください。

- SAP Mass Ingestion コネクタを使用した SAP S/4 HANA（トリガベース）ソースは、データベースビュー（初期ロード）と透過テーブルをサポートします。
- SAP Mass Ingestion コネクタを使用した SAP S/4 HANA ソースを初めて使用する場合は、ソースに対してアプリケーション取り込みタスクを設定する前に、次の手順を実行します。

1. アクセスする SAP マシンに、最新バージョンの SAP トランスポートファイルをインストールする必要があります。

SAP トランスポートファイルへのアクセスの詳細については、Informatica グローバルカスタマサポートにお問い合わせください。

次の表は、SAP 開発システムでのトランスポートについて説明しています。

データおよび Co ファイル名	データおよび Co ファイル名	トランスポート要求	機能
R<number>.EP6 K<number>.EP6	必須	EP6K<number>	SAP 透過テーブル、クラスターテーブル、プールテーブル、およびビューから読み取るには、トランスポートをインストールします。

トランスポートファイルのインストールに関する詳細については、[「トランスポートファイルのインストール」](#)（ページ 22）を参照してください。

2. 適切な SAP Notes が SAP サーバーで使用可能であることを確認します。詳細については、『SAP コネクタガイド』を参照してください。
3. SAP Java Connector（SAP JCo）ライブラリを [SAP](#) Web サイトからダウンロードします。
4. ダウンロードした SAP JCo ライブラリからネイティブファイルと JAR ファイルを次のディレクトリにコピーします。  
<Secure\_Agent>\ext\connectors\thirdparty\informatica.sapami\
  5. SAP Mass Ingestion コネクタは、JDBC を使用して SAP S/4 HANA に接続し、データとメタデータを読み取って、接続プロパティをテストします。SAP HANA JDBC ドライバファイル ngdbc.jar をダウンロードし、Secure Agent が実行されているマシンの Secure Agent インストールディレクトリの特定のサブディレクトリにコピーする必要があります。
    - a. SAP HANA JDBC ドライバの jar ファイル ngdbc.jar を、Secure Agent が実行されている Linux または Windows マシンにダウンロードします。  
ダウンロードするファイルが最新バージョンであることを確認します。ファイルのダウンロードで問題が発生した場合は、SAP カスタマサポートにお問い合わせください。
    - b. 次のディレクトリに ngdbc.jar ファイルをコピーします。  
<Secure Agent installation directory>\ext\connectors\thirdparty\informatica.hanami
    - c. Secure Agent を再起動します。
  6. 一括取り込みデータベースユーザーを作成します。管理者特権を持つユーザーとしてソースデータベースに接続し、次の文を実行します。  
CREATE USER dbmi\_user password "<password>" NO FORCE\_FIRST\_PASSWORD\_CHANGE;

この文は、デフォルトの権限を使用してデータベースにユーザーを作成します。これにより、基本的なデータディクショナリビューを読み取ることができ、必要な CDC オブジェクトをユーザー自身のスキーマに作成することができます。

7. SAP S/4 HANA ソースを含むデータベース取り込みタスクをデプロイして実行するには、ソース接続で、次のシステムビューからメタデータやその他の情報を読み取る特権を持つ一括取り込みデータベースユーザー (dbmi\_user) を指定する必要があります。

- SYS.M\_DATABASE - データベースのバージョンを取得するために使用されます。
- SYS.M\_CS\_PARTITIONS - テーブルがパーティション化されているかどうかを識別するために使用されます。(SAP HANA Cloud には適用されません)
- SYS.SCHEMAS - データベースのスキーマのリストを取得するために使用されます。
- SYS.TABLES - スキーマのテーブル名のリストを取得するために使用されます。
- SYS.TABLE\_COLUMNS - テーブルのカラムメタデータを取得するために使用されます。
- SYS.INDEXES - テーブルのインデックス情報を取得するために使用されます。
- SYS.INDEX\_COLUMNS - テーブルのインデックス情報を取得するために使用されます。

8. 増分ロードジョブの場合、次の権限を付与します。

- dbmi\_user の PKLOG およびシャドー\_CDC テーブルに行を書き込むトリガの場合は、dbmi\_user のスキーマに対する INSERT アクセス権を、ソーステーブルのスキーマを所有するユーザー (schema\_user) に付与します。

```
GRANT INSERT ON SCHEMA dbmi_user TO schema_user;
```

- トリガを使用して schema\_user のスキーマ内のソーステーブルから変更データをキャプチャするには、次のいずれかの文を実行します。

```
GRANT TRIGGER ON SCHEMA schema_user TO dbmi_user;
```

この文は、スキーマ内のすべてのテーブルに対するトリガアクセス権を付与します。

- or -

```
GRANT TRIGGER ON database.table_name TO dbmi_user;
```

この文は、特定のソーステーブルに対するトリガアクセス権を付与します。選択した少数のテーブルからデータをキャプチャする場合は、この文を使用します。CDC 対象のソーステーブルごとに、付与を繰り返します。

9. 初期ロードジョブの場合、ソーステーブルからデータを読み取るために、次の GRANT 文のいずれかを実行します。

```
GRANT SELECT ON SCHEMA schema_user TO dbmi_user;
```

この文は、スキーマ内のすべてのテーブルに対する SELECT アクセス権を付与します。

- or -

```
GRANT SELECT ON database.table_name TO dbmi_user;
```

この文は、特定のソーステーブルに対する SELECT アクセス権を付与します。データを読み取るソーステーブルごとに、この付与を繰り返します。

10. SAP S/4 HANA ソースの場合、暗号化するために、接続には SAP HANA JDBC カスタム接続プロパティ設定が必要です。SAP Mass Ingestion 接続プロパティの **【詳細接続プロパティ】** フィールドに次のプロパティを入力します。

```
encrypt=true&validateCertificate=false
```

- 一括取り込みアプリケーションは、次の廃止された SAP データ型のフィールドをプロパゲートしません。

- D16S

- D34S

- PREC
- VARC
- DATN
- TIMN
- DECFLOAT\_16
- DECFLOAT\_34
- UTCLONG
- STRG データ型（増分ロードの場合）

さらに、アプリケーション取り込みジョブは、RSTR データ型のフィールドをプロパゲートしません。

- SAP Mass Ingestion コネクタを使用した SAP ECC ソースは、スキーマドリフトをサポートしません。

SAP ユーザー権限の設定の詳細については、[「SAP ユーザー権限の設定」](#)（ページ 17）を参照してください。

## トランスポートファイルのインストール

Unicode SAP システムから読み取るために、Secure Agent ディレクトリの最新のトランスポートファイルをインストールします。トランスポートファイルは、SAP バージョン ECC 5.0 以降に対応します。

1. Secure Agent マシンの次のディレクトリに、トランスポートファイルがあります:  
`<Secure_Agent>\downloads\package-SAPAmi.<version>\package\sapami\SAPTableReader\`
2. アクセスする各 SAP マシンの SAP トランスポート管理ディレクトリにある Cofile ディレクトリに、cofile トランスポートファイルをコピーします。  
 cofile トランスポートファイルでは、次の命名規則を使用します: K<number>.EP6。
3. アクセスする各 SAP マシンの SAP トランスポート管理ディレクトリにある Data ディレクトリにデータトランスポートファイルをコピーします。  
 データトランスポートファイルでは、次の命名規則を使用します: R<number>.EP6。
4. STMS でトランスポートを SAP にインポートするには、**【補足】** > **【その他の依頼】** > **【追加】** をクリックし、トランスポート要求をシステムキューに追加します。
5. **【インポートキューに移送依頼追加】** ダイアログボックスに、cofile トランスポートの要求番号を入力します。  
 要求番号は、名前を変更した cofile を次のような順序に置き替えたものです: EP6K<number>  
 例えば、K900215.EP6 と名前変更された cofile トランスポートファイルの場合、要求番号として「EP6K900215」と入力します。
6. インポートキューの要求領域で、追加したトランスポート要求番号を選択して、**【インポート】** をクリックします。
7. Informatica Transports の以前のバージョンからアップグレードする場合、**【オリジナルを上書き】** オプションを選択します。

## ServiceNow ソースのガイドライン

ServiceNow ソースを使用する場合は、次のガイドラインを考慮してください。

- ServiceNow ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。

- 一部の ServiceNow オブジェクトに格納されているデータは、API クライアントが maint および nobody ロールで設定されている場合にのみ取得できます。一括取り込みアプリケーションは、ServiceNow 一括取り込み接続を maint および nobody ロールで設定する必要があるソースオブジェクトからデータを取得しません。
- ServiceNow ソース用に設定された増分ロードジョブは、sys\_updated\_on フィールドを含まないオブジェクトの変更データを取得しません。
- ServiceNow ソース用に設定された増分ロードジョブ、および初期ロードジョブと増分ロードジョブの組み合わせは、ソースフィールドの名前変更を検出およびレプリケートしません。
- 一括取り込みアプリケーションは、ソースフィールドのマスキング設定をターゲットで保持します。例えば、ソースのマスクされたフィールドに値 1234\*\*\*\*が含まれている場合、ターゲットの対応するフィールドにも値 1234\*\*\*\*が含まれます。
- Secure Agent 設定プロパティで、  
metadata\_manager.jvm.ai.servicenow.enable.cdc.using.inherited.columns および  
task\_container.jvm.ai.servicenow.enable.cdc.using.inherited.columns カスタムプロパティを **true** に設定し、親テーブルから CDC カラムを継承する子テーブルをタスクに含めます。
- ai.servicenow.enable.additional.datatypes カスタムプロパティを true に設定して、次の ServiceNow データ型を有効にします。
  - Audio
  - Basic Image
  - Encrypted Text
  - FX Currency
  - Image
  - Journal List
  - List
  - Name-Value Pairs
  - Reference
  - ビデオ
  - Wiki

**注: [Reference]** データ型については、ai.servicenow.enable.additional.datatypes カスタムプロパティが true に設定されている場合、columnName\_value、columnName\_display\_value、および columnName\_link 属性がターゲットで使用可能になります。ai.servicenow.enable.additional.datatypes カスタムプロパティが false に設定されている場合、ターゲットでは columnName 属性のみが使用可能になります。

## Workday ソースのガイドライン

Workday ソースを使用する場合は、次のガイドラインを考慮してください。

一括取り込みアプリケーションには、次の Web サービスを介して Workday データを抽出するオプションがあります。

- **Workday Web Services:** SOAP API を介して Workday データへのアクセスを実現します。
- **Workday Report-as-a-Service (RaaS) :** カスタムレポートを介して、カスタムオブジェクトおよびカスタムフィールドのデータへのアクセスを実現します。

必要な Web サービスを選択するためのオプションが、アプリケーション取り込みタスクウィザードの【ソース】タブに表示されます。

## Workday Web Services のガイドライン

- Workday Web Services を使用して、Workday Human Capital Management (HCM) のデータを取り込むことができます。アプリケーション取り込みタスクウィザードの【ソース】タブで、ターゲットにレプリケートする HCM サービスを選択できます。
- 一括取り込みアプリケーションは、名前が Get\_ で始まるソース操作のデータのみをレプリケートします。
- アプリケーション取り込みジョブは、XML 構造でソースデータを取得し、そのデータを JSON または XML 形式の単一オブジェクトとしてターゲットに書き込みます。アプリケーション取り込みジョブを設定するときに、タスクウィザードの【ソース】タブで、ターゲットデータの形式を指定できます。ターゲットでは、各レコードに次のフィールドが含まれます。
  - **WID:** レコードの一意の識別子またはプライマリキーを格納します。
  - **Data:** レコードの内容を JSON または XML 形式で格納します。
- 一括取り込みアプリケーションは、ソースフィールドの階層構造をターゲットで保持します。
- 一括取り込みアプリケーションは、カスタムオブジェクトに格納されているデータをプロパゲートしません。
- アプリケーション取り込みジョブが操作をターゲットに取り込むと、操作の名前が次の形式に変更されます。  
<Service\_Name>\_\_<Operation\_Name>
- Get\_Budget\_Pools 操作のデータは、Workday で初期ロードジョブでのレプリケートが許可されない場合があります。
- Workday ソースと Microsoft Azure Data Lake Storage Gen2 ターゲット用に設定された初期ロードジョブは、レコードが含まれていない次の操作に対してスキーマファイルを生成しません。
  - Benefits\_Administration\_\_Get\_ACA\_1095-C\_Forms\_Data
  - Compensation\_Review\_\_Get\_Budget\_Pools
  - Compensation\_Review\_\_Get\_Employee\_Awards
  - Recruiting\_\_Get\_Assess\_Candidate
  - Recruiting\_\_Get\_Interview\_Feedbacks
- Workday ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、アプリケーション取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- 増分ロードジョブは、ソースで実行された挿入と更新操作を自動的にキャプチャできます。ターゲットで更新/挿入操作を実行して、ソース データで実行された挿入と更新操作をレプリケートします。更新/挿入操作はソーススキーマに加えられたすべての変更を自動的にレプリケートするため、アプリケーション取り込みタスクウィザードは、Workday ソースのスキーマドリフトオプションを提供していません。
- 一括取り込みアプリケーションは、Workday の次のサービスおよび操作の変更データをキャプチャできます。

サービス	操作
Human_Resources	Get_Job_Profiles
	Get_Organizations
	Get_Workers



サービス	操作
Recruiting	Get_Evergreen_Requisitions
	Get_Job_Requisitions
	Get_Organizations
	Get_Positions
Staffing	Get_Organizations
	Get_Positions
	Get_Workers

## Workday RaaS のガイドライン

- Workday RaaS は、初期ロードタスクでのみ使用できます。
- アプリケーション取り込みタスクウィザードの【ソース】タブで、1 つ以上のカスタムレポートからデータを読み取ることを選択できます。複数のレポートを抽出する場合は、Secure Agent ホストで CSV ファイルを作成し、ファイルの最初のカラムにレポートの名前をリストします。
- Workday RaaS を使用するように設定されたアプリケーション取り込みジョブは、ソーススキーマに加えられた変更を検出できません。

## Zendesk ソースのガイドライン

Zendesk ソースを使用する場合は、次のガイドラインを考慮してください。

- Zendesk ソースの変更データをキャプチャするために増分ロードジョブを初めて実行するとき、ジョブは、関連付けられた取り込みタスクで指定されている日時以降に作成された変更レコードのみを取得してロードします。日付と時刻は、アプリケーション取り込みタスクウィザードでソースを設定するときに指定できます。ただし、停止または強制終了されたジョブを再開すると、ジョブは最後に中断したところから変更データのプロパゲートを開始します。
- Zendesk ソース用に構成された増分ロードジョブ、および初期ロードジョブと増分ロードジョブの組み合わせは、ソーススキーマに加えられた変更を検出およびレプリケートしません。
- ソースレコードに複数のカスタムフィールドが含まれている場合、アプリケーション取り込みジョブは、すべてのカスタムフィールドのデータを JSON オブジェクトとしてターゲットテーブルの単一のカラムに格納します。
- アプリケーション取り込みジョブは、次の Zendesk オブジェクトに存在するカスタムフィールドからデータを取得できます。
  - Organizations
  - Requests
  - Tickets
  - Users
- 初期ロードジョブは、次の Zendesk フィールドに保存されているデータの NULL 値をプロパゲートします。
  - Support Addresses オブジェクトの dns results フィールド。
  - Target Failures オブジェクトの raw request フィールドと raw response フィールド。
  - Satisfactory Ratings オブジェクトの reason code フィールド。

- Sharing Agreements オブジェクトの URL フィールド。
- 初期ロードジョブは、次の Zendesk オブジェクトに保存されているデータをプロパゲートしません。
  - Attachments
  - Channel Framework
  - Dynamic Content Item Variants
  - End users
  - Incremental Skill-based Routing
  - NPS<sup>®</sup> Invitations
  - NPS<sup>®</sup> Recipients
  - NPS<sup>®</sup> Responses
  - OAuth Tokens for Grant Types
  - Push Notification Devices
  - Search
  - Side Conversations
  - Side Conversation Attachment
  - Side Conversation Events
  - Skill-based Routing
  - Ticket Comments
  - Ticket Import
  - User Identities
  - User Passwords
- 一括取り込みアプリケーションは、次の Zendesk 標準オブジェクトの変更データを取得できます。
  - NPS Recipients
  - Organizations
  - Side Conversation Events
  - Tickets
  - Tickets Metric Events
  - Users
- 増分ロードジョブは、次のオブジェクトに保存されているレコードの削除を取得できます。
  - Organizations
  - Tickets
  - Users
- 一括取り込みアプリケーションは、タイプ Object のソースフィールドの階層構造をレプリケートしません。ターゲットテーブルでは、すべてのフィールドが同じ階層レベルにあります。アプリケーション取り込みジョブが、複数レベルの階層を持つオブジェクトフィールドをレプリケートする場合、ジョブは対応する同じ階層レベルのカラムを作成します。

# サポートされるターゲット

一括取り込みアプリケーションがサポートするターゲットは、アプリケーション取り込みタスクに指定されているソースによって異なります。

次の表に、一括取り込みアプリケーションが各ソースタイプに対してサポートするターゲットを示します。

ソースタイプ	サポートされるターゲットタイプ
Adobe Analytics	Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
Google Analytics	Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
Marketo	Amazon Redshift、Amazon S3、Apache Kafka（増分ロード）、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
Microsoft Dynamics 365	Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
NetSuite	Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
Oracle Fusion Cloud	<ul style="list-style-type: none"><li>- REST API の使用 - Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake</li><li>- BICC の使用 - Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake</li></ul>
Salesforce	Amazon Aurora PostgreSQL、Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Oracle、Microsoft Azure SQL Database、Microsoft Fabric OneLake、および Snowflake
Salesforce Marketing Cloud	Amazon Redshift、Amazon S3、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
SAP	<ul style="list-style-type: none"><li>- SAP ODP Extractor コネクタの使用 - Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake</li><li>- SAP Mass Ingestion コネクタの使用 - Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Azure SQL Database（初期ロード）、Microsoft Fabric OneLake、Oracle、および Snowflake</li></ul>

ソースタイプ	サポートされるターゲットタイプ
ServiceNow	Amazon Redshift、Apache Kafka、Databricks Delta、Amazon S3、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake
Workday	<ul style="list-style-type: none"> <li>- SOAP。Amazon Redshift、Amazon S3、Apache Kafka（増分ロード）、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake</li> <li>- RaaS。Amazon Redshift、Amazon S3、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake</li> </ul>
Zendesk	Amazon Redshift、Amazon S3、Apache Kafka、Databricks Delta、Google BigQuery、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、および Snowflake

ターゲットタイプに使用するコネクタを判断するには、「コネクタと接続」 > 「一括取り込みアプリケーションコネクタ」を参照してください。

## Amazon Redshift ターゲットのガイドライン

Amazon Redshift ターゲットを使用する場合は、次のガイドラインを考慮してください。

- Amazon Redshift ターゲットテーブルにデータを書き込む前に、アプリケーション取り込みジョブはデータを Amazon S3 バケットにステージングします。アプリケーション取り込みタスクを設定するときに、バケットの名前を指定する必要があります。取り込みジョブは COPY コマンドを使用して、Amazon S3 バケットから Amazon Redshift ターゲットテーブルにデータをロードします。COPY コマンドの詳細については、Amazon Web Services のドキュメントを参照してください。
- Amazon Redshift ターゲットの接続を定義するときに、アプリケーション取り込みジョブによってデータをステージングして Amazon Redshift ターゲットテーブルにロードする、Amazon S3 バケットのアクセスキーとシークレットアクセスキーを指定します。
- ソースから Amazon Redshift ターゲットにデータを取り込むときに、データソースに含まれているデータフィールドまたはデータカラムのうち、プライマリキーとして定義されているものが 32 を超える場合、アプリケーション取り込みジョブは失敗します。
- 増分ロードジョブと、初期ロードジョブと増分ロードジョブの組み合わせにより、ターゲット上に INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルが生成され、内部サービス情報が格納されます。リカバリテーブルのデータによって、障害後に再開されたジョブが以前に処理されたデータを再度ブローパゲートすることが防止されます。リカバリテーブルは、ターゲットテーブルのスキーマで生成されます。

## Amazon S3、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Fabric OneLake、および Oracle Cloud Object Storage ターゲットのガイドライン

Amazon S3、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Fabric OneLake、および Oracle Cloud Infrastructure (OCI) Object Storage ターゲットを使用する場合は、次のガイドラインを考慮してください。

- Microsoft Fabric OneLake は、初期ロードジョブでのみターゲットとして使用できます。

- Amazon S3、Google Cloud Storage、Microsoft Azure Data Lake Storage Gen2、Microsoft Fabric OneLake、または Oracle Cloud Object Storage ターゲットを使用するアプリケーション取り込みタスクを設定する場合は、ターゲットに適用するソースデータを含む出力ファイルの形式として CSV、Avro、Parquet のいずれかを選択できます。
- 出力ファイル形式として **【CSV】** を選択した場合、一括取り込みアプリケーションは、ソースフィールドごとに次のファイルをターゲットに作成します。
  - フィールドのスキーマを記述する schema.ini ファイル。このファイルには、ターゲットの出力ファイルに対するいくつかの設定も含まれています。
  - ソースフィールドに保存されているデータを含む出力ファイル。一括取り込みアプリケーションは、日付と時刻が追加されたソースフィールドの名前に基づいて、出力ファイルに名前を付けます。

schema.ini ファイルには、対応する出力ファイルの行の一連の列名が一覧表示されます。次の表で、schema.ini ファイルの列名について説明します。

カラム	説明
ColNameHeader	ソースデータファイルに列ヘッダーが含まれているかどうかを示します。
Format	出力ファイルの形式。一括取り込みアプリケーションはカンマ (,) を使用して列の値を区切ります。
CharacterSet	対応する出力ファイルに使用される文字セット。デフォルトでは、一括取り込みアプリケーションは UTF-8 文字セットでファイルを生成します。
COL<sequence_number>	<p>ソースフィールドの名前とデータ型。</p> <p><b>注意事項:</b></p> <ul style="list-style-type: none"> <li>- タスクウィザードの <b>【ターゲット】</b> ページの <b>【詳細】</b> で <b>【操作の追加...】</b> プロパティのいずれかを選択した場合、列名リストには、操作のタイプ、時間、所有者、またはトランザクション ID のメタデータ列が含まれます。</li> <li>- <b>【前のイメージを追加】</b> チェックボックスを選択した場合、ソース列ごとにジョブが UNDO データの <i>column_name_OLD</i> 列と REDO データの <i>column_name_NEW</i> 列を作成します。</li> </ul>

**注:** schema.ini ファイルを編集してはいけません。

- [Avro] 出力フォーマットを選択した場合、Avro 形式タイプ、ファイル圧縮タイプ、Avro データ圧縮タイプ、および各ソーステーブルに対して生成された Avro スキーマ定義を格納するディレクトリを選択できます。スキーマ定義ファイルの命名パターンは、スキーマ名\_テーブル名.txt です。
- Parquet 出力形式を選択した場合、必要に応じて、Parquet がサポートする圧縮タイプを選択できます。
- Microsoft Azure Data Lake Storage Gen2 および Microsoft Fabric OneLake ターゲット用に設定されたアプリケーション取り込みタスクの場合、一括取り込みアプリケーションは、空のソースフィールドごとに、ターゲットに空のディレクトリを作成します。
- Amazon S3 ターゲットについては、接続プロパティでアクセスキーと秘密鍵を指定しない場合、一括取り込みアプリケーションは DefaultAWSCredentialsProviderChain クラスによって実装されているデフォルトの資格情報プロバイダチェーンを使用して、AWS 資格情報を見つけようとします。詳細については、Amazon Web サービスのドキュメントを参照してください。

- CSV 出力形式を使用する、ターゲット用に設定された増分ロードジョブまたは初期ロードジョブと増分ロードジョブの組み合わせが、ソースのプライマリキーの値を変更した更新操作をプロパゲートする場合、ジョブは関連するターゲット行に対して削除操作を実行してから、同じ行に対して挿入操作を実行することで、ソースオブジェクトに加えられた変更をレプリケートします。削除操作は操作前のイメージをターゲットに書き込み、後続の挿入イメージは操作後のイメージをターゲットに書き込みます。

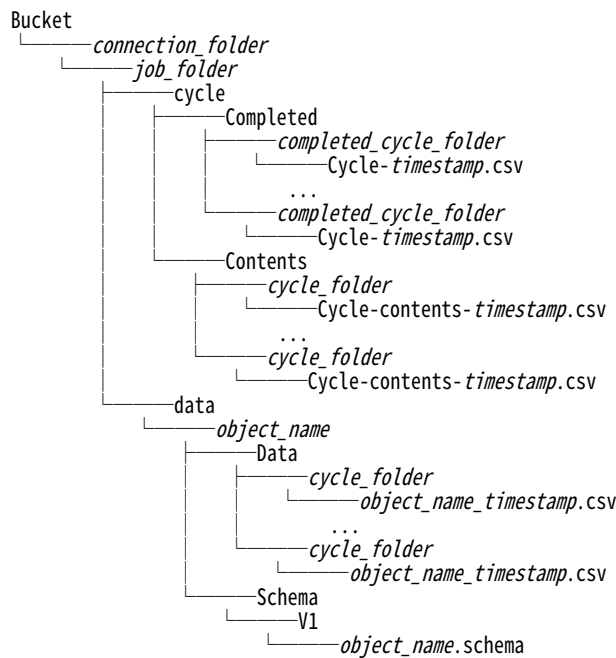
プライマリキー値を変更しない更新操作の場合、アプリケーション取り込みジョブは各更新操作を 1 つの操作として処理し、操作後のイメージのみをターゲットに書き込みます。

**注:** ソースオブジェクトにプライマリキーが含まれていない場合、一括取り込みアプリケーションは、オブジェクトのすべてのフィールドがプライマリキーの一部であると見なします。そのようなシナリオでは、一括取り込みアプリケーションはソースに対して実行された各更新操作を、ターゲットに対する削除操作とそれに続く挿入操作として処理します。

## Amazon S3、Google Cloud Storage、および Microsoft Azure Data Lake Storage Gen2 ターゲット上の CDC ファイルのデフォルトのディレクトリ構造

アプリケーション取り込みジョブは、Amazon S3、Google Cloud Storage、および Microsoft Azure Data Lake Storage Gen2 ターゲットにディレクトリを作成して、変更データ処理に関する情報を格納します。

次のディレクトリ構造がデフォルトでターゲットに作成されます。



次の表に、デフォルト構造に含まれるディレクトリを示します。

フォルダ	説明
<i>connection_folder</i>	一括取り込みアプリケーションのオブジェクトが含まれています。このフォルダは、Amazon S3 接続プロパティの【フォルダパス】フィールドまたは Microsoft Azure Data Lake Storage Gen2 接続プロパティの【ディレクトリパス】フィールドで指定されています。 <b>注:</b> このフォルダは、Google Cloud Storage ターゲット用には作成されていません。
<i>job_folder</i>	ジョブ出力ファイルが含まれています。このフォルダは、アプリケーション取り込みタスクウィザードの【ターゲット】ページの【ディレクトリ】フィールドで指定されています。
cycle/Completed	各完了 CDC サイクルのサブフォルダが含まれています。各サイクルサブフォルダには、完了サイクルファイルが含まれています。
cycle/Contents	各 CDC サイクルのサブフォルダが含まれます。各サイクルサブフォルダには、サイクルコンテンツファイルが含まれています。
data	各オブジェクトの出力データファイルとスキーマファイルが含まれています。
data/ <i>object_name</i> /Schema/V1	スキーマファイルが含まれています。 <b>注:</b> 出力ファイルが Parquet 形式を使用している場合、一括取り込みアプリケーションはスキーマファイルをこのフォルダに保存しません。
data/ <i>object_name</i> /Data	出力データファイルを生成する各 CDC サイクルのサブフォルダが含まれています。

## サイクルディレクトリ

一括取り込みアプリケーションは、次のパターンを使用してサイクルディレクトリに名前を付けます。

[dt=]yyyy-mm-dd-hh-mm-ss

アプリケーション取り込みタスクウィザードの【ターゲット】ページの【ディレクトリタグの追加】チェックボックスを選択すると、サイクルフォルダ名に「dt=」プレフィックスが追加されます。

## サイクルコンテンツファイル

サイクルコンテンツファイルは cycle/Contents/*cycle\_folder* サブディレクトリにあります。サイクルコンテンツファイルには、サイクル中に DML イベントが発生した各オブジェクトのレコードが含まれています。サイクル内のオブジェクトで DML 操作が発生しなかった場合、そのオブジェクトはサイクルコンテンツファイルに表示されません。

一括取り込みアプリケーションは、次のパターンを使用してサイクルコンテンツファイルに名前を付けます。

Cycle-contents-*timestamp*.csv

サイクルコンテンツの csv ファイルには、次の情報が含まれています。

- オブジェクト名
- サイクル名
- オブジェクトのサイクルフォルダへのパス
- オブジェクトの開始シーケンス
- オブジェクトの終了シーケンス
- 挿入操作の数

- 更新操作の数
- 削除操作の数
- **ロードジョブの組み合わせのみ。**初期ロードフェーズ中に発生した挿入操作の数
- **ロードジョブの組み合わせのみ。**初期ロードフェーズ中に発生した削除操作の数
- スキーマバージョン
- スキーマバージョンのスキーマファイルへのパス

**注:** 出力データファイルが Parquet 形式を使用している場合、一括取り込みアプリケーションは、サイクルコンテンツファイルで指定されたパスにスキーマファイルを保存しません。代わりに、アプリケーション取り込みタスクウィザードの【ターゲット】ページの【Avro スキーマディレクトリ】フィールドで指定されているフォルダ内のスキーマファイルを使用します。

## 完了サイクルファイル

完了サイクルファイルは `cycle/Completed/completed_cycle_folder` サブディレクトリにあります。アプリケーション取り込みジョブは、サイクルが完了した後、このサブディレクトリにサイクルファイルを作成します。このファイルが存在しない場合、サイクルはまだ完了していません。

一括取り込みアプリケーションは、次のパターンを使用して完了サイクルファイルに名前を付けます。

`Cycle-timestamp.csv`

完了サイクルの csv ファイルには、次の情報が含まれています。

- サイクル名
  - サイクル開始時刻
  - サイクル終了時刻
  - サイクル終了時の現在のシーケンス番号
  - サイクルコンテンツファイルへのパス
  - サイクル終了の理由
- 有効な理由の値は、以下のとおりです。

- **NORMAL\_COMMIT.** サイクルが DML 制限に達した後、またはサイクル間隔の終了後に、コミット操作が発生しました。サイクルはコミット境界でのみ終了できます。
- **NORMAL\_EXPIRY.** サイクル間隔が経過したため、サイクルが終了しました。最後の操作はコミットでした。
- **ロードジョブの組み合わせのみ:** BACKLOG\_COMPLETED。CDC バックログ処理が完了したため、サイクルが終了しました。CDC バックログは、ジョブの組み合わせの初期ロードフェーズ中にキャプチャされたイベントで構成されます。バックログには、初期ロードフェーズの開始時または終了時、および初期ロードフェーズからメインの CDC 増分処理への移行中にキャプチャされた可能性のある DML 変更が含まれます。
- **ロードジョブの組み合わせのみ:** INITIAL\_LOAD\_COMPLETED。初期ロードが完了したため、サイクルが終了しました。
- **ロードジョブの組み合わせのみ:** RESYNC\_STARTED。オブジェクトの再同期が開始されたため、サイクルが終了しました。

## 出力データファイル

データファイルには、次の情報を含むレコードが含まれています。

- 操作タイプ。有効な値は以下のとおりです。
  - I: 挿入操作
  - E: 更新操作



- D: 削除操作
- **ロードジョブの組み合わせのみ: X:** ロードジョブの組み合わせの初期ロードフェーズ中に発生した削除操作
- **ロードジョブの組み合わせのみ: Y:** ロードジョブの組み合わせの初期ロードフェーズ中に発生した挿入操作
- ソート可能なシーケンス番号。初期ロードジョブと増分ロードジョブの組み合わせでは、ソート可能なシーケンス番号には 20 桁のプレフィックスが含まれており、これを使用して行を再同期バージョンおよびロードジョブに合わせることができます。プレフィックスは次の属性を組み合わせたものです。
  1. インカネーション。この 9 桁の数字は、オブジェクトが再同期されるたびに増加します。初期値は 1 です。
  2. スキーマバージョン。この 9 桁の数字は、オブジェクトにスキーマドリフトの変更がプロパゲートされるたびに増加します。初期値は 1 です。
  3. フェーズ。この 2 桁の数字は、アンロードからバックログ、CDC への移行が実行されると変化します。有効な値は以下のとおりです。
    - 00: 切り詰め（初期ロードまたは再同期中に最初に書き込まれたデータレコード）
    - 01: 初期ロードまたは再同期中の通常の挿入
    - 02: 初期ロード中に検出された変更
    - 03: 初期ロードまたは再同期が完了した後、メインの CDC フェーズに戻る前に検出された変更
    - 04: 通常の CDC フェーズで検出された変更
- データフィールド
 

**注:** レコードの挿入と削除には、操作後のイメージのみが含まれています。レコードの更新には、更新前後のイメージが含まれています。

## Amazon S3、Google Cloud Storage、Microsoft Fabric OneLake、および ADLS Gen2 ターゲット上の出力ファイルのカスタムディレクトリ構造

デフォルトの構造を使用しない場合は、初期ロードジョブ、増分ロードジョブ、または初期ロードと増分ロードの組み合わせジョブが Amazon S3、Google Cloud Storage、および Microsoft Azure Data Lake Storage (ADLS) Gen2、および Microsoft Fabric OneLake ターゲットに書き込む出力ファイルのカスタムディレクトリ構造を設定できます。

### 初期ロード

デフォルトでは、初期ロードジョブは出力ファイルを親ディレクトリの下に *tablename\_timestamp* サブディレクトリに書き込みます。Amazon S3 および ADLS Gen2 のターゲットについては、タスクウィザードの【**ターゲット**】ページで【**親としての接続ディレクトリ**】チェックボックスが選択されている場合、親ディレクトリはターゲット接続プロパティで指定されます。

- Amazon S3 接続では、この親ディレクトリは【**フォルダパス**】フィールドで指定されます。
- ADLS Gen2 接続では、この親ディレクトリは【**ディレクトリパス**】フィールドで指定されます。

Google Cloud Storage ターゲットでは、親ディレクトリは、タスクウィザードの【**ターゲット**】ページにある【**バケット**】フィールドで指定されたバケットコンテナです。

Microsoft Fabric OneLake ターゲットの場合、親ディレクトリは、Microsoft Fabric OneLake 接続プロパティの【**レイクハウスのパス**】フィールドで指定されたパスです。

ニーズに合うようにディレクトリ構造をカスタマイズできます。例えば、初期ロードの場合、環境に合わせてファイルを整理したり、ファイルを見つけやすくしたりするために、ルートディレクトリや、接続プロパティで指定された親ディレクトリとは異なるディレクトリパスに出力ファイルを書き込むことができます。または、

すべてのファイルの自動処理を容易にするために、タイムスタンプ付きのサブディレクトリにファイルを別々に書き込むのではなく、テーブルのすべての出力ファイルをテーブル名の付いたディレクトリに直接統合できます。

ディレクトリ構造を設定するには、取り込みタスクウィザードの【ターゲット】ページにある【データディレクトリ】フィールドを使用する必要があります。デフォルト値は{TableName}\_{Timestamp}です。これにより、出力ファイルが親ディレクトリの下に *tablename\_timestamp* サブディレクトリに書き込まれます。大文字と小文字を区別しないプレースホルダとディレクトリ名の任意の組み合わせで構成されるディレクトリパターンを作成することにより、カスタムディレクトリパスを設定できます。プレースホルダは次のとおりです。

- ターゲットテーブル名は{TableName}
- 初期ロードジョブがターゲットへのデータの転送を開始した日付と時刻は{Timestamp} (yyyyMMdd\_hhmissms 形式)
- ターゲットスキーマ名は{Schema}
- 2桁の年は{YY}
- 4桁の年は{YYYY}
- 2桁の月の値は{MM}
- 月の2桁の日は{DD}

パターンには、次の関数を含めることもできます。

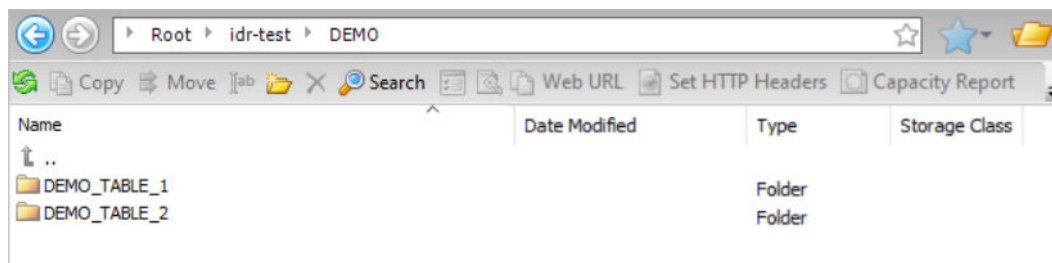
- toLower()は、カッコ内のプレースホルダで表される値を小文字にするのに使用します
- toUpper()は、カッコ内のプレースホルダで表される値を大文字にするのに使用します

デフォルトでは、ターゲットスキーマもデータディレクトリに書き込まれます。スキーマに別のディレクトリを使用する場合は、【スキーマディレクトリ】フィールドでディレクトリパターンを定義できます。

#### 例 1:

Amazon S3 ターゲットを使用していて、出力ファイルとターゲットスキーマを、接続プロパティの【フォルダパス】フィールドで指定された親ディレクトリの下にある同じディレクトリに書き込みたいとします。この場合、親ディレクトリは *idr-test/DEMO/* です。テーブルのすべての出力ファイルを、タイムスタンプなしでテーブル名と一致する名前を持つディレクトリに書き込みたいと考えています。【データディレクトリ】フィールドに入力し、【親としての接続ディレクトリ】チェックボックスを選択する必要があります。

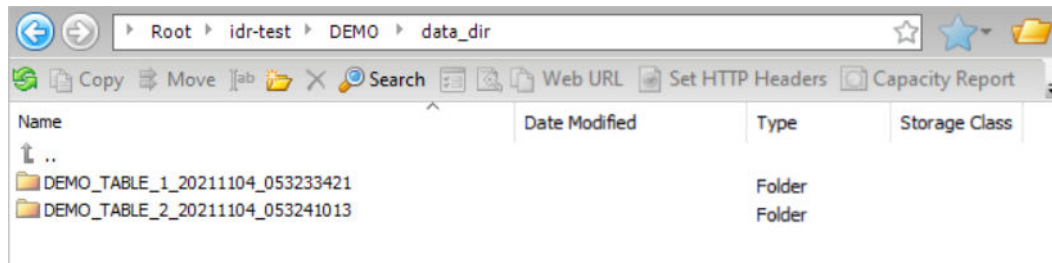
この設定に基づき、結果として得られるディレクトリ構造は次のようになります。



#### 例 2

Amazon S3 ターゲットを使用していて、出力データファイルをカスタムディレクトリパスに書き込み、ターゲットスキーマを別のディレクトリパスに書き込みたいとします。Amazon S3 接続プロパティの【フォルダパス】フィールドで指定されたディレクトリをデータディレクトリとスキーマディレクトリの親ディレクトリとして使用するには、【親としての接続ディレクトリ】を選択します。この場合、親ディレクトリは *idr-test/DEMO/* です。【データディレクトリ】フィールドと【スキーマディレクトリ】フィールドで、*data\_dir* や *schema\_dir* などの特定のディレクトリ名を使用して、その後にデフォルトの {TableName}\_{Timestamp} プレースホルダ値を指定することによって、ディレクトリパターンを定義します。プレースホルダは書き込み先の *tablename\_timestamp* ディレクトリを作成します。

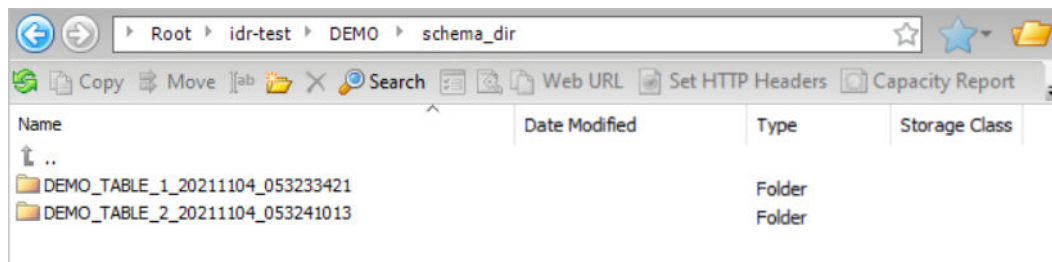
この設定に基づき、結果として得られるデータディレクトリ構造は次のようになります。



The screenshot shows a file browser window with the address bar set to 'Root > idr-test > DEMO > data\_dir'. The table below lists the contents of this directory.

Name	Date Modified	Type	Storage Class
..			
DEMO_TABLE_1_20211104_053233421		Folder	
DEMO_TABLE_2_20211104_053241013		Folder	

結果として得られるスキーマディレクトリ構造は次のようになります。



The screenshot shows a file browser window with the address bar set to 'Root > idr-test > DEMO > schema\_dir'. The table below lists the contents of this directory.

