

Pentaho 機能説明 PDI 編

前書き

■ 対象プログラム

Pentaho 10.2

■ 輸出時の注意

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。

なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ 発行

2025 年 3 月

■ 著作権

All Rights Reserved. Copyright (C) 2025, Hitachi, Ltd.

Copyright (C) 2025 Hitachi Vantara LLC. All Rights Reserved.

はじめに

このドキュメントは PDI の機能について説明したものです。

■ このドキュメントで使用する略語

このドキュメントで使用している英略語を次の表に示します。

英略語	英字での表記
CSV	Character Separated Values
CR	Carriage Return
ETL	Extract, Transform, Loading
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
JVM	Java Virtual Machine
LDAP	Lightweight Directory Access Protocol
LF	Line Feed
PDF	Portable Document Format
PDI	Pentaho Data Integration
POSIX	Portable Operating System Interface for UNIX
PUC	Pentaho User Console
RTF	Rich Text Format
SQL	Structured Query Language
SSL	Secure Sockets Layer
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VFS	Virtual File System
XMI	XML Metadata Interchange
XML	Extensible Markup Language

■ このドキュメントで使用する製品名

このドキュメントで使用している表記と製品名を次の表に示します。

表記	製品名
Excel	Microsoft Excel

■ このドキュメントで使用する記号

このドキュメントで使用している記号を次の表に示します。

記号	意味
[]	画面名、ボタン名や画面上のテキストを示します。
<>	ユーザーまたはシステムによって提供される実際のテキストの代用文字列を表します。

目次

前書き 2

はじめに 3

1 PDI クライアント (Spoon) 10

1.1 PDI クライアント (Spoon) の概要 11

1.2 [データ変換]パースpekティブ 12

1.3 [Scheduler]パースpekティブ 18

1.4 [Kettle オプション] 20

2 データ変換 24

2.1 ステップの一覧 25

2.1.1 ステップの説明について 30

2.1.2 データタイプの対応表 31

2.2 入力 32

2.2.1 [入力]Access 入力 (Microsoft Access input) 32

2.2.2 [入力]CSV 入力 (CSV file input) 38

2.2.3 [入力]ESRI shapefile reader (ESRI Shapefile Reader) 41

2.2.4 [入力]Excel 入力 (Microsoft Excel input) 42

2.2.5 [入力]Get data from XML (Get data from XML) 50

2.2.6 [入力]JSON input (JSON Input) 56

2.2.7 [入力]キューブ入力 (De-serialize from file) 62

2.2.8 [入力]システムデータ取得 (Get system info) 63

2.2.9 [入力]テキストファイル入力 (Text file input) 67

2.2.10 [入力]テーブル入力 (Table input) 78

2.2.11 [入力]データグリッド (Data grid) 79

2.2.12 [入力]ファイルメモリロード (Load file content in memory) 81

2.2.13 [入力]ファイル名取得 (Get file names) 87

2.2.14 [入力]固定幅ファイル入力 (Fixed File input) 90

2.2.15 [入力]行生成 (Generate rows) 92

2.3 出力 94

2.3.1 [出力]JSON output (JSON output) 94

2.3.2 [出力]Microsoft Excel writer (Microsoft Excel writer) 98

2.3.3 [出力]Pentaho レポート出力 (Pentaho reporting output) 103

2.3.4 [出力]XML output (XML output) 104

2.3.5 [出力]キューブ出力 (Serialize to file) 108

2.3.6	[出力]テキストファイル出力 (Text file output)	109
2.3.7	[出力]テーブル出力 (Table output)	113
2.3.8	[出力]削除 (Delete)	118
2.3.9	[出力]挿入/更新 (Insert / update)	120
2.3.10	[出力]更新 (Update)	122
2.4	変換	125
2.4.1	[変換]Add a checksum (Add a checksum)	125
2.4.2	[変換]シーケンス追加 (Add sequence)	127
2.4.3	[変換]シーケンス追加 (変化値) (Add value fields changing sequence)	129
2.4.4	[変換]フィールド値設定 (Set field value)	130
2.4.5	[変換]フィールド分割 (Split fields)	131
2.4.6	[変換]フィールド分割 (行) (Split field to rows)	133
2.4.7	[変換]フィールド定数設定 (Set field value to a constant)	134
2.4.8	[変換]フィールド結合 (Concat fields)	135
2.4.9	[変換]値マッピング (Value mapper)	139
2.4.10	[変換]定数追加 (Add constants)	140
2.4.11	[変換]数値範囲 (Number range)	141
2.4.12	[変換]文字列オペレーション (String operations)	143
2.4.13	[変換]文字列カット (Strings cut)	145
2.4.14	[変換]文字列置換 (Replace in string)	146
2.4.15	[変換]行フラット化 (Row Flattener)	148
2.4.16	[変換]行整列 (Sort rows)	148
2.4.17	[変換]行正規化 (Row normaliser)	150
2.4.18	[変換]行非正規化 (Row denormaliser)	151
2.4.19	[変換]計算 (Calculator)	153
2.4.20	[変換]選択/名前変更 (Select values)	159
2.4.21	[変換]重複行削除 (Unique rows)	162
2.4.22	[変換]重複行削除(HashSet) (Unique rows (HashSet))	163
2.5	ユーティリティ	165
2.5.1	[ユーティリティ]NULL 値定数変換 (If field value is null)	165
2.5.2	[ユーティリティ]ファイル処理 (Process files)	167
2.5.3	[ユーティリティ]プロセス実行 (Execute a process)	169
2.5.4	[ユーティリティ]メタデータ構造 (Metadata structure of stream)	170
2.5.5	[ユーティリティ]メール (Mail)	171
2.5.6	[ユーティリティ]ログ書き込み (Write to log)	179
2.5.7	[ユーティリティ]定数 NULL 変換 (Null if)	181
2.5.8	[ユーティリティ]行複製 (Clone row)	182
2.6	フロー	184
2.6.1	[フロー]Blocking step (Blocking step)	184

2.6.2	[フロー]Java フィルター (Java Filter)	185
2.6.3	[フロー]ジョブ実行 (Job Executor)	186
2.6.4	[フロー]ステップブロック (Block this step until steps finish)	192
2.6.5	[フロー]ストリーム結合 (Append streams)	192
2.6.6	[フロー]ダミー (何もしない) (Dummy (do nothing))	193
2.6.7	[フロー]トランスフォーメーション実行 (Transformation executor)	194
2.6.8	[フロー]フィルター (Filter rows)	201
2.6.9	[フロー]停止 (Abort)	203
2.6.10	[フロー]優先順位 (Prioritize streams)	204
2.6.11	[フロー]条件分岐 (Switch/case) (Switch / case)	205
2.6.12	[フロー]空ストリーム探知 (Detect empty stream)	206
2.7	スクリプト	208
2.7.1	[スクリプト]Java スクリプト (Modified Java Script value)	208
2.7.2	[スクリプト]Python Executor (Python Executor)	220
2.7.3	[スクリプト]SQL 文実行 (Execute SQL script)	228
2.7.4	[スクリプト]SQL 文実行(行) (Execute row SQL script)	231
2.7.5	[スクリプト]ユーザー定義 Java クラス (User defined java expression)	232
2.7.6	[スクリプト]数式 (Formula)	236
2.7.7	[スクリプト]正規表現 (Regex evaluation)	245
2.8	参照	250
2.8.1	[参照]HTTP クライアント (HTTP Client)	250
2.8.2	[参照]REST クライアント (REST Client)	254
2.8.3	[参照]ストリーム参照 (Stream lookup)	260
2.8.4	[参照]テーブル確認 (Table Exists (Step))	262
2.8.5	[参照]データベース参照 (Database lookup)	263
2.8.6	[参照]データベース結合 (Database join)	265
2.8.7	[参照]ファイル確認 (File exists (Step))	266
2.9	結合	269
2.9.1	[結合]XML 結合 (XML Join)	269
2.9.2	[結合]マージ結合 (Merge join)	271
2.9.3	[結合]多方向 Merge Join (Multiway merge join)	272
2.9.4	[結合]整列併合 (Sorted merge)	273
2.9.5	[結合]行マージ (比較) (Merge rows (diff))	274
2.9.6	[結合]行結合 (デカルト積) (Join rows (cartesian product))	275
2.10	データウェアハウス	278
2.10.1	[データウェアハウス]ディメンジョン 参照/更新 (Dimension lookup/update)	278
2.11	データ検証	286
2.11.1	[データ検証]データ検証 (Data Validator)	286
2.12	統計	290

2.12.1	[統計]グループ化 (Group by)	290
2.12.2	[統計]サンプル行 (Sample rows)	293
2.12.3	[統計]メモリーグループ化 (Memory group by)	293
2.12.4	[統計]分析クエリー (Analytic query)	296
2.12.5	[統計]単変量統計 (Univariate statistics)	297
2.12.6	[統計]貯留層サンプリング (Reservoir sampling)	299
2.13	ジョブ	301
2.13.1	[ジョブ]変数取得 (Get Variables)	301
2.13.2	[ジョブ]変数設定 (Set Variables)	302
2.13.3	[ジョブ]行取得 (Get rows from result)	303
2.13.4	[ジョブ]行送信 (Copy rows to result)	304
2.14	マッピング	306
2.14.1	[マッピング]シンプルマッピング (サブデータ変換) (Simple Mapping (sub-transformation))	306
2.14.2	[マッピング]マッピング (サブデータ変換) (Mapping (sub-transformation))	310
2.14.3	[マッピング]マッピング入力 (Mapping input specification)	314
2.14.4	[マッピング]マッピング出力 (Mapping output specification)	315
2.15	バルクロード	316
2.15.1	[バルクロード]PostgreSQL バルクロード (PostgreSQL bulk loader)	316
2.16	新機能	319
2.16.1	[新機能]SFTP put (SFTP put)	319

3 ジョブ 323

3.1	ジョブエントリーの一覧	324
3.1.1	ジョブエントリーの説明について	326
3.2	全般	327
3.2.1	[全般]START (Start)	327
3.2.2	[全般]Dummy (Dummy)	329
3.2.3	[全般]ジョブ (Job)	329
3.2.4	[全般]データ変換 (Transformation)	336
3.2.5	[全般]変数設定 (Set variables (job entry))	342
3.2.6	[全般]成功 (Success)	343
3.3	メール	344
3.3.1	[メール]メール (Mail)	344
3.4	ファイル管理	351
3.4.1	[ファイル管理]HTTP (HTTP)	351
3.4.2	[ファイル管理]ファイルへ書き込み (Write to file)	354
3.4.3	[ファイル管理]ファイルコピー (Copy Files)	355
3.4.4	[ファイル管理]ファイル作成 (Create file)	358
3.4.5	[ファイル管理]ファイル削除 (Delete file)	358

3.4.6	[ファイル管理]ファイル削除（複数）(Delete files)	359
3.4.7	[ファイル管理]ファイル圧縮 (Zip file)	360
3.4.8	[ファイル管理]ファイル待機 (Wait for a file)	365
3.4.9	[ファイル管理]ファイル移動 (Move files)	366
3.4.10	[ファイル管理]ファイル解凍 (Unzip file)	371
3.4.11	[ファイル管理]フォルダー作成 (Create a folder)	377
3.4.12	[ファイル管理]フォルダー削除 (Delete folders)	377
3.5	条件	380
3.5.1	[条件]テーブル検証 (Evaluate rows number in a table)	380
3.5.2	[条件]テーブル確認 (Table exists)	382
3.5.3	[条件]データベース接続確認 (Check Db Connections)	382
3.5.4	[条件]ファイル確認 (File Exists)	383
3.5.5	[条件]ファイル確認（複数）(Checks if files exists)	384
3.5.6	[条件]待機 (Wait for)	385
3.5.7	[条件]簡易検証 (Simple evaluation)	385
3.6	スクリプト	389
3.6.1	[スクリプト]JavaScript (JavaScript)	389
3.6.2	[スクリプト]SQL (SQL)	391
3.6.3	[スクリプト]シェル (Shell)	392
3.7	ユーティリティ	396
3.7.1	[ユーティリティ]Ping 送信 (Ping a host)	396
3.7.2	[ユーティリティ]ログ書き込み (Write to log)	397
3.7.3	[ユーティリティ]停止 (Abort job)	398
3.8	ファイル転送	399
3.8.1	[ファイル転送]FTP ファイル取得 (Get a file with FTP)	399
3.8.2	[ファイル転送]FTP ファイル追加 (Put a file with FTP)	404
3.8.3	[ファイル転送]FTP 削除 (FTP delete)	408
3.8.4	[ファイル転送]SFTP ファイル取得 (Get a file with SFTP)	413
3.8.5	[ファイル転送]SFTP ファイル追加 (Put a file with SFTP)	416

索引 420

1

PDI クライアント（Spoon）

PDI クライアント（Spoon）の機能概要や画面について説明します。

1.1 PDI クライアント (Spoon) の概要

PDI クライアント (Spoon) は、データ変換やジョブを GUI を使った直感的な操作で作成、定義構築、スケジュール設定するためのデスクトップアプリケーションです。PDI クライアント (Spoon) で作成、定義したデータ変換やジョブは Pentaho Server で実行します。

PDI クライアント (Spoon) では、次のパースペクティブを使って、データ変換やジョブを作成、定義します。

- [\[データ変換\]パースペクティブ](#)
- [\[Scheduler\]パースペクティブ](#)

また、[\[Kettle オプション\]](#)を使用して、ログや表示などの各種設定をします。

1.2 [データ変換]パースペクティブ

[データ変換]パースペクティブでは、データ変換やジョブに関する各種操作ができます。

図 1-1 [データ変換]パースペクティブの[データ変換]画面

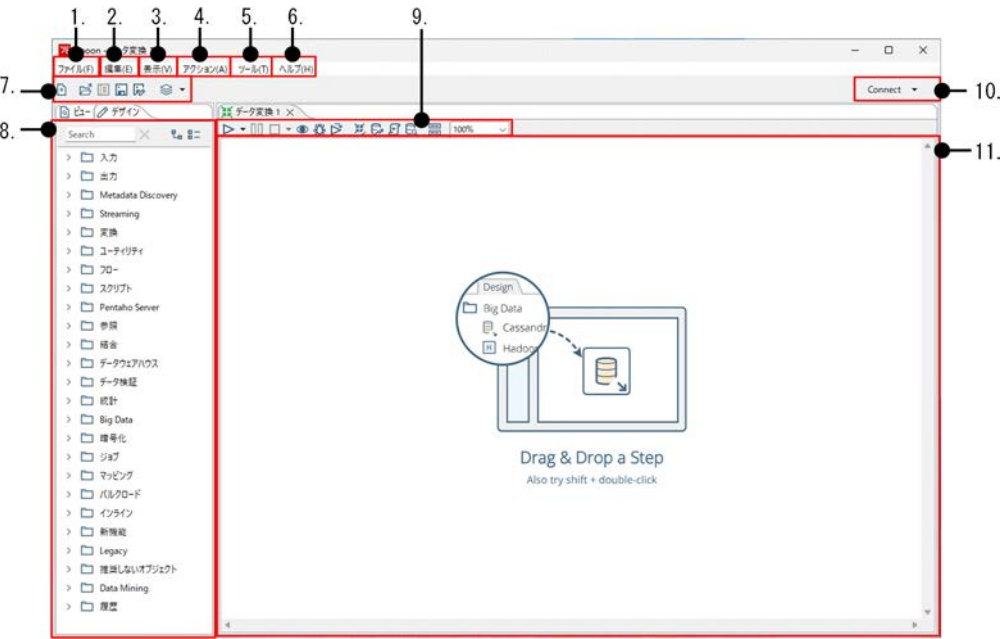


表 1-1 [データ変換]パースペクティブの[データ変換]画面の項目

項番	項目	説明
1	[ファイル]	ファイルについて次の操作ができます。 • [新規(N)] データ変換、ジョブ、データベース接続を新規に作成します。 • [開く(O)] リポジトリに接続していない場合、ローカルのファイルシステムからファイルを開きます。リポジトリに接続している場合、リポジトリに保存されたファイルを開きます。 • [閉じる(C)] キャンバスに現在表示しているタブを閉じます。 • [すべて閉じる] キャンバスに表示しているすべてのタブを閉じます。 • [保存(S)] リポジトリに接続していない場合、ローカルのファイルシステムにファイルを保存します。 リポジトリに接続している場合、リポジトリにファイルを保存します。 • [名前を付けて保存(A)] リポジトリに接続していない場合、ローカルのファイルシステムに名前を付けてファイルを保存します。 リポジトリに接続している場合、リポジトリに名前を付けてファイルを保存します。 • [印刷]

項番	項目	説明
		[印刷] ダイアログボックスが開きます。 • [XML ファイルからインポート(I)] • [エクスポート] • [終了(X)] Spoon を終了します。
2	[編集]	次の操作ができます。 • [元に戻す] • [やり直し] • [切り取り] • [コピー] • [ファイルコピー] • [貼り付け] • [キャンバスをクリップボードに保存] • [選択解除(C)] • [全て選択(S)] • [メタデータ検索(M)] データ変換またはジョブのメタデータを検索します。 メタデータとは、データ変換／ジョブファイル名、ステップ／DB 接続／メモ、設定値、設定値の設定個所のことです。 メタデータ検索をする際、条件を指定して、結果を絞り込みます。 • [環境変数設定(V)] 環境変数を設定します。環境変数については、Pentaho Documentation を参照してください。 • [設定済み環境変数表示(V)] 設定済みの環境変数を表示します。 • [kettle.properties ファイル編集] kettle.properties ファイルを編集します。 • [引数表示(U)] Spoon 起動時に指定した引数を表示します。 • [設定(S)] [データ変換プロパティ]画面、または[ジョブプロパティ]画面を表示します。
3	[表示]	データの表示に関する次の操作ができます。 • [パースペクティブ] パースペクティブを切り替えます。[データ変換]パースペクティブと、[Scheduler]パースペクティブがあります。 • [拡大(I)] キャンバスを拡大します。 • [縮小(O)] キャンバスを縮小します。 • [リセット] キャンバスの拡大・縮小をリセットします。 • [実行結果]

項番	項目	説明
		[実行結果]パネルに、[ログ]、[実行履歴]、[実行状況]、[パフォーマンスグラフ]、[メトリクス]、[データをプレビュー]の情報が表示されます。
4	[アクション]	データ変換に関する次の操作ができます。 • [実行(R)] データ変換、またはジョブを実行します。 • [Run Options] [データ変換の実行]ダイアログボックスを開きます。[Run Configuration]を指定して、データ変換の実行をPDI クライアント (Spoon) で実施するか、Pentaho Server で実施するかなどを選択します。 • [プレビュー(P)] • [デバッグ(D)] デバッグ条件を設定する画面でブレーク条件を設定します。ブレーク条件に合致したときに、処理を途中で止められます。 • [再実行(R)] データ変換の処理を再実行します。 • [確認(V)] [データ変換チェックの結果]ダイアログボックスでデータ変換の結果を確認できます。 • [影響(I)] データソースが、データ変換によってどのような影響を受けるのかを確認できます。データ変換が正常に完了した場合、影響分析が実行され、データソースへの影響について判断されます。 • [SQL 取得(G)] ロードされたデータ変換を実行するために必要な SQL を生成します。 • [直前を表示] 影響分析、結果確認、および結果プレビューができます。 • [Schedule] [Scheduler]パースペクティブ画面を表示します。
5	[ツール]	ツールに関する次の情報を表示します。 • [データベース] データベースエクスプローラーを表示します。 • [リポジトリ(R)] リポジトリ接続してる場合に有効です。リポジトリエクスプローラーを表示します。リポジトリのインポート、エクスポートができます。 • [ウィザード(W)] • [オプション(O)] [Kettle オプション]画面を表示します。 • [プラグイン情報を表示] インストール済みのプラグイン情報を表示します。
6	[ヘルプ]	次の操作ができます。 • [ようこそ画面(S)] [キャンバスにようこそ]画面を表示します。 • [ドキュメント]

項番	項目	説明
		Pentaho Documentation のページを開きます。 • [License Manager] [ライセンス管理]画面を表示します。 • [Spoon について(A)] Spoon のバージョン、コピーライトを表示します。
7	ツールバー	次の操作ができます。 • [新規] 新しいジョブ、データ変換、データベース接続、スレーブサーバ定義を作成します。 • [開く] ファイルからデータ変換、またはジョブファイルを開きます。 • [リポジトリ照会] リポジトリを照会します。 • [上書き保存] 現在開いているデータ変換またはジョブを、ファイルまたはリポジトリに保存します。 • [別名で保存] リポジトリに接続していない場合、ローカルのファイルシステムに名前を付けてファイルを保存します。 リポジトリに接続している場合、リポジトリに名前を付けてファイルを保存します。 • [パースペクティブ] パースペクティブを切り替えます。パースペクティブは次のとおりです。 ・ [データ変換] データ変換とジョブの作成をします。 ・ [Scheduler] Pentaho Server でスケジュールされた ETL アクティビティを管理します。
8	エクスプローラーパネル	次の操作ができます。 • [ビュー]タブ 使用できるデータベース接続とデータ変換、またはジョブに使用されるステップとホップに関する情報が表示されます。 • [デザイン]タブ データ変換またはジョブの作成に使用されるステップ、またはエントリーのリストが表示されます。
9	サブツールバー	次の操作ができます。 • [実行] データ変換、またはジョブを実行します。 • [一時停止] 実行中のデータ変換、またはジョブを一時停止します。 • [停止] 実行中のデータ変換、またはジョブを停止します。次のどちらかを選択します。 ・ [Stop] データ変換、またはジョブをすぐに停止します。 ・ [Stop input processing]

項番	項目	説明
		すでに取得または開始されたレコードを処理したあとで、データ変換、またはジョブへの入力ステップを停止します。 • [プレビュー] プレビューモードでデータ変換を実行して、選択したステップによって生成された行を調べます。 • [デバッグ] データ変換をデバッグモードで実行して、実行エラーのトラブルシューティングを実行します。 • [再実行] データ変換、またはジョブの処理を再実行します。 • [確認] [データ変換チェックの結果]ダイアログボックスでデータ変換の結果を確認できます。 • [影響分析] データベースで影響分析を実行します。 • [実行に必要な SQL を取得] ロードされたデータ変換、またはジョブを実行するために必要な SQL を生成します。 • [データベース接続の照会] データベースエクスプローラーを起動して、データのプレビュー、SQL クエリーの実行、DDL の生成などのアクションを実行します。 • [実行結果パネルの表示] 実行結果パネルを表示します。
10	[Connect] ボタン	次のどちらかの場合に、このボタンを使用します。 • リポジトリ接続の定義を作成する • リポジトリに接続する なお、リポジトリとは、ETL ジョブやデータ変換を保存する場所を指します。
11	キャンバス	実行する ETL アクティビティのデータ変換とジョブを設計、および構築します。

図 1-2 [データ変換]パースペクティブの[ジョブ]画面

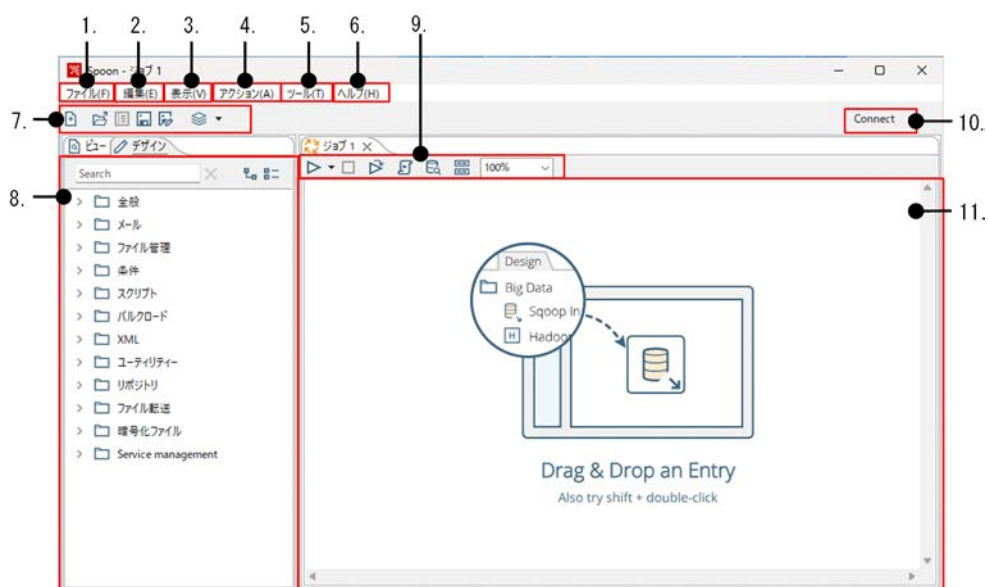


表 1-2 [データ変換]パースペクティブの[ジョブ]画面の項目

項番	項目	説明
1	[ファイル]	表 1-1[データ変換]パースペクティブの[データ変換]画面の項目を参照してください。
2	[編集]	
3	[表示]	
4	[アクション]	
5	[ツール]	
6	[ヘルプ]	
7	ツールバー	
8	エクスプローラーパネル	
9	サブツールバー	次の操作ができます。 • [実行] • [停止] • [再実行] • [実行に必要な SQL を取得] • [データベース接続の照会] • [実行結果パネルの表示] 詳細は、表 1-1[データ変換]パースペクティブの[データ変換]画面の項目を参照してください。
10	[Connect]ボタン	表 1-1[データ変換]パースペクティブの[データ変換]画面の項目を参照してください。
11	キャンバス	表 1-1[データ変換]パースペクティブの[データ変換]画面の項目を参照してください。

1.3 [Scheduler]パースペクティブ

[Scheduler]パースペクティブでは、データ変換やジョブの実行のスケジュールに関する各種設定ができます。

図 1-3 [Scheduler]パースペクティブ

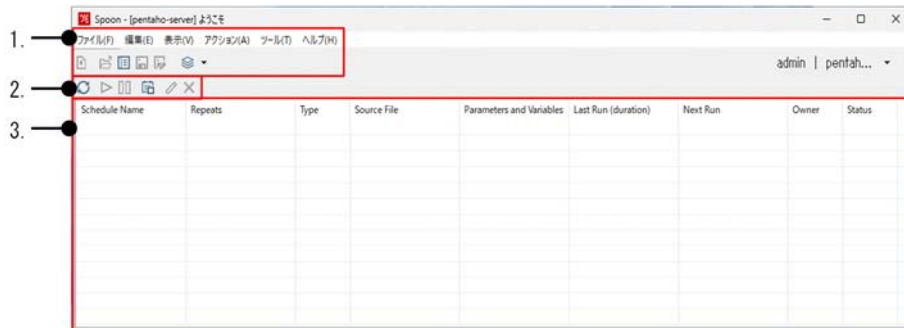


表 1-3 [Scheduler]パースペクティブ画面の項目

項番	項目	説明
1	メニューバー	アクション、ツールなどの一般的な機能を使用できます。メニューバーの右アイコンで、パースペクティブを切り替えることもできます。
2	メインツールバー	一覧更新、スケジュール有効化、スケジュール無効化、停止、編集、削除などの一般的な機能をシングルクリックで使用できます。
3	スケジュールパネル	スケジュールのリストが表示されます。強調表示されたスケジュールをダブルクリックすると、編集用の[Schedule]ダイアログボックスが表示されます。

図 1-4 [Schedule]ダイアログボックス

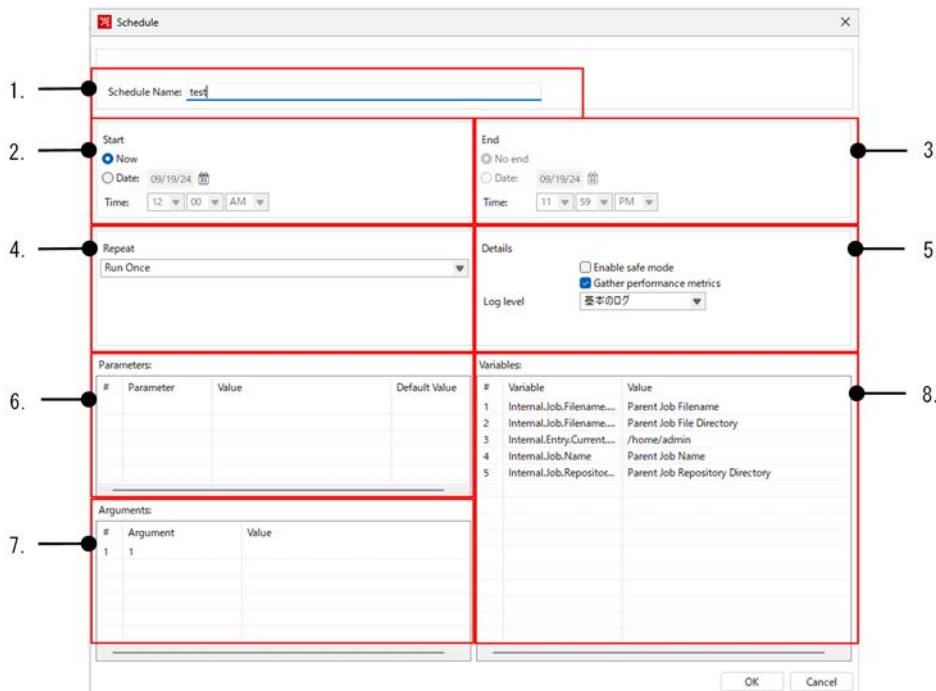


表 1-4 [Schedule]ダイアログボックスの項目

項番	項目	説明
1	Schedule Name	スケジュール名を指定します。
2	Start	• [Now] この項目を選択すると、データ変換、またはジョブをすぐに実行します。 • [Date] この項目を選択すると、設定した日時にデータ変換、またはジョブを実行します。
3	End	• [No end] この項目を選択すると、スケジュールに有効期限を設定しません。 • [Date] この項目を選択すると、設定した日時がスケジュールの有効期限になります。
4	Repeat	• [Run Once] データ変換、またはジョブを 1 回実行します。 • [Run Once 以外] 実行間隔を指定します。指定した間隔でデータ変換、またはジョブを実行します。
5	Details	• [Enable Safe Mode] この項目にチェックを付けると、セーフモードでデータ変換、またはジョブを実行します。セーフモードでは、ストリームを通過するすべての行をチェックして、すべてのレイアウトが最初の行と同じであることを確認します。レイアウトが異なる場合、エラーが発生します。 • [Gather performance metrics] • [Log level] ログレベルを設定します。設定できるログレベルは次のとおりです。 ・ [ログなし]：ログを出力しません。 ・ [エラーログのみ]：エラーログだけを出力します。 ・ [最小限のログ]：最小限のログだけを出力します。 ・ [基本のログ]：デフォルトのログレベルです。 ・ [詳細なログ]：詳細なログを出力します。 ・ [デバッグ]：非常に詳細なログを出力します。目的がデバッグの場合に設定します。 ・ [ローレベル（非常に詳細）]：大量のログデータを出力します。
6	Parameters	データ変換、またはジョブ実行時に指定するパラメータです。
7	Arguments	データ変換、またはジョブ実行時に指定する引数です。
8	Variables	データ変換、またはジョブ実行時に指定する変数です。

1.4 [Kettle オプション]

[Kettle オプション]では、PDI クライアント（Spoon）のオプション設定ができます。[一般]タブでは表示や保存などに関する設定、[外観]タブではフォントや背景色などに関する設定ができます。

図 1-5 [Kettle オプション]-[一般]タブ

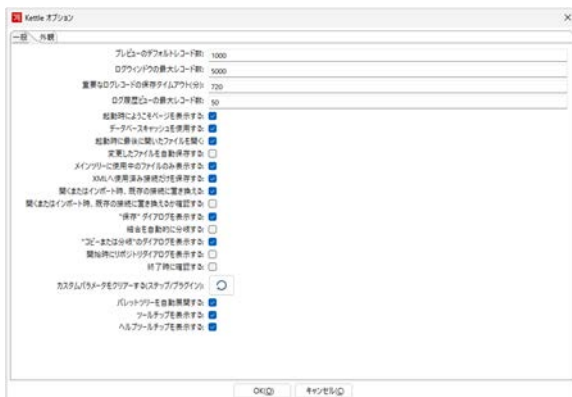


表 1-5 [Kettle オプション]-[一般]タブの項目

項番	項目	説明
1	[プレビューのデフォルトレコード数]	プレビューのデフォルトレコード数を設定します。
2	[ログウィンドウの最大レコード数]	ログウィンドウに表示する行の最大数を指定します。
3	[重要なログレコードの保存タイムアウト(分)]	この機能は動作しません。
4	[ログ履歴ビューの最大レコード数]	この機能は動作しません。
5	[起動時にようこそページを表示する]	この項目にチェックを付けると、PDI クライアント（Spoon）の起動時にウェルカムページを表示します。
6	[データベースキャッシュを使用する]	この項目にチェックを付けると、ソースデータベースとターゲットデータベースに保存されている情報をキャッシュします。 データベースを変更するときに、キャッシュによって誤った結果が発生することがあります。その際、エラーを防ぐために、毎回キャッシュをクリアする代わりに、キャッシュを完全に無効にできます。
7	[起動時に最後に開いたファイルを開く]	この項目にチェックを付けると、最後に開いたデータ変換、または保存したデータ変換を、XML またはリポジトリから自動的にロードします。
8	[変更したファイルを自動保存する]	この項目にチェックを付けると、実行する前に変更されたデータ変換を自動的に保存します。
9	[メインツリーに使用中のファイルのみを表示する]	この項目にチェックを付けると、現在アクティブなファイルだけを表示します。これにより、左側のメインツリーのデータ変換とジョブアイテムの数を減らせます。

項番	項目	説明
10	[XML へ使用済み接続だけを保存する]	この項目にチェックを付けると、トランスフォーメーションの XML エクスポートを、そのトランスフォーメーションで使用される接続に制限します。これは、サンプル変換を交換するときに、定義されたすべての接続が含まれるのを防ぎます。
11	[開くまたはインポート時、既存の接続に置き換える]	この項目にチェックを付けると、インポート中に、既存のデータベース接続などのオブジェクトを置き換えます。置き換えの前には、アクセス許可が要求されます。 [開くまたはインポート時、既存の接続に置き換えるか確認する]にもチェックを付けている場合は、インポートの実行前にプロンプトが表示されます。
12	[開くまたはインポート時、既存の接続に置き換えるか確認する]	この項目にチェックを付けると、インポート中に、既存のデータベース接続などのオブジェクトを置き換える前に、アクセス許可が要求されます。
13	["保存"ダイアログを表示する]	この機能は動作しません。
14	[結合を自動的に分岐する]	この項目にチェックを付けると、ホップを分割するときに起動する確認メッセージを無効にします。
15	["コピーまたは分岐"のダイアログを表示する]	この項目にチェックを付けると、複数のステップに出力をリンクしたときに警告メッセージを表示します（デフォルト）。この警告メッセージは、複数の出力を処理する際に次のどちらのオプションを選択するかを確認しています。 • [分配] 宛先のステップは、行を順番に受け取ります（ラウンドロビン）。 • [コピー] すべての行がすべての宛先に送信されます。
16	[開始時にリポジトリダイアログを表示する]	この項目にチェックを付けると、起動時に[リポジトリ]ダイアログボックスを表示します。
17	[終了時に確認する]	この項目にチェックを付けると、ユーザーがアプリケーションを終了するときに、[確認]ダイアログボックスを表示します。
18	[カスタムパラメータをクリアする(ステップ/プラグイン)]	この項目にチェックを付けると、プラグインまたは[ステップ]ダイアログボックスで設定されたすべてのパラメータとフラグをクリアします。
19	[パレットツリーを自動展開する]	この項目にチェックを付けると、パレットツリーを自動的に折りたたみます。この項目は、項目名と異なり、パレットツリーを自動で折りたたむ機能です。
20	[ツールチップを表示する]	この機能は動作しません。
21	[ヘルプツールチップを表示する]	この機能は動作しません。

図 1-6 [Kettle オプション]-[外観]タブ



表 1-6 [Kettle オプション]-[外観]タブの項目

項番	項目	説明
1	[固定フォント]	ダイアログボックス、ツリー、入力フィールドなどで使用されるフォントをカスタマイズします。[編集]ボタンをクリックするとフォントを編集できます。また、[このオプションをデフォルト値にリセット]ボタンをクリックすると、フォントをデフォルト値に戻せます。
2	[ワークスペースのフォント]	PDI クライアント (Spoon) のインターフェイスで使用されるフォントをカスタマイズします。[編集]ボタンをクリックするとフォントを編集できます。また、[削除]ボタンをクリックするとフォントをデフォルト値に戻せます。
3	[メモのフォント]	PDI クライアント (Spoon) に表示されるメモで使用されるフォントをカスタマイズします。[編集]ボタンをクリックするとフォントを編集できます。また、[削除]ボタンをクリックするとフォントをデフォルト値に戻せます。
4	[背景色]	PDI クライアント (Spoon) の背景色を設定します。この設定は、すべてのダイアログボックスに影響します。[編集]ボタンをクリックすると色を編集できます。また、[削除]ボタンをクリックすると背景色をデフォルト値に戻せます。
5	[ワークスペースの背景色]	PDI クライアント (Spoon) のグラフィカルビューの背景色を設定します。[編集]ボタンをクリックすると背景色を編集できます。また、[削除]ボタンをクリックすると背景色をデフォルト値に戻せます。
6	[タブの色]	アクティブ選択されているタブを示すために使用されている色をカスタマイズします。[編集]ボタンをクリックするとタブの色を編集できます。また、[削除]ボタンをクリックすると色をデフォルト値に戻せます。
7	[ワークスペースのアイコンサイズ]	グラフウィンドウのアイコンのサイズを変更します。アイコンのデフォルトのサイズは 32x32 ピクセルです。最良の結果 (グラフィック) は、サイズ 16、24、32、48、64 およびその他の 32 の倍数です。
8	[ワークスペースの線幅]	PDI クライアント (Spoon) のグラフィカルビューのホップの線幅とステップの周囲の境界線を変更します。
9	[ワークスペースの影のサイズ]	このサイズが 0 より大きい場合、ステップ、ホップ、およびノートの影がキャンバスに描画され、データ変換がキャンバスの上に浮かんでいるように見えます。
10	[ダイアログの中心(%)]	デフォルトでは、パラメータは左から数えてダイアログボックスの幅の 35% で描画されます。非常に大きなフォントを使用するなどして、ダイアログのラベルが切れて表示される場合は、このオプションで変更できます。