Name	Date Modified	Type	Storage Class
..			
DEMO_TABLE_1_20211104_053233421		Folder	
DEMO_TABLE_2_20211104_053241013		Folder	

### 増分ロード、および初期ロードと増分ロードの組み合わせ

デフォルトでは、増分ロードジョブ、および初期ロードジョブと増分ロードジョブの組み合わせは、サイクルファイルとデータファイルを親ディレクトリの下の子ディレクトリに書き込みます。ただし、カスタムディレクトリ構造を作成して、組織の要件に最も合うようにファイルを整理できます。

この機能は、Salesforce ソースおよび Amazon S3、Google Cloud Storage、または Microsoft Azure Data Lake Storage (ADLS) Gen2 ターゲットを持つアプリケーション取り込み増分ロードジョブに適用されます。

Google Cloud Storage を除くすべてのターゲットについては、タスクウィザードの **【ターゲット】** ページで **【親としての接続ディレクトリ】** チェックボックスがオンになっている場合、親ディレクトリはターゲット接続プロパティで設定されます。

- Amazon S3 接続では、親ディレクトリは **【フォルダパス】** フィールドで指定されます。
- ADLS Gen2 接続では、この親ディレクトリは **【ディレクトリパス】** フィールドで指定されます。

Google Cloud Storage ターゲットでは、親ディレクトリは、タスクウィザードの **【ターゲット】** ページにある **【バケット】** フィールドで指定されたバケットコンテナです。

ニーズに合うようにディレクトリ構造をカスタマイズできます。例えば、データファイルとサイクルファイルは、接続プロパティで指定された親ディレクトリではなく、タスクのターゲットディレクトリに書き込むことができます。または、1) テーブル名を含むサブディレクトリの下にテーブル固有のデータとスキーマファイルを統合する、2) CDC サイクルごとにデータファイルとサマリコンテンツおよび完了したファイルをパーティション化する、または 3) リテラル値とプレースホルダを含むパターンを定義することによって完全にカスタマイズされたディレクトリ構造を作成することができます。例えば、SQL タイプの式を実行して時間に基づいてデータを処理する場合は、CDC サイクルごとにパーティション化せずに、すべてのデータファイルをタイムスタンプサブディレクトリに直接書き込むことができます。

増分ロードタスクのカスタムディレクトリ構造を設定するには、取り込みタスクウィザードの **【ターゲット】** ページで、次のオプションフィールドのいずれかのパターンを定義します。

フィールド	説明	デフォルト
タスクターゲットディレクトリ	<p>増分ロードタスクの出力ファイルを保存するために使用するルートディレクトリの名前。</p> <p><b>【親としての接続ディレクトリ】</b> オプションを選択した場合でも、必要に応じてタスクターゲットディレクトリを指定できます。これは親ディレクトリに追加され、データ、スキーマ、サイクル完了、およびサイクルコンテンツディレクトリのルートを形成します。</p> <p>このフィールドは、次のディレクトリフィールドのいずれかのパターンで{TaskTargetDirectory}プレースホルダが指定されている場合は必須です。</p>	なし
親としての接続ディレクトリ	接続プロパティで指定された親ディレクトリを使用するには、このチェックボックスを選択します。	選択済み
データディレクトリ	<p>データファイルを含むサブディレクトリへのパス。</p> <p>データファイルとスキーマファイルが CDC サイクルごとにパーティション化されていない場合、ディレクトリパスでは、{TableName}プレースホルダが必要です。</p>	{TaskTargetDirectory}/data/{TableName}/data
スキーマディレクトリ	<p>スキーマファイルをデータディレクトリに保存しない場合は、スキーマファイルを保存するサブディレクトリへのパス。</p> <p>データファイルとスキーマファイルが CDC サイクルごとにパーティション化されていない場合、ディレクトリパスでは、{TableName}プレースホルダが必要です。</p>	{TaskTargetDirectory}/data/{TableName}/schema
サイクル完了ディレクトリ	サイクル完了ファイルが含まれているディレクトリへのパス。	{TaskTargetDirectory}/cycle/completed
サイクルコンテンツディレクトリ	サイクルコンテンツファイルが含まれているディレクトリへのパス。	{TaskTargetDirectory}/cycle/contents
データディレクトリにサイクルのパーティション化を使用してください	<p>各データディレクトリの下に、CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p> <p>このオプションが選択されていない場合、別のディレクトリ構造を定義しない限り、個々のデータファイルがタイムスタンプなしで同じディレクトリに書き込まれます。</p>	選択済み

フィールド	説明	デフォルト
サマリディレクトリにサイクルのパーティション化を使用してください	サマリコンテンツサブディレクトリおよび完了サブディレクトリの下に、CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。	選択済み
コンテンツ内の個々のファイルを一覧表示します	<p>コンテンツサブディレクトリの下にある個々のデータファイルを一覧表示します。</p> <p><b>【サマリディレクトリにサイクルのパーティション化を使用する】</b> がオフの場合、このオプションはデフォルトで選択されています。タイムスタンプや日付などのプレースホルダを使用してカスタムサブディレクトリを設定できる場合を除き、コンテンツサブディレクトリ内の個々のファイルがすべて一覧表示されます。</p> <p><b>【データディレクトリにサイクルのパーティション化を使用する】</b> が選択されている場合でも、必要に応じてこのチェックボックスを選択して、個々のファイルを一覧表示し、CDC サイクルごとにグループ化できます。</p>	<p><b>【サマリディレクトリにサイクルのパーティション化を使用する】</b> が選択されている場合は選択されません。</p> <p><b>【サマリディレクトリにサイクルのパーティション化を使用する】</b> をオフにした場合は選択されます。</p>

ディレクトリパターンは、中括弧{ }で示される大文字と小文字を区別しないプレースホルダと特定のディレクトリ名の任意の組み合わせで構成されます。次のプレースホルダがサポートされています。

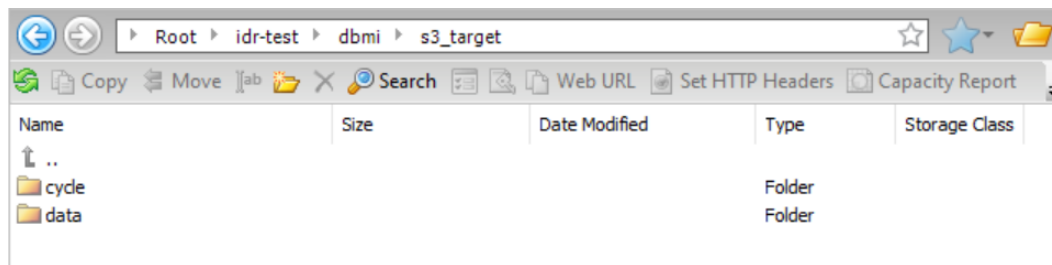
- 接続プロパティのディレクトリの代わりに使用するターゲット上のタスク固有のベースディレクトリを表す {TaskTargetDirectory}
- ターゲットテーブル名は{TableName}
- 日付と時刻を表す{Timestamp}、形式は yyyyymmdd\_hhmissms
- ターゲットスキーマ名は{Schema}
- 2桁の年は{YY}
- 4桁の年は{YYYY}
- 2桁の月の値は{MM}
- 月の2桁の日は{DD}

**注:** タイムスタンプ、年、月、および日のプレースホルダは、データ、コンテンツ、および完了ディレクトリのパターンで指定した場合は CDC サイクルの開始時点、スキーマディレクトリパターンで指定した場合は CDC ジョブの開始時点を表します。

#### 例 1:

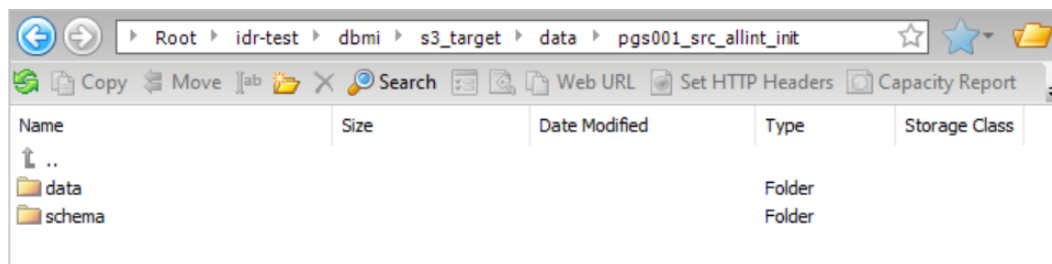
タスクウィザードに表示される増分ロードジョブのデフォルトのディレクトリ設定を使用するつもりだとします。ターゲットタイプは Amazon S3 です。**【親としての接続ディレクトリ】** チェックボックスがデフォルトで選択されているため、Amazon S3 接続プロパティの **【フォルダパス】** フィールドで指定されている親ディレクトリパスが使用されます。この親ディレクトリは idr-test/dbmi/ です。{TaskTargetDirectory}プレースホルダは、後続のディレクトリフィールドのデフォルトパターンで使用されるため、タスクターゲットディレクトリ名（この場合は s3\_target）も指定する必要があります。{TableName}プレースホルダはデフォルトのパターンに含まれているため、データディレクトリとスキーマディレクトリ内のファイルはテーブル名でグループ化されます。また、サイクルのパーティション化が有効になっているため、データディレクトリ、スキーマディレクトリ、およびサイクルサマリディレクトリ内のファイルは、CDC サイクルごとにさらに分割されます。

この設定に基づき、結果として得られるデータディレクトリ構造は次のようになります。



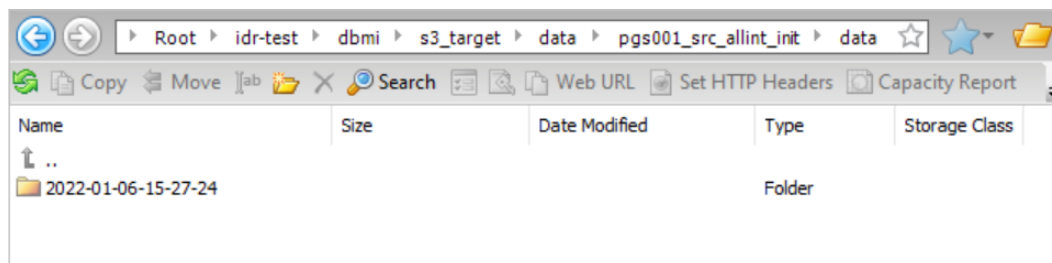
Name	Size	Date Modified	Type	Storage Class
↑ ..				
cycle			Folder	
data			Folder	

データフォルダをドリルダウンしてから、そのフォルダ内のテーブル（pgs001\_src\_allint\_init）をドリルダウンすると、データサブディレクトリとスキーマサブディレクトリにアクセスできます。



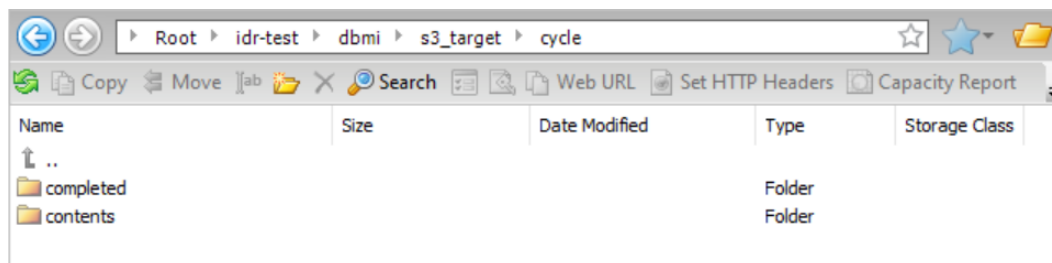
Name	Size	Date Modified	Type	Storage Class
↑ ..				
data			Folder	
schema			Folder	

データフォルダをドリルダウンすると、データファイルのタイムスタンプディレクトリにアクセスできます。



Name	Size	Date Modified	Type	Storage Class
↑ ..				
2022-01-06-15-27-24			Folder	

サイクルをドリルダウンすると、サマリコンテンツサブディレクトリと完了サブディレクトリにアクセスできます。



Name	Size	Date Modified	Type	Storage Class
↑ ..				
completed			Folder	
contents			Folder	

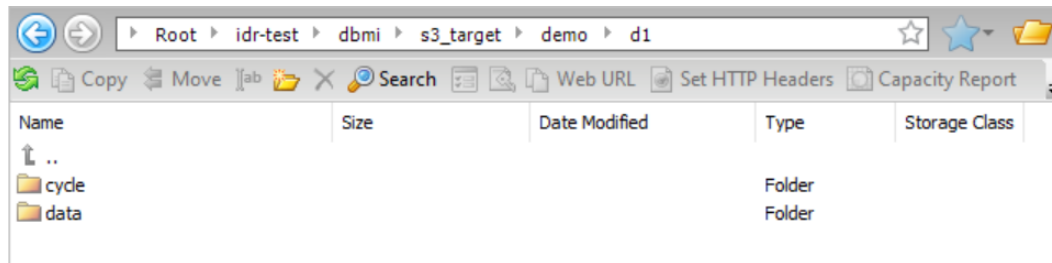
## 例 2

スキーマディレクトリを除くすべてのディレクトリパスにサブディレクトリ「demo」と「d1」を追加する増分ロードジョブ用のカスタムディレクトリ構造を作成して、デモ用のファイルを見つめられるようにしたいと考えています。[親としての接続ディレクトリ] チェックボックスが選択されているため、Amazon S3 接続プロパティの [フォルダパス] フィールドで指定されている親ディレクトリパス（idr-test/dbmi/）が使用されます。{TaskTargetDirectory} プレースホルダは、後続のディレクトリフィールドのパターンで使用されるため、タスクターゲットディレクトリも指定する必要があります。データディレクトリとスキーマディレクトリ内のファイルは、テーブル名でグループ化されます。また、サイクルのパーティション化が有効になってい



るため、データディレクトリ、スキーマディレクトリ、およびサイクルサマリディレクトリ内のファイルは、CDC サイクルごとにさらに分割されます。

この設定に基づき、結果として得られるデータディレクトリ構造は次のようになります。



## Apache Kafka ターゲットのガイドライン

Apache Kafka ターゲットを使用する場合は、次のガイドラインを考慮してください。

- 一括取り込みアプリケーションは、増分ロードジョブのターゲットとして Apache Kafka をサポートしません。

Kafka ターゲットタイプを指定するには、タスク定義または Kafka 接続プロパティで Kafka プロデューサプロパティを指定する必要があります。タスクのこれらのプロパティを指定するには、タスクウィザードの【ターゲット】ページの【プロデューサ設定プロパティ】フィールドに、*key.value* ペアのカンマ区切りのリストを入力します。Kafka 接続を使用するすべてのタスクのプロデューサプロパティを指定するには、プロパティのリストを接続プロパティの【追加接続プロパティ】フィールドに入力します。タスクレベルでプロデューサプロパティを定義することにより、特定のタスクの接続レベルのプロパティをオーバーライドできます。プロデューサプロパティの詳細については、Apache Kafka のドキュメントを参照してください。

- 【AVRO】を Kafka ターゲットの出力形式として選択した場合、一括取り込みアプリケーションは次の形式の名前で、各テーブルのスキーマ定義ファイルを生成します。

*schemaname\_tablename.txt*

ソーススキーマの変更により、増分ロードジョブのターゲットが変更されることが予想される場合は、一括取り込みアプリケーションがタイムスタンプを含む一意の名前で Avro スキーマ定義ファイルを再生成します。

*schemaname\_tablename\_YYYYMMDDhhmmss.txt*

この一意の命名パターンにより、古いスキーマ定義ファイルが監査目的で保持されます。

- Kafka プロデューサプロパティをタスクウィザードの【ターゲット】ページの【プロデューサ設定プロパティ】フィールド、または Kafka 接続プロパティの【追加接続プロパティ】フィールドいずれかで指定できます。ビジネスニーズに合うように、Kafka ベンダーによってサポートされている *property=value* のペアを入力します。

## Databricks Delta ターゲットのガイドライン

Databricks Delta ターゲットを使用する場合は、次のガイドラインを考慮してください。

- Databricks Delta ターゲットを初めて使用する場合は、ターゲットに対してアプリケーション取り込みタスクを設定する前に、次の手順を実行します。

- Databricks JDBC ドライバダウンロード Web サイトから DatabricksJDBC42 ドライババージョン 2.6.25 をダウンロードします。

- 次のディレクトリに DatabricksJDBC42.jar ファイルをコピーします。

*Secure\_Agent\_installation\_directory/apps/Database\_Ingestion/ext/*

3. Databricks Delta 接続プロパティで、**[JDBC ドライバクラス名]** プロパティを `com.databricks.client.jdbc.Driver` に設定します。
  4. Windows では、Secure Agent が実行されているコンピュータに Visual Studio2013 用の Visual C++ 再配布可能パッケージをインストールします。
- 増分ロードジョブの場合、すべてのソースフィールドに対して変更データキャプチャ（Change Data Capture: CDC）を有効にする必要があります。
  - 次のストレージタイプの上に作成された Databricks Delta テーブルにアクセスできます。
    - Microsoft Azure Data Lake Storage (ADLS) Gen2
    - Amazon Web Services (AWS) S3

Databricks Delta 接続は、JDBC URL を使用して Databricks クラスタに接続します。ターゲットを設定するときは、クラスタへの接続に使用する JDBC URL と資格情報を指定します。また、ターゲットが Amazon S3 または ADLS Gen2 のステージングロケーションに接続するために使用する接続情報を定義します。

- Databricks Delta ターゲットテーブルにデータを書き込む前に、アプリケーション取り込みジョブはデータを Amazon S3 バケットまたは ADLS ディレクトリにステージングします。アプリケーション取り込みタスクを設定するときに、データのディレクトリを指定する必要があります。

**注:** 一括取り込みアプリケーションは、ディレクトリを決定する際に Databricks Delta 接続プロパティの **[ADLS ステージングファイルシステム名]** と **[S3 ステージングバケット]** プロパティは使用しません。

- 一括取り込みアプリケーションは、1 回だけ実行されるジョブを使用して、AWS S3 または ADLS Gen2 のステージングファイルから外部テーブルにデータをロードします。  
デフォルトでは、一括取り込みアプリケーションは、Databricks Delta 接続プロパティで指定されたクラスタでジョブを実行します。別のクラスタでジョブを実行する場合は、アプリケーション取り込みタスクウィザードの **[ターゲット]** ページで `dbDeltaUseExistingCluster` カスタムプロパティを `FALSE` に設定します。
- Databricks Delta 接続プロパティで指定されたクラスタが稼働していない場合、アプリケーション取り込みジョブはクラスタが開始するまで待機します。デフォルトでは、ジョブは 10 分間待機します。クラスタが 10 分以内に開始されない場合、接続がタイムアウトし、ジョブのデプロイが失敗します。  
接続のタイムアウト値を増やす場合は、`dbClusterStartWaitingTime` カスタムプロパティを、クラスタが起動して実行されるまで取り込みジョブが待機する必要がある最大時間（秒単位）に設定します。カスタムプロパティはアプリケーション取り込みタスクウィザードの **[ターゲット]** ページで設定できます。
- デフォルトでは、一括取り込みアプリケーションは、Databricks Delta の COPY INTO 機能を使用して、ステージングファイルから Databricks Delta ターゲットテーブルにデータをロードします。アプリケーション取り込みタスクウィザードの **[ターゲット]** ページで `writerDatabricksUseSqlLoad` カスタムプロパティを `FALSE` に設定すると、すべてのロードタイプでこれを無効にできます。
- AWS クラスタを使用する場合は、Databricks Delta 接続プロパティの **[S3 サービスリージョナルエンドポイント]** の値を指定する必要があります。以下に例を示します。  
`s3.us-east-2.amazonaws.com`  
Databricks Delta 接続をテストする前には、Databricks Delta 接続プロパティの **[SQL エンドポイント JDBC URL]** フィールドで JDBC URL を指定する必要があります。接続をテストしたら、**[SQL エンドポイント JDBC URL]** の値を削除します。そうしないと、接続を使用するアプリケーション取り込みタスクを定義するときに設計時エラーが発生します。これは、一括取り込みアプリケーションが、JDBC URL の他に、必須の **[Databricks ホスト]**、**[クラスタ ID]**、**[組織 ID]**、**[Databricks トークン]** の値を使用してターゲットに接続しようとし、ログインエラーを引き起こすためです。
- 基になる Parquet ファイルを書き換える必要なく、Databricks Delta ターゲットテーブルに対してフィールド名の変更操作を処理するには、Databricks Runtime 10.2 以降の Databricks Delta カラムマッピング機能が必要です。タスクウィザードの **[スケジュールおよびランタイムオプション]** ページで **[フィールド名の変更]** オプションを **[レプリケート]** に設定した場合、タスクのデプロイ後、ジョブを実行する前に、



生成されたターゲットテーブルを変更して、次の Databricks テーブルプロパティを設定する必要があります。

```
ALTER TABLE <target_table> SET TBLPROPERTIES (  
  'delta.columnMapping.mode' = 'name',  
  'delta.minReaderVersion' = '2',  
  'delta.minWriterVersion' = '5')
```

これらのプロパティにより、必要なリーダーとライターのバージョンの Databricks Delta カラムマッピング機能が有効になります。これらのプロパティを設定しない場合、アプリケーション取り込みジョブは失敗します。

- Databricks Delta ターゲットを含むアプリケーション取り込みジョブは、Databricks Unity Catalog からターゲットテーブルを生成するためのスキーマ情報を取得できます。Unity Catalog 内の情報へのアクセスを有効にするには、Databricks Delta 接続プロパティの **【カタログ名】** フィールドにカタログ名を指定します。カタログ名は、データウェアハウスの **【SQL ウェアハウス JDBC URL】** 値に追加されます。

**注:** カatalogの使用は SQL ウェアハウスの場合はオプションであり、ジョブクラスタには適用されません。

Unity Catalog を使用する場合は、個人用ストレージの場所が自動的にプロビジョニングされます。個人用のステージング場所を使用するには、接続プロパティの **【ステージング環境】** フィールドで **【個人用のステージング場所】** を選択します。その後、取り込みジョブの Parquet データファイルをローカルの個人用ストレージの場所にステージングできます。データ保持期間は 7 日間です。デフォルトでは、ステージング場所は AWS または Azure のルートの場所 `stage://tmp/<user_name>` です。<user\_name>は、**【データベーストークン】** 接続プロパティから取得されます。このユーザーには、個人用のステージング場所に対する読み取りおよび書き込みアクセス権が必要です。

## Google BigQuery ターゲットのガイドライン

Google BigQuery ターゲットを使用する場合は、次のガイドラインを考慮してください。

- Google BigQuery ターゲットを初めて使用する場合は、ターゲットに対してアプリケーション取り込みタスクを設定する前に、次の手順を実行します。
  1. Google BigQuery JDBC ドライババージョン 1.2.25.1029 を [Google Cloud](#) の Web サイトからダウンロードします。
  2. JDBC ドライバ jar ファイルを次のディレクトリにコピーします。  
`<Secure_Agent_installation_directory>/apps/Database_Ingestion/ext/`
  3. Secure Agent を再起動します。
- Google BigQuery と Google Cloud Storage にアクセスするためのサービスアカウントが Google アカウントにある必要があります。
- サービスアカウントに `client_email`、`project_id`、`private_key`、および `regionID` の値があることを確認してください。Google BigQuery 接続を作成するときに、対応する **【サービスアカウント ID】**、**【プロジェクト ID】**、**【サービスアカウントキー】**、および **【リージョン ID】** 接続プロパティに値を入力します。
- Google BigQuery 接続のタイムアウト間隔を設定する場合は、接続プロパティの **【オプションのプロパティを指定】** フィールドでタイムアウト間隔プロパティを指定します。次の形式を使用します。  
`"timeout": "<timeout_interval_in_seconds>"`
- 次のエンティティへの読み取りおよび書き込みアクセスが必要です。
  - ターゲットテーブルを含む Google BigQuery データセット。
  - 一括取り込みアプリケーションがステージングファイルを作成する Google Cloud Storage バス。
- Google BigQuery ターゲット用に設定されたアプリケーション取り込みジョブは、ソースフィールドの変更と名前変更をターゲットにレプリケートしません。
- 増分ロードタスクの場合、すべてのソースフィールドに対してソースデータベース変更データキャプチャ (Change Data Capture: CDC) を有効にする必要があります。

- Google BigQuery テーブルにデータを書き込むには、次の権限が必要です。
  - bigquery.datasets.get
  - bigquery.datasets.getIamPolicy
  - bigquery.models.\*
  - bigquery.routines.\*
  - bigquery.tables.create
  - bigquery.tables.delete
  - bigquery.tables.export
  - bigquery.tables.get
  - bigquery.tables.getData
  - bigquery.tables.list
  - bigquery.tables.update
  - bigquery.tables.updateData
  - bigquery.tables.updateTag
  - resourceManager.projects.get
  - resourceManager.projects.list
  - bigquery.jobs.create
- アプリケーション取り込みジョブをデプロイすると、一括取り込みアプリケーションはデフォルトでは、プライマリキーカラムまたは一意のキーカラムでクラスタ化された Google BigQuery ターゲットテーブルを生成します。各キーカラムには、Google BigQuery がクラスタリングでサポートする次のいずれかのデータ型が必要です。
  - STRING
  - INT64
  - NUMERIC
  - BIGNUMERIC
  - DATE
  - DATETIME
  - TIMESTAMP
  - BOOL
  - GEOGRAPHY

プライマリキーまたは一意のキーのカラムにサポートされていないデータ型がある場合、クラスタリング時にそのカラムはスキップされます。例えば、プライマリキーに C1、C2、C3、C4、C5 カラムが含まれていて、C2 にサポートされていないデータ型がある場合、ターゲットテーブルは CLUSTER BY 句に C1、C3、C4、および C5 カラムを使用して作成されます。

## Microsoft Azure Synapse Analytics ターゲットのガイドライン

Microsoft Azure Synapse Analytics ターゲットを使用する場合は、次のガイドラインを考慮してください。

- Microsoft Azure Synapse Analytics ターゲットを使用してアプリケーション取り込みタスクをデプロイして実行するには、ターゲット接続でターゲットデータベースに対する CONTROL 権限を持つデータベース

ユーザーを指定する必要があります。ユーザーに CONTROL 権限を付与するには、次の SQL 文を使用します。

```
USE database_name;  
GRANT CONTROL TO user_name;
```

CONTROL 権限は、初期ロード、増分ロード、および初期ロードと増分ロードの組み合わせジョブに必要です。この権限により、一括取り込みアプリケーションは、ターゲットテーブルと、外部データソース、外部ファイル形式、データベーススコープの資格情報オブジェクト（データベースに存在しない場合）などのデータベースオブジェクトを作成できます。CONTROL 権限は、外部データソースおよびデータベーススコープの資格情報オブジェクトを作成するために特に必要です。

**注:** マスターキーを手動で作成する必要があります。マスターキーを作成するには、データベースに対する CONTROL 権限が必要です。

- アプリケーション取り込みジョブは、データを Microsoft Azure Synapse Analytics ターゲットテーブルに書き込む前に、まず Microsoft Azure Data Lake Storage Gen2 ステージングファイルにデータを送信します。ステージングファイルは、フィールド区切り文字として 16 進数の x1d 区切り文字を使用します。データがターゲットに書き込まれた後、ステージングファイルを含むテーブル固有のディレクトリに保存されているデータが削除されます。
- Microsoft Azure Synapse Analytics 接続で Microsoft Azure Data Lake Storage Gen2 を使用する場合は、Microsoft Azure Data Lake Storage の【階層型名前空間】オプションを有効にする必要があります。この設定では、blob ストレージは推奨されません。
- Microsoft Azure Synapse Analytics ターゲットのアプリケーション取り込みタスクを構成するときに、レプリケーションの対象として選択した各ソースオブジェクトが次の条件を満たしていることを確認します。
  - オブジェクトには 1024 を超えるフィールドが含まれていてはならず、各フィールドのサイズは 500 KB 未満である必要があります。
  - オブジェクトには、サイズが 1 MB を超えるレコードが含まれていてはいけません。
  - オブジェクトには、32 を超えるプライマリキーが含まれていてはいけません。
  - クラスティンデックスキーには、サイズが 900 バイトを超えるデータを含めることはできません。
  - オブジェクトのプライマリキーは、Microsoft Azure Synapse Analytics がプライマリキーに対してサポートするデータ型である必要があります。
- 増分ロードジョブと、初期ロードジョブと増分ロードジョブの組み合わせにより、ターゲット上に INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルが生成され、内部サービス情報が格納されます。リカバリテーブルのデータによって、障害後に再開されたジョブが以前に処理されたデータを再度プロパゲートすることが防止されます。リカバリテーブルは、ターゲットテーブルのスキーマで生成されます。
- アプリケーション取り込みジョブが外部テーブルを使用して Microsoft Azure Synapse Analytics ターゲットにデータをロードすると、ジョブの再開時にこれらのテーブルが再作成される場合でも、ジョブはターゲット上に作成されたログテーブルと外部テーブルを削除しません。
- Microsoft Azure Synapse Analytics ターゲット用に設定されたアプリケーション取り込みジョブは、ソースフィールドの名前変更をターゲットにレプリケートしません。

## Microsoft Azure SQL Database ターゲットのガイドライン

SAP（初期ロードでのみ SAP Mass Ingestion コネクタを使用）ソースと Salesforce ソース、および Microsoft Azure SQL Database ターゲットを、Microsoft Amazon RDS（Relational Database Service）for SQL Server、Microsoft Azure SQL Database、Azure SQL Managed Instance などのインスタンスのすべてのロードタイプに使用する場合は、次のガイドラインを考慮してください。

- SQL Server JDBC ドライバは、一括取り込みアプリケーションとともに提供されます。個別にインストールする必要はありません。

- 一括取り込みアプリケーションユーザーには、ターゲットテーブルの作成とテーブルへのデータの書き込みを行うため、少なくとも次のデータベースロールが必要です。
  - db\_datareader
  - db\_datawriter
  - db\_ddladmin
- SQL Server ターゲットに接続するために Administrator で SQL Server 接続を定義する場合は、Administrator で必要なプロパティのみを入力します。
  - SQL Server のバージョン。[SQL Server 2017] または [SQL Server 2019] を選択します。
  - 認証モード。[SQL Server 認証] または [Windows 認証 v2] を選択します。
  - ユーザー名
  - パスワード
  - ホスト
  - ポート
  - データベース名
  - スキーマ
  - コードページ

他のプロパティはサポートされていません。

- SQL Server ターゲットを持つアプリケーション取り込み増分ロードジョブ、初期および増分ロードジョブは、ターゲットテーブルスキーマに基づいて、いくつかの追加のメタデータカラムとともに LOG テーブルを生成します。LOG テーブルは、変更データがターゲットにフラッシュされる直前に作成されます。受信 DML データは、ローカル CSV ファイルを SQL Server ドライバの Bulk Copy API に提供することで LOG テーブルに挿入されます。LOG テーブルの情報に基づいて merge apply 文が生成され、DML 操作が実際のターゲットテーブルに適用されます。この処理が完了すると、LOG テーブルは削除されます。複数のジョブ、または複数のテーブルを含むジョブを実行している場合、LOG テーブルによって、顧客データベースインスタンスの追加スペースまたはサイズ要件が一時的に急増する可能性があります。LOG テーブルに必要なスペースとサイズは、フラッシュサイクルの一部として受信される行数によって異なります。
- アプリケーション取り込み増分ロードジョブ、または初期ロードおよび増分ロードジョブが SQL Server ターゲットにプロパゲートできるソースオブジェクト内のフィールド数は、508 フィールドを超えることはできません。ソースオブジェクトに 508 を超えるフィールドが含まれている場合、ジョブは LOG テーブルの作成中に失敗します。
- アプリケーション取り込み増分ロードジョブ、および初期ロードと増分ロードの組み合わせジョブの SQL Server ターゲットには、スキーマドリフトオプションはサポートされていません。
- アプリケーション取り込み増分ロードジョブにより、ターゲット上に INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルが生成され、障害後に再開されたジョブが以前に処理されたデータを再度プロパゲートするのを防ぐ内部サービス情報が格納されます。このリカバリテーブルは、ターゲットテーブルと同じスキーマで生成されます。

## Oracle ターゲットのガイドライン

Oracle ターゲットを使用する場合は、次のガイドラインを考慮してください。

- アプリケーション取り込みジョブで Oracle ターゲットを使用して、Salesforce ソースからデータをレプリケートできます。
- デフォルトでは、一括取り込みアプリケーションではパフォーマンスを最適化するために Oracle ターゲットテーブルのログが無効になっています。ログを有効にするには、アプリケーション取り込みタスクウィザードの **【ターゲット】** ページで writerOracleNoLogging カスタムプロパティを false に設定します。

## PostgreSQL ターゲットのガイドライン

PostgreSQL ターゲットを使用する場合は、次のガイドラインを考慮してください。

- Amazon Aurora PostgreSQL は、Salesforce ソースを持つ初期ロードジョブ、増分ロードジョブ、および初期ロードと増分ロードの組み合わせジョブでターゲットとして使用できます。
- PostgreSQL ターゲットを含むアプリケーション取り込みタスクをデプロイして実行するには、ターゲット接続に必要な特権を持つデータベースユーザーを指定する必要があります。  
ユーザーには、接続で指定されたデータベースに対する CONNECT 特権と TEMPORARY 特権、およびターゲットプロパティで指定されたターゲットスキーマに対する USAGE 特権と CREATE 特権が必要です。

次の SQL 文を使用して、これらの特権をユーザーロールに付与し、そのロールをユーザーに割り当てます。

```
CREATE ROLE dbmi_role;  
GRANT CONNECT ON DATABASE database TO dbmi_role;  
GRANT TEMPORARY ON DATABASE database TO dbmi_role;  
GRANT CREATE ON SCHEMA schema TO dbmi_role;  
GRANT USAGE ON SCHEMA schema TO dbmi_role;  
CREATE USER dbmi_user with PASSWORD 'password';  
GRANT dbmi_role to dbmi_user;
```

**注:** ジョブの実行時に生成されるターゲットテーブルに対する特権は、ジョブを実行するユーザーに付与されます。

- PostgreSQL ターゲットを使用したアプリケーション取り込み増分ロードジョブでは、ターゲットテーブルスキーマに基づいて LOG テーブルが生成され、いくつかのメタデータカラムが追加されます。LOG テーブルは、変更データがターゲットにフラッシュされる直前に作成されます。受信 DML データは、ローカル CSV ファイルを PostgreSQL ドライバの Bulk Copy API に提供することによって LOG テーブルに挿入されます。LOG テーブルの情報に基づいて一連の merge apply 文が生成され、DML 操作が実際のターゲットテーブルに適用されます。DML の変更が適用されると、LOG テーブルは削除されます。  
複数のジョブ、または複数のテーブルを含むジョブを実行している場合、LOG テーブルによって、顧客データベースインスタンスの追加スペースまたはサイズ要件が一時的に急増する可能性があります。LOG テーブルに必要なスペースとサイズは、フラッシュサイクルの一部として受信される行数によって異なります。
- アプリケーション取り込み増分ロードジョブが PostgreSQL ターゲットにプロパゲートできるソースオブジェクト内のフィールド数は、796 フィールドを超えることはできません。ソースオブジェクトに 796 を超えるフィールドが含まれている場合、ジョブは LOG テーブルの作成中に失敗します。
- PostgreSQL でサポートするソースオブジェクト識別子の最大長は 63 文字です。PostgreSQL ターゲットを持つアプリケーション取り込みタスクのデプロイは、ソーステーブル名またはカラム名の長さが 63 文字を超えると、検証中に失敗します。
- スキーマドリフトオプションは、アプリケーション取り込みジョブの PostgreSQL ターゲットではサポートされていません。
- アプリケーション取り込み増分ロードジョブにより、ターゲット上に INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルが生成され、障害後に再開されたジョブが以前に処理されたデータを再度プロパゲートするのを防ぐ内部サービス情報が格納されます。このリカバリテーブルは、ターゲットテーブルと同じスキーマで生成されます。

## Snowflake ターゲットのガイドライン

### ターゲットの準備

ターゲットの準備は、Snowflake ターゲットテーブルへのデータの高パフォーマンスストリーミングに Superpipe 機能を使用するか、中間ステージファイルにデータを書き込むかによって異なります。

### Superpipe を使用

Superpipe 機能を使用する場合は、次の手順を実行します。

1. 一括取り込みユーザーを作成します。次の SQL 文を使用します。  

```
create user INFACMI_User password 'Xxxx@xxx';
```
2. 新しいユーザーロールを作成し、一括取り込みユーザーに付与します。以下の SQL 文を使用します。  

```
create role INFACMI_superpipe;  
grant role INFACMI_superpipe to user INFACMI_User;
```
3. Snowflake 仮想ウェアハウスの使用権限を新しいロールに付与します。次の SQL 文を使用します。  

```
grant usage on warehouse warehouse_name to role INFACMI_superpipe;
```
4. Snowflake データベースの使用権限を新しいロールに付与します。次の SQL 文を使用します。  

```
grant usage on database INFACMI_DB1 to role INFACMI_superpipe;
```
5. 新しいスキーマを作成します。以下の SQL 文を使用します。  

```
use database INFACMI_DB1;  
create schema sh_superpipe;
```
6. 新しい Snowflake スキーマに対する create stream、create view、および create table 特権を新しいロールに付与します。次の SQL 文を使用します。  

```
grant create stream, create view, create table, usage on schema INFACMI_DB1.sh_superpipe to role INFACMI_superpipe;
```
7. 新しく作成されたユーザーのデフォルトのロールを設定します。次の SQL 文を使用します。  

```
alter user INFACMI_User set default_role=INFACMI_superpipe;
```
8. ターゲットへの Snowflake Data Cloud 接続を定義します。認証方法として **[KeyPair]** オプションを使用する必要があります。「コネクタと接続」>「Snowflake Data Cloud 接続プロパティ」を参照してください。
9. OpenSSL バージョン 3.x.x と PBE-SHA1-2DES または PBE-SHA1-3DES 暗号を使用してプライベートキーを生成します。次の openssl コマンドを使用し、プライベートキーを生成してフォーマットします。  

```
openssl genrsa 2048 | openssl pkcs8 -topk8 -inform PEM -v1 PBE-SHA1-3DES -out rsa_key.p8
```
10. パブリックキーを生成します。次の openssl コマンドを使用し、-in オプションで暗号化されたプライベートキーを含むファイル (rsa\_key.p8) を参照します。  

```
openssl rsa -in rsa_key.p8 -pubout -out rsa_key.pub
```
11. Snowflake で、パブリックキーを Snowflake ユーザーに割り当てます。次の SQL コマンドを使用します。  

```
alter user INFACMI_User set rsa_public_key='key_value';
```

**次の手順:** 取り込みタスクを作成するときに、タスクウィザードの **[ターゲット]** ページで **[Superpipe]** オプションを選択します。必要に応じて、変更データ行がマージされて Snowflake ターゲットテーブルに適用される頻度を制御する **[マージ頻度]** 値を指定することもできます。

### Superpipe を使用しない

Snowflake ターゲットに Superpipe 機能を使用しない場合は、ACCOUNTADMIN ユーザーとして次の手順を実行します。

1. 一括取り込みユーザーを作成します。次の SQL 文のいずれかを使用します。  

```
create user INFACMI_User password 'Xxxx@xxx';
```

  
または  

```
replace user INFACMI_User password 'Xxxx@xxx';
```
2. 新しいロールを作成し、そのロールを一括取り込みユーザーに付与します。以下の SQL 文を使用します。  

```
create role INFA_CMI_Role;  
grant role INFA_CMI_Role to user INFACMI_User;
```
3. Snowflake 仮想ウェアハウスの使用権限を新しいロールに付与します。次の SQL 文を使用します。  

```
grant usage on warehouse CMIWH to role INFA_CMI_Role;
```



4. Snowflake データベースの使用権限を新しいロールに付与します。次の SQL 文を使用します。

```
grant usage, CREATE SCHEMA on database CMIDB to role INFACMI_Role;
```

5. 新しく作成されたユーザーのデフォルトのロールを設定します。次の SQL 文を使用します。

```
alter user INFACMI_User set default_role=INFACMI_Role;
```

また、INFACMI\_User として新しいスキーマを作成します。

```
create schema CMISchema;
```

**注:** ユーザーのデフォルトロールが取り込みタスクに使用され、必要な権限がない場合、実行時に次のエラーが発行されます。

SQL compilation error: Object does not exist, or operation cannot be performed.

## 使用のガイドライン

- 一括取り込みは、Snowflake Data Cloud ターゲットにデータを移動する代替方法を提供しています。
  - 取り込みタスクを定義するときに **[Superpipe]** オプションを選択した場合、取り込みジョブは Snowpipe Streaming API を使用して、短い待ち時間でデータ行をターゲットテーブルに直接ストリーミングします。この方法は、すべてのロードタイプで使用できます。**[KeyPair]** 認証を使用する必要があります。
  - Superpipe を使用しない場合、取り込みジョブはまず、タスク定義で指定した名前を持つ内部ステージのデータファイルにデータを書き込みます。

- Superpipe を使用せず、取り込みジョブのターゲットプロパティで指定した内部ステージが存在しない場合、一括取り込みデータベースは、次の SQL コマンドを実行してステージを自動的に作成します。

```
Create stage if not exists "Schema"."Stage_Bucket"
```

コマンドを正常に実行するには、次の特権をユーザーロールに付与する必要があります。

```
GRANT CREATE STAGE ON SCHEMA "Schema" TO ROLE <your_role>;
```

- Snowflake ターゲットの接続を定義するときは、**[JDBC URL の追加パラメータ]** フィールドに `database=target_database_name` と設定する必要があります。それ以外の場合は、アプリケーション取り込みタスクウィザードでターゲットを定義しようとすると、スキーマのリストを取得できないことを示すエラーメッセージが表示されます。

- **[KeyPair]** オプションを認証方法として使用して Snowflake ターゲットの接続を定義し、OpenSSL 3.x.x バージョンでプライベートキーを生成する場合は、プライベートキーの生成時に PBE-SHA1-2DES または PBE-SHA1-3DES の暗号を使用します。以下のコマンドのいずれかを実行します。

```
openssl genrsa 2048 | openssl pkcs8 -topk8 -inform PEM -v1 PBE-SHA1-3DES -out rsa_key.p8
```

または

```
openssl genrsa 2048 | openssl pkcs8 -topk8 -inform PEM -v1 PBE-SHA1-2DES -out rsa_key.p8
```

PBE-SHA1-2DES または PBE-SHA1-3DES の暗号なしで汎用コマンドを使用すると、アプリケーション取り込みタスクウィザードのターゲット定義ステップでターゲットスキーマを取得するときに、プライベートキーが無効またはサポートされていないことを示すエラーメッセージが表示される場合があります。

OpenSSL 1.1.1 を使用してプライベートキーを生成すると、エラーメッセージは表示されません。

- 増分ロードジョブにより、ターゲット上に INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルが生成され、内部サービス情報が格納されます。リカバリテーブルのデータによって、障害後に再開されたジョブが以前に処理されたデータを再度プロパゲートすることが防止されます。リカバリテーブルは、ターゲットテーブルを含むスキーマで生成されます。
- オブジェクトには、16MB を超えるレコードが含まれてはいけません。
- Snowflake ターゲットの場合、NUMBER フィールドのスケールを変更したり、既存のフィールドのデータ型を別のデータ型に変更したりすることはできません。Snowflake ではこれらの操作がサポートされていないためです。

## Snowflake へのプライベート接続の設定

AWS または Azure プライベートリンクエンドポイントを使用して Snowflake にアクセスできます。

AWS または Azure プライベートリンクの設定によって、Snowflake への接続が AWS または Azure 内部ネットワークを使用して確立され、パブリックインターネットを介して行われなくなります。

プライベート AWS ネットワーク経由で Snowflake アカウントに接続するには、[AWS Private Link and Snowflake](#) を参照してください。

プライベート Azure ネットワーク経由で Snowflake アカウントに接続するには、[Azure Private Link and Snowflake](#) を参照してください。

## Avro データ型

一括取り込みアプリケーションは、Avro スキーマが提供するプリミティブデータ型と論理データ型の一部のみをサポートします。

プリミティブデータ型は、単一のデータ値を表すことができる型です。論理型は、派生型を表す追加の属性を持つ Avro プリミティブ型または複合型です。このトピックは、Avro または Parquet の出力形式をサポートするすべてのターゲットタイプに適用されます。

次の表に、一括取り込みアプリケーションがサポートするプリミティブ Avro データ型を示します。

プリミティブデータ型	説明
INT	32 ビット符号付き整数
LONG	64 ビット符号付き整数
FLOAT	単精度 (32 ビット) IEEE 754 浮動小数点数
DOUBLE	倍精度 (64 ビット) IEEE 754 浮動小数点数
BYTES	8 ビットの符号なしバイトのシーケンス
STRING	Unicode 文字シーケンス

次の表に、一括取り込みアプリケーションがサポートする論理 Avro データ型を示します。

論理データ型	説明
DECIMAL	スケーリングされていない形式の任意精度の符号付き 10 進数 $\times 10^{-scale}$
DATE	時刻またはタイムゾーンへの参照を含まない日付。
TIME	タイムゾーンまたは日付への参照を含まない、1 ミリ秒または 1 マイクロ秒の精度の時刻。
TIMESTAMP	特定のカレンダーまたはタイムゾーンへの参照を含まない、1 ミリ秒または 1 マイクロ秒の精度の日時値。



# ソーススキーマの変更の処理

一括取り込みアプリケーションは、スキーマドリフトとも呼ばれるソーススキーマの変更を自動的に検出し、ターゲットでこれらの変更を処理するように設定できます。この機能は、増分ロードタスクおよび初期ロードと増分ロードの組み合わせタスクでのみ使用できます。

アプリケーション取り込みタスクを設定するとき、アプリケーション取り込みタスクウィザードの【**スケジュールおよびランタイムオプション**】ページで、一括取り込みアプリケーションがタスクに関連付けられたジョブに対してプロパゲートする必要があるソーススキーマの変更のタイプを指定できます。また、ジョブが各タイプのソーススキーマの変更を処理する方法を指定することもできます。例えば、ソースで特定のタイプのスキーマ変更が発生したときに、変更を無視するか、レプリケートするか、ジョブを停止するようにタスクを設定できます。詳細については、[「スケジュールとランタイムオプションの設定」 \(ページ 132\)](#)を参照してください。

**注:** 初期ロードジョブと増分ロードジョブの組み合わせでは、一括取り込みアプリケーションは、ソースオブジェクトが通常の状態に達した後でのみ、ソーススキーマの変更の検出とレプリケートを開始します。

次の表に、一括取り込みアプリケーションが各ソースタイプについて検出する、スキーマ変更の種類を示します。

ソースタイプ	サポートされるスキーマの変更
Adobe Analytics	<ul style="list-style-type: none"><li>- フィールドの追加</li><li>- フィールドの変更</li><li>- フィールドの削除</li><li>- フィールドの名前変更</li></ul>
Google Analytics	<ul style="list-style-type: none"><li>- カラムの追加</li><li>- カラムの変更</li><li>- カラムの削除</li><li>- カラム名の変更</li></ul>
Marketo	サポートされていません。
Microsoft Dynamics 365	<ul style="list-style-type: none"><li>- カラムの追加</li><li>- カラムの変更</li><li>- カラムの削除</li></ul>
NetSuite	<ul style="list-style-type: none"><li>- カラムの追加</li><li>- カラムの変更</li></ul>
Oracle Fusion Cloud	<ul style="list-style-type: none"><li>- REST - サポートされていません</li><li>- BICC - サポートされていません</li></ul>
Salesforce	<ul style="list-style-type: none"><li>- フィールドの追加</li><li>- フィールドの変更</li><li>- フィールドの削除</li><li>- フィールドの名前変更</li></ul>
SAP	<ul style="list-style-type: none"><li>- SAP ODP Extractor コネクタの使用 - サポートされていません</li><li>- SAP Mass Ingestion コネクタの使用 - サポートされていません</li></ul>
ServiceNow	<ul style="list-style-type: none"><li>- カラムの追加</li><li>- カラムの変更</li><li>- カラムの削除</li></ul>

ソースタイプ	サポートされるスキーマの変更
Workday	- SOAP - サポートされていません - RaaS - サポートされていません
Zendesk	サポートされていません。

一括取り込みアプリケーションがソースオブジェクトのスキーマ変更を検出するのは、変更されたソースオブジェクトでデータ操作言語（DML）操作が発生した後のみです。DML 操作に干渉せずに複数のスキーマ変更が発生した場合、一括取り込みアプリケーションは DML 操作が発生すると、スキーマのすべての変更をまとめて検出します。

#### 注:

- アプリケーション取り込みジョブは、プライマリキーまたは一意キーの制約を追加、削除、または変更するソースの変更をレプリケートしません。ソースでこれらのタイプの変更が発生した場合は、ターゲットテーブルを再同期する必要があります。
- Microsoft Azure Synapse Analytics ターゲット用に設定されたアプリケーション取り込みジョブは、ソースフィールドの名前変更をターゲットにレプリケートしません。
- Google BigQuery ターゲット用に設定されたアプリケーション取り込みジョブは、ソースフィールドの変更と名前変更をターゲットにレプリケートしません。
- Snowflake ターゲット用に設定されたアプリケーション取り込みジョブは、ソースカラムに対する変更操作をサポートしますが、次の制限があります。
  - Snowflake ターゲットは、NUMBER カラムのスケールを変更できません。
  - Snowflake ターゲットは、既存のカラムのデータ型を別のデータ型に変更することはサポートしていません。
- ターゲットでサポートされていないタイプのスキーマ変更をレプリケートしようとすると、そのタスクに関連付けられたアプリケーション取り込みジョブはエラーが発生して失敗します。
- 一括取り込みアプリケーションがスキーマの変更を検出したときにジョブを停止するスキーマドリフトオプションを設定している場合は、**【オプションを指定して再開】** コマンドを使用して、オーバーライドスキーマドリフトオプションを指定してジョブを再開することができます。

## ターゲットに対する削除を論理削除として適用する機能

サポートされているソースタイプと Databricks Delta および Snowflake ターゲットを持つアプリケーション取り込み増分ロードジョブおよび初期ロードと増分ロードの組み合わせジョブの場合、ターゲットでソースの削除操作を論理削除として処理するようにタスクを設定できます。

論理削除では、削除された行をデータベースから実際には削除せずに、削除済みとしてマークします。この行は、生成された INFA\_OPERATION\_TYPE メタデータカラムに「D」という値を設定してターゲットに適用されます。

**重要:** 論理削除は、すべてのソースオブジェクトにプライマリキーがあり、行が最初に作成された後は行のプライマリキー値を変更することがソースで許可されていない場合にのみ使用できます。プライマリキーの値が変更されると、重複した行がターゲットに書き込まれ、ターゲットが破損する可能性があります。

シナリオ例: 組織は、データウェアハウスで論理削除を使用して、監査目的で行を保持しながら、ソースで削除された行をマークする必要があるとします。

論理削除を有効にするには、取り込みタスクを設定するときに、タスクウィザードの【ターゲット】ページの【適用モード】フィールドを【論理削除】に設定します。

## すべてのソーステーブル変更操作の監査履歴のターゲットテーブルへの適用

ソーステーブルに対して行われたすべての DML 変更操作の監査証跡をターゲットに書き込むために、Snowflake ターゲットを持つアプリケーション取り込み増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクを構成できます。ソーステーブルの各 DML 変更の行が、選択した監査カラムとともに、生成されたターゲットテーブルに書き込まれます。監査カラムには、DML 操作タイプ、時刻、所有者、トランザクション ID、生成された昇順シーケンス番号、前のイメージなどの変更に関するメタデータが含まれています。

タスクを定義するときに、【ターゲット】ページの【適用モード】フィールドで【監査】を選択します。【適用モード】フィールドは、新しいタスクまたはデプロイされていないタスクで使用できます。

追加する監査メタデータカラムを指定するには、【ターゲット】ページの【詳細】の下にある次のチェックボックスを 1 つ以上選択します:

- **最終レプリケート時刻を追加。** ターゲットテーブルでレコードが挿入または最後に更新された時点のタイムスタンプを記録するカラムを追加します。初期ロードの場合、ロードされたすべてのレコードのタイムスタンプは同じですが、Superpipe オプションを使用する Snowflake ターゲットに限り、秒または分がわずかに異なる可能性があります。増分ロード、および初期ロードと増分ロードの組み合わせの場合、このカラムには、ターゲットに適用された最後の DML 操作のタイムスタンプが記録されます。
- **操作の追加 <metadata\_type>。** DML 操作タイプ、時刻、所有者、トランザクション ID、生成された昇順シーケンス番号など、変更操作のメタデータを含むカラムが追加されます。カラムには、データがターゲットテーブルにロードされる時に入力されます。
- **前のイメージを追加。** 更新の操作前のイメージデータを含む\_OLD カラムが追加されます。テーブルの以前のカラムの値と新しいカラムの値を比較できます。
- **メタデータカラムのプレフィックス。** 追加された監査カラムの名前にプレフィックスを追加して、他のテーブルカラムと区別できるようにします。デフォルトは INFA\_ です。

これらのフィールドはオプションです。【操作タイプの追加】チェックボックスのみデフォルトで選択されており、次の DML 操作タイプを示すカラムが追加されます: D (削除)、または U (更新)。

タスクに関連付けられたジョブを初めて実行すると、ジョブは、選択した監査メタデータカラムを含むターゲットテーブルを生成します。ターゲットテーブルにインデックス以外の制約が存在しないことを確認します。

Snowflake ターゲットなど、【論理削除】モードをサポートするターゲットの場合、論理削除として処理された各削除操作に対して、操作タイプ「D」が INFA\_OPERATION\_TYPE カラムに表示され、選択した他のメタデータカラムに値が書き込まれます。ただし、更新操作と挿入操作では、INFA\_OPERATION\_TYPE カラムと選択されている他のすべてのメタデータカラムは NULL です。

### 監査モードの例

たとえば、ソーステーブルに対して次の DML 変更操作が次の順序で発生するとします。

```
Insert into tableA pkey = 1
Update tableA where pkey=1
Update tableA where pkey=1
Delete from tableA where pkey = 1
```

次のすべての行がターゲットテーブルに表示され、すべてのソース DML 変更の監査証跡が提供されます。

```
opType=I, pkey=1...
opType=E, pkey=1...
opType=E, pkey=1...
opType=D, pkey=1...
```

この例では、選択されている監査カラムは opType のみです。

タスクの適用モードが「標準」の場合、最後の DML 操作は削除であり、以前の変更が上書きされるため、これらの行はターゲットテーブルに表示されません。

**注:** 初期ロードと増分ロードの組み合わせジョブで、初期アンロードフェーズ中に増分挿入変更レコードをキャプチャした場合、ジョブは同じ行に対して削除を実行して、初期アンロードで取得された可能性のある重複を削除します。この意図的に作成されたアクティビティは、監査適用モードで反映されます。

## アプリケーション取り込みタスクの設定

一括取り込みのアプリケーション取り込みタスクウィザードを使用して、アプリケーション取り込みタスクを設定します。

アプリケーション取り込みタスクを設定するには、ウィザードで次のタスクを実行します。

1. 「[基本的なタスク情報の定義](#)」(ページ 52) (タスク名、プロジェクトの場所、ランタイム環境、ロードタイプなど)。
2. 「[ソースの設定](#)」(ページ 53)。
3. 「[ターゲットの設定](#)」(ページ 98)。
4. 「[スケジュールとランタイムオプションの設定](#)」(ページ 132)。

【次へ】または【戻る】をクリックして別のページに移動します。いつでも【保存】をクリックして、それまでに入力した情報を保存できます。

すべてのウィザードページを完了したら、情報を保存し、【デプロイ】をクリックして、タスクを実行可能ジョブとして Secure Agent で使用できるようにします。

### 始める前に

アプリケーション取り込みタスクを設定する前に、Administrator で次の前提条件タスクを完了します。

- ランタイム環境の Secure Agent が実行されていること、および一括取り込みサービスにアクセスできることを確認します。
- ソース接続およびターゲット接続を定義します。

### 基本的なタスク情報の定義

アプリケーション取り込みタスクを定義するには、最初に、タスク名、プロジェクトまたはプロジェクトフォルダの場所、ロード操作のタイプなど、タスクに関するいくつかの基本情報を入力する必要があります。

1. 「[一括取り込み]」で、【新規】 > 【**アプリケーション取り込みタスク**】をクリックします。  
アプリケーション取り込みタスクウィザードの【定義】ページが表示されます。

2. 以下のプロパティを設定します。

プロパティ	説明
名前	アプリケーション取り込みタスクの名前。 アプリケーション取り込みタスクの名前は組織内で一意である必要があります。名前には、英数字、スペース、ピリオド (.)、コンマ (,)、アンダースコア (_)、プラス記号 (+)、およびハイフン (-) を含めることができます。 タスク名では大文字と小文字が区別されません。最大長は 50 文字です。 <b>注:</b> アプリケーション取り込みタスクの名前にスペースを含めた場合、スペースはタスクに関連付けられたジョブの名前には表示されません。
場所	タスクを保存するプロジェクトまたはフォルダ。
ランタイム環境	タスクを実行するランタイム環境。 アプリケーション取り込みタスクは Secure Agent でのみ実行できます。ランタイム環境には、エージェントが 1 つだけの Secure Agent グループを含めることができます。 <b>注:</b> ホステッドエージェントやサーバーレスランタイム環境でアプリケーション取り込みタスクを実行することはできません。
説明	タスクの簡単な説明。 最大長は 4000 文字です。
ロードタイプ	アプリケーション取り込みタスクで実行するロード操作のタイプ。タスクには、次のロードタイプのいずれかを選択できます。 <ul style="list-style-type: none"><li>- <b>初期ロード:</b> 特定の時点で読み取られたデータを、バッチ操作でソースアプリケーションからターゲットにロードします。初期ロードを実行して、増分変更データの送信先となるターゲットをマテリアライズできます。</li><li>- <b>増分ロード:</b> ソースデータの変更を継続的に、またはジョブが停止または終了するまでターゲットにプロパゲートします。ジョブは、ジョブが最後に実行されてから、または最初のジョブ実行の特定の開始点から発生した変更をプロパゲートします。</li><li>- <b>初期ロードと増分ロード:</b> ターゲットへのポイントインタイムデータの初期ロードを実行してから、同じソースオブジェクトに対して継続的に行われた増分データ変更のプロパゲートに自動的に切り替わります。</li></ul>

3. **【次へ】** をクリックします。

## ソースの設定

アプリケーション取り込みタスクウィザードの **【ソース】** ページでソースを設定できます。

ソースを設定する前に、組織が使用するランタイム環境のソースへの接続が Administrator で作成されていることを確認してください。

1. **【接続】** リストから、ソースアプリケーション用に設定された接続を選択します。接続タイプは、接続名の後に括弧で囲まれて表示されます。

リストには、**【定義】** ページで選択したロードタイプに有効な接続のみが含まれます。

**注:** 取り込みタスクをデプロイした後は、関連する取り込みジョブをデプロイ解除せずに接続を変更することはできません。接続を変更した後、タスクを再度デプロイする必要があります。

2. 設定するソースのタイプに基づいて、次のトピックに記載されている手順を実行します。

- [「Adobe Analytics ソースの設定」 \(ページ 54\)](#)

- [「Google Analytics ソースの設定」 \(ページ 57\)](#)
- [「Marketo ソースの設定」 \(ページ 61\)](#)
- [「Microsoft Dynamics 365 ソースの設定」 \(ページ 64\)](#)
- [「NetSuite ソースの設定」 \(ページ 67\)](#)
- [「Oracle Fusion Cloud ソースの設定」 \(ページ 70\)](#)
- [「Salesforce ソースの設定」 \(ページ 73\)](#)
- [「Salesforce Marketing Cloud ソースの設定」 \(ページ 77\)](#)
- [「SAP ODP Extractor コネクタを使用した SAP ソースの設定」 \(ページ 80\)](#)
- [「SAP Mass Ingestion コネクタを使用した SAP ソースの設定」 \(ページ 84\)](#)
- [「ServiceNow ソースの設定」 \(ページ 89\)](#)
- [「Workday ソースの設定」 \(ページ 92\)](#)
- [「Zendesk ソースの設定」 \(ページ 95\)](#)

## Adobe Analytics ソースの設定

**アプリケーション取り込み**タスクウィザードのアプリケーション取り込みページで、取り込むオブジェクトを指定し、Adobe Analytics ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【レポート構成ファイルへのパス】** フィールドに、レポート構成を含む JSON ファイルへのパスを入力します。
2. **【オブジェクトの選択】** セクションでは、データレプリケーションのためにすべてのソースオブジェクトとソースフィールドを選択する場合のみ、**【すべて選択】** を選択します。後続のフィールドで選択内容を編集することはできません。

**【選択されたオブジェクト】** フィールドには、選択されたすべてのオブジェクトの数が表示されます。ソースオブジェクトが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースオブジェクトのサブセットを定義できます。

3. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【オブジェクトビュー】** で、選択したオブジェクトのセットをオブジェクトごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソースオブジェクトを選択します。このルールによって選択されたオブジェクトの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたオブジェクトの合計】** にオブジェクト数を表示します。次に、**【ルールの適用】** をクリックして、**【オブジェクトビュー】** でオブジェクト数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】 (+)** アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【オブジェクトルール】** フィールドで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。

- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための1つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
- マスクには、次のワイルドカードの1つまたは両方を含めることができます: 1つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。
- ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。
- e. 終了したら、**【ルールの適用】** をクリックします。
- ヒント: 【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるオブジェクト】** と **【選択されたオブジェクトの合計】** の数を更新します。
- ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、オブジェクト数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【オブジェクトビュー】** リストに引き続きオブジェクトがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。
4. ルールに基づいて選択されたソースオブジェクトのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。
- 次の手順を実行して、フィールドアクションルールを作成します。
- ルールタイプとして **【フィールドアクション】** を選択します。
  - 隣接するリストから、次のいずれかのアクションタイプを選択します。
    - **LTRIM**。文字フィールド値の左側のスペースを切り捨てます。
    - **RTRIM**。文字フィールド値の右側のスペースを切り捨てます。
    - **TRIM**。文字フィールド値の左側と右側のスペースを切り捨てます。
  - 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を1つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したソースオブジェクトのフィールドと照合されます。
  - 【ルールの追加】** をクリックします。
- 注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。
5. **【オブジェクトビュー】** で、選択したオブジェクト（各オブジェクトのフィールド数、フィールド名、データ型を含む）を表示します。
- **【すべて選択】** を選択した場合、オブジェクトのリストは表示のみ可能です。



- ルールを適用した場合は、個々のオブジェクトの横にあるチェックボックスをクリックして、選択したオブジェクトのセットを絞り込むことができます。レプリケートしないオブジェクトの選択を解除するか、レプリケートする追加のオブジェクトを選択します。[更新] アイコンをクリックして、選択したオブジェクトの数を更新します。

**注:** オブジェクトのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【オブジェクトビュー】** の選択内容が優先されます。ただし、[ルールの追加] (+) アイコンを再度クリックすると、個別に選択解除または選択したオブジェクトが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【オブジェクトビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

オブジェクトごとに、フィールド名とデータ型のリストを表示できます。**【フィールド】** カラムにある、強調表示されたフィールド数をクリックし、右側にフィールドを一覧表示します。

オブジェクトやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【オブジェクト名】**、**【フィールド】**、または **【すべて】** を選択し、[検索] ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

6. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。
  - a. **【ルールタイプ別のオブジェクトのリスト】** から、選択したソースオブジェクトのリストをダウンロードする選択ルールのタイプを選択します。
  - b. リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
  - c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。  
ダウンロードした情報

は、次の形式です。

*status, object\_name, object\_type, field\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E: オブジェクトは、除外ルールによって処理から除外されます。</li> <li>- I: オブジェクトは処理対象として含まれます。</li> <li>- X: オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。</li> </ul>
object_name	ソースオブジェクトの名前。
object_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: オブジェクトを示します。</li> <li>- F: フィールドを示します。</li> </ul>



フィールド	説明
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に <b>【フィールドを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソースオブジェクトが選択ルールに一致していても処理から除外される理由。

7. **【詳細】** セクションを展開します。
8. 初期ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、取り込みジョブがソースデータのレプリケーションを開始する日時を指定します。  
**注:** 日付と時刻は、レポート構成の JSON ファイルの ReportSuiteID に指定されたタイムゾーンの範囲内にある必要があります
9. 初期ロードタスクの場合は、取り込みジョブがソースデータのレプリケーションを停止する日時を指定します。  
**注:** 日付と時刻は、レポート構成の JSON ファイルの ReportSuiteID に指定されたタイムゾーンの範囲内にある必要があります
10. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。  
**注:** 日付と時刻を協定世界時（UTC）で指定する必要があります。
11. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 1 日です。
12. **【取得サイズ】** フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルト値は 50000 です。
13. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。  
 カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。  
**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。
14. **【次へ】** をクリックします。

## Google Analytics ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むオブジェクトを指定し、Google Analytics ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【アカウント ID】** フィールドに、Google Analytics サービスアカウントの一意の識別子を入力します。
2. **【プロパティ ID】** フィールドに、データをレプリケートするプロパティの一意の識別子を入力します。
3. **【ビュー ID】** フィールドに、データをレプリケートするビューの一意の識別子を入力します。
4. **【レポート構成ファイルへのパス】** フィールドに、レポート構成を含む JSON ファイルへのパスを入力します。

5. **【レポートの選択】** セクションでは、データレプリケーションのためにすべてのソースレポートを選択する場合のみ、**【すべて選択】** を選択します。後続のディメンションとメトリックで選択内容を編集することはできません。

**【選択されたレポート】** フィールドには、選択されたすべてのレポートの数が表示されます。ソースレポートが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースレポートのサブセットを定義できます。

6. **【ルールベースの選択】** を選択した場合、ターゲットにレプリケートするソースレポートを選択するルールを作成します。

デフォルトでは、すべてのソースレポートを選択するように設定された含めるルールがタスクに定義されています。一部のソースレポートをターゲットにレプリケートする場合は、含めるルールと除外ルールを追加で定義して、レプリケートする特定のレポートを選択できます。

次の手順を実行して、レポート選択ルールを作成します。

- ルールタイプとして **【レポートの選択】** を選択します。
- 隣接するリストから、ルールで実行するアクションとして **【含む】** または **【除外】** を選択します。
- 条件フィールドにレポート名またはレポート名マスクを入力して、選択したレポートのリストに含めるソースレポート、または同リストから除外するソースレポートを指定します。

**注意事項:**

- マスクには、1 つ以上の文字を表すアスタリスク (\*) ワイルドカード文字、単一の文字を表す疑問符 (?) ワイルドカード文字、または両方のタイプのワイルドカード文字を含めることができます。レポート名マスクでは、ワイルドカード文字を複数回使用できます。
- レポート名の大文字と小文字は区別されます。レポート選択ルールを定義するときは、ソースで定義されているとおりに大文字小文字を区別してレポート名またはマスクを指定する必要があります。
- レポート名に引用符や括弧などの区切り文字が含まれている場合、ルールでレポート名を指定するときには区切り文字を含めないでください。
- レポート名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールでレポート名を指定するときに名前の各特殊文字にバックスラッシュ (\) をつけて置き換えます。

- 【ルールの追加】** をクリックします。

ルールが **【ルール】** リストに表示されます。

選択を絞り込むには、追加の含めるルールと除外ルールを定義します。レポート選択ルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。複数のルールを使用する例については、[「ソースオブジェクトを選択するルールの例」 \(ページ 98\)](#) を参照してください。

ルールを作成した後、**【レポート数】** をクリックすると、**【影響を受けるオブジェクト】** カラムに各ルールと一致するソースレポートの数が表示され、**【選択されたレポートの合計】** フィールドにすべての選択ルールに基づいて選択されたレポートの総数が表示されます。

- e. すべてのルールに基づいて選択するレポートをプレビューするには、**【ルールの適用】** をクリックします。

レポートは、**【選択したレポート】** タブに一覧表示されます。リストには、レポート名とカラム数が表示されます。

テーブルとカラムを検索するには、オブジェクトのリストを参照するか、**【検索】** ボックスに検索文字列を入力します。

**ヒント:** **【更新済み】** の日付の横にある **【更新】** アイコンをクリックし、レポートの総数と現在のルールに基づいて選択されたレポートのリストを更新します。この方法で、新しいルールの結果を確認できます。**【設定】** アイコンをクリックして、レポートリストの行間隔を **【十分】**（最大の間隔）から **【コンパクト】**（最小の間隔）の間で制御します。

7. ルールに基づいて選択されたソースレポートのカラムに対してトリムアクションを実行するには、カラムアクションルールを作成します。

次の手順を実行して、カラムアクションルールを作成します。

- a. ルールタイプとして **【カラムアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM.**文字カラム値の左側のスペースを切り捨てます。
  - **RTRIM.**文字カラム値の右側のスペースを切り捨てます。
  - **TRIM.**文字カラム値の左側と右側のスペースを切り捨てます。
- c. 条件フィールドに、カラム名または1つ以上のアスタリスク (\*) または疑問符 (?) を含むカラム名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるカラムを識別するために、選択したソースレポートのカラムと照合されます。
- d. **【ルールの追加】** をクリックします。

**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。カラムアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

8. **【レポートビュー】** で、選択したレポート（各レポートのルール数を含む）を表示します。

- **【すべて選択】** を選択した場合、レポートのリストは表示のみ可能です。
- ルールを適用した場合は、個々のレポートの横にあるチェックボックスをクリックして、選択したレポートのセットを絞り込むことができます。レプリケートしないレポートの選択を解除するか、レプリケートする追加のレポートを選択します。**【更新】** アイコンをクリックして、選択したレポートの数を更新します。

**注:** レポートのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【レポートビュー】** の選択内容が優先されます。ただし、**【ルールの追加】** (+) アイコンを再度クリックすると、個別に選択解除または選択したレポートが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【レポートビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

レポートごとに、カラムのリストを表示できます。**【カラム】** にある、強調表示されたカラム数をクリックし、右側にフィールドを一覧表示します。

オブジェクトやフィールドを検索するには、**【カラム】** の上のドロップダウンリストで **【レポート名】**、**【カラム】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを1つ含めることができます。

9. 選択ルールに一致するソースレポートのリストをダウンロードするには、次の手順を実行します。
  - a. **【ルールタイプ別のレポートのリスト】** から、選択したソースレポートのリストをダウンロードする選択ルールのタイプを選択します。
  - b. リストにカラムを含める場合は、**【カラムを含める】** を選択します。
  - c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースレポートのリストがローカルドライブにダウンロードされます。  
ダウンロードしたファイルの情報の形式は次のとおりです。

*status, report\_name, report\_type, column\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースレポートを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E: レポートは、除外ルールによって処理から除外されます。</li> <li>- I: レポートは処理対象として含まれます。</li> <li>- X: レポートは選択ルールに一致していても処理から除外されます。ファイルの [comment] フィールドに、レポートが除外された詳細な理由が記述されます。</li> </ul>
report_name	ソースレポートの名前。
report_type	ソースレポートのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: レポートを示します。</li> <li>- F: カラムを示します。</li> </ul>
column_name	ソースカラムの名前。この情報は、リストをダウンロードする前に <b>【カラムを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソースレポートが選択ルールに一致していても処理から除外される理由。

10. **【詳細】** セクションを展開します。
11. 初期ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、取り込みジョブがソースデータのレプリケーションを開始する日時を指定します。
12. 初期ロードタスクの場合は、取り込みジョブがソースデータのレプリケーションを停止する日時を指定します。
13. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。
 

**注:** 日付は、Google Analytics ビューについて設定されたタイムゾーンで指定する必要があります。
14. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 1 日です。
15. **【取得サイズ】** フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルト値は 50000 です。
16. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

17. **【次へ】** をクリックします。

## Marketo ソースの設定

**アプリケーション取り込み**タスクウィザードのアプリケーション取り込みページで、取り込むオブジェクトを指定し、Marketo ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【オブジェクトの選択】** セクションでは、データレプリケーションのためにすべてのソースオブジェクトとソースフィールドを選択する場合のみ、**【すべて選択】** を選択します。後続のフィールドで選択内容を編集することはできません。

**【選択されたオブジェクト】** フィールドには、選択されたすべてのオブジェクトの数が表示されます。ソースオブジェクトが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースオブジェクトのサブセットを定義できます。

2. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【オブジェクトビュー】** で、選択したオブジェクトのセットをオブジェクトごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソースオブジェクトを選択します。このルールによって選択されたオブジェクトの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたオブジェクトの合計】** にオブジェクト数を表示します。次に、**【ルールの適用】** をクリックして、**【オブジェクトビュー】** でオブジェクト数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】** (+) アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【オブジェクトルール】** フィールドで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。

- e. 終了したら、**【ルールの適用】** をクリックします。

**ヒント:** **【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるオブジェクト】** と **【選択されたオブジェクトの合計】** の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、オブジェクト数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【オブジェクトビュー】** リストに引き続きオブジェクトがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。

3. ルールに基づいて選択されたソースオブジェクトのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。

次の手順を実行して、フィールドアクションルールを作成します。

- a. ルールタイプとして **【フィールドアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM**。文字フィールド値の左側のスペースを切り捨てます。
  - **RTRIM**。文字フィールド値の右側のスペースを切り捨てます。
  - **TRIM**。文字フィールド値の左側と右側のスペースを切り捨てます。

- c. 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を 1 つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したソースオブジェクトのフィールドと照合されます。

- d. **【ルールの追加】** をクリックします。

**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

4. **【オブジェクトビュー】** で、選択したオブジェクト（各オブジェクトのフィールド数、フィールド名、データ型を含む）を表示します。

- **【すべて選択】** を選択した場合、オブジェクトのリストは表示のみ可能です。
- ルールを適用した場合は、個々のオブジェクトの横にあるチェックボックスをクリックして、選択したオブジェクトのセットを絞り込むことができます。レプリケートしないオブジェクトの選択を解除するか、レプリケートする追加のオブジェクトを選択します。**【更新】** アイコンをクリックして、選択したオブジェクトの数を更新します。

**注:** オブジェクトのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【オブジェクトビュー】** の選択内容が優先されます。ただし、**【ルールの追加】** (+) アイコンを再度クリックすると、個別に選択解除または選択したオブジェクトが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【オブジェクトビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

オブジェクトごとに、フィールド名とデータ型のリストを表示できます。**【フィールド】** カラムにある、強調表示されたフィールド数をクリックし、右側にフィールドを一覧表示します。

オブジェクトやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【オブジェクト名】**、**【フィールド】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

5. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。

- a. **【ルールタイプ別のオブジェクトのリスト】** から、選択したソースオブジェクトのリストをダウンロードする選択ルールのタイプを選択します。

- b. リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
- c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。  
ダウンロードしたファイルの情報の形式は次のとおりです。

*status,MARKETO,object\_name,object\_type,field\_name,comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 - E。オブジェクトは、除外ルールによって処理から除外されます。 - I: オブジェクトは処理対象として含まれます。 - X。オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。
MARKETO	ソースアプリケーションの名前。
object_name	ソースオブジェクトの名前。
object_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 - O: オブジェクトを示します。 - F: フィールドを示します。
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に <b>【フィールドを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソースオブジェクトが選択ルールに一致していても処理から除外される理由。

6. **【詳細】** セクションを展開します。
7. 初期ロードタスクの場合、アプリケーション取り込みタスクに関連付けられた取り込みジョブがソースのリードオブジェクトとカスタムオブジェクトからレコードの読み取りを開始する日付を、**【開始日】** フィールドに指定します。
8. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。  
**注:** 日付と時刻を協定世界時 (UTC) で指定する必要があります。
9. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
10. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。  
カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。  
**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。
11. **【次へ】** をクリックします。



## Microsoft Dynamics 365 ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むテーブルを指定し、Microsoft Dynamics 365 ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【テーブルの選択】** セクションでは、データレプリケーションのためにすべてのソーステーブルとソースカラムを選択する場合のみ、**【すべて選択】** を選択します。後続のカラムで選択内容を編集することはできません。

**【選択されたテーブル】** フィールドには、選択されたすべてのテーブルの数が表示されます。ソーステーブルが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソーステーブルのサブセットを定義できます。

2. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【テーブルビュー】** で、選択したテーブルのセットをテーブルごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソーステーブルを選択します。このルールによって選択されたテーブルの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたテーブルの合計】** にテーブル数を表示します。次に、**【ルールの適用】** をクリックして、**【テーブルビュー】** でテーブル数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】** (+) アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【テーブルルール】** カラムで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。



- e. 終了したら、**【ルールの適用】** をクリックします。

**ヒント:** **【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるテーブル】** と **【選択されたテーブルの合計】** の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、テーブル数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【テーブルビュー】** リストに引き続きテーブルがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。

3. ルールに基づいて選択されたソーステーブルのカラムに対してトリムアクションを実行するには、カラムアクションルールを作成します。

次の手順を実行して、カラムアクションルールを作成します。

- a. ルールタイプとして **【カラムアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM.**文字カラム値の左側のスペースを切り捨てます。
  - **RTRIM.**文字カラム値の右側のスペースを切り捨てます。
  - **TRIM.**文字カラム値の左側と右側のスペースを切り捨てます。

- c. 条件フィールドに、カラム名または1つ以上のアスタリスク (\*) または疑問符 (?) を含むカラム名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるカラムを識別するために、選択したソーステーブルのカラムと照合されます。

- d. **【ルールの追加】** をクリックします。

**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。カラムアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

4. **【テーブルビュー】** で、選択したテーブル（各テーブルのカラム数を含む）を表示します。

- **【すべて選択】** を選択した場合、テーブルのリストは表示のみ可能です。
- ルールを適用した場合は、個々のテーブルの横にあるチェックボックスをクリックして、選択したテーブルのセットを絞り込むことができます。レプリケートしないレポートの選択を解除するか、レプリケートする追加のテーブルを選択します。**【更新】** アイコンをクリックして、選択したテーブルの数を更新します。

**注:** テーブルのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【テーブルビュー】** の選択内容が優先されます。ただし、**【ルールの追加】 (+)** アイコンを再度クリックすると、個別に選択解除または選択したテーブルが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【テーブルビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

テーブルごとに、カラムのリストを表示できます。**【カラム】** にある、強調表示されたカラム数をクリックし、右側にフィールドを一覧表示します。

テーブルやカラムを検索するには、**【カラム】** の上のドロップダウンリストで **【テーブル名】**、**【カラム】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを1つ含めることができます。

5. 選択ルールに一致するソーステーブルのリストをダウンロードするには、次の手順を実行します。

- a. **【ルールタイプ別のテーブルのリスト】** から、選択したソーステーブルのリストをダウンロードする選択ルールのタイプを選択します。
- b. リストにカラムを含める場合は、**【カラムを含める】** を選択します。

- c. [ダウンロード] アイコンをクリックします。

選択ルールに一致するソーステーブルのリストがローカルドライブにダウンロードされます。  
ダウンロードしたファイルの情報の形式は次のとおりです。

*status, table\_name, table\_type, column\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソーステーブルを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 - E: テーブルは、除外ルールによって処理から除外されます。 - I: テーブルは処理対象として含まれます。 - X: テーブルは選択ルールに一致していても処理から除外されます。ファイルの [comment] フィールドに、テーブルが除外された詳細な理由が記述されます。
table_name	ソーステーブルの名前。
table_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 - O: テーブルを示します。 - F: カラムを示します。
column_name	ソースカラムの名前。この情報は、リストをダウンロードする前に <b>【カラムを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソーステーブルが選択ルールに一致していても処理から除外される理由。

6. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【詳細】** セクションを展開します。
7. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。

**注:** 日付と時刻を協定世界時 (UTC) で指定する必要があります。

8. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
9. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

10. **【次へ】** をクリックします。

## NetSuite ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むテーブルを指定し、NetSuite ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【テーブルの選択】** セクションでは、データレプリケーションのためにすべてのソーステーブルとソースカラムを選択する場合のみ、**【すべて選択】** を選択します。後続のカラムで選択内容を編集することはできません。

**【選択されたテーブル】** フィールドには、選択されたすべてのテーブルの数が表示されます。ソーステーブルが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソーステーブルのサブセットを定義できます。

2. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【テーブルビュー】** で、選択したテーブルのセットをテーブルごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソーステーブルを選択します。このルールによって選択されたテーブルの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたテーブルの合計】** にテーブル数を表示します。次に、**【ルールの適用】** をクリックして、**【テーブルビュー】** でテーブル数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】** (+) アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【テーブルルール】** カラムで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。

- e. 終了したら、[**ルールの適用**] をクリックします。

**ヒント:** [更新済み] タイムスタンプの右側にある [更新] アイコンをクリックして、[影響を受けるテーブル] と [選択されたテーブルの合計] の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、[**ルールの適用**] を再度クリックする必要があります。[更新] アイコンをクリックして、テーブル数を更新します。[**ルールの適用**] をクリックしないですべてのルールを削除すると、[**テーブルビュー**] リストに引き続きテーブルがリストされていても、デプロイ時に検証エラーが発生します。[**すべて選択**] に切り替えると、ルールは表示されなくなります。

3. ルールに基づいて選択されたソーステーブルのカラムに対してトリムアクションを実行するには、カラムアクションルールを作成します。

次の手順を実行して、カラムアクションルールを作成します。

- a. ルールタイプとして [**カラムアクション**] を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM.**文字カラム値の左側のスペースを切り捨てます。
  - **RTRIM.**文字カラム値の右側のスペースを切り捨てます。
  - **TRIM.**文字カラム値の左側と右側のスペースを切り捨てます。

- c. 条件フィールドに、カラム名または1つ以上のアスタリスク (\*) または疑問符 (?) を含むカラム名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるカラムを識別するために、選択したソーステーブルのカラムと照合されます。

- d. [**ルールの追加**] をクリックします。

**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。カラムアクションルールは、[**ルール**] リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

4. [**テーブルビュー**] で、選択したテーブル（各テーブルのカラム数を含む）を表示します。

- [**すべて選択**] を選択した場合、テーブルのリストは表示のみ可能です。
- ルールを適用した場合は、個々のテーブルの横にあるチェックボックスをクリックして、選択したテーブルのセットを絞り込むことができます。レプリケートしないレポートの選択を解除するか、レプリケートする追加のテーブルを選択します。[更新] アイコンをクリックして、選択したテーブルの数を更新します。

**注:** テーブルのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。[**テーブルビュー**] の選択内容が優先されます。ただし、[**ルールの追加**] (+) アイコンを再度クリックすると、個別に選択解除または選択したテーブルが、新しいルールとして [ルール] リストに反映され、ルールが再び優先されます。[**テーブルビュー**] リストに戻るには、[**ルールの適用**] を再度クリックします。

テーブルごとに、カラムのリストを表示できます。[**カラム**] にある、強調表示されたカラム数をクリックし、右側にフィールドを一覧表示します。

テーブルやカラムを検索するには、[**カラム**] の上のドロップダウンリストで [**テーブル名**]、[**カラム**]、または [**すべて**] を選択し、[検索] ボックスに検索文字列を入力して [**検索**] をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを1つ含めることができます。

5. 選択ルールに一致するソーステーブルのリストをダウンロードするには、次の手順を実行します。

- a. [**ルールタイプ別のテーブルのリスト**] から、選択したソーステーブルのリストをダウンロードする選択ルールのタイプを選択します。
- b. リストにカラムを含める場合は、[**カラムを含める**] を選択します。

- c. [ダウンロード] アイコンをクリックします。

選択ルールに一致するソーステーブルのリストがローカルドライブにダウンロードされます。  
ダウンロードしたファイルの情報の形式は次のとおりです。

*status, table\_name, table\_type, column\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソーステーブルを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 - E: テーブルは、除外ルールによって処理から除外されます。 - I: テーブルは処理対象として含まれます。 - X: テーブルは選択ルールに一致していても処理から除外されます。ファイルの [comment] フィールドに、テーブルが除外された詳細な理由が記述されます。
table_name	ソーステーブルの名前。
table_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 - O: テーブルを示します。 - F: カラムを示します。
column_name	ソースカラムの名前。この情報は、リストをダウンロードする前に <b>【カラムを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソーステーブルが選択ルールに一致していても処理から除外される理由。

6. **【詳細】** セクションを展開します。
7. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。  
**注:** 日付と時刻をグリニッジ標準時 (GMT) で指定する必要があります。
8. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
9. **【取得サイズ】** フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルトは 5000 です。
10. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。  
カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。  
**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。
11. **【次へ】** をクリックします。

## Oracle Fusion Cloud ソースの設定

**アプリケーション取り込み**タスクウィザードのアプリケーション取り込みページで、取り込むオブジェクトを指定し、Oracle Fusion Cloud ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. 次のレプリケーション方法のいずれかを選択します。
  - Oracle Fusion の各種アプリケーション（ERP、SCM、HCM、Sales、Services など）からデータを抽出し、データをターゲットに転送する場合は、**[REST]** を選択します。
  - ソースからターゲットに大量のデータを抽出する場合は、**[BICC]**（Business Intelligence Cloud Connector）を選択します。
2. **[Oracle Fusion アプリケーション]** リストから、データをレプリケートするアプリケーションを選択します。
3. **[オブジェクトの選択]** セクションでは、データレプリケーションのためにすべてのソースオブジェクトとソースフィールドを選択する場合のみ、**[すべて選択]** を選択します。後続のフィールドで選択内容を編集することはできません。

**[選択されたオブジェクト]** フィールドには、選択されたすべてのオブジェクトの数が表示されます。ソースオブジェクトが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースオブジェクトのサブセットを定義できます。
4. ルールを使用してソースオブジェクトを選択するには、**[すべて選択]** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**[オブジェクトビュー]** で、選択したオブジェクトのセットをオブジェクトごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソースオブジェクトを選択します。このルールによって選択されたオブジェクトの数を確認するには、**[更新]** アイコンをクリックして、**[選択されたオブジェクトの合計]** にオブジェクト数を表示します。次に、**[ルールの適用]** をクリックして、**[オブジェクトビュー]** でオブジェクト数を確認します。