項番	項目	説明
11	[グリッドサイズ]	PDI クライアント (Spoon) のキャンバス上のグリッドのサイズを示します。
12	[Show Canvas Grid]	この項目にチェックを付けると、PDI クライアント (Spoon) のキャンバスに点線のグリッドが表示されます。
13	[キャンバスのアンチエイリアス]	この項目にチェックを付けると、グラフビューの線とアイコンを滑らかにします。
14	[ボトルネックのトランスフォーメーションステップを表示]	この項目にチェックを付けると、データ変換のステップの処理が遅い場合はそのステップの周囲にグラフィックが表示され、ボトルネックが表示されます。
15	[OS の外観を使用する]	Windows でこの項目にチェックを付けると、PDI クライアント (Spoon) のフォントと色のデフォルトのシステム設定を使用できます。他のプラットフォームでは、デフォルトのシステム設定が常に有効になっています。
16	[ブランドロゴなどの画像を表示する]	この機能は動作しません。
17	[推奨言語 (1 次言語)]	優先言語設定を指定します。

2

データ変換

この章では、PDI クライアント（Spoon）の[データ変換]パースペクティブの[データ変換]画面を使用してデータ変換を定義する際に、使用できるステップについて説明します。

2.1 ステップの一覧

PDI クライアント（Spoon）で利用できるステップの一覧を次の表に示します。

表 2-1 PDI クライアント（Spoon）で利用できるステップの一覧

カテゴリ	ステップ名	説明
入力	Access 入力 (Microsoft Access input)	Access 形式のファイルからデータを読み込みます。
	CSV 入力 (CSV file input)	CSV 形式のファイルからデータを読み込みます。
	ESRI shapefile reader (ESRI Shapefile Reader)	地図データなどのシェープファイルを読み取ります。
	Excel 入力 (Microsoft Excel input)	Excel 形式のファイルからセルに含まれるデータを読み込みます。
	Get data from XML (Get data from XML)	XML ファイルまたは先行のステップから入力されたフィールド名を使用して、XML データを読み込みます。
	JSON input (JSON Input)	JSON 形式のファイルからデータを読み込みます。先行のステップで定義された JSON ファイルを取得することもできます。
	キューブ入力 (De-serialize from file)	cube 拡張子の入力ファイルからデータを読み取ります。
	システムデータ取得 (Get system info)	システムからデータ（システム情報）を取得し、新たなフィールドに出力します。
	テキストファイル入力 (Text file input)	CSV 形式、またはフィールド分割単位が固定幅のファイルのデータを読み込みます。
	テーブル入力 (Table input)	データベースのテーブルから SQL 文 (SELECT 文) を使用してデータを読み込みます。
	データグリッド (Data grid)	ユーザーが定義したテーブル形式の静的データを生成します。
	ファイルメモリロード (Load file content in memory)	入力ファイルからファイルの内容とファイルサイズを読み込みます。
	ファイル名取得 (Get file names)	ファイルシステムからファイルまたはディレクトリを検索し、ファイルパスやファイル名などのメタ情報を取得します。
	固定幅ファイル入力 (Fixed File input)	固定幅データを含むテキストファイルから、値を取得します。
	行生成 (Generate rows)	ユーザーが定義した形式の静的データを複数行生成します。
出力	JSON output (JSON output)	先行ステップからの入力値に基づいて、JSON データのブロックを生成します。
	Microsoft Excel writer (Microsoft Excel writer)	Excel 形式のファイルにデータを書き込みます。

カテゴリ	ステップ名	説明
	Pentaho レポート出力 (Pentaho reporting output)	Pentaho レポートファイル (PRPT) をレンダリングし、選択した形式 (PDF、CSV、Excel など) でファイルを出力します。
	XML output (XML output)	任意のソースから 1 つ以上の XML ファイルに出力します。
	キューブ出力 (Serialize to file)	入力データを cube 拡張子のバイナリ形式で出力します。
	テキストファイル出力 (Text file output)	区切り文字で区切られたテキストファイル (CSV 形式) にデータを書き込みます。
	テーブル出力 (Table output)	データベースのテーブルに追加 (INSERT) し、データを書き込みます。
	削除 (Delete)	データベースのテーブルに対し、入力ストリームおよび出力先のテーブルからキーとなる列を選択し、指定した条件に応じて入力行をテーブルから削除 (DELETE) します。
	挿入/更新 (Insert / update)	データベースのテーブルに対し、入力ストリームおよび出力先のテーブルからキーとなる列を選択し、指定した条件に応じて入力行をテーブルに挿入 (INSERT) または更新 (UPDATE) をします。
	更新 (Update)	データベースのテーブルに対して、入力ストリームおよび出力先のテーブルからキーとなる列を選択し、指定した条件に応じて入力行をテーブルに更新 (UPDATE) します。
変換	Add a checksum (Add a checksum)	入力ストリームのフィールドのチェックサムを計算し、その結果を出力します。
	シーケンス追加 (Add sequence)	一連の整数 (シーケンス) を新たなフィールドに出力します。
	シーケンス追加 (変化値) (Add value fields changing sequence)	一連の整数 (シーケンス) を新たなフィールドに出力します。カウント中にキーとフィールドの値が変化した時にシーケンスカウンタが開始値に戻ります。
	フィールド値設定 (Set field value)	フィールドの値を別のフィールドの値に置き換えます。
	フィールド分割 (Split fields)	指定文字で区切られた 1 フィールドを複数のフィールドに展開して出力します。
	フィールド分割 (行) (Split field to rows)	指定文字で区切られた 1 フィールドを複数の行に展開して出力します。
	フィールド定数設定 (Set field value to a constant)	フィールドの値を、定義した定数に置き換えます。
	フィールド結合 (Concat fields)	指定した複数のフィールドの値を指定した書式で文字列に変換し、区切り文字で区切って文字列結合して出力します。
	値マッピング (Value mapper)	指定したフィールドの特定の値を、ほかの値に置き換えて出力します。
	定数追加 (Add constants)	入力ストリームのすべてのレコードに対して、同一の定数値が設定されたフィールドを追加して出力します。
	数値範囲 (Number range)	指定したフィールドの値の数値範囲に応じた値を、新たなフィールドに出力します。

カテゴリ	ステップ名	説明
	文字列オペレーション (String operations)	入力文字列を実行フィールドの値に従って加工します。
	文字列カット (Strings cut)	始点と終点を指定して、文字列をカットします。
	文字列置換 (Replace in string)	入力文字列から指定文字列を検索し、指定した文字に置換して出力します。
	行フラット化 (Row Flattener)	一定数の連続した入力データ (フィールド) を 1 行に構造化します。
	行整列 (Sort rows)	指定したフィールドの値をソートします。
	行正規化 (Row normaliser)	指定した行の正規化 (ピボット) をします。
	行非正規化 (Row denormaliser)	指定した行の非正規化 (アンピボット) をします。
	計算 (Calculator)	選択された入力フィールドの値を計算式に従って計算し、出力フィールドに出力します。
	選択/名前変更 (Select values)	入力ストリームにあるフィールドの取捨選択、名前変更、メタ情報の設定/変更などの処理を実行し、出力ストリームに出力します。
	重複行削除 (Unique rows)	ソート済みのデータの重複行を削除します。
	重複行削除 (HashSet) (Unique rows (HashSet))	入力データの重複行を削除します。
ユーティリティ	NULL 値定数変換 (If field value is null)	あるフィールドの NULL 値を特定の値に変換します。
	ファイル処理 (Process files)	指定したファイルをコピー、移動、または削除 (Delete) します。
	プロセス実行 (Execute a process)	プロセスを実行し、結果を出力します。すべての入力行に対して実行します。
	メタデータ構造 (Metadata structure of stream)	ストリームのフィールドを読み取り、それらのメタデータを出力します。
	メール (Mail)	SMTP サーバを使用して、前のステップからの電子メールを送信するための入力データを使用して電子メールを送信します。
	ログ書き込み (Write to log)	入力行ごとに、所定のフォーマットのログを出力します。
	定数 NULL 変換 (Null if)	あるフィールドの特定の定数を NULL 値に変換します。
	行複製 (Clone row)	入力行のコピーを作成し、コピーデータを後続のステップに渡します。
フロー	Blocking step (Blocking step)	前ステップからすべての行が入力し終わるまで待機し、前ステップからの入力処理が終了したときに、入力データの最終行を出力します。
	Java フィルター (Java Filter)	入力レコードを、指定された Java の評価式に従って処理し、出力ストリームに出力します。
	ジョブ実行 (Job Executor)	指定したジョブを呼び出し、入力ストリームのデータに応じた設定でジョブを実行します。

カテゴリ	ステップ名	説明
	ステップブロック (Block this step until steps finish)	前ステップからの入力を阻止し、指定したステップの処理がすべて終了したときに、入力データの出力を開始します。
	ストリーム結合 (Append streams)	2つの入力データを連結します。
	ダミー (何もしない) (Dummy (do nothing))	入力ストリームのデータに対して、何も処理をしないで後続のステップに出力します。
	トランスフォーメーション実行 (Transformation executor)	指定したデータ変換を呼び出し、入力ストリームのデータに応じた設定でデータ変換を実行します。
	フィルター (Filter rows)	指定した条件式に従って、入力レコードを異なる出力ストリームに出力します。
	停止 (Abort)	しきい値を超えた入力があった場合、データ変換を停止します。
	優先順位 (Prioritize streams)	後続のステップに出力する順序を定義します。
	条件分岐 (Switch/case) (Switch / case)	キーフィールドの値で、入力レコードの出力先を振り分けます。
	空ストリーム探知 (Detect empty stream)	入力ストリームのデータが0行の場合は、入力ストリームのフィールド定義ですべての値がNULLのレコードを1行出力します。1行以上の場合は、データを出力しません。
スクリプト	Java スクリプト (Modified Java Script value)	データ変換するスクリプトを作成するためのユーザーインターフェースを提供します。画面の JavaScript に設定したコードを、入力ストリームの各レコードに対して実行します。
	Python Executor (Python Executor)	先行のステップから送信された入力データを Python スクリプトの変数として使用し、Python スクリプトを実行します。また、Python スクリプトの変数を後続のステップへ出力します。
	SQL 文実行 (Execute SQL script)	指定した SQL 文を実行します。
	SQL 文実行(行) (Execute row SQL script)	前のステップから入力された SQL 文を実行します。または、前のステップから入力された SQL ファイルを読み込んで SQL 文を実行します。
	ユーザー定義 Java クラス (User defined java expression)	Java ソースコードを入力し、独自のステップ機能を実行します。
	数式 (Formula)	入力フィールドの値を使用して、設定した数式で計算をします。
	正規表現 (Regex evaluation)	入力フィールドの文字列値を、正規表現で定義されたテキストパターンと照合します。
参照	HTTP クライアント (HTTP Client)	指定した URL に HTTP リクエストを送信し、結果を取得します。
	REST クライアント (REST Client)	指定した URL に HTTP リクエストを送信し、結果を取得します。おもに、REST API を実行するために使用します。

カテゴリ	ステップ名	説明
	ストリーム参照 (Stream lookup)	入力ストリームのデータに対して、参照ストリームに入力されたデータと突き合わせ処理を実行し、参照ストリームのデータを取得（検索）します。
	テーブル確認 (Table Exists (Step))	指定されたテーブルが、指定したデータベースに存在するかどうかを確認します。
	データベース参照 (Database lookup)	入力ストリームのデータに対して、参照ストリームに入力されたデータベースのデータと突き合わせ処理を実行し、参照ストリームのデータを取得（検索）します。
	データベース結合 (Database join)	SQL 文中のプレースホルダーを、先行ステップで指定したパラメータの値に置き換えて、SQL 文を実行し、データを取得します。
	ファイル確認 (File exists (Step))	前のステップで指定されたファイルパスまたはフォルダパスが存在するかどうかを確認します。
結合	XML 結合 (XML Join)	ストリームソース要素を、対象ストリーム要素に追加します。
	マージ結合 (Merge join)	入力ストリームの 2 つのテーブルを結合します。
	多方向 Merge Join (Multiway merge join)	複数の入力ストリームテーブルを結合します。
	整列併合 (Sorted merge)	複数の入力ストリームのデータに対し、指定したフィールドの値でソートして出力します。
	行マージ (比較) (Merge rows (diff))	フィールドの値を変更する前と変更したあとのデータを入力して比較し、結果を新たなフィールドに出力します。
	行結合 (デカルト積) (Join rows (cartesian product))	複数ステップから入力されるデータをクロス結合し、条件に合うレコードだけを出力します。
データウェアハウス	ディメンジョン 参照/更新 (Dimension lookup/update)	ディメンジョンテーブルの参照と更新ができます。
データ検証	データ検証 (Data Validator)	フィールド内のデータのルール（値の範囲、値の個別のリスト、またはデータ長など）を定義し、データを検証します。
統計	グループ化 (Group by)	キーでレコードをグループ化して選択した値を集計し、集計結果を出力します。
	サンプル行 (Sample rows)	入力データから指定した行だけを出力します。
	メモリーグループ化 (Memory group by)	キーでレコードをグループ化して選択した値を集計し、出力します。
	分析クエリー (Analytic query)	あるフィールドについて、N 行前、または N 行後のレコードの値を参照して付与します。参照する値が存在しない場合は、NULL 値を付与します。グループキーを設定すると、キー値ごとにテーブルを分割し、分割したテーブルごとに処理を行います。
	単変量統計 (Univariate statistics)	入力フィールドのデータからシンプルな単変量統計を計算できます。
	貯留層サンプリング (Reservoir sampling)	入力データから指定した行をサンプリングします。

カテゴリ	ステップ名	説明
ジョブ	変数取得 (Get Variables)	入力ストリームのすべてのレコードに対して、フィールドを追加して出力します。フィールドに設定される値は、変数で指定できます。
	変数設定 (Set Variables)	1 レコードのフィールドの値を変数に設定します。[変数のスコープ]の設定によって、変数を参照できるジョブの範囲が異なります。
	行取得 (Get rows from result)	行送信ステップまたはトランスフォーメーション実行ステップがメモリに格納したレコードを取得します。
	行送信 (Copy rows to result)	ジョブの次のデータ変換またはジョブエントリに渡すレコードを、メモリに格納します。
マッピング	シンプルマッピング (サブデータ変換) (Simple Mapping (sub-transformation))	指定したデータ変換を呼び出し、入力ストリームのデータを引き渡して実行して、処理結果を後続のステップへ出力します。シンプルマッピングステップで呼び出されたデータ変換は親データ変換の一部として動作します。 シンプルマッピング (サブデータ変換) ステップでは、入力と出力のステップを1つだけ指定できます。
	マッピング (サブデータ変換) (Mapping (sub-transformation))	指定したデータ変換を呼び出し、入力ストリームのデータを引き渡して実行して、処理結果を後続のステップへ出力します。マッピングステップで呼び出されたデータ変換は親データ変換の一部として動作します。 マッピングステップへの入力および出力は複数ストリームを指定できます。
	マッピング入力 (Mapping input specification)	このステップが配置されたデータ変換が、マッピングステップで呼び出された際の入力となります。
	マッピング出力 (Mapping output specification)	このステップが配置されたデータ変換が、マッピングステップで呼び出された際の出力となります。
バルクロード	PostgreSQL バルクロード (PostgreSQL bulk loader)	PostgreSQL データベースのテーブルにデータを書き込みます。
新機能	SFTP put (SFTP put)	ローカルのファイルシステムから FTP サーバへ、SFTP を使用してファイルを転送します。

この表に記載されていないステップについては、Pentaho Documentation を参照してください。

2.1.1 ステップの説明について

変数について

変数は「\${変数}」と記述します。テーブル名や列名にも変数を使用できます。Ctrl キーとスペースキーを同時に押すと、使用できる変数の一覧が表示されます。

2.1.2 データタイプの対応表

PDI データタイプと Java データタイプの対応表を次に示します。

表 2-2 データタイプの対応表

PDI データタイプ	Java データタイプ	PDI データタイプの説明
BigNumber	BigDecimal	任意精度の数値です。
Binary	Byte[]	バイトの配列です。
Boolean	Boolean	true または false の真偽値です。
Date	Date	ミリ秒精度の日時値です。
Integer	Long	符号付 64 ビット整数値です。
Internet Address	InetAddress	インターネットプロトコルアドレス値です。
Number	Double	倍精度浮動小数点値です。
String	String	可変長の無制限のテキストです。
Timestamp	Timestamp	ナノ秒精度の日時値です。

2.2 入力

- [入力]Access 入力 (Microsoft Access input)
- [入力]CSV 入力 (CSV file input)
- [入力]ESRI shapefile reader (ESRI Shapefile Reader)
- [入力]Excel 入力 (Microsoft Excel input)
- [入力]Get data from XML (Get data from XML)
- [入力]JSON input (JSON Input)
- [入力]キューブ入力 (De-serialize from file)
- [入力]システムデータ取得 (Get system info)
- [入力]テキストファイル入力 (Text file input)
- [入力]テーブル入力 (Table input)
- [入力]データグリッド (Data grid)
- [入力]ファイルメモリロード (Load file content in memory)
- [入力]ファイル名取得 (Get file names)
- [入力]固定幅ファイル入力 (Fixed File input)
- [入力]行生成 (Generate rows)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.2.1 [入力]Access 入力 (Microsoft Access input)

Access 形式のファイルからデータを読み込みます。

(1) [ファイル]タブ

入力ファイルに関する設定をします。

[ファイル名の参照]ボタンを押すと、入力ファイルリストの一覧が表示されます。

図 2-1 [Access 入力]画面-[ファイル]タブ

ステップ名: Access入力

ファイル 全般 ファイルド 追加出力フィールド

ファイル設定

先行のステップから値を引き継ぐ ☐

フィールド名: []

ファイルとディレクトリ: [] [追加(A)] [参照(B)...]

検索文字列: []

除外正規表現: []

ファイル名のリスト

	ファイルとディレクトリ	検索文字列	除外検索	必須	サブフォルダーを含む
1					

[ファイル名の参照(S)]

[ヘルプ] [OK(O)] [プレビュー(V)] [キャンセル(C)]

表 2-3 [Access 入力]画面の項目-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値をファイルパスとして使用します。 フィールドの値として指定するファイルパスには、正規表現は使用できません。 チェックを付けると、[ファイル名のリスト]を使用したデータの読み込みはできません。
3	[フィールド名]	ファイルおよびディレクトリが含まれているフィールドを指定します。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
4	[ファイルとディレクトリ]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、入力ファイルのパス、または入力ファイルが格納されているディレクトリのパスを指定します。 ディレクトリを指定した場合、[検索文字列]が空欄のときにエラーが発生します。 [追加]ボタンを押すと、入力ファイルリストに登録されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
5	[検索文字列]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、指定したファイルとディレクトリ内を入力フィールドで指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを入力ファイルリストに追加します。 正規表現を使用した検索もできます。
6	[除外正規表現]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、指定した文字列で検索し、検索文字列と完全一致するファイルパスを入力ファイルリストから除外します。 正規表現を使用した検索もできます。
7	[ファイル名のリスト]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、次の検索対象のリストを指定します。 • [ファイルとディレクトリ] 指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。 • [検索文字列]

項番	項目名	説明
		[検索文字列]に指定した文字列を表示します。 • [除外検索] [除外正規表現]に指定した文字列を表示します。 • [必須] 該当の入力ファイルが入力必須かどうかを[Y]または[N]で指定します。 [Y]を指定すると、入力ファイルに該当するファイルが1つも見つからない場合に、エラーが発生します。 • [サブフォルダを含む] ファイルの検索対象として、該当のディレクトリのサブフォルダを検索するかどうかを[Y]または[N]で指定します。 [Y]を指定すると、サブフォルダも検索します。
8	[削除]ボタン	このボタンを押すと、[ファイル名のリスト]で選択した項目を削除します。
9	[編集]ボタン	このボタンを押すと、[ファイル名のリスト]で選択した項目を編集します。

(2) [全般]タブ

読み込んだデータの出力に関する設定をします。

図 2-2 [Access 入力]画面-[全般]タブ

表 2-4 [Access 入力]画面の項目-[全般]タブの項目

項番	項目名	説明
1	[テーブル名]	取得したファイルに存在するテーブル名を指定します。 [参照]ボタンを押すと、データベース内のテーブル名を参照できます。
2	[出力にファイル名を含む]	この項目にチェックを付けると、出力結果にファイル名を含む文字列フィールドを追加します。 [ファイル]タブの[先行のステップから値を引き継ぐ]にチェックを付けている場合、この項目は非活性になります。

項番	項目名	説明
3	[フィールド名]	ファイル名を含む文字列フィールドの名前を指定します。 [出力にファイル名を含む]にチェックを付けると指定できます。
4	[出力にテーブル名を含む]	この項目にチェックを付けると、出力結果にテーブル名を含む文字列フィールドを追加します。
5	[フィールド名]	テーブル名を含む文字列フィールドの名前を指定します。 [出力にテーブル名を含む]にチェックを付けると指定できます。
6	[出力にレコード番号を含む]	この項目にチェックを付けると、出力結果にレコード番号を含む文字列フィールドを追加します。
7	[フィールド名]	レコード番号を含む文字列フィールドの名前を指定します。 [出力にレコード番号を含む]にチェックを付けると指定できます。
8	[ファイル毎にレコード番号をリセットする]	この項目にチェックを付けると、レコード番号が、ファイルごとに 1 にリセットされます。 [出力にレコード番号を含む]にチェックを付けると指定できます。 【注意事項】 この項目にチェックを付けると、リセット時のレコード番号が 0 から採番されます。
9	[最大レコード数]	読み込むデータのレコード数の最大を指定します。SQL 文の LIMIT 句と同様の処理です。 0 を指定すると、すべてのレコードを読み込みます。
10	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。

(3) [フィールド]タブ

読み込むデータのフィールド情報を定義します。

[フィールドを取得]ボタンを押すと、入力ファイルからフィールド定義を自動推定できます。

図 2-3 [Access 入力]画面-[フィールド]タブ

表 2-5 [Access 入力]画面の項目-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。
2	[列名]	列名を指定します。
3	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[書式]	入力書式を指定します。
5	[長さ]	データの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
6	[精度]	浮動小数点の桁数を指定します。 データタイプが Number と BigNumber 場合に指定できます。
7	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
8	[桁区切り文字]	「. (ドット)」など、小数点を表す文字を指定します。
9	[数値囲み文字]	「, (コンマ)」など、桁を区切る文字を指定します。
10	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [無し] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。

項番	項目名	説明
		• [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
11	[データを代替する]	フィールドの値が NULL 値の場合に、NULL 値を同じフィールドの 1 つ前のレコードの値に変換するかどうかを指定します。 • [Y] NULL 値を同じフィールドの 1 つ前のレコードの値に変換します。1 つ前のレコードの値がない場合は、NULL 値のまま出力します。 • [N] NULL 値の変換をしません。

(4) [追加出力フィールド]タブ

フィールドに追加の設定をします。

図 2-4 [Access 入力]画面-[追加出力フィールド]タブ



表 2-6 [Access 入力]画面の項目-[追加出力フィールド]タブの項目

項番	項目名	説明
1	[短縮ファイル名フィールド]	入力ファイルのファイル名を出力するためのフィールドを指定します。
2	[拡張子フィールド]	入力ファイルの拡張子を出力するためのフィールドを指定します。
3	[パスフィールド]	入力ファイルが格納されているディレクトリの絶対パスを出力するためのフィールドを指定します。
4	[サイズフィールド]	入力ファイルのファイルサイズ（単位：バイト）を出力するためのフィールドを指定します。
5	[隠しフィールド]	入力ファイルが隠しファイルかどうかを出力するためのフィールドを指定します。
6	[最終更新フィールド]	入力ファイルの最終更新日を出力するためのフィールドを指定します。
7	[Uri フィールド]	入力ファイルの URI を出力するためのフィールドを指定します。

項番	項目名	説明
8	[ルート Uri フィールド]	入力ファイルのルートディレクトリの URI を出力するためのフィールドを指定します。

2.2.2 [入力]CSV 入力 (CSV file input)

CSV 形式のファイルからデータを読み込みます。

[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。他のステップから CSV 入力ステップにデータを入力した場合、[ファイル名]が[フィールド名]に置き換わり、入力フィールドの値をファイルパスとして利用できます。[フィールドを取得]ボタンを押すと、入力ファイルのヘッダー情報からフィールド定義を自動推定できます。

図 2-5 [CSV 入力]画面 (他のステップからデータを入力していない場合)

The screenshot shows the 'CSV Input' dialog box. The 'Step Name' is 'CSV入力'. The 'File Name' field is empty, with a '参照(B)...' button next to it. The 'Field Separator' is set to comma (`,`), with a 'タブの挿入(T)' button. The 'Quote Character' is set to double quote (`"`). The 'Buffer Size' is 50000. There are checkboxes for 'データ型を自動変換' (checked), 'ヘッダー・レコードを含む' (checked), and 'ファイル名を結果に含む' (unchecked). The 'Field Name' field is empty. There are checkboxes for '分散処理' (unchecked) and 'フィールド内で新たな行を許可' (unchecked). The 'Format' is set to 'mixed'. The 'Text Code' field is empty. Below these fields is a table with 10 columns: #, フィールド名, データ・タイプ, 書式, 長さ, 精度, 通貨記号, 桁区切り文字, 数値囲み文字, and 空白除去. The table has one row with the number '1' in the first column. At the bottom are buttons: ヘルプ, OK(O), フィールドを取得(G), プレビュー(V), and キャンセル(C).

図 2-6 [CSV 入力]画面（他のステップからデータを入力した場合）

表 2-7 [CSV 入力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ファイル名]/[フィールド名]	他のステップから CSV 入力ステップにデータを入力していない場合 項目名は[ファイル名]で、入力ファイルのパスを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 他のステップから CSV 入力ステップにデータを入力している場合 項目名は[フィールド名]で、入力フィールドの値を指定します。指定した入力フィールドの値をファイルパスとして利用できます。
3	[フィールド区切り文字]	データを読み込む CSV ファイルの区切り文字を指定します。 [タブの挿入]ボタンを押すと、タブ文字を入力できます。
4	[引用符]	データを読み込む CSV ファイルの引用符を指定します。
5	[バッファ・サイズ]	IO バッファのサイズ（単位：バイト）を指定します。 バッファサイズに「1」を指定して実行した場合、エラーが発生します。
6	[データ型を自動変換]	この項目にチェックを付けると、読み込んだデータを Java のデータ型に変換しないで、バイトデータのまま後続ステップへ出力します。この項目は項目名と異なり、データ型を自動変換しない機能です。 【注意事項】 入力の処理性能は向上しますが、後続処理のエラーの原因となるおそれがあるため、チェックを付けないでください。
7	[ヘッダ・レコードを含む]	この項目にチェックを付けると、入力ファイルの先頭行だけをヘッダレコードとして読み飛ばします。

項番	項目名	説明
8	[ファイル名を結果に含む]	この項目にチェックを付けると、ファイル取得ステップを使用して、入力ファイルのパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。 CSV 入力ステップを処理したあとに、ファイル取得ステップを処理することで正しい結果が得られるため、ジョブなどで処理順序を制御する必要があります。 取得できるデータは、次のとおりです。 • type • filename 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [ファイル名を結果に含む]を[Y]にしたステップ名 • comment • timestamp
9	[フィールド名]	行番号を出力する場合に、フィールド名を指定します。空白の場合、行番号は出力されません。
10	[分散処理]	この項目にチェックを付けると、CSV 入力ステップをコピーしてデータ読み込みを分散処理できます。 【注意事項】 ファイルの内容を正しく読み込めないことがあるため、チェックを付けないでください。
11	[フィールド内で新たな行を許可]	この項目にチェックを付けると、フィールドの値に改行文字がある場合でも、改行文字をフィールドの値として読み込みます。 【注意事項】 最後のフィールドに改行文字が含まれている場合、正しく読み込めません。このため、フィールドの値に改行文字を含む場合はこの機能を使用しないで、引用符を使用してください。
12	[フォーマット]	改行コードを選択します。
13	[文字コード]	入力ファイルの文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。
14	[フィールド名]	フィールド名を指定します。
15	[データ・タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。
16	[書式]	入力書式を指定します。
17	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合

項番	項目名	説明
		数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
18	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
19	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
20	[桁区切り文字]※	「. (ドット)」など、小数点を表す文字を指定します。
21	[数値囲み文字]※	「, (コンマ)」など、桁を区切る文字を指定します。
22	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。

注※ 画面の項目名と指定する内容が異なります。

2.2.3 [入力]ESRI shapefile reader (ESRI Shapefile Reader)

地図データなどのシェープファイルを読み取ります。

図 2-7 [ESRI shapefile reader]画面

表 2-8 [ESRI shapefile reader]画面の項目

項番	項目名	説明
1	[Step name]	ステップ名を指定します。
2	[Name of the shapefile (.shp)]	シェープファイルをフルパスまたは変数で指定します。

項番	項目名	説明
		[Browse]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 この項目と[Name of the DBF file (.dbf)]は、指定が必須の項目です。 この項目と[Name of the DBF file (.dbf)]のどちらかが空の場合、エラーが発生します。
3	[Name of the DBF file (.dbf)]	DBF ファイルををフルパスまたは変数で指定します。 [Browse]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 この項目と[Name of the shapefile (.shp)]は、指定が必須の項目です。 この項目と[Name of the shapefile (.shp)]のどちらかが空の場合、エラーが発生します。
4	[Encoding]	入力ファイルの文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。

2.2.4 [入力]Excel 入力（Microsoft Excel input）

Excel 形式のファイルからセルに含まれるデータを読み込みます。

[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。

(1) [ファイル]タブ

入力ファイルに関する設定をします。

図 2-8 [Excel 入力]画面-[ファイル]タブ

ステップ名 Microsoft Excel input

フィールドの追加

ファイル シート全般 エラー処理 フィールド 追加出力フィールド

スプレッドシートタイプ(エンジン) Excel 97-2003 XLS (.xls)

ファイルとディレクトリ

検索文字列

除外正規表現

Password

ファイル名のリスト

#	ファイルとディレクトリ	検索文字列	検索除外	必須	サブフォルダーを含む
1					

ファイル設定

先行のステップから値を引き継ぐ ☐

ステップ名

フィールド名

ファイル名の参照(S)

ヘルプ OK(O) プレビュー(V) キャンセル(C)

表 2-9 [Excel 入力]画面-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[スプレッドシートタイプ(エンジン)]	Excel ファイルのシートタイプを次の 4 つから選択します。 • Excel 97-2003 XLS (JXL) 拡張子が.xls の Excel ファイルを入力する場合に選択します。 • Excel 2007 XLSX (Apache POI) 拡張子が.xlsx の Excel ファイルを入力する場合に選択します。 • Excel 2007 XLSX (Apache POI Streaming) 拡張子が.xlsx の Excel ファイルを入力する場合に指定します。 「Excel 2007 XLSX (Apache POI)」よりも大きなファイルを読み込みます。 • Open Office ODS (ODFDOM) Open Office ODS 形式のファイルを入力する場合に選択します。
3	[ファイルとディレクトリ]	入力ファイルのパス、または入力ファイルが格納されているディレクトリのパスを指定します。 ディレクトリを指定した場合に、[検索文字列]が空欄のときはエラーが発生します。 [追加]ボタンを押すと、入力ファイルリストに登録されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
4	[検索文字列]	指定した文字列で検索し、検索文字列と完全一致するファイルパスだけを入力ファイルリストに追加します。 正規表現を使用した検索もできます。
5	[除外正規表現]	指定した文字列で検索し、検索文字列と完全一致するファイルパスを入力ファイルリストから除外します。 正規表現を使用した検索もできます。
6	[password]	Excel ファイルにパスワード保護が設定されている場合にパスワードを指定します。
7	[ファイル名のリスト]	• ファイルとディレクトリ 指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。 • 検索文字列 [検索文字列]に指定した文字列を表示します。 • 検索除外 [除外正規表現]に指定した文字列を表示します。 • 必須 該当の入力ファイルが入力必須かどうかを[Y]または[N]で指定します。 [Y]を指定すると、入力ファイルに該当するファイルが 1 つも見つからない場合に、エラーが発生します。 • サブフォルダを含む

項番	項目名	説明
		ファイルの検索対象として、該当のディレクトリのサブフォルダを検索するかどうかを[Y]または[N]で指定します。 [Y]を指定すると、サブフォルダも検索します。
8	[削除]ボタン	このボタンを押すと、選択対象をファイル名のリストから削除します。
9	[編集]ボタン	このボタンを押すと、選択対象のファイルまたはディレクトリのパスを編集します。
10	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値をファイルパスとして使用します。 フィールドの値として指定するファイルパスには、正規表現は使用できません。 チェックを付けると、[ファイル名のリスト]を使用したデータの読み込みはできません。
11	[ステップ名]	先行のステップ名を指定します。
12	[フィールド名]	値がファイルパスとなっているフィールド名を指定します。
13	[ファイル名の参照]	Excel 入力ステップに正常に接続されたソースのファイル名を表示します。

(2) [シート]タブ

読み込む Excel ファイルのシートの情報を設定します。

図 2-9 [Excel 入力]画面-[シート]タブ

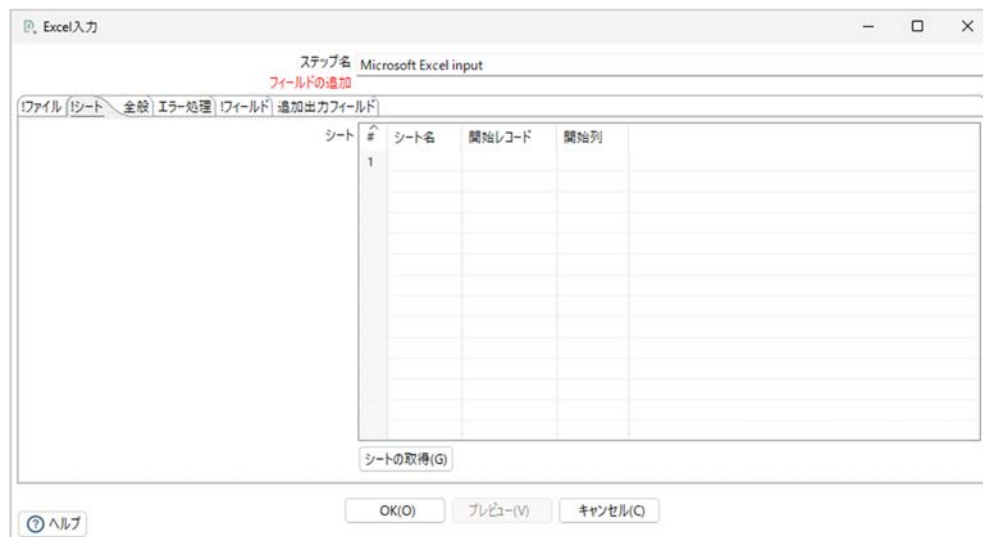


表 2-10 [Excel 入力]画面-[シート]タブの項目

項番	項目名	説明
1	[シート名]	入力したいデータが存在するシートの名称を指定します。
2	[開始レコード]	入力データの開始レコードを指定します。

項番	項目名	説明
		1 行目から入力したい場合は、0 を指定します。
3	[開始列]	入力データの開始列を指定します。 1 列目から入力したい場合は、0 を指定します。
4	[シートの取得] ボタン	このボタンを押すと、[ファイル] タブで指定された入力ファイルから、すべてのシートを取得します。

(3) [全般] タブ

読み込む Excel ファイルのフォーマットを設定します。

図 2-10 [Excel 入力]画面-[全般]タブ



表 2-11 [Excel 入力]画面-[全般]タブの項目

項番	項目名	説明
1	[ヘッダーを含む]	この項目にチェックを付けると、入力ファイルの先頭行だけをヘッダーレコードとして読み飛ばします。
2	[空のレコードを削除]	この項目にチェックを付けると、レコード内の値がすべて NULL 値のレコードを読み飛ばします。
3	[空白のレコードで処理を停止する]	この項目にチェックを付けると、レコード内の値がすべて NULL 値のレコードが出現した時点で、入力ファイルからのデータの読み込みを停止します。
4	[最大レコード数]	読み込むデータのレコード数の最大を指定します。SQL 文の LIMIT 句と同様の処理です。 0 を指定すると、すべてのレコードを読み込みます。
5	[文字コード]	入力ファイルの文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。
6	[ファイル名を結果に含む]	この項目にチェックを付けると、データ入力に使用したファイルパスをメモリに格納します。

項番	項目名	説明
		ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type • filename 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [ファイル名を結果に含む]を[Y]にしたステップ名 • comment • timestamp

(4) [エラー処理]タブ

エラーが発生した場合にステップがどのように反応するかを設定します。

図 2-11 [Excel 入力]画面-[エラー処理]タブ



表 2-12 [Excel 入力]画面-[エラー処理]タブの項目

項番	項目名	説明
1	[形式の違うファイルはエラーとする]	この項目にチェックを付けると、入力ファイルの形式が異なる場合に、読み込み中にデータタイプエラーを出力します。次の文字列のあとにエラー内容が記載されたメッセージが出力されます。 Invalid type Label:

項番	項目名	説明
2	[エラーを無視する]	この項目にチェックを付けると、あるレコードを読み込む際にエラーが発生した場合でも処理を停止しません。 エラーが発生したフィールドの値は NULL 値になります。
3	[エラーのレコードをスキップ]	この項目にチェックを付けると、あるレコードを読み込むときに、エラーが発生すると、該当レコードの読み込みをスキップします。
4	[警告を保管するディレクトリ]	—
5	[エラーを保管するディレクトリ]	指定したファイルが存在しない場合やファイルエラーが発生した場合に、指定したディレクトリにエラーメッセージを出力します。 エラーメッセージを出力するファイルの拡張子を[拡張子]で指定します。 [変数]ボタンを押すと、[環境変数の選択]画面が表示されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
6	[除外したレコードを保管するディレクトリ]	エラーレコードが発生した場合に、そのエラーレコード番号を、指定したディレクトリに出力します。 エラーレコードを出力するファイルの拡張子を[拡張子]で指定します。 [変数]ボタンを押すと、[環境変数の選択]画面が表示されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。

(凡例) —：説明はありません。

(5) [フィールド]タブ

読み込むデータのフィールド情報を定義します。

[フィールドの取得]ボタンを押すと、入力ファイルからフィールド定義を自動推定できます。

図 2-12 [Excel 入力]画面-[フィールド]タブ



表 2-13 [Excel 入力]画面-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。
2	[データタイプ]	データタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
3	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
4	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
5	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
6	[データを代替する]	フィールドの値が NULL 値の場合に、NULL 値を同じフィールドの 1 つ前のレコードの値に変換するかどうかを指定します。 • [Y] NULL 値を同じフィールドの 1 つ前のレコードの値に変換します。1 つ前のレコードの値がない場合は、NULL 値のまま出力します。 • [N] NULL 値の変換をしません。
7	[書式]	入力書式を指定します。
8	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
9	[桁区切り文字]*	「. (ドット)」など、小数点を表す文字を指定します。
10	[数値囲み文字]*	「, (コンマ)」など、桁を区切る文字を指定します。

注※ 画面の項目名と指定する内容が異なります。

(6) [追加出力フィールド]タブ

フィールドに追加の設定をします。

フィールドを追加設定すると、設定したフィールドに入力ファイルの情報を出力できます。

図 2-13 [Excel 入力]画面-[追加出力フィールド]タブ



表 2-14 [Excel 入力]画面-[追加出力フィールド]タブの項目

項番	項目名	説明
1	ファイル名フィールド	入力ファイルの絶対パスを出力するためのフィールドを指定します。
2	シート名フィールド	入力データを読み込んだシート名を出力するためのフィールドを指定します。
3	シートレコード番号フィールド	入力データを読み込んだシート内でのレコード番号を出力するためのフィールドを指定します。
4	レコード番号フィールド	レコード番号を出力するためのフィールドを指定します。
5	[短縮ファイル名フィールド]	入力ファイルのファイル名を出力するためのフィールドを指定します。
6	[拡張子フィールド]	入力ファイルの拡張子を出力するためのフィールドを指定します。
7	[パスフィールド]	入力ファイルが格納されているディレクトリの絶対パスを出力するためのフィールドを指定します。
8	[サイズフィールド]	入力ファイルのファイルサイズ（単位：バイト）を出力するためのフィールドを指定します。
9	[隠しフィールド]	入力ファイルが隠しファイルかどうかを出力するためのフィールドを指定します。
10	[最終更新フィールド]	入力ファイルの最終更新日を出力するためのフィールドを指定します。
11	[Uri フィールド]	入力ファイルの URI を出力するためのフィールドを指定します。
12	[ルート Uri フィールド]	入力ファイルのルートディレクトリの URI を出力するためのフィールドを指定します。

2.2.5 [入力]Get data from XML (Get data from XML)

XML ファイルまたは先行のステップから入力されたフィールド名を使用して、XML データを読み込みます。

先行のステップから入力されたフィールド名を使用して XML データを読み込む方法を次に示します。

- フィールド名のデータを XML データとして読み込む
- フィールド名のファイルパスから XML データを読み込む
- フィールド名の URL から XML データを読み込む

(1) [ファイル]タブ

次の方法で XML データを読み込む場合の設定をします。

- 先行のステップから入力されたフィールド名のデータを XML データとして読み込む方法
- 先行のステップから入力されたフィールド名のファイルパスから XML データを読み込む方法
- 先行のステップから入力されたフィールド名の URL から XML データを読み込む方法

図 2-14 [XML データ取得]画面-[ファイル]タブ

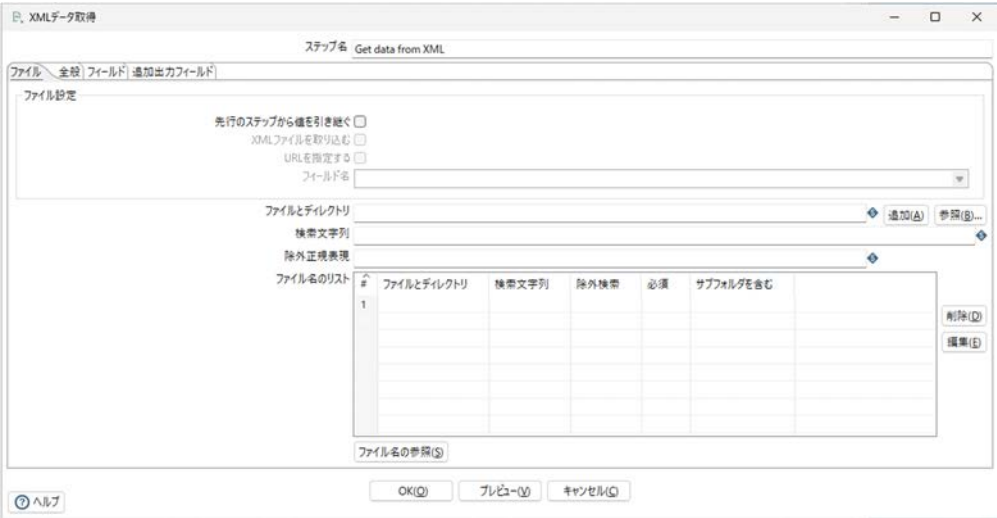


表 2-15 [XML データ取得]画面の項目-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドから XML を読み込みます。 [XML ファイルを取り込む]と[URL を指定する]のどちらも指定していない場合は、[フィールド名]に指定したデータを XML データとして読み込みます。
3	[XML ファイルを取り込む]	この項目にチェックを付けると、[フィールド名]に指定したファイルパスから XML を読み込みます。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。

項番	項目名	説明
4	[URL を指定する]	この項目にチェックを付けると、[フィールド名]に指定した URL から XML を読み込みます。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
5	[フィールド名]	指定した入力フィールドのデータを読み込みます。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
6	[ファイルとディレクトリ]	入力ファイルのパス、または入力ファイルが格納されているディレクトリを指定します。
7	[検索文字列]	[ファイルとディレクトリ]で指定したファイルとディレクトリ内を、指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを出力対象に追加します。 正規表現で入力する必要があります。
8	[除外正規表現]	[ファイルとディレクトリ]で指定したファイルとディレクトリ内を、指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを出力対象から除外します。 正規表現で入力する必要があります。
9	[ファイル名のリスト]	検索対象のリストを表示します。 • ファイルとディレクトリ 指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。 • 検索文字列 [検索文字列]に指定した文字列を表示します。 • 除外検索 [除外正規表現]に指定した文字列を表示します。 • 必須 該当の入力ファイルが入力必須かどうかを[Y]または[N]で指定します。 [Y]を指定すると、入力ファイルに該当するファイルが1つも見つからない場合に、エラーが発生します。 • サブフォルダを含む 検索対象として、該当のディレクトリのサブフォルダを検索するかどうかを[Y]または[N]で指定します。 [Y]を指定すると、サブフォルダも検索します。

(2) [全般]タブ

読み込むファイルのフォーマットなどを設定します。

図 2-15 [XML データ取得]画面-[全般]タブ

表 2-16 [XML データ取得]画面の項目-[全般]タブの項目

項番	項目名	説明
1	[ループ XPath]	行の単位として繰り返し読み込みたいタグを XPath 形式で指定します。 [XPath を取得]ボタンを押すと、XML ドキュメント内で繰り返される可能性のある XPath を検索できます。
2	[文字コード]	文字コードを指定します。
3	[ドキュメントの名前空間を使用する]	この項目にチェックを付けると、XML 内でデフォルト以外の名前空間を利用している場合、名前空間を区別してデータを読み込みます。
4	[コメントを無視する]	—
5	[XML を検証する]	この項目にチェックを付けると、解析する前に XML を検証します。
6	[トークンを使用する]	この項目にチェックを付けると、[フィールド]タブで指定する各フィールドのデータを使って XPath を動的に置き換えます。 例えば、[フィールド]タブの[フィールド名]と[XPath]に次のような値を指定して使用できます。 - フィールド名：User XPath：User - フィールド名：Data XPath：/Session/SessionData[@User=@_User-]/Data
7	[空のファイルは無視する]	この項目にチェックを付けると、空のファイルは無視します。
8	[ファイルの有無は無視する]	この項目にチェックを付けると、[ファイル]タブの[ファイルとディレクトリ]で指定したファイルが 1 つもない場合でも、エラーが発生しません。
9	[エラーを許容する回数]※	入力するタグ数の上限を指定します。0 を指定した場合、すべてのタグを読み込みます。
10	[大容量データを処理する XPath]	[ループ XPath]と同じ値を入力すると、指定した XPath 単位に区切られたデータのブロックごとに処理します。

項番	項目名	説明
		この項目を設定すると、大容量のデータを読み込む際に、メモリ使用量を削減できる場合があります。
11	[ファイル名を出力に含む]	この項目にチェックを付けると、指定したフィールドにファイル名を含めて出力します。 フィールド名 指定したフィールドに、ファイル名(String)を含めます。
12	[レコード番号を出力に含む]	この項目にチェックを付けると、指定したフィールドにレコード番号を含めて出力します。 フィールド名 指定したフィールドに、レコード番号(Integer)を含めます。
13	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 チェックを付けない場合、メモリ使用量を削減できます。

(凡例) - : 説明はありません。

注※ 画面の項目名と指定する内容が異なります。

(3) [フィールド]タブ

先行のステップから入力されたフィールド名のデータを XML データとして読み込む場合に、読み込む XML データのフィールド情報を定義します。

[フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-16 [XML データ取得]画面-[フィールド]タブ

The screenshot shows a software window titled 'XMLデータ取得' (XML Data Acquisition) with a sub-tab 'フィールド' (Fields). The main area is a table with the following headers: #, フィールド名 (Field Name), XPath, 要素タイプ (Element Type), 結果タイプ (Result Type), データタイプ (Data Type), 書式 (Format), 長さ (Length), 精度 (Precision), 逃避記号 (Escape Character), 桁区切り文字 (Thousands Separator), 数値囲み文字 (Numeric Text), 空白除去 (Blank Removal), and データを代替する (Replace Data). The table currently contains one row with the number '1' in the first column. At the bottom of the window, there are buttons for 'ヘルプ' (Help), 'OK', 'プレビュー' (Preview), and 'キャンセル' (Cancel). A 'フィールドの取得' (Get Fields) button is located just above the OK button.

表 2-17 [XML データ取得]画面の項目-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。
2	[XPath]	読み取る要素ノードまたは属性のパスを指定します。
3	[要素タイプ]	読み取る要素のタイプを次の項目から指定します。 • ノード • 属性
4	[結果タイプ]	結果タイプを次の項目から指定します。 • 値 指定したノードの値だけをフィールドに出力します。 • シングルノード 指定したノード全体をフィールドに出力します。
5	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
6	[書式]	入力書式を指定します。
7	[長さ]	データの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
8	[精度]	浮動小数点の桁数を指定します。 データタイプが Number と BigNumber の場合に指定できます。
9	[通貨記号]	「\$（ドル）」など、通貨を表す記号を指定します。
10	[桁区切り文字]	「.（ドット）」など、小数点を表す文字を指定します。
11	[数値囲み文字]	「,（コンマ）」など、桁を区切る文字を指定します。
12	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [none] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [left] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [right] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [both] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
13	[データを代替する]	フィールドの値が空文字の場合の出力方法を指定します。

項番	項目名	説明
		[Y]を指定すると、フィールドの値が空文字の場合、1つ前のレコードの値に変換します。ただし、[Y]を指定しても1つ前のレコードの値がない場合は、空文字のまま出力します。

(4) [追加出力フィールド]タブ

フィールドに追加の設定をします。

フィールドを追加設定すると、設定したフィールドに入力ファイルの情報を出力できます。

図 2-17 [XML データ取得]画面-[追加出力フィールド]タブ

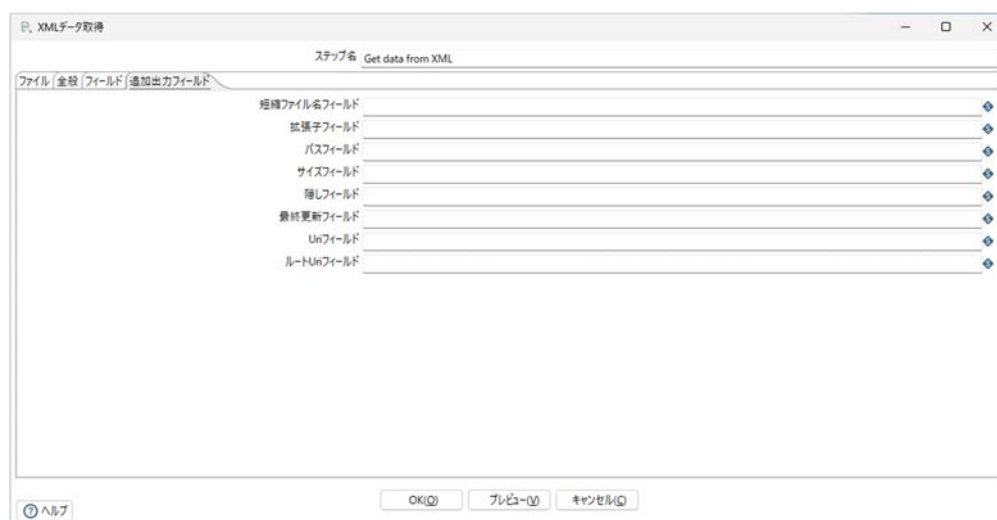


表 2-18 [XML データ取得]画面の項目-[追加出力フィールド]タブの項目

項番	項目名	説明
1	[短縮ファイル名フィールド]	入力ファイルのファイル名を出力するフィールド名を指定します。
2	[拡張子フィールド]	入力ファイルの拡張子を出力するフィールド名を指定します。
3	[パスフィールド]	入力ファイルが格納されているディレクトリの絶対パスを出力するフィールド名を指定します。
4	[サイズフィールド]	入力ファイルのファイルサイズ（バイト）を出力するフィールド名を指定します。 【注意事項】 この項目に指定したフィールド名は、一度リポジトリに保存して再度[データ変換]パースペクティブ画面を表示すると、空白文字に置き換わる可能性があります。
5	[隠しフィールド]	入力ファイルが隠しファイルかどうかを出力するフィールド名を指定します。
6	[最終更新フィールド]	入力ファイルの最終更新日を出力するフィールド名を指定します。 【注意事項】 リモート実行時、次に示す JVM 環境を除いて、対象ファイルのミリ秒単位以下の最終更新時間を取得できません。

項番	項目名	説明
		Open Java Development Kit 8u302(x64)以降の JVM
7	[Uri フィールド]	入力ファイルの URI を出力するフィールド名を指定します。
8	[ルート Uri フィールド]	入力ファイルのルートディレクトリの URI を出力するフィールド名を指定します。

2.2.6 [入力]JSON input (JSON Input)

JSON 形式のファイルからデータを読み込みます。先行のステップで定義された JSON ファイルを取得することもできます。

[プレビュー行]ボタンを押すと、読み込むデータのプレビューができます。

(1) [ファイル]タブ

入力ファイルに関する設定をします。[ファイル名表示]ボタンを押すと、入力ファイルリストの一覧が表示されます。

図 2-18 [JSON 入力]画面-[ファイル]タブ

表 2-19 [JSON 入力]画面-[ファイル]タブの項目

項番	項目名	説明
1	[フィールドにソースを定義]	この項目にチェックを付けると、先行のステップで定義したフィールドからソースを取得します。
2	[フィールドからソースを取得]	先行のステップで定義したフィールド名を指定します。 [フィールドにソースを定義]にチェックを付けると指定できます。
3	[ソースがファイル名]	ソースがファイル名の場合に、この項目にチェックを付けます。 [フィールドにソースを定義]にチェックを付けると指定できます。

項番	項目名	説明
4	[URL としてソースを読み込む]	フィールド名に指定した URL からソースを読み込む場合に、この項目にチェックを付けます。 [フィールドにソースを定義]にチェックを付けると指定できます。 補足：項目名と動作が不一致ですが、「URL としてソースを読み込む」と読み替えてお使いください。 【注意事項】 この項目にチェックを付けていなく、かつ、先行のステップに URL 形式のパスが設定されていても、エラーは発生しません。
5	[Do not pass field downstream]	JSON ファイルのパスを含むフィールドを出力しない場合に、この項目にチェックを付けます。 [フィールドにソースを定義]にチェックを付けると指定できます。
6	[ファイルもしくはディレクトリ]	[フィールドにソースを定義]にチェックを付けない場合、入力ファイルのパス、または入力ファイルが格納されているディレクトリのパスを指定します。 [追加]ボタンを押すと、指定したパスを選択ファイル欄のリストに登録します。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
7	[正規表現]	[フィールドにソースを定義]にチェックを付けない場合、指定したディレクトリ内のファイル名に一致する正規表現を指定します。
8	[除外正規表現]	[フィールドにソースを定義]にチェックを付けない場合、指定したディレクトリ内のファイル名を除外する正規表現を指定します。
9	[ファイル/ディレクトリ]	[フィールドにソースを定義]にチェックを付けない場合、指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。
10	[検索 (正規表現)]	[フィールドにソースを定義]にチェックを付けない場合、[検索 (正規表現)]に指定した文字列を表示します。
11	[除外検索]	[フィールドにソースを定義]にチェックを付けない場合、[除外検索]に指定した文字列を表示します。
12	[必須]	[フィールドにソースを定義]にチェックを付けない場合、該当の入力ファイルが入力必須かどうかを[Y]または[N]で指定します。 [Y]を指定すると、入力ファイルに該当するファイルが 1 つも見つからない場合に、エラーが発生します。
13	[サブフォルダーを含む]	[フィールドにソースを定義]にチェックを付けない場合、ファイルの検索対象として該当するディレクトリのサブフォルダを検索するかどうかを[Y]または[N]で指定します。 [Y]を指定すると、サブフォルダも検索します。
14	[削除]ボタン	このボタンを押すと、選択対象を[選択ファイル]リストから削除します。
15	[編集]ボタン	このボタンを押すと、[選択ファイル]リストで選択したファイルまたはディレクトリのパスを編集します。

(2) [内容]タブ

取得するデータを構成するためのオプションを設定します。

図 2-19 [JSON 入力]画面-[内容]タブ

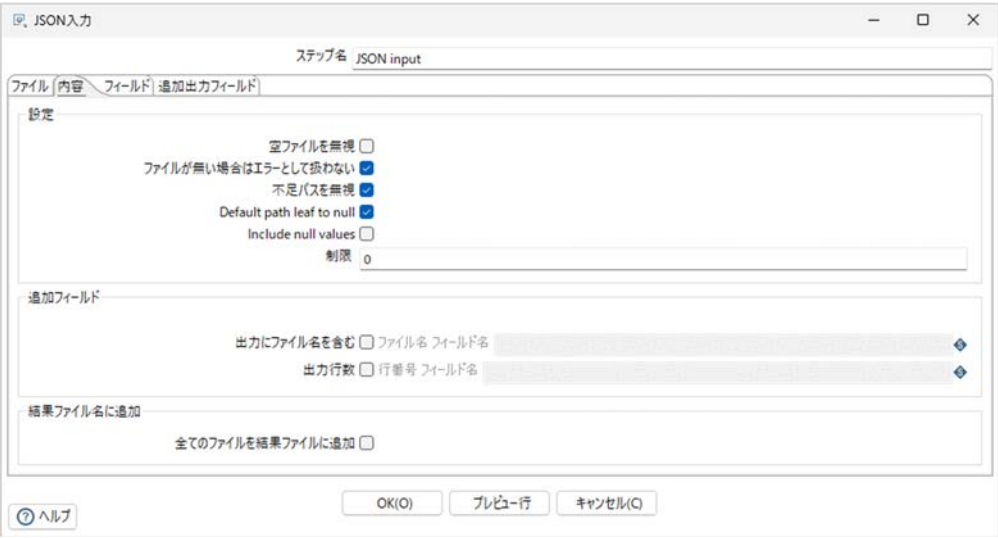


表 2-20 [JSON 入力]画面-[内容]タブの項目

項番	項目名	説明
1	[空ファイルを無視]	この項目にチェックを付けると、ソースファイルの中身が空のファイルが無視します。 チェックを付けない場合、ファイルが空だったときは、エラーが発生します。
2	[ファイルが無い場合はエラーとして扱わない]	この項目にチェックを付けると、ソースで指定したファイルが存在しない場合でも、エラーが発生しません。 チェックを付けない場合、ファイルが存在しなかったときは、エラーが発生します。
3	[不足パスを無視]	この項目にチェックを付けると、次のどちらかの場合にファイルの処理を続行します。 - JSON パスに一致するフィールドがない - すべての値が NULL である チェックを付けない場合、上記のどちらかのときにエラーが発生します。
4	[Default path leaf to null]	この項目にチェックを付けると、レコードごとに欠落している JSON パスに対して NULL 値を返します。 チェックを付けない場合、JSON パスの数がレコードごとに異なるときは、エラーが発生します。
5	[Include null values]	この項目にチェックを付けると、入力データの key-value の値が NULL のデータの場合も 1 行とカウントして出力します。
6	[制限]	入力データのレコード数の制限を指定します。 「0」を指定すると制限されません。 [ファイル]タブの[フィールドにソースを定義]を[Y]に設定している場合、この項目は非活性になります。
7	[出力にファイル名を含む]	この項目にチェックを付けると、出力結果にファイル名を含む文字列フィールドを追加します。

項番	項目名	説明
		[ファイル]タブの[フィールドにソースを定義]にチェックを付けている場合、この項目は非活性になります。
8	[ファイル名 フィールド名]	ファイル名を含む文字列フィールドの名前を指定します。 [出力にファイル名を含む]にチェックを付けると指定できます。
9	[出力行数]	この項目にチェックを付けると、出力結果に行番号を含む整数フィールドを追加します。
10	[行番号 フィールド名]	行番号を含む整数フィールドの名前を指定します。 [出力行数]にチェックを付けると指定できます。
11	[全てのファイルを結果ファイルに追加]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [全てのファイルを結果ファイルに追加]を[Y]にしたステップ名 • comment • timestamp [ファイル]タブの[フィールドにソースを定義]にチェックを付けている場合、この項目は非活性になります。

(3) [フィールド]タブ

読み込むデータのフィールド情報を設定します。

[Select fields]ボタンを押すと、[Select fields]ウィンドウが開きます。[Select fields]ウィンドウでは、ソースファイルの各フィールドのチェックボックスで、出力に含めたいフィールドを選択できます。また、検索ボックスにフィールド名を入力すると、フィールド名を検索できます。

図 2-20 [JSON 入力]画面-[フィールド]タブ

表 2-21 [JSON 入力]画面-[フィールド]タブの項目

項番	項目名	説明
1	[名称]	フィールド名を指定します。
2	[パス]	入力ファイルのフィールド名のフルパスを指定します。パスにアスタリスク（*）を追加すると、すべてのレコードを取得できます。 例：\$.mydata.*
3	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。 【注意事項】 BigNumber を指定しないでください。BigNumber を指定して実行すると、エラーが発生します。
4	[形式]	入力書式を指定します。
5	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
6	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
7	[通貨]	「\$（ドル）」など、通貨を表す記号を指定します。
8	[数値]	「.（ドット）」など、小数点を表す文字を指定します。
9	[グループ]※	「,（コンマ）」など、桁を区切る文字を指定します。
10	[トリムタイプ]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。

項番	項目名	説明
		• [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
11	[繰り返し]	この機能は動作しません。

注※ 画面の項目名と指定する内容が異なります。

(4) [追加出力フィールド]タブ

処理するファイルに関する追加情報を指定するためのオプションを定義します。

図 2-21 [JSON 入力]画面-[追加出力フィールド]タブ



表 2-22 [JSON 入力]画面-[追加出力フィールド]タブの項目

項番	項目名	説明
1	[短縮ファイル名フィールド]	入力ファイルのファイル名を出力するためのフィールドを指定します。
2	[拡張子フィールド]	入力ファイルの拡張子を出力するためのフィールドを指定します。
3	[パスフィールド]	入力ファイルが格納されているディレクトリの絶対パスを出力するためのフィールドを指定します。
4	[サイズフィールド]	入力ファイルのファイルサイズ（単位：バイト）を出力するためのフィールドを指定します。
5	[隠しフィールド]	入力ファイルが隠しファイルかどうかを出力するためのフィールドを指定します。

項番	項目名	説明
		【注意事項】 入力ファイルの隠しファイル属性は取得できません。
6	[最終更新フィールド]	入力ファイルの最終更新日を出力するためのフィールドを指定します。
7	[Uri フィールド]	入力ファイルの URI を出力するためのフィールドを指定します。
8	[ルート Uri フィールド]	入力ファイルのルートディレクトリの URI を出力するためのフィールドを指定します。 【注意事項】 この項目に値を設定してリモート実行すると、エラーが発生します。

2.2.7 [入力]キューブ入力 (De-serialize from file)

cube 拡張子の入力ファイルからデータを読み取ります。

図 2-22 [キューブ入力]画面

表 2-23 [キューブ入力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ファイル名]	入力ファイルのパスを指定します。 [参照] ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
3	[最大レコード数]	読み込むデータのレコード数の最大を指定します。SQL 文の LIMIT 句と同様の処理です。 0 を指定すると、すべてのレコードを読み込みます。
4	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 【注意事項】 ファイル名の結果のフィールドが[origin]の場合、正しいステップ名が取得できません。

2.2.8 [入力]システムデータ取得 (Get system info)

システムからデータ（システム情報）を取得し、新たなフィールドに出力します。[プレビュー]ボタンを押すと、出力するデータのプレビューができます。

図 2-23 [システムデータ取得]画面

表 2-24 [システムデータ取得]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	システムデータ（システム情報）を出力するフィールド名を指定します。
3	[システムデータ]	システムから取得するデータ（システム情報）の種類を指定します。 システムから取得するデータ（システム情報）の種類については、「 システムから取得するデータの種類 」を参照してください。 【注意事項】 コマンドライン引数の場合、リモート実行では引数を取得できません。

表 2-25 システムから取得するデータの種類

項番	システム情報	説明
1	システム日付 (可変)	現在のシステム時刻を取得します。
2	システム日付 (固定)	データ変換開始時のシステム日時を取得します。
3	開始日付の範囲 (トランスフォーメーション)	データ変換ログテーブルの STARTDATE の値を取得します。
4	終了日付の範囲 (トランスフォーメーション)	データ変換ログテーブルの ENDDATE の値を取得します。
5	開始日付の範囲 (ジョブ)	ジョブログテーブルの STARTDATE の値を取得します。
6	終了日付の範囲 (ジョブ)	ジョブログテーブルの ENDDATE の値を取得します。
7	昨日の開始 00:00:00	データ変換実行時の昨日の開始日時を返します。時刻は 00:00:00 です。
8	昨日の終了 23:59:59	データ変換実行時の昨日の終了日時を返します。時刻は 23:59:59 です。

項番	システム情報	説明
9	今日の開始 00:00:00	データ変換実行当日の開始日時を返します。時刻は 00:00:00 です。
10	今日の終了 23:59:59	データ変換実行当日の終了日時を返します。時刻は 23:59:59 です。
11	明日の開始 00:00:00	データ変換実行時の翌日の開始日時を返します。時刻は 00:00:00 です。
12	明日の終了 23:59:59	データ変換実行時の翌日の終了日時を返します。時刻は 23:59:59 です。
13	最終月の最初の日 00:00:00	データ変換実行時の先月の最初の日の日時を返します。時刻は 00:00:00 です。
14	最終月の最終日 23:59:59	データ変換実行時の先月の最終日の日時を返します。時刻は 23:59:59 です。
15	今月の最初の日 00:00:00	データ変換実行時の当月の最初の日の日時を返します。時刻は 00:00:00 です。
16	今月の最終日 23:59:59	データ変換実行時の当月の最終日の日時を返します。時刻は 23:59:59 です。
17	翌月の最初の日 00:00:00	データ変換実行時の翌月の最初の日の日時を返します。時刻は 00:00:00 です。
18	翌月の最終日 23:59:59	データ変換実行時の翌月の最終日の日時を返します。時刻は 23:59:59 です。
19	ステップのコピー	ステップのコピー番号を取得します。
20	トランスフォーメーション 名	データ変換名を取得します。
21	トランスフォーメーション ファイル名	データ変換ファイル名を取得します。
22	最後にトランスフォーメーション を変更したユーザ	最後にデータ変換定義を変更したユーザー名を取得します。
23	トランスフォーメーション が最終更新された日付	データ変換定義の最終更新日付を取得します。
24	トランスフォーメーション バッチ ID	ログテーブルの ID_BATCH の値を取得します。
25	親ジョブのバッチ ID	親のジョブ ID を取得します。ジョブプロパティの設定で、「バッチ ID を渡す」に チェックを付けておく必要があります。
26	ホスト名 (ネットワーク設 定)	—
27	ホスト名	実行環境のホスト名を取得します。
28	IP アドレス	実行環境の IP アドレスを取得します。
29	コマンドライン引数 1	引数 1 番目に指定した値を取得します。
30	コマンドライン引数 2	引数 2 番目に指定した値を取得します。
31	コマンドライン引数 3	引数 3 番目に指定した値を取得します。
32	コマンドライン引数 4	引数 4 番目に指定した値を取得します。
33	コマンドライン引数 5	引数 5 番目に指定した値を取得します。
34	コマンドライン引数 6	引数 6 番目に指定した値を取得します。

項番	システム情報	説明
35	コマンドライン引数 7	引数 7 番目に指定した値を取得します。
36	コマンドライン引数 8	引数 8 番目に指定した値を取得します。
37	コマンドライン引数 9	引数 9 番目に指定した値を取得します。
38	コマンドライン引数 10	引数 10 番目に指定した値を取得します。
39	Kettle のバージョン	kettle のバージョンを取得します。
40	Kettle のビルドバージョン	kettle のビルドバージョンを取得します。
41	Kettle のビルド日付	kettle のビルド日時を取得します。
42	プロセス ID(PID)	プロセス ID を取得します。
43	JVM の最大メモリ	JVM の最大メモリサイズを取得します。
44	JVM のトータルメモリ	JVM のトータルメモリサイズを取得します。
45	JVM のフリーなメモリ	JVM のフリーメモリサイズを取得します。
46	JVM の利用可能なメモリ	JVM の利用可能なメモリサイズを取得します。
47	利用可能なプロセッサ	利用可能なプロセッサ数を取得します。
48	JVM の CPU 時間 (ミリ秒)	JVM の CPU 時間 (ミリ秒)を取得します。
49	トータル物理メモリサイズ (バイト)	トータル物理メモリサイズ (バイト)を取得します。
50	トータル swap スペースサイズ (バイト)	トータル swap スペースサイズ (バイト)を取得します。
51	コミットされた仮想メモリサイズ (バイト)	コミットされた仮想メモリサイズ (バイト)を取得します。
52	空いている物理メモリサイズ (バイト)	空いている物理メモリサイズ (バイト)を取得します。
53	空いている swap スペースサイズ (バイト)	空いている swap スペースサイズ (バイト)を取得します。
54	最終週の最初の日 00:00:00※	実行時の先週の最初の日 (日曜日) の日時を取得します。時刻は 00:00:00 です。
55	最終週の最終日 23:59:59※	実行時の先週の最終日 (土曜日) の日時を取得します。時刻は 23:59:59 です。
56	最終週の最終 open 日 23:59:59※	実行時の先週の最後の営業日 (金曜日) の日時を取得します。時刻は 23:59:59 です。
57	最終週の最初の日 00:00:00 (US)※	実行時の先週の最初の日 (日曜日) の日時を取得します。時刻は 00:00:00 です。US は US ロケールを意味します。日曜日を週の始まりとします。
58	最終週の最終日 23:59:59 (US)※	実行時の先週の最終日 (土曜日) の日時を取得します。時刻は 23:59:59 です。US は US ロケールを意味します。日曜日を週の始まりとします。
59	今週の最初の日 00:00:00	実行時の今週の最初の日 (日曜日) の日時を取得します。時刻は 00:00:00 です。

項番	システム情報	説明
60	今週の最終日 23:59:59	実行時の今週の最終日（土曜日）の日時を取得します。時刻は 23:59:59 です。
61	今週の最終 open 日 23:59:59	実行時の今週の最後の営業日（金曜日）の日時を取得します。時刻は 23:59:59 です。
62	今週の最初の日 00:00:00 (US)	実行時の今週の最初の日（日曜日）の日時を取得します。時刻は 00:00:00 です。US は US ロケールを意味します。日曜日を週の始まりとします。
63	今週の最終日 23:59:59 (US)	実行時の今週の最終日（土曜日）の日時を取得します。時刻は 23:59:59 です。US は US ロケールを意味します。日曜日を週の始まりとします。
64	翌週の最初の日 00:00:00	実行時の翌週の最初の日（日曜日）の日時を取得します。時刻は 00:00:00 です。
65	翌週の最終日 23:59:59	実行時の翌週の最終日（土曜日）の日時を取得します。時刻は 23:59:59 です。
66	翌週の最終 open 日 23:59:59	実行時の翌週の最後の営業日（金曜日）の日時を取得します。時刻は 23:59:59 です。
67	翌週の最初の日 00:00:00 (US)	実行時の翌週の最初の日（日曜日）の日時を取得します。時刻は 00:00:00 です。US は US ロケールを意味します。日曜日を週の始まりとします。
68	翌週の最終日 23:59:59 (US)	実行時の翌週の最終日（土曜日）の日時を取得します。時刻は 23:59:59 です。US は US ロケールを意味します。日曜日を週の始まりとします。
69	前四半期の最初の日 00:00:00	実行時の前四半期の最初の日の日時を取得します。時刻は 00:00:00 です。
70	前四半期の最終日 23:59:59	実行時の前四半期の最終日の日時を取得します。時刻は 23:59:59 です。
71	今四半期の最初の日 00:00:00	実行時の今四半期の最初の日の日時を取得します。時刻は 00:00:00 です。
72	今四半期の最終日 23:59:59	実行時の今四半期の最終日の日時を取得します。時刻は 23:59:59 です。
73	翌四半期の最初の日 00:00:00	実行時の来四半期の最初の日の日時を取得します。時刻は 00:00:00 です。
74	翌四半期の最終日 23:59:59	実行時の来四半期の最終日の日時を取得します。時刻は 23:59:59 です。
75	前年の最初の日 00:00:00	実行時の前年の最初の日の日時を取得します。時刻は 00:00:00 です。
76	前年の最終日 23:59:59	実行時の前年の最終日の日時を取得します。時刻は 23:59:59 です。
77	今年の最初の日 00:00:00	実行時の今年の最初の日の日時を取得します。時刻は 00:00:00 です。
78	今年の最終日 23:59:59	実行時の今年の最終日の日時を取得します。時刻は 23:59:59 です。
79	翌年の最初の日 00:00:00	実行時の翌年の最初の日の日時を取得します。時刻は 00:00:00 です。
80	翌年の最終日 23:59:59	実行時の翌年の最終日の日時を取得します。時刻は 23:59:59 です。
81	前のジョブエントリ結果	前のジョブエントリ結果を取得します。
82	前のジョブエントリーの終了ステータス	前のジョブエントリの終了ステータスを取得します。

項番	システム情報	説明
83	前のジョブエントリー番号	前のジョブエントリーの番号を取得します。
84	前のジョブエントリーでのエラー数	—
85	過去のジョブエントリーの入力行数	—
86	過去のジョブエントリーの出力行数	—
87	過去のジョブエントリーの読み込み行数	—
88	過去のジョブエントリーの更新行数	—
89	過去のジョブエントリーの書き込み行数	—
90	過去のジョブエントリーの削除行数	—
91	過去のジョブエントリーの拒否行数	—
92	前のジョブエントリーの行数	—
93	前のジョブエントリーが停止	—
94	前のステップでのジョブエントリーファイル数	—
95	前のジョブエントリーで取得したファイル数	—
96	前のジョブエントリーでのログ	前のジョブエントリーのログを取得します。

(凡例) —：説明はありません。

注※ 画面の項目名と指定する内容が異なります。

2.2.9 [入力]テキストファイル入力 (Text file input)

CSV 形式のファイル、またはフィールド分割単位が固定幅のファイルのデータを読み込みます。入力ファイルのリストを作成し、そのリストに含まれるすべての入力ファイルからデータを読み込むため、複数ファイルも入力できます。

[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。

(1) [ファイル]タブ

入力ファイルについて設定します。

[ファイル名の参照]ボタンを押すと、入力ファイルリストの一覧が表示されます。

[ファイル内容の表示]ボタンおよび[最初のレコード]ボタンを押すと、[入力ファイルリスト]の最初のファイルのテキスト内容が表示されます。

図 2-24 [テキストファイル入力]画面-[ファイル]タブ

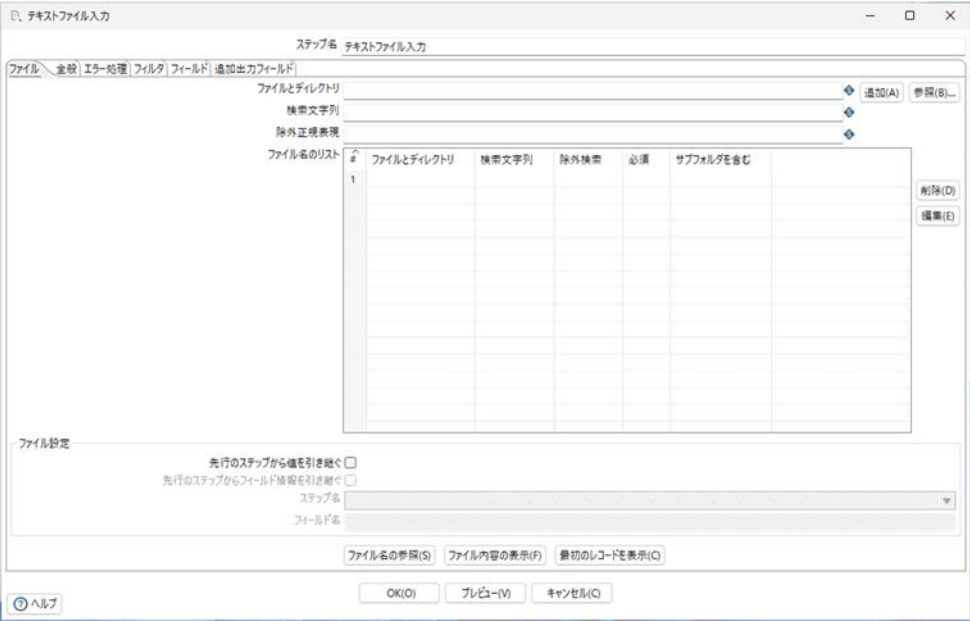


表 2-26 [テキストファイル入力]画面-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ファイルとディレクトリ]	入力ファイルのパス、または入力ファイルが格納されているディレクトリのパスを指定します。 ディレクトリを指定する場合、[検索文字列]が空欄のときはエラーが発生します。 [追加]ボタンを押すと、パスが[ファイル名のリスト]に登録されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[検索文字列]	ファイルパスを検索する文字列を指定します。指定した文字列と完全一致するファイルパスを入力ファイルリストに追加します。 正規表現を使用した検索もできます。
4	[除外正規表現]	ファイルパスを検索する文字列を指定します。指定した文字列と完全一致するファイルパスを入力ファイルリストから削除します。 正規表現を使用した検索もできます。
5	[ファイル名のリスト]	入力ファイルのリストを表示します。 ファイルとディレクトリ