ルールを追加する手順:

- a. **[ルール]** の下で、最初のテーブルの上にある **[ルールの追加]** (+) アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **[オブジェクトルール]** フィールドで、**[含む]** または **[除外]** を選択して、包含ルールまたは除外ルールを作成します。
- c. **[条件]** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。

- e. 終了したら、**【ルールの適用】** をクリックします。

**ヒント:** **【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるオブジェクト】** と **【選択されたオブジェクトの合計】** の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、オブジェクト数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【オブジェクトビュー】** リストに引き続きオブジェクトがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。

5. ルールに基づいて選択されたソースオブジェクトのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。

次の手順を実行して、フィールドアクションルールを作成します。

- a. ルールタイプとして **【フィールドアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM**。文字フィールド値の左側のスペースを切り捨てます。
  - **RTRIM**。文字フィールド値の右側のスペースを切り捨てます。
  - **TRIM**。文字フィールド値の左側と右側のスペースを切り捨てます。
- c. 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を 1 つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したソースオブジェクトのフィールドと照合されます。
- d. **【ルールの追加】** をクリックします。

**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

6. **【オブジェクトビュー】** で、選択したオブジェクト（各オブジェクトのフィールド数、フィールド名、データ型を含む）を表示します。

- **【すべて選択】** を選択した場合、オブジェクトのリストは表示のみ可能です。
- ルールを適用した場合は、個々のオブジェクトの横にあるチェックボックスをクリックして、選択したオブジェクトのセットを絞り込むことができます。レプリケートしないオブジェクトの選択を解除するか、レプリケートする追加のオブジェクトを選択します。**【更新】** アイコンをクリックして、選択したオブジェクトの数を更新します。

**注:** オブジェクトのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【オブジェクトビュー】** の選択内容が優先されます。ただし、**【ルールの追加】** (+) アイコンを再度クリックすると、個別に選択解除または選択したオブジェクトが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【オブジェクトビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

オブジェクトごとに、フィールド名とデータ型のリストを表示できます。**【フィールド】** カラムにある、強調表示されたフィールド数をクリックし、右側にフィールドを一覧表示します。

オブジェクトやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【オブジェクト名】**、**【フィールド】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

7. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。

- a. **【ルールタイプ別のオブジェクトのリスト】** から、選択したソースオブジェクトのリストをダウンロードする選択ルールのタイプを選択します。

- b. リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
- c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。  
ダウンロードしたファイルの情報の形式は次のとおりです。

*status, object\_name, object\_type, field\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E: オブジェクトは、除外ルールによって処理から除外されます。</li> <li>- I: オブジェクトは処理対象として含まれます。</li> <li>- X: オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。</li> </ul>
object_name	ソースオブジェクトの名前。
object_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: オブジェクトを示します。</li> <li>- F: フィールドを示します。</li> </ul>
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に <b>【フィールドを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソースオブジェクトが選択ルールに一致していても処理から除外される理由。

テーブルとカラムを検索するには、オブジェクトのリストを参照するか、**【検索】** ボックスに検索文字列を入力します。

8. **【詳細】** セクションを展開します。
9. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。  
**注:** 日付と時刻は、Oracle Fusion Cloud インスタンスについて設定されたタイムゾーンで指定する必要があります。
10. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
11. **【子オブジェクトを含める】** オプションを選択し、Oracle Fusion Cloud ソースを使用してオブジェクトの子オブジェクトデータを取得します。これは、REST レプリケーションアプローチのみに適用されますが、ターゲットが Google Big Query の場合に限り、すべてのロードタイプに適用されます。
12. **【取得サイズ】** フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルト値は 50000 です。
13. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。



**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

14. **【次へ】** をクリックします。

## Salesforce ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むオブジェクトを指定し、Salesforce ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. 初期ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、ソースデータの取得に使用する Salesforce API のタイプを選択します。

次のオプションがあります。

- **標準 (REST) API:** Base64 データ型のソースフィールドをレプリケートします。データの初期ロード時に Base64 データ型のフィールド、または Bulk API 2.0 でサポートされていないオブジェクトを取り込む場合を除き、Bulk API 2.0 を使用することをお勧めします。すべての増分ロードアクティビティは、標準の REST API のみを使用します。
- **Bulk API 2.0:** Base64 データ型のソースフィールドはレプリケーとされません。Bulk API 2.0 は、初期ロードタスク、および初期ロードと増分ロードの組み合わせタスクにおける初期ロードのデフォルトの API です。

**注:** デフォルトでは、増分ロードタスクは、Base64 データ型のソースフィールドから変更データをキャプチャしてレプリケートできます。

2. **【オブジェクトの選択】** セクションでは、データレプリケーションのためにすべてのソースオブジェクトとソースフィールドを選択する場合のみ、**【すべて選択】** を選択します。後続のフィールドで選択内容を編集することはできません。

**【選択されたオブジェクト】** フィールドには、選択されたすべてのオブジェクトの数が表示されます。ソースオブジェクトが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースオブジェクトのサブセットを定義できます。

3. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【オブジェクトビュー】** で、選択したオブジェクトのセットをオブジェクトごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソースオブジェクトを選択します。このルールによって選択されたオブジェクトの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたオブジェクトの合計】** にオブジェクト数を表示します。次に、**【ルールの適用】** をクリックして、**【オブジェクトビュー】** でオブジェクト数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】 (+)** アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【オブジェクトルール】** フィールドで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。

- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
    - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
    - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
    - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
    - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
  - d. 必要に応じて追加のルールを定義します。  
 ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。
  - e. 終了したら、**【ルールの適用】** をクリックします。  
**ヒント: 【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるオブジェクト】** と **【選択されたオブジェクトの合計】** の数を更新します。  
 ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、オブジェクト数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【オブジェクトビュー】** リストに引き続きオブジェクトがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。
4. ルールに基づいて選択されたソースオブジェクトのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。  
 次の手順を実行して、フィールドアクションルールを作成します。
    - a. ルールタイプとして **【フィールドアクション】** を選択します。
    - b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
      - **LTRIM.**文字フィールド値の左側のスペースを切り捨てます。
      - **RTRIM.**文字フィールド値の右側のスペースを切り捨てます。
      - **TRIM.**文字フィールド値の左側と右側のスペースを切り捨てます。
    - c. 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を 1 つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したソースオブジェクトのフィールドと照合されます。
    - d. **【ルールの追加】** をクリックします。
- 注:**
- Salesforce は空の値ではなく制御文字を使用してスペースを表すため、フィールドアクションルールを使用して Salesforce のリッチテキストエリアフィールドのスペースをトリミングすることはできません。
  - 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

5. **【オブジェクトビュー】** で、選択したオブジェクト（各オブジェクトのフィールド数、フィールド名、データ型を含む）を表示します。

- **【すべて選択】** を選択した場合、オブジェクトのリストは表示のみ可能です。
- ルールを適用した場合は、個々のオブジェクトの横にあるチェックボックスをクリックして、選択したオブジェクトのセットを絞り込むことができます。レプリケートしないオブジェクトをクリアするか、レプリケートする追加のオブジェクトを選択します。**【更新】** アイコンをクリックして、選択したオブジェクトの数を更新します。
- また、選択したソースオブジェクトのフィールドを個別にクリアまたは再選択することもできます。選択したオブジェクトのデータのレプリケート元のフィールドを表示または変更するには、**【フィールド】** カラムにある、強調表示されたフィールド数をクリックします。フィールド名とデータ型が右側に表示されます。デフォルトでは、選択したソースオブジェクトのすべてのフィールドが選択されています。カラムをクリアまたは再選択するには、フィールド名の横にあるチェックボックスをクリックします。プライマリキーカラムはクリアできません。

**注:**

- オブジェクトやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【オブジェクト名】**、**【フィールド】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。
- オブジェクトのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【オブジェクトビュー】** の選択内容が優先されます。ただし、**【ルールの追加】** (+) アイコンを再度クリックすると、個別に選択解除または選択したオブジェクトが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【オブジェクトビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。
- **【稼働中】** 状態のときにカラムの名前を変更した場合、新しい名前はカラムの選択に表示されますが、**【表示】** タブでは更新されません。
- フィールドを個別に選択した場合、結果として得られるフィールドのセットはスキーマドリフトの設定に関係なく固定で、スキーマの変更によって更新されることはありません。  
  
例えば、ソースフィールドが追加または名前変更された場合、そのフィールドは選択されたフィールドのリストに含まれていないため、CDC 処理からサイレントに除外されます。ただし、選択したフィールドがソースで削除された場合、スキーマドリフトの **【フィールドの削除】** オプションによって処理方法が制御されます。削除されたフィールドの操作は、ルールを再度適用するまでフィールドのリストに反映されません。
- 初期ロードと増分ロードの組み合わせジョブの場合、フィールド選択を追加または変更し、タスクを再デプロイすると、再同期操作が自動的にトリガされます。再同期は、ターゲットテーブルがソースと同じ値を持つようにするために必要です。再同期オプションがない増分ロードジョブの場合、フィールド選択を変更するとエラーになります。デプロイしたジョブのフィールド選択を変更してから再デプロイすると、ソースとターゲットは一致しくなくなります。  
  
選択したフィールドを削除しても、再同期操作はトリガされず、エラーは報告されません。選択したフィールドの削除は、フィールド削除イベントと同じように扱われ、タスクのフィールド削除スキーマのドリフト設定がトリガされます。

6. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。

- a. **【ルールタイプ別のオブジェクトのリスト】** から、選択したソースオブジェクトのリストをダウンロードする選択ルールのタイプを選択します。
- b. リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
- c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。  
ダウンロードした情報

は、次の形式です。

*status,object\_name,object\_type,field\_name,comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"><li>- E。オブジェクトは、除外ルールによって処理から除外されます。</li><li>- I: オブジェクトは処理対象として含まれます。</li><li>- X。オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。</li></ul>
object_name	ソースオブジェクトの名前。
object_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"><li>- O: オブジェクトを示します。</li><li>- F: フィールドを示します。</li></ul>
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に【フィールドを含める】チェックボックスを選択した場合にのみ表示されます。
comment	ソースオブジェクトが選択ルールに一致していても処理から除外される理由。

7. 【詳細】 セクションを展開します。

8. 増分ロードタスクの場合は、【増分ロード操作の当初の開始点】 フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。

**注:** 日付と時刻をグリニッジ標準時（GMT）で指定する必要があります。

9. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、【CDC 間隔】 フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。

10. 【取得サイズ】 フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。初期ロード操作のデフォルト値は 50000、増分ロード操作のデフォルト値は 2000 です。

**注:** 初期ロードと増分ロードの組み合わせタスクの場合、初期ロード操作と増分ロード操作の取得サイズを別々に指定する必要があります。

11. 初期ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、アーカイブされた行と論理削除された行をデータの初期ロード中にソースからレプリケートするには、【アーカイブ済みの行と削除済みの行を含める】を選択します。

12. 初期ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、【パーティション化の有効化】を選択して、初期ロード用にソースオブジェクトをパーティション化します。【チャンクサイズ】 フィールドに、1 つのパーティションで処理するレコードの数を入力します。チャンクサイズに基づいて、Salesforce で一括処理ジョブが作成されます。デフォルト値は 50000、最小値は 100 です。

オブジェクトをパーティション化すると、アプリケーション取り込みジョブは、各パーティションのレコードを並行して処理します。一括取り込みアプリケーションは、オブジェクトのプライマリー値を均等に分散することによって、パーティションの範囲を決定します。

**注:** オブジェクトをパーティション化できるのは、Salesforce API として **【Bulk API 2.0】** を選択した場合のみです。

13. Base64 データ型のソースフィールドをレプリケートするため、**【Base64 フィールドを含める】** を選択します。

**注:**

- Base64 フィールドをレプリケートできるのは、Salesforce API として **【標準 (REST) API】** を選択した場合のみです。
- Base64 データをレプリケートすると、アプリケーション取り込みジョブの初期ロード操作が遅くなる場合があります。

14. **【Base64 フィールドを含める】** チェックボックスをオンにした場合、**【最大 Base64 ボディサイズ】** フィールドに Base64 エンコードデータのボディサイズを指定します。Base64 エンコードデータのデフォルトのボディサイズは 7 MB です。

15. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

16. **【次へ】** をクリックします。

## Salesforce Marketing Cloud ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むオブジェクトを指定し、Salesforce Marketing Cloud ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【MID】** フィールドに、Salesforce Marketing Cloud アカウントに割り当てられた一意のメンバ識別コードを入力します。
2. **【オブジェクトの選択】** セクションでは、データレプリケーションのためにすべてのソースオブジェクトとソースフィールドを選択する場合のみ、**【すべて選択】** を選択します。後続のフィールドで選択内容を編集することはできません。

**【選択されたオブジェクト】** フィールドには、選択されたすべてのオブジェクトの数が表示されます。ソースオブジェクトが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースオブジェクトのサブセットを定義できます。

3. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【オブジェクトビュー】** で、選択したオブジェクトのセットをオブジェクトごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソースオブジェクトを選択します。このルールによって選択されたオブジェクトの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたオブジェクトの合計】** にオブジェクト数を表示します。次に、**【ルールの適用】** をクリックして、**【オブジェクトビュー】** でオブジェクト数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】 (+)** アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【オブジェクトルール】** フィールドで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。

- e. 終了したら、**【ルールの適用】** をクリックします。

**ヒント: 【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるオブジェクト】** と **【選択されたオブジェクトの合計】** の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、オブジェクト数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【オブジェクトビュー】** リストに引き続きオブジェクトがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。

4. ルールに基づいて選択されたソースオブジェクトのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。

次の手順を実行して、フィールドアクションルールを作成します。

- a. ルールタイプとして **【フィールドアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
  - **LTRIM.**文字フィールド値の左側のスペースを切り捨てます。
  - **RTRIM.**文字フィールド値の右側のスペースを切り捨てます。
  - **TRIM.**文字フィールド値の左側と右側のスペースを切り捨てます。
- c. 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を 1 つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したソースオブジェクトのフィールドと照合されます。
- d. **【ルールの追加】** をクリックします。

**注:**

- Salesforce は空の値ではなく制御文字を使用してスペースを表すため、フィールドアクションルールを使用して Salesforce のリッチテキストエリアフィールドのスペースをトリミングすることはできません。

- 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

5. **【オブジェクトビュー】** で、選択したオブジェクト（各オブジェクトのフィールド数、フィールド名、データ型を含む）を表示します。

- 【すべて選択】** を選択した場合、オブジェクトのリストは表示のみ可能です。
- ルールを適用した場合は、個々のオブジェクトの横にあるチェックボックスをクリックして、選択したオブジェクトのセットを絞り込むことができます。レプリケートしないオブジェクトの選択を解除するか、レプリケートする追加のオブジェクトを選択します。**【更新】** アイコンをクリックして、選択したオブジェクトの数を更新します。

**注:** オブジェクトのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【オブジェクトビュー】** の選択内容が優先されます。ただし、**【ルールの追加】** (+) アイコンを再度クリックすると、個別に選択解除または選択したオブジェクトが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【オブジェクトビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

オブジェクトごとに、フィールド名とデータ型のリストを表示できます。**【フィールド】** カラムにある、強調表示されたフィールド数をクリックし、右側にフィールドを一覧表示します。

オブジェクトやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【オブジェクト名】**、**【フィールド】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

6. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。

- 【ルールタイプ別のオブジェクトのリスト】** から、選択したソースオブジェクトのリストをダウンロードする選択ルールのタイプを選択します。
- リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
- 【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。  
ダウンロードした情報

は、次の形式です。

*status, object\_name, object\_type, field\_name, comment*



次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E: オブジェクトは、除外ルールによって処理から除外されます。</li> <li>- I: オブジェクトは処理対象として含まれます。</li> <li>- X: オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。</li> </ul>
object_name	ソースオブジェクトの名前。
object_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: オブジェクトを示します。</li> <li>- F: フィールドを示します。</li> </ul>
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に【フィールドを含める】チェックボックスを選択した場合にのみ表示されます。
comment	ソースオブジェクトが選択ルールに一致していても処理から除外される理由。

7. 【詳細】セクションを展開します。
8. 【バッチサイズ】フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルトは 2500 です。
9. 【カスタムプロパティ】セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、【プロパティの追加】をクリックします。  
 カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。  
**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。
10. 【次へ】をクリックします。

## SAP ODP Extractor コネクタを使用した SAP ソースの設定

**アプリケーション取り込み**タスクウィザードのアプリケーション取り込みページで、取り込むデータソースを指定し、SAP ODP Extractor コネクタを使用して SAP ECC ソースまたは SAP S4/HANA ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. 【コンテキスト】リストから、ターゲットにレプリケートするソースデータソースを含むコンテキストを選択します。

SAP ODP Extractor コネクタは、すべてのロードタイプで次のような ODP プロバイダまたはコンテキストをサポートします。

プロバイダ/コンテキスト	ソース SAP システムと ODP
SAP Service Application Programming Interface (S-API)	Enterprise Search (ESH) なしの SAP Data Sources/SAP Data Extractors
HANA	SAP HANA 情報ビュー
BW	SAP NetWeaver Business Warehouse
ABAP_CDS	ABAP Core Data Services
SAP SLT	SLT キュー

2. **【データソースの選択】** セクションでは、データレプリケーションのためにすべてのソースレポートを選択する場合のみ、**【すべて選択】** を選択します。後続のディメンションとメトリックで選択内容を編集することはできません。

**【選択されたデータソース】** フィールドには、選択されたすべてのレポートの数が表示されます。ソースレポートが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースレポートのサブセットを定義できます。

3. ルールを使用してデータソースを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【データソースビュー】** で、選択したデータソースのセットをデータソースごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのデータソースを選択します。このルールによって選択されたレポートの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたデータソースの合計】** にレポート数を表示します。次に、**【ルールの適用】** をクリックして、**【データソースビュー】** でデータソース数を確認します。

ルールを追加する手順:

- 【ルール】** の下で、最初のデータソースの上にある **【ルールの追加】 (+)** アイコンをクリックします。新しいルールを定義する行が追加されます。
- 【データソースルール】** カラムで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- 条件フィールドにデータソース名またはデータソース名マスクを入力して、選択したデータソースのリストに含めるデータソース、またはリストから除外するデータソースを指定します。

**注意事項:**

- マスクには、1 つ以上の文字を表すアスタリスク (\*) ワイルドカード文字、単一の文字を表す疑問符 (?) ワイルドカード文字、または両方のタイプのワイルドカード文字を含めることができます。データソース名マスクでは、ワイルドカード文字を複数回使用できます。
- データソース名は大文字と小文字が区別されます。データソース選択ルールを定義するときは、ソースで定義されているとおりに大文字小文字を区別してデータソース名またはマスクを指定する必要があります。
- データソース名に引用符や括弧などの区切り文字が含まれている場合は、ルールでデータソース名を指定するときに区切り文字を含めないでください。

- データソース名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールでデータソース名を指定したときに名前の各特殊文字がバックスラッシュ (\) に置き換えられます。
  - d. 必要に応じて追加のルールを定義します。  
 ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。
  - e. 終了したら、**【ルールの適用】** をクリックします。  
**ヒント: 【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるデータソース】** と **【選択されたデータソースの合計】** の数を更新します。  
 ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、レポート数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【データソースビュー】** リストに引き続きレポートがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。
4. ルールに基づいて選択されたデータソースのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。  
 次の手順を実行して、フィールドアクションルールを作成します。
- a. ルールタイプとして **【フィールドアクション】** を選択します。
  - b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
    - **LTRIM.**文字フィールド値の左側のスペースを切り捨てます。
    - **RTRIM.**文字フィールド値の右側のスペースを切り捨てます。
    - **TRIM.**文字フィールド値の左側と右側のスペースを切り捨てます。
  - c. 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を 1 つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したデータソースのフィールドと照合されます。
  - d. **【ルールの追加】** をクリックします。  
**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。
5. **【データソースビュー】** で、選択したデータソース（各データソースのフィールド数を含む）を表示します。
- **【すべて選択】** を選択した場合、データソースのリストは表示のみ可能です。
  - ルールを適用した場合は、個々のデータソースの横にあるチェックボックスをクリックして、選択したデータソースのセットを絞り込むことができます。レプリケートしないデータソースの選択を解除するか、レプリケートする追加のデータソースを選択します。**【更新】** アイコンをクリックして、選択したデータソースの数を更新します。  
**注:** データソースのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【データソースビュー】** の選択内容が優先されます。ただし、**【ルールの追加】 (+)** アイコンを再度クリックすると、個別に選択解除または選択したデータソースが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【データソースビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。
- データソースごとに、フィールドのリストを表示できます。**【フィールド】** カラムにある、強調表示されたカラム数をクリックし、右側にフィールドを一覧表示します。

データカラムやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【データソース名】**、**【フィールド】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク（\*）ワイルドカードを 1 つ含めることができます。

6. 選択ルールに一致するデータソースのリストをダウンロードするには、次の手順を実行します。
  - a. **【ルールタイプ別のデータソースのリスト】** から、選択したデータソースのリストをダウンロードする選択ルールのタイプを選択します。
  - b. リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
  - c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するデータソースのリストがローカルドライブにダウンロードされます。  
ダウンロードしたファイルの情報の形式は次のとおりです。

*status,data sources\_name,data sources\_type,field\_name,comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでデータソースを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E。データソースは、除外ルールによって処理から除外されます。</li> <li>- I。データソースは処理対象として含まれます。</li> <li>- X。データソースは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、データソースが除外された詳細な理由が記述されます。</li> </ul>
data_source_name	データソースの名前。
data_source_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: データソースを示します。</li> <li>- F: フィールドを示します。</li> </ul>
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に <b>【フィールドを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	データソースが選択ルールに一致していても処理から除外される理由。

7. **【詳細】** セクションを展開します。
8. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。
 

**注:** デフォルトでは、取り込みジョブは、データストリーム内で使用可能な最新の位置から変更レコードを取得します。
9. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
 

**注:** CDC 間隔は、オペレーショナルデルタキュー（ODQ）用に SAP システムに設定されたデータ保持期間よりも短くする必要があります。
10. **【取得サイズ】** フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るデータのサイズを入力します。値はメガバイト（MB）単位で入力する必要があります。初期ロードタスクのデフォルト値は 2、初期ロードと増分ロードの組み合わせタスクのデフォルト値は 8 です。

11. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

12. **【次へ】** をクリックします。

## SAP Mass Ingestion コネクタを使用した SAP ソースの設定

**アプリケーション取り込み**タスクウィザードのアプリケーション取り込みページで、SAP Mass Ingestion コネクタを使用して SAP ECC ソース用に取り込むテーブルを指定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【テーブルの選択】** セクションでは、データレプリケーションのためにすべてのソーステーブルとソースカラムを選択する場合のみ、**【すべて選択】** を選択します。後続のカラムで選択内容を編集することはできません。

**【選択されたテーブル】** フィールドには、選択されたすべてのテーブルの数が表示されます。ソーステーブルが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソーステーブルのサブセットを定義できます。

2. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【テーブルビュー】** で、選択したテーブルのセットをテーブルごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:**

- デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソーステーブルを選択します。このルールによって選択されたテーブルの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたテーブルの合計】** にテーブル数を表示します。次に、**【ルールの適用】** をクリックして、**【テーブルビュー】** でテーブル数を確認します。
- 各オブジェクトに対して取得されたメタデータには、**NULL** データ型のダミーカラムがあります。実際のカラムを含むオブジェクトのメタデータを取得するには、Informatica グローバルカスタマサポートに問い合わせるか、**【選択ルールに一致するカラムを含める】** オプションを有効にしてソースオブジェクトのリストをダウンロードしてください。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】 (+)** アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【テーブルルール】** カラムで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。

- ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。
- ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。
- e. 終了したら、**【ルールの適用】** をクリックします。
- ヒント:** **【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるテーブル】** と **【選択されたテーブルの合計】** の数を更新します。
- ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、テーブル数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【テーブルビュー】** リストに引き続きテーブルがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。
3. ルールに基づいて選択されたソーステーブルのカラムに対してトリムアクションを実行するには、カラムアクションルールを作成します。
- 次の手順を実行して、カラムアクションルールを作成します。
- a. ルールタイプとして **【カラムアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM.**文字カラム値の左側のスペースを切り捨てます。
  - **RTRIM.**文字カラム値の右側のスペースを切り捨てます。
  - **TRIM.**文字カラム値の左側と右側のスペースを切り捨てます。
- c. 条件フィールドに、カラム名または 1 つ以上のアスタリスク (\*) または疑問符 (?) を含むカラム名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるカラムを識別するために、選択したソーステーブルのカラムと照合されます。
- d. **【ルールの追加】** をクリックします。
- 注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。カラムアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。
4. **【テーブルビュー】** で、選択したテーブル（各テーブルのカラム数を含む）を表示します。
- **【すべて選択】** を選択した場合、テーブルのリストは表示のみ可能です。
  - ルールを適用した場合は、個々のテーブルの横にあるチェックボックスをクリックして、選択したテーブルのセットを絞り込むことができます。レプリケートしないレポートの選択を解除するか、レプリケートする追加のテーブルを選択します。**【更新】** アイコンをクリックして、選択したテーブルの数を更新します。
- 注:** テーブルのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【テーブルビュー】** の選択内容が優先されます。ただし、**【ルールの追加】 (+)** アイコンを再度クリックすると、個別に選択解除または選択したテーブルが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【テーブルビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。
- テーブルごとに、カラムのリストを表示できます。**【カラム】** にある、強調表示されたカラム数をクリックし、右側にフィールドを一覧表示します。

テーブルやカラムを検索するには、**【カラム】** の上のドロップダウンリストで **【テーブル名】**、**【カラム】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

5. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。
  - a. **【ルールタイプ別のテーブルのリスト】** から、選択したテーブルのリストをダウンロードする選択ルールのタイプを選択します。
  - b. リストにカラムを含めて、**【カラムを含める】** を選択します。
  - c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。

ダウンロードしたファイルの情報の形式は次のとおりです。

status,object\_name,object\_type,field\_name,comment

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E。オブジェクトは、除外ルールによって処理から除外されます。</li> <li>- I。オブジェクトは処理対象として含まれます。</li> <li>- X。オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。</li> </ul>
table_name	テーブルの名前。 <ul style="list-style-type: none"> <li>- O。テーブルを示します。</li> <li>- F。カラムを示します。</li> <li>- V。ビューを示します。</li> </ul>
table_type	テーブルのタイプ。可能な値は次のとおりです。
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に <b>【カラムを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	テーブルが選択ルールに一致していても処理から除外される理由。

6. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

7. 増分ロードタスクの場合は、**【スキーマ】** フィールドに、ソーステーブルを含んだ基になるデータベーススキーマを入力します。スキーマ値を入力するには、次の手順を実行します。
  - a. SAP アプリケーションにログインします。
  - b. **【システム】** > **【ステータス】** を参照します。
  - c. **【所有者】** の値を確認します。この値を **【スキーマ】** フィールドに入力します。



8. SAP ソースを含む増分ロードタスクを定義しており、選択した 1 つ以上のソーステーブルで変更データキャプチャが有効になっていない場合は、CDC を有効にするためのスクリプトを生成し、そのスクリプトを実行またはダウンロードできます。
- 【CDC スクリプト】** フィールドで、**【すべてのカラムの CDC を有効化】** を選択します。これにより、選択したソーステーブルのすべてのカラムに対して CDC が有効になります。
  - スクリプトを実行するには、**【実行】** をクリックします。  
スクリプトを実行できるデータベースロールまたは特権がない場合は、**【ダウンロード】** アイコンをクリックしてスクリプトをダウンロードします。スクリプトファイル名には、`cdc_script_taskname_number.txt` という形式が使用されます。次に、データベース管理者にスクリプトの実行を依頼します。  
データベース取り込みタスクを実行する前に、スクリプトが実行されていることを確認してください。
9. テーブル選択条件に一致するソーステーブルのリストを作成してダウンロードするには、次のサブ手順を実行します。
- ルールベースのテーブル選択を使用した場合は、**【テーブル名】** リストで、使用する選択ルールのタイプを選択します。次のようなオプションがあります。
    - **含めるルールのみ**
    - **除外ルールのみ**
    - **含めるルールと除外ルール**
  - 使用したテーブル選択方法に関係なく、カラムを一覧表示するには、**【カラムを含める】** チェックボックスを選択します。
  - 【ダウンロード】** アイコンをクリックします。  
カラムを含むダウンロードされたリストには、次の形式が使用されます。  
`status, schema_name, table_name, object_type, column_name, comment`  
次の表に、ダウンロードしたリストに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションにサポートされていないタイプがある場合に、ソーステーブルまたはカラムを処理から除外するかどうかを示します。有効な値は以下のとおりです。 <ul style="list-style-type: none"> <li>- E。オブジェクトは、除外ルールによって処理から除外されます。</li> <li>- I。オブジェクトは処理に含まれます。</li> <li>- X。このオブジェクトはサポートされていないタイプのオブジェクトであるため、処理から除外されます。例えば、サポートされていないタイプのオブジェクトには、サポートされていないデータ型のカラムと、サポートされていないカラムのみを含むテーブルが含まれます。コメントフィールドには、サポートされていないタイプに関する情報が表示されます。</li> </ul>
schema_name	ソーススキーマの名前を指定します。
table_name	ソーステーブルの名前を指定します。
object_type	ソースオブジェクトのタイプを指定します。有効な値は以下のとおりです。 <ul style="list-style-type: none"> <li>- C。カラム。</li> <li>- T。テーブル。</li> </ul>

フィールド	説明
column_name	ソースカラムの名前を指定します。この情報は、 <b>[カラム]</b> チェックボックスを選択した場合にのみ表示されます。
comment	サポートされていないタイプのソースオブジェクトが、選択ルールに一致していても処理から除外される理由を指定します。

10. **【詳細】** で、ソースタイプとロードタイプに使用できる詳細プロパティを設定します。

プロパティ	ソースとロードタイプ	説明
増分ロード操作の当初の開始点	SAP Mass Ingestion コネクタを使用した SAP ソース - 増分ロード	アプリケーション取り込みジョブが初めて実行されたときに変更レコードの読み取りを開始するソースログ内の位置をカスタマイズするには、このフィールドを設定します。 <b>【使用可能な最新】</b> オプションは、データベースログまたは構造内の最新の使用可能な位置を提供します。

11. **【次へ】** をクリックします。

#### スキーマ変更処理

一部のデータオブジェクトタイプに対して行われたスキーマ変更を SAP Mass Ingestion コネクタで処理する方法を選択できます。

#### SAP S4 HANA ソースのスキーマ変更処理

デフォルトでは、スキーマに変更を加えると、SAP Mass Ingestion コネクタによってスキーマの変更が検出され、警告メッセージが生成されます。

**注:** スキーマ検出は、カスタムプロパティを使用して無効にできます。詳細については、Informatica グローバルカスタマサポートにお問い合わせください。

スキーマの変更には、データオブジェクトに対する次の 1 つ以上の変更が含まれます。

- フィールドの追加
- フィールドの削除
- フィールドの名前の変更
- フィールドのデータ型、精度、またはスケールの更新

スキーマの変更は、次の条件に基づいて処理できます。

- 増分ロードジョブが **【稼働中】** ステータスのときにデータオブジェクトにスキーマ変更を行うと、SAP Mass Ingestion コネクタは警告メッセージを生成します。警告が表示されると、データオブジェクトの処理が停止し、変更されたデータオブジェクトを使用して新しい初期ジョブを作成する必要があります。初期ジョブが正常に完了した後、新しい増分ジョブを実行します。
- 増分ロードジョブが **【停止】** ステータスのときにデータオブジェクトにスキーマ変更を行った場合は、変更が行われたデータオブジェクトのトリガとシャドウテーブルを削除する必要があります。SAP のユーザーインターフェースを使用してデータオブジェクトをアクティブ化し、更新されたデータオブジェクトを使用して新しい初期ジョブを作成します。初期ジョブが正常に完了したら、新しい増分ジョブを実行します。

**注:** データの損失を防ぐために、初期ジョブが正常に完了した直後に増分ジョブを実行します。

## SAP ECC ソースのスキーマ変更処理

データオブジェクトにスキーマ変更を行うと、データオブジェクトの処理が停止し、SAP Mass Ingestion コネクタによって警告メッセージが生成されます。データオブジェクトに対するそれ以降の変更はキャプチャされません。

スキーマの変更には、データオブジェクトに対する次の 1 つ以上の変更が含まれます。

- フィールドの追加
- フィールドの削除
- フィールドの名前の変更
- フィールドのデータ型、精度、またはスケールの更新

スキーマの変更は、次の条件に基づいて処理できます。

- 組み合わせロードジョブのデータオブジェクトにスキーマ変更を行うと、SAP Mass Ingestion コネクタによって警告メッセージが生成されます。その後、データオブジェクトを再同期できます。
- 増分ロードジョブのデータオブジェクトにスキーマ変更を行った場合は、変更されたデータオブジェクトを使用して初期ジョブを作成し、実行する必要があります。初期ジョブが正常に完了したら、変更されたデータオブジェクトを使用して新しい増分ジョブを実行します。

**注:** データの損失を防ぐために、初期ジョブが正常に完了した直後に増分ジョブを実行します。

## ServiceNow ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むテーブルを指定し、ServiceNow ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【テーブルの選択】** セクションでは、データレプリケーションのためにすべてのソーステーブルとソースカラムを選択する場合のみ、**【すべて選択】** を選択します。後続のカラムで選択内容を編集することはできません。

**【選択されたテーブル】** フィールドには、選択されたすべてのテーブルの数が表示されます。ソーステーブルが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソーステーブルのサブセットを定義できます。

2. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【テーブルビュー】** で、選択したテーブルのセットをテーブルごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソーステーブルを選択します。このルールによって選択されたテーブルの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたテーブルの合計】** にテーブル数を表示します。次に、**【ルールの適用】** をクリックして、**【テーブルビュー】** でテーブル数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】** (+) アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【テーブルルール】** カラムで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。

- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための1つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
- マスクには、次のワイルドカードの1つまたは両方を含めることができます: 1つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。
- ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。
- e. 終了したら、**【ルールの適用】** をクリックします。
- ヒント: 【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるテーブル】** と **【選択されたテーブルの合計】** の数を更新します。
- ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、テーブル数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【テーブルビュー】** リストに引き続きテーブルがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。
3. ルールに基づいて選択されたソーステーブルのカラムに対してトリムアクションを実行するには、カラムアクションルールを作成します。
- 次の手順を実行して、カラムアクションルールを作成します。
- a. ルールタイプとして **【カラムアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM.**文字カラム値の左側のスペースを切り捨てます。
  - **RTRIM.**文字カラム値の右側のスペースを切り捨てます。
  - **TRIM.**文字カラム値の左側と右側のスペースを切り捨てます。
- c. 条件フィールドに、カラム名または1つ以上のアスタリスク (\*) または疑問符 (?) を含むカラム名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるカラムを識別するために、選択したソーステーブルのカラムと照合されます。
- d. **【ルールの追加】** をクリックします。
- 注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。カラムアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。
4. **【テーブルビュー】** で、選択したテーブル（各テーブルのカラム数を含む）を表示します。
- **【すべて選択】** を選択した場合、テーブルのリストは表示のみ可能です。

- ルールを適用した場合は、個々のテーブルの横にあるチェックボックスをクリックして、選択したテーブルのセットを絞り込むことができます。レプリケートしないレポートの選択を解除するか、レプリケートする追加のテーブルを選択します。[更新] アイコンをクリックして、選択したテーブルの数を更新します。

**注:** テーブルのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。[テーブルビュー] の選択内容が優先されます。ただし、[ルールの追加] (+) アイコンを再度クリックすると、個別に選択解除または選択したテーブルが、新しいルールとして [ルール] リストに反映され、ルールが再び優先されます。[テーブルビュー] リストに戻るには、[ルールの適用] を再度クリックします。

テーブルごとに、カラムのリストを表示できます。[カラム] にある、強調表示されたカラム数をクリックし、右側にフィールドを一覧表示します。

テーブルやカラムを検索するには、[カラム] の上のドロップダウンリストで [テーブル名]、[カラム]、または [すべて] を選択し、[検索] ボックスに検索文字列を入力して [検索] をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

5. 選択ルールに一致するソーステーブルのリストをダウンロードするには、次の手順を実行します。
  - a. [ルールタイプ別のテーブルのリスト] から、選択したソーステーブルのリストをダウンロードする選択ルールのタイプを選択します。
  - b. リストにカラムを含める場合は、[カラムを含める] を選択します。
  - c. [ダウンロード] アイコンをクリックします。

選択ルールに一致するソーステーブルのリストがローカルドライブにダウンロードされます。ダウンロードしたファイルの情報の形式は次のとおりです。

*status, table\_name, table\_type, column\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソーステーブルを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E: テーブルは、除外ルールによって処理から除外されます。</li> <li>- I: テーブルは処理対象として含まれます。</li> <li>- X: テーブルは選択ルールに一致していても処理から除外されます。ファイルの [comment] フィールドに、テーブルが除外された詳細な理由が記述されます。</li> </ul>
table_name	ソーステーブルの名前。
table_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: テーブルを示します。</li> <li>- F: カラムを示します。</li> </ul>
column_name	ソースカラムの名前。この情報は、リストをダウンロードする前に [カラムを含める] チェックボックスを選択した場合にのみ表示されます。
comment	ソーステーブルが選択ルールに一致していても処理から除外される理由。

6. 増分ロードタスクの場合は、[増分ロード操作の当初の開始点] フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。

**注:** 日付と時刻をグリニッジ標準時 (GMT) で指定する必要があります。

7. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
8. **【取得サイズ】** フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルトは 10000 です。
9. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。  
カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。  
**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。
10. **【次へ】** をクリックします。

## Workday ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込む操作を指定し、Workday ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【Workday API】** リストで、ソースデータの読み取りに使用する Web サービスの種類を選択します。  
次のオプションがあります。
  - **SOAP:** SOAP API を使用して Workday データを抽出します。
  - **RaaS:** Workday Report-as-a-Service (RaaS) を使用して、ソースデータをカスタムレポートのカスタムのオブジェクトとフィールドから抽出します。Workday RaaS は、初期ロードタスクでのみ使用できます。
2. SOAP API の使用を選択した場合は、次の手順を実行します。
  - a. **【製品】** リストで、**【Human Capital Management】** を選択します。
  - b. **【サービス】** リストで、ターゲットにデータを取り込む Human Capital Management (HCM) サービスを選択します。  
**【サービス】** リストから複数のサービスを選択できます。
  - c. **【出力タイプ】** リストで、データをターゲットに保存する形式を選択します。  
取り込みジョブは、XML 構造でソースデータを抽出します。選択した形式に基づいて、抽出されたデータが JSON または XML 形式の単一オブジェクトとしてターゲットに書き込まれます。
3. RaaS API を使用する場合は、次の手順を実行します。
  - a. **【レポートの数】** フィールドで、ソースから抽出するレポートの数を選択します。
  - b. 単一のレポートを抽出する場合は、**【レポート名または URL】** フィールドに、ソースから読み取るカスタムレポートの名前または URL を入力します。
  - c. 複数のレポートを抽出する場合は、**【レポート構成ファイル】** フィールドに、ソースから読み取るカスタムレポートのリスト用に作成した CSV ファイルへのパスを入力します。
4. **【操作の選択】** セクションでは、データレプリケーションのためにすべての操作を選択する場合のみ、**【すべて選択】** を選択します。後続の操作で選択内容を編集することはできません。  
**【選択された操作】** フィールドには、選択されたすべての操作の数が表示されます。ソース操作が多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。  
または、ルールを使用して、レプリケートするソース操作のサブセットを定義できます。
5. ルールを使用してソース操作を選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【操作ビュー】**で、選択した操作のセットを操作ごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべての操作を選択します。このルールによって選択された操作の数を確認するには、**【更新】**アイコンをクリックして、**【選択された操作の合計】**に操作数を表示します。次に、**【ルールの適用】**をクリックして、**【操作ビュー】**で操作数を確認します。

ルールを追加する手順:

- a. **【ルール】**の下で、最初の操作の上にある**【ルールの追加】**(+)アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【操作ルール】**カラムで、**【含む】**または**【除外】**を選択して、包含ルールまたは除外ルールを作成します。
- c. 条件フィールドに操作名を入力して、選択した操作ソースのリストに含める、またはリストから除外する操作を指定します。

**注意事項:**

- マスクには、1つ以上の文字を表すアスタリスク(\*)ワイルドカード文字、単一の文字を表す疑問符(?)ワイルドカード文字、または両方のタイプのワイルドカード文字を含めることができます。データソース名マスクでは、ワイルドカード文字を複数回使用できます。
  - データソース名は大文字と小文字が区別されます。データソース選択ルールを定義するときは、ソースで定義されているとおりに大文字小文字を区別してデータソース名またはマスクを指定する必要があります。
  - データソース名に引用符や括弧などの区切り文字が含まれている場合は、ルールでデータソース名を指定するときに区切り文字を含めないでください。
  - データソース名にバックスラッシュ(\)、アスタリスク(\*)、ドル記号(\$)、キャレット(^)、疑問符(?)などの特殊文字が含まれている場合は、ルールでデータソース名を指定したときに名前の各特殊文字がバックスラッシュ(\)に置き換えられます。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。

- e. 終了したら、**【ルールの適用】**をクリックします。

**ヒント:** **【更新済み】**タイムスタンプの右側にある**【更新】**アイコンをクリックして、**【影響を受ける操作】**と**【選択された操作の合計】**の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】**を再度クリックする必要があります。**【更新】**アイコンをクリックして、操作数を更新します。**【ルールの適用】**をクリックしないですべてのルールを削除すると、**【操作ビュー】**リストに引き続き操作がリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】**に切り替えると、ルールは表示されなくなります。

6. **【操作ビュー】**で、選択した操作（各操作の属性数を含む）を表示します。

- **【すべて選択】**を選択した場合、操作のリストは表示のみ可能です。
- ルールを適用した場合は、個々の操作の横にあるチェックボックスをクリックして、選択した操作のセットを絞り込むことができます。レプリケートしない操作の選択を解除するか、レプリケートする追加の操作を選択します。**【更新】**アイコンをクリックして、選択した操作の数を更新します。