項番	項目名	説明
		指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。 • 検索文字列 [検索文字列]に指定した文字列を表示します。 • 検索除外 [除外正規表現]に指定した文字列を表示します。 • 必須 該当の入力ファイルが入力必須かどうかを[Y]または[N]で指定します。 [Y]を指定すると、入力ファイルに該当するファイルが1つも見つからない場合に、エラーが発生します。 • サブフォルダを含む ファイルの検索対象として、該当のディレクトリのサブフォルダを検索するかどうかを[Y]または[N]で指定します。 [Y]を指定すると、サブフォルダも検索します。
6	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値をファイルパスとして使用します。 フィールドの値として指定するファイルパスには、正規表現は使用できません。 チェックを付けると、[ファイル名のリスト]を使用したデータの読み込みはできません。
7	[先行のステップからフィールド情報を引き継ぐ]	この項目にチェックを付けると、先行のステップのデータをテキストファイル入力ステップで読み込んだデータと結合して出力します。
8	[ステップ名]	先行のステップ名を指定します。
9	[フィールド名]	値がファイルパスとなっているフィールド名を指定します。
10	[ファイル名の参照] ボタン	このボタンを押すと、[Files read]画面を表示します。
11	[ファイル内容の表示] ボタン	このボタンを押すと、[最初のファイルの内容]画面を表示します。この画面では、ファイルの内容が表示されます。
12	[最初のレコードを表示] ボタン	このボタンを押すと、[出力するレコード数]画面を表示します。この画面で出力するレコード数を入力し、[OK]ボタンを押すと、[最初のファイルの内容]画面を表示します。

(2) [全般]タブ

読み込むテキストファイルのフォーマットを設定します。

図 2-25 [テキストファイル入力]画面-[全般]タブ

テキストファイル入力

ステップ名: テキストファイル入力

ファイル

全般

エラー処理

フィルタ

フィールド

追加出力フィールド

ファイルタイプ

CSV

フィールド区切り文字

:

引用符

・

エスケープ文字

ヘッダー

☒

ヘッダーレコード数

1

フッター

☐

フッターレコード数

1

含むレコード数

☐

含まれる回数

1

ページレイアウト(印刷)

☐

1ページあたりのレコード数

80

ヘッダーレコード数

0

圧縮形式

None

空のレコードを除外

☒

ファイル名を出力に含む

☐

フィールド名

レコード数を出力に含む

☐

フィールド名

ファイル毎にレコード数をリセットする

☐

フォーマット

DOS

文字コード

Length

Characters

最大レコード数

0

日付を自動処理

☒

ローカル日付形式

ja_JP

オプション設定

ファイル名を結果に含む

☒

ヘルプ

OK(O)

プレビュー(V)

キャンセル(Q)

表 2-27 [テキストファイル入力]画面-[全般]タブの項目

項番	項目名	説明
1	[ファイルタイプ]	入力ファイルの形式を選択します。 [CSV]または[Fixed]（固定長）のどちらかを選択します。
2	[フィールド区切り文字]	データを読み込む CSV ファイルの区切り文字を指定します。[入力タブ]ボタンを押すと、タブ文字を入力できます。 なお、デフォルトでは、「, (コンマ)」ではなく、「; (セミコロン)」が指定されています。
3	[引用符]	この項目にチェックを付けると、データを読み込む CSV ファイルの引用符を指定します。
4	[エスケープ文字]	エスケープ文字を指定します。
5	[ヘッダー]	この項目にチェックを付けると、入力ファイルにヘッダーを含めます。 [ヘッダーレコード数] 入力ファイルの先頭から指定したレコード数分をヘッダーレコードとして読み飛ばします。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
6	[フッター]	この項目にチェックを付けると、入力ファイルにフッターを含めます。 [フッターレコード数] 入力ファイルの末尾から指定したレコード数分をフッターレコードとして読み飛ばします。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
7	[含むレコード数]	この項目にチェックを付けると、フィールド内の改行を読み飛ばします。

項番	項目名	説明
		[含まれる回数] フィールド内に改行文字が含まれている場合に、1 レコードにつき指定した回数（含まれる回数）だけ改行文字を読み飛ばします。 フィールドの値に改行文字を含む場合は、引用符を使用してください。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
8	[ページレイアウト(印刷)]	この項目にチェックを付けると、印刷時の画面レイアウトを設定できます。 1 ファイルを固定レコード数（1 ページ当たりのレコード数）のページに区切り、各ページの最初から指定したレコード数分をヘッダーレコードとして読み飛ばしたレイアウトにします。 [1 ページ当たりのレコード数] 1 ページ当たりのレコード数を指定します。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。 [ヘッダーレコード数] 各ページの最初からヘッダーレコードとして読み飛ばすレコード数を指定します。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
9	[圧縮形式]	入力ファイルの圧縮形式を選択します。 アーカイブファイルの場合、始めのファイルだけを読み込みます。 非圧縮ファイルの場合は、[None]を指定します。
10	[空のレコードを除去]	この項目にチェックを付けると、レコード内の値がすべて NULL 値のレコードを読み飛ばします。
11	[ファイル名を出力に含む]	この項目にチェックを付けると、指定したフィールドにレコードを読み込んだ入力ファイルのパスを出力します。 [フィールド名] レコードを読み込んだ入力ファイルのパスを出力するフィールドを指定します。
12	[レコード数を出力に含む]	この項目にチェックを付けると、指定したフィールドにレコード番号を出力します。 [フィールド名] レコードを読み込んだ入力ファイルのレコード番号を出力するフィールドを指定します。
13	[ファイル毎にレコード数をリセットする]	この項目にチェックを付けると、入力ファイルごとにレコード数をリセットします。 【注意事項】 チェックを付けない場合も、入力ファイルごとにレコード数がリセットされます。
14	[フォーマット]	改行文字のフォーマットを検証します。選択したフォーマットと一致しない場合は、エラーが発生します。 • [DOS] 改行文字が CR+LF の場合に選択します。ファイルに含まれるすべての改行文字が CR+LF の場合に、正常にデータを読み込みます。 • [Unix]

項番	項目名	説明
		改行文字が LF または CR の場合に選択します。ファイルに含まれるすべての改行文字が LF または CR の場合に、正常にデータを読み込みます。 • [mixed] 改行文字のフォーマットを検証しません。改行文字が混在する場合は、[mixed] を選択してください。
15	[文字コード]	入力ファイルの文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。
16	[Length]	マルチバイト文字を読み込む際の、データの長さの単位を[Characters]または[Bytes]のどちらかで選択します。デフォルトでは、[Characters]が選択されています。 • [Characters] マルチバイト文字を文字数で読み込みます。 • [Bytes] [ファイルタイプ]に「Fixed」を選択し、[文字コード]に値を選択している場合に、マルチバイト文字をバイト数で読み込みます。
17	[最大レコード数]	読み込むデータのレコード数の最大を指定します。SQL 文の LIMIT 句と同様の処理です。 0 を指定すると、すべてのレコードを読み込みます。
18	[日付を自動処理]	この項目にチェックを付けると、文字列などを日付データ型に変換するときに、変換できる文字列のパターンが多くなります。 チェックを付けない場合、該当するパターンの文字列などを日付データ型に変換するときに、エラーが発生します。 変換できる文字列のパターンとして、"2000/02/00"（文字列）を変換する例を次に示します。 変換前 "2000/02/00" 変換後 "2000/01/31"
19	[ローカル日付書式]	日付のローケルを選択します。
20	[ファイル名を結果に含む]	この項目にチェックを付けると、データ入力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type • filename 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin

項番	項目名	説明
		[ファイル名を結果に含む]を[Y]にしたステップ名 - comment - timestamp

(3) [エラー処理]タブ

エラーが発生した場合にステップがどのように反応するかを設定します。

図 2-26 [テキストファイル入力]画面-[エラー処理]タブ

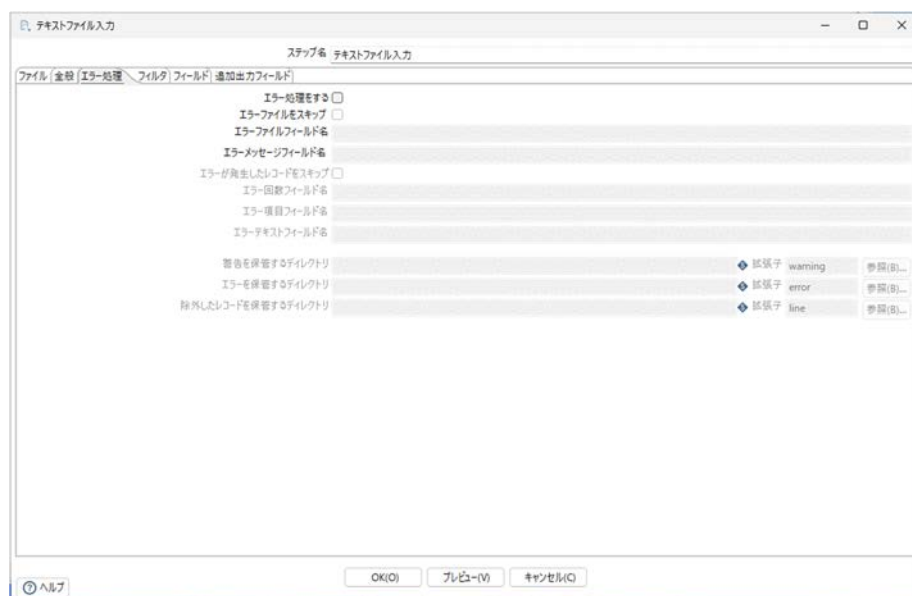


表 2-28 [テキストファイル入力]画面-[エラー処理]タブの項目

項番	項目名	説明
1	[エラー処理をする]	この項目にチェックを付けると、エラーが発生したときの処理を定義できます。
2	[エラーファイルをスキップ]	この項目にチェックを付けると、アクセス権がないなどの理由で、ファイルエラーが発生したとき、エラーにしません。
3	[エラーファイルフィールド名]	ファイルエラーの発生したファイル名を、指定したフィールドに出力します。このフィールドはエラーフローに出力されます。
4	[エラーメッセージフィールド名]	ファイルエラーのエラーメッセージを、指定したフィールドに出力します。このフィールドはエラーフローに出力されます。
5	[エラーが発生したレコードをスキップ]	この項目にチェックを付けると、あるレコードを読み込むときに、エラーが発生すると、該当レコードの読み込みをスキップします。
6	[エラー回数フィールド名]	設定したフィールドに、エラーレコード内でエラーが発生したフィールド数を出力します。
7	[エラー項目フィールド名]	設定したフィールドに、エラーレコード内でエラーが発生するフィールド名を出力します。

項番	項目名	説明
8	[エラーテキストフィールド名]	設定したフィールドに、エラーの詳細を出力します。
9	[警告を保管するディレクトリ]	—
10	[エラーを保管するディレクトリ]	指定したファイルが存在しない場合やファイルエラーが発生した場合に、指定したディレクトリにエラーメッセージを出力します。 エラーメッセージを出力するファイルの拡張子を[拡張子]で指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
11	[除外したレコードを保管するディレクトリ]	エラーレコードが発生した場合に、そのエラーレコード番号を、指定したディレクトリに出力します。 エラーレコードを出力するファイルの拡張子を[拡張子]で指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。

(凡例) —：説明はありません。

(4) [フィルタ]タブ

各レコードに対して、指定した位置に検索文字列が含まれているかどうかを判断し、フィルタリングします。

なお、検索文字列の開始位置を固定値で指定する必要があるため、各フィールドが固定長でない場合は、使用しないでください。

図 2-27 [テキストファイル入力]画面-[フィルタ]タブ

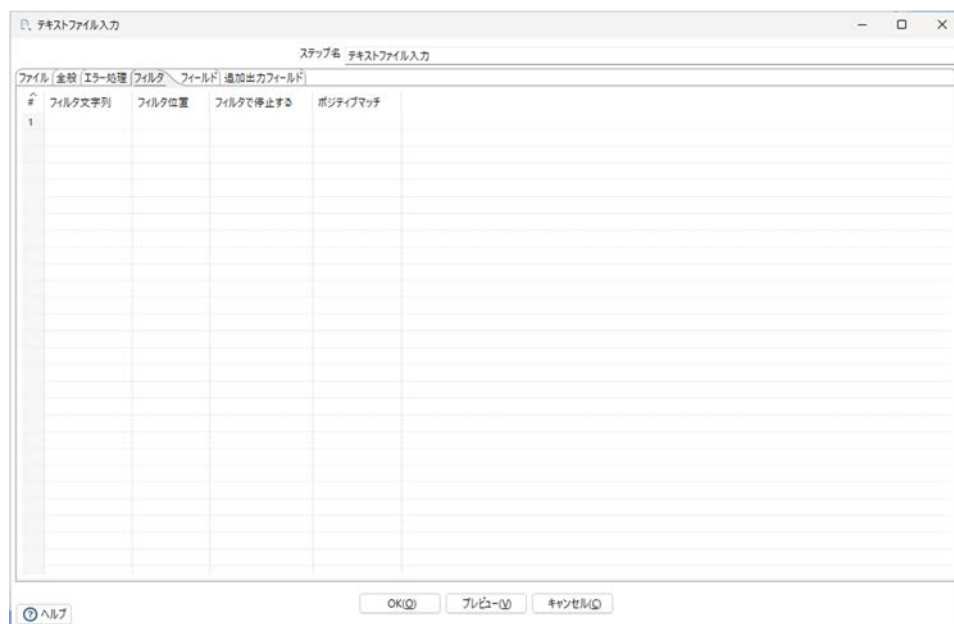


表 2-29 [テキストファイル入力]画面-[フィルタ]タブの項目

項番	項目名	説明
1	[フィルタ文字列]	検索文字列を指定します。

項番	項目名	説明
2	[フィルタ位置]	検索対象の文字列の開始位置を指定します。 例えば、1 文字目以降の文字列が検索文字列と一致するかを判定したい場合は、0 を指定します。 位置を指定する場合、区切り文字も 1 文字としてカウントします。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
3	[フィルタで停止する]	検索文字列が含まれたレコードがあった場合に、その時点でファイルの入力を停止するかどうかを指定します。 [Y]を指定した場合、ファイルの入力を停止します。 この項目は、[ポジティブマッチ]に[N]を指定した場合に有効です。
4	[ポジティブマッチ]	検索文字列が含まれないレコードの読み込みをスキップするか、検索文字列が含まれるレコードをスキップするかのどちらかを指定します。 • [Y] 検索文字列が含まれないレコードの読み込みをスキップします。 • [N] 検索文字列が含まれるレコードの読み込みをスキップします。

(5) [フィールド]タブ

読み込むデータのフィールド情報を定義します。

[フィールドを取得]ボタンを押すと、入力ファイルからフィールド定義を自動推定できます。

図 2-28 [テキストファイル入力]画面-[フィールド]タブ

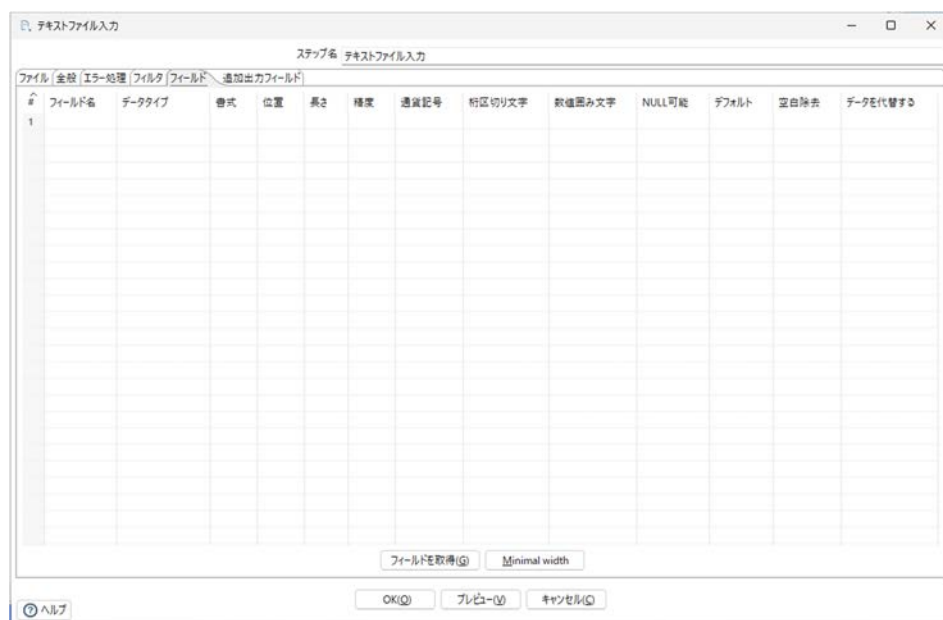


表 2-30 [テキストファイル入力]画面-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。

項番	項目名	説明
2	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
3	[書式]	入力書式を指定します。
4	[位置]	ファイルタイプが[Fixed]の場合に、該当フィールドの開始位置を指定します。最初の文字は位置 0 から始まります。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
5	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
6	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
7	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
8	[桁区切り文字]※	「. (ドット)」など、小数点を表す文字を指定します。
9	[数値囲み文字]※	「, (コンマ)」など、桁を区切る文字を指定します。
10	[NULL 可能]	読み込んだデータがこの項目で指定した値の場合、NULL 値に変換します。 大文字および小文字は区別して変換します。 [フィールド名]を「data1」に指定し、data1 の値を次のとおり（3 行のデータ）とした場合の例を示します。 aaa bbb ccc [NULL 可能]に「bbb」を指定した場合、 読み込んだ結果は次のとおりです。 aaa <null> ccc
11	[デフォルト]	値が NULL 値の場合、指定した値に変換します。
12	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。

項番	項目名	説明
13	[データを代替する]	フィールドの値が NULL 値の場合に、NULL 値を同じフィールドの 1 つ前のレコードの値に変換するかどうかを指定します。 • [Y] NULL 値を同じフィールドの 1 つ前のレコードの値に変換します。1 つ前のレコードの値がない場合は、NULL 値のまま出力します。 • [N] NULL 値の変換をしません。

注※ 画面の項目名と指定する内容が異なります。

(6) [追加出力フィールド]タブ

フィールドに追加の設定をします。

フィールドを追加設定すると、設定したフィールドに入力ファイルの情報を出力できます。

図 2-29 [テキストファイル入力]画面-[追加出力フィールド]タブ



表 2-31 [テキストファイル入力]画面-[追加出力フィールド]タブの項目

項番	項目名	説明
1	[短縮ファイル名フィールド]	入力ファイルのファイル名を出力するためのフィールドを指定します。
2	[拡張子フィールド]	入力ファイルの拡張子を出力するためのフィールドを指定します。
3	[パスフィールド]	入力ファイルが格納されているディレクトリの絶対パスを出力するためのフィールドを指定します。
4	[サイズフィールド]	入力ファイルのファイルサイズ（単位：バイト）を出力するためのフィールドを指定します。

項番	項目名	説明
5	[隠しフィールド]	入力ファイルが隠しファイルかどうかを出力するためのフィールドを指定します。
6	[最終更新フィールド]	入力ファイルの最終更新日を出力するためのフィールドを指定します。
7	[Uri フィールド]	入力ファイルの URI を出力するためのフィールドを指定します。
8	[ルート Uri フィールド]	入力ファイルのルートディレクトリの URI を出力するためのフィールドを指定します。

2.2.10 [入力]テーブル入力 (Table input)

データベースのテーブルから SQL 文 (SELECT 文) を使用してデータを読み込みます。

[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。

図 2-30 [テーブル入力]画面

表 2-32 [テーブル入力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	入力テーブルのデータソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。

項番	項目名	説明
3	[SQL ステートメント]	データ読み込みのための SQL 文（SELECT 文）を入力します。 [SQL の選択]ボタンを押すと、データベース内のテーブル情報を参照できます。
4	[Store column info in step meta data]	この機能の動作は不明です。
5	[データ型を自動変換]	この項目にチェックを付けると、読み込んだデータを Java のデータ型に変換しないで、バイトデータのまま後続ステップへ出力します。 【注意事項】 入力の処理性能は向上しますが、後続処理のエラーの原因となるおそれがあるため、この項目にチェックを付けないでください。
6	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、[SQL ステートメント]で変数を使用できます。
7	[ステップ名]	先行ステップ名を入力すると、先行ステップから入力されたレコードの値を[SQL ステートメント]でパラメータとして使用できます。 パラメータ変数は、「?（クエスチョンマーク）」を使用します。 WHERE 句にパラメータを使用する場合の例を次に示します。 【先行ステップからの入力レコード】 value1 value2 0 100 【SQL ステートメント】 SELECT * FROM test_table WHERE chek_column between ? AND ? この例では、1 つ目の「?」は 0、2 つ目の「?」は 100 に置き換えられて実行されます。 なお、テーブル名や列名にパラメータは使用できません。 【注意事項】 - SQL 文内に入力されたパラメータの数と同数の「?」が記述されていない場合、エラーが発生します。また、入力したパラメータは、SQL 文内ですべて使用する必要があります。 - エラーの原因となるおそれがあるため、[レコード単位で実行]にチェックを付けない場合、入力レコード数を 1 レコードとしてください。
8	[レコード単位で実行]	この項目にチェックを付けると、先行ステップからの入力レコードごとにパラメータを反映した SQL 文を発行し、データを繰り返し読み込みます。
9	[最大レコード数]	読み込むデータのレコード数の最大を指定します。 0 を指定すると、すべてのレコードを読み込みます。

2.2.11 [入力]データグリッド (Data grid)

ユーザーが定義したテーブル形式の静的データを生成します。

データ変換のデバッグやテストなどによく使用されます。[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。

(1) [メタ]タブ

生成するデータのメタデータを定義します。

図 2-31 [データグリッド]画面-[メタ]タブ



表 2-33 [データグリッド]画面-[メタ]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	フィールド名を指定します。
3	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[書式]	出力書式を選択します。 【注意事項】 [データタイプ]の指定に関係なく、同じリストが表示されます。
5	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
6	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
7	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
8	[桁区切り文字]※	「. (ドット)」など、小数点を表す文字を指定します。
9	[数値囲み文字]※	「, (コンマ)」など、桁を区切る文字を指定します。
10	[Null if]	読み込んだデータがこの項目で指定した値の場合、NULL 値に変換します。 大文字と小文字は区別しません。
11	[空文字を設定]	該当フィールドの値をすべて空文字にするかどうかを[Y]または[N]で指定します。

注※ 画面の項目名と指定する内容が異なります。

(2) [データ]タブ

[メタ]タブで指定したメタデータに対し、実データ情報を定義します。入力された値はすべて String 型で表示されるため、設定する値は[メタ]タブで定義したメタデータと形式が一致している必要があります。

図 2-32 [データグリッド]画面-[データ]タブ



2.2.12 [入力]ファイルメモリロード (Load file content in memory)

入力ファイルからファイルの内容とファイルサイズを読み込みます。

(1) [ファイル]タブ

入力ファイルに関する設定をします。

[ファイル名を参照]ボタンを押すと、入力ファイルリストの一覧が表示されます。

図 2-33 [ファイルメモリロード]画面-[ファイル]タブ



表 2-34 [ファイルメモリロード]画面-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

項番	項目名	説明
2	[ファイル名はフィールドに定義されている]	この項目にチェックを付けると、先行のステップで定義したフィールドからファイル名を取得できます。[フィールドからファイル名を取得]を指定してください。
3	[フィールドからファイル名を取得]	先行のステップで定義したフィールド名を指定します。 [ファイル名はフィールドに定義されている]にチェックを付けると指定できます。
4	[ファイル/ディレクトリ]	[ファイル名はフィールドに定義されている]にチェックを付けない場合、入力ファイルのパス、または入力ファイルが格納されているディレクトリのパスを指定します。 [追加]ボタンを押すと、[選択したファイル]欄のリストに登録されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 【注意事項】 アクセス権のないファイルまたはディレクトリを指定してもエラーは発生しません。ただし、ファイルは読み込めません。
5	[検索文字列]	[ファイル名はフィールドに定義されている]にチェックを付けない場合、[ファイル/ディレクトリ]で指定したディレクトリ内のファイル名に一致する文字列を指定します。 指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを入力ファイルリストに追加します。 正規表現を使用した検索もできます。
6	[除外正規表現]	[ファイル名はフィールドに定義されている]にチェックを付けない場合、[ファイル/ディレクトリ]で指定したディレクトリ内のファイル名を入力ファイルリストから除外する正規表現を指定します。 指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを入力ファイルリストから除外します。 正規表現を使用した検索もできます。
7	[選択したファイル]	[ファイル名はフィールドに定義されている]にチェックを付けない場合、次の検索対象のリストを指定します。 • [ファイル/ディレクトリ] [ファイル/ディレクトリ]に指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。 • [検索文字列] [検索文字列]に指定した文字列を表示します。 • [除外検索] [除外正規表現]に指定した文字列を表示します。 • [必須] [Y]を指定すると、入力ファイルに該当するファイルが1つも見つからない場合に、エラーが発生します。 • [サブフォルダーを含む] [Y]を指定すると、ファイルの検索範囲がサブフォルダまで広がります。
8	[除去]ボタン	このボタンを押すと、ファイルリストで選択した項目を削除します。

項番	項目名	説明
9	[編集]ボタン	このボタンを押すと、ファイルリストで選択した項目を編集します。

(2) [コンテンツ]タブ

取得するデータのコンテンツに関する設定をします。

図 2-34 [ファイルメモリロード]画面-[コンテンツ]タブ

The screenshot shows the 'File Memory Load' dialog box with the 'Contents' tab selected. The 'Settings' section includes a dropdown for 'Encoding', checkboxes for 'Ignore empty files' and 'Ignore missing path', and a text field for 'Limit size' set to 0. The 'Add fields' section has checkboxes for 'Output file name' and 'Output row number'. The 'Add to result file name' section has a checked checkbox 'Add files to result file'.

表 2-35 [ファイルメモリロード]画面-[コンテンツ]タブの項目

項番	項目名	説明
1	[エンコーディング]	入力ファイルの文字コードを指定します。 MS932 を指定する場合は、[windows-31j]を指定してください。
2	[空のファイルを無視]	この項目にチェックを付けると、空のファイルは無視します。無視された場合はエラーメッセージが出力されますが、データ変換は成功します。チェックを付けない場合は、空のファイルも 1 レコードとしてカウントされます。
3	[Ignore missing path]	[ファイル]タブの[ファイル名はフィールドに定義されている]にチェックを付けている場合で、かつ、フィールドに指定したファイルが存在しない場合、この項目にチェックを付けると、エラーが発生しません。チェックを付けない場合、ファイルが存在しなかったときは、エラーが発生します。
4	[リミットサイズ]	入力データの読み込むファイルの数の制限を指定します。 0 を設定すると制限されません。 [ファイル]タブの[ファイル名はフィールドに定義されている]にチェックを付けている場合、この項目は非活性になります。
5	[出力にファイル名を含む]	この項目にチェックを付けると、出力結果にファイル名を含む文字列フィールドを追加します。

項番	項目名	説明
		[ファイル]タブの[ファイル名はフィールドに定義されている]にチェックを付けている場合、この項目は非活性になります。
6	[フィールド名]	[出力にファイル名を含む]にチェックが付いている場合、ファイル名を含む文字列フィールドの名前を指定します。 [ファイル]タブの[ファイル名はフィールドに定義されている]にチェックを付けている場合、この項目は非活性になります。
7	[出力に Rownum を含む]	この項目にチェックを付けると、出力結果にファイルを読み込んだ数を表す整数フィールドを追加します。
8	[Rownum フィールド名]	[出力に Rownum を含む]にチェックが付いている場合、ファイルを読み込んだ数を表す整数フィールドの名前を指定します。
9	[ファイルを結果ファイルに追加]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type • filename 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [ファイルを結果ファイルに追加]を[Y]にしたステップ名 • comment • timestamp [ファイル]タブの[ファイル名はフィールドに定義されている]にチェックを付けている場合、この項目は非活性になります。

(3) [フィールド]タブ

読み込むデータのフィールド情報を定義します。

[フィールドの取得]ボタンを押すと、入力ファイルからフィールド定義を自動推定できます。

図 2-35 [ファイルメモリロード]画面-[フィールド]タブ



表 2-36 [ファイルメモリロード]画面-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。
2	[要素]	ファイルの要素を次の項目から指定します。 - ファイルコンテンツ 入力ファイルに記載されている内容を読み込みます。 - ファイルサイズ 入力ファイルのサイズをバイト単位で読み込みます。
3	[データタイプ]	データタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[書式]	入力書式を指定します。
5	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
6	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
7	[通貨記号]	「\$（ドル）」など、通貨を表す記号を指定します。
8	[桁区切り文字]※	「.（ドット）」など、小数点を表す文字を指定します。
9	[数値囲み文字]※	「,（コンマ）」など、桁を区切る文字を指定します。
10	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 [左]

項番	項目名	説明
		文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [both] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
11	[データを代替する]	この機能の動作は不明です。

注※ 画面の項目名と指定する内容が異なります。

(4) [追加出力フィールド]タブ

フィールドに追加の設定をします。

フィールドを追加設定すると、設定したフィールドに入力ファイルの情報を出力できます。

図 2-36 [ファイルメモリロード]画面-[追加出力フィールド]タブ

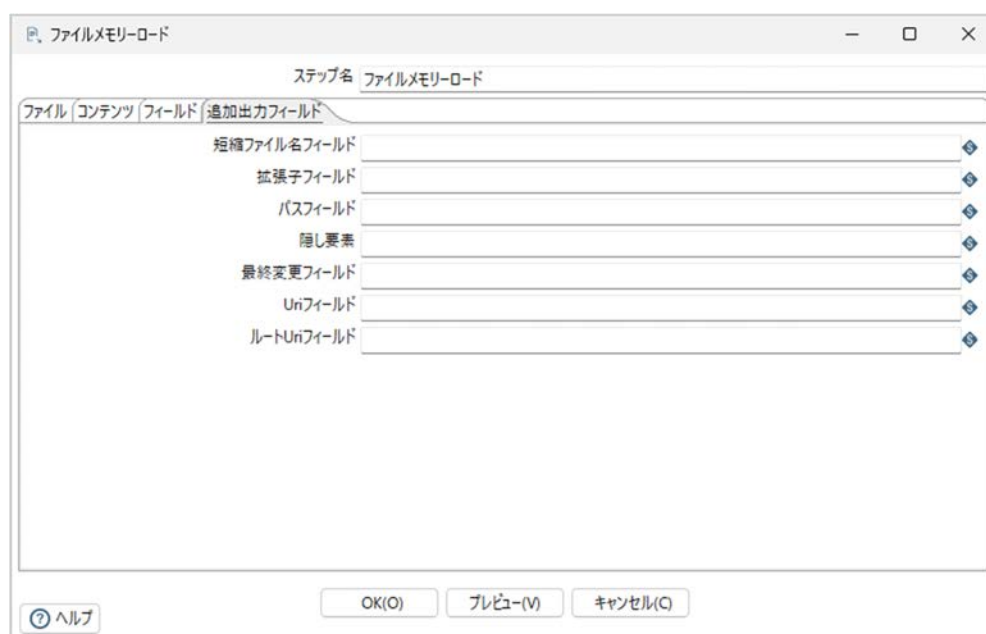


表 2-37 [ファイルメモリロード]画面-[追加出力フィールド]タブの項目

項番	項目名	説明
1	[短縮ファイル名フィールド]	入力ファイルのファイル名を出力するためのフィールドを指定します。
2	[拡張子フィールド]	入力ファイルの拡張子を出力するためのフィールドを指定します。
3	[パスフィールド]	入力ファイルが格納されているディレクトリの絶対パスを出力するためのフィールドを指定します。
4	[隠し要素]	入力ファイルが隠しファイルかどうかを出力するためのフィールドを指定します。
5	[最終更新フィールド]	入力ファイルの最終更新日を出力するためのフィールドを指定します。

項番	項目名	説明
6	[Uri フィールド]	入力ファイルの URI を出力するためのフィールドを指定します。
7	[ルート Uri フィールド]	入力ファイルのルートディレクトリの URI を出力するためのフィールドを指定します。

2.2.13 [入力]ファイル名取得 (Get file names)

ファイルシステムからファイルまたはディレクトリを検索し、ファイルパスやファイル名などのメタ情報を取得します。

[プレビュー]ボタンを押すと、出力するメタ情報のプレビューができます。

(1) [ファイル]タブ

ファイルパスやファイル名などのメタ情報を取得する対象ファイルの検索条件を指定します。

[ファイル名の参照]ボタンを押すと、ファイル名の一覧が表示されます。

図 2-37 [ファイル名取得]画面-[ファイル]タブ

表 2-38 [ファイル名取得]画面の項目-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値を検索条件として使用します。
3	[フィールド名]	ファイルおよびディレクトリが含まれているフィールドを指定します。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。

項番	項目名	説明
4	[検索文字列]	[フィールド名]で指定したファイルとディレクトリ内を、入力フィールドで指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを出力対象に追加します。 正規表現で入力する必要があります。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
5	[ワイルドカードフィールドを除く]	[フィールド名]で指定したファイルとディレクトリ内を、入力フィールドで指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを出力対象から除外します。 正規表現で入力する必要があります。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
6	[サブフォルダーを含む]	この項目にチェックを付けると、ファイルの検索範囲がサブフォルダまで広がります。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
7	[ファイルとディレクトリ]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、入力ファイルのパス、または入力ファイルが格納されているディレクトリのパスを指定します。 ディレクトリを指定した場合、[検索文字列]が空欄のときにエラーが発生します。 [追加]ボタンを押すと、ファイル名のリストに登録されます。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
8	[検索文字列]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、指定したファイルとディレクトリ内を入力フィールドで指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを出力対象に追加します。 正規表現で入力する必要があります。
9	[除外正規表現]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、指定したファイルとディレクトリ内を入力フィールドで指定した文字列で検索し、検索文字列と完全一致するファイルおよびディレクトリを出力対象から除外します。 正規表現で入力する必要があります。
10	[ファイル名のリスト]	[先行のステップから値を引き継ぐ]にチェックを付けない場合、次の検索対象のリストを指定します。 • [ファイルとディレクトリ] 指定した入力ファイル、または入力ファイルが格納されているディレクトリのパスを表示します。 • [検索文字列] [検索文字列]に指定した文字列を表示します。 • [除外検索] [除外正規表現]に指定した文字列を表示します。 • [必須] [Y]を指定すると、入力ファイルに該当するファイルが1つも見つからない場合、エラーが発生します。 • サブフォルダを含む [Y]を指定すると、ファイルの検索範囲がサブフォルダまで広がります。
11	[削除]ボタン	このボタンを押すと、選択対象をファイル名のリストから削除します。

項番	項目名	説明
12	[編集]ボタン	このボタンを押すと、選択対象のファイルまたはディレクトリのパスを編集します。

(2) [フィルター]タブ

メタ情報を出力する対象をより絞り込みたい場合に使用します。

図 2-38 [ファイル名取得]画面-[フィルター]タブ

図 2-38 は「ファイル名取得」画面の「フィルター」タブのスクリーンショットです。このタブには、フィルタリングの設定を行うための項目がいくつかあります。

- ステップ名:** ファイル名取得
- タブ:** ファイル / フィルター (現在「フィルター」タブが選択されています)
- フィルター形式:** 全てのファイル (プルダウンメニュー)
- レコード番号設定:**
 - レコード番号を出力に含む ☐ フィールド名 (テキスト入力欄)
- ファイルまたはフォルダの無い場合はエラーとしない** ☐
- 最大ファイル数:** 0 (テキスト入力欄)
- オプション設定:**
 - ファイル名を結果に含む ☒
- ボタン:** ヘルプ、OK(O)、プレビュー(V)、キャンセル(C)

表 2-39 [ファイル名取得]画面の項目-[フィルター]タブの項目

項番	項目名	説明
1	[フィルター形式]	メタ情報を出力する対象をすべてのファイルとディレクトリにしたい場合は、[すべてのファイル]、ファイルだけにしたい場合は[ファイルのみ]、ディレクトリだけにしたい場合は[フォルダのみ]を指定します。
2	[レコード番号を出力に含む]	この項目にチェックを付けると、出力データにレコード番号が付与されます。
3	[フィールド名]	レコード番号を出力するフィールド名を指定します。
4	[ファイルまたはフォルダの無い場合はエラーとしない]	この項目にチェックを付けると、[ファイル]タブの[ファイル名のリスト]の[必須]で[Y]を指定したファイルが存在しない場合でも、エラーを発生させません。
5	[最大ファイル数]	メタ情報を出力するファイル数の最大を指定します。 検索にヒットした順に、指定した数-1 ファイル分のメタ情報が出力されます。 0 を指定した場合、レコード数の上限は設定されません。
6	[ファイル名を結果に含む]	この項目にチェックを付けると、データ入力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 チェックを付けない場合、メモリ使用量を削減できます。

2.2.14 [入力]固定幅ファイル入力 (Fixed File input)

固定幅データを含むテキストファイルから、値を取得します。

[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。[フィールドを取得]ボタンを押すと、固定幅やフィールド情報を編集できます。

図 2-39 [固定幅ファイル入力]画面

表 2-40 [固定幅ファイル入力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ファイル名]	入力ファイルのパスを指定します。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
3	[固定長バイト数 (改行コード除く)]	改行文字 (CR や LF) を除いた、入力ファイルのデータの幅 (単位: バイト) を指定します。
4	[改行を含む]	入力ファイルに改行を含む場合に、この項目にチェックを付けます。
5	[バッファサイズ]	IO バッファのサイズ (単位: バイト) を指定します。 【注意事項】 ここで指定した値が[固定長バイト数 (改行コード除く)]で指定した値以下の場合にデータ変換を実行すると、エラーが発生します。 ここで指定する値は、[固定長バイト数 (改行コード除く)]よりも大きな値を指定してください。
6	[データ型を自動変換]	この項目にチェックを付けると、読み込んだデータを Java のデータ型に変換しないで、バイトデータのまま後続ステップへ出力します。 【注意事項】 入力の処理性能は向上しますが、後続処理のエラーの原因となるおそれがあるため、チェックを付けしないでください。
7	[ヘッダーレコードを含む]	この項目にチェックを付けると、入力ファイルの先頭行だけをヘッダーレコードとして読み飛ばします。

項番	項目名	説明
8	[分散処理する]	この項目にチェックを付けると、固定幅ファイル入力ステップをコピーしてデータ読み込みを分散処理できます。 【注意事項】 ファイルの内容を正しく読み込めないことがあるため、チェックを付けないでください。
9	[行区切り記号タイプ]	入力ファイルの改行コードのタイプを選択します。 • [行区切り記号なし] • [Unix (1 バイト)] • [MS-DOS (2 バイト)] [分散処理する]にチェックを付けると指定できます。
10	[文字コード]	入力ファイルの文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。
11	[ファイル名を結果に含む]	この項目にチェックを付けると、データ入力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [ファイル名を結果に含む]を[Y]にしたステップ名 • comment • timestamp
12	[フィールド名]	フィールド名を指定します。
13	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。
14	[書式]	入力書式を指定します。
15	[データ幅]	データの幅（単位：バイト）を指定します。
16	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。

項番	項目名	説明
17	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
18	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
19	[桁区切り文字]	「. (ドット)」など、小数点を表す文字を指定します。
20	[数値囲み文字]	「, (コンマ)」など、桁を区切る文字を指定します。
21	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。

2.2.15 [入力]行生成 (Generate rows)

ユーザーが定義した形式の静的データを複数行生成します。

生成するフィールドの値はユーザーが指定した定数です。データ変換のデバッグやテストなどによく使用されます。[プレビュー]ボタンを押すと、読み込むデータのプレビューができます。

図 2-40 [行生成]画面

表 2-41 [行生成]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

項番	項目名	説明
2	[リミット]	生成する行数を指定します。 [行の生成を停止しない]にチェックを付けていない場合に、指定できます。
3	[行の生成を停止しない]	この項目にチェックを付けると、データ変換の実行中にリアルタイムで行が生成されます。行生成を止めるには、データ変換の実行を停止してください。 チェックを付けると、[リミット]のテキストボックスが非活性になります。
4	[ミリ秒単位の間隔(遅延)]	リアルタイムで行が生成される間隔をミリ秒単位で指定します。 [行の生成を停止しない]にチェックを付けると指定できます。
5	[現在の行の時間フィールド名]	行が生成された時間を Date 型のフォーマットで出力するためのフィールド名を指定します。 [行の生成を停止しない]にチェックを付けると指定できます。
6	[前の行の時間フィールド名]	1 つ前の行が生成された時間を Date 型のフォーマットで出力するためのフィールド名を指定します。 [行の生成を停止しない]にチェックを付けると指定できます。
7	[名称]	生成するフィールド名を指定します。
8	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
9	[書式]	出力書式を指定します。 【注意事項】 [書式]に「  (通貨記号 (国際通貨記号))」を指定した場合は、異常終了します。
10	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
11	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
12	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
13	[小数] [※]	「. (ドット)」など、小数点を表す文字を指定します。
14	[グループ] [※]	「, (コンマ)」など、桁を区切る文字を指定します。
15	[値]	実データ情報を定義します。
16	[空文字を設定]	該当フィールドの値をすべて空文字にするかどうかを[Y]または[N]で指定します。

注※ 画面の項目名と指定する内容が異なります。

2.3 出力

- [出力]JSON output (JSON output)
- [出力]Microsoft Excel writer (Microsoft Excel writer)
- [出力]Pentaho レポート出力 (Pentaho reporting output)
- [出力]XML output (XML output)
- [出力]キューブ出力 (Serialize to file)
- [出力]テキストファイル出力 (Text file output)
- [出力]テーブル出力 (Table output)
- [出力]削除 (Delete)
- [出力]挿入/更新 (Insert / update)
- [出力]更新 (Update)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.3.1 [出力]JSON output (JSON output)

先行ステップからの入力値に基づいて、JSON データのブロックを生成します。出力方法は、次に示すどちらかまたは両方です。

- ステップ内でファイルを出力する
- 出力ストリームに JSON データを出力する

(1) [全般]タブ

JSON 出力に関する設定をします。

図 2-41 [JSON 出力]画面-[全般]タブ

表 2-42 [JSON 出力]画面-[全般]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[オペレーション]	オペレーションタイプを選択します。 • [値を出力] ステップ内でファイル出力をしないで、[値を出力]で指定した出力フィールドへ JSON データを出力します。 • [ファイルへ書き込み] ステップ内でファイル出力します。出力フィールドへは出力しません。 • [値を出力しファイルへ書き込む] ステップ内でファイル出力し、かつ[値を出力]で指定した出力フィールドへ JSON データを出力します。

項番	項目名	説明
3	[Json ブロック名]	出力する JSON データのブロック名を指定します。
4	[ブロック内の行数]	ブロック内の行数を指定します。 「0」を指定した場合、すべての要素が 1 つのデータブロックへ出力されます。
5	[値を出力]	出力するフィールド名を指定します。 [オペレーション]で[ファイルへ書き込み]以外の値を指定している場合に指定できます。
6	[互換モード]	この項目にチェックを付けると、互換性を保つために、PDI 4.3.0 より前の形式で出力します。 チェックを付けない場合、PDI 4.3.0 以降の形式で出力します。 例えば、「name=aaa, value=1」の入力が 2 行あった場合、次のように出力します。 - チェックを付けない場合（デフォルト） <pre>{ "data": [{ "name": "aaa", "value": 1 }, { "name": "aaa", "value": 1 }] }</pre> - チェックを付けた場合 <pre>{ "data": [{ "name": "aaa", "value": 1 }, { "name": "aaa", "value": 1 }] }</pre>
7	[ファイル名]	出力ファイルのパス（拡張子を除く）を指定します。出力ファイルの拡張子は[拡張子]で指定します。 ファイル名の指定には、変数も指定できます。 [オペレーション]で[値を出力]以外の値を指定している場合で、かつ空白の場合、エラーが発生します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
8	[追記]	この項目にチェックを付けると、既存ファイルがある場合にそのファイルに追記する形でデータを出力します。
9	[親フォルダーを作成]	この項目にチェックを付けると、[ファイル名]で指定したパスの中のディレクトリが存在しない場合でも、自動でディレクトリを作成します。 チェックを付けない場合、[ファイル名]で指定したパスの中のディレクトリが存在しないときに、エラーが発生します。
10	[開始時に作成しない]	この項目にチェックを付けると、入力データがない場合は、ファイルを作成しません。 チェックを付けない場合、入力データがなくてもファイルを作成するため、空のファイルが作成されることがあります。
11	[拡張子]	出力ファイルの拡張子を指定します。 [オペレーション]で[値を出力]以外の値を指定している場合に指定できます。
12	[エンコード]	出力ファイルの文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。 [オペレーション]で[値を出力]以外の値を指定している場合に指定できます。
13	[サブレットに出力]	この項目にチェックを付けると、出力をサブレットへ渡します。 [オペレーション]で[値を出力]以外の値を指定している場合に指定できます。
14	[ファイル名に日を含む]	この項目にチェックを付けると、データを出力したときの日付（形式：yyyyMMdd）をファイル名に追加します。

項番	項目名	説明
		[ファイル名表示]ボタンを押すと、出力ファイル名の例を表示します。
15	[ファイル名に時間を含む]	この項目にチェックを付けると、データを出力したときの時刻（形式：HHmmss）をファイル名に追加します。 [ファイル名表示]ボタンを押すと、出力ファイル名の例を表示します。
16	[結果ファイル名にファイルを追加]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 チェックを付けない場合、メモリ使用量を削減できます。

(2) [フィールド]タブ

入力されたフィールドを、出力する要素名（キー名）にマッピングするための情報を定義します。

[フィールド取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。要素名にはフィールド名と同名が使用されます。

図 2-42 [JSON 出力]画面-[フィールド]タブ

The screenshot shows a window titled 'JSON出力' (JSON Output) with a 'JSON output' step name. The 'Fields' tab is selected, showing a table with columns '#', 'フィールド名' (Field Name), and '要素名' (Element Name). The table has one row with the number '1' in the first column. Below the table is a 'フィールド取得' (Get Fields) button. At the bottom of the window are buttons for 'ヘルプ' (Help), 'OK(O)', and 'キャンセル(C)' (Cancel).

表 2-43 [JSON 出力]画面-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	入力されるフィールド名を指定します。
2	[要素名]	出力する要素名を指定します。

2.3.2 [出力]Microsoft Excel writer (Microsoft Excel writer)

Excel 形式のファイルにデータを書き込みます。同様の機能を持つステップとして Excel 出力ステップがありますが、Excel 出力ステップは、xlsx ファイルに対応していないため、Microsoft Excel writer ステップを使用することを推奨します。

(1) [ファイル&シート]タブ

出力する Excel ファイルとシートに関する設定をします。

図 2-43 [Microsoft Excel writer]画面-[ファイル&シート]タブ

The screenshot shows the 'Microsoft Excel writer' dialog box with the 'File & Sheet' tab selected. The 'File' section includes fields for 'File name' (file), 'Extension' (xls [Excel 97以降]), and checkboxes for 'Create parent folder', 'Stream XSLX data', 'Data separator rows', 'Include step number in file name', 'Include date in file name', 'Include time in file name', and 'Specify date time format'. There is also a 'Date time format' dropdown and a 'Show file name' button. The 'Output file already exists' dropdown is set to 'Replace with new output file'. The 'Create file on first row' checkbox is unchecked, and the 'Add file name to results' checkbox is checked. The 'Sheet' section includes a 'Sheet name' field (Sheet1), an 'Active sheet' checkbox (checked), an 'Output file sheet exists' dropdown (Replace with new sheet), a 'Protect sheet (XLS format only)' checkbox (unchecked), and fields for 'User name' and 'Password'. The 'Template' section includes checkboxes for 'Use template for new file creation' and 'Use template for new sheet creation', a 'Template file' field (template.xls), a 'Template sheet' field, and a 'Hide template sheet' checkbox. At the bottom are 'Help', 'OK(O)', and 'Cancel(C)' buttons.

表 2-44 [Microsoft Excel writer]画面-[ファイル&シート]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

項番	項目名	説明
2	[ファイル名]	出力ファイルのパス（拡張子を除く）を指定します。出力ファイルの拡張子は[拡張子]で指定します。 ファイル名の指定には、変数も使用できます。 空白の場合でもエラーは発生しませんが、ファイルは出力されません。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[拡張子]	出力ファイルの拡張子（xls またはxlsx）を選択します。
4	[Create parent folder]	この項目にチェックを付けると、[ファイル名]に指定したパスの中のディレクトリが存在しない場合でも、自動でディレクトリを作成します。 チェックを付けない場合、[ファイル名]に指定したパスの中のディレクトリが存在しないとエラーが発生します。
5	[ストリーム XSLX データ]	この項目にチェックを付けると、Streaming API を使用して、メモリ消費量を抑えながらデータ書き出しができます。この項目は、[拡張子]で[xlsx [Excel 2007 以降]]を選択した場合にだけ設定できます。
6	[データを区切る行数]	出力ファイルを分ける際のレコード数を指定します。0 を指定した場合、出力ファイルを分けません。
7	[ファイル名にステップ番号を含める]	データ変換内の各ステップにはユニークな番号が割り当てられています。この項目にチェックを付けると、テキストファイル出力ステップに割り当てられている番号をファイル名に追加します。
8	[ファイル名に日付を含める]	この項目にチェックを付けると、データを出力したときの日付（形式：yyyyMMdd）をファイル名に追加します。
9	[ファイル名に時間を含める]	この項目にチェックを付けると、データを出力したときの時刻（形式：HHmmss）をファイル名に追加します。
10	[デットタイムのフォーマットを指定する]	この項目にチェックを付けると、ファイル名に、データを出力したときの日付と時刻を任意の形式で追加します。 チェックを付けると、[ファイル名に日付を含める]および[ファイル名に時間を含める]の設定は無効になります。
11	[デットタイムフォーマット]	ファイル名に追加する日付と時刻の形式を選択します。
12	[ファイル名を表示]ボタン	このボタンをクリックすると、[出力ファイル]画面を表示します。
13	[出力ファイルが既に存在する場合は]	出力ファイルがすでに存在する場合に、既存ファイルを置き換えるか、既存ファイルに追記するかを選択します。
14	[最初の行を取得時にファイルを作成する]	この項目にチェックを付けると、入力データがない場合はファイルを作成しません。 チェックを付けない場合、入力データがなくてもファイルを作成するため、空のファイルが作成されることがあります。
15	[結果にファイル名を追加する]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。
16	[シート名]	データを出力するシートの名称を指定します。

項番	項目名	説明
17	[アクティブシートに設定]	この項目にチェックを付けると、ファイルを開いたときに[シート名]で指定したシートを表示します。
18	[出力ファイルにシートが存在する場合は]	[シート名]で指定したシートがすでに存在する場合に、既存シートを置き換えるか、既存シートに追記するかを指定します。 [出力ファイルが既に存在する場合は]に、[既存のファイルに書き込む]を設定している場合だけ有効です。
19	[シートを保護する（XLS フォーマットのみ）]	この項目にチェックを付けると、ユーザー名とパスワードで、[シート名]で指定したシートを保護します。 [ユーザー名] シートの保護に使用するユーザー名を指定します。 [パスワード] シートの保護に使用するパスワードを指定します。
20	[新規ファイル作成時にテンプレートを使用する]	この項目にチェックを付けると、新規ファイルを作成するときに、指定したテンプレートファイルをコピーします。 [テンプレートファイル] テンプレートとなるファイルのパスを指定します。 指定したファイルが存在しない場合は、エラーが発生します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
21	[新規シート作成時にテンプレートを使用する]	この項目にチェックを付けると、新規シートを作成するときに、指定したテンプレートシートをコピーします。 テンプレートとなるシートは、新規シートを作成する Excel ファイルです。 指定したシートが存在しない場合は、エラーが発生します。 [テンプレートシート] テンプレートとなるシートの名前を指定します。
22	[Hide Template Sheet]	この項目にチェックを付けると、出力ファイルにテンプレートシートを出力しません。ただし、データを書き出すシートとテンプレートシートが同一シートの場合を除きます。

(2) [コンテンツ]タブ

出力する Excel のコンテンツに関する設定をします。

図 2-44 [Microsoft Excel writer]画面-[コンテンツ]タブ

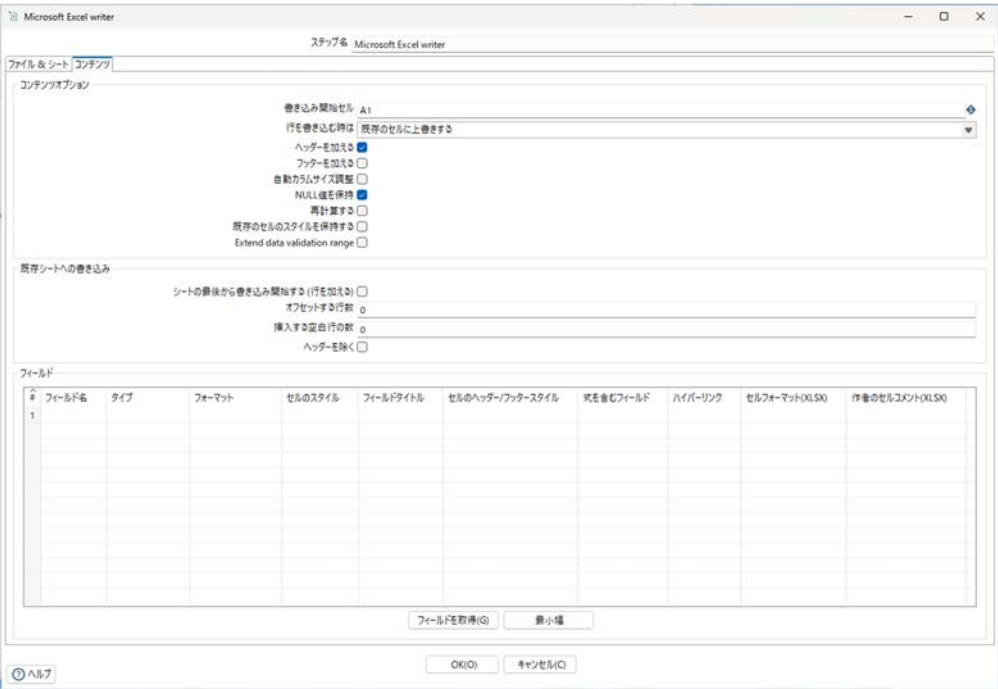


表 2-45 [Microsoft Excel writer]画面-[コンテンツ]タブの項目

項番	項目名	説明
1	[書き込み開始セル]	データの書き込みを開始するセルの番号を指定します。例えば、3 行目 B 列の場合、「B3」と指定します。
2	[行を書き込む時は]	データを書き込むセルにデータが記入されている場合に、データを上書きするか、既存セルを下に移動するかを指定します。 この項目は、[ファイル&シート]タブで次の設定をしている場合だけ有効です。 [出力ファイルが既に存在する場合は] 「既存のファイルに書き込む」 [出力ファイルにシートが存在する場合は] 「既存のシートに書き込む」
3	[ヘッダーを加える]	この項目にチェックを付けると、先頭のレコードにヘッダーレコードを出力します。
4	[フッターを加える]	この項目にチェックを付けると、末尾のレコードにフッターレコードを出力します。
5	[自動カラムサイズ調整]	この項目にチェックを付けると、データを書き込んだシートのカラムサイズを自動調節します。
6	[NULL 値を保持]	この項目にチェックを付けると、データの読み込み時にセル内の NULL 値を保持します。 チェックを付けない場合、NULL 値は空の文字列に置き換えられます。
7	[再計算する]	この項目にチェックを付けると、データを書き込んだ Excel ファイル内に数式が存在する場合に、データ書き込みが終了した時にその数式を再計算します。

項番	項目名	説明
8	[既存のセルのスタイルを保持する]	この項目にチェックを付けると、シート内で定義済みのスタイルを保持したまま、データを置き換えられます。
9	[Extend data validation range]	この項目にチェックを付けると、Excel でデータ検証を実行するときの検証範囲が拡張されて、書き込まれたデータの最終行まで検証します。 Excel シートの 1 行目から 10 行目までをデータ検証範囲として設定しているが、15 行目まで新たにデータが追加された場合 - データ検証範囲を拡張したい（15 行目まで検証したい）ときは、この項目にチェックを付ける - 既存のデータ検証範囲を利用したい（10 行目まで検証したい）ときは、この項目にチェックを付けない
10	[シートの最後から書き込み開始する（行を加える）]	この項目にチェックを付けると、シート内の最後の行の次からデータを書き込みます。 チェックを付けた場合、[書き込み開始セル]の行指定と、[行を書き込む時は]の設定が無効になります。
11	[オフセットする行数]	書き込み開始行を、指定した値の行数分だけ下にずらします。負の値も指定できます。
12	[挿入する空白行の数]	データ書き込みの前に、指定した行数分だけ空白行を挿入します。
13	[ヘッダーを除く]	この項目にチェックを付けると、フィールド名を表すヘッダー行を出力しません。
14	[フィールド名]	フィールド名を指定します。
15	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。
16	[フォーマット]	Excel でデータを表示する時のフォーマットを選択します。
17	[セルのスタイル]	該当フィールドのセルのスタイルを指定します。 特定のセル番号を指定することで、そのセルのスタイルをコピーします。例えば、「B3」と指定すると、B3 のセルのスタイルが適応されます。
18	[フィールドタイトル]	ヘッダーやフッターを出力する際のフィールド名を指定します。 PDI 上のフィールド名と Excel に出力したフィールド名を異なる名前にしたい場合に指定します。
19	[セルのヘッダー/フッタースタイル]	ヘッダーおよびフッターのセルのスタイルを指定します。指定方法は[セルのスタイル]と同じです。
20	[式を含むフィールド]	フィールドの値が Excel の数式を表すかどうかを指定します。Excel の数式を表す場合は、「Y」を指定します。 「Y」を指定した場合、フィールドの値を出力する際に「=」が自動で挿入されます。 【注意事項】 フィールドの値に「=」が含まれている場合、エラーが発生します。

項番	項目名	説明
21	[ハイパーリンク]	指定したフィールドの値（他のセル、http、ftp、email など）を利用して、ハイパーリンクを作成します。
22	[セルフォーマット(XLSX)]	xlsx ファイルでは、各セルにコメントが付与できます。指定したフィールドの値をコメントとして出力します。
23	[作者のセルコメント(XLSX)]	指定したフィールドの値を、出力したコメントの作者名として設定します。
24	[フィールドを取得]ボタン	このボタンを押すと、入力ストリームのフィールドを自動入力できます。
25	[最小幅]ボタン	—

(凡例) —：説明はありません。

2.3.3 [出力]Pentaho レポート出力 (Pentaho reporting output)

Pentaho レポートファイル (PRPT) をレンダリングし、選択した形式 (PDF、CSV、Excel など) でファイルを出力します。

図 2-45 [Pentaho レポーティングファイル出力]画面

表 2-46 [Pentaho レポートティングファイル出力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Report definition file name field]	Pentaho レポートデザイナーで作成した PRPT ファイルのパスが含まれるフィールドを指定します。
3	[Output file name field]	出力ファイルのパスが含まれるフィールドを指定します。 【注意事項】 パス情報を直接入力した場合、エラーになります。
4	[Accept file name from field?]	この項目にチェックを付けると、先行のステップから入力されたフィールド情報をレポート定義ファイルや出力ファイルとして使用します。
5	[レポート定義ファイル]	Pentaho レポートファイル (PRPT) のファイルパスを指定します。 [Accept file name from field?]にチェックを付けていない場合に、指定できます。
6	[出力ファイル]	出力ファイルのファイルパスを指定します。 [Accept file name from field?]にチェックを付けていない場合に、指定できます。
7	[Create Parent Folder]	この項目にチェックを付けると、[出力ファイル]で指定したパスの中のディレクトリが存在しない場合でも、自動でディレクトリを作成します。
8	[パラメーター名]	PRPT ファイルの「パラメーター」で定義されている名前を指定します。 [レポートパラメーター取得]ボタンを押し、テンプレートとして指定している PRPT ファイルを選択すると、PRPT ファイルで定義しているフィールドを自動入力できます。
9	[フィールド名]	「パラメーター」として渡す値の含まれるフィールドを指定します。 1 つまたは複数の入力フィールドの値をレポートのあらかじめ定義されたパラメータに渡せます。 String 配列にデータを渡すには、データをタブで区切ります。
10	[出力プロセッサ]	出力するファイルの形式を指定します。 出力できる形式は、[PDF]、[Paged HTML]、[Streaming HTML]、[CSV]、[Excel]、[Excel 2007]、[RTF]です。

2.3.4 [出力]XML output (XML output)

任意のソースから 1 つ以上の XML ファイルに出力します。

(1) [ファイル]タブ

出力ファイルに関する設定をします。[ファイル名の参照]ボタンを押すと、出力ファイルリストの一覧が表示されます。

図 2-46 [XML 出力]画面-[ファイル]タブ

XML出力

ステップ名XML output

ファイル全般フィールド

ファイル名file参照(B)...

開始時にファイルを作成しない

サブレットに出力を渡す

拡張子xml

ファイル名にステップ番号を含む

ファイル名に日付を含む

ファイル名に時刻を含む

日付の表示形式を指定する

日付表示形式

ファイル名の参照(S)

ファイル名を結果に含む

ヘルプ

OK(O)

キャンセル(C)

表 2-47 [XML 出力]画面-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ファイル名]	出力ファイルのパス（拡張子を除く）を指定します。出力ファイルの拡張子は[拡張子]で指定します。 ファイル名の指定には、変数も使用できます。 空白の場合、エラーが発生します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[開始時にファイルを作成しない]	この項目にチェックを付けると、入力データがない場合は、ファイルを作成しません。チェックを付けない場合、入力データがなくてもファイルを作成するため、空のファイルが作成されることがあります。
4	[サブレットに出力を渡す]	この項目にチェックを付けると、ファイルに出力する代わりに Java のサブレットへ渡します。
5	[拡張子]	出力ファイルの拡張子（xml）を指定します。
6	[ファイル名にステップ番号を含む]	データ変換内の各ステップにはユニークな番号が割り当てられています。この項目にチェックを付けると、XML output ステップに割り当てられている番号をファイル名に追加します。
7	[ファイル名に日付を含む]	この項目にチェックを付けると、データを出力したときの日付（形式：yyyyMMdd）をファイル名に追加します。
8	[ファイル名に時刻を含む]	この項目にチェックを付けると、データを出力したときの時刻（形式：HHmmss）をファイル名に追加します。
9	[日付の表示形式を指定する]	この項目にチェックを付けると、ファイル名に、データを出力したときの日付と時刻を任意の形式で追加します。 チェックを付けると、[ファイル名に日付を含む]および[ファイル名に時刻を含む]の設定は無効になります。

2. データ変換

Pentaho 機能説明 PDI 編

105

項番	項目名	説明
10	[日付表示形式]	日付表示形式をプルダウンから指定します。
11	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type 先行ジョブのファイル名 • filename 先行ジョブのファイルパス • path 先行ジョブのジョブ名 • parentorigin 先行ジョブのジョブ名 • origin [ファイル名を結果に含む]にチェックを付けたステップ名 • comment • timestamp

(2) [全般]タブ

読み込んだデータの出力に関する設定をします。

図 2-47 [XML 出力]画面-[全般]タブ

The screenshot shows the 'XML Output' dialog box with the 'General' tab selected. The 'Steps' field is set to 'XML output'. The 'File' tab is active, showing the following settings:

- 圧縮する (Compress): ☐
- 文字コード (Character Code): UTF-8
- 名前空間 (Namespace):
- 親XML要素名 (Parent XML Element Name): Rows
- 反復XML要素名 (Repeating XML Element Name): Row
- 指定レコード数でファイルを分割 (Split file by specified record count): 0
- XML出力からNULL値を除く (Exclude NULL values from XML output): ☐

Buttons at the bottom include 'ヘルプ' (Help), 'OK(O)', and 'キャンセル(C)' (Cancel).

表 2-48 [XML 出力]画面-[全般]タブの項目

項番	項目名	説明
1	[圧縮する]	この項目にチェックを付けると、出力された XML ファイルを ZIP 形式で圧縮します。

項番	項目名	説明
2	[文字コード]	出力ファイルの文字コードを指定します。 MS932 を指定する場合は、[windows-31j]を指定してください。
3	[名前空間]	XML ファイル内に Namespace を定義する場合、この項目にチェックを付けます。
4	[親 XML 要素名]	XML ファイル内のルート要素の名前を指定します。
5	[反復 XML 要素名]	XML ファイル内の行要素の名前を指定します。
6	[指定レコード数でファイルを分割]	1 つの XML ファイルに出力する最大行数を指定します。
7	[XML 出力から NULL 値を除く]	この項目にチェックを付けると、XML の出力データに NULL 値が含まれる場合に、その要素の出力を除外します。

(3) [フィールド]タブ

書き込むデータのフィールド情報や出力形式を定義します。

[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-48 [XML 出力]画面-[フィールド]タブ

表 2-49 [XML 出力]画面-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。
2	[要素名]	XML ファイル内の要素名を指定します。
3	[コンテンツタイプ]	コンテンツのタイプを、次の項目から指定します。 • [Element] 入力データを要素値として XML ファイルに出力します。 • [Attribute] 入力データを属性値として XML ファイルに出力します。
4	[データタイプ]	PDI 上のデータタイプを指定します。

項番	項目名	説明
		PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
5	[書式]	出力書式を指定します。
6	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
7	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
8	[通貨記号]	「\$（ドル）」など、通貨を表す記号を指定します。
9	[小数点]	「.（ドット）」など、小数点を表す文字を指定します。
10	[桁区切り文字]	「,（コンマ）」など、桁を区切る文字を指定します。
11	[NULL 可能]	フィールドの値が NULL の場合、この項目に指定した値をフィールドに挿入します。
12	[最小桁数にする]ボタン	－

(凡例) －：説明はありません。

2.3.5 [出力]キューブ出力 (Serialize to file)

入力データを cube 拡張子のバイナリ形式で出力します。

図 2-49 [キューブ出力]画面

表 2-50 [キューブ出力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ファイル名]	出力ファイルのパス（拡張子を含む）を指定します。

項番	項目名	説明
		[参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[ファイルを作成しない]	この項目にチェックを付けると、入力データがない場合は、ファイルが作成されません。 チェックを付けない場合、入力データがなくてもファイルを作成するため、空のファイルが作成されることがあります。
4	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。

2.3.6 [出力]テキストファイル出力（Text file output）

区切り文字で区切られたテキストファイル（CSV 形式）にデータを書き込みます。

(1) [ファイル]タブ

出力ファイルに関する設定をします。

[ファイル名の参照]ボタンを押すと、出力ファイルリストの一覧が表示されます。

図 2-50 [テキストファイル出力]画面-[ファイル]タブ

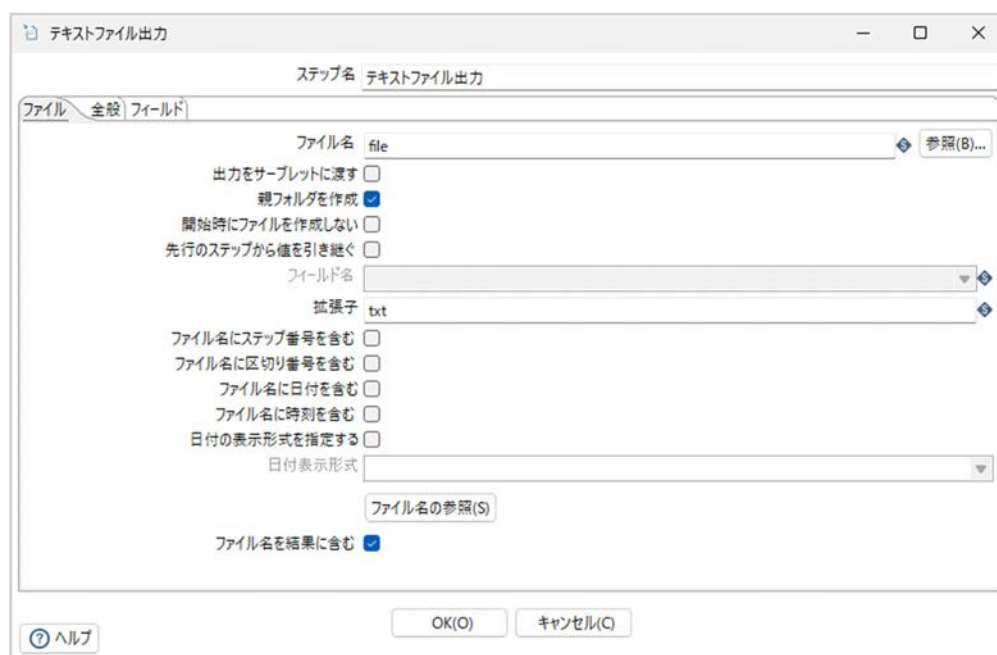


表 2-51 [テキストファイル出力]画面-[ファイル]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

項番	項目名	説明
2	[ファイル名]	出力ファイルのパス（拡張子を除く）を指定します。出力ファイルの拡張子は[拡張子]で指定します。 ファイル名の指定には、変数も使用できます。 空白の場合、エラーが発生します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[出力をサーブレットに渡す]	この項目にチェックを付けると、ファイルに出力する代わりに Java のサーブレットへ渡します。 また、REST サービスでデータを変換すると、出力ファイルをサーブレットからレスポンスとして取得できます。
4	[親フォルダを作成]	この項目にチェックを付けると、[ファイル名]で指定したパスの中のディレクトリがない場合に、そのディレクトリを自動作成します。 チェックを付けない場合、[ファイル名]で指定したパスの中のディレクトリがないときは、エラーが発生します。
5	[開始時にファイルを作成しない]	この項目にチェックを付けると、入力データがない場合は、ファイルが作成されません。 チェックを付けない場合、入力データがなくてもファイルを作成するため、空のファイルが作成されることがあります。
6	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値をファイルパスとして使用します。 フィールドの値として指定するファイルパスには、正規表現を使用できません。 [ファイル名]を使用したデータ書き込みはできません。
7	[フィールド名]	値がファイルパスとなっているフィールド名を指定します。
8	[拡張子]	出力ファイルの拡張子を指定します。
9	[ファイル名にステップ番号を含む]	データ変換内の各ステップにはユニークな番号が割り当てられています。この項目にチェックを付けると、テキストファイル出力ステップに割り当てられている番号をファイル名に追加します。
10	[ファイル名に区切り番号を含む]	この項目にチェックを付けると、パーティション機能を使用するとき、各パーティションのスキーマ名をファイル名に追加します。 パーティション機能については、Pentaho Documentation を参照してください。
11	[ファイル名に日付を含む]	この項目にチェックを付けると、データを出力したときの日付（形式：yyyyMMdd）をファイル名に追加します。
12	[ファイル名に時刻を含む]	この項目にチェックを付けると、データを出力したときの時刻（形式：HHmmss）をファイル名に追加します。
13	[日付の表示形式を指定する]	この項目にチェックを付けると、ファイル名に、データを出力したときの日付と時刻を任意の形式で追加します。 チェックを付けると、[ファイル名に日付を含む]および[ファイル名に時刻を含む]の設定は無効になります。
14	[日付表示形式]	日付表示形式をプルダウンから指定します。
15	[ファイル名の参照]ボタン	テキストファイル出力ステップで出力するファイル名を表示します。

項番	項目名	説明
16	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に使用するファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。

(2) [全般]タブ

読み込んだデータを出力するためのオプションです。

図 2-51 [テキストファイル出力]画面-[全般]タブ

表 2-52 [テキストファイル出力]画面-[全般]タブの項目

項番	項目名	説明
1	[既存のファイルに追加する]	この項目にチェックを付けると、既存ファイルがある場合に、そのファイルに追記する形でデータを出力します。
2	[区切り文字]	出力する区切り文字で区切られたテキストファイルの区切り文字を指定します。 [タブの挿入]ボタンを押すと、タブ文字を挿入できます。
3	[引用符]	出力する区切り文字で区切られたテキストファイルの引用符を指定します。
4	[フィールドを引用符で囲む]	この項目にチェックを付けると、各フィールドの値を引用符で囲んで出力します。
5	[囲みフィックス無効化]	デフォルト（この項目にチェックを付けない場合）では、フィールドの値に引用符が含まれているときに、自動的にそのフィールドの値が引用符で囲まれます。 この項目にチェックを付けると、フィールドの値は引用符で囲まれません。
6	[ヘッダー]	この項目にチェックを付けると、先頭のレコードにフィールド名を表すヘッダーレコードを出力します。

項番	項目名	説明
7	[フッター]	この項目にチェックを付けると、末尾のレコードにフィールド名を表すフッターレコードを出力します。
8	[フォーマット]	改行文字のフォーマットを選択します。 [No new-line terminator]を選択した場合、改行しません。
9	[圧縮形式]	出力ファイルを圧縮する形式を選択します。
10	[文字コード]	出力ファイルの文字コードを指定します。 MS932を指定する場合は、[windows-31j]を指定してください。
11	[フィールドの右側を空白で埋める]	この項目にチェックを付けると、フィールド定義で指定した長さに足りない場合、右側を空白で埋めます。
12	[ダンプ出力する]	この項目にチェックを付けると、書式、長さおよび精度の設定を出力形式に反映しない代わりに、出力性能が向上します。
13	[指定されたレコード数でファイルを分ける]	指定したレコード数で出力ファイルを分けます。 なお、0を指定した場合、出力ファイルを分けません。
14	[終了レコードを追加する]	設定した文字列を出力データの末尾に追加します。

(3) [フィールド]タブ

書き込むデータのフィールド情報や出力形式を定義します。

[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-52 [テキストファイル出力]画面-[フィールド]タブ

表 2-53 [テキストファイル出力]画面-[フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	フィールド名を指定します。

項番	項目名	説明
2	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
3	[書式]	出力書式を指定します。
4	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
5	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
6	[通貨記号]	「\$（ドル）」など、通貨を表す記号を指定します。
7	[桁区切り文字]*	「.（ドット）」など、小数点を表す文字を指定します。
8	[数値囲み文字]*	「,（コンマ）」など、桁を区切る文字を指定します。
9	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
10	[NULL 可能]	フィールドの値が NULL の場合、この項目に指定した値をフィールドに挿入します。

注※ 画面の項目名と指定する内容が異なります。

2.3.7 [出力]テーブル出力 (Table output)

データベースのテーブルに追加 (INSERT) し、データを書き込みます。追加と更新が必要な場合は、Insert ステップまたは Update ステップを使用してください。[SQL] ボタンを押すと、入力ストリームのデータ定義に合わせた CREATE TABLE 文を自動生成します。

図 2-53 [テーブル出力]画面

表 2-54 [テーブル出力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	データソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。[参照]ボタンを押すと、データベース内のスキーマ名を参照できます。
4	[テーブル名]	テーブル名を指定します。[参照]ボタンを押すと、データベース内のテーブル名を参照できます。

項番	項目名	説明
5	[コミットサイズ]	[全般]タブの[挿入および更新]にチェックを付けている場合に、バッチ処理でデータを書き込む際のコミットサイズを行数で指定します。
6	[開始時にテーブル内容を削除する]	この項目にチェックを付けると、データ変換の開始時にテーブル内容を削除する SQL 文を発行します。データソースの接続タイプによって、ここで発行される SQL 文が異なります。 なお、[全般]タブの[テーブルパーティションを生成する]にチェックを付けている場合、この項目は無効です。
7	[挿入エラーを無視する]	この項目にチェックを付けると、INSERT 時にデータベースでエラーが発生した場合に、PDI はエラーを無視して処理を継続します。 なお、[全般]タブの[挿入および更新]にチェックを付けている場合、この項目は無効です。
8	[列名を指定する]	この項目にチェックを付けると、入力ストリームからテーブルへ出力するフィールドを指定できます。また、入力ストリームのフィールドと出力するテーブルカラムを対応付けられるため、それらの名前が異なるときでも入力データを出力できます。 チェックを付けない場合、入力ストリームのフィールド名と、出力するテーブルのカラム名が同じである必要があります。

(1) [全般]タブ

テーブル出力に関する設定をします。

図 2-54 [テーブル出力]画面-[全般]タブ

テーブル出力

ステップ名

テーブル出力

データソース名

編集(E)

新規作成(N)...

ウィザード

スキーマ名

参照(B)...

テーブル名

参照(B)...