**注:** 操作のチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【操作ビュー】**の選択内容が優先されます。ただし、**【ルールの追加】**(+)アイコンを再度クリックすると、個別に選択解除または選択した操作が、新しいルールとして**【ルール】**リストに反映され、ルールが再び優先されます。**【操作ビュー】**リストに戻るには、**【ルールの適用】**を再度クリックします。



操作ごとに、属性のリストを表示できます。【属性】 カラムにある、強調表示された属性数をクリックし、右側に属性を一覧表示します。

操作や属性を検索するには、【属性】 の上のドロップダウンリストで【操作名】、【属性】、または【すべて】を選択し、[検索] ボックスに検索文字列を入力して【検索】をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

7. 選択ルールに一致するソース操作のリストをダウンロードするには、次の手順を実行します。
  - a. 【ルールタイプ別の操作のリスト】 から、選択したソース操作のリストをダウンロードする選択ルールのタイプを選択します。
  - b. [ダウンロード] アイコンをクリックします。

選択ルールに一致するソース操作のリストがローカルドライブにダウンロードされます。

ダウンロードしたファイルの情報の形式は次のとおりです。

*status,operation\_name,operation\_type,comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソース操作を処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"><li>- E。操作は、除外ルールによって処理から除外されます。</li><li>- I: 操作は処理対象として含まれます。</li><li>- X。操作は選択ルールに一致していても処理から除外されます。ファイルの [comment] フィールドに、操作が除外された詳細な理由が記述されます。</li></ul>
operation_name	ソース操作の名前。
operation_type	ソースオブジェクトのタイプ。このフィールドの値 0 は、オブジェクトが操作であることを示します。
comment	ソース操作が選択ルールに一致していても処理から除外される理由。

8. 増分ロードタスクの場合は、【増分ロード操作の当初の開始点】 フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。

**注:** 日付と時刻を協定世界時 (UTC) で指定する必要があります。

9. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、【CDC 間隔】 フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。

10. 【取得サイズ】 フィールドに、タスクに関連付けられたアプリケーション取り込みジョブがソースから一度に読み取るレコードの数を入力します。デフォルトは 100 です。

**注:** 【取得サイズ】 フィールドは、SOAP API の場合のみ表示されます。

11. デフォルト値を含まないソースフィールドをレプリケートするには、【デフォルト以外のフィールドを抽出】 チェックボックスをオンにします。

**注:** 【デフォルト以外のフィールドを抽出】 チェックボックスは、SOAP API の場合のみ表示されます。

12. 【カスタムプロパティ】 セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、【プロパティの追加】をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

13. **【次へ】** をクリックします。

## Zendesk ソースの設定

**アプリケーション取り込みタスクウィザード**のアプリケーション取り込みページで、取り込むオブジェクトを指定し、Zendesk ソースの詳細プロパティを設定できます。カスタムプロパティを指定して、固有な環境と特別な使用例に対応することもできます。

1. **【オブジェクトの選択】** セクションでは、データレプリケーションのためにすべてのソースオブジェクトとソースフィールドを選択する場合のみ、**【すべて選択】** を選択します。後続のフィールドで選択内容を編集することはできません。

**【選択されたオブジェクト】** フィールドには、選択されたすべてのオブジェクトの数が表示されます。ソースオブジェクトが多数ある場合、インタフェースではそれらの取得に時間がかかることがあります。

または、ルールを使用して、レプリケートするソースオブジェクトのサブセットを定義できます。

2. ルールを使用してソースオブジェクトを選択するには、**【すべて選択】** チェックボックスをオフにしてから、ルールを追加します。

ルールベースの選択を使用する際は、**【オブジェクトビュー】** で、選択したオブジェクトのセットをオブジェクトごとに絞り込み、文字データ内のスペースをトリミングするオプションを設定できます。

**注:** デフォルトの「Include \*」ルールは、選択した接続でアクセスされるすべてのソースオブジェクトを選択します。このルールによって選択されたオブジェクトの数を確認するには、**【更新】** アイコンをクリックして、**【選択されたオブジェクトの合計】** にオブジェクト数を表示します。次に、**【ルールの適用】** をクリックして、**【オブジェクトビュー】** でオブジェクト数を確認します。

ルールを追加する手順:

- a. **【ルール】** の下で、最初のテーブルの上にある **【ルールの追加】 (+)** アイコンをクリックします。新しいルールを定義する行が追加されます。
- b. **【オブジェクトルール】** フィールドで、**【含む】** または **【除外】** を選択して、包含ルールまたは除外ルールを作成します。
- c. **【条件】** カラムに、オブジェクト選択に含める、またはオブジェクト選択から除外するソースオブジェクトを識別するための 1 つ以上のワイルドカードを含むオブジェクト名またはオブジェクト名マスクを入力します。次のガイドラインを使用します。
  - マスクには、次のワイルドカードの 1 つまたは両方を含めることができます: 1 つ以上の文字を表すアスタリスク (\*) ワイルドカードと単一の文字を表す疑問符 (?) ワイルドカード。ワイルドカードは、マスク値内で複数回使用することができ、値内のどこでも使用できます。
  - タスクウィザードでは大文字と小文字が区別されます。オブジェクトが定義された際の指定どおりに大文字小文字を区別してオブジェクト名またはマスクを入力します。
  - ソースで使用されている場合でも、引用符や括弧などの区切り文字は含めないでください。
  - オブジェクト名にバックスラッシュ (\)、アスタリスク (\*)、ドル記号 (\$)、キャレット (^)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを入力するときに各特殊文字をバックスラッシュ (\) でエスケープします。
- d. 必要に応じて追加のルールを定義します。

ルールは、リストされている順序で上から下に処理されます。矢印アイコンを使用して順序を変更します。

- e. 終了したら、**【ルールの適用】** をクリックします。

**ヒント:** **【更新済み】** タイムスタンプの右側にある **【更新】** アイコンをクリックして、**【影響を受けるオブジェクト】** と **【選択されたオブジェクトの合計】** の数を更新します。

ルールの適用後にルールを追加、削除、または変更する場合は、**【ルールの適用】** を再度クリックする必要があります。**【更新】** アイコンをクリックして、オブジェクト数を更新します。**【ルールの適用】** をクリックしないですべてのルールを削除すると、**【オブジェクトビュー】** リストに引き続きオブジェクトがリストされていても、デプロイ時に検証エラーが発生します。**【すべて選択】** に切り替えると、ルールは表示されなくなります。

3. ルールに基づいて選択されたソースオブジェクトのフィールドに対してトリミングアクションを実行するには、フィールドアクションルールを作成します。

次の手順を実行して、フィールドアクションルールを作成します。

- a. ルールタイプとして **【フィールドアクション】** を選択します。
- b. 隣接するリストから、次のいずれかのアクションタイプを選択します。
- **LTRIM**。文字フィールド値の左側のスペースを切り捨てます。
  - **RTRIM**。文字フィールド値の右側のスペースを切り捨てます。
  - **TRIM**。文字フィールド値の左側と右側のスペースを切り捨てます。

- c. 条件フィールドに、フィールド名、またはアスタリスク (\*) または疑問符 (?) を 1 つ以上含むフィールド名マスクを入力します。ワイルドカード。入力した値は、アクションが適用されるフィールドを識別するために、選択したソースオブジェクトのフィールドと照合されます。

- d. **【ルールの追加】** をクリックします。

**注:** 異なるアクションタイプに対して、あるいは条件が異なる同じアクションタイプに対して複数のルールを定義できます。フィールドアクションルールは、**【ルール】** リストに表示されている順序で処理されます。リストの一番上にあるルールが最初に処理されます。矢印アイコンを使用して、ルールが表示される順序を変更できます。

4. **【オブジェクトビュー】** で、選択したオブジェクト（各オブジェクトのフィールド数、フィールド名、データ型を含む）を表示します。

- **【すべて選択】** を選択した場合、オブジェクトのリストは表示のみ可能です。
- ルールを適用した場合は、個々のオブジェクトの横にあるチェックボックスをクリックして、選択したオブジェクトのセットを絞り込むことができます。レプリケートしないオブジェクトの選択を解除するか、レプリケートする追加のオブジェクトを選択します。**【更新】** アイコンをクリックして、選択したオブジェクトの数を更新します。

**注:** オブジェクトのチェックボックス設定を初めて変更する際、ルールは有効ではなくなっています。**【オブジェクトビュー】** の選択内容が優先されます。ただし、**【ルールの追加】** (+) アイコンを再度クリックすると、個別に選択解除または選択したオブジェクトが、新しいルールとして **【ルール】** リストに反映され、ルールが再び優先されます。**【オブジェクトビュー】** リストに戻るには、**【ルールの適用】** を再度クリックします。

オブジェクトごとに、フィールド名とデータ型のリストを表示できます。**【フィールド】** カラムにある、強調表示されたフィールド数をクリックし、右側にフィールドを一覧表示します。

オブジェクトやフィールドを検索するには、**【フィールド】** の上のドロップダウンリストで **【オブジェクト名】**、**【フィールド】**、または **【すべて】** を選択し、**【検索】** ボックスに検索文字列を入力して **【検索】** をクリックします。文字列の先頭または末尾にアスタリスク (\*) ワイルドカードを 1 つ含めることができます。

5. 選択ルールに一致するソースオブジェクトのリストをダウンロードするには、次の手順を実行します。

- a. **【ルールタイプ別のオブジェクトのリスト】** から、選択したソースオブジェクトのリストをダウンロードする選択ルールのタイプを選択します。

- b. リストにフィールドを含める場合は、**【フィールドを含める】** を選択します。
- c. **【ダウンロード】** アイコンをクリックします。

選択ルールに一致するソースオブジェクトのリストがローカルドライブにダウンロードされます。  
ダウンロードした情報

は、次の形式です。

*status, object\_name, object\_type, field\_name, comment*

次の表に、ダウンロードしたファイルに表示される情報を示します。

フィールド	説明
status	一括取り込みアプリケーションでソースオブジェクトを処理対象として含めるか、処理から除外するかを示します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- E: オブジェクトは、除外ルールによって処理から除外されます。</li> <li>- I: オブジェクトは処理対象として含まれます。</li> <li>- X: オブジェクトは選択ルールに一致していても処理から除外されます。ファイルの[comment]フィールドに、オブジェクトが除外された詳細な理由が記述されます。</li> </ul>
object_name	ソースオブジェクトの名前。
object_type	ソースオブジェクトのタイプ。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>- O: オブジェクトを示します。</li> <li>- F: フィールドを示します。</li> </ul>
field_name	ソースフィールドの名前。この情報は、リストをダウンロードする前に <b>【フィールドを含める】</b> チェックボックスを選択した場合にのみ表示されます。
comment	ソースオブジェクトが選択ルールに一致していても処理から除外される理由。

6. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【詳細】** セクションを展開します。
7. 増分ロードタスクの場合は、**【増分ロード操作の当初の開始点】** フィールドで、アプリケーション取り込みタスクに関連付けられた取り込みジョブが変更レコードの抽出を開始するソースデータストリーム内の時点を指定します。

**注:** 日付と時刻を協定世界時（UTC）で指定する必要があります。

8. 増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、**【CDC 間隔】** フィールドで、アプリケーション取り込みジョブが増分ロードで変更レコードを取得する時間間隔を指定します。デフォルトの間隔は 5 分です。
9. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、プロパティ名と値を追加し、**【プロパティの追加】** をクリックします。

カスタムプロパティは、通常は固有の環境と特別な使用例に対応するために設定します。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

10. **【次へ】** をクリックします。

## ソースオブジェクトを選択するルールの例

アプリケーション取り込みタスクのソースを定義するときに、オブジェクト選択ルールを定義して、ターゲットにロードするソースオブジェクトを選択できます。次の例は、選択ルールを使用して必要なオブジェクトを選択する方法を示しています。

### 例

ソースには、プレフィックスが異なる 1,000 個のオブジェクトがあります。プレフィックスが「2021\_SALES」のオブジェクトと、「2021\_」を除く他のプレフィックスを持つすべてのオブジェクトを選択します。

次のルールをリストされている順序で定義します。

- ルール 1: **ルールアクション=含めると条件=\***で、ソース上のすべてのオブジェクトを含めます。アスタリスク (\*) ワイルドカードのみを指定すると、ソース上のすべてのオブジェクトが選択されます。
- ルール 2: **ルールアクション=除外と条件=2021\_\***で、プレフィックスが「2021\_」の名前を持つソースオブジェクトを除外します。
- ルール 3: **ルールアクション=含めると条件=2021\_SALES\***で、プレフィックスが「2021\_SALES」の名前を持つソースオブジェクトを含めます。

次の画像は、[ソース] ページの [オブジェクト選択ルール] セクションのルールを示しています。

▼ Object Selection Rules

Create Rule: 

Object Selection ▼

Include ▼

Add Rule

Object Rules ⓘ 

Object Count | ⌵ ⌶

Action	Condition	Objects Affected
Include	*	
Exclude	2021_*	
Include	2021_SALES*	

## ターゲットの設定

アプリケーション取り込みタスクウィザードの [ターゲット] ページでターゲットを設定できます。

ターゲットを設定する前に、組織が使用するランタイム環境のターゲットへの接続が Administrator で作成されていることを確認してください。

1. **【接続】** リストから、ソースアプリケーション用に設定された接続を選択します。接続タイプは、接続名の後に括弧で囲まれて表示されます。

リストには、**【定義】** ページで選択したロードタイプに有効な接続のみが含まれます。

**注:** 取り込みタスクをデプロイした後は、関連する取り込みジョブをデプロイ解除せずに接続を変更することはできません。接続を変更した後、タスクを再度デプロイする必要があります。

2. ターゲットプロパティを設定します。

ターゲットプロパティの説明については、次のトピックを参照してください。

- [「Amazon Redshift ターゲットのプロパティ」 \(ページ 101\)](#)
- [「Amazon S3 ターゲットのプロパティ」 \(ページ 102\)](#)
- [「Databricks Delta ターゲットのプロパティ」 \(ページ 106\)](#)
- [「Google BigQuery ターゲットのプロパティ」 \(ページ 108\)](#)
- [「Google Cloud Storage ターゲットのプロパティ」 \(ページ 110\)](#)

- [「Kafka ターゲットのプロパティ」 \(ページ 114\)](#)
  - [「Microsoft Azure Data Lake Storage Gen2 ターゲットのプロパティ」 \(ページ 117\)](#)
  - [「Microsoft Azure Synapse Analytics ターゲットのプロパティ」 \(ページ 121\)](#)
  - [「Microsoft SQL Server ターゲットのプロパティ」 \(ページ 124\)](#)
  - [「Microsoft Fabric OneLake ターゲットプロパティ」 \(ページ 122\)](#)
  - [「Oracle Cloud Object Storage ターゲットのプロパティ」 \(ページ 124\)](#)
  - [「Oracle ターゲットのプロパティ」 \(ページ 128\)](#)
  - [「PostgreSQL ターゲットプロパティ」 \(ページ 129\)](#)
  - [「Snowflake ターゲットのプロパティ」 \(ページ 129\)](#)
3. 選択したソースオブジェクトに関連付けられているターゲットオブジェクトの名前を変更する場合は、テーブルの名前変更ルールを定義します。
- テーブルの名前変更ルールの詳細については、「[「ターゲット上のテーブルの名前を変更するためのルール」 \(ページ 100\)](#)」を参照してください。
4. ソースデータ型からターゲットデータ型へのデフォルトのマッピングをオーバーライドする場合は、**【データ型ルール】** セクションで次の手順を実行してデータ型ルールを定義します。
- a. **【ルールの作成】** フィールドに、マッピングをカスタマイズするソースデータ型を入力してから、ソースデータ型にマッピングするターゲットデータ型を入力します。
- 重要:**
- 一括取り込みアプリケーションは、データ型マッピングルールの BYTE および CHAR セマンティクスをサポートしていません。
  - ソースデータ型にデフォルト値がある場合は、ルールで指定する必要があります。
- b. **【ルールの追加】** をクリックします。
- ルールが作成され、ルールリストに表示されます。
- 注:**
- 1 つのターゲットに対して複数のデータ型マッピングルールを定義できます。ただし、1 つのソースデータ型に対して定義できるルールは 1 つだけです。
  - カスタムマッピングルールを使用してタスクをデプロイした後は、タスクをデプロイ解除せずにルールを編集することはできません。
5. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、**【プロパティの追加】** アイコンをクリックしてから、プロパティの名前と値を追加します。
- カスタムプロパティは、固有の環境と特別な使用例に対応するように設定されています。
- 注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。
6. **【次へ】** をクリックします。

## ターゲット上のテーブルの名前を変更するためのルール

既存のスキーマを使用してターゲットを設定する場合、オプションで、選択したソースオブジェクトに対応するターゲットテーブルの名前を変更するためのルールを定義できます。

テーブルの名前を変更するためのルールを作成するには、**【テーブルの名前変更ルール】** セクションで次の手順を実行します。

1. **【ルールの作成】** フィールドに、名前を変更するソースオブジェクトの名前を入力してから、そのオブジェクトに対応するターゲットテーブルに割り当てる名前を入力します。

### 注意事項:

- アスタリスク (\*) ワイルドカード文字のみを入力すると、**【ソース】** ページで定義されている選択基準に一致するすべてのソースオブジェクトを選択できます。あるいは、特定のソースオブジェクトの名前、またはアスタリスク (\*) ワイルドカード文字を含むオブジェクト名パターンを入力することもできます。
- オブジェクト名またはテーブル名にバックスラッシュ (\)、アスタリスク (\*)、ドット (.)、疑問符 (?) などの特殊文字が含まれている場合は、ルールを作成するときに名前の各特殊文字をバックスラッシュ (\) に置き換えます。
- ターゲットテーブルにワイルドカード文字を含むテーブル名パターンを使用する場合は、対応するソースオブジェクトの名前にもワイルドカード文字を使用する必要があります。
- Windows では、ターゲットテーブルの名前変更条件を入力することによってターゲットテーブル名の長さが 232 文字を超える場合、名前は 222 文字に切り詰められます。一括取り込みアプリケーションでは、名前に日時の yyyyMMddHHmmss 値 (14 文字) が追加されるため、名前は Windows の上限である 255 文字を超えてしまいます。名前を変更するターゲットテーブルの名前が 232 文字を超えないようにしてください。

2. **【ルールの追加】** をクリックします。  
ルールが作成され、ルールリストに表示されます。

複数のテーブルの名前変更ルールを定義できます。テーブルが複数のルールに一致していない限り、ルールが処理される順序は、**【テーブルの名前変更ルール】** セクションにリストされている順序に依存しません。テーブルが複数のルールに一致する場合、最後に一致するルールによってテーブルの名前が決まります。

ルールを削除するには、ルールを含む行にある **【削除】** アイコンをクリックします。

### 例

選択したすべてのソースオブジェクトに関連付けられているターゲットテーブルの名前にプレフィックス「PROD\_」を追加します。**【テーブルの名前変更ルール】** セクションの **【ルールの作成】** フィールドに、次の値を入力します。

- ソースには、アスタリスク (\*) ワイルドカード文字を入力して、**【ソース】** ページで定義されているオブジェクト選択ルールに一致するすべてのソースオブジェクトを指定します。
- ターゲットには、「PROD\_\*」と入力して、ソースオブジェクトに対応するすべてのターゲットテーブルの名前にこのプレフィックスを追加します。

## データ型マッピングをカスタマイズするためのルール

アプリケーション取り込みタスクのターゲットを設定する場合、必要に応じてデータ型マッピングルールを定義して、ソースデータ型からターゲットデータ型へのデフォルトのマッピングをオーバーライドできます。

例えば、Snowflake CHAR(72)データ型へのデフォルトのマッピングを使用する代わりに、精度のない Salesforce ID フィールドを、同じく精度のない Snowflake ターゲットの NUMBER() カラムにマッピングするデータ型ルールを作成できます。



データ型マッピングルールを作成するには、次の手順を実行します。

1. **【データ型ルール】** を展開します。
2. **【ルールの作成】** フィールドに、ソースデータ型とそれをマッピングするターゲットデータ型を入力します。  
**【ソース】** フィールドでは、例えば NUMBER(%、4)、NUMBER(8、%)、NUMBER(%)のように、パーセント(%) ワイルドカードを使用してデータ型の精度、スケール、またはサイズを表すことができます。データ型は同じで精度、スケール、またはサイズ値が異なるソースフィールドを個別に指定する代わりに、ワイルドカードを使用してすべてのソースフィールドを指定できます。例えば、FLOAT(16)、FLOAT(32)、FLOAT(84)をカバーするには、FLOAT(%)と入力します。ターゲットデータ型に%ワイルドカードを入力することはできません。%ワイルドカードを使用するソースデータ型は、特定の精度、スケール、またはサイズ値を使用するターゲットデータ型にマッピングする必要があります。例えば、ソースデータ型 FLOAT(%)を NUMBER(38,10)などのターゲットデータ型の仕様にマッピングできます。
3. **【ルールの追加】** をクリックします。  
ルールがルールリストに表示されます。

カスタムマッピングルールを使用してタスクをデプロイした後は、タスクがデプロイ解除されるまでルールを編集できません。

**注:**

- 一般に、バイナリデータ型は文字データ型にマッピングできません。
- 長さまたは精度とスケール値が同じソースデータ型に対して複数のデータ型ルールを定義すると、アプリケーション取り込みタスクを保存できなくなります。
- 同じソースデータ型に対して複数のデータ型ルールを定義したが、%ワイルドカードを使用して、1つのルールで長さまたは精度とスケール値を表し、2番目のルールで特定の長さまたは精度とスケール値を表す場合、%ワイルドカードを使用したルールの前に、特定の値を含むルールが最初に処理されます。例えば、ソースデータ型 FLOAT(84)と FLOAT(%)をマッピングする場合、FLOAT(84)ルールが最初に処理され、次に FLOAT(%)ルールが処理されて、サイズの異なる他の FLOAT ソースカラムがカバーされます。
- ソースデータ型に長さまたは精度とスケール値が必要な場合は、%ワイルドカードまたは特定の値（例えば、VARCHAR(%)または VARCHAR(10)）を使用して必要な属性を設定してください。
- 無効なマッピングを定義すると、エラーメッセージがログに書き込まれます。
- ソースデータ型にデフォルト値がある場合は、ルールで指定する必要があります。例えば、TIMESTAMP の代わりに TIMESTAMP(6)を使用する必要があります。

## Amazon Redshift ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Amazon Redshift ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Amazon Redshift ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	利用可能なただ 1 つのオプションは、 <b>【ターゲットテーブルを作成する】</b> であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマを選択します。

プロパティ	説明
バケット	Amazon Redshift に読み込むデータオブジェクトへのアクセスを保存、整理、制御する Amazon S3 のバケットコンテナの名前を指定します。
データディレクトリまたはタスクターゲットディレクトリ	一括取り込みアプリケーションがタスクに関連付けられたジョブの出力ファイルを格納するサブディレクトリを指定します。このフィールドは、初期ロードジョブの場合は <b>【データディレクトリ】</b> 、増分ロードジョブ、または初期ロードと増分ロードの組み合わせジョブの場合は <b>【タスクターゲットディレクトリ】</b> と呼ばれます。

次の表は、**【詳細】** に表示されるターゲットの詳細プロパティについて説明しています。

プロパティ	説明
大文字と小文字の変換を有効にする	デフォルトでは、ターゲットテーブル名およびカラム名は、対応するソース名と同じ大文字と小文字で生成されます。ただし、ターゲットのクラスタレベルまたはセッションレベルのプロパティがこの大文字と小文字を区別する動作をオーバーライドしている場合を除きます。ターゲット名の大文字と小文字を制御する場合は、このチェックボックスを選択します。次に、 <b>【大文字と小文字の変換ストラテジ】</b> オプションを選択します。
大文字と小文字の変換ストラテジ	<p><b>【大文字と小文字の変換を有効にする】</b> を選択した場合は、以下のいずれかのオプションを選択して、生成されたターゲットテーブル（またはオブジェクト）名およびカラム（またはフィールド）名の大文字と小文字の処理方法を指定します。</p> <ul style="list-style-type: none"> <li>- <b>ソースと同じ</b>。ソーステーブル（またはオブジェクト）名およびカラム（またはフィールド）名と同じ大文字と小文字を使用します。</li> <li>- <b>UPPERCASE</b>。すべて大文字を使用します。</li> <li>- <b>lowercase</b>。すべて小文字を使用します。</li> </ul> <p>デフォルト値は、<b>【ソースと同じ】</b> です。</p> <p><b>注:</b> 選択したストラテジは、大文字と小文字の制御に関するターゲットのクラスタレベルまたはセッションレベルのプロパティをオーバーライドします。</p>

## Amazon S3 ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Amazon S3 ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Amazon S3 ターゲットのプロパティについて説明しています。

プロパティ	説明
出力形式	<p>出力ファイルの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- CSV</li> <li>- AVRO</li> <li>- PARQUET</li> </ul> <p>デフォルトの値は <b>【CSV】</b> です。</p> <p><b>注:</b> CSV 形式の出力ファイルでは、各フィールドの区切り文字として二重引用符 ("" ) が使用されます。</p>
CSV ファイルへのヘッダーの追加	<b>【CSV】</b> が出力形式として選択されている場合は、このチェックボックスをオンにして、ソース列名を含むヘッダーを出力 CSV ファイルに追加します。

プロパティ	説明
Parquet 圧縮タイプ	<p>【PARQUET】出力形式が選択されている場合、Parquet でサポートされている圧縮タイプを選択できます。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>【なし】</b>、これは圧縮が使用されないことを意味します。</p>
Avro 形式	<p>出力形式として <b>【AVRO】</b> を選択した場合、ソーステーブルごとに作成される Avro スキーマの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- Avro-Flat。すべての Avro フィールドを 1 つのレコードに一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Generic。ソーステーブルのすべてのカラムを Avro フィールドの単一の配列に一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Nested。各タイプの情報を個別のレコードに編成する場合に、この Avro スキーマ形式を使用します。</li> </ul> <p>デフォルト値は <b>【Avro-Flat】</b> です。</p>
Avro シリアル化形式	<p>出力形式として <b>【AVRO】</b> が選択されている場合は、Avro 出力ファイルのシリアル化形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Binary</li> <li>- JSON</li> </ul> <p>デフォルト値は <b>【Binary】</b> です。</p>
Avro スキーマディレクトリ	<p>出力形式として <b>【AVRO】</b> が選択されている場合は、一括取り込みアプリケーションが各ソーステーブルの Avro スキーマ定義を格納しているローカルディレクトリを指定します。スキーマ定義ファイルには、次の命名パターンがあります。</p> <p><i>schemaname_tablename.txt</i></p> <p><b>注:</b> このディレクトリが指定されていない場合、Avro スキーマ定義ファイルは作成されません。</p>
ファイル圧縮タイプ	<p>CSV または AVRO 出力形式の出力ファイルのファイル圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Deflate</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>【なし】</b>、これは圧縮が使用されないことを意味します。</p>
Avro 圧縮タイプ	<p><b>【AVRO】</b> が出力形式としてが選択されている場合は、Avro 圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Bzip2</li> <li>- Deflate</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>【なし】</b>、これは圧縮が使用されないことを意味します。</p>
Deflate 圧縮レベル	<p><b>【Deflate】</b> が <b>【Avro 圧縮タイプ】</b> フィールドで選択されている場合、圧縮レベルとして 0〜9 を指定します。デフォルトは 0 です。</p>
ディレクトリタグの追加	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、Hive パーティショニングの命名規則と互換性を持たせるために適用サイクルディレクトリの名前に「dt=」プレフィックスを追加するには、このチェックボックスをオンにします。このチェックボックスはデフォルトでオフになっています。</p>

プロパティ	説明
タスクターゲットディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、出力データファイル、スキーマファイル、および CDC サイクルのコンテンツと完了ファイルを保持する他のディレクトリのルートディレクトリ。このフィールドを使用して、タスクのカスタムルートディレクトリを指定できます。<b>【親としての接続ディレクトリ】</b> オプションを有効にしている場合は、必要に応じて、接続プロパティで指定された親ディレクトリで使用するタスクターゲットディレクトリを指定できます。</p> <p>このフィールドは、次のディレクトリフィールドのいずれかのパターンで {TaskTargetDirectory} プレースホルダが指定されている場合は必須です。</p>
データディレクトリ	<p>初期ロードタスクの場合は、一括取り込みアプリケーションが出力データファイルとオプションでスキーマを保存するディレクトリのディレクトリ構造を定義します。ディレクトリパターンを定義するには、次のタイプのエントリを使用できます。</p> <ul style="list-style-type: none"> <li>- プレースホルダ {SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および {DD} ( {YY}、{YYYY}、{MM}、および {DD} は日付要素です)。{Timestamp} 値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、初期ロードジョブがターゲットへのデータの転送を開始した日付と時刻を表します。</li> <li>- 特定のディレクトリ名。</li> <li>- <code>toUpper()</code> および <code>toLower()</code> 関数。これは、関連付けられた (<i>placeholder</i>) の値を強制的に大文字または小文字に変換します。</li> </ul> <p><b>注:</b> プレースホルダの値の大文字と小文字は区別されません。</p> <p>例:</p> <pre>myDir1/{SchemaName}/{TableName} myDir1/myDir2/{SchemaName}/{YYYY}/{MM}/{TableName}_{Timestamp} myDir1/{toLower(SchemaName)}/{TableName}_{Timestamp}</pre> <p>デフォルトのディレクトリパターンは {TableName}_{Timestamp} です。</p> <p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、<code>cdc-data</code> データファイルを含むサブディレクトリへのカスタムパスを定義します。ディレクトリパターンを定義する場合は、次のタイプのエントリを使用します。</p> <ul style="list-style-type: none"> <li>- プレースホルダ {TaskTargetDirectory}、{SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および {DD} ( {YY}、{YYYY}、{MM}、および {DD} は日付要素です)。{Timestamp} 値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、CDC サイクルが開始された日付と時刻を表します。</li> <li>- <code>toUpper</code> または <code>toLower</code> 関数を含める場合は、前の例に示すように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</li> <li>- 特定のディレクトリ名。</li> </ul> <p>デフォルトのディレクトリパターンは {TaskTargetDirectory}/data/{TableName}/data です。</p> <p><b>注:</b> Amazon S3、フラットファイルおよび Microsoft Azure Data Lake Storage Gen2 ターゲットでは、<b>【親としての接続ディレクトリ】</b> が選択されている場合、一括取り込みアプリケーションは、ターゲット接続プロパティで指定されたディレクトリをデータディレクトリパスのルートとして使用します。Google Cloud Storage ターゲットの場合、一括取り込みアプリケーションは取り込みタスクのターゲットプロパティで指定したバケット名を使用します。</p>
親としての接続ディレクトリ	<p>ターゲット接続プロパティで指定されたディレクトリ値を、タスクターゲットプロパティで指定されたカスタムディレクトリパスの親ディレクトリとして使用するようになるには、このチェックボックスをオンにします。初期ロードタスクの場合、親ディレクトリは、<b>データディレクトリ</b> と <b>スキーマディレクトリ</b> で使用されます。増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、親ディレクトリは <b>データディレクトリ</b>、<b>スキーマディレクトリ</b>、<b>サイクル完了ディレクトリ</b>、および <b>サイクルコンテンツディレクトリ</b> で使用されます。</p> <p>このチェックボックスはデフォルトで選択されています。オフにしたとき、初期ロードの場合は、<b>【データディレクトリ】</b> フィールドで出力ファイルへのフルパスを定義します。増分ロードの場合は、必要に応じて <b>【タスクターゲットディレクトリ】</b> でタスクのルートディレクトリを指定します。</p>

プロパティ	説明
スキーマディレクトリ	<p>デフォルトディレクトリ以外のディレクトリにスキーマファイルを保存する場合は、スキーマファイルを保存するカスタムディレクトリを指定できます。初期ロードの場合、便利になるように、以前に使用した値を使用できる場合はそれがドロップダウンリストに表示されます。このフィールドはオプションです。</p> <p>初期ロードの場合、デフォルトでは、スキーマはデータディレクトリに保存されます。増分ロード、および初期ロードと増分ロードの組み合わせの場合、スキーマファイルのデフォルトディレクトリは<code>{TaskTargetDirectory}/data/{TableName}/schema</code> です。</p> <p><b>[データディレクトリ]</b> フィールドと同じプレースホルダを使用できます。プレースホルダは必ず中かっこ<code>{ }</code>で囲んでください。</p> <p><code>toUpper</code> または <code>toLowerCase</code> 関数を含める場合は、<code>{toLowerCase(SchemaName)}</code> のように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</p> <p><b>注:</b> スキーマは、CSV 形式で出力データファイルにのみ書き込まれます。Parquet および Avro 形式のデータファイルには、独自の埋め込みスキーマが含まれています。</p>
サイクル完了ディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、サイクル完了ファイルを含むディレクトリへのパス。デフォルトは<code>{TaskTargetDirectory}/cycle/completed</code> です。</p>
サイクルコンテンツディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、サイクルコンテンツファイルを含むディレクトリへのパス。デフォルトは<code>{TaskTargetDirectory}/cycle/contents</code> です。</p>
データディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、各データディレクトリの下に、CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p> <p>このオプションが選択されていない場合、別のディレクトリ構造を定義しない限り、個々のデータファイルがタイムスタンプなしで同じディレクトリに書き込まれます。</p>
サマリディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、サマリコンテンツサブディレクトリおよび完了サブディレクトリの下に CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p>
コンテンツ内の個々のファイルを一覧表示する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、コンテンツサブディレクトリの下に個々のデータファイルが一覧表示されます。</p> <p><b>[サマリディレクトリにサイクルのパーティション化を使用する]</b> がオフの場合は、このオプションがデフォルトでオンになります。タイムスタンプや日付などのプレースホルダを使用してカスタムサブディレクトリを設定できる場合を除き、コンテンツサブディレクトリ内の個々のファイルがすべて一覧表示されます。</p> <p><b>[データディレクトリにサイクルのパーティション化を使用する]</b> が選択されている場合でも、必要に応じてこのチェックボックスを選択して、個々のファイルを一覧表示し、CDC サイクルごとにグループ化することができます。</p>

次の表は、**【詳細】** セクションに表示される Amazon S3 ターゲットの詳細プロパティについて説明しています。

フィールド	説明
操作タイプの追加	ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイプを記録するメタデータカラムを追加するには、このチェックボックスを選択します。 増分ロードの場合、ジョブは、挿入を表す「I」、更新を表す「U」、または削除を表す「D」を書き込みます。初期ロードの場合、ジョブは常に、挿入を表す「I」を書き込みます。 デフォルトでは、このチェックボックスは、増分ロードジョブ、初期および増分ロードジョブの場合はオンになっており、初期ロードジョブの場合はオフになっています。
操作時間の追加	ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。 初期ロードの場合、ジョブは常に現在の日付と時刻を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。
操作所有者の追加	ジョブがターゲットにプロパゲートする出力にソース SQL 操作の所有者を記録するメタデータカラムを追加するには、このチェックボックスを選択します。 初期ロードの場合、ジョブは常に所有者として「INFA」を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。
操作トランザクション ID の追加	ジョブが SQL 操作のターゲットにプロパゲートする出力にソーストランザクション ID を含むメタデータカラムを追加するには、このチェックボックスを選択します。 初期ロードの場合、ジョブは常に ID として「1」を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。
前のイメージを追加	ジョブがターゲットに書き込む出力に UNDO データを含めるには、このチェックボックスを選択します。 初期ロードの場合、ジョブは null を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。

## Databricks Delta ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Databricks Delta ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Databricks Delta ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	利用可能なただ 1 つのオプションは、 <b>【ターゲットテーブルを作成する】</b> であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマ。



プロパティ	説明
適用モード	<p>増分ロードジョブ、および初期ロードジョブと増分ロードジョブの組み合わせの場合に、挿入、更新、削除といったソース DML の変更がターゲットにどのように適用されるかを示します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- <b>標準</b>。1 回の適用サイクルの間の変更を累積し、それをターゲットに適用する前に、より少ない SQL 文になるようにそれらをインテリジェントにマージします。例えば、ソース行で更新とそれに続く削除が発生した場合、ターゲットに行は適用されません。同じカラムまたはフィールドで複数の更新が発生した場合、最後の更新のみがターゲットに適用されます。異なるカラムまたはフィールドで複数の更新が発生した場合、更新はターゲットに適用される前に 1 つの更新レコードにマージされます。</li> <li>- <b>論理削除</b>。ソース削除操作を論理削除としてターゲットに適用します。論理削除では、削除された行をデータベースから実際には削除せずに、削除済みとしてマークします。例えば、ソースで削除を行うと、ターゲットの変更レコードの INFA_OPERATION_TYPE カラムに「D」が表示されます。ソースで更新とそれに続く削除が発生した場合、2 つのレコードがターゲットに書き込まれ、両方の INFA_OPERATION_TYPE カラムに「D」が表示されます。</li> </ul> <p>処理を完了するために論理的に削除されたデータが必要となる、長期にわたるビジネスプロセスがある場合、誤って削除したデータを復元する必要がある場合、または削除された値を監査目的で追跡する必要がある場合は、論理削除の使用を検討してください。デフォルトは [標準] です。</p>
データディレクトリまたはタスクターゲットディレクトリ	<p>一括取り込みアプリケーションがタスクに関連付けられたジョブの出力ファイルを格納するサブディレクトリを指定します。このフィールドは、初期ロードジョブの場合は <b>【データディレクトリ】</b>、増分ロードジョブ、または初期ロードと増分ロードの組み合わせジョブの場合は <b>【タスクターゲットディレクトリ】</b> と呼ばれます。</p>

【詳細】で、【適用モード】を【論理削除】に設定して、各削除操作のメタデータカラムを追加するようにしている場合は、次のターゲット詳細プロパティを入力できます。

フィールド	説明
操作タイプの追加	<p>ジョブがターゲットテーブルにプロパゲートする出力にソース SQL 操作タイプを含むメタデータカラムを追加します。デフォルトでは、このカラムは INFA_OPERATION_TYPE という名前です。</p> <p>このチェックボックスは、【適用モード】オプションが【論理削除】に設定されている場合にのみ表示されます。</p> <p>論理削除モードでは、ジョブは削除操作の場合は「D」を書き込み、INFA_OPERATION_TYPE カラムへの挿入と更新の場合は NULL を書き込みます。操作タイプが NULL の場合、他の【操作の追加...】メタデータカラムも NULL です。操作タイプが「D」の場合にのみ、他のメタデータカラムに NULL 以外の値が含まれます。</p> <p>デフォルトでは、このチェックボックスは選択されています。選択を解除することはできません。</p>
操作時間の追加	<p>ジョブがターゲットに伝播する出力にソース SQL 操作時間を含むメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
操作所有者の追加	<p>ジョブがターゲットに伝播する出力にソース SQL 操作の所有者を含むメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>デフォルトでは、このチェックボックスは選択されています。</p> <p>注: このプロパティは、PostgreSQL ソースを持つジョブでは使用できません。</p>



フィールド	説明
操作トランザクション ID の追加	ジョブが SQL 操作のターゲットに伝播する出力にソーストランザクション ID を含むメタデータカラムを追加するには、このチェックボックスを選択します。 デフォルトでは、このチェックボックスは選択されています。
メタデータカラムのプレフィックス	追加されたメタデータカラムの名前にプレフィックスを追加し、それらを簡単に識別して、既存のカラムの名前との競合を防ぐことができるようにします。 デフォルト値は INFA_ です。
アンマネージドテーブルの作成	タスクで Databricks Delta ターゲットテーブルをアンマネージドテーブルとして作成する場合は、このチェックボックスを選択します。タスクをデプロイした後は、このフィールドを編集してマネージドテーブルに切り替えることはできません。 デフォルトでは、このオプションはオフになっており、マネージドテーブルが作成されます。 Databricks Delta のマネージドテーブルとアンマネージドテーブルの詳細については、Databricks Delta のドキュメントを参照してください。
非管理対象テーブルの親ディレクトリ	Databricks Delta アンマネージドテーブルを作成する場合は、キャプチャされた DML レコードの処理時にターゲットテーブルごとに生成される Parquet ファイルを保持するために、Amazon S3 または Microsoft Azure Data Lake Storage に存在する親ディレクトリを指定する必要があります。

## Google BigQuery ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Google BigQuery ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Google BigQuery ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	利用可能なただ 1 つのオプションは、 <b>【ターゲットテーブルを作成する】</b> であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマ。

プロパティ	説明
適用モード	<p>増分ロードジョブ、および初期ロードと増分ロードの組み合わせジョブの場合に、挿入、更新、削除といったソース DML の変更がターゲットにどのように適用されるかを示します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- <b>標準。</b> 1 回の適用サイクルの間の変更を累積し、それをターゲットに適用する前に、より少ない SQL 文になるようにそれらをインテリジェントにマージします。例えば、ソース行で更新とそれに続く削除が発生した場合、ターゲットに行は適用されません。同じカラムまたはフィールドで複数の更新が発生した場合、最後の更新のみがターゲットに適用されます。異なるカラムまたはフィールドで複数の更新が発生した場合、更新はターゲットに適用される前に 1 つの更新レコードにマージされます。</li> <li>- <b>監査。</b> ソーステーブルで実行されたすべての DML 操作の監査証拠をターゲットに適用します。ソーステーブルの各 DML 変更の行が、<b>【詳細】</b> セクションで選択した監査カラムとともに、生成されたターゲットテーブルに書き込まれます。監査カラムには、DML 操作タイプ、時刻、所有者、トランザクション ID、生成された昇順シーケンス番号、前のイメージなどの変更に関するメタデータが含まれています。監査履歴を使用して、データをターゲットデータベースに書き込む前にダウストリームの計算または処理を実行する場合、またはキャプチャされた変更に関するメタデータを調べる場合は、監査適用モードの使用を検討してください。</li> </ul> <p>デフォルト値は <b>【標準】</b> です。</p> <p><b>注:</b> <b>【監査】</b> 適用モードは、SAP Mass Ingestion コネクタを使用した SAP ソースに適用されません。</p>
バケット	Google Cloud Storage に読み込むデータオブジェクトへのアクセスを保存、整理、制御する既存のバケットコンテナの名前を指定します。
データディレクトリまたはタスクターゲットディレクトリ	一括取り込みアプリケーションがタスクに関連付けられたジョブの出力ファイルを格納するサブディレクトリを指定します。このフィールドは、初期ロードジョブの場合は <b>【データディレクトリ】</b> 、増分ロードジョブ、または初期ロードと増分ロードの組み合わせジョブの場合は <b>【タスクターゲットディレクトリ】</b> と呼ばれます。