コミットサイズ

1000

開始時にテーブル内容を削除する

☐

挿入エラーを無視する

☐

列名を指定する

☐

全般

列名

テーブルパーティションを生成する

☐

フィールド名

月単位

☒

日単位

☐

挿入および更新

☒

テーブル名を定義したフィールドを含む

☐

フィールド名

フィールドを保存する

☒

自動索引キーを生成する

☐

自動索引キーフィールド名

ヘルプ

OK(O)

キャンセル(C)

SQL(S)

表 2-55 [テーブル出力]画面-[全般]タブの項目

項番	項目名	説明
1	[テーブルパーティションを生成する]	この項目にチェックを付けると、キーとなるフィールドの日付によって、入力データを複数テーブルに分けて出力します。 [テーブル名]で指定したテーブルが「TableName」の場合、キー値に応じて、「TableName_yyyyMM」（月単位）または「TableName_yyyyMMdd」（日単位）というテーブルに出力されます。 [テーブル名]で指定したテーブルが「testTable」、キーとなるフィールドの日付が「2010 年 4 月 1 日」の場合の出力例を次に示します。 - 月単位の場合の出力例 testTable_201004 - 日単位の場合の出力例 testTable_20100401 なお、出力用の各テーブルは事前に作成する必要があります。
2	[フィールド名]	データ出力先振り分けのキーとなる日付型のフィールドを指定します。

項番	項目名	説明
3	[月単位]	この項目を選択すると、データ振り分けの単位を月単位にします。 [テーブルパーティションを生成する]にチェックを付けている場合、この項目か、[日単位]のどちらかを選択します。
4	[日単位]	この項目を選択すると、データ振り分けの単位を日単位にします。 [テーブルパーティションを生成する]にチェックを付けている場合、この項目か、[日単位]のどちらかを選択します。
5	[挿入および更新]※	この項目にチェックを付けると、挿入をバッチ処理します。 この項目を使用するには、次の条件を満たす必要があります。 • [コミットサイズ]が0以上である • [自動索引キーを生成する]のチェックを外す • [データ変換プロパティ]-[その他]タブ-[データ変換でデータベーストランザクション処理する]のチェックを外す • データソースが batch update に対応している 【注意事項】 この項目は条件を満たした上でチェックを付けてください。条件を満たしていない場合、性能が著しく低下します。
6	[テーブル名を定義したフィールドを含む]	この項目にチェックを付けると、フィールドの値によって、指定したテーブルに入力レコードを出力します。レコードごとに出力テーブルを分けたいときに使用します。
7	[フィールド名]	出力テーブルを指定するフィールド名です。
8	[フィールドを保存する]	この項目にチェックを付けると、[フィールド名]で指定したフィールドをテーブルに出力します。
9	[自動索引キーを生成する]	この項目にチェックを付けると、入力データの各レコードに対し、ユニークなキーを自動生成して付与します。
10	[自動索引キーフィールド名]	キーフィールド名を指定します。

注※ 画面の項目名と実際の機能が異なります。

(2) [列名]タブ

入力ストリームからテーブルへ出力するフィールドを設定します。また、入力ストリームのフィールドと出力するテーブルカラムを対応付けをします。

[illegible]

項番	項目名	説明
1	[列名]	出力するテーブルの列名を指定します。
2	[フィールド名]	入力ストリームからテーブルへ出力するフィールドを指定します。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。 [フィールドマッピング]ボタンを押すと、フィールドマッピング専用のダイアログで入力ストリームのフィールドと出力するテーブルカラムを対応付けできます。

データベースのテーブルに対し、入力ストリームおよび出力先のテーブルからキーとなる列を選択し、指定した条件に応じて入力行をテーブルから削除（DELETE）します。

削除ステップでは、ステップへの入力行ごとに次の処理を実行します。

- 1. 指定した条件に応じた DELETE 文を発行する。この時、データベースからの応答が 0 件の場合はエラーで処理を停止し、1 件以上の場合は 2.の処理に遷移する。
- 2. 次の入力行があれば、1.から処理を繰り返す。次の入力が無ければ、処理を終了する。

[フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-56 [削除]画面

削除

ステップ名

削除

データソース名

編集(E)

新規作成(N)...

ウィザード

スキーマ名

参照(B)...

テーブル名

lookup table

参照(B)...

コミットサイズ

100

参照キー

#	列名	比較	フィールド名1	フィールド名2
1				

フィールドの取得(G)

ヘルプ

OK(O)

キャンセル(C)

表 2-57 [削除]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	出力テーブルのデータソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。[参照]ボタンを押すと、データベース内のスキーマ名を参照できます。
4	[テーブル名]	テーブル名を指定します。[参照]ボタンを押すと、データベース内のテーブル名を参照できます。

項番	項目名	説明
5	[コミットサイズ]	バッチ処理でデータを書き込む際のコミットサイズを 0 以上の整数で指定します。 ここで指定する値は、削除 (DELETE) の発行回数です。 0 を指定した場合、各行の処理がオートコミットされます。
6	[参照キー]	入力ストリームおよび出力先のテーブルからキーとなる列を選択し、処理の判断基準となる条件を指定します。
7	[列名]	比較の対象となる列名を指定します。
8	[比較]	列同士を比較する際の比較演算子を指定します。 比較演算子の動作は、接続するデータベースごとに異なります。
9	[フィールド名 1]	入力ストリームのフィールド名を指定します。 引数が必要な演算子を指定した場合だけフィールドを指定し、それ以外の場合は空欄にします。 演算子によって必要な引数が異なり、指定するフィールド数に過不足があった場合はエラーが発生します。
10	[フィールド名 2]	入力ストリームのフィールド名を指定します。 引数が必要な演算子を指定した場合だけフィールドを指定し、それ以外の場合は空欄にします。 演算子によって必要な引数が異なり、指定するフィールド数に過不足があった場合はエラーが発生します。

2.3.9 [出力]挿入/更新 (Insert / update)

データベースのテーブルに対し、入力ストリームおよび出力先のテーブルからキーとなる列を選択し、指定した条件に応じて入力行をテーブルに挿入 (INSERT) または更新 (UPDATE) をします。

挿入/更新ステップでは、ステップへの入力行ごとに次の処理を実行します。

1. 入力ストリームおよび出力先のテーブルからキーとなる列について、指定した条件に応じた SELECT 文を発行する。この時、データベースからの応答が 0 件の場合は 2. の処理に、1 件以上の場合は 3. の処理に遷移する。
2. 入力行をデータベースのテーブルへ挿入し、5. の処理へ遷移する。
3. データベースから選択したすべての行について、入力行と各カラムの値を比較する。この時、値が異なる行が 1 行でもあれば 4. の処理に、1 行もなければ 5. の処理に遷移する。
4. データベースから選択したすべての行に対し、入力行で更新し、5. の処理に遷移する。
5. 次の入力行があれば、1. から処理を繰り返す。次の入力がない場合は、処理を終了する。

図 2-57 [挿入/更新]画面

表 2-58 [挿入/更新]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	出力テーブルのデータソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。[参照]ボタンを押すと、データベース内のスキーマ名を参照できます。
4	[テーブル名]	挿入/更新先のテーブル名を指定します。[参照]ボタンを押すと、データベース内のテーブル名を参照できます。
5	[コミットサイズ]	コミットするレコード数を指定します。

項番	項目名	説明
6	[更新しない]	この項目にチェックを付けると、更新しないで挿入だけ実行します。
7	[フィールド (参照)]	[フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
8	[列名]	データベースのデータの参照キーとなるフィールドです。
9	[比較演算子]	参照条件となる比較演算子を指定します。
10	[フィールド名 1]	比較演算の第一引数です。
11	[フィールド名 2]	比較演算の第二引数です。
12	[フィールド (更新)]	[フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
13	[列名]	更新/挿入先の列名です。
14	[フィールド名]	データを参照するフィールドです。
15	[更新する]	列ごとに更新する/しないを指定します。
16	[フィールドマッピング]ボタン	[フィールドマッピング]ボタンを押すと、[マッピング入力]画面を表示します。この画面で、入力ストリームのフィールドと、データベースのカラムのマッピングをします。
17	[SQL]ボタン	SQL を生成します。

2.3.10 [出力]更新 (Update)

データベースのテーブルに対し、入力ストリームおよび出力先のテーブルからキーとなる列を選択し、指定した条件に応じて入力行をテーブルに更新 (UPDATE) します。

図 2-58 [更新]画面

更新

ステップ名

更新

データソース名

編集(E)

新規作成(N)...

ウィザード

スキーマ名

参照(B)...

テーブル名

lookup table

参照(B)...

コミットサイズ

100

batchアップデートを使用する

ルックアップをスキップする

更新エラーを無視する

フィールド名 (更新キー)

フィールド (参照)

#	列名	比較演算子	フィールド名1	フィールド名2
1				

フィールドの取得(G)

フィールド (更新)

#	列名	フィールド名
1		

更新フィールドの取得(U)

ヘルプ

OK(O)

キャンセル(C)

SQL(S)

表 2-59 [更新]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	出力テーブルのデータソースを指定します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。[参照]ボタンを押すと、データベース内のスキーマ名を参照できます。

項番	項目名	説明
4	[テーブル名]	更新先のテーブル名を指定します。[参照]ボタンを押すと、データベース内のテーブル名を参照できます。
5	[コミットサイズ]	バッチ処理でデータを書き込む際のコミットサイズを 0 以上の整数で指定します。 ここで指定する値は、更新（UPDATE）の発行回数です。 0 を指定した場合、各行の処理がオートコミットされます。
6	[batch アップデートを使用する]	－
7	[ルックアップをスキップする]	－
8	[更新エラーを無視する]	この項目にチェックを付けると、ルックアップ処理で条件を満たすレコードがテーブルにない場合でもエラーが発生せず、処理が継続されます。
9	[フィールド名(更新キー)]	フィールド名を入力すると、データベースからの応答が 0 件の場合は「N」、1 件以上の場合は「Y」が出力されます。 指定しない場合、何も出力されません。
10	[フィールド(参照)]	入力ストリームおよび出力先のテーブルからキーとなる列を選択し、処理の判断基準となる条件を指定します。 [フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
11	[列名]	出力先のテーブルの列名を指定します。
12	[比較演算子]	列同士を比較する際の演算子を指定します。
13	[フィールド名 1]	入力ストリームのフィールド名を指定します。 引数が必要な演算子を指定した場合だけフィールドを指定し、それ以外の場合は空欄にします。 演算子によって必要な引数が異なり、指定するフィールド数に過不足があった場合はエラーが発生します。
14	[フィールド名 2]	入力ストリームのフィールド名を指定します。 引数が必要な演算子を指定した場合だけフィールドを指定し、それ以外の場合は空欄にします。 演算子によって必要な引数が異なり、指定するフィールド数に過不足があった場合はエラーが発生します。
15	[フィールド（更新）]	更新対象のフィールドを選択します。 [更新フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
16	[列名]	出力先のテーブルの列名を指定します。
17	[フィールド名]	入力ストリームのフィールド名を指定します。

(凡例) －：説明はありません。

2.4 変換

- [変換]Add a checksum (Add a checksum)
- [変換]シーケンス追加 (Add sequence)
- [変換]シーケンス追加 (変化値) (Add value fields changing sequence)
- [変換]フィールド値設定 (Set field value)
- [変換]フィールド分割 (Split fields)
- [変換]フィールド分割 (行) (Split field to rows)
- [変換]フィールド定数設定 (Set field value to a constant)
- [変換]フィールド結合 (Concat fields)
- [変換]値マッピング (Value mapper)
- [変換]定数追加 (Add constants)
- [変換]数値範囲 (Number range)
- [変換]文字列オペレーション (String operations)
- [変換]文字列カット (Strings cut)
- [変換]文字列置換 (Replace in string)
- [変換]行フラット化 (Row Flattener)
- [変換]行整列 (Sort rows)
- [変換]行正規化 (Row normaliser)
- [変換]行非正規化 (Row denormaliser)
- [変換]計算 (Calculator)
- [変換]選択/名前変更 (Select values)
- [変換]重複行削除 (Unique rows)
- [変換]重複行削除(HashSet) (Unique rows (HashSet))

上記以外のステップについては、Pentaho Documentation を参照してください。

2.4.1 [変換]Add a checksum (Add a checksum)

入力ストリームのフィールドのチェックサムを計算し、その結果を出力します。

[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-59 [チェックサム追加]画面

チェックサム追加

ステップ名

Add a checksum

タイプ

CRC 32

結果タイプ

Hexadecimal

フィールド (結果)

Field separator string

互換性モード

☐

Evaluation method

Byte Representation of fields (default behavior 8.1 forward)

フィールド

#	フィールド
1	

ヘルプ

OK(O)

フィールドを取得(G)

キャンセル(C)

表 2-60 [チェックサム追加]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[タイプ]	チェックサムのタイプを、次の項目から指定します。 • [CRC32] • [ADLER 32] • [MD5] • [SHA-1] • [SHA-256]
3	[結果タイプ]	出力結果タイプを選択します。 • [String] • [Hexadecimal] • [Binary] [タイプ]で[MD5]、[SHA-1]、[SHA-256]のどれかを選択した場合に選択できます。[String]を選択した場合、入力（各アルゴリズムのチェックサム計算後の byte 配列）を文字列に変換して連結した結果が出力されます。
4	[フィールド (結果)]	チェックサムのフィールド名を指定します。
5	[Field separator string]	入力データの区切り文字を指定します。指定することで異なるチェックサム結果が得られます。 例えば、データグリッドで次の値が入力されている場合、 メタ : data1,data2 データ#1 : data1:abc data2:def データ#2 : data1:abcde data2:f

項番	項目名	説明
		データ#3 : data1:abcd data2:ef 「[, (コンマ)」を指定したとき データ#1 が「abc,def」、データ#2 が「abcde,f」、データ#3 が「abcd,ef」として認識されてチェックサムが計算されます。 - 何も指定しないとき データ#1、データ#2、データ#3 ともに「abcdef」として認識されてチェックサムが計算されます。
6	[互換性モード]	この項目にチェックを付けると、前バージョンと互換性のある hexadecimal 出力を有効にします。 この場合、入力（各アルゴリズムのチェックサム計算後の byte 配列）を 16 進数に変換し、文字列連結した結果が出力されます。 [タイプ]で[MD5]、[SHA-1]、[SHA-256]のどれかを選択し、かつ[結果タイプ]で[Hexadecimal]を選択した場合に指定できます。
7	[Evaluation method]	評価方式を選択します。 [Byte Representation of fields (default behavior 8.1 forward)] [Pentaho String representation of fields using format masks (7.1 and below behavior)] [Native String representation of fields (8.0 behavior)]
8	[フィールド]	入カストリームのフィールドを指定します。

2.4.2 [変換]シーケンス追加 (Add sequence)

一連の整数（シーケンス）を新たなフィールドに出力します。負の値も入力できます。

出力例を次に示します。

例

シーケンス追加前

年	Date
2015	Date1
2015	Date2
2016	Date3

シーケンス追加後

年	Date	カウンタ
2015	Date1	1
2015	Date2	2
2016	Date3	3

図 2-60 [シーケンス追加]画面

表 2-61 [シーケンス追加]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[シーケンスカウンタ名]	シーケンスを出力するフィールドの名前を指定します。
3	[シーケンスカウンタを取得する]	この項目にチェックを付けると、データベースを使用してシーケンスカウンタを永続的に保存します。
4	[データソース名]	データソース名を指定します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
5	[スキーマ名]	スキーマ名を指定します。[スキーマ]ボタンを押すと、データベース内のスキーマ名を参照できます。
6	[シーケンス名]	シーケンス名を指定します。[シーケンス]ボタンを押すと、データベース内のシーケンス名を参照できます。
7	[内部カウンタを使用する]	シーケンスカウンタを永続的に保存する必要がない場合に、この項目にチェックを付けます。

項番	項目名	説明
8	[カウンタ名(オプション)]	使用するカウンタの名前を指定します。 データ変換内の他のシーケンス追加ステップで同名のステップカウンタを使う場合、異なる名前を指定してください。同じ名前の場合、競合して正しい出力にならないおそれがあります。 この項目の設定を推奨します。
9	[開始値]	シーケンス番号の開始値を指定します。 整数以外を入力するとエラーが発生します。
10	[増分]	シーケンス番号の増分を指定します。 負の値も入力できます。 整数以外を入力するとエラーが発生します。
11	[最大値]	シーケンス番号の最大値を指定します。 カウンタの値が最大値を超えた場合は、開始値に戻ります。 最大値を超えた場合のカウンタの値の例を次に示します。 [開始値] : 「-1」、[増分] : 「2」、[最大値] : 「1」を設定した場合 カウンタの値 : -1,1,-1,1,-1,1,...

2.4.3 [変換]シーケンス追加（変化値）（Add value fields changing sequence）

一連の整数（シーケンス）を新たなフィールドに出力します。

カウント中にキーとフィールドの値が変化した時にシーケンスカウンタが開始値に戻ります。[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

カウント中に「年」フィールドの値が変化した場合の出力例を次に示します。

例

シーケンス追加前

年	Date
2015	Date1
2015	Date2
2016	Date3

シーケンス追加後

年	Date	カウンタ
2015	Date1	1
2015	Date2	2
2016	Date3	1

図 2-61 [シーケンス追加 (変化値)]画面

シーケンス追加 (変化値)

ステップ名

シーケンス追加 (変化値) |

結果フィールド

result

開始値

1

増分

1

以下のフィールドの値が変更されるとInitシーケンスです。

#

フィールド名

1

ヘルプ

OK(O)

フィールドを取得(G)

キャンセル(C)

表 2-62 [シーケンス追加 (変化値)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[結果フィールド]	シーケンスを出力するフィールドの名前を指定します。
3	[開始値]	シーケンス番号の開始値を指定します。 整数以外を入力した場合は、値を 1 と見なします。
4	[増分]	シーケンス番号の増分を指定します。 整数以外を入力した場合は、値を 1 と見なします。
5	[フィールド名]	カウント中にこのフィールドの値が変化するとシーケンスカウンタが開始値に戻ります。 値の変化でカウンタがリセットされるため、フィールドの値の出現回数をカウントする場合などはフィールドの値はソートされている必要があります。

2.4.4 [変換]フィールド値設定 (Set field value)

フィールドの値を別のフィールドの値に置き換えます。フィールド間のデータ型が異なる場合、エラーが発生します。

[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-62 [フィールド値設定]画面

フィールド値設定

ステップ名:

フィールド値設定

フィールド:

#	フィールド名	フィールドからの値による置換
1		

ヘルプ

OK(O)

フィールドを取得(G)

キャンセル(C)

表 2-63 [フィールド値設定]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	置き換え前のフィールド名を指定します。
3	[フィールドからの値による置換]	置き換え後の値を持つフィールドを指定します。

2.4.5 [変換]フィールド分割 (Split fields)

指定文字で区切られた 1 フィールドを複数のフィールドに展開して出力します。

出力例を次に示します。

例 1

展開前

年	Data
2015	Data1, Data2, Data3

展開後

年	Data	D1	D2	D3
2015	Data1, Data2, Data3	Data1	Data2	Data3

例 2

展開前

年	Data
2014	D1:Data1, D3:Data3
2015	D1:Data1, D2:Data2, D3:Data3
2016	D2:Data2, D3:Data3

展開後

年	Data	D1	D2	D3
2014	D1:Data1, D3:Data3	Data1	<null>	Data3
2015	D1:Data1, D2:Data2, D3:Data3	Data1	Data2	Data3
2016	D2:Data2, D3:Data3	<null>	Data2	Data3

図 2-63 [フィールド分割]画面

表 2-64 [フィールド分割]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[分割するフィールド名]	分割するフィールドの名前を指定します。
3	[フィールド区切り文字]	分割するフィールドのデータの区切り文字を指定します。
4	[囲い文字]	囲み文字を指定します。 囲み文字で囲まれた文字列内に区切り文字に指定した文字がある場合でも、区切り文字として扱わずに処理します。
5	[フィールド名 (新規)]	展開先のフィールド名を指定します。
6	[ID]	フィールド名に ID を設定して、分割したデータの出力先フィールドを指定する場合に、ID とする文字列を入力します。分割後のデータに対して、この項目で指定した文字列で検索して一致した場合に、その ID に対応するフィールドにデータを出力します。 ID を指定する場合は、すべてのフィールドに設定してください。すべてのフィールドに設定されていない場合、エラーが発生します。
7	[ID を除去する]	展開先のフィールドで設定した ID を除去する場合に指定します。
8	[データタイプ]	データタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
9	[長さ]	データ出力時の長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。
10	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。 数値以外を入力した場合に[OK]ボタンを押すと、自動的にデフォルトの値に戻ります。

項番	項目名	説明
11	[書式]	出力書式を指定します。
12	[数値囲み文字]※	「, (コンマ)」など、桁を区切る文字を指定します。
13	[桁区切り文字]※	「. (ドット)」など、小数点を表す文字を指定します。
14	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
15	[NULL 可能]	フィールドに展開した値が設定値と一致した場合、NULL に変換します。 出力データタイプを String にした場合、NULL ではなく空白文字になります。
16	[デフォルト]	NULL の場合に入力される値です。
17	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。

注※ 画面の項目名と指定する内容が異なります。

2.4.6 [変換]フィールド分割（行）(Split field to rows)

指定文字で区切られた 1 フィールドを複数の行に展開して出力します。

フィールド区切り文字で区切られた「Data」フィールドを複数行に展開した出力例を次に示します。

例

展開前

年	Data
2015	Data1, Data2, Data3

展開後

年	Data	Data_Split
2015	Data1, Data2, Data3	Data1
2015	Data1, Data2, Data3	Data2
2015	Data1, Data2, Data3	Data3

図 2-64 [フィールド分割 (行)]画面

表 2-65 [フィールド分割 (行)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[分割するフィールド名]	分割するフィールドの名前を指定します。 例では、「Data」に該当します。
3	[フィールド区切り文字]	分割するフィールドのデータの区切り文字を指定します。 例では、「, (コンマ)」に該当します。
4	[区切り文字は正規表現]	この項目にチェックを付けると、[フィールド区切り文字]の指定に正規表現を使用できます。
5	[フィールド名(新規)]	展開先のフィールド名を指定します。 例では、展開後の「Data_Split」に該当します。
6	[レコード番号を出力する]	この項目にチェックを付けると、レコード番号を出力します。
7	[フィールド名]	レコード番号を出力するフィールドの名前を指定します。
8	[レコード番号を初期化する]	この項目にチェックを付けると、展開元のレコードが変わった時にレコード番号を初期化します。 【注意事項】 レコード番号を出力する[フィールド名]と展開先の[フィールド名(新規)]は異なる名前にしてください。同じ名前の場合、この項目にチェックを付けても初期化されないおそれがあります。

2.4.7 [変換]フィールド定数設定 (Set field value to a constant)

フィールドの値を、定義した定数に置き換えます。

[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-65 [定数設定]画面

定数設定

ステップ名

フィールド定数設定

定数に変数を使用

フィールド

#	フィールド	置換	変換マスク (日付)	空文字を設定?

ヘルプ

OK(O)

フィールドを取得(G)

キャンセル(C)

表 2-66 [定数設定]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[定数に変数を使用]	この項目にチェックを付けると、変数の値に置き換えます。
3	[フィールド]	値を置き換えたいフィールドを指定します。
4	[置換]	置き換え後の値を指定します。
5	[変換マスク(日付)]	フィールドが Date 型や Timestamp 型の場合、[置換]で指定する値の日付マスクを指定します。
6	[空文字を設定?]	String 型の場合、[Y]を指定すると空文字が設定されます。[N]を指定すると NULL が設定されます。

2.4.8 [変換]フィールド結合 (Concat fields)

指定した複数のフィールドの値を指定した書式で文字列に変換し、区切り文字で区切って文字列結合して出力します。

出力例を次に示します。

例

フィールド結合前

Data1	Data2	Data3
2017	09	26
2017	09	27
2017	09	28
2017	09	29
2017	09	30

フィールド結合後

```
Data1/2/3
2017/09/26
2017/09/27
2017/09/28
2017/09/29
2017/09/30
```

(1) [フィールド]タブ

読み込むデータのフィールド情報を定義します。

[フィールドを取得]ボタンを押すと、入力ストリームからフィールドを自動入力できます。

図 2-66 [フィールド結合]画面-[フィールド]タブ

フィールド結合

ステップ名: フィールド結合

ターゲットフィールド名:

ターゲットフィールドの長さ: 0

区切り文字: ; TABを挿入

囲い文字: *

#	名前	タイプ	フォーマット	長さ	精度	通貨	小数	グループ	Trimタイプ	Null
1										

フィールドを取得(G) 最小値

ヘルプ OK(O) キャンセル(C)

表 2-67 [フィールド結合]画面-[フィールド]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ターゲットフィールド名]	結合結果を出力するフィールド名を指定します。
3	[ターゲットフィールドの長さ]	ターゲットフィールドのデータの長さを指定します。[アドバンスド]タブの[速いデータダンプ(フォーマットなし)]にチェックを付けた場合に使用されます。 ターゲットフィールドの最大の長さを指定すると、バッファの再割り当て防止による性能劣化を防止できます。 【注意事項】 この項目が未入力の場合、ターゲットフィールドの長さは0となり、後続のステップのデータが出力されません。そのため、この項目で適切な値を設定するか、後続の処理で適切な長さを指定してください。
4	[区切り文字]	ターゲットフィールドの値に対し、結合したフィールドを区切る文字を指定します。

項番	項目名	説明
		[TAB を挿入]ボタンを押すと、タブ文字を入力できます。
5	[囲い文字]	結合するフィールドに区切り文字が含まれている場合、該当フィールドの前後に指定した囲み文字を付けます。デフォルトは「" (ダブルクォーテーション)」です。 【注意事項】 [アドバンスド]タブの[速いデータダンプ(フォーマットなし)]または[囲い文字の修正を無効に設定]にチェックを付けた場合、この設定は無効になります。
6	[名前]	結合対象のフィールドの名前を指定します。
7	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
8	[フォーマット]	出力書式を指定します。
9	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
10	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
11	[通貨]	「\$ (ドル)」など、通貨を表す記号を指定します。
12	[小数]	「. (ドット)」など、小数点を表す文字を指定します。
13	[グループ]*	「, (コンマ)」など、桁を区切る文字を指定します。
14	[Trim タイプ]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
15	[Null]	フィールドの値が NULL の場合は、この文字列をテキストファイルに挿入します。
16	[最小幅]ボタン	—

(凡例) —：説明はありません。

注※ 画面の項目名と指定する内容が異なります。

(2) [アドバンスド]タブ

読み込むフィールドのフォーマットを設定します。

図 2-67 [フィールド結合]画面-[アドバンスド]タブ

フィールド結合

ステップ名フィールド結合

ターゲットフィールド名

ターゲットフィールドの長さ0

区切り文字 ;TABを挿入

囲い文字 *

フィールドアドバンスド

選択したフィールドを削除する

フィールドを囲い文字で囲む

囲い文字の修正を無効に設定

ヘッダー

フッター

エンコード

右パディングフィールド

速いデータダンプ (フォーマットなし)

分割行数0

最終行の後にエンドラインを追加

ヘルプ

OK(O)

キャンセル(C)

表 2-68 [フィールド結合]画面-[アドバンスド]タブの項目

項番	項目名	説明
1	[選択したフィールドを削除する]	この項目にチェックを付けると、[フィールド]タブで選択したすべてのフィールドが出力ストリームから削除され、結合結果の格納されたターゲットフィールドだけが出力されます。
2	[フィールドを囲い文字で囲む]	この項目にチェックを付けると、すべてのフィールド名を[囲い文字]で指定した文字で囲みます。
3	[囲い文字の修正を無効に設定]	フィールドの値に引用符が含まれている場合、自動的にそのフィールドの値が引用符で囲まれますが、この項目にチェックを付けると、その機能が無効になります。
4	[ヘッダー]	この項目にチェックを付けると、1行目にヘッダー行が出力されます。 この行が出力されると、ヘッダー行のターゲットフィールド名以外の出力ストリームフィールドには、NULL が設定されます。
5	[フッター]	この項目にチェックを付けると、最終行にフッター行が出力されます。 この行が出力されると、フッター行のターゲットフィールド名以外の出力ストリームフィールドには、NULL が設定されます。
6	[エンコード]	文字コードを選択します。 MS932 を指定する場合は、[windows-31j]を指定してください。
7	[右パディングフィールド]	この項目にチェックを付けると、指定した長さになるまで、フィールドの末尾に空白を追加、または末尾の文字を削除します。

項番	項目名	説明
		【注意事項】 チェックを付けた場合でも、[ターゲットフィールドの長さ]に合わせて右パディングはされません。
8	[速いデータダンプ(フォーマットなし)]	この項目にチェックを付けると、書式、長さおよび精度の設定を出力形式に反映しない代わりに、出力性能が向上します。
9	[分割行数]	フッターを出力する行数を指定します。 【注意事項】 [フッター]にチェックを付けた場合、[分割行数]にフッター分も含めて出力されます。 100 を指定した場合、レコード 99 ごとにフッターが出力されます。ただし、1 を指定した場合は 1 レコードごとにフッターが出力されます。
10	[最終行の後にエンドラインを追加]	設定した文字列を出力データの末尾に追加します。 この行が出力されると、エンドライン行のターゲットフィールド名以外の出力ストリームフィールドには、NULL が設定されます。

2.4.9 [変換]値マッピング (Value mapper)

指定したフィールドの特定の値を、ほかの値に置き換えて出力します。出力例を次に示します。

例

「Code」フィールドの国名コードを国名に置き換えて、「Name」フィールドに出力します。

入力データ

```
Code
JPN
USA
FRA
```

出力データ

```
Code  Name
JPN   Japan
USA   U. S. A
FRA   France
```

図 2-68 [値マッピング]画面

値マッピング

ステップ名 Value mapper

フィールド名

出力フィールド名※空の場合は上書き

合致しない場合の初期値

値

#	入力値	出力値
1		

ヘルプ OK(⓪) キャンセル(⓪)

表 2-69 [値マッピング]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	置き換える前の値を持つフィールドを指定します。
3	[出力フィールド名※空の場合は上書き]	値を置き換えた結果の出力先フィールド名を指定します。 指定しない場合、[フィールド名]で指定したフィールドの値が上書きされます。
4	[合致しない場合の初期値]	置き換える対象に指定した値と合致しない場合に置き換える値を指定します。 指定しない場合、置き換えられません。
5	[入力値]	値を置き換える対象の値を指定します。[出力値]とセットで指定してください。[入力値][出力値]のセットは、複数指定できます。
6	[出力値]	置き換える値を指定します。[入力値]とセットで指定してください。[入力値][出力値]のセットは、複数指定できます。

2.4.10 [変換]定数追加 (Add constants)

入力ストリームのすべてのレコードに対して、同一の定数値が設定されたフィールドを追加して出力します。

図 2-69 [定数追加]画面

定数追加

ステップ名 Add constants

フィールド

#	フィールド名	データタイプ	書式	長さ	精度	通貨記号	桁区切り文字	数値固み文字	値	Set empty string?
1										

ヘルプ OK(⓪) キャンセル(⓪)

表 2-70 [定数追加]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	追加する定数のフィールド名を指定します。
3	[データタイプ]	追加する定数のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。 【注意事項】 Internet Address 型の定数を追加しようとする、エラーが発生します。
4	[書式]	追加する定数の入力書式を指定します。
5	[長さ]	追加する定数の長さを指定します。
6	[精度]	追加する定数の精度を指定します。
7	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
8	[桁区切り文字]*	「. (ドット)」など、小数点を表す文字を指定します。
9	[数値囲み文字]*	「, (コンマ)」など、桁を区切る文字を指定します。 【注意事項】 「. (ピリオド)」を指定しても有効になりません。
10	[値]	定数値を指定します。
11	[Set empty string?]	定数に空文字を設定するかどうかを[Y]または[N]で指定します。[Y]を指定しても、NULL にはなりません。

注※ 画面の項目名と指定する内容が異なります。

2.4.11 [変換]数値範囲 (Number range)

指定したフィールドの値の数値範囲に応じた値を、新たなフィールドに出力します。

出力例を次に示します。

例

「Val」フィールドが 0 以上 1 未満の場合「small」、1 以上の場合「big」を「Result」に出力します。

入力データ

Key	Val
1	0.3
2	0.8
3	1.5

出力データ

Key	Val	Result
1	0.3	small
2	0.8	small
3	1.5	big

図 2-70 [Number range]画面

#	下限値	上限値	値
1		5.0	Less than 5
2	5.0	10.0	5-10
3	10.0		More than 10

表 2-71 [Number range]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド (入力)]	出力値を判定するフィールドを指定します。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
3	[フィールド (出力)]	結果を出力するフィールド名を指定します。 結果は文字列データとして出力されます。
4	[デフォルト]	[フィールド (入力)]で指定したフィールドの値が[範囲]で指定した数値範囲外の場合に、結果として出力する文字列を指定します。
5	[範囲]	数値範囲を[下限値]と[上限値]に指定します。 指定した範囲に値が含まれていた場合、[値]で指定した値を出力します。 範囲は、重複していても指定できます。複数の範囲に値が含まれる場合、[値]で指定した値を「,」で区切って連結した文字列を出力します。
6	[下限値]	数値範囲の下限値を指定します。 空白の場合、下限値は設定されません。 指定した値は、範囲内に含まれます。
7	[上限値]	数値範囲の上限値を指定します。 空白の場合、上限値は設定されません。

項番	項目名	説明
		指定した値は、範囲内に含まれません。
8	[値]	値が範囲内に含まれている場合に出力する文字列を指定します。

2.4.12 [変換]文字列オペレーション (String operations)

入力文字列を実行フィールドの値に従って加工します。

この時、実行フィールドの左側にあるものから順番に処理が行われます。

図 2-71 [文字列オペレーション]画面



表 2-72 [文字列オペレーション]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[入力ストリームフィールド]	変換対象のフィールドを指定します。
3	[出力ストリームフィールド]	変換の結果を出力するフィールドを指定します。
4	[トリムタイプ]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。
5	[小文字/大文字]	入力された文字列を小文字にするか、大文字にするかを指定します。 全角英数字にも対応しています。 [none] 変換しません。 [小文字] 入力された文字列をすべて小文字にします。 [大文字] 入力された文字列をすべて大文字にします。

項番	項目名	説明
6	[パディング]	文字列を[パッドの長さ]で指定した長さまで[パッド文字]で埋めるかどうかを指定します。 パディングする場合、[パッドの長さ]と[パッド文字]の指定が必要です。 • [なし] パディングしません。 • [左] 文字列の左を文字で埋めます。 • [右] 文字列の右を文字で埋めます。
7	[パッド文字]	パディングで埋める文字です。
8	[パッドの長さ]	パディングする長さです。
9	[InitCap]	単語の先頭の文字を大文字、その他の文字を小文字に変換するかどうかを[Y]または[N]で指定します。
10	[エスケープ]	各コンテンツ（XML、SQL、HTML）でそのまま使用できない文字をエスケープ文字（エンティティ参照による表記）に変換するかどうかを指定します。 • [無し] 変換しません。 • [XML エスケープ] XML で使用できない文字をエスケープ文字に変換します。 • [CDATA を使用] [CDATA を使用]する場合に指定します。CDATA の場合、エスケープ文字を使う必要がないため変換しません。「&apos;」などもそのままの文字で扱われます。 • [XML アンエスケープ] XML でエスケープ文字を変換前の文字に戻します。 • [SQL エスケープ] SQL で使用できない文字をエスケープ文字に変換します。 • [HTML エスケープ] HTML で使用できない文字をエスケープ文字に変換します。 • [HTML アンエスケープ] HTML でエスケープ文字を変換前の文字に戻します。
11	[Digits]	文字列を数字だけに変換するか、または数字以外に変換するかどうかを指定します。 • [none] 変換しません。 • [only] 数字だけに変換します。 • [remove] 数字以外に変換します。
12	[特殊文字を削除]	選択した特殊文字を削除するかどうかを指定します。 • [none] 削除しません。

項番	項目名	説明
		• [carriage return(CR)] CR を削除します。 • [line feed(LF)] LF を削除します。 • [carriage return & line feed] CR+LF を削除します。 • [horizontal tab] タブを削除します。 • [space] スペースを削除します。

2.4.13 [変換]文字列カット (Strings cut)

始点と終点を指定して、文字列をカットします。

全角文字も 1 文字として扱います。空白、タブ、CR、および LF は 1 文字、CR+LF は 2 文字として計算します。

例えば、文字列「abcdefg」のとき、[始点]＝「1」、[終点]＝「4」を指定した場合は、[始点]が「1」の文字は「b」、[終点]が「4」の文字は「e」なので、カットされる文字は「bcd」です。切り出す文字列の長さは「[終点]-[始点]」となり、「4-1」で「3」になります。終점에指定した文字はカットする文字に含まれません。

図 2-72 [文字列カット]画面

表 2-73 [文字列カット]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド(入力)]	カットする文字列を持つフィールドの名前を指定します。
3	[フィールド(出力)]	カットした文字列を出力するフィールドの名前を指定します。 指定しない場合、元のフィールドの名前になります。

項番	項目名	説明
4	[始点]	始点の文字番号です。0 以上を指定します。 負の値を入力すると入力文字列をそのまま出力します。
5	[終点]	終点の文字番号です。1 以上を指定します。 終点の番号が始点の番号以下の場合、出力は NULL になります。 負の値を入力すると入力文字列をそのまま出力します。

2.4.14 [変換]文字列置換 (Replace in string)

入力文字列から指定文字列を検索し、指定した文字に置換して出力します。[フィールド取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-73 [文字列置換]画面

表 2-74 [文字列置換]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[入力フィールド]	文字列置換するフィールドの名前を指定します。 【注意事項】 XML 文字 (<&文字列>) を指定した場合、プレビューでは「&」が除去されて表示されますが、テキストファイル出力ステップで出力されたデータには、「&」が出力されます。
3	[出力フィールド]	置換した文字列を出力するフィールドの名前を指定します。 【注意事項】 XML 文字 (<&文字列>) を指定した場合、プレビューでは「&」が除去されて表示されますが、テキストファイル出力ステップで出力されたデータには、「&」が出力されます。
4	[正規表現]	検索時に、[検索]で指定した文字列を正規表現として処理するかどうかを[Y]または[N]で指定します。
5	[検索]	検索文字列を指定します。
6	[置換]	置換文字列を指定します。 置換の優先度は次の順です。 1. [空文字を設定] 2. [フィールド置き換え]

項番	項目名	説明
		3. [置換] 【注意事項】 [正規表現]に[Y]を指定した場合、[置換]に「¥t」を指定しても「t」に置換されます。タブに置換されません。
7	[空文字を設定]	検索条件に一致した文字列を空文字に置換するかどうかを[Y]または[N]で指定します。 置換の優先度は次の順です。 1. [空文字を設定] 2. [フィールド置き換え] 3. [置換]
8	[フィールド置き換え]	選択したフィールドの値を置換文字列とします。 置換の優先度は次の順です。 1. [空文字を設定] 2. [フィールド置き換え] 3. [置換]
9	[全文字列]	文字列がフィールドの値と完全一致した時だけ置換するかどうかを[Y]または[N]で指定します。
10	[大文字小文字の区別]	検索時に、検索文字列の大文字と小文字を区別するかどうかを[Y]または[N]で指定します。全角文字にも対応しています。 文字列：「abcdef123」、[検索]：「ABC」、[置換]：「xyz」の場合の例を次に示します。 [Y]を指定した場合 大文字と小文字を区別するため、文字列は置換されません。 [N]を指定した場合 大文字と小文字を区別しないため、文字列は「xyzdef123」に置換されます。
11	[Is Unicode]	正規表現を使用する際に、Pattern.UNICODE_CHARACTER_CLASS オプション※を使用するかどうかを[Y]または[N]で指定します。 例えば、半角数字だけがマッチする正規表現で検索した場合でも、この機能を[Y]にすると、全角数字もマッチします。 [正規表現]：[Y]、[検索]：「¥d」の場合の例を次に示します。 [Y]を指定した場合 半角数字と全角数字がマッチします。 [N]を指定した場合 半角数字だけマッチします。 注※ UNICODE_CHARACTER_CLASS オプションを使用すると、検索文字列に正規表現を使用した場合に、POSIX 文字クラスを使用して検索します。 使用できる正規表現については、次の URL の「次の定義済の文字クラスと POSIX 文字クラスは」を参照ください。

項番	項目名	説明
		https://docs.oracle.com/javase/jp/8/docs/api/java/util/regex/Pattern.html

2.4.15 [変換]行フラット化 (Row Flattener)

一定数の連続した入力データ（フィールド）を 1 行に構造化します。

図 2-74 [行フラット化]画面

表 2-75 [行フラット化]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名 (入力)]	一定数の連続した入力データが格納されているフィールド名を指定します。
3	[フィールド名]	出力するフィールド名を指定します。

2.4.16 [変換]行整列 (Sort rows)

指定したフィールドの値をソートします。

指定したレコード数（デフォルトは 100 万）まではインメモリで処理しますが、上限を超えた場合、超えた分のレコードを一時ファイルに保存してソートします。

図 2-75 [行整列]画面

行整列

ステップ名 行整列

一時ディレクトリ %%java.io.tmpdir%%参照(8)...