次の表は、**【詳細】** に表示される Google BigQuery ターゲットの詳細プロパティについて説明しています。

フィールド	説明
最終レプリケート時刻を追加	<p>ターゲットテーブルでレコードが挿入または最後に更新された時点のタイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。初期ロードでは、ロードされたすべてのレコードのタイムスタンプが同じになります。増分ロード、および初期ロードと増分ロードの組み合わせの場合、このカラムには、ターゲットに適用された最後の DML 操作のタイムスタンプが記録されます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
メタデータカラムのプレフィックス	<p>追加されたメタデータカラムの名前にプレフィックスを追加し、それらを簡単に識別して、既存のカラムの名前との競合を防ぐことができます。</p> <p>プレフィックスには特殊文字を含めないようにしてください。特殊文字を含めた場合、タスクのデプロイメントが失敗します。</p> <p>デフォルト値は INFA_ です。</p>

フィールド	説明
大文字と小文字の変換を有効にする	デフォルトでは、ターゲットテーブル名およびカラム名は、対応するソース名と同じ大文字と小文字で生成されます。ただし、ターゲットのクラスタレベルまたはセッションレベルのプロパティがこの大文字と小文字を区別する動作をオーバーライドしている場合を除きます。ターゲット名の大文字と小文字を制御する場合は、このチェックボックスを選択します。次に、 <b>【大文字と小文字の変換ストラテジ】</b> オプションを選択します。
大文字と小文字の変換ストラテジ	<p><b>【大文字と小文字の変換を有効にする】</b> を選択した場合は、以下のいずれかのオプションを選択して、生成されたターゲットテーブル（またはオブジェクト）名およびカラム（またはフィールド）名の大文字と小文字の処理方法を指定します。</p> <ul style="list-style-type: none"> <li>- <b>ソースと同じ</b>。ソーステーブル（またはオブジェクト）名およびカラム（またはフィールド）名と同じ大文字と小文字を使用します。</li> <li>- UPPERCASE。すべて大文字を使用します。</li> <li>- lowercase。すべて小文字を使用します。</li> </ul> <p>デフォルト値は、<b>【ソースと同じ】</b> です。</p> <p><b>注:</b> 選択したストラテジは、大文字と小文字の制御に関するターゲットのクラスタレベルまたはセッションレベルのプロパティをオーバーライドします。</p>

## Google Cloud Storage ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Google Cloud Storage ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Google Cloud Storage ターゲットのプロパティについて説明しています。

プロパティ	説明
出力形式	<p>出力ファイルの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- CSV</li> <li>- AVRO</li> <li>- PARQUET</li> </ul> <p>デフォルトの値は <b>【CSV】</b> です。</p> <p><b>注:</b> CSV 形式の出力ファイルでは、各フィールドの区切り文字として二重引用符 (") が使用されます。</p>
CSV ファイルへのヘッダーの追加	<p><b>【CSV】</b> が出力形式として選択されている場合は、このチェックボックスをオンにして、ソース列名を含むヘッダーを出力 CSV ファイルに追加します。</p>
Parquet 圧縮タイプ	<p><b>【PARQUET】</b> 出力形式が選択されている場合、Parquet でサポートされている圧縮タイプを選択できます。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>【なし】</b>、これは圧縮が使用されないことを意味します。</p>

プロパティ	説明
Avro 形式	<p>出力形式として【AVRO】を選択した場合、ソーステーブルごとに作成される Avro スキーマの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- Avro-Flat。すべての Avro フィールドを 1 つのレコードに一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Generic。ソーステーブルのすべてのカラムを Avro フィールドの単一の配列に一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Nested。各タイプの情報を個別のレコードに編成する場合に、この Avro スキーマ形式を使用します。</li> </ul> <p>デフォルト値は【Avro-Flat】です。</p>
Avro シリアル化形式	<p>出力形式として【AVRO】が選択されている場合は、Avro 出力ファイルのシリアル化形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Binary</li> <li>- JSON</li> </ul> <p>デフォルト値は【Binary】です。</p>
Avro スキーマディレクトリ	<p>出力形式として【AVRO】が選択されている場合は、一括取り込みアプリケーションが各ソーステーブルの Avro スキーマ定義を格納しているローカルディレクトリを指定します。スキーマ定義ファイルには、次の命名パターンがあります。</p> <p><i>schemaname_tablename.txt</i></p> <p><b>注:</b> このディレクトリが指定されていない場合、Avro スキーマ定義ファイルは作成されません。</p>
ファイル圧縮タイプ	<p>CSV または AVRO 出力形式の出力ファイルのファイル圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Deflate</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は【なし】、これは圧縮が使用されないことを意味します。</p>
Avro 圧縮タイプ	<p>【AVRO】が出力形式としてが選択されている場合は、Avro 圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Bzip2</li> <li>- Deflate</li> <li>- Snappy</li> </ul> <p>デフォルト値は【なし】、これは圧縮が使用されないことを意味します。</p>
Deflate 圧縮レベル	<p>【Deflate】が【Avro 圧縮タイプ】フィールドで選択されている場合、圧縮レベルとして 0～9 を指定します。デフォルトは 0 です。</p>
ディレクトリタグの追加	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、Hive パーティショニングの命名規則と互換性を持たせるために適用サイクルディレクトリの名前に「dt=」プレフィックスを追加するには、このチェックボックスをオンにします。このチェックボックスはデフォルトでオフになっています。</p>
バケット	<p>Google Cloud Storage に読み込むデータオブジェクトへのアクセスを保存、整理、制御する既存のバケットコンテナの名前を指定します。</p>

プロパティ	説明
タスクターゲットディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、出力データファイル、スキーマファイル、および CDC サイクルのコンテンツと完了ファイルを保持する他のディレクトリのルートディレクトリ。このフィールドを使用して、タスクのカスタムルートディレクトリを指定できます。<b>【親としての接続ディレクトリ】</b> オプションを有効にしている場合は、必要に応じて、接続プロパティで指定された親ディレクトリで使用するタスクターゲットディレクトリを指定できます。</p> <p>このフィールドは、次のディレクトリフィールドのいずれかのパターンで {TaskTargetDirectory} プレースホルダが指定されている場合は必須です。</p>
データディレクトリ	<p>初期ロードタスクの場合は、一括取り込みアプリケーションが出力データファイルとオプションでスキーマを保存するディレクトリのディレクトリ構造を定義します。ディレクトリパターンを定義するには、次のタイプのエントリを使用できます。</p> <ul style="list-style-type: none"> <li>- プレースホルダ {SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および {DD} ( {YY}、{YYYY}、{MM}、および {DD} は日付要素です)。{Timestamp} 値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、初期ロードジョブがターゲットへのデータの転送を開始した日付と時刻を表します。</li> <li>- 特定のディレクトリ名。</li> <li>- <code>toUpper()</code> および <code>toLower()</code> 関数。これは、関連付けられた (<i>placeholder</i>) の値を強制的に大文字または小文字に変換します。</li> </ul> <p><b>注:</b> プレースホルダの値の大文字と小文字は区別されません。</p> <p>例:</p> <pre>myDir1/{SchemaName}/{TableName} myDir1/myDir2/{SchemaName}/{YYYY}/{MM}/{TableName}_{Timestamp} myDir1/{toLower(SchemaName)}/{TableName}_{Timestamp}</pre> <p>デフォルトのディレクトリパターンは {TableName}_{Timestamp} です。</p> <p><b>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、cdc-data データファイルを含むサブディレクトリへのカスタムパスを定義します。ディレクトリパターンを定義する場合は、次のタイプのエントリを使用します。</b></p> <ul style="list-style-type: none"> <li>- プレースホルダ {TaskTargetDirectory}、{SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および {DD} ( {YY}、{YYYY}、{MM}、および {DD} は日付要素です)。{Timestamp} 値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、CDC サイクルが開始された日付と時刻を表します。</li> </ul> <p><code>toUpper</code> または <code>toLower</code> 関数を含める場合は、前の例に示すように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</p> <ul style="list-style-type: none"> <li>- 特定のディレクトリ名。</li> </ul> <p>デフォルトのディレクトリパターンは {TaskTargetDirectory}/data/{TableName}/data です。</p> <p><b>注:</b> Amazon S3、フラットファイルおよび Microsoft Azure Data Lake Storage Gen2 ターゲットでは、<b>【親としての接続ディレクトリ】</b> が選択されている場合、一括取り込みアプリケーションは、ターゲット接続プロパティで指定されたディレクトリをデータディレクトリパスのルートとして使用します。Google Cloud Storage ターゲットの場合、一括取り込みアプリケーションは取り込みタスクのターゲットプロパティで指定したバケット名を使用します。</p>

プロパティ	説明
スキーマディレクトリ	<p>デフォルトディレクトリ以外のディレクトリにスキーマファイルを保存する場合は、スキーマファイルを保存するカスタムディレクトリを指定できます。初期ロードの場合、便利になるように、以前に使用した値を使用できる場合はそれがドロップダウンリストに表示されます。このフィールドはオプションです。</p> <p>初期ロードの場合、デフォルトでは、スキーマはデータディレクトリに保存されます。増分ロード、および初期ロードと増分ロードの組み合わせの場合、スキーマファイルのデフォルトディレクトリは{TaskTargetDirectory}/data/{TableName}/schema です。</p> <p><b>[データディレクトリ]</b> フィールドと同じプレースホルダを使用できます。プレースホルダは必ず中かっこ{ }で囲んでください。</p> <p>toUpper または toLower 関数を含める場合は、{toLower(SchemaName)}のように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</p> <p><b>注:</b> スキーマは、CSV 形式で出力データファイルにのみ書き込まれます。Parquet および Avro 形式のデータファイルには、独自の埋め込みスキーマが含まれています。</p>
サイクル完了ディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、サイクル完了ファイルを含むディレクトリへのパス。デフォルトは{TaskTargetDirectory}/cycle/completed です。</p>
サイクルコンテンツディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、サイクルコンテンツファイルを含むディレクトリへのパス。デフォルトは{TaskTargetDirectory}/cycle/contents です。</p>
データディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、各データディレクトリの下に、CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p> <p>このオプションが選択されていない場合、別のディレクトリ構造を定義しない限り、個々のデータファイルがタイムスタンプなしで同じディレクトリに書き込まれます。</p>
サマリディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、サマリコンテンツサブディレクトリおよび完了サブディレクトリの下に CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p>
コンテンツ内の個々のファイルを一覧表示する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、コンテンツサブディレクトリの下に個々のデータファイルが一覧表示されます。</p> <p><b>[サマリディレクトリにサイクルのパーティション化を使用する]</b> がオフの場合は、このオプションがデフォルトでオンになります。タイムスタンプや日付などのプレースホルダを使用してカスタムサブディレクトリを設定できる場合を除き、コンテンツサブディレクトリ内の個々のファイルがすべて一覧表示されます。</p> <p><b>[データディレクトリにサイクルのパーティション化を使用する]</b> が選択されている場合でも、必要に応じてこのチェックボックスを選択して、個々のファイルを一覧表示し、CDC サイクルごとにグループ化することができます。</p>

次の表は、**【詳細】** セクションに表示される Google Cloud Storage ターゲットの詳細プロパティについて説明しています。

フィールド	説明
操作タイプの追加	<p>ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイプを記録するメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>増分ロードの場合、ジョブは、挿入を表す「I」、更新を表す「U」、または削除を表す「D」を書き込みます。初期ロードの場合、ジョブは常に、挿入を表す「I」を書き込みます。</p> <p>デフォルトでは、このチェックボックスは、増分ロードジョブ、初期および増分ロードジョブの場合はオンになっており、初期ロードジョブの場合はオフになっています。</p>
操作時間の追加	<p>ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。</p> <p>初期ロードの場合、ジョブは常に現在の日付と時刻を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
操作所有者の追加	<p>ジョブがターゲットにプロパゲートする出力にソース SQL 操作の所有者を記録するメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>初期ロードの場合、ジョブは常に所有者として「INFA」を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
操作トランザクション ID の追加	<p>ジョブが SQL 操作のターゲットにプロパゲートする出力にソーストランザクション ID を含むメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>初期ロードの場合、ジョブは常に ID として「1」を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
前のイメージを追加	<p>ジョブがターゲットに書き込む出力に UNDO データを含めるには、このチェックボックスを選択します。</p> <p>初期ロードの場合、ジョブは null を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>

## Kafka ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Kafka ターゲットのプロパティをタスクウィザードの**【ターゲット】** ページで指定する必要があります。

これらのプロパティは、増分ロード操作にのみ適用されます。



次の表は、**【ターゲット】** セクションに表示される Kafka ターゲットのプロパティについて説明しています。

プロパティ	説明
テーブル名をトピック名として使用	<p>一括取り込みアプリケーションがソースデータを含むメッセージを、ソースオブジェクトごとに1つずつ個別のトピックに書き込むか、すべてのメッセージを1つのトピックに書き込むかを示します。</p> <p>テーブル固有のトピックを区切るメッセージを書き込むには、このチェックボックスを選択します。トピック名は、<b>【スキーマ名を含める】</b>、<b>【テーブルプレフィックス】</b>、または<b>【テーブルサフィックス】</b> プロパティに追加しない限り、トピック名はソーステーブル名に一致します。デフォルトでは、このチェックボックスはオフです。</p>
スキーマ名を含める	<p><b>【テーブル名をトピック名として使用】</b> が選択されている場合、このチェックボックスが表示され、デフォルトで選択されています。この設定により、テーブル固有のトピック名にソーススキーマ名が追加されます。トピック名の形式は次のとおりです。スキーマ名_テーブル名。スキーマ名を含めない場合は、このチェックボックスをオフにします。</p>
テーブルプレフィックス	<p><b>テーブル名をトピック名として使用</b>を選択すると、このプロパティが表示され、オプションでプレフィックスを入力してテーブル固有のトピック名に追加できます。例えば、myprefix_を指定すると、トピック名の形式は「myprefix_テーブル名」になります。プレフィックスの後のアンダースコア ( ) を省略すると、プレフィックスがテーブル名の前に追加されます。</p>
テーブルサフィックス	<p><b>テーブル名をトピック名として使用</b>を選択すると、このプロパティが表示され、オプションでサフィックスを入力してテーブル固有のトピック名に追加できます。例えば、_mysuffix を指定すると、トピック名の形式は「テーブル名_mysuffix」になります。サフィックスの前のアンダースコア ( ) を省略すると、サフィックスがテーブル名に追加されます。</p>
出力形式	<p>出力ファイルの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- CSV</li> <li>- AVRO</li> <li>- JSON</li> </ul> <p>デフォルトの値は <b>【CSV】</b> です。</p> <p><b>注:</b> CSV 形式の出力ファイルでは、各フィールドの区切り文字として二重引用符 (") が使用されます。</p> <p>Kafka ターゲットが Confluent Schema Registry を使用して増分ロードジョブのスキーマを格納する場合は、形式として <b>【AVRO】</b> を選択する必要があります。</p>
JSON 形式	<p>出力形式として <b>【JSON】</b> が選択されている場合は、出力の詳細レベルを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- <b>簡潔。</b> この形式では、操作タイプやカラムの名前と値など、最も関連性の高いデータのみが出力に記録されます。</li> <li>- <b>詳細。</b> この形式では、テーブル名やカラムタイプなどの詳細情報が記録されます。</li> </ul>
Avro 形式	<p>出力形式として <b>【AVRO】</b> を選択した場合、ソーステーブルごとに作成される Avro スキーマの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- Avro-Flat。すべての Avro フィールドを1つのレコードに一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Generic。ソーステーブルのすべてのカラムを Avro フィールドの単一の配列に一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Nested。各タイプの情報を個別のレコードに編成する場合に、この Avro スキーマ形式を使用します。</li> </ul> <p>デフォルト値は <b>【Avro-Flat】</b> です。</p>

プロパティ	説明
Avro シリアル化形式	<p>出力形式として AVRO が選択されている場合は、Avro 出力ファイルのシリアル化形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- Binary</li> <li>- JSON</li> <li>- なし</li> </ul> <p>デフォルト値は [Binary] です。</p> <p>Confluent Schema Registry を使用してスキーマを格納する Confluent Kafka ターゲットがある場合は、[なし] を選択します。それ以外の場合、Confluent Schema Registry はスキーマを登録しません。Confluent Schema Registry を使用していない場合、[なし] は選択しないでください。</p>
Avro スキーマディレクトリ	<p>出力形式として [AVRO] が選択されている場合は、一括取り込みデータベースが各ソーステーブルの Avro スキーマ定義を格納しているローカルディレクトリを指定します。スキーマ定義ファイルには、次の命名パターンがあります。</p> <p><code>schemaname_tablename.txt</code></p> <p>注: このディレクトリが指定されていない場合、Avro スキーマ定義ファイルは作成されません。</p> <p>ソーススキーマの変更によってターゲットが変更されることが予想される場合、Avro スキーマ定義ファイルは、タイムスタンプを含む一意の名前で次の形式で再生成されます。</p> <p><code>schemaname_tablename_YYYYMMDDhhmmss.txt</code></p> <p>この一意の命名パターンにより、古いスキーマ定義ファイルが監査目的で保持されます。</p>
Avro 圧縮タイプ	<p>[AVRO] が出力形式としてが選択されている場合は、Avro 圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Bzip2</li> <li>- Deflate</li> <li>- Snappy</li> </ul> <p>デフォルト値は [なし]、これは圧縮が使用されないことを意味します。</p>

次の表は、[詳細] に表示される Kafka ターゲットの詳細プロパティについて説明しています。

プロパティ	説明
操作タイプの追加	<p>ジョブがターゲットに伝播する出力にソース SQL 操作タイプを含むメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>増分ロードの場合、ジョブは挿入を表す「I」、更新を表す「U」、または削除を表す「D」を書き込みます。初期ロードの場合、ジョブは常に挿入を表す「I」を書き込みます。</p> <p>デフォルトでは、このチェックボックスはオンです。</p>
操作時間の追加	<p>ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。</p> <p>初期ロードの場合、ジョブは常に現在の日付と時刻を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
前のイメージを追加	<p>ジョブがターゲットに書き込む出力に UNDO データを含めるには、このチェックボックスを選択します。</p> <p>初期ロードの場合、ジョブは null を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>

プロパティ	説明
非同期書き込み	<p>Kafka へのメッセージの同期配信を使用するかどうかを制御します。</p> <ul style="list-style-type: none"> <li>- 同期配信を使用するには、このチェックボックスをオフにします。Kafka は、一括取り込みアプリケーションが次のメッセージを送信する前に、各メッセージの受信を確認する必要があります。このモードでは、Kafka が重複メッセージを受信する可能性はほとんどありません。ただし、パフォーマンスが低下する可能性があります。</li> <li>- 非同期配信を使用するには、このチェックボックスを選択します。一括取り込みアプリケーションから変更が取得された順序に関係なく、できるだけ早くメッセージを送信します。</li> </ul> <p>デフォルトでは、このチェックボックスは選択されています。</p>
プロデューサ設定プロパティ	<p>Apache Kafkatargets の Kafka プロデューサプロパティを入力するための、<i>key=value</i> ペアのコンマ区切りリストを指定します。</p> <p>Kafka プロデューサーのプロパティは、このフィールドまたは Kafka 接続の追加の接続プロパティフィールドのいずれかに設定します。</p> <p>このフィールドにプロデューサプロパティを入力すると、プロパティは、このタスクにのみ関連付けられたアプリケーション取り込みジョブに関連します。接続のプロデューサプロパティを入力する場合、<b>【プロデューサ設定プロパティ】</b> フィールドのプロパティを指定して特定のタスクの接続レベルのプロパティをオーバーライドしない限り、プロパティは接続定義を使用するすべてのタスクのジョブに関係します。</p> <p>Kafka プロデューサプロパティについては、Apache Kafka のドキュメントを参照してください。</p>

## Microsoft Azure Data Lake Storage Gen2 ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Microsoft Azure Data Lake Storage Gen2 ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Microsoft Azure Data Lake Storage Gen2 ターゲットのプロパティについて説明しています。

プロパティ	説明
出力形式	<p>出力ファイルの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- CSV</li> <li>- AVRO</li> <li>- PARQUET</li> </ul> <p>デフォルトの値は <b>【CSV】</b> です。</p> <p>注: CSV 形式の出力ファイルでは、各フィールドの区切り文字として二重引用符 (") が使用されます。</p>
CSV ファイルへのヘッダーの追加	<p><b>【CSV】</b> が出力形式として選択されている場合は、このチェックボックスをオンにして、ソース列名を含むヘッダーを出力 CSV ファイルに追加します。</p>
Parquet 圧縮タイプ	<p><b>【PARQUET】</b> 出力形式が選択されている場合、Parquet でサポートされている圧縮タイプを選択できます。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>【なし】</b>、これは圧縮が使用されないことを意味します。</p>

プロパティ	説明
Avro 形式	<p>出力形式として【AVRO】を選択した場合、ソーステーブルごとに作成される Avro スキーマの形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- Avro-Flat。すべての Avro フィールドを 1 つのレコードに一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Generic。ソーステーブルのすべてのカラムを Avro フィールドの単一の配列に一覧表示する場合に、この Avro スキーマ形式を使用します。</li> <li>- Avro-Nested。各タイプの情報を個別のレコードに編成する場合に、この Avro スキーマ形式を使用します。</li> </ul> <p>デフォルト値は【Avro-Flat】です。</p>
Avro シリアル化形式	<p>出力形式として【AVRO】が選択されている場合は、Avro 出力ファイルのシリアル化形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Binary</li> <li>- JSON</li> </ul> <p>デフォルト値は【Binary】です。</p>
Avro スキーマディレクトリ	<p>出力形式として【AVRO】が選択されている場合は、一括取り込みアプリケーションが各ソーステーブルの Avro スキーマ定義を格納しているローカルディレクトリを指定します。スキーマ定義ファイルには、次の命名パターンがあります。</p> <p><i>schemaname_tablename.txt</i></p> <p>注: このディレクトリが指定されていない場合、Avro スキーマ定義ファイルは作成されません。</p>
ファイル圧縮タイプ	<p>CSV または AVRO 出力形式の出力ファイルのファイル圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Deflate</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は【なし】、これは圧縮が使用されないことを意味します。</p>
Avro 圧縮タイプ	<p>【AVRO】が出力形式としてが選択されている場合は、Avro 圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Bzip2</li> <li>- Deflate</li> <li>- Snappy</li> </ul> <p>デフォルト値は【なし】、これは圧縮が使用されないことを意味します。</p>
Deflate 圧縮レベル	<p>【Deflate】が【Avro 圧縮タイプ】フィールドで選択されている場合、圧縮レベルとして 0～9 を指定します。デフォルトは 0 です。</p>
ディレクトリタグの追加	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、Hive パーティショニングの命名規則と互換性を持たせるために適用サイクルディレクトリの名前に「dt=」プレフィックスを追加するには、このチェックボックスをオンにします。このチェックボックスはデフォルトでオフになっています。</p>

プロパティ	説明
タスクターゲットディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、出力データファイル、スキーマファイル、および CDC サイクルのコンテンツと完了ファイルを保持する他のディレクトリのルートディレクトリ。このフィールドを使用して、タスクのカスタムルートディレクトリを指定できます。<b>【親としての接続ディレクトリ】</b> オプションを有効にしている場合は、必要に応じて、接続プロパティで指定された親ディレクトリで使用するタスクターゲットディレクトリを指定できます。</p> <p>このフィールドは、次のディレクトリフィールドのいずれかのパターンで {TaskTargetDirectory} プレースホルダが指定されている場合は必須です。</p>
データディレクトリ	<p>初期ロードタスクの場合は、一括取り込みアプリケーションが出力データファイルとオプションでスキーマを保存するディレクトリのディレクトリ構造を定義します。ディレクトリパターンを定義するには、次のタイプのエントリを使用できます。</p> <ul style="list-style-type: none"> <li>- プレースホルダ {SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および {DD} ( {YY}、{YYYY}、{MM}、および {DD} は日付要素です)。{Timestamp} 値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、初期ロードジョブがターゲットへのデータの転送を開始した日付と時刻を表します。</li> <li>- 特定のディレクトリ名。</li> <li>- <code>toUpper()</code> および <code>toLower()</code> 関数。これは、関連付けられた (<i>placeholder</i>) の値を強制的に大文字または小文字に変換します。</li> </ul> <p><b>注:</b> プレースホルダの値の大文字と小文字は区別されません。</p> <p>例:</p> <pre>myDir1/{SchemaName}/{TableName} myDir1/myDir2/{SchemaName}/{YYYY}/{MM}/{TableName}_{Timestamp} myDir1/{toLower(SchemaName)}/{TableName}_{Timestamp}</pre> <p>デフォルトのディレクトリパターンは {TableName}_{Timestamp} です。</p> <p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、<code>cdc-data</code> データファイルを含むサブディレクトリへのカスタムパスを定義します。ディレクトリパターンを定義する場合は、次のタイプのエントリを使用します。</p> <ul style="list-style-type: none"> <li>- プレースホルダ {TaskTargetDirectory}、{SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および {DD} ( {YY}、{YYYY}、{MM}、および {DD} は日付要素です)。{Timestamp} 値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、CDC サイクルが開始された日付と時刻を表します。</li> <li>- <code>toUpper</code> または <code>toLower</code> 関数を含める場合は、前の例に示すように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を丸かっこで囲みます。</li> <li>- 特定のディレクトリ名。</li> </ul> <p>デフォルトのディレクトリパターンは {TaskTargetDirectory}/data/{TableName}/data です。</p> <p><b>注:</b> Amazon S3、フラットファイルおよび Microsoft Azure Data Lake Storage Gen2 ターゲットでは、<b>【親としての接続ディレクトリ】</b> が選択されている場合、一括取り込みアプリケーションは、ターゲット接続プロパティで指定されたディレクトリをデータディレクトリパスのルートとして使用します。Google Cloud Storage ターゲットの場合、一括取り込みアプリケーションは取り込みタスクのターゲットプロパティで指定したバケット名を使用します。</p>
親としての接続ディレクトリ	<p>ターゲット接続プロパティで指定されたディレクトリ値を、タスクターゲットプロパティで指定されたカスタムディレクトリパスの親ディレクトリとして使用するようになるには、このチェックボックスをオンにします。初期ロードタスクの場合、親ディレクトリは、<b>データディレクトリ</b> と <b>スキーマディレクトリ</b> で使用されます。増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、親ディレクトリは <b>データディレクトリ</b>、<b>スキーマディレクトリ</b>、<b>サイクル完了ディレクトリ</b>、および <b>サイクルコンテンツディレクトリ</b> で使用されます。</p> <p>このチェックボックスはデフォルトで選択されています。オフにしたとき、初期ロードの場合は、<b>【データディレクトリ】</b> フィールドで出力ファイルへのフルパスを定義します。増分ロードの場合は、必要に応じて <b>【タスクターゲットディレクトリ】</b> でタスクのルートディレクトリを指定します。</p>

プロパティ	説明
スキーマディレクトリ	<p>デフォルトディレクトリ以外のディレクトリにスキーマファイルを保存する場合は、スキーマファイルを保存するカスタムディレクトリを指定できます。初期ロードの場合、便利になるように、以前に使用した値を使用できる場合はそれがドロップダウンリストに表示されます。このフィールドはオプションです。</p> <p>初期ロードの場合、デフォルトでは、スキーマはデータディレクトリに保存されます。増分ロード、および初期ロードと増分ロードの組み合わせの場合、スキーマファイルのデフォルトディレクトリは<code>{TaskTargetDirectory}/data/{TableName}/schema</code> です。</p> <p><b>[データディレクトリ]</b> フィールドと同じプレースホルダを使用できます。プレースホルダは必ず中かっこ<code>{ }</code>で囲んでください。</p> <p><code>toUpper</code> または <code>toLowerCase</code> 関数を含める場合は、<code>{toLowerCase(SchemaName)}</code> のように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</p> <p><b>注:</b> スキーマは、CSV 形式で出力データファイルにのみ書き込まれます。Parquet および Avro 形式のデータファイルには、独自の埋め込みスキーマが含まれています。</p>
サイクル完了ディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、サイクル完了ファイルを含むディレクトリへのパス。デフォルトは<code>{TaskTargetDirectory}/cycle/completed</code> です。</p>
サイクルコンテンツディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、サイクルコンテンツファイルを含むディレクトリへのパス。デフォルトは<code>{TaskTargetDirectory}/cycle/contents</code> です。</p>
データディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、各データディレクトリの下に、CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p> <p>このオプションが選択されていない場合、別のディレクトリ構造を定義しない限り、個々のデータファイルがタイムスタンプなしで同じディレクトリに書き込まれます。</p>
サマリディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、サマリコンテンツサブディレクトリおよび完了サブディレクトリの下に CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p>
コンテンツ内の個々のファイルを一覧表示する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、コンテンツサブディレクトリの下に個々のデータファイルが一覧表示されます。</p> <p><b>[サマリディレクトリにサイクルのパーティション化を使用する]</b> がオフの場合は、このオプションがデフォルトでオンになります。タイムスタンプや日付などのプレースホルダを使用してカスタムサブディレクトリを設定できる場合を除き、コンテンツサブディレクトリ内の個々のファイルがすべて一覧表示されます。</p> <p><b>[データディレクトリにサイクルのパーティション化を使用する]</b> が選択されている場合でも、必要に応じてこのチェックボックスを選択して、個々のファイルを一覧表示し、CDC サイクルごとにグループ化することができます。</p>



次の表は、**【詳細】** セクションに表示される Microsoft Azure Data Lake Storage Gen2 ターゲットの詳細プロパティについて説明しています。

フィールド	説明
操作タイプの追加	ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイプを記録するメタデータカラムを追加するには、このチェックボックスを選択します。 増分ロードの場合、ジョブは、挿入を表す「I」、更新を表す「U」、または削除を表す「D」を書き込みます。初期ロードの場合、ジョブは常に、挿入を表す「I」を書き込みます。 デフォルトでは、このチェックボックスは、増分ロードジョブ、初期および増分ロードジョブの場合はオンになっており、初期ロードジョブの場合はオフになっています。
操作時間の追加	ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。 初期ロードの場合、ジョブは常に現在の日付と時刻を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。
操作所有者の追加	ジョブがターゲットにプロパゲートする出力にソース SQL 操作の所有者を記録するメタデータカラムを追加するには、このチェックボックスを選択します。 初期ロードの場合、ジョブは常に所有者として「INFA」を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。
操作トランザクション ID の追加	ジョブが SQL 操作のターゲットにプロパゲートする出力にソーストランザクション ID を含むメタデータカラムを追加するには、このチェックボックスを選択します。 初期ロードの場合、ジョブは常に ID として「1」を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。
前のイメージの追加	ジョブがターゲットに書き込む出力に UNDO データを含めるには、このチェックボックスを選択します。 初期ロードの場合、ジョブは null を書き込みます。 デフォルトでは、このチェックボックスは選択されていません。

## Microsoft Azure Synapse Analytics ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Microsoft Azure Synapse Analytics ターゲットのプロパティをタスクウィザードの **【ターゲット】** ページで指定する必要があります。

次の表は、**【ターゲット】** セクションに表示される Microsoft Azure Synapse Analytics ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	利用可能なただ 1 つのオプションは、 <b>【ターゲットテーブルを作成する】</b> であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマを選択します。接続プロパティで指定されたスキーマ名がデフォルトで表示されます。 このフィールドは大文字と小文字が区別されます。そのため、接続プロパティのスキーマ名が適切な大文字小文字表記で入力されていることを確認してください。



## Microsoft Fabric OneLake ターゲットプロパティ

Microsoft Fabric OneLake ターゲットのあるアプリケーション取り込みタスクを定義する場合、タスクウィザードの **【ターゲット】** ページでいくつかのターゲットプロパティを入力する必要があります。

次の表は、**【ターゲット】** セクションに表示される Microsoft Fabric OneLake ターゲットのプロパティについて説明しています。

プロパティ	説明
出力形式	出力ファイルの形式を選択します。次のオプションがあります。 <ul style="list-style-type: none"><li>- CSV</li><li>- AVRO</li><li>- PARQUET</li></ul> デフォルトの値は <b>【CSV】</b> です。 <b>注:</b> CSV 形式の出力ファイルでは、各フィールドの区切り文字として二重引用符 (") が使用されます。
CSV ファイルへのヘッダーの追加	<b>【CSV】</b> が出力形式として選択されている場合は、このチェックボックスをオンにして、ソース列名を含むヘッダーを出力 CSV ファイルに追加します。
Parquet 圧縮タイプ	<b>【PARQUET】</b> 出力形式が選択されている場合、Parquet でサポートされている圧縮タイプを選択できます。次のオプションがあります。 <ul style="list-style-type: none"><li>- なし</li><li>- Gzip</li><li>- Snappy</li></ul> デフォルト値は <b>【なし】</b> 、これは圧縮が使用されないことを意味します。
Avro 形式	出力形式として <b>【AVRO】</b> を選択した場合、ソーステーブルごとに作成される Avro スキーマの形式を選択します。次のオプションがあります。 <ul style="list-style-type: none"><li>- Avro-Flat。すべての Avro フィールドを 1 つのレコードに一覧表示する場合に、この Avro スキーマ形式を使用します。</li><li>- Avro-Generic。ソーステーブルのすべてのカラムを Avro フィールドの単一の配列に一覧表示する場合に、この Avro スキーマ形式を使用します。</li><li>- Avro-Nested。各タイプの情報を個別のレコードに編成する場合に、この Avro スキーマ形式を使用します。</li></ul> デフォルト値は <b>【Avro-Flat】</b> です。
Avro シリアル化形式	出力形式として <b>【AVRO】</b> が選択されている場合は、Avro 出力ファイルのシリアル化形式を選択します。次のオプションがあります。 <ul style="list-style-type: none"><li>- なし</li><li>- Binary</li><li>- JSON</li></ul> デフォルト値は <b>【Binary】</b> です。
Avro スキーマディレクトリ	出力形式として <b>【AVRO】</b> が選択されている場合は、一括取り込みアプリケーションが各ソーステーブルの Avro スキーマ定義を格納しているローカルディレクトリを指定します。スキーマ定義ファイルには、次の命名パターンがあります。  <code>schemaname_tablename.txt</code>  <b>注:</b> このディレクトリが指定されていない場合、Avro スキーマ定義ファイルは作成されません。

プロパティ	説明
ファイル圧縮タイプ	<p>CSV または AVRO 出力形式の出力ファイルのファイル圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Deflate</li> <li>- Gzip</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>[なし]</b>、これは圧縮が使用されないことを意味します。</p>
Avro 圧縮タイプ	<p><b>[AVRO]</b> が出力形式としてが選択されている場合は、Avro 圧縮タイプを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Bzip2</li> <li>- Deflate</li> <li>- Snappy</li> </ul> <p>デフォルト値は <b>[なし]</b>、これは圧縮が使用されないことを意味します。</p>
Deflate 圧縮レベル	<p><b>[Deflate]</b> が <b>[Avro 圧縮タイプ]</b> フィールドで選択されている場合、圧縮レベルとして 0 ~9 を指定します。デフォルトは 0 です。</p>
データディレクトリ	<p>一括取り込みアプリケーションが出力データファイルとオプションでスキーマを保存するディレクトリのディレクトリ構造を定義します。ディレクトリパターンを定義するには、次のタイプのエントリを使用できます。</p> <ul style="list-style-type: none"> <li>- プレースホルダ{SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および{DD}（{YY}、{YYYY}、{MM}、および{DD}は日付要素です）。{Timestamp}値の形式は <code>yyyymmdd_hhmissms</code> です。ディレクトリパスに生成された日付と時刻は、初期ロードジョブがターゲットへのデータの転送を開始した日付と時刻を表します。</li> <li>- 特定のディレクトリ名。</li> <li>- <code>toUpper()</code>および <code>toLower()</code>関数。これは、関連付けられた(<i>placeholder</i>)の値を強制的に大文字または小文字に変換します。</li> </ul> <p><b>注:</b> プレースホルダの値の大文字と小文字は区別されません。</p> <p>例:</p> <pre>myDir1/{SchemaName}/{TableName} myDir1/myDir2/{SchemaName}/{YYYY}/{MM}/{TableName}_{Timestamp} myDir1/{toLower(SchemaName)}/{TableName}_{Timestamp}</pre> <p>デフォルトのディレクトリパターンは{TableName}_{Timestamp}です。</p>
スキーマディレクトリ	<p>デフォルトディレクトリ以外のディレクトリにスキーマファイルを保存する場合は、スキーマファイルを保存するカスタムディレクトリを指定できます。初期ロードの場合、便利になるように、以前に使用した値を使用できる場合はそれがドロップダウンリストに表示されます。このフィールドはオプションです。</p> <p>初期ロードの場合、デフォルトでは、スキーマはデータディレクトリに保存されます。増分ロード、および初期ロードと増分ロードの組み合わせの場合、スキーマファイルのデフォルトディレクトリは{TaskTargetDirectory}/data/{TableName}/schema です。</p> <p><b>[データディレクトリ]</b> フィールドと同じプレースホルダを使用できます。プレースホルダは必ず中かっこ{ }で囲んでください。</p> <p><code>toUpper</code> または <code>toLower</code> 関数を含める場合は、{toLower(SchemaName)}のように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</p> <p><b>注:</b> スキーマは、CSV 形式で出力データファイルにのみ書き込まれます。Parquet および Avro 形式のデータファイルには、独自の埋め込みスキーマが含まれています。</p>

## Microsoft SQL Server ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Microsoft SQL Server ターゲットのプロパティをタスクウィザードの【ターゲット】ページで指定する必要があります。

これらのプロパティは、初期ロード操作にのみ適用されます。

次の表は、【ターゲット】セクションに表示される Microsoft SQL Server ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	【ターゲットテーブルの作成】オプションは、ソーステーブルに基づいてターゲットテーブルを生成します。 <b>注:</b> ターゲットテーブルが作成された後、一括取り込みアプリケーションは、後続のジョブ実行でターゲットテーブルをインテリジェントに処理します。一括取り込みアプリケーションは、特定の状況に応じて、ターゲットテーブルを切り詰めたり再作成したりする場合があります。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマを選択します。接続プロパティで指定されたスキーマ名がデフォルトで表示されます。 このフィールドは大文字と小文字が区別されます。そのため、接続プロパティのスキーマ名が適切な大文字小文字表記で入力されていることを確認してください。

## Oracle Cloud Object Storage ターゲットのプロパティ

Oracle Cloud Object Storage ターゲットを持つデータベース統合タスクを定義する場合は、タスクウィザードの【ターゲット】タブで一部のターゲットのプロパティを入力する必要があります。

【ターゲット】では、次の Oracle Cloud Object Storage ターゲットのプロパティを入力できます。

プロパティ	説明
出力形式	出力ファイルの次の形式のいずれかを選択します。 <ul style="list-style-type: none"><li>- CSV</li><li>- AVRO</li><li>- PARQUET</li></ul> デフォルトは【CSV】です。 <b>注:</b> CSV 形式の出力ファイルでは、各フィールドの区切り文字として二重引用符 (") が使用されます。
CSV ファイルへのヘッダーの追加	出力形式として【CSV】を選択した場合は、このチェックボックスをオンにして、ソースカラム名を含むヘッダーを出力 CSV ファイルに追加します。
Avro 形式	出力形式として【AVRO】を選択した場合は、ソーステーブルごとに作成される Avro スキーマの次の形式のいずれかを選択します。 <ul style="list-style-type: none"><li>- Avro-Flat。すべての Avro フィールドを 1 つのレコードに一覧表示する場合に、この Avro スキーマ形式を使用します。</li><li>- Avro-Generic。ソーステーブルのすべてのカラムを Avro フィールドの単一の配列に一覧表示する場合に、この Avro スキーマ形式を使用します。</li><li>- Avro-Nested。各タイプの情報を個別のレコードに編成する場合に、この Avro スキーマ形式を使用します。</li></ul> デフォルトは【Avro-Flat】です。

プロパティ	説明
Avro シリアル化形式	<p>出力形式として【AVRO】を選択した場合は、Avro 出力ファイルの次のシリアル化形式のいずれかを選択します。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Binary</li> <li>- JSON</li> </ul> <p>デフォルトは【Binary】です。</p>
Avro スキーマディレクトリ	<p>出力形式として【AVRO】を選択した場合は、一括取り込みアプリケーションがソーステーブルごとに Avro スキーマ定義を保存するローカルディレクトリを指定します。スキーマ定義ファイルには、次の命名パターンがあります。</p> <p>schemaname_tablename.txt</p> <p>注: このディレクトリが指定されていない場合、Avro スキーマ定義ファイルは作成されません。</p>
ファイル圧縮タイプ	<p>CSV または AVRO 出力形式の出力ファイルに対して、次のファイル圧縮タイプのいずれかを選択します。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Deflate</li> <li>- gzip</li> <li>- Snappy</li> </ul> <p>デフォルトは【なし】で、これは圧縮が使用されないことを意味します。</p>
Avro 圧縮タイプ	<p>出力形式として【AVRO】を選択した場合は、次の Avro 圧縮タイプのいずれかを選択します。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- Bzip2</li> <li>- Deflate</li> <li>- Snappy</li> </ul> <p>デフォルトは【なし】で、これは圧縮が使用されないことを意味します。</p>
Parquet 圧縮タイプ	<p>出力形式として【PARQUET】を選択した場合は、Parquet でサポートされる次の圧縮タイプのいずれかを選択します。</p> <ul style="list-style-type: none"> <li>- なし</li> <li>- gzip</li> <li>- Snappy</li> </ul> <p>デフォルトは【なし】で、これは圧縮が使用されないことを意味します。</p>
Deflate 圧縮レベル	<p>【Avro 圧縮タイプ】として【Deflate】を選択した場合は、圧縮レベルを 0-9 から指定します。デフォルトは 0 です。</p>
タスクターゲットディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、このフィールドで出力データファイル、スキーマファイル、および CDC サイクルのコンテンツと完了ファイルを保持する他のディレクトリのルートディレクトリを定義します。このフィールドを使用して、タスクのカスタムルートディレクトリを指定できます。[親としての接続ディレクトリ] オプションを有効にしている場合は、必要に応じて、接続プロパティで指定された親ディレクトリで使用するタスクターゲットディレクトリを指定できます。</p> <p>このフィールドは、ディレクトリフィールドのいずれかのパターンで {TaskTargetDirectory} プレースホルダが指定されている場合は必須です。</p>
ディレクトリタグの追加	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、Hive パーティショニングの命名規則と互換性を持たせるために適用サイクルディレクトリの名前に「dt=」プレフィックスを追加するには、このチェックボックスをオンにします。このチェックボックスはデフォルトでオフになっています。</p>

プロパティ	説明
親としての接続ディレクトリ	<p>ターゲット接続プロパティで指定されたディレクトリ値を、タスクターゲットプロパティで指定されたカスタムディレクトリパスの親ディレクトリとして使用するようには、このチェックボックスをオンにします。初期ロードタスクの場合、親ディレクトリは、<b>[データディレクトリ]</b> と <b>[スキーマディレクトリ]</b> フィールドで使用されます。増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合、親ディレクトリは、<b>[データディレクトリ]</b>、<b>[スキーマディレクトリ]</b>、<b>[サイクル完了ディレクトリ]</b>、および <b>[サイクルコンテンツディレクトリ]</b> フィールドで使用されます。</p> <p>このチェックボックスはデフォルトで選択されています。オフにしたとき、初期ロードの場合は、<b>[データディレクトリ]</b> フィールドで出力ファイルへのフルパスを定義します。増分ロードの場合は、必要に応じて <b>[タスクターゲットディレクトリ]</b> でタスクのルートディレクトリを指定します。</p>
データディレクトリ	<p>初期ロードタスクの場合は、一括取り込みアプリケーションが出力データファイルとオプションでスキーマを保存するディレクトリのディレクトリ構造を定義します。ディレクトリパターンを定義する場合は、次のタイプのエントリを使用します。</p> <ul style="list-style-type: none"> <li>- プレースホルダ{SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および{DD} ({YY}、{YYYY}、{MM}、および{DD}は日付要素です)。{Timestamp}値の形式は yyyyymmdd_hhmissms です。ディレクトリパスに生成された日付と時刻は、初期ロードジョブがターゲットへのデータの転送を開始した日付と時刻を表します。</li> <li>- 特定のディレクトリ名。</li> <li>- toUpper()および toLower()関数。これは、関連付けられた(placeholder)の値を強制的に大文字または小文字に変換します。</li> </ul> <p><b>注:</b> プレースホルダの値の大文字と小文字は区別されません。</p> <p>例:</p> <pre>myDir1/{SchemaName}/{TableName}myDir1/ myDir2/{SchemaName}/{YYYY}/{MM}/{TableName}_{Timestamp} myDir1/{toLower(SchemaName)}/{TableName}_{Timestamp}</pre> <p>デフォルトは{TableName}_{Timestamp}です。</p> <p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、cdc-data データファイルを含むサブディレクトリへのカスタムパスを定義します。ディレクトリパターンを定義する場合は、次のタイプのエントリを使用します。</p> <ul style="list-style-type: none"> <li>- プレースホルダ{TaskTargetDirectory}、{SchemaName}、{TableName}、{Timestamp}、{YY}、{YYYY}、{MM}、および{DD} ({YY}、{YYYY}、{MM}、および{DD}は日付要素です)。{Timestamp}値の形式は yyyyymmdd_hhmissms です。ディレクトリパスに生成された日付と時刻は、CDC サイクルが開始された日付と時刻を表します。</li> <li>- toUpper または toLower 関数を含める場合は、前の例に示したように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</li> <li>- 特定のディレクトリ名。</li> </ul> <p>デフォルトのディレクトリパターンは{TaskTargetDirectory}/data/{TableName}/data です。</p> <p><b>注:</b> Amazon S3、フラットファイル、Microsoft Azure Data Lake Storage Gen2、および Oracle Cloud Object Store ターゲットの場合、<b>[親としての接続ディレクトリ]</b> を選択すると、一括取り込みアプリケーションは、ターゲット接続プロパティで指定されたディレクトリをデータディレクトリパスのルートとして使用します。Google Cloud Storage ターゲットの場合、一括取り込みアプリケーションは取り込みタスクのターゲットプロパティで指定したバケット名を使用します。</p>

プロパティ	説明
スキーマディレクトリ	<p>デフォルトディレクトリ以外のディレクトリにスキーマファイルを保存する場合は、スキーマファイルを保存するカスタムディレクトリを指定します。初期ロードの場合は、以前使用した値がリストに表示されます（存在する場合）。このフィールドはオプションです。</p> <p>初期ロードの場合、デフォルトでは、スキーマはデータディレクトリに保存されます。増分ロード、および初期ロードと増分ロードの組み合わせの場合、スキーマファイルのデフォルトディレクトリは<code>{TaskTargetDirectory}/data/{TableName}/schema</code> です。</p> <p>[データディレクトリ] フィールドと同じプレースホルダを使用できます。プレースホルダは必ず中かっこ<code>{ }</code>で囲んでください。</p> <p><code>toUpper</code> または <code>toLowerCase</code> 関数を含める場合は、<code>{toLowerCase(SchemaName)}</code> のように、プレースホルダ名を丸かっこで囲み、関数とプレースホルダの両方を中かっこで囲みます。</p> <p><b>注:</b> スキーマは、CSV 形式で出力データファイルにのみ書き込まれます。Parquet および Avro 形式のデータファイルには、独自の埋め込みスキーマが含まれています。</p>
サイクル完了ディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、このフィールドでサイクル完了ファイルを含むディレクトリへのパスを定義します。デフォルトは<code>{TaskTargetDirectory}/cycle/completed</code> です。</p>
サイクルコンテンツディレクトリ	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、このフィールドでサイクルコンテンツファイルを含むディレクトリへのパスを定義します。デフォルトは<code>{TaskTargetDirectory}/cycle/contents</code> です。</p>
データディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、このオプションを選択すると、各データディレクトリの下に CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p> <p>このオプションが選択されていない場合、別のディレクトリ構造を定義しない限り、個々のデータファイルがタイムスタンプなしで同じディレクトリに書き込まれます。</p>
サマリディレクトリにサイクルのパーティション化を使用する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、サマリコンテンツサブディレクトリおよび完了サブディレクトリの下に CDC サイクルごとにタイムスタンプサブディレクトリが作成されます。</p>
コンテンツ内の個々のファイルを一覧表示する	<p>増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクの場合は、コンテンツサブディレクトリの下に個々のデータファイルが一覧表示されます。</p> <p><b>[サマリディレクトリにサイクルのパーティション化を使用する]</b> がオフの場合は、このオプションがデフォルトでオンになります。タイムスタンプや日付などのプレースホルダを使用してカスタムサブディレクトリを設定する場合を除き、コンテンツサブディレクトリ内の個々のファイルがすべて一覧表示されます。</p> <p><b>[データディレクトリにサイクルのパーティション化を使用する]</b> が選択されている場合でも、必要に応じてこのチェックボックスを選択して、個々のファイルを一覧表示し、CDC サイクルごとにグループ化することができます。</p>



【詳細】で次のような詳細ターゲットプロパティを入力して、監査テーブルに記録された各削除操作または各 DML 変更に変タデータカラムを追加することができます。

フィールド	説明
操作タイプの追加	<p>ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイプを記録するメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>増分ロードの場合、ジョブは、挿入を表す「I」、更新を表す「U」、または削除を表す「D」を書き込みます。初期ロードの場合、ジョブは常に、挿入を表す「I」を書き込みます。</p> <p>デフォルトでは、このチェックボックスは、増分ロードジョブ、初期および増分ロードジョブの場合はオンになっており、初期ロードジョブの場合はオフになっています。</p>
操作時間の追加	<p>ジョブがターゲットにプロパゲートする出力にソース SQL 操作タイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。</p> <p>初期ロードの場合、ジョブは常に現在の日付と時刻を書き込みます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
前のイメージを追加	<p>ジョブがターゲットに書き込む出力に UNDO データを含めるには、このチェックボックスを選択します。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>

## Oracle ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Oracle ターゲットのプロパティをタスクウィザードの【ターゲット】ページで指定する必要があります。

次の表は、【ターゲット】セクションに表示される Oracle ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	<p>利用可能なただ 1 つのオプションは、【ターゲットテーブルを作成する】であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。</p>
スキーマ	<p>一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマを選択します。</p>
適用モード	<p>増分ロードジョブ、および初期ロードと増分ロードの組み合わせジョブの場合に、挿入、更新、削除といったソース DML の変更がターゲットにどのように適用されるかを示します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- <b>標準。</b> 1 回の適用サイクルの間の変更を累積し、それをターゲットに適用する前に、より少ない SQL 文になるようにそれらをインテリジェントにマージします。例えば、ソース行で更新とそれに続く削除が発生した場合、ターゲットに行は適用されません。同じカラムまたはフィールドで複数の更新が発生した場合、最後の更新のみがターゲットに適用されます。異なるカラムまたはフィールドで複数の更新が発生した場合、更新はターゲットに適用される前に 1 つの更新レコードにマージされます。</li> <li>- <b>監査。</b> ソーステーブルで実行されたすべての DML 操作の監査証跡をターゲットに適用します。ソーステーブルの各 DML 変更の行が、【詳細】セクションで選択した監査カラムとともに、生成されたターゲットテーブルに書き込まれます。監査カラムには、DML 操作タイプ、時刻、所有者、トランザクション ID、生成された昇順シーケンス番号、前のイメージなどの変更に含まれるメタデータが含まれています。監査履歴を使用して、データをターゲットデータベースに書き込む前にダウンストリームの計算または処理を実行する場合、またはキャプチャされた変更に関するメタデータを調べる場合は、監査適用モードの使用を検討してください。</li> </ul> <p>デフォルト値は【標準】です。</p> <p>注: 【監査】適用モードは、SAP Mass Ingestion コネクタを使用した SAP ソースに適用されます。</p>



## PostgreSQL ターゲットプロパティ

アプリケーション取り込みタスクを定義する場合は、タスクウィザードの【ターゲット】ページで PostgreSQL ターゲットのプロパティを指定する必要があります。

次の表は、【ターゲット】セクションに表示される PostgreSQL ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	利用可能なただ 1 つのオプションは、【ターゲットテーブルを作成する】であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマ。

## Snowflake ターゲットのプロパティ

アプリケーション取り込みタスクを定義する場合は、Snowflake ターゲットのプロパティをタスクウィザードの【ターゲット】ページで指定する必要があります。

次の表は、【ターゲット】セクションに表示される Snowflake ターゲットのプロパティについて説明しています。

プロパティ	説明
ターゲット作成	利用可能なただ 1 つのオプションは、【ターゲットテーブルを作成する】であり、これによりソースオブジェクトをベースにしてターゲットテーブルを生成します。
スキーマ	一括取り込みアプリケーションがターゲットテーブルを作成するターゲットスキーマ。

プロパティ	説明
ステージ	データがターゲットテーブルに書き込まれる前にソースから読み取られたデータを保持する内部ステージング領域の名前。この名前にスペースを含めることはできません。指定されたステージング領域が存在しない場合、自動的に作成されます。
適用モード	<p>Snowflake ターゲットを持つ増分ロードジョブ、および初期ロードジョブと増分ロードジョブの組み合わせの場合に、挿入、更新、削除といったソース DML の変更がターゲットにどのように適用されるかを示します。次のオプションがあります。</p> <ul style="list-style-type: none"> <li>- <b>標準。</b> 1 回の適用サイクルの間の変更を累積し、それをターゲットに適用する前に、より少ない SQL 文になるようにそれらをインテリジェントにマージします。例えば、ソース行で更新とそれに続く削除が発生した場合、ターゲットに行は適用されません。同じカラムまたはフィールドで複数の更新が発生した場合、最後の更新のみがターゲットに適用されます。異なるカラムまたはフィールドで複数の更新が発生した場合、更新はターゲットに適用される前に 1 つの更新レコードにマージされます。</li> <li>- <b>論理削除。</b> ソース削除操作を論理削除としてターゲットに適用します。論理削除では、削除された行をデータベースから実際には削除せずに、削除済みとしてマークします。例えば、ソースで削除を行うと、ターゲットの変更レコードの INFA_OPERATION_TYPE カラムに「D」が表示されます。  <p>論理削除を有効にした後、ノーマルモードまたはバックログモードでソーステーブルを更新すると、一致するレコードが削除され、更新されたレコードが挿入されて、ターゲットテーブルで INFA_OPERATION_TYPE 操作が NULL としてマークされます。同様に、バックログモード中にソーステーブルにレコードを挿入すると、ターゲットテーブルレコードで INFA_OPERATION_TYPE 操作が E としてマークされます。</p> <p>処理を完了するために論理的に削除されたデータが必要となる、長期にわたるビジネスプロセスがある場合、誤って削除したデータを復元する必要がある場合、または削除された値を監査目的で追跡する必要がある場合は、論理削除の使用を検討してください。</p></li> <li>- <b>監査。</b> ソーステーブルで実行されたすべての DML 操作の監査証跡をターゲットに適用します。ソーステーブルの各 DML 変更の行が、<b>【詳細】</b> セクションで選択した監査カラムとともに、生成されたターゲットテーブルに書き込まれます。監査カラムには、DML 操作タイプ、トランザクション ID、前のイメージなどの変更に関するメタデータが含まれています。監査履歴を使用して、データをターゲットデータベースに書き込む前にダウンストリームの計算または処理を実行する場合、またはキャプチャされた変更に関するメタデータを調べる場合は、<b>監査適用モード</b>の使用を検討してください。  <p>監査適用モードを有効にした後、バックログモードまたはノーマルモード中にソーステーブルを更新すると、ターゲットテーブルレコードで INFA_OPERATION_TYPE 操作が E としてマークされます。同様に、バックログモード中にソーステーブルにレコードを挿入すると、ターゲットテーブルレコードで INFA_OPERATION_TYPE 操作が E としてマークされます。</p> <p><b>注：【監査】</b> 適用モードは、SAP Mass Ingestion コネクタを使用した SAP ソースに適用されません。  デフォルトは<b>【標準】</b>です。</p></li> </ul>

【詳細】で次のような詳細ターゲットプロパティを入力できます。

フィールド	説明
最終レプリケート時刻を追加	<p>ターゲットテーブルでレコードが挿入または最後に更新された時点のタイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスをオンにします。初期ロードの場合、ロードされたすべてのレコードのタイムスタンプは同じですが、Superpipe オプションを使用する Snowflake ターゲットに限り、分と秒がわずかに異なる可能性があります。増分ロード、および初期ロードと増分ロードの組み合わせの場合、このカラムには、ターゲットに適用された最後の DML 操作のタイムスタンプが記録されます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
操作タイプの追加	<p>ジョブがターゲットテーブルにプロパゲートする出力にソース SQL 操作タイプを含むメタデータカラムを追加します。デフォルトでは、このカラムは INFA_OPERATION_TYPE という名前です。</p> <p>このフィールドは、【適用モード】 オプションが【監査】または【論理削除】に設定されている場合にのみ表示されます。</p> <p>監査モードでは、ジョブはこのメタデータカラムに、挿入の場合は「I」、更新の場合は「E」、削除の場合は「D」を書き込みます。</p> <p>論理削除モードでは、ジョブは削除の場合は「D」を書き込み、挿入と更新の場合は NULL を書き込みます。操作タイプが NULL の場合、他の【操作の追加...】メタデータカラムも NULL です。操作タイプが「D」の場合にのみ、他のメタデータカラムに NULL 以外の値が含まれます。</p> <p>デフォルトでは、このチェックボックスは選択されています。選択を解除することはできません。</p>
操作時間の追加	<p>ジョブがターゲットテーブルにプロパゲートする出力にソース SQL 操作タイムスタンプを記録するメタデータカラムを追加するには、このチェックボックスを選択します。</p> <p>このフィールドは、【適用モード】が【監査】または【論理削除】に設定されている場合にのみ使用できます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
操作シーケンスの追加	<p>ジョブがターゲットテーブルに挿入する変更操作ごとに、生成された昇順のシーケンス番号を記録するメタデータカラムを追加するには、このチェックボックスを選択します。シーケンス番号には、操作の変更ストリーム位置が反映されます。</p> <p>このフィールドは、【適用モード】が【監査】に設定されている場合にのみ使用できます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
前のイメージを追加	<p>ジョブがターゲットテーブルに挿入する出力に UNDO の「操作前のイメージ」データを含む _OLD カラムを追加するには、このチェックボックスを選択します。これにより、各データカラムの以前の値と現在の値を比較できるようになります。削除操作の場合、現在の値は NULL になります。</p> <p>このフィールドは、【適用モード】が【監査】に設定されている場合にのみ使用できます。</p> <p>デフォルトでは、このチェックボックスは選択されていません。</p>
メタデータカラムのプレフィックス	<p>追加されたメタデータカラムの名前にプレフィックスを追加し、それらを簡単に識別して、既存のカラムの名前との競合を防ぐことができますようにします。</p> <p>デフォルト値は INFA_ です。</p>

フィールド	説明
Superpipe	<p>最初にデータをステージファイルに書き込むのではなく、Snowpipe Streaming API を使用して、短い待ち時間で Snowflake Data Cloud ターゲットテーブルにデータ行を直接すばやくストリーミングするには、このチェックボックスを選択します。このオプションは、すべてのロードタイプで有効にすることができます。</p> <p>ターゲット接続を設定するときに、KeyPair 認証を選択します。</p> <p>デフォルトでは、このチェックボックスは選択されています。中間ステージファイルにデータを書き込む場合は、選択解除します。</p>
マージ頻度	<p>【Superpipe】が選択されている場合、必要に応じて、変更データ行がマージされて Snowflake ターゲットテーブルに適用される頻度を秒単位で設定できます。このフィールドは、増分ロードタスク、および初期ロードと増分ロードの組み合わせタスクに適用されます。有効な値は 300～604800 です。デフォルトは 3600 秒です。</p>
大文字と小文字の変換を有効にする	<p>デフォルトでは、ターゲットテーブル名およびカラム名は、対応するソース名と同じ大文字と小文字で生成されます。ただし、ターゲットのクラスタレベルまたはセッションレベルのプロパティがこの大文字と小文字を区別する動作をオーバーライドしている場合を除きます。ターゲット名の大文字と小文字を制御する場合は、このチェックボックスを選択します。次に、<b>【大文字と小文字の変換ストラテジ】</b> オプションを選択します。</p> <p><b>注:</b> このチェックボックスは、【Superpipe】オプションを選択した場合は使用できません。Snowflake の Superpipe オプションを使用している場合は、大文字と小文字の変換を有効にできません。</p>
大文字と小文字の変換ストラテジ	<p><b>【大文字と小文字の変換を有効にする】</b> を選択した場合は、以下のいずれかのオプションを選択して、生成されたターゲットテーブル（またはオブジェクト）名およびカラム（またはフィールド）名の大文字と小文字の処理方法を指定します。</p> <ul style="list-style-type: none"> <li>- <b>ソースと同じ。</b> ソーステーブル（またはオブジェクト）名およびカラム（またはフィールド）名と同じ大文字と小文字を使用します。</li> <li>- <b>UPPERCASE。</b> すべての大文字を使用します。</li> <li>- <b>lowercase。</b> すべての小文字を使用します。</li> </ul> <p>デフォルト値は、<b>【ソースと同じ】</b> です。</p> <p><b>注:</b> 選択したストラテジは、大文字と小文字の制御に関するターゲットのクラスタレベルまたはセッションレベルのプロパティをオーバーライドします。</p>

## スケジュールとランタイムオプションの設定

アプリケーション取り込みタスクウィザードの**【スケジュールおよびランタイムオプション】** ページでは、初期ロードジョブを実行するスケジュールを指定し、すべてのロードタイプのジョブのランタイムオプションを設定できます。

1. **【スキーマドリフトオプション】** セクションで、各タイプのデータ定義言語（DDL）操作に使用するスキーマドリフトオプションを指定します。

**注:** **【スキーマドリフトオプション】** セクションは、増分ロードタスクおよび初期ロードと増分ロードの組み合わせタスクに対してのみ表示されます。さらに、このセクションは、スキーマ変更の自動検出をサポートするソースに対してのみ表示されます。

次の表に、DDL 操作に指定できるスキーマドリフトオプションを示します。

オプション	説明
無視	ソーススキーマで発生する DDL の変更をターゲットにレプリケートしません。
レプリケート	<p>アプリケーション取り込みジョブが DDL 変更をターゲットにレプリケートできるようにします。</p> <p><b>注:</b></p> <ul style="list-style-type: none"> <li>- プライマリキーフィールドを追加するフィールドの追加操作はサポートされておらず、予期しない結果を引き起こす可能性があります。</li> <li>- フィールドの NULL または NOT NULL 制約を変更するフィールドの変更操作は、ターゲットにレプリケートされません。</li> </ul> <p>サポートされている DDL 操作のタイプは次のとおりです。</p> <ul style="list-style-type: none"> <li>- カラムの追加</li> <li>- カラムの変更</li> <li>- カラムの削除</li> <li>- カラム名の変更</li> </ul> <p>アプリケーション取り込みジョブは、Google BigQuery ターゲットのカラムの変更または名前変更、および Oracle ターゲットのカラムの追加をサポートしていません。</p>
ジョブの停止	アプリケーション取り込みジョブを停止します。
オブジェクトを停止	<p>DDL 変更が発生したソースオブジェクトの処理を停止します。</p> <p><b>注:</b> 1 つ以上のオブジェクトが [オブジェクトを停止] スキーマドリフトオプションによってレプリケーションから除外された場合、ジョブのステータスが <b>【実行中(警告あり)】</b> に変わります。アプリケーション取り込みジョブは、ジョブが変更の処理を停止した後、ソースオブジェクトで発生したデータ変更を取得できません。このアクションにより、ターゲットでデータ損失が発生します。データの損失を回避するには、アプリケーション取り込みジョブを再開する前に、ジョブが処理を停止したソースオブジェクトとターゲットオブジェクトを再同期する必要があります。</p>

- 必要に応じ、**【詳細】** セクションで **【出力ファイルの行数】** の値を変更し、アプリケーション取り込みタスクが、Amazon Redshift、Amazon S3、Google Big Query、Google Cloud Storage、Microsoft Azure Data Lake Storage、Microsoft Azure Synapse Analytics、Microsoft Fabric OneLake、Oracle、または Snowflake ターゲットの出力ファイルに書き込む最大カラム数を指定します。

有効な値は 1~100000000 で、デフォルト値は 100000 行です。

**注:** 増分ロード操作と、初期ロード操作と増分ロード操作の組み合わせの場合、指定した行数に達したとき、またはフラッシュ待ち時間が経過して、トランザクションの処理の途中でジョブが実行されない場合に、変更データがターゲットにフラッシュされます。フラッシュ待ち時間は、ジョブがデータをターゲットにフラッシュする前に、さらに変更データを待機する時間です。待ち時間は 10 秒に設定されており、変更できません。

- 初期ロードジョブの場合のみ、Amazon S3、Google Cloud Storage、Microsoft Azure Data Lake Storage、または Microsoft Fabric OneLake ターゲットの出力データファイルに .dat 拡張子を付ける場合は、必要に応じて **【ファイルタイプに基づくファイル拡張子】** チェックボックスをクリアします。このチェックボックスはデフォルトで選択されており、出力ファイルのファイルタイプに基づいてファイル名拡張子が付けられます。

**注:** これらのターゲットタイプの増分ロードジョブの場合、このオプションは使用できません。一括取り込みアプリケーションは、常にファイルタイプに基づいて出力ファイル名拡張子を使用します。

- 必要に応じて、適用サイクルを設定します。適用サイクルとは、変更データを適用するサイクルであり、ソースの中間データを取得する処理から始まって、ターゲットにデータをコミットして終わります。連続レプリケーションの場合、ソースは待ち時間の短い複数の適用サイクルでデータを処理します。

Amazon S3、Google Cloud Storage、または Microsoft Azure Data Lake Storage Gen2 ターゲットを持つアプリケーション取り込み増分ロードタスクの場合、次の適用サイクルオプションを設定できます。

オプション	説明
サイクル間隔の適用	アプリケーション取り込みジョブが適用サイクルを終了するまでに経過する必要がある時間を指定します。日、時間、分、秒を指定するか、これらの時間フィールドのサブセットに値を指定して、他のフィールドを空白のままにすることができます。 デフォルト値は 15 分です。
サイクル変更制限の適用	アプリケーション取り込みジョブが適用サイクルを終了するまでに処理する必要があるレコード数を指定します。このレコード制限に達すると、取り込みジョブは適用サイクルを終了し、変更データをターゲットに書き込みます。 デフォルト値は 10000 レコードです。 <b>注:</b> 起動中に、古いデータのバックログの処理に追いつく必要がある場合、ジョブは適用サイクル間隔よりも頻繁にこの制限に達する可能性があります。
低アクティビティのフラッシュ間隔	アプリケーション取り込みジョブが適用サイクルを終了する前に、ソースで変更アクティビティがない期間中に経過する必要がある時間を時間、分、またはその両方で指定します。この時間制限に達すると、取り込みジョブは適用サイクルを終了し、変更データをターゲットに書き込みます。 このオプションの値を指定しない場合、データベース取り込みジョブは、 <b>【サイクル変更制限の適用】</b> または <b>【サイクル間隔の適用】</b> のいずれかの制限に達した後にのみ適用サイクルを終了します。 デフォルト値は指定されていません。

**注:**

- **【サイクル間隔の適用】** または **【サイクル変更制限の適用】** フィールドのいずれかが、ゼロ以外の値であるか、デフォルト値を使用する必要があります。
- 適用サイクルは、ジョブが 3 つの制限のいずれかで、最初に満たされた制限に達すると終了します。

5. Apache Kafka ターゲットを持つ増分ロードジョブの場合、次のチェックポイントオプションを設定します。

オプション	説明
チェックポイントのすべての行	データベース取り込みジョブが、Kafka ターゲットに送信されるすべてのメッセージに対してチェックポイント処理を実行するかどうかを示します。 <b>注:</b> このチェックボックスが選択されている場合、 <b>【チェックポイントすべてのコミット】</b> 、 <b>【チェックポイントの行数】</b> 、および <b>【チェックポイントの頻度(秒)】</b> オプションは無視されます。
チェックポイントのすべてのコミット	アプリケーション取り込みジョブが、ソースで発生するすべてのコミットに対してチェックポイント処理を実行するかどうかを示します。



オプション	説明
チェックポイントの行数	チェックポイントを追加する前に、アプリケーション取り込みジョブがターゲットに送信するメッセージの最大数を指定します。このオプションを 0 に設定すると、データベース取り込みジョブはメッセージの数に基づいてチェックポイント処理を実行しません。このオプションを 1 に設定すると、データベース取り込みジョブは各メッセージにチェックポイントを追加します。
チェックポイントの頻度 (秒)	アプリケーション取り込みジョブがチェックポイントを追加するまでに経過する必要がある最大秒数を指定します。このオプションを 0 に設定すると、データベース取り込みジョブは経過時間に基づいてチェックポイント処理を実行しません。

6. タスクに関連付けられたアプリケーション取り込みジョブをスケジュールに基づいて特定の間隔で実行する場合は、**【スケジュール】** セクションで **【このタスクは指定したスケジュールを使用する】** を選択して、ジョブの事前定義されたスケジュールを選択します。

デフォルトでは、**【このタスクはスケジュールを使用しない】** が選択されており、手動でトリガされた場合にのみジョブを実行するように設定されます。

**注:** このフィールドは、初期ロードタスクでのみ使用できます。

Administrator でジョブスケジュールオプションを表示および編集できます。スケジュールを編集すると、変更は、スケジュールに基づいて実行するように設定されているすべてのジョブに自動的に適用されます。すでにデプロイされているタスクのスケジュールを変更すると、更新されたスケジュールは、タスクに関連付けられたアプリケーション取り込みジョブに自動的に適用されます。

前の実行がまだ進行中のときに、ジョブがスケジュールに基づいてトリガされる段階になっても、一括取り込みアプリケーションはジョブを実行せず、すでに進行中のジョブ実行を完了させます。

7. **【カスタムプロパティ】** セクションで、特別なケースに対応するために Informatica が提供するカスタムプロパティを指定できます。プロパティを追加するには、**【プロパティの追加】** アイコンをクリックしてから、プロパティの名前と値を追加します。

カスタムプロパティは、固有の環境と特別な使用例に対応するように設定されています。

**注:** カスタムプロパティを指定する場合は、Informatica グローバルカスタマサポートにお問い合わせください。

## アプリケーション取り込みタスクのデプロイ

アプリケーション取り込みタスクを定義した後、タスクをデプロイして、Secure Agent とデータベース取り込みエージェントサービスを含むオンプレミスシステム上に実行可能なジョブインスタンスを作成します。アプリケーション取り込みジョブは、関連するタスクをデプロイした後にのみ実行できます。タスクをデプロイすると、一括取り込みアプリケーションはタスク定義も検証します。

ジョブをデプロイ解除してから、そのジョブを再度実行する場合は、タスクを再度デプロイして、新しいジョブインスタンスを作成する必要があります。新しいジョブインスタンス名は、形式 `taskname-job_instance_number` の増分番号で終わります。ジョブインスタンス番号は、すべての取り込みジョブの最大インスタンス数に 1 を加算することにより、取り込みタスクをデプロイするたびに増加します。

Snowflake ターゲット用に設定されたタスクをデプロイする前に、ソースオブジェクトの構造と一致しない既存のターゲットテーブルを削除する必要があります。既存のターゲットテーブルがソースオブジェクトの構造と一致なくなる理由としては、ソースフィールドの新規追加、ソースフィールドまたはターゲットフィールドの削除、フィールドの null 制約またはデータ型の変更が挙げられます。既存のターゲットテーブルを削除し



た後にタスクをデプロイすると、ソースオブジェクトの選択ルールとターゲットテーブルの名前変更ルールに基づいて、新しいターゲットテーブルのセットが生成されます。

- ▶ タスクをデプロイするには、アプリケーション取り込みタスクウィザードで、タスクを保存し、**【デプロイ】** をクリックします。

アプリケーション取り込みタスクの名前にスペースを含めた場合、対応するアプリケーション取り込みジョブの名前からスペースが省略されます。

タスクを正常にデプロイすると、アプリケーション取り込みジョブが作成され、ジョブのステータスが**【デプロイ】** になります。一括取り込みの**【マイジョブ】** ページ、またはオペレーションインサイトの**【一括取り込み】** ページの**【すべてのジョブ】** タブのいずれかからジョブを実行できます。

デプロイメントに失敗した場合、対応するアプリケーション取り込みジョブのステータスは**【失敗】** になります。エラーを診断するには、一括取り込みの**【マイジョブ】** ページ、またはオペレーションインサイトの**【一括取り込み】** ページの**【すべてのジョブ】** タブからエラーログをダウンロードします。エラーログをダウンロードするには、ジョブの**【アクション】** メニューで**【エラーログ】** をクリックします。問題を解決したら、アプリケーション取り込みタスクウィザードまたはジョブの**【アクション】** メニューからタスクを再度デプロイします。

**注:**

- タスクのデプロイ中に Secure Agent を再起動すると、ジョブのステータスが「失敗」に切り替わります。タスクのデプロイ中に Secure Agent を再起動しないでください。
- タスクが「デプロイ中」状態でハングしているように見える場合は、Secure Agent を再起動します。関連付けられたジョブインスタンスは、「失敗」のステータスになります。その後、再度デプロイできます。

## アプリケーション取り込みジョブの実行

アプリケーション取り込みジョブのうち、デプロイ済みで**【デプロイ解除】** 以外の状態にあるものは実行できます。

アプリケーション取り込みサービスの**【マイジョブ】** ページまたは一括取り込みの**【一括取り込み】** ページの**【すべてのジョブ】** タブからオペレーションインサイトジョブを実行できます。

初期ロードジョブの場合、対応するアプリケーション取り込みタスクを設定するときに、ジョブを実行するスケジュールを指定できます。

- ▶ 実行するジョブの**【アクション】** メニューで、**【実行】** をクリックします。

サブタスクは、ソースオブジェクトごとに開始されます。

**注意事項:**

- 初期ロードジョブでターゲットテーブルにデータをロードできなかった場合、アプリケーション取り込みジョブは、オブジェクトのサブタスクを最大 3 回再試行します。再試行の最小間隔は 60 秒です。初期ロードの再試行がすべて失敗した場合、一括取り込みアプリケーションは、そのオブジェクトをレプリケーションから除外します。
- 初期ロードジョブがソースオブジェクトとターゲットオブジェクトのフィールド定義間の不整合を検出した場合、ジョブはターゲットテーブルを削除し、ソースデータをターゲットにロードする前に、ソースオブジェクトと一致するように再作成します。
- ソースオブジェクトに多くのレコードが含まれている場合、初期ロードジョブが取り込みを完了するまでに長い時間がかかる可能性があります。

# アプリケーション取り込みジョブの停止

すべてのロードタイプのアプリケーション取り込みジョブのうち、[稼働中]、[実行中(警告あり)]、または[保留] ステータスにあるものは停止できます。

一括取り込みサービスの一括取り込みページまたはオペレーションインサイトの【一括取り込み】ページのオペレーションインサイトタブからジョブを停止できます。

増分ロードジョブを停止すると、一括取り込みアプリケーションは、増分処理を停止した変更ストリーム内の位置の識別子を記録します。識別子は、ターゲット上の INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルに格納されます。ジョブを再開すると、一括取り込みアプリケーションはこの識別子を使用して、ターゲットに最後にロードされた変更レコードを識別し、変更ストリーム内のその時点以降に行われた変更のロードを開始します。

初期ロードジョブの場合、ジョブは、すでに実行されているサブタスクの操作が完了した後にのみ停止します。実行されていないサブタスクは現在の状態のままです。

- ▶ 停止するジョブの【アクション】メニューで、[停止] を選択します。

ジョブのステータスが[停止中] に変わり、次に[停止] に変わります。

**ヒント:** ジョブの停止に時間がかかりすぎる場合は、ジョブを強制終了できます。

# アプリケーション取り込みジョブの再開

ステータスが停止、強制終了、または失敗のアプリケーション取り込みジョブを再開できます。

アプリケーション取り込みサービスの【マイジョブ】ページまたは一括取り込みの【一括取り込み】ページの【すべてのジョブ】タブからオペレーションインサイトジョブを再開できます。

複数のサブタスクを持つ初期ロードジョブを再開すると、一括取り込みアプリケーションは失敗、停止、強制終了、またはキューに格納のステータスのサブタスクのみを開始します。

増分ロードジョブを再開すると、一括取り込みアプリケーションは最後に中断したところからソースデータのブロックを再開します。

- ▶ 実行するジョブの【アクション】メニューで、[再開] をクリックします。

サブタスクは、ソースオブジェクトごとに開始されます。

**注:** タスクのデプロイメントが失敗したためにジョブが[失敗] 状態になっている場合、[再開] オプションは使用できません。

## 増分ロードジョブの再開とリカバリ

一括取り込みアプリケーションは、エラーが原因で停止した増分ロードジョブと、ユーザーによって停止または強制終了されたジョブを、変更データを失うことなく再開できます。

最初のジョブの実行後、一括取り込みアプリケーションは変更がターゲットに適用されると、変更ストリーム内の処理位置の識別子を継続的に記録します。識別子は、ターゲット上の INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルに格納されます。

増分ロードジョブを再開すると、ジョブはリカバリテーブルに記録された最後の位置を使用して、ターゲットにロードする必要のある変更レコードを識別します。このプロセスにより、すべての変更がターゲットに確実に取り込まれます。

# アプリケーション取り込みジョブの再開時のスキーマドリフトオプションのオーバーライド

停止状態、強制終了状態、または失敗状態のアプリケーション取り込みジョブを再開するときにスキーマドリフトオプションをオーバーライドできます。オーバーライドは、**[オブジェクトの停止]**、**[テーブルの停止]**、**[レポートの停止]**、または **[ジョブの停止]** スキーマドリフトオプションによって、現在エラー状態にあるオブジェクトにのみ影響します。オーバーライドを使用して、これらのエラーを修正または解決します。

スキーマドリフトオプションをオーバーライドして、**一括取り込みサービス**の**一括取り込みページ**または**オペレーションインサイト**の**【一括取り込み】** ページにある**オペレーションインサイト**タブから**増分ロードジョブ**、または**初期および増分ロードジョブ**の組み合わせを再開できます。

- 1. オーバーライドを使用して再開するジョブの行に移動します。
- 2. 行の **[アクション]** メニューで、**[オプションで再開]** をクリックします。

**注:** タスクのデプロイメントが失敗したためにジョブが **[失敗]** 状態になっている場合、**[オプションで再開]** コマンドは使用できません。

**[オプションの再開]** ダイアログボックスが表示されます。

Resume Options

Schema Drift Options:

Select a schema drift option

Resume With Options

Cancel

- 3. **[スキーマドリフトオプション]** リストで、アプリケーション取り込みジョブを停止させたソースでの DDL 操作の処理に使用されるスキーマドリフトオプションを選択します。

次の表に、スキーマドリフトのオプションを示します。

オプション	説明
無視	ソースで発生する DDL の変更をターゲットにレプリケートしません。
テーブルの停止	DDL 変更が発生したソースオブジェクトの処理を停止します。 <b>重要:</b> アプリケーション取り込みジョブは、ジョブがソースオブジェクトの変更の処理を停止した後にソースオブジェクトで発生したデータ変更を取得できません。その結果、ターゲットでデータ損失が発生する可能性があります。データの損失を回避するには、ジョブが処理を停止したソースオブジェクトとターゲットオブジェクトを再同期する必要があります。 <b>[オプションで再開]</b> > <b>[再同期]</b> オプションを使用します。

オプション	説明
再同期	ターゲットテーブルをソースオブジェクトと再同期します。このオプションは、[スキーマドリフト] オプションの [オブジェクトの停止]、[テーブルの停止]、または [レポートの停止] 設定によって、ジョブが処理を停止したオブジェクトに使用します。 <b>重要:</b> このオプションは、初期ロードジョブと増分ロードジョブを組み合わせただけの場合のみ使用できます。
レプリケート	アプリケーション取り込みジョブが DDL 変更をターゲットにレプリケートできるようにします。 <b>重要:</b> Microsoft Azure Synapse Analytics ターゲットでのカラムの名前変更操作に [レプリケート] オプションを選択すると、エラーが発生してジョブが終了します。

4. [オプションで再開] をクリックします。

再開されたジョブは、ステップ 3 で指定したスキーマドリフトオプションを使用して、ジョブを停止させる原因となったスキーマの変更を処理します。その後、タスクの作成時に指定したスキーマドリフトオプションが再び有効になります。

**重要:** 一括取り込みアプリケーションがソースオブジェクトのスキーマ変更を処理するのは、オブジェクトで DML 操作が発生した後のみです。したがって、ジョブを再開した後にオブジェクトで最初の DML 操作が発生するまで、オブジェクトのサブタスクの状態は変更されません。

## アプリケーション取り込みジョブの再デプロイ

アプリケーション取り込みタスクの一部のフィールドは、関連するアプリケーション取り込みジョブをデプロイ解除しなくても編集可能です。関連するジョブをデプロイ解除せずにアプリケーション取り込みタスクの使用可能なフィールドを編集する場合、変更を有効にするためにジョブを再デプロイする必要があります。

編集する必要のあるフィールドが編集不可能な場合は、関連するジョブをデプロイ解除してから編集する必要があります。ジョブのデプロイ解除の詳細については、「[「アプリケーション取り込みジョブのデプロイ解除」\(ページ 140\)](#)」を参照してください。

ジョブを再デプロイするとき、一括取り込みアプリケーションは、ソースオブジェクトのすべてのサブタスクを停止します。サブタスクが停止した後、一括取り込みアプリケーションは、更新された取り込みタスクをデプロイしてから、サブタスクを開始します。開始されるサブタスクには、以前に停止されたサブタスクと、タスクの設定変更のために新しく作成されたサブタスクが含まれます。

**注:** 増分ロードジョブと初期および増分ロードジョブの組み合わせの場合、再デプロイでは、前のデプロイ中に取り込み対象として選択されたソースオブジェクトは変更されません。オブジェクトのリストを更新するには、関連するタスクのオブジェクト選択ルールを編集してから、ジョブを再デプロイする必要があります。

1. [マイジョブ] ページで、再デプロイするジョブの行に移動します。
2. 行の [アクション] メニューで、[再デプロイ] をクリックします。

ジョブインスタンスの実行が更新された設定で開始されます。

# アプリケーション取り込みジョブのデプロイ解除

実行する必要がなくなったアプリケーション取り込みジョブはデプロイ解除できます。また、関連付けられた取り込みタスクを編集して、ジョブをデプロイ解除しないと更新できない接続またはプロパティを更新する場合も、アプリケーション取り込みジョブをデプロイ解除できます。

ジョブをデプロイ解除した後は、ジョブを再度実行したり、再デプロイしたりすることはできません。デプロイ解除したジョブを実行する場合は、アプリケーション取り込みタスクウィザードから関連するタスクを再度デプロイして、新しいジョブインスタンスを作成する必要があります。例えば、ジョブのターゲット接続を変更する場合は、ジョブをデプロイ解除し、取り込みタスクを編集して接続を変更し、タスクを再度デプロイしてから、新しいジョブインスタンスを実行する必要があります。

1. ジョブが実行されていないことを確認します。
2. 一括取り込みの【**マイジョブ**】ページ、またはオペレーションインサイトの【**一括取り込み**】ページの【**すべてのジョブ**】タブで、デプロイ解除するジョブの行に移動します。
3. 行の【**アクション**】メニューで、【**デプロイ解除**】をクリックします。

デプロイ解除操作が失敗すると、ジョブのステータスは、[強制終了] 状態であっても[失敗] に切り替わるか、[失敗] のままになります。

**注:** ジョブをデプロイ解除した直後に、Secure Agent をシャットダウンしないようにしてください。一括取り込みアプリケーションが/root/infaagent/apps/Database\_Ingestion/data/tasks ディレクトリ内のタスクのファイルをクリーンアップするまでにはしばらく時間がかかります。

## アプリケーション取り込みジョブの強制終了

すべてのロードタイプアプリケーション取り込みジョブのうち、[稼働中]、[実行中(警告あり)]、[保留]、または[停止中] ステータスにあるものは強制終了できます。

アプリケーション取り込みサービスの【**マイジョブ**】ページまたは一括取り込みの【**一括取り込み**】ページの【**すべてのジョブ**】タブからオペレーションインサイトジョブを強制終了できます。

増分ロードジョブを強制終了すると、一括取り込みアプリケーションは、増分処理を停止した変更ストリーム内の位置の識別子を記録します。識別子は、ターゲット上の INFORMATICA\_CDC\_RECOVERY という名前のリカバリテーブルに格納されます。ジョブを再開すると、一括取り込みアプリケーションはこの識別子を使用して、ターゲットに最後にロードされた変更レコードを識別し、変更ストリーム内のその時点以降に行われた変更のロードを開始します。