一時ファイルの接頭子 out

並替えのサイズ (レコード数) 1000000

空きメモリのしきい値 (%)

一時ファイルを圧縮する☐

ユニークな行を対象にする☐

フィールド

#	フィールド名	昇順で並替え	大文字と小文字を区別する	Sort based on current locale?	Collator Strength	事前ソート
1						

ヘルプ

OK(O)

キャンセル(C)

フィールドを取得(G)

表 2-76 [行整列]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[一時ディレクトリ]	一時ファイルを保存するディレクトリを指定します。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。 【注意事項】 エラーが発生した場合、一時ファイルが削除されずにディスク上に残り続けるおそれがあります。
3	[一時ファイルの接頭子]	一時ファイルの接頭辞を指定します。
4	[並替えのサイズ(レコード数)]	インメモリで処理するレコード数の上限です。 このレコード数を超える場合、一時ファイルに保存されます。
5	[空きメモリのしきい値(%)]	この項目に指定した値(%)よりも空きメモリが少ない場合、データが一時的に保存されます。 この設定を有効にしたい場合、[並び替えのサイズ(レコード数)]を空欄にしてください。 【注意事項】 JVM では正確な空きメモリが把握できないため、正確な動作は期待できません。
6	[一時ファイルを圧縮する]	この項目にチェックを付けると、一時ファイルを圧縮します。
7	[ユニークな行を対象にする]	この項目にチェックを付けると、ソートキーの値が重複した場合、初めに入力されたレコードだけを出力します。
8	[フィールド名]	ソート対象のフィールド名 (ソートキー) を指定します。ソートキーは複数指定できます。フィールド番号[#]が「1」のフィールドからソート、N 番目のフィールドは (N-1) 番目までのフィールドのソート結果で等しかった行だけをソートします。
9	[昇順で並び替え]	昇順で並び替えるかどうかを[Y]または[N]で指定します。 [Y]を指定した場合、昇順でソートされます。[N]を指定した場合、降順でソートされず。

項番	項目名	説明
10	[大文字と小文字を区別する]	大文字と小文字を区別するかどうかを[Y]または[N]で指定します。
11	[sort based on current locale?]	ロケールの設定に基づいてソートするかどうかを[Y]または[N]で指定します。 ローカル実行の場合は、Spoon の[推奨言語 (1 次言語)]の設定に基づいてソートします。 例えば、日本語環境の場合、[N]を指定すると、「カ、が、ガ、か、あ、あ」は「あ、あ、か、が、カ、ガ」とソートされますが、[Y]を指定して、[Collator Strength]で[2]を指定すると、「あ、あ、か、カ、が、ガ」とソートされます。
12	[Collator Strength]	コレータの強さを指定します。 この項目は Java の Collator クラス*で設定できるレベルを 4 段階で指定します。 Pentaho の設定値と Java の Collator クラスの強度の対応は次のとおりです。 • Pentaho の設定値：[0] Java の Collator クラスの強度：「PRIMARY」 • Pentaho の設定値：[1] Java の Collator クラスの強度：「SECONDARY」 • Pentaho の設定値：[2] Java の Collator クラスの強度：「TERTIARY」 • Pentaho の設定値：[3] Java の Collator クラスの強度：「IDENTICAL」 注※ Collator クラスの詳細については、「 https://docs.oracle.com/javase/jp/8/docs/api/java/text/Collator.html 」を参照してください。
13	[事前ソート]	該当のフィールドがソートされているかどうかを[Y]または[N]で指定します。 [Y]を指定した場合、該当のフィールドはソートせずに出力します。 該当のフィールドがすでにソートされている場合、処理コストを抑えるため、[Y]を指定してください。

2.4.17 [変換]行正規化 (Row normaliser)

指定した行の正規化（ピボット）をします。[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

指定した行を正規化した出力例を次に示します。

例

正規化前

Month	ProductA	ProductB	ProductC
2003/01	10	5	17
2003/02	12	7	19

正規化後

Month	Product	sales
2003/01	A	10

2003/01	B	5
2003/01	C	17
2003/02	A	12
2003/02	B	7
2003/02	C	19

図 2-76 [行正規化]画面

表 2-77 [行正規化]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名(タイプ)]	新たなタイプとして定義するフィールド名を指定します。 例では、正規化後の「Product」に該当します。
3	[フィールド名]	タイプとして入力するフィールド名です。正規化するフィールドの名前を指定します。 例では、正規化前の「ProductA」「ProductB」「ProductC」に該当します。
4	[データタイプ]	タイプとして入力する値を指定します。 例では、正規化後の「A」「B」「C」に該当します。
5	[フィールド名(新規)]	タイプの値として入力するフィールド名を指定します。 例では、正規化後の「sales」に該当します。

2.4.18 [変換]行非正規化 (Row denormaliser)

指定した行の非正規化（アンピボット）をします。

行正規化ステップで行われる変換の逆の処理を実行します。

指定した行を非正規化した出力例を次に示します。

例

非正規化前

Month	Product	sales
2003/01	A	10
2003/01	B	5
2003/01	C	17

2003/02	A	12
2003/02	B	7
2003/02	C	19

非正規化後

Month	ProductA	ProductB	ProductC
2003/01	10	5	17
2003/02	12	7	19

図 2-77 [行非正規化]画面

表 2-78 [行非正規化]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名(キー値)]	キーとなるフィールドの名前を指定します。 例では、非正規化前の「Product」に該当します。
3	[フィールド(グループ)]	グループキーを設定します。 [フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
4	[フィールド名]	グループ化するフィールドの名前を指定します。 例では、「Month」に該当します。
5	[フィールド(出力先)]	出力先キーを設定します。 [フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
6	[フィールド名(出力先)]	出力するフィールドの名前を指定します。 例では、非正規化後の「ProductA」、「ProductB」、「ProductC」に該当します。
7	[フィールド名(参照元)]	値を参照するフィールドの名前を指定します。 例では、非正規化前の「sales」に該当します。
8	[キー値]	キーフィールドから参照元の値を参照するためのキーを指定します。 例では、非正規化前の「A」「B」「C」に該当します。
9	[データタイプ]	データタイプを選択します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。

項番	項目名	説明
		【注意事項】 Internet Address 型を指定すると、エラーが発生します。
10	[書式]	出力書式を指定します。
11	[長さ]	データ出力時の長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
12	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
13	[数値囲み文字]※	[「, (コンマ)」] など、桁を区切る文字を指定します。
14	[桁区切り文字]※	[「. (ドット)」] など、小数点を表す文字を指定します。
15	[通貨記号]	[「\$ (ドル)」] など、通貨を表す記号を指定します。
16	[NULL 可能]	フィールドに展開した値が設定値と一致した場合、NULL に変換します。 出力データタイプを String にした場合、NULL ではなく空白文字になります。
17	[演算子]	同一グループ内に同一キーを持つレコードがある場合の演算処理を指定します。 デフォルトは[-]で、最後に参照した値を出力します。 • [-] 最後のレコードの値を出力します。 • [合計] 値の合計を出力します。 • [平均] 値の平均を出力します。 • [最小] 最小値を出力します。 • [最大] 最大値を出力します。 • [Number of Values] 値の数を出力します。 • [(コンマ)で分けられた文字列を結合] 文字列を連結して出力します。

注※ 画面の項目名と指定する内容が異なります。

2.4.19 [変換]計算 (Calculator)

選択された入力フィールドの値を計算式に従って計算し、出力フィールドに出力します。

図 2-78 [計算]画面

計算処理

ステップ名

[計算]

Throw an error on non existing files

☒

フィールド

#	フィールド名	計算式	フィールドA	フィールドB	フィールドC	データタイプ	長さ	精度	除去する	書式	桁区切り文字	数値囲み文字	通貨記号
1													

ヘルプ

OK(O) キャンセル(Q)

表 2-79 [計算]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Throw an error on non existing files]	この項目にチェックを付けると、ファイルパスを引数に持つ計算式で、指定されたファイルが見つからない場合にエラーが発生します。
3	[フィールド名]	出力するフィールドの名前を指定します。
4	[計算式]	計算式を指定します。 計算式の種類については、「 計算式の詳細 」を参照してください。
5	[フィールド A]	引数として使用するフィールドです。
6	[フィールド B]	引数として使用するフィールドです。
7	[フィールド C]	引数として使用するフィールドです。
8	[データタイプ]	出力データのデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
9	[長さ]	データ出力時の長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
10	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
11	[除去する]	出力フィールドをアウトプットストリームに出力するかどうかを[Y]または[N]で指定します。
12	[書式]	出力書式を指定します。
13	[桁区切り文字]*	[. (ドット)] など、データ出力時の小数点を表す文字を指定します。
14	[数値囲み文字]*	[, (コンマ)] など、データ出力時の桁を区切る文字を指定します。
15	[通貨記号]	[\$ (ドル)] など、データ出力時の通貨を表す記号を指定します。

注※ 画面の項目名と指定する内容が異なります。

表 2-80 計算式の詳細

項番	計算式	説明
1	-	—
2	フィールドに一定値 A を設定	フィールド A に指定した値を設定します。 フィールド A にはフィールド名ではなく、実際に設定する値を指定します。
3	フィールド A のコピーを作成	フィールド A の値をコピーします。
4	$A + B$	フィールド A の値とフィールド B の値を加算します。
5	$A - B$	フィールド A の値からフィールド B の値を減算します。
6	$A * B$	フィールド A の値にフィールド B の値を乗算します。
7	A / B	フィールド A の値をフィールド B の値で除算します。
8	$A * A$	フィールド A の値を 2 乗します。
9	$\text{SQRT}(A)$	フィールド A の値の平方根を計算します。
10	$100 * A / B$	フィールド A の値に 100 を乗算します。求めた値をフィールド B の値で除算します。
11	$A - (A * B / 100)$	フィールド A の値とフィールド B の値を乗算した値を 100 で除算します。求めた値をフィールド A の値から減算します。
12	$A + (A * B / 100)$	フィールド A の値とフィールド B の値を乗算した値を 100 で除算します。求めた値をフィールド A の値に加算します。
13	$A + B * C$	フィールド B とフィールド C の値を乗算します。求めた値をフィールド A の値に加算します。
14	$\text{SQRT}(A * A + B * B)$	フィールド A の値の 2 乗+フィールド B の値の 2 乗の平方根を計算します。
15	$\text{ROUND}(A)$	フィールド A の値に最も近い整数を計算します。
16	$\text{ROUND}(A, B)$	フィールド A の値を正の無限大に近づくように丸めます。 フィールド B の値は小数部の桁数です。
17	$\text{STDROUND}(A)$	フィールド A の値を四捨五入します。
18	$\text{STDROUND}(A, B)$	フィールド A の値を四捨五入します。 フィールド B の値は小数部の桁数です。
19	$\text{CEIL}(A)$	フィールド A の値を正の無限大に近づくように丸めた整数を計算します。
20	$\text{FLOOR}(A)$	フィールド A の値を負の無限大に近づくように丸めた整数を計算します。
21	$\text{NVL}(A, B)$	フィールド A の値が NULL ではない場合、フィールド A の値を返し、フィールド A の値が NULL の場合、フィールド B の値を返します。
22	日付 A + B 日数	フィールド A の値にフィールド B の日数分加算します。
23	日付 A の年	フィールド A の値の年を計算します。
24	日付 A の月	フィールド A の値の月を計算します。

項番	計算式	説明
25	日付 A の年ごとの日数	フィールド A の値が、その年の何日目かを計算します。
26	日付 A の月ごとの日数	フィールド A の値が、その月の何日目かを計算します。
27	日付 A の週ごとの日数	フィールド A の値が、その週の何日目かを計算します。
28	日付 A で指定した年の週ごと	フィールド A の値が、その年の何週目かを計算します。
29	ISO8601 日付 A の年ごとの週数	ISO8601 形式で、フィールド A の値がその年の何週目かを計算します。
30	ISO8601 日付 A の年	ISO8601 形式で、フィールド A の値の年を計算します。
31	文字列 A のバイト数の hex(16 進) エンコード変換	フィールド A の値を 1 バイトずつ 16 進数でエンコードします。
32	文字列 A のバイト数の hex(16 進) デコード変換	フィールド A の値を 16 進数でデコードします。 「文字列 A のバイト数の hex(16 進)エンコード変換」のエンコードが前提です。
33	文字列 A のキャラクタ文字の hex(16 進)エンコード変換	フィールド A の値を 1 文字ずつ 16 進数でエンコードします。
34	hex(16 進)ストリング A のデコード (キャラクタ)	フィールド A の値を 16 進数でデコードします。 「文字列 A のキャラクタ文字の hex(16 進)エンコード変換」のエンコードが前提です。
35	CRC-32 を使ったファイル A の checksum	CRC-32 を使用してフィールド A で指定したファイルのチェックサムを計算します。
36	Adler-32 を使ったファイル A の checksum	Adler-32 を使用してフィールド A で指定したファイルのチェックサムを計算します。
37	MD5 を使ったファイル A の checksum	MD5 を使用してフィールド A で指定したファイルのチェックサムを計算します。
38	SHA-1 を使ったファイル A の checksum	SHA-1 を使用してフィールド A で指定したファイルのチェックサムを計算します。
39	レーベンシュタイン距離 (編集距離) (ソース A とターゲット B)	フィールド A で指定した値とフィールド B で指定した値のレーベンシュタイン距離を計算します。
40	A の metaphone(phonetics)	フィールド A の値の Metaphone を計算します。
41	A の Double metaphone(phonetics)	フィールド A の値の Double Metaphone を計算します。
42	絶対値 ABS(A)	フィールド A の値の絶対値を計算します。
43	日付 A から時間を除去	フィールド A の値の時刻部分を削除します。
44	日付 A - 日付 B (日付の違い)	フィールド A の値とフィールド B の日付の差 (日数) を計算します。
45	A + B + C	フィールド A の値とフィールド B の値とフィールド C の値を加算します。
46	文字列 A (大文字) 内の各単語の最初の文字	フィールド A の値の各単語の先頭の文字を大文字、各単語の先頭以外の文字を小文字に変換します。

項番	計算式	説明
47	文字列 A を大文字変換	フィールド A の値をすべて大文字にします（アルファベットだけ）。
48	文字列 A を小文字変換	フィールド A の値をすべて小文字にします（アルファベットだけ）。
49	文字列 A の XML コンテンツをマスク	フィールド A の値で、XML では使用できない文字をエンティティ参照による表記に変換します。
50	文字列 A の XML コンテンツ (CDATA) で保護	フィールド A の値を XML の CDATA セクションに変換します。
51	文字列 A から CR を除去	フィールド A の値から制御文字 CR を除去します。
52	文字列 A から LF を除去	フィールド A の値から制御文字 LF を除去します。
53	文字列 A から CRLF を除去	フィールド A の値から制御文字 CRLF を除去します。
54	文字列 A から TAB を除去	フィールド A の値から制御文字 TAB を除去します。
55	文字列 A から桁数だけ返す	フィールド A の値から数字以外を除去します。
56	文字列 A から数字を除去	フィールド A の値から数字を除去します。
57	文字列 A の長さを返す	フィールド A の値の長さを計算します。
58	バイナリファイルの内容をロード	フィールド A で指定したファイルをバイナリデータ型にロードします。
59	日付 A に時間 B を追加	フィールド A の値にフィールド B の値を追加します。
60	日付 A の四半期	フィールド A の値の四半期を計算します。
61	文字列 A の変数置換	フィールド A の値に含まれる変数を対応する値に変換します。
62	XML コンテンツをエスケープしない	フィールド A の値のエンティティ参照による表記を XML のコンテンツに変換します。
63	HTML コンテンツをエスケープ	フィールド A の値で、HTML では使用できない文字をエンティティ参照による表記に変換します。
64	HTML コンテンツをエスケープしない	フィールド A の値のエンティティ参照による表記を HTML のコンテンツに変換します。
65	SQL コンテンツをエスケープ	フィールド A の値に含まれる SQL のシングルクォートをエスケープします。
66	日付 A - 日付 B (平日)	フィールド A の値とフィールド B の日付の差（日数）を計算します。平日（月曜日～金曜日）だけを計上します。
67	日付 A + 月数 B	フィールド A の値にフィールド B の値を加算します。フィールド B の値は月として加算されます。
68	XML ファイル A が正しく構成されているか確認	フィールド A で指定したファイルが XML として正しく構成されているか確認します。
69	XML 文字列 A が正しく構成されているか確認	フィールド A の値が XML として正しく構成されているか確認します。
70	ファイル A をエンコード	フィールド A で指定したファイルのエンコード方式を計算します。
71	文字列 A と文字列 B の DamerauLevenshtein 距離	フィールド A の値とフィールド B の値の DamerauLevenshtein 距離を計算します。

項番	計算式	説明
72	文字列 A と文字列 B の NeedlemanWunsch 距離	フィールド A の値とフィールド B の値の NeedlemanWunsch 距離を計算します。
73	文字列 A と文字列 B の Jaro similitude	フィールド A の値とフィールド B の値の Jaro similitude を計算します。
74	文字列 A と文字列 B の JaroWinkler similitude	フィールド A の値とフィールド B の値の JaroWinkler similitude を計算します。
75	文字列 A の SoundEx	フィールド A の値の発音を SoundEx を使用して計算します。
76	文字列 A の RefinedSoundEx	フィールド A の値の発音を RefinedSoundEx を使用して計算します。
77	日付 A + B 時間	フィールド A の値にフィールド B の値を加算します。フィールド B の値は時間として加算されます。
78	日付 A + B 分	フィールド A の値にフィールド B の値を加算します。フィールド B の値は分として加算されます。
79	日付 A - 日付 B (ミリ秒)	フィールド A の値とフィールド B の日付の差をミリ秒単位で計算します。
80	日付 A - 日付 B (秒)	フィールド A の値とフィールド B の日付の差を秒単位で計算します。
81	日付 A - 日付 B (分)	フィールド A の値とフィールド B の日付の差を分単位で計算します。
82	日付 A - 日付 B (時間)	フィールド A の値とフィールド B の日付の差を時間単位で計算します。
83	日付 A の日ごとの時間	フィールド A の値の時間の部分を計算します。
84	日付 A の日ごとの分	フィールド A の値の分の部分を計算します。
85	日付 A の日ごとの秒	フィールド A の値の秒の部分を計算します。
86	ROUND_CUSTOM(A, B)	フィールド B の値で指定した丸めモードに従い、フィールド A の値を丸めます。 0-UP ゼロから離れるように丸めます。 1-DOWN ゼロに向かって丸めます。 2-CEILING 正の無限大に向かって丸めます。 3-FLOOR 負の無限大に向かって丸めます。 4-HALF_UP 最も近い整数に向かって丸めます。等距離の場合、切り上げます。 5-HALF_DOWN 最も近い整数に向かって丸めます。等距離の場合、切り捨てます。 6-HALF_EVEN 最も近い整数に向かって丸めます。等距離の場合、偶数に向かって丸めます。 7-UNNECESSARY 演算の結果が正確であり、丸めが必要ないことを表します。

項番	計算式	説明
87	ROUND_CUSTOM(A, B, C)	フィールド C の値で指定した丸めモードに従い、フィールド A の値を丸めます。 フィールド B の値は小数部の桁数です。
88	Date A + B Seconds	フィールド A の値にフィールド B の値を加算します。フィールド B の値は秒として加算されます。
89	Remainder of A / B	フィールド A の値をフィールド B の値で割った余りを計算します。

(凡例) - : 説明はありません。

2.4.20 [変換]選択/名前変更 (Select values)

入力ストリームにあるフィールドの取捨選択、名前変更、メタ情報の設定/変更などの処理を実行し、出力ストリームに出力します。処理は次の順で実行されます。

1. 選択フィールド
2. 除去フィールド
3. メタ情報

選択フィールドで名前を変更した場合、以降の処理で変更後の名前を指定しないとエラーが発生します。

(1) [選択フィールド]タブ

選択フィールドに関する設定をします。出力するフィールドの名称、長さ、精度を変更できます。

図 2-79 [選択/名前変更]画面-[選択フィールド]タブ

The screenshot shows a window titled '値選択' (Value Selection) with a sub-tab 'ステップ名 選択/名前変更' (Step Name Select/Rename). Inside, there are three tabs: '選択フィールド' (Select Fields), '除去フィールド' (Remove Fields), and 'メタ情報' (Meta Information). The '選択フィールド' tab is active, showing a table with the following columns: '#', 'フィールド名' (Field Name), '変更名称' (New Name), '長さ' (Length), and '精度' (Precision). The first row has the number '1' in the '#' column. To the right of the table are two buttons: 'フィールドの選択' (Fields Selection) and 'マッピング' (Mapping). Below the table is a checkbox labeled '未定義のフィールドを含む' (Include undefined fields). At the bottom of the window are three buttons: 'ヘルプ' (Help), 'OK(O)' (OK), and 'キャンセル(C)' (Cancel).

表 2-81 [選択/名前変更]画面-[選択フィールド]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	出力対象のフィールド名を指定します。
3	[変更名称]	出力対象のフィールド名を変更する場合、変更後のフィールド名を指定します。

項番	項目名	説明
4	[長さ]	データ出力時の長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
5	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
6	[フィールドの選択]ボタン	このボタンを押すと、入力ストリームのフィールドを自動入力できます。
7	[マッピング]ボタン	このボタンを押すと、専用のダイアログで、入力ステップと出力ステップのフィールド定義を参照し、フィールド同士の対応付けをします。
8	[未定義の精度を含む]	この項目にチェックを付けると、選択していないフィールドも出力します。

(2) [除去フィールド]タブ

除去フィールドに関する設定をします。指定したフィールドを除去して出力します。

図 2-80 [選択/名前変更]画面-[除去フィールド]タブ



表 2-82 [選択/名前変更]画面-[除去フィールド]タブの項目

項番	項目名	説明
1	[フィールド名]	除去対象のフィールド名を指定します。
2	[フィールドの除去]ボタン	このボタンを押すと、入力ストリームのフィールドをリストに追加します。

(3) [メタ情報]タブ

メタ情報に関する設定をします。指定したフィールドのメタ情報を変更します。

図 2-81 [選択/名前変更]画面-[メタ情報]タブ

表 2-83 [選択/名前変更]画面-[メタ情報]タブの項目

項番	項目名	説明
1	[フィールド名]	メタ情報を変更するフィールド名を指定します。
2	[変更フィールド名]	フィールド名を変更する場合に、変更後のフィールド名を指定します。空欄の場合は名称を変更しません。
3	[データタイプ]	データタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[長さ]	データ出力時の長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
5	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
6	[バイナリから変換]	バイナリ文字列を数値データに変換するかどうかを指定します。
7	[書式]	出力書式を指定します。
8	[デート型を自動処理]	[Y]を指定すると、文字列などを日付データ型に変換するときに、変換できる文字列のパターンが多くなります。 変換できる文字列のパターンとして、「2000/02/00」（文字列）を変換する例を次に示します。 変換前 "2000/02/00" 変換後 "2000/01/31" [N]を指定すると、該当するパターンの文字列などを日付データ型に変換するときに、エラーが発生します。
9	[現地日付]	日付のロケールを指定します。 デフォルトはシステムで設定されている値を使用します。
10	[日付のタイムゾーン]	日付のタイムゾーンを指定します。変更すると、日付や時刻も自動で変更されます。

項番	項目名	説明
		デフォルトはシステムで設定されている値を使用します。
11	[大まかな数値変換]	[Y]を指定すると、文字列などを日付データ型に変換する際に、数字以外が含まれている場合でもエラーが発生しないで変換できます。 変換対象は、数字以外の文字が出現するまでの数字列です。 変換前 "090-9999-9999"[String] 変換後 "090"[Integer] [N]を指定すると、文字列などを日付データ型に変換する際に、数字以外が含まれていると、エラーが発生します。 エラーが発生する例 "2000/02/00"
12	[文字エンコード]	文字のエンコードを変更します。
13	[桁区切り文字]※	[. (ドット)] など、小数点を表す文字を指定します。
14	[数値囲み文字]※	[, (コンマ)] など、桁を区切る文字を指定します。
15	[通貨記号]	[\$ (ドル)] など、通貨を表す記号を指定します。
16	[フィールドの変更]ボタン	このボタンを押すと、入力ストリームのフィールドをリストに追加します。

注※ 画面の項目名と指定する内容が異なります。

2.4.21 [変換]重複行削除 (Unique rows)

ソート済みのデータの重複行を削除します。

また、エラーハンドリングを使って、重複行だけを出力することもできます。

図 2-82 [重複行削除]画面

重複行削除

ステップ名 重複行削除

設定

カウンタを出力する ☐ フィールド名

重複した行を転送 ☐ エラー明細

フィールド

#	フィールド名	大文字/小文字を区別しない
1		

? ヘルプ OK(O) キャンセル(C) フィールドを取得(G)

表 2-84 [重複行削除]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[カウンタを出力する]	この項目にチェックを付けると、重複行の数をカウントして出力します。 重複がない場合、1 が出力されます。 [フィールド名] 重複行の数を出力するフィールド名を指定します。
3	[重複した行を転送]	この項目にチェックを付けると、重複行をエラーストリームに転送します。 あらかじめデータがソートされている必要があります。連続する 2 行を比較します。ステップの右クリックメニューから[エラー処理定義]を設定することで、重複行だけを出力できます。 [エラー明細] 重複する行が検出されたときに表示される、エラー処理を説明する文字列を設定します。 設定した文字列は、[エラー処理定義]の[エラー詳細]で設定したフィールドに出力されます。
4	[フィールド名]	重複を確認するフィールドを指定します。 フィールド名を指定するか、[フィールドを取得]ボタンを押して、入力ストリームからすべてのフィールドを取得します。
5	[大文字/小文字を区別しない]	大文字と小文字を区別するかどうかを[Y]または[N]で指定します。 [Y]に設定すると、大文字と小文字を区別しません。例えば、「Kettle」、「KETTLER」、「kettle」の場合、最初の「Kettle」以外は重複行として削除されます。

2.4.22 [変換]重複行削除(HashSet) (Unique rows (HashSet))

入力データの重複行を削除します。

処理できるデータ量は JVM に割り当てるメモリに依存します。重複行削除と同様にエラーハンドリングを使って、重複行だけを出力することもできます。

図 2-83 [重複行削除(HashSet)]画面

重複行削除(HashSet)

ステップ名

重複行削除 (HashSet) |

設定

保存行の値を利用して比較 ☐

重複行を転送 ☐ エラー明細

フィールド

#

フィールド名

1

ヘルプ

OK(O)

キャンセル(C)

フィールドを取得(G)

表 2-85 [重複行削除(HashSet)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[保存行の値を利用して比較]	この項目にチェックを付けると、すべてのレコードをメモリに保持して比較します。 チェックを付けることで、ハッシュ衝突の発生で、異なるレコードが重複レコードと見なされ行削除されることを防ぎます。 【注意事項】 ハッシュ衝突を防ぐため、チェックを付けてください。
3	[重複行を転送]	この項目にチェックを付けると、重複行をエラーストリームに転送します。 ソートしていないデータでも変換できますが、大量のデータを扱う場合はメモリ不足が起きやすくなります。 ステップの右クリックメニューから[エラー処理定義]を設定することで、重複行だけを出力できます。 【注意事項】 Java の HashSet を使用して処理するため、ハッシュ衝突の発生によって、異なるレコードを重複レコードと見なして処理されるおそれがあります。[保存行の値を利用して比較]にチェックを付けることで、ハッシュ衝突を防げます。 [エラー明細] 重複する行が検出されたときに表示される、エラー処理を説明する文字列を設定します。 設定した文字列は、[エラー処理定義]の[エラー詳細]で設定したフィールドに出力されます。
4	[フィールド名]	フィールド名を指定するか、[フィールドを取得]ボタンを押して、入力ストリームからすべてのフィールドを取得します。

2.5 ユーティリティ

- [ユーティリティ]NULL 値定数変換 (If field value is null)
- [ユーティリティ]ファイル処理 (Process files)
- [ユーティリティ]プロセス実行 (Execute a process)
- [ユーティリティ]メタデータ構造 (Metadata structure of stream)
- [ユーティリティ]メール (Mail)
- [ユーティリティ]ログ書き込み (Write to log)
- [ユーティリティ]定数 NULL 変換 (Null if)
- [ユーティリティ]行複製 (Clone row)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.5.1 [ユーティリティ]NULL 値定数変換 (If field value is null)

あるフィールドの NULL 値を特定の値に変換します。

図 2-84 [NULL 値定数変換]画面

NULL値定数変換

ステップ名

If field value is null

すべてのフィールドのNullを置き換える。

置き換える値

空文字を設定

☐

マスク(日付)

フィールドを選択

☐

値のタイプを選択

☐

値のタイプ

#	タイプ	置き換える値	変換マスク(日付)	空文字を設定

フィールド

#	フィールド	置き換える値	変換マスク(日付)	空文字を設定
1				

ヘルプ

OK(O)

フィールドを取得(G)

キャンセル(C)

表 2-86 [NULL 値定数変換]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[置き換える値]	NULL 値を置き換える値を指定します。 [フィールドを選択]と[値のタイプを選択]のどちらの項目もチェックを付けない場合に指定できます。
3	[空文字を設定]	置き換える値が空白の場合、この項目にチェックを付けると空文字が挿入されます。 [フィールドを選択]と[値のタイプを選択]のどちらの項目もチェックを付けない場合に指定できます。
4	[マスク(日付)]	日付データ型（Data, Timestamp）のフィールドの NULL 値を置換する際に、[置き換える値]を日付データに変換する書式を指定します。 [フィールドを選択]と[値のタイプを選択]のどちらの項目もチェックを付けない場合に指定できます。
5	[フィールドを選択]	この項目にチェックを付けると、指定したフィールドに対してだけ NULL 値を変換します。 [値タイプを選択]にチェックを付けた場合、この項目は指定できません。

項番	項目名	説明
6	[値のタイプを選択]	この項目にチェックを付けると、指定したデータタイプを持つフィールドの NULL 値だけを変換します。 [フィールドを選択]にチェックを付けた場合、この項目は指定できません。
7	[タイプ]	NULL 値を変換するデータタイプを指定します。 [値のタイプを選択]にチェックを付けると指定できます。
8	[置き換える値]	NULL 値を置き換える値を指定します。 [値のタイプを選択]にチェックを付けると指定できます。
9	[変換マスク(日付)]	日付データ型 (Data, Timestamp) のフィールドの NULL 値を置換する際に、[置き換える値]を日付データに変換する書式を指定します。 [値のタイプを選択]にチェックを付けると指定できます。
10	[空文字を設定]	置き換える値が空白の場合、この項目にチェックを付けると空文字が挿入されます。 [値のタイプを選択]にチェックを付けると指定できます。
11	[フィールド]	NULL 値を変換するフィールドを指定します。 [フィールドを選択]にチェックを付けると指定できます。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
12	[置き換える値]	NULL 値を置き換える値を指定します。 [フィールドを選択]にチェックを付けると指定できます。
13	[変換マスク(日付)]	日付データ型 (Data, Timestamp) のフィールドの NULL 値を置換する際に、[置き換える値]を日付データに変換する書式を指定します。 [フィールドを選択]にチェックを付けると指定できます。
14	[空文字を設定]	置き換える値が空白の場合、この項目にチェックを付けると空文字が挿入されます。 [フィールドを選択]にチェックを付けると指定できます。

2.5.2 [ユーティリティ]ファイル処理 (Process files)

指定したファイルをコピー、移動、または削除 (Delete) します。

図 2-85 [ファイル処理]画面

表 2-87 [ファイル処理]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[実行]	実行する処理を次の項目から指定します。 • [コピー] • [移動] • [Delete]
3	[親ディレクトリを作成]	この項目にチェックを付けると、[フィールド (結果)] (コピー先または移動先) で指定したパスの中のディレクトリがない場合でも、自動でディレクトリが作成されます。 チェックを付けない場合、[フィールド (結果)] (コピー先または移動先) で指定したパスの中のディレクトリがないと、エラーが発生します。 [実行]で[Delete]を指定した場合、この項目は指定できません。
4	[ファイルを上書き]	この項目にチェックを付けると、[フィールド (結果)] (コピー先または移動先) で指定したファイルを上書きします。 [実行]で[Delete]を指定した場合、この項目は指定できません。
5	[ファイル名を結果に含む]	この項目にチェックを付けると、コピーまたは移動されたファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 [実行]で[Delete]を指定した場合、この項目は指定できません。
6	[シミュレーションモードに設定]	この項目にチェックを付けると、実行時に指定されたファイル操作ができるかどうかをテストします。実際のファイル操作は行われません。 ただし、[親ディレクトリを作成]にチェックを付けた場合、指定したパスのディレクトリがないときは、親ディレクトリが作成されます。
7	[フィールド (結果)] (1 行目) ※	コピー、移動、および Delete の実行対象となるファイルパスのフィールド名を指定します。

項番	項目名	説明
8	[フィールド（結果）]（2行目）	コピーまたは移動先のファイルパスのフィールド名を指定します。

注※ 画面の項目名と指定する内容が異なります。

2.5.3 [ユーティリティ]プロセス実行（Execute a process）

プロセスを実行し、結果を出力します。すべての入力行に対して実行します。

(1) [General]タブ

プロセス実行に関する設定をします。

図 2-86 [プロセス実行]画面-[General]タブ

表 2-88 [プロセス実行]画面-[General]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[プロセスフィールド]	開始するプロセスが指定されているフィールド名を指定します。フィールドデータは実行されるシェルスクリプトやバッチファイルを指定します。
3	[Arguments in Fields]	この項目にチェックを付けると、プロセスに対する引数を指定できます。
4	[success でない場合は失敗]	この項目にチェックを付けると、プロセスが正常終了（Exit value=0）以外で終了した場合、ステップがエラー終了します。
5	[Field Name]	引数となるフィールドがある場合、フィールド名を指定します。

項番	項目名	説明
		[Arguments in Fields]にチェックを付けると指定できます。

(2) [Output]タブ

ステップの出力について設定します。

図 2-87 [プロセス実行]画面-[Output]タブ

表 2-89 [プロセス実行]画面-[Output]タブの項目

項番	項目名	説明
1	[出力行区切り文字]	出力行の区切り文字を指定します。
2	[結果フィールド名]	結果（標準出力）を出力するフィールドの名前を指定します。デフォルト値は「Result output」です。
3	[エラーフィールド名]	標準エラー出力を出力するフィールドの名前を指定します。デフォルト値は「Error output」です。
4	[Exit 値フィールド名]	終了値を出力するフィールドの名前を指定します。デフォルト値は「Exit value」です。

2.5.4 [ユーティリティ]メタデータ構造 (Metadata structure of stream)

ストリームのフィールドを読み取り、それらのメタデータを出力します。

出力するメタデータを次に示します。

- Position：列番号

- Fieldname：フィールド名
- Comments：-
- Type：タイプ
- Length：長さ
- Precision：精度
- Origin：フィールドが生成されたステップ名

図 2-88 [メタデータ読み込み(入力ストリーム)]画面

メタデータ読み込み(入力ストリーム)

ステップ名メタデータ構造

レコード数を出力する☐

フィールド名

ヘルプ

OK(O)

キャンセル(C)

表 2-90 [メタデータ読み込み(入力ストリーム)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[レコード数を出力する]	この項目にチェックを付けると、レコードの行数を出力できます。
3	[フィールド名]	レコード数を出力するフィールド名を指定します。 [レコード数を出力する]にチェックを付けると指定できます。

2.5.5 [ユーティリティ]メール (Mail)

SMTP サーバを使用して、前のステップからの電子メールを送信するための入力データを使用して電子メールを送信します。

(1) [アドレス]タブ

宛先や差出人について設定します。

図 2-89 [メール]画面-[アドレス]タブ

メール

ステップ名 Mail

アドレス サバ メッセージ 添付ファイル 埋め込みイメージ

宛先設定

宛先
Cc
Bcc

差出人情報

名前
メールアドレス

返信先
返信名
電話番号

ヘルプ

OK(O)

キャンセル(C)

表 2-91 [メール]画面-[アドレス]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[宛先]	前のステップから宛先のメールアドレスを含むフィールドを指定します。 スペースで区切ると複数のアドレスを指定できます。
3	[Cc]	前のステップからメールアドレスを含むフィールドを指定します。 スペースで区切ると複数のアドレスを指定できます。 メッセージのコピーが送信され、他の受信者のメールにもここで指定したメールアドレスが表示されます。
4	[Bcc]	前のステップからメールアドレスを含むフィールドを指定します。 スペースで区切ると複数のアドレスを指定できます。 他の受信者のメールにはメールアドレスが表示されません。
5	[名前]	前のステップから送信者の名前を含むフィールドを指定します。 【注意事項】 [名前]に特殊文字（<,>）を使用してデータ変換を実行した場合、メール送信で きずエラーになります。

項番	項目名	説明
		[名前]に特殊文字（< , >）を使用したい場合は、次の例のように、設定値全体を「」（ダブルクォーテーション）で囲って指定してください。 例："<pentaho>"
6	[メールアドレス]	前のステップから送信者のメールアドレスを含むフィールドを指定します。メールヘッダーの「From」に該当します。
7	[返信先]	前のステップから返信先のメールアドレスを含むフィールドを指定します。「Reply」の送信先に該当します。
8	[返信名]	前のステップからメールに記載する担当者の名前を含むフィールドを指定します。「Contact information」の「宛先担当者」に該当します。 【注意事項】 [返信名]を指定する場合は、[電話番号]も入力してください。
9	[電話番号]	前のステップからメールに記載する連絡先の電話番号を含むフィールドを指定します。「Contact information」の「電話番号」に該当します。 【注意事項】 [返信名]を指定した場合は、[電話番号]を入力してください。「電話番号」を設定せずにデータ変換を実行すると、メール本文の「Contact information」の電話番号が null になります。