初期ロードジョブの場合、すでに実行されているサブタスクはすぐに停止し、その後ジョブも停止します。実行されていないサブタスクは現在の状態のままです。

- ▶ 強制終了するジョブの【**アクション**】メニューで、【**強制終了**】を選択します。

ジョブのステータスが[強制終了中] に変わり、次に[強制終了] に変わります。

初期ロードジョブの場合、実行されていたサブタスクのステータスが[強制終了] に変わります。増分ロードジョブの場合、サブタスクのステータスが[停止] に変わります。

# ソースオブジェクトとターゲットオブジェクトの再同期

初期ロードタイプと増分ロードタイプを組み合わせた実行中のアプリケーション取り込みジョブの一部であるサブタスクのソースオブジェクトとターゲットオブジェクトを再同期できます。サブタスクは、キューに格納または開始以外の状態である必要があります。

例えば、初期ロードまたは増分ロードの処理が失敗した場合、または特定の再起動ポイントからジョブを最初からやり直す場合は、ターゲットをソースと再同期することをお勧めします。

**重要:** [オブジェクトの停止]、[テーブルの停止]、または [レポートの停止] の [スキーマドリフト] 設定によって停止して現在 [エラー] 状態であるオブジェクトを再同期するには、[アクション] メニューの [オプションで再開] > [再同期] オプションを使用する必要があります。詳細については、[「アプリケーション取り込みジョブの再開時のスキーマドリフトオプションのオーバーライド」](#) (ページ 138) を参照してください。

1. 一括取り込みサービスの [マイジョブ] ページ、またはオペレーションインサイトの [一括取り込み] ページの [すべてのジョブ] タブで、取り込みジョブをドリルダウンすることにより、ジョブの詳細を表示します。

ジョブは [稼働中] 状態で、初期ロード操作と増分ロード操作を組み合わせたものである必要があります。

2. [オブジェクトの詳細] タブをクリックします。
3. 再同期するソースオブジェクトとターゲットオブジェクトのサブタスク行で、[アクション] メニューをクリックし、[再同期] を選択します。再同期操作では、CDC 用に処理されたソーステーブルとターゲットテーブルの構造が保持されます。

**注:** [アクション] メニューと [再同期] オプションを使用可能にするには、サブタスクがキューに格納または開始以外の状態である必要があります。

Salesforce ソースを持つアプリケーション取り込み組み合わせロードジョブのサブタスクを再同期する場合は、[再同期] オプションの代わりに次のいずれかの再同期オプションを使用します。

- **再同期 (更新)。** ターゲットテーブルを最新のソースオブジェクト定義 (スキーマドリフトで無視された DDL 変更を含む) と再同期するには、このオプションを使用します。ターゲットテーブルが更新されると、ターゲットテーブルの構造は現行のソースオブジェクト構造と一致します。このオプションは、[再同期] オプションの動作を模倣します。
- **再同期 (保持)。** CDC 用に処理されたのと同じフィールドを再同期し、ソーステーブルとターゲットテーブルの現在の構造を保持するには、このオプションを使用します。ソーステーブル定義またはターゲットテーブル定義の変更のチェックは行われません。ソース DDL の変更がソースオブジェクト構造に影響を与えた場合、それらの変更は処理されません。

**注:**

- ソーステーブルに多くの行が含まれている場合、再同期の実行に長い時間がかかる可能性があります。
- ソーステーブルスキーマがターゲットテーブルスキーマと一致しない場合、取り込みサブタスクはターゲットテーブルを削除し、ソーススキーマと一致する新しいテーブルを作成します。ターゲットテーブルが再作成されるかどうかに関係なく、サブタスクはターゲットテーブルを切り詰めてから、ソースデータをテーブルに再ロードします。
- アプリケーション取り込みサブタスクを Snowflake ターゲットと再同期し、監査適用モードを使用すると、監査情報を保持できます。一括取り込みでは、ターゲットテーブルが再作成され、監査情報を含む既存のテーブルの名前にタイムスタンプが付加されて <target\_table\_name>\_<current\_UTC\_timestamp> 形式の名前に変更されます。実際のターゲットテーブルに監査情報が必要な場合は、結合操作などでロードする必要があります。既存のテーブル名にタイムスタンプを追加することによって名前が最大文字数を超えると、サブタスクはエラーで失敗します。スキーマドリフトを有効にし、スキーマドリフトの変更 (カラムの追加など) が発生した場合、新しいカラムは再作成されたターゲットテーブルには含まれますが、名前が変更され



たテーブルには含まれません。この動作を有効にするには、タスクウィザードの【ターゲット】 ページで backupTargetTableBeforeResync カスタムプロパティを **true** に設定します。

既存の監査テーブルがある組み合わせロードジョブを再同期する場合は、以下の制限を考慮してください。

- 既存の監査テーブルをターゲットに格納すると、データベースストレージが余分に消費されます。
- 監査情報の統合ビューを取得するには、監査テーブルの複数のバージョンを結合する必要があります。

## デフォルトデータ型のマッピング

このリファレンスは、ソースアプリケーションと Amazon Redshift、Databricks Delta、Google BigQuery、Microsoft Azure Synapse Analytics、Oracle、および Snowflake ターゲットのデフォルトのデータ型マッピングを提供します。一括取り込みアプリケーションでは、ターゲットデータの生成時にこれらのマッピングを使用します。

ターゲットを設定する場合、必要に応じてデータ型マッピングルールを定義して、ソースデータ型からターゲットテーブルへのデフォルトのマッピングをカスタマイズできます。詳細については、[「データ型マッピングをカスタマイズするためのルール」](#) (ページ 100)を参照してください。

ソースデータ型がリストされていない場合は、一括取り込みアプリケーションはこのデータ型のソースカラムからデータを抽出できないか、抽出したデータを適切なターゲットデータ型に適用できません。

## Adobe Analytics ソースと Amazon Redshift ターゲット

次の表は、Adobe Analytics ソースと Amazon Redshift ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Adobe Analytics ソースデータ型	Amazon Redshift ターゲットデータ型
currency(p,s), 1 <= p <= 38, 0 <= s <= 19	numeric(p,s), 1 <= p <= 38, 0 <= s <= 19
datetime(3)	timestamp without time zone
decimal(p,s), 1 <= p <= 38, 0 <= s <= 19	numeric(p,s), 1 <= p <= 38, 0 <= s <= 19
enum	binary varying(size), 264004 <= size <= 1020004
enum	character varying(65535)
enum	character varying(size), 4 <= size <= 65535
int(p,s), 1 <= p <= 38, 0 <= s <= 19	numeric(p,s), 1 <= p <= 38, 0 <= s <= 19
ordered-enum	binary varying(size), 264004 <= size <= 1020004
ordered-enum	character varying(65535)
ordered-enum	character varying(size), 4 <= size <= 65535
percent(p,s), 1 <= p <= 38, 0 <= s <= 19	numeric(p,s), 1 <= p <= 38, 0 <= s <= 19
string	binary varying(size), 264004 <= size <= 1020004



Adobe Analytics ソースデータ型	Amazon Redshift ターゲットデータ型
string	character varying(65535)
string	character varying( <i>size</i> ), 4 <= size <= 65535

## Adobe Analytics ソースと Google BigQuery ターゲット

次の表は、Adobe Analytics ソースと Google BigQuery ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Adobe Analytics ソースデータ型	Google BigQuery ターゲットデータ型
currency(p,s), 1 <= p <= 38, 0 <= s <= 19	bignumeric
datetime(3)	datetime
decimal(p,s), 1 <= p <= 38, 0 <= s <= 19	bignumeric
enum	string
int(p,s), 1 <= p <= 38, 0 <= s <= 19	bignumeric
ordered-enum	string
percent(p,s), 1 <= p <= 38, 0 <= s <= 19	bignumeric
string	string

## Adobe Analytics ソースと Microsoft Azure Synapse Analytics ターゲット

次の表は、Adobe Analytics ソースと Microsoft Azure Synapse Analytics ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Adobe Analytics ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
currency(p,s), 1 <= p <= 38, 0 <= s <= 19	decimal(p,s), 1 <= p <= 38, 1 <= s <= 19
datetime(3)	datetime
decimal(p,s), 1 <= p <= 38, 0 <= s <= 19	decimal(p,s), 1 <= p <= 38, 1 <= s <= 19
enum	varchar( <i>size</i> ), 4 <= size <= max
int(p,s), 1 <= p <= 38, 0 <= s <= 19	decimal(p,s), 1 <= p <= 38, 1 <= s <= 19
ordered-enum	varchar( <i>size</i> ), 4 <= size <= max

Adobe Analytics ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
percent(p,s), 1 <= p <= 38, 0 <= s <= 19	decimal(p,s), 1 <= p <= 38, 1 <= s <= 19
string	varchar(size), 4 <= size <= max

## Adobe Analytics ソースと Oracle ターゲット

次の表は、Adobe Analytics ソースと Oracle ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Adobe Analytics ソースデータ型	Oracle ターゲットデータ型
currency(p,s), 1 <= p <= 38, 0 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
datetime(3)	timestamp(3)
decimal(p,s), 1 <= p <= 38, 0 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
enum	char(s char), 4 <= s <= 2000
enum	clob
int(p,s), 1 <= p <= 38, 0 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
ordered-enum	char(s char), 4 <= size <= 2000
ordered-enum	clob
percent(p,s), 1 <= p <= 38, 0 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
string	char(s char), 4 <= size <= 2000
string	clob

## Adobe Analytics ソースと Snowflake ターゲット

次の表は、Adobe Analytics ソースと Snowflake ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Adobe Analytics ソースデータ型	Snowflake ターゲットデータ型
currency(p,0), 1 <= p <= 38	integer
currency(p,s), 1 <= p <= 38, 1 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
datetime(3)	datetime(3)
decimal(p,0), 1 <= p <= 38	integer
decimal(p,s), 1 <= p <= 38, 1 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38

Adobe Analytics ソースデータ型	Snowflake ターゲットデータ型
enum	char( <i>size</i> ), 4 <= size <= 4192004
int(p,0), 1 <= p <= 38	integer
int(p,s), 1 <= p <= 38, 1 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
ordered-enum	char( <i>size</i> ), 4 <= size <= 4192004
percent(p,0), 1 <= p <= 38	integer
percent(p,s), 1 <= p <= 38, 1 <= s <= 19	number(p,s), 1 <= p <= 19, 1 <= s <= 38
string	char( <i>size</i> ), 4 <= size <= 16777216

## NetSuite ソースと Amazon Redshift ターゲット

次の表は、NetSuite ソースと Amazon Redshift ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

NetSuite ソースデータ型	Amazon Redshift ターゲットデータ型
bigint	bigint
char	character varying(4)
clob	binary varying( <i>size</i> ), 262144 <= size <= 400000
clob	character varying( <i>size</i> ), 4 <= size <= 65535
double(p,s), 1 <= p <= 38, 0 <= s <= 18	numeric(p,s), 1 <= p <= 38, 0 <= s <= 18
number	character varying( <i>size</i> ), 40 <= size <= 41
number	numeric(p,s), 1 <= p <= 38, 0 <= s <= 16
timestamp	timestamp without time zone
varchar2	character varying( <i>size</i> ), 4 <= size <= 65535

## NetSuite ソースと Databricks Delta ターゲット

次の表は、NetSuite ソースと Databricks Delta ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

NetSuite ソースデータ型	Databricks Delta ターゲットデータ型
bigint	byte
bigint	integer

NetSuite ソースデータ型	Databricks Delta ターゲットデータ型
bigint	long
char	string
clob	string
double(p,s), 1 <= p <= 38, 0 <= s <= 18	decimal(p,s), 1 <= p <= 38, 1 <= s <= 18
number	decimal(p,s), 1 <= p <= 38, 0 <= s <= 16
number	string
timestamp	timestamp
varchar2	string

## NetSuite ソースと Google BigQuery ターゲット

次の表は、NetSuite ソースと Google BigQuery ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

NetSuite ソースデータ型	Google BigQuery ターゲットデータ型
bigint	int64
char	string
clob	string
double(p,s), 1 <= p <= 38, 0 <= s <= 18	bignumeric
number	bignumeric
number	bignumeric
number	numeric
timestamp	datetime
varchar2	string

## NetSuite ソースと Microsoft Azure Synapse Analytics ターゲット

次の表は、NetSuite ソースと Microsoft Azure Synapse Analytics ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

NetSuite ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
bigint	bigint
char	nchar
clob	char(8000)
clob	nchar( <i>size</i> ), 1 <= size <= 4000
clob	nvarchar(max)
clob	nvarchar(max)
double(p,s), 1 <= p <= 38, 0 <= s <= 18	decimal(p,s), 1 <= p <= 38, 1 <= s <= 18
number	decimal(p,s), 1 <= p <= 38, 0 <= s <= 16
number	varchar( <i>size</i> ), 40 <= size <= 41
timestamp	datetime
varchar2	varchar( <i>size</i> ), 4 <= size <= max

## NetSuite ソースと Oracle ターゲット

次の表は、NetSuite ソースと Oracle ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

NetSuite ソースデータ型	Oracle ターゲットデータ型
bigint	char(s char), 2 <= s <= 13
char	char(4 char)
clob	char(s char), 4 <= s <= 2000
clob	clob
double(p,s), 1 <= p <= 38, 0 <= s <= 18	number(p,s), 1 <= p <= 18, 1 <= s <= 38
number	char(s char), 40 <= s <= 41
number	number(p,s), 1 <= p <= 16, 0 <= s <= 38
timestamp	timestamp(1)

NetSuite ソースデータ型	Oracle ターゲットデータ型
varchar2	char(s char), 4 <= s <= 2000
varchar2	clob

## NetSuite ソースと Snowflake ターゲット

次の表は、NetSuite ソースと Snowflake ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

NetSuite ソースデータ型	Snowflake ターゲットデータ型
bigint	char(size), 2 <= size <= 13
char	char(4)
clob	char(size), 4 <= size <= 400000
double(p,0), 1 <= p <= 38	integer
double(p,s), 1 <= p <= 38, 1 <= s <= 18	number(p,s), 1 <= p <= 18, 1 <= s <= 38
number	number(p,s), 1 <= p <= 16, 0 <= s <= 38
number	varchar(size), 40 <= size <= 41
timestamp	datetime(1)
varchar2	varchar(size), 4 <= size <= 262136

## Salesforce ソースと Amazon Redshift ターゲット

次の表は、Salesforce ソースと Amazon Redshift ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	Amazon Redshift ターゲットデータ型
anytype(precision), 1 <= p <= 255	character varying(size), 1 <= size <= 255
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	character varying(size), 1 <= size <= 65535
boolean	boolean
combobox(precision), 1 <= p <= 255	character varying(size), 1 <= size <= 255
currency(p,s), 1 <= p <= 18, 0 <= s <= 17	numeric(p,s), 1 <= p <= 18, 0 <= s <= 17
date	date

Salesforce ソースデータ型	Amazon Redshift ターゲットデータ型
datetime	timestamp without time zone
double(p,s), 1 <= p <= 18, 0 <= s <= 17	numeric(p,s), 1 <= p <= 18, 0 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	character varying( <i>size</i> ), 1 <= size <= 80
encryptedstring( <i>precision</i> ), 1 <= p <= 175	character varying( <i>size</i> ), 1 <= size <= 175
id( <i>precision</i> ), 1 <= p <= 18	character varying( <i>size</i> ), 1 <= size <= 18
int	integer
long	bigint
multipicklist( <i>precision</i> ), 1 <= p <= 4099	character varying( <i>size</i> ), 1 <= size <= 4099
percent(p,s), 1 <= p <= 18, 0 <= s <= 17	numeric(p,s), 1 <= p <= 18, 0 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	character varying( <i>size</i> ), 1 <= size <= 40
picklist( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
reference( <i>precision</i> ), 1 <= p <= 18	character varying( <i>size</i> ), 1 <= size <= 18
string( <i>precision</i> ), 1 <= p <= 100	character varying( <i>size</i> ), 1 <= size <= 100
textarea( <i>precision</i> ), 1 <= p <= 131072	character varying( <i>size</i> ), 1 <= size <= 65535
time	time without time zone
url( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255

## Salesforce ソースと Databricks Delta ターゲット

次の表は、Salesforce ソースと Databricks Delta ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	Databricks Delta ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	string
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	string
boolean	boolean
combobox( <i>precision</i> ), 1 <= p <= 255	string
currency(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
date	string



Salesforce ソースデータ型	Databricks Delta ターゲットデータ型
datetime	timestamp
double(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	string
encryptedstring( <i>precision</i> ), 1 <= p <= 175	string
id( <i>precision</i> ), 1 <= p <= 18	string
int	integer
long	long
multipicklist( <i>precision</i> ), 1 <= p <= 4099	string
percent(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	string
picklist( <i>precision</i> ), 1 <= p <= 255	string
reference( <i>precision</i> ), 1 <= p <= 18	string
string( <i>precision</i> ), 1 <= p <= 100	string
textarea( <i>precision</i> ), 1 <= p <= 131072	string
time	string
url( <i>precision</i> ), 1 <= p <= 255	string

## Salesforce ソースと Google BigQuery ターゲット

次の表は、Salesforce ソースと Google BigQuery ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	Google BigQuery ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	string
base64 <b>注:</b> 精度は、選択した base64 ボディサイズによって異なります。	string
boolean	bool
combobox( <i>precision</i> ), 1 <= p <= 255	string
currency(p,0), 1 <= p <= 18	numeric
currency(p,s), 1 <= p <= 18, 1 <= s <= 9	numeric

Salesforce ソースデータ型	Google BigQuery ターゲットデータ型
currency(p,s), 1 <= p <= 18, 10 <= s <= 17	bignumeric
date	date
datetime	datetime
double(p,0), 1 <= p <= 18	numeric
double(p,s), 1 <= p <= 18, 1 <= s <= 9	numeric
double(p,s), 1 <= p <= 18, 10 <= s <= 17	bignumeric
email( <i>precision</i> ), 1 <= p <= 80	string
encryptedstring( <i>precision</i> ), 1 <= p <= 175	string
id( <i>precision</i> ), 1 <= p <= 18	string
int	int64
long	int64
multipicklist( <i>precision</i> ), 1 <= p <= 4099	string
percent(p,0), 1 <= p <= 18	numeric
percent(p,s), 1 <= p <= 18, 1 <= s <= 9	numeric
percent(p,s), 1 <= p <= 18, 10 <= s <= 17	bignumeric
phone( <i>precision</i> ), 1 <= p <= 40	string
picklist( <i>precision</i> ), 1 <= p <= 255	string
reference( <i>precision</i> ), 1 <= p <= 18	string
string( <i>precision</i> ), 1 <= p <= 100	string
textarea( <i>precision</i> ), 1 <= p <= 131072	string
time	time
url( <i>precision</i> ), 1 <= p <= 255	string

## Salesforce ソースと Microsoft Azure Synapse Analytics ターゲット

次の表は、Salesforce ソースと Microsoft Azure Synapse Analytics ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	varchar( <i>size</i> ), 1 <= size <= 255
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	nchar( <i>size</i> ), 1 <= size <= 255
boolean	bit
combobox( <i>precision</i> ), 1 <= p <= 255	varchar( <i>size</i> ), 1 <= size <= 255
currency(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
date	date
datetime	datetime2(3)
double(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	varchar( <i>size</i> ), 1 <= size <= 80
encryptedstring( <i>precision</i> ), 1 <= p <= 175	varchar( <i>size</i> ), 1 <= size <= 175
id( <i>precision</i> ), 1 <= p <= 18	varchar( <i>size</i> ), 1 <= size <= 18
int	int
long	bigint
multipicklist( <i>precision</i> ), 1 <= p <= 4099	varchar( <i>size</i> ), 1 <= size <= 4099
percent(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	varchar( <i>size</i> ), 1 <= size <= 40
picklist( <i>precision</i> ), 1 <= p <= 255	varchar( <i>size</i> ), 1 <= size <= 255
reference( <i>precision</i> ), 1 <= p <= 18	varchar( <i>size</i> ), 1 <= size <= 18
string( <i>precision</i> ), 1 <= p <= 100	varchar( <i>size</i> ), 1 <= size <= 100
textarea( <i>precision</i> ), 1 <= p <= 131072	varchar( <i>size</i> ), 1 <= size <= max
time	time(3)
url( <i>precision</i> ), 1 <= p <= 255	varchar( <i>size</i> ), 1 <= size <= 255

## Salesforce ソースと Microsoft Azure SQL Database ターゲット

次の表は、Salesforce ソースと Microsoft Azure SQL Database ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	PostgreSQL ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	character varying( <i>size</i> ), 1 <= size <= max
boolean	bit
combobox( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
currency(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
date	date
datetime	datetime2(3)
double(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	character varying( <i>size</i> ), 1 <= size <= 80
encryptedstring( <i>precision</i> ), 1 <= p <= 175	character varying( <i>size</i> ), 1 <= size <= 175
id( <i>precision</i> ), 1 <= p <= 18	character varying( <i>size</i> ), 1 <= size <= 18
int	<b>整数型</b>
long	long
multipicklist( <i>precision</i> ), 1 <= p <= 4099	character varying( <i>size</i> ), 1 <= size <= 4099
percent(p,s), 1 <= p <= 18, 0 <= s <= 17	decimal(p,s), 1 <= p <= 18, 1 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	character varying( <i>size</i> ), 1 <= size <= 40
picklist( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
reference( <i>precision</i> ), 1 <= p <= 18	character varying( <i>size</i> ), 1 <= size <= 18
string( <i>precision</i> ), 1 <= p <= 100	character varying( <i>size</i> ), 1 <= size <= 100
textarea( <i>precision</i> ), 1 <= p <= 131072	character varying( <i>size</i> ), 1 <= size <= 131072
time	time( <i>precision</i> ) without time zone, 3 <= p <= null
url( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255

## Salesforce ソースと Oracle ターゲット

次の表は、Salesforce ソースと Oracle ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	Oracle ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	char(s char), 1 <= s <= 255
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	clob
boolean	char(1 char)
combobox( <i>precision</i> ), 1 <= p <= 255	char(s char), 1 <= s <= 255
currency(p,s), 1 <= p <= 18, 0 <= s <= 17	number(p,s), 1 <= p <= 18, 1 <= s <= 17
date	date
datetime	timestamp(3)
double(p,s), 1 <= p <= 18, 0 <= s <= 17	number(p,s), 1 <= p <= 18, 1 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	char(s char), 1 <= s <= 80
encryptedstring( <i>precision</i> ), 1 <= p <= 175	char(s char), 1 <= s <= 175
id( <i>precision</i> ), 1 <= p <= 18	char(s char), 1 <= s <= 18
int	number(10)
long	number(19)
multipicklist( <i>precision</i> ), 1 <= p <= 4099	clob
percent(p,s), 1 <= p <= 18, 0 <= s <= 17	number(p,s), 1 <= p <= 18, 1 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	char(s char), 1 <= s <= 40
picklist( <i>precision</i> ), 1 <= p <= 255	char(s char), 1 <= s <= 255
reference( <i>precision</i> ), 1 <= p <= 18	char(s char), 1 <= s <= 18
string( <i>precision</i> ), 1 <= p <= 100	char(s char), 1 <= s <= 100
textarea( <i>precision</i> ), 1 <= p <= 4000	char(s char), 1 <= s <= 4000
textarea( <i>precision</i> ), 4001 <= p <= 131072	clob
time	timestamp(3)
url( <i>precision</i> ), 1 <= p <= 255	char(s char), 1 <= s <= 255

## Salesforce ソースと PostgreSQL ターゲット

次の表は、Salesforce ソースと PostgreSQL ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	PostgreSQL ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	character varying( <i>size</i> ), 1 <= size <= max
boolean	boolean
combobox( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
currency(p,s), 1 <= p <= 18, 0 <= s <= 17	numeric(p,s), 1 <= p <= 18, 1 <= s <= 17
date	date
datetime	timestamp( <i>precision</i> ) without time zone, 3 <= p <= null
double(p,s), 1 <= p <= 18, 0 <= s <= 17	numeric(p,s), 1 <= p <= 18, 1 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	character varying( <i>size</i> ), 1 <= size <= 80
encryptedstring( <i>precision</i> ), 1 <= p <= 175	character varying( <i>size</i> ), 1 <= size <= 175
id( <i>precision</i> ), 1 <= p <= 18	character varying( <i>size</i> ), 1 <= size <= 18
int	integer
long	bigint
multipicklist( <i>precision</i> ), 1 <= p <= 4099	character varying( <i>size</i> ), 1 <= size <= 4099
percent(p,s), 1 <= p <= 18, 0 <= s <= 17	numeric(p,s), 1 <= p <= 18, 1 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	character varying( <i>size</i> ), 1 <= size <= 40
picklist( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255
reference( <i>precision</i> ), 1 <= p <= 18	character varying( <i>size</i> ), 1 <= size <= 18
string( <i>precision</i> ), 1 <= p <= 100	character varying( <i>size</i> ), 1 <= size <= 100
textarea( <i>precision</i> ), 1 <= p <= 131072	character varying( <i>size</i> ), 1 <= size <= 131072
time	time( <i>precision</i> ) without time zone, 3 <= p <= null
url( <i>precision</i> ), 1 <= p <= 255	character varying( <i>size</i> ), 1 <= size <= 255

## Salesforce ソースと Snowflake ターゲット

次の表は、Salesforce ソースと Snowflake ターゲットを使用した一括取り込みアプリケーション設定のデフォルトデータ型マッピングを示しています。

Salesforce ソースデータ型	Snowflake ターゲットデータ型
anytype( <i>precision</i> ), 1 <= p <= 255	char( <i>size</i> ), 4 <= size <= 1020
base64 注: 精度は、選択した base64 ボディサイズによって異なります。	char( <i>size</i> ), 4 <= size <= 1020
boolean	boolean
combobox( <i>precision</i> ), 1 <= p <= 255	char( <i>size</i> ), 4 <= size <= 1020
currency(p,0), 1 <= p <= 18	integer
currency(p,s), 1 <= p <= 18, 1 <= s <= 17	number(p,s), 1 <= p <= 18, 1 <= s <= 17
date	date
datetime	datetime(3)
double(p,0), 1 <= p <= 18	integer
double(p,s), 1 <= p <= 18, 1 <= s <= 17	number(p,s), 1 <= p <= 18, 1 <= s <= 17
email( <i>precision</i> ), 1 <= p <= 80	char( <i>size</i> ), 4 <= size <= 320
encryptedstring( <i>precision</i> ), 1 <= p <= 175	char( <i>size</i> ), 4 <= size <= 700
id( <i>precision</i> ), 1 <= p <= 18	char( <i>size</i> ), 4 <= size <= 72
int	number(10)
long	number(19)
multipicklist( <i>precision</i> ), 1 <= p <= 4099	char( <i>size</i> ), 4 <= size <= 16396
percent(p,0), 1 <= p <= 18	integer
percent(p,s), 1 <= p <= 18, 1 <= s <= 17	number(p,s), 1 <= p <= 18, 1 <= s <= 17
phone( <i>precision</i> ), 1 <= p <= 40	char( <i>size</i> ), 4 <= size <= 160
picklist( <i>precision</i> ), 1 <= p <= 255	char( <i>size</i> ), 4 <= size <= 1020
reference( <i>precision</i> ), 1 <= p <= 18	char( <i>size</i> ), 4 <= size <= 72
string( <i>precision</i> ), 1 <= p <= 100	char( <i>size</i> ), 4 <= size <= 400
textarea( <i>precision</i> ), 1 <= p <= 131072	char( <i>size</i> ), 4 <= size <= 131072



Salesforce ソースデータ型	Snowflake ターゲットデータ型
time	time(3)
url( <i>precision</i> ), 1 <= p <= 255	char( <i>size</i> ), 4 <= size <= 1020

## Salesforce Marketing Cloud ソースと Amazon Redshift ターゲット

次の表は、Salesforce Marketing Cloud ソースと Amazon Redshift ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Salesforce Marketing Cloud ソースデータ型	Amazon Redshift ターゲットデータ型
boolean	boolean
date(0)	timestamp without time zone
decimal(p,s), 1 <= p <= 29, 0 <= s <= 8	numeric(p,s), 1 <= p <= 29, 0 <= s <= 8
emailaddress( <i>precision</i> ), 1 <= p <= 254	character varying( <i>size</i> ), 4 <= size <= 1016
locale( <i>precision</i> ), 1 <= p <= 5	character varying( <i>size</i> ), 4 <= size <= 20
number	integer
phone( <i>precision</i> ), 1 <= p <= 50	character varying( <i>size</i> ), 4 <= size <= 200
text( <i>precision</i> ), 1 <= p <= 4000	character varying( <i>size</i> ), 4 <= size <= 16000

## Salesforce Marketing Cloud ソースと Databricks Delta ターゲット

次の表は、Salesforce Marketing Cloud ソースと Databricks Delta ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Salesforce Marketing Cloud ソースデータ型	Databricks Delta ターゲットデータ型
boolean	boolean
date(0)	timestamp
decimal(p,s), 1 <= p <= 29, 0 <= s <= 8	decimal(p,s), 1 <= p <= 29, 1 <= s <= 8
emailaddress( <i>precision</i> ), 1 <= p <= 254	string
locale( <i>precision</i> ), 1 <= p <= 5	string
number	integer

Salesforce Marketing Cloud ソースデータ型	Databricks Delta ターゲットデータ型
phone( <i>precision</i> ), 1 <= p <= 50	string
text( <i>precision</i> ), 1 <= p <= 4000	string

## Salesforce Marketing Cloud ソースと Google BigQuery ターゲット

次の表は、Salesforce Marketing Cloud ソースと Google BigQuery ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Salesforce Marketing Cloud ソースデータ型	Google BigQuery ターゲットデータ型
boolean	bool
date(0)	datetime
decimal(p,s), 1 <= p <= 29, 0 <= s <= 8	bignumeric
emailaddress( <i>precision</i> ), 1 <= p <= 254	string
locale( <i>precision</i> ), 1 <= p <= 5	string
number	int64
phone( <i>precision</i> ), 1 <= p <= 50	string
text( <i>precision</i> ), 1 <= p <= 4000	string

## Salesforce Marketing Cloud ソースと Microsoft Azure Synapse Analytics ターゲット

次の表は、Salesforce Marketing Cloud ソースと Microsoft Azure Synapse Analytics ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Salesforce Marketing Cloud ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
boolean	bit
date(0)	datetime
decimal(p,s), 1 <= p <= 29, 0 <= s <= 8	decimal(p,s), 1 <= p <= 29, 1 <= s <= 8
emailaddress( <i>precision</i> ), 1 <= p <= 254	nchar( <i>size</i> ), 1 <= size <= 254
locale( <i>precision</i> ), 1 <= p <= 5	nchar( <i>size</i> ), 1 <= size <= 5
number	int

Salesforce Marketing Cloud ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
phone( <i>precision</i> ), 1 <= p <= 50	nchar( <i>size</i> ), 1 <= size <= 50
text( <i>precision</i> ), 1 <= p <= 4000	nchar( <i>size</i> ), 1 <= size <= 4000

## Salesforce Marketing Cloud ソースと Oracle ターゲット

次の表は、Salesforce Marketing Cloud ソースと Oracle ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Salesforce Marketing Cloud ソースデータ型	Oracle ターゲットデータ型
boolean	char(1 char)
date(0)	date
decimal(p,s), 1 <= p <= 29, 0 <= s <= 8	number(p,s), 1 <= p <= 29, 1 <= s <= 8
emailaddress( <i>precision</i> ), 1 <= p <= 254	char(s char), 4 <= s <= 1016
locale( <i>precision</i> ), 1 <= p <= 5	char(s char), 4 <= s <= 20
number	number(10)
phone( <i>precision</i> ), 1 <= p <= 50	char(s char), 4 <= s <= 200
text( <i>precision</i> ), 1 <= p <= 2000	char(s char), 4 <= s <= 2000
text( <i>precision</i> ), 2001 <= p <= 4000	clob

## Salesforce Marketing Cloud ソースと Snowflake ターゲット

次の表は、Salesforce Marketing Cloud ソースと Snowflake ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

Salesforce Marketing Cloud ソースデータ型	Snowflake ターゲットデータ型
boolean	boolean
date(0)	datetime(0)
decimal(p,0), 1 <= p <= 29	integer
decimal(p,s), 1 <= p <= 29, 1 <= s <= 8	number(p,s), 1 <= p <= 29, 1 <= s <= 8
emailaddress( <i>precision</i> ), 1 <= p <= 254	char( <i>size</i> ), 4 <= size <= 1016
locale( <i>precision</i> ), 1 <= p <= 5	char( <i>size</i> ), 4 <= size <= 20
number	number(10)

Salesforce Marketing Cloud ソースデータ型	Snowflake ターゲットデータ型
phone( <i>precision</i> ), 1 <= p <= 50	char( <i>size</i> ), 4 <= size <= 200
text( <i>precision</i> ), 1 <= p <= 4000	char( <i>size</i> ), 4 <= size <= 16000

## SAP ソースと Amazon Redshift ターゲット

次の表は、SAP ソースと Amazon Redshift ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

SAP ソースデータ型	Amazon Redshift ターゲットデータ型
d16r	double precision
d34r	double precision
fltp	double precision
int1	smallint
int2	smallint
int4	integer
int8	bigint

### サポートされていないソースデータ型

一括取り込みアプリケーションは、次の廃止された SAP データ型をサポートしません。

- D16S
- D34S
- PREC
- VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型をサポートしません。

## SAP ソースと Databricks Delta ターゲット

次の表は、SAP ソースと Databricks Delta ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

SAP ソースデータ型	Databricks Delta ターゲットデータ型
d16r	double
d34r	double
fltp	double
int1	integer

SAP ソースデータ型	Databricks Delta ターゲットデータ型
int2	integer
int4	integer
int8	long

#### サポートされていないソースデータ型

一括取り込みアプリケーションは、次の廃止された SAP データ型をサポートしません。

- D16S
- D34S
- PREC
- VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型をサポートしません。

## SAP ソースと Google BigQuery ターゲット

次の表は、SAP ソースと Google BigQuery ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

SAP ソースデータ型	Google BigQuery ターゲットデータ型
d16r	float64
d34r	float64
fltp	float64
int1	int64
int2	int64
int4	int64
int8	int64

#### サポートされていないソースデータ型

一括取り込みアプリケーションは、次の廃止された SAP データ型をサポートしません。

- D16S
- D34S
- PREC
- VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型をサポートしません。

## SAP ソースと Microsoft Azure Synapse Analytics ターゲット

次の表は、SAP ソースと Microsoft Azure Synapse Analytics ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

SAP ソースデータ型	Microsoft Azure Synapse Analytics ターゲットデータ型
d16r	float
d34r	float
fltp	float
int1	tinyint
int2	smallint
int4	int
int8	bigint

### サポートされていないソースデータ型

一括取り込みアプリケーションは、次の廃止された SAP データ型をサポートしません。

- D16S
- D34S
- PREC
- VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型をサポートしません。

## SAP ソースと Oracle ターゲット

次の表は、SAP ソースと Oracle ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

SAP ソースデータ型	Oracle ターゲットデータ型
CHAR	VARCHAR2 (20)
ACCP	VARCHAR2(24)
CURR	NUMBER(31,8)
CUKY	VARCHAR2 (20)
DF16_DEC	BINARY_DOUBLE
DF16_RAW	BINARY_DOUBLE
DF34_DEC	BINARY_DOUBLE
DF34_RAW	BINARY_DOUBLE

SAP ソースデータ型	Oracle ターゲットデータ型
DEC	NUMBER(31,9)
DATS	DATE
CLNT	VARCHAR2(12)
INT2	NUMBER(5,0)
LANG	VARCHAR2(4)
NUMC	VARCHAR2 (40)
RAW	BLOB
SSTRING	CLOB
STRING	CLOB
TIMS	TIMESTAMP
QUAN	NUMBER(31,6)
UNIT	VARCHAR2(12)
INT1	NUMBER(3,0)
INT4	NUMBER(10,0)
LCHR	VARCHAR2 (2000)

#### サポートされていないソースデータ型

一括取り込みアプリケーションは、次の廃止された SAP データ型をサポートしません。

- D16S
- D34S
- PREC
- VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型をサポートしません。

## SAP ソースと Snowflake ターゲット

次の表は、SAP ソースと Snowflake ターゲットを使用した一括取り込みアプリケーション設定の推奨データ型マッピングを示しています。

SAP ソースデータ型	Snowflake ターゲットデータ型
d16r	float
d34r	float



SAP ソースデータ型	Snowflake ターゲットデータ型
fltp	float
int1	number(3)
int2	number(5)
int4	number(10)
int8	number(19)

#### サポートされていないソースデータ型

一括取り込みアプリケーションは、次の廃止された SAP データ型をサポートしません。

- D16S
- D34S
- PREC
- VARC

さらに、アプリケーション取り込みジョブは、RSTR データ型をサポートしません。

# 索引

## A

### Adobe Analytics ソース

- Amazon Redshift ターゲットを使用したマッピング [142](#)
- Google BigQuery ターゲットを使用したマッピング [143](#)
- Microsoft Azure Synapse Analytics ターゲットを使用したマッピング [143](#)
- Oracle ターゲットを使用したマッピング [144](#)
- Snowflake ターゲットを使用したマッピング [144](#)

### Amazon Redshift ターゲット

- Adobe Analytics ソースを使用したマッピング [142](#)
- NetSuite ソースを使用したマッピング [145](#)
- Salesforce Marketing Cloud ソースを使用したマッピング [157](#)
- Salesforce ソースを使用したマッピング [148](#)
- SAP ソースを使用したマッピング [160](#)

## C

### Cloud アプリケーション統合コミュニティ

URL [6](#)

### Cloud 開発者コミュニティ

URL [6](#)

## D

### Databricks Delta ターゲット

- NetSuite ソースを使用したマッピング [145](#)
- Salesforce Marketing Cloud ソースを使用したマッピング [157](#)
- Salesforce ソースを使用したマッピング [149](#)
- SAP ソースを使用したマッピング [160](#)

## G

### Google BigQuery ターゲット

- Adobe Analytics ソースを使用したマッピング [143](#)
- NetSuite ソースを使用したマッピング [146](#)
- Salesforce Marketing Cloud ソースを使用したマッピング [158](#)
- Salesforce ソースを使用したマッピング [150](#)
- SAP ソースを使用したマッピング [161](#)

## I

### Informatica Intelligent Cloud Services

Web サイト [6](#)

### Informatica グローバルカスタマサポート

連絡先情報 [7](#)

## M

### Microsoft Azure SQL Database ターゲット

- Salesforce ソースを使用したマッピング [153](#)

### Microsoft Azure Synapse Analytics ターゲット

- Adobe Analytics ソースを使用したマッピング [143](#)
- NetSuite ソースを使用したマッピング [147](#)
- Salesforce Marketing Cloud ソースを使用したマッピング [158](#)
- Salesforce ソースを使用したマッピング [152](#)
- SAP ソースを使用したマッピング [162](#)

## N

### NetSuite ソース

- Amazon Redshift ターゲットを使用したマッピング [145](#)
- Databricks Delta ターゲットを使用したマッピング [145](#)
- Google BigQuery ターゲットを使用したマッピング [146](#)
- Microsoft Azure Synapse Analytics ターゲットを使用したマッピング [147](#)
- Oracle ターゲットを使用したマッピング [147](#)
- Snowflake ターゲットを使用したマッピング [148](#)

## O

### Oracle

- アプリケーション取り込みのターゲットガイドライン [44](#)

### Oracle ターゲット

- Adobe Analytics ソースを使用したマッピング [144](#)
- NetSuite ソースを使用したマッピング [147](#)
- Salesforce Marketing Cloud ソースを使用したマッピング [159](#)
- Salesforce ソースを使用したマッピング [154](#)
- SAP ソースを使用したマッピング [162](#)

## P

### PostgreSQL ターゲット

- Salesforce ソースを使用したマッピング [155](#)

## S

### Salesforce Marketing Cloud ソース

- Amazon Redshift ターゲットを使用したマッピング [157](#)
- Databricks Delta ターゲットを使用したマッピング [157](#)
- Google BigQuery ターゲットを使用したマッピング [158](#)
- Microsoft Azure Synapse Analytics ターゲットを使用したマッピング [158](#)
- Oracle ターゲットを使用したマッピング [159](#)
- Snowflake ターゲットを使用したマッピング [159](#)

### Salesforce ソース

- Amazon Redshift ターゲットを使用したマッピング [148](#)
- Databricks Delta ターゲットを使用したマッピング [149](#)
- Google BigQuery ターゲットを使用したマッピング [150](#)
- Microsoft Azure SQL Database ターゲットを使用したマッピング [153](#)
- Microsoft Azure Synapse Analytics ターゲットを使用したマッピング [152](#)

Salesforce ソース (続く)

Oracle ターゲットを使用したマッピング [154](#)

PostgreSQL ターゲットを使用したマッピング [155](#)

Snowflake ターゲットを使用したマッピング [156](#)

SAP

SAP ODP を処理するためのトランスポートファイルのインストール [18](#)

SAP ODP を読み取るためのトランスポートファイルのインストール [18](#)

SAP ソース

Amazon Redshift ターゲットを使用したマッピング [160](#)

Databricks Delta ターゲットを使用したマッピング [160](#)

Google BigQuery ターゲットを使用したマッピング [161](#)

Microsoft Azure Synapse Analytics ターゲットを使用したマッピング [162](#)

Oracle ターゲットを使用したマッピング [162](#)

Snowflake ターゲットを使用したマッピング [163](#)

SAP ユーザー権限

テーブルデータの処理設定 [17](#)

Snowflake ターゲット

Adobe Analytics ソースを使用したマッピング [144](#)

NetSuite ソースを使用したマッピング [148](#)

Salesforce Marketing Cloud ソースを使用したマッピング [159](#)

Salesforce ソースを使用したマッピング [156](#)

SAP ソースを使用したマッピング [163](#)

アプリケーション取り込みジョブ

スキーマドリフトオプションのオーバーライド [138](#)

ソースオブジェクトとターゲットオブジェクトの再同期 [141](#)

アプリケーション取り込みタスク

アプリケーション取り込みタスクの基本情報の定義 [52](#)

ソースの設定 [53](#)

ターゲットの設定 [98](#)

ランタイムオプションの設定 [132](#)

## し

システムステータス [7](#)

## す

ステータス

Informatica Intelligent Cloud Services [7](#)

## て

データ型マッピング

デフォルトマッピングのカスタマイズ [100](#)

## W

Web サイト [6](#)

## あ

アップグレード通知 [7](#)

## め

メンテナンスの停止 [7](#)