(2) [サーバ]タブ

使用する SMTP サーバについて設定します。

図 2-90 [メール]画面-[サーバ]タブ

メール

ステップ名Mail

アドレスサーバメッセージ添付ファイル埋め込みイメージ

サーバ設定

SMTPサーバ
ポート番号

認証設定

認証を使用する
ユーザ名
パスワード
セキュリティで保護された認証を使用す
安全な接続タイプSSL

ヘルプ

OK(O)

キャンセル(C)

表 2-92 [メール]画面-[サーバ]タブの項目

項番	項目名	説明
1	[SMTP サーバ]	前のステップから SMTP サーバアドレスを含むフィールドを指定します。
2	[ポート番号]	前のステップから SMTP サーバのポート番号を含むフィールドを指定します。
3	[認証を使用する]	この項目にチェックを付けると、SMTP サーバへの認証の使用を有効にします。
4	[ユーザ名]	前のステップから SMTP ユーザのアカウント名を含むフィールドを指定します。 [認証を使用する]にチェックを付けると指定できます。
5	[パスワード]	前のステップから SMTP ユーザのパスワードを含むフィールドを指定します。 [認証を使用する]にチェックを付けると指定できます。
6	[セキュリティで保護された 認証を使用する]	この項目にチェックを付けると、認証時にコネクションタイプ (SSL または TLS) を使用できます。
7	[安全な接続タイプ]	コネクションタイプを次の項目から指定します。 • [SSL] • [TLS] (STARTTLS) [セキュリティで保護された認証を使用する]にチェックを付けると指定できます。

(3) [メッセージ]タブ

メールのテキスト内容について設定します。

図 2-91 [メール]画面-[メッセージ]タブ

メール

ステップ名Mail

アドレスサーバメッセージ添付ファイル埋め込みイメージ

メッセージ設定

メッセージに日付を含む

コメントのみ送信

HTMLメールを使用する

文字コードUTF-8

レベルを管理

優先度Normal

緊急度Normal

重要度Normal

メッセージ

件名

本文

ヘルプ

OK(O)

キャンセル(C)

表 2-93 [メール]画面-[メッセージ]タブの項目

項番	項目名	説明
1	[メッセージに日付を含む]	この項目にチェックを付けると、メッセージに日付を含めます。
2	[コメントのみ送信]	この項目にチェックを付けると、メールにはデータ変換の実行に関する情報は含まれません。 チェックを付けない場合、メールにはデータ変換の実行に関する情報が含まれます。
3	[HTML メールを使用する]	この項目にチェックを付けると、メッセージは HTML 形式で送信されます。 「Context-Type」は「text/html」です。
4	[文字コード]	文字コードを指定します。 [HTML メールを使用する]にチェックを付けると指定できます。
5	[レベルを管理]	この項目にチェックを付けると、「優先度」「緊急度」「重要度」を選択できるようになります。

項番	項目名	説明
6	[優先度]	優先度を次の項目から指定します。 • [Low] • [Normal] • [High] 選択した優先度によって、メールヘッダーの「X-Priority」に次の値が表示されます。 • Low を選択した場合：3 • Normal を選択した場合：2 • High を選択した場合：1
7	[緊急度]	緊急度を次の項目から指定します。 • [Low] • [Normal] • [High] 選択した緊急度によって、メールヘッダーの「Importance」に次の値が表示されます。 • Low を選択した場合：low • Normal を選択した場合：normal • High を選択した場合：high
8	[重要度]	重要度を次の項目から指定します。 • [Normal] • [Personal] • [Private] • [Confidential] 選択した重要度によって、メールヘッダーの「Sensitivity」に次の値が表示されます。 • Normal を選択した場合：normal • Personal を選択した場合：personal • Private を選択した場合：private • Confidential を選択した場合：company-confidential
9	[件名]	前のステップからメールの件名を含むフィールドを指定します。
10	[本文]	前のステップからメール本文の内容を含むフィールドを指定します。

(4) [添付ファイル]タブ

添付ファイルについて設定します。

図 2-92 [メール]画面-[添付ファイル]タブ

表 2-94 [メール]画面-[添付ファイル]タブの項目

項番	項目名	説明
1	[コンテンツファイルを添付]	この項目にチェックを付けると、コンテンツファイルを添付します。
2	[コンテンツのフィールド名]	前のステップからコンテンツファイルのフィールド名を含むフィールドを指定します。 [コンテンツファイルを添付]にチェックを付けると指定できます。
3	[ファイル名のフィールド名]	前のステップからコンテンツファイルのファイル名を含むフィールドを指定します。 [コンテンツファイルを添付]にチェックを付けると指定できます。
4	[先行のステップから値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値をファイルパスとして使用します。 チェックを付けると、[ファイルとディレクトリ]を使用したファイルの添付はできません。
5	[フィールド]	前のステップからファイル名およびディレクトリ名を含むフィールドを指定します。 フィールドの値として指定するファイルパスには、正規表現は使用できません。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
6	[フィールド (検索文字列)]	前のステップからフィールド (検索文字列) を含むフィールドを指定します。

項番	項目名	説明
		[フィールド]で指定したファイル名およびディレクトリ名を、指定した文字列で検索し、[フィールド（検索文字列）]と完全一致するファイルだけ添付します。 正規表現で入力する必要があります。 [先行のステップから値を引き継ぐ]にチェックを付けると指定できます。
7	[ファイルとディレクトリ]	添付ファイルのパス、または添付ファイルが格納されているディレクトリのパスを指定します。 [圧縮サイズを確認する]で0以外の数値を指定した場合、[ファイルとディレクトリ]の指定は無効になります。
8	[サブディレクトリを含む]	[ファイルとディレクトリ]にディレクトリを指定し、この項目にチェックを付けると、指定したディレクトリのサブディレクトリに含まれるファイルも添付されます。
9	[検索文字列]	[ファイルとディレクトリ]で指定したファイルとディレクトリ内を、指定した文字列で検索し、検索文字列と完全一致するファイルを添付します。 正規表現で入力する必要があります。 [ファイルとディレクトリ]にチェックを付けると指定できます。
10	[圧縮ファイル]	この項目にチェックを付けると、複数のファイルを1つのアーカイブに圧縮してからメールに添付します。
11	[先行のステップより値を引き継ぐ]	この項目にチェックを付けると、先行のステップから入力されたフィールドの値を圧縮ファイル名として使用します。 この項目にチェックを付けると、[圧縮ファイル名]を使用した指定はできません。
12	[フィールド]	前のステップから圧縮ファイル名が含まれているフィールドを指定します。 [先行のステップより値を引き継ぐ]にチェックを付けると指定できます。
13	[圧縮ファイル名]	圧縮ファイル名を指定します。 [先行のステップより値を引き継ぐ]にチェックを付けると、この項目を指定できません。
14	[圧縮サイズを確認する]	添付ファイルの合計サイズがこの数値（バイト）以上の場合だけ、圧縮されます。 0以外の数値を指定した場合、[ファイルとディレクトリ]の指定は無効になります。 [ファイルとディレクトリ]で指定されたファイルの合計サイズが指定値を超えても超えなくても、ファイルは添付されません。 【注意事項】 [圧縮サイズを確認する]に0以外の数値を指定してデータ変換を実行した場合、添付ファイルが圧縮されず、ファイルが添付されません。

(5) [埋め込みイメージ]タブ

メールに本文に埋め込む画像ファイルについて設定します。

[illegible]

項番	項目名	説明
1	[ファイル]	メールに組み込む画像ファイルを指定します。 メール本文中に次の文字列を指定します。 <pre></pre> 「myimageid」はコンテンツ ID です。 [メッセージ]タブの[HTML メールを使用する]にチェックを付けると指定できます。
2	[イメージ]	[追加]ボタンを押すと、ファイル名に指定した内容が[埋め込みイメージ]の欄に反映されます。
3	[コンテンツ ID]	[ファイル]に画像ファイルを指定すると、自動で設定されます。 [メッセージ]タブの[HTML メールを使用する]にチェックを付けると指定できます。

入力行ごとに、所定のフォーマットのログを出力します。

図 2-94 [ログ書き込み]画面

ログ書き込み

ステップ名

ログ書き込み

ログレベル

基本のログ

ヘッダーを含む

☒

行数指定

☐

出力行数

0

ログ書き込み

フィールド

#	フィールド
1	

ヘルプ

OK(O)

フィールドを取得(G)

キャンセル(C)

表 2-96 [ログ書き込み]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[ログレベル]	出力するログのログレベルを次の項目から指定します。 [ログなし] ログを出力しません。 [エラーログのみ] エラーログだけを出力します。 [最小限のログ] 最小限のログだけを出力します。 [基本のログ] デフォルトのログレベルです。 [詳細なログ] 詳細なログを出力します。 [デバッグ] 非常に詳細なログを出力します。目的がデバッグの場合に設定します。 [ローレベル (非常に詳細)] 大量のログデータを出力します。 【注意事項】 データ変換実行時、ここで指定したよりも詳細度が低いログを出力する指定を すると、ログは出力されません。例えば、ここで[基本のログ]を指定した場合、デー タ変換実行時に[最小限のログ]を指定するとログは出力されません。

項番	項目名	説明
3	[ヘッダーを含む]	この項目にチェックを付けると、入力データのフィールド名を出力します。
4	[行数指定]	この項目にチェックを付けると、[出力行数]を指定できます。
5	[出力行数]	ログを出力する入力行数を 1 以上の整数で指定します。それ以外の値でもエラーは発生しませんが、「1」と見なされます。 [行数指定]にチェックを付けると指定できます。
6	[ログ書き込み]	指定した任意の内容がログに出力されます。
7	[フィールド]	ログに値を出力するフィールドを指定します。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

2.5.7 [ユーティリティ]定数 NULL 変換 (Null if)

あるフィールドの特定の定数を NULL 値に変換します。

図 2-95 [定数 NULL 変換]画面

表 2-97 [定数 NULL 変換]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	定数を NULL 値に変換するフィールドを指定します。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
3	[NULL 可能]	NULL 値に変換する定数を指定します。 大文字と小文字を区別して変換されます。

2.5.8 [ユーティリティ]行複製（Clone row）

入力行のコピーを作成し、コピーデータを後続のステップに渡します。

図 2-96 [Clone row]画面

Clone row

ステップ名

行複製

レコード数

0

レコード番号を追加する

☐

フィールド名

出力フィールド

フラグを追加する

☐

フィールド名

複製数を出力に追加する

☐

複製数フィールド

ヘルプ

OK(O)

キャンセル(C)

表 2-98 [Clone row]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[レコード数]	入力行のコピーする数を指定します。 [レコード番号を追加する]にチェックを付けていない場合に、指定できます。
3	[レコード番号を追加する]	この項目にチェックを付けると、先行のステップのフィールドから複製数を制御します。 チェックを付けると、[レコード数]のテキストボックスが非活性になります。
4	[フィールド名]	フィールド名を指定します。 [レコード番号を追加する]にチェックを付けると指定できます。
5	[フラグを追加する]	この項目にチェックを付けると、その行が複製されたものかどうかを判定した結果を出力します。 - Y：複製されたものである - N：複製されたものではない
6	[フィールド名]	フィールド名を指定します。 [フラグを追加する]にチェックを付けると指定できます。
7	[複製数を出力に追加する]	この項目にチェックを付けると、複製された数を出力します。 オリジナルの行には「0」が出力されます。
8	[複製数フィールド]	フィールド名を指定します。

項番	項目名	説明
		[複製数を出力に追加する]にチェックを付けると指定できます。

2.6 フロー

- [フロー]Blocking step (Blocking step)
- [フロー]Java フィルター (Java Filter)
- [フロー]ジョブ実行 (Job Executor)
- [フロー]ステップブロック (Block this step until steps finish)
- [フロー]ストリーム結合 (Append streams)
- [フロー]ダミー (何もしない) (Dummy (do nothing))
- [フロー]トランスフォーメーション実行 (Transformation executor)
- [フロー]フィルター (Filter rows)
- [フロー]停止 (Abort)
- [フロー]優先順位 (Prioritize streams)
- [フロー]条件分岐 (Switch/case) (Switch / case)
- [フロー]空ストリーム探知 (Detect empty stream)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.6.1 [フロー]Blocking step (Blocking step)

前ステップからすべての行が入力し終わるまで待機し、前ステップからの入力処理が終了したときに、入力データの最終行を出力します。オプションを設定すると、入力の最終行だけではなく、入力されたすべての行を出力することもできます。

図 2-97 [ブロック]画面

表 2-99 [ブロック]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[すべての行を許可する]	この項目にチェックを付けると、入力されたすべての行を出力します。
3	[テンポラリディレクトリ]	入力データをキャッシュメモリに格納するために作成される一時ファイルを格納するディレクトリを指定します。 [すべての行を許可する]にチェックを付けると指定できます。
4	[テンポラリファイルの接頭子]	一時ファイルの接頭子を指定します。 [すべての行を許可する]にチェックを付けると指定できます。
5	[キャッシュサイズ]	入力データをキャッシュメモリに格納する最大行数を指定します。この項目で指定した最大行数を超えたデータが入力されたときに、一時ファイルが作成されます。 [すべての行を許可する]にチェックを付けると指定できます。
6	[テンポラリファイルを圧縮する]	この項目にチェックを付けると、一時ファイルを圧縮します。 [すべての行を許可する]にチェックを付けると指定できます。

2.6.2 [フロー]Java フィルター (Java Filter)

入力レコードを、指定された Java の評価式に従って処理し、出力ストリームに出力します。

図 2-98 [Java フィルター]画面

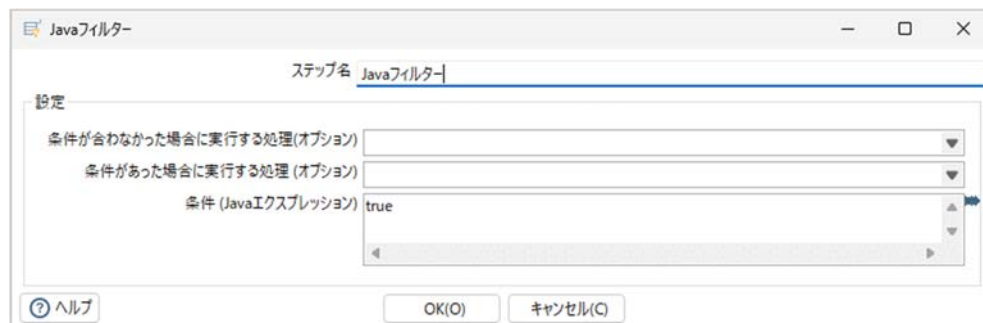


表 2-100 [Java フィルター]画面

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[条件が合わなかった場合に実行する処理 (オプション)]※	定義された条件によって true と評価された行を、ここで指定したステップに送信します。
3	[条件があった場合に実行する処理 (オプション)]※	定義された条件によって false と評価された行を、ここで指定したステップに送信します。

項番	項目名	説明
4	[条件 (Java エクスプレッション)]	データをフィルタリングする条件を Java で定義します。使用できる評価式は、戻り値が true または false のものだけです。 【注意事項】 Timestamp 型または Internet Address 型のフィールドを指定すると、エラーが発生します。

注※ 画面の項目名と指定する内容が異なります。

2.6.3 [フロー]ジョブ実行 (Job Executor)

指定したジョブを呼び出し、入力ストリームのデータに応じた設定でジョブを実行します。

デフォルトでは、指定されたジョブは入力行ごとに 1 回実行されます。入力行を使用してパラメータと変数を設定できます。ジョブを実行する行数は個別に設定することもできます。

(1) [パラメーター]タブ

データ変換実行時の変数を指定します。値は、入力ストリームのフィールド値または固定値のどちらかで設定します。[パラメータを取得]ボタンを押すと、指定したジョブで設定したパラメータを取得できます。

図 2-99 [ジョブ実行]画面-[パラメーター]タブ

ジョブ実行

ステップ名
ジョブ実行

Job:
Browse...

パラメーター 実行結果 行グルーピング 結果行 結果ファイル

変数 / パラメータ名	使用するフィールド	静的入力値

☒ トランスフォーメーションから全ての変数を継承? パラメータを取得

? ヘルプ OK(O) キャンセル(C)

表 2-101 [ジョブ実行]画面-[パラメーター]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Job]	実行対象のジョブのファイルパスを指定します。
3	[変数 / パラメータ名]	変数名を指定します。
4	[使用するフィールド]	変数の値として設定する入力ストリームのフィールド名を指定します。 [行グルーピング]タブの[ジョブに送信する行数]に 1 より大きい数値を指定すると、1 回の子ジョブの実行で複数のレコードが渡されます。その場合は、先頭レコードのフィールドの値が変数に設定されます。 【注意事項】 [静的入力値]を指定する場合は、空欄にする必要があります。
5	[静的入力値]	変数の値として設定する固定値を指定します。親データ変換の変数を利用することもできます。
6	[トランスフォーメーションから全ての変数を継承?]	この項目にチェックを付けると、親データ変換の変数を子ジョブに引き継ぎます。

(2) [実行結果]タブ

子ジョブの実行結果を、親データ変換の特定のステップへ出力するための設定をします。

図 2-100 [ジョブ実行]画面-[実行結果]タブ



表 2-102 [ジョブ実行]画面-[実行結果]タブの項目

項番	項目名	説明
1	[実行結果のターゲットステップ]	実行結果を出力するステップを指定します。空欄の場合、実行結果を出力しません。
2	[Field description]	—
3	[Field name]	—

(凡例) —：説明はありません。詳細は、Pentaho Documentation を参照してください。

(3) [行グルーピング]タブ

親データ変換の入力データに対し、子ジョブに引き渡すレコード群を決定する方法を定義します。

図 2-101 [ジョブ実行]画面-[行グルーピング]タブ

表 2-103 [ジョブ実行]画面-[行グルーピング]タブの項目

項番	項目名	説明
1	[ジョブに送信する行数]※	データ変換をループ実行する場合の 1 つのレコード群の行数を指定します。 例えば、入力データに 10 行のレコードがあり、この項目に「2」を指定した場合のデータ変換の処理を次に示します。 • 2 行を 1 レコード群として扱う • 2 行ごとに子ジョブを実行する • 子ジョブの実行を 5 回繰り返す
2	[行グループフィールド]※	グルーピングのキーとしたい、入力ストリーム内のフィールドの名前を指定します。 この項目で指定したフィールドの値が同じ、かつ、連続したレコードを 1 つのレコード群として、子ジョブを実行します。
3	[実行の前に行を集めて待つ時間 (ms)]※	ジョブの実行前に待機する時間を指定します。指定した時間が経過すると、時間内に入力されたレコードをまとめて、1 つのレコード群として子ジョブに引き渡して実行します。

注※ どれか 1 つしか設定できません。そのほかの値は空欄にする必要があります。

(4) [結果行]タブ

子ジョブから引き渡されたデータを出力するステップおよびフィールド情報を指定します。

図 2-102 [ジョブ実行]画面-[結果行]タブ

ジョブ実行

ステップ名
ジョブ実行

Job:

Browse...

パラメーター

実行結果

行グルーピング

結果行

結果ファイル

結果行のターゲットステップ

結果行の予測されるレイアウト:

フィールド名	データタイプ	長さ	Precision

ヘルプ

OK(O)

キャンセル(C)

表 2-104 [ジョブ実行]画面-[結果行]タブの項目

項番	項目名	説明
1	[結果行のターゲットステップ]	引き渡されたデータを出力するステップの名前を指定します。空欄の場合、結果を出力しません。
2	[フィールド名]	フィールド名を指定します。
3	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。

2. データ変換

Pentaho 機能説明 PDI 編

190

項番	項目名	説明
		【注意事項】 数値以外の値を指定したあとに画面を閉じて再度この画面を開いた場合、空欄に戻ります。
5	[Precision]	該当フィールドのデータタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。

(5) [結果ファイル]タブ

メモリに渡されたファイル名のリストを取得します。

図 2-103 [ジョブ実行]画面-[結果ファイル]タブ

ジョブ実行

ステップ名
ジョブ実行

Job:
Browse...

パラメーター 実行結果 行グルーピング 結果行 結果ファイル

結果ファイル情報を送信するステップ
結果ファイルのフィールド名
FileName

ヘルプ OK(O) キャンセル(C)

表 2-105 [ジョブ実行]画面-[結果ファイル]タブの項目

項番	項目名	説明
1	[結果ファイル情報を送信するステップ]	データを出力するステップの名前を指定します。空欄の場合、結果を出力しません。
2	[結果ファイルのフィールド名]	データを出力するフィールドの名前を指定します。

2.6.4 [フロー]ステップブロック (Block this step until steps finish)

前ステップからの入力を阻止し、指定したステップの処理がすべて終了したときに、入力データの出力を開始します。このステップを使用するとデータ変換がデッドロック状態となり、処理が停止する場合があります。使用する前に、デッドロック状態が発生しないことを確認してください。

図 2-104 [ステップ終了までブロック]画面

ステップ終了までブロック

ステップ名

ステップブロック

後述のステップを確認

#	ステップ名	コピー数
1		

ヘルプ

OK(O)

ステップ取得

キャンセル(C)

表 2-106 [ステップ終了までブロック]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[後述のステップを確認]	ここで指定したステップの処理が終了するまで、データ入力を阻止します。 複数のステップを指定した場合は、すべてのステップの処理が終了するまでデータ入力を阻止します。
3	[ステップ名]	処理の終了を待つステップの名前を指定します。
4	[コピー数]	処理の終了を待つステップのコピー数を指定します。 ステップをコピーして多重化している場合は、コピー数によってそれぞれのステップを区別します。 多重化していない場合は「0」を、多重化している場合は「0 より大きい整数値」を指定します。それ以外の値を指定した場合、エラーが発生します。

2.6.5 [フロー]ストリーム結合 (Append streams)

2つの入力データを連結します。正常な結果を得るには、連結するデータのフィールド定義が同じである必要があります。

出力例を次に示します。

入力データ

入力データA	入力データB
Data	Data
A	B

A	B
A	

出力データ

Date
A
A
A
B
B

図 2-105 [ストリーム結合]画面

?

ストリーム結合

—

□

×

ステップ名

ストリーム結合|

上位の結合ステップ名

▼

下位の結合ステップ名

▼

?

ヘルプ

OK(O)

キャンセル(C)

表 2-107 [ストリーム結合]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[上位の結合ステップ名]	連結後、上部に配置したいデータを入力するステップを指定します。
3	[下位の結合ステップ名]	連結後、下部に配置したいデータを入力するステップを指定します。

2.6.6 [フロー]ダミー（何もしない）（Dummy (do nothing)）

入力ストリームのデータに対して、何も処理をしないで後続のステップに出力します。
デバッグや複数フローデータの縦結合などに使用します。

図 2-106 [ダミー（何もしない）]画面

?

ダミー（何もしない）

—

□

×

ステップ名

ダミー（何もしない）|

?

ヘルプ

OK(O)

キャンセル(C)

表 2-108 [ダミー（何もしない）]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

2.6.7 [フロー]トランスフォーメーション実行 (Transformation executor)

指定したデータ変換を呼び出し、入力ストリームのデータに応じた設定でデータ変換を実行します。

トランスフォーメーション実行ステップが配置されたデータ変換を「親データ変換」、トランスフォーメーション実行ステップで呼び出すデータ変換を「子データ変換」と呼びます。トランスフォーメーション実行ステップを使ったデータ変換の主な処理を次に示します。

- データの引き渡しと受け取り

トランスフォーメーション実行ステップにデータを入力すると、入力ストリームからデータを引き渡します。入力されたデータを取得するには、子データ変換内に行取得ステップを配置します。また、入力されたデータを親データ変換に引き渡すには、子データ変換に行送信ステップを配置してください。

行取得ステップについては「[\[ジョブ\]行取得 \(Get rows from result\)](#)」を、行送信ステップについては、「[\[ジョブ\]行送信 \(Copy rows to result\)](#)」を参照してください。

- 変数の設定

参照した入力ストリーム内の値を変数として設定し、データ変換を実行します。

- ループ実行

入力されたデータを特定のレコード群に区切って子データ変換に引き渡し、ループ実行します。ループ実行の対象に設定できるレコード群の例を次に示します。

- 1 行を 1 つのレコード群とする
- キーとなるフィールドの値が同じレコードを 1 つのレコード群とする

(1) [パラメーター]タブ

データ変換実行時の変数を設定します。入力ストリームのフィールド値または固定値のどちらかを設定します。

図 2-107 [トランスフォーメーション実行]画面-[パラメーター]タブ

[illegible]

表 2-109 [トランスフォーメーション実行]画面-[パラメーター]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Transformation]	実行対象のデータ変換のパスを指定します。
3	[変数/パラメーター名]	変数名を指定します。
4	[フィールド名]	変数の値として設定する入力ストリームのフィールド名を指定します。 [行のグルーピング]タブの[トランスフォーメーションに送信する行数]に 1 より大きい数値を指定すると、1 回の子データ変換で複数のレコードが渡されます。その場合は、先頭レコードのフィールドの値が変数に設定されます。 【注意事項】 [静的入力値]を指定する場合は、空欄にする必要があります。

項番	項目名	説明
5	[静的入力値]	変数の値として設定する固定値を指定します。親データ変換の変数を指定することもできます。 【注意事項】 [フィールド名]を指定する場合は、空欄にする必要があります。
6	[トランスフォーメーションからすべての変数を引き継ぐ]	この項目にチェックを付けると、親データ変換の変数を子データ変換に引き継ぎます。

(2) [実行結果]タブ

子データ変換の実行結果を、親データ変換のステップへ出力するための設定をします。

図 2-108 [トランスフォーメーション実行]画面-[実行結果]タブ

トランスフォーメーション実行

ステップ名
トランスフォーメーション実行

Transformation:
Browse...

パラメーター 実行結果 行のグルーピング 結果行 結果ファイル

実行結果のターゲットステップ

Field description	Field name
実行時間 (ms)フィールド	ExecutionTime
実行結果フィールド	ExecutionResult
エラー数フィールド	ExecutionNrErrors
読み込み行数フィールド	ExecutionLinesRead
書き込み行数フィールド	ExecutionLinesWritten
入力行数フィールド	ExecutionLinesInput
出力された行数フィールド	ExecutionLinesOutput
拒否された行数フィールド	ExecutionLinesRejected
アップデートされた行数フィールド	ExecutionLinesUpdated
削除した行数フィールド	ExecutionLinesDeleted
ファイルの取得数フィールド	ExecutionFilesRetrieved
exitステータスフィールド	ExecutionExitStatus
実行ログテキストフィールド	ExecutionLogText
ログチャンネルIDフィールド	ExecutionLogChannelId

ヘルプ OK(O) キャンセル(C)

表 2-110 [トランスフォーメーション実行]画面-[実行結果]タブの項目

項番	項目名	説明
1	[実行結果のターゲットステップ]	実行結果を出力する親データ変換のステップを指定します。空欄の場合、実行結果を出力しません。
2	[Field description]	—
3	[Field name]	—

(凡例) —：説明はありません。詳細は、Pentaho Documentation を参照してください。

(3) [行のグルーピング]タブ

親データ変換の入力データに対し、子データ変換に引き渡すレコード群を決定する方法を定義します。

図 2-109 [トランスフォーメーション実行]画面-[行のグルーピング]タブ

トランスフォーメーション実行

ステップ名
トランスフォーメーション実行

Transformation:
Browse...

パラメーター 実行結果 行のグルーピング 結果行 結果ファイル

トランスフォーメーションに送信する行数
1

行のグルーピングを行うフィールド
▼

行の取得の待ち時間 (ms)
\$

? ヘルプ OK(O) キャンセル(C)

表 2-111 [トランスフォーメーション実行]画面-[行のグルーピング]タブの項目

項番	項目名	説明
1	[トランスフォーメーションに送信する行数]※	データ変換をループ実行する場合の 1 つのレコード群の行数を指定します。 例えば、入力データに 10 行のレコードがあり、この項目に「2」を指定した場合のデータ変換の処理内容を次に示します。 • 2 行を 1 レコード群として扱う • 2 行ごとに子データ変換が実行する • 子データ変換を 5 回繰り返す
2	[行のグルーピングを行うフィールド]※	グルーピングのキーとしたい、入力ストリーム内のフィールドの名前を指定します。 この項目で指定したフィールドの値が同じ、かつ、連続したレコードを 1 つのレコード群として、子データ変換を実行します。
3	[行の取得の待ち時間 (ms)]※	レコードの入力時間をミリ秒単位で指定します。この項目で指定した時間が経過するまでは、子データ変換を実行しません。ただし、この項目で指定した時間内にすべてのレコードの入力が完了した場合は、その時点で子データ変換を開始します。 この項目で指定した時間内に入力されたレコードをまとめて、1 つのレコード群として子データ変換に引き渡します。指定した時間内にすべてのレコードの入力が完了しなかった場合は、レコード群は複数になります。子データ変換は、すべてのレコード群に対する処理が完了するまで、繰り返します。

注※ どれか 1 つしか設定できません。そのほかの値は空欄にする必要があります。

(4) [結果行]タブ

子データ変換の行送信ステップから引き渡されたデータを出力するステップおよびフィールド情報を指定します。

図 2-110 [トランスフォーメーション実行]画面-[結果行]タブ

トランスフォーメーション実行

ステップ名

トランスフォーメーション実行

Transformation:

Browse...

パラメーター

実行結果

行のグルーピング

結果行

結果ファイル

出力行のソースステップ

結果行の予想レイアウト:

フィールド名	データタイプ	長さ	Precision

? ヘルプ

OK(O)

キャンセル(C)

表 2-112 「トランスフォーメーション実行」画面-「結果行」タブの項目

項番	項目名	説明
1	[出力行のソースステップ]	引き渡されたデータを出力するステップの名前を指定します。空欄の場合、結果を出力しません。
2	[フィールド名]	フィールド名を指定します。
3	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。 【注意事項】 子データ変換から受け取った結果のフィールド情報が、この項目で指定したデータタイプと一致しない場合は、原則としてデータを参照できません。ただし、データタイプによっては、一致していなくてもデータが参照できてしまうため、注意してください。

項番	項目名	説明
4	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
5	[Precision]	該当フィールドのデータタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。 それ以外のデータタイプの場合は、この項目を指定しても無効です。

(5) [結果ファイル]タブ

メモリに渡されたファイル名のリストを取得します。

図 2-111 [トランスフォーメーション実行]画面-[結果ファイル]タブ

トランスフォーメーション実行

ステップ名
トランスフォーメーション実行

Transformation:
Browse...

パラメーター 実行結果 行のグルーピング 結果行 結果ファイル

結果ファイル情報の送信先ステップ
▼

結果ファイル名フィールド
FileName

? ヘルプ OK(O) キャンセル(C)

表 2-113 [トランスフォーメーション実行]画面-[結果ファイル]タブの項目

項番	項目名	説明
1	[結果ファイル情報の送信先ステップ]	データを出力するステップの名前を指定します。空欄の場合、結果を出力しません。
2	[結果ファイル名フィールド]	データを出力するフィールドの名前を指定します。

2.6.8 [フロー]フィルター (Filter rows)

指定した条件式に従って、入力レコードを異なる出力ストリームに出力します。

図 2-112 [フィルター]画面

表 2-114 [フィルター]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[条件式が真(True)のときに実行する処理]	条件式が真の時にレコードを出力するステップを指定します。
3	[条件式が偽(False)のときに実行する処理]	条件式が偽の時にレコードを出力するステップを指定します。
4	[フィルタ条件]	出力先を決定するための条件式を記述します。 条件式を記述するときには使用できる論理演算子は、NOT、AND、OR、AND NOT、OR NOT、XOR です。 右辺に値を指定する場合は、[フィルタ条件]の「<value>」をクリックして[入力値]ダイアログを表示します。[入力値]ダイアログでは、[タイプ]、[値]、[変換形式]、[長さ]、[精度]を指定できます。 フィルターステップの条件式に指定できる比較演算子については、「フィルターステップの条件式に指定できる比較演算子」を参照してください。 【注意事項】 AND NOT と OR NOT は右辺の式に否定が付いた AND と OR です。NAND、NOR ではありませんので、ご注意ください。

項番	項目名	説明
		- エラーが発生するおそれがあるため、条件式の左辺と右辺のデータタイプは一致させてください。 [入力値]ダイアログで[タイプ]に BigInteger 型を指定すると、[変換形式]のプルダウンに書式が表示されません。 [入力値]ダイアログで[タイプ]に Internet Address 型の値を選択すると、コンバートエラーが発生します。 [入力値]ダイアログで[タイプ]に Timestamp 型を指定したあとにデータ変換を保存して開き直し、フィルタステップの設定画面を開くとエラーが発生します。
5	[状態追加]	フィルタ条件を追加します。 条件を追加するには + を押します。 追加した条件は、元の条件をサブレベルの条件に変換します。サブレベルの条件を押すと、条件ツリーの 1 レベル下に移動して編集できます。

表 2-115 フィルターステップの条件式に指定できる比較演算子

項番	比較演算子	説明
1	=	左辺と右辺が同値であれば真を返します。
2	<>	左辺と右辺が異なる値であれば真を返します。
3	<	Integer、Number、BigInteger の場合、左辺が右辺よりも小さければ真を返します。 Date、Timestamp の場合、左辺の時間が右辺の時間よりも前であれば真を返します。 例えば「2021/04/01<2021/04/02」の場合は、真を返します。
4	<=	Integer、Number、BigInteger の場合、左辺が右辺以下であれば真を返します。 Date、Timestamp の場合、左辺の時間が右辺の時間以前であれば真を返します。
5	>	Integer、Number、BigInteger の場合、左辺が右辺よりも大きければ真を返します。 Date、Timestamp の場合、左辺の時間が右辺の時間よりも後であれば真を返します。
6	>=	Integer、Number、BigInteger の場合、左辺が右辺以上であれば真を返します。 Date、Timestamp の場合、左辺の時間が右辺の時間以後であれば真を返します。
7	REGEXP	左辺が検索対象、右辺が正規表現の検索文字列として検索され、マッチすれば真を返します。入力データタイプに関わらず、すべて文字列として処理されます。
8	IS NULL	左辺が NULL 値であれば真を返します。
9	IS NOT NULL	左辺が NULL 値以外であれば真を返します。
10	IN LIST	左辺を検索対象、右辺が検索文字列のリストとして、リスト内の値と一致する値が左辺に存在する場合、真を返します。 右辺のリストはセミコロン区切りで指定します。 指定例を次に示します。 aaa;bbb;ccc
11	CONTAINS	左辺を検索対象、右辺が検索文字列として、対象に検索文字列が含まれている場合、真を返します。

項番	比較演算子	説明
12	STARTS WITH	左辺が検索対象、右辺が検索文字列として、対象が検索文字列から始まっている場合、真を返します。
13	ENDS WITH	左辺が検索対象、右辺が検索文字列として、対象が検索文字列で終わっている場合、真を返します。
14	LIKE	右辺の検索文字列の「%」を「.*」、「?」を「.」に置換して、REGEXP と同様に処理します。
15	TRUE	真を返します。

2.6.9 [フロー]停止 (Abort)

しきい値を超えた入力があった場合、データ変換を停止します。

図 2-113 [停止]画面

① 停止

ステップ名
停止

Options

☒ Abort the running transformation

☐ Abort and log as an error

☐ Stop input processing

しきい値
0

Logging

メッセージ

☒ ログに出力する

ヘルプ OK(O) キャンセル(C)

表 2-116 [停止]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Abort the running transformation]	[しきい値]を超えた入力があった場合にこの項目を選択すると、データ変換を停止します。データ変換のステータスは正常終了となります。
3	[Abort and log as an error]	[しきい値]を超えた入力があった場合にこの項目を選択すると、データ変換を停止します。データ変換のステータスは異常終了となります。
4	[Stop input processing]	[しきい値]を超えた入力があった場合にこの項目を選択すると、入力を停止し、入力されたデータは最後まで処理されます。データ変換のステータスは正常終了となります。
5	[しきい値]	この項目に指定した値を超えた入力があった場合に、データ変換を停止します。 例えば、[しきい値]に 0 を指定すると、最初の行が処理されたあと、停止ステップによってデータ変換を停止します。 [しきい値]に 5 を指定すると、6 行目の入力時にデータ変換を停止します。 デフォルト値は 0 です。 【注意事項】 [しきい値]に 2,147,483,647 を指定すると、データ変換は停止されません。 2,147,483,648 以上の値を指定すると、データ変換実行時にエラーが発生します。
6	[メッセージ]	データ変換が停止された際に出力するログメッセージを指定します。 指定しない場合、次のメッセージが出力されます。 「Aborting after having seen ## rows.」 (凡例) ##: [しきい値]に指定した値+1 例えば、[しきい値]に 3 を指定した場合、次のメッセージが表示され、4 行目以降のデータ変換が停止されます。 Aborting after having seen 4 rows.
7	[ログに出力する]	この項目にチェックを付けると、入力行ごとにログが出力されます。

2.6.10 [フロー]優先順位 (Prioritize streams)

後続のステップに出力する順序を定義します。[先行ステップ取得]ボタンを押すと、前のすべてのステップの名前を自動入力できます。

図 2-114 [優先ストリーム]画面

優先ストリーム

ステップ名

優先順位

優先ステップ名: + --> -

#	ステップ名
1	

ヘルプ

OK(O)

先行ステップ取得

キャンセル(C)

表 2-117 [優先ストリーム]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[優先ステップ名]の[ステップ名]	後続のステップに出力する順序を指定します。リストの一番上にある名前は、それより下にある名前よりも優先されます。

2.6.11 [フロー]条件分岐 (Switch/case) (Switch / case)

キーフィールドの値で、入力レコードの出力先を振り分けます。

図 2-115 [条件分岐 (Switch/Case))]画面

条件分岐 (Switch/Case)

ステップ名

条件分岐(Switch / case)

フィールド名

比較を含む文字列を使用

データタイプ

None

検索文字列

桁区切り文字

数値区切り文字

値

#	値	ステップ名
1		

ステップ名 (デフォルト)

ヘルプ

OK(O)

キャンセル(C)

表 2-118 [条件分岐 (Switch/Case)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	データ振り分けのキーとなるフィールドを指定します。
3	[比較を含む文字列を使用]※	この項目にチェックを付けると、部分一致で文字列の値を評価します。そのため、項目名と実際の機能が異なります。 チェックを付けない場合は、完全一致で文字列の値を評価します。 キーフィールドのデータタイプが String 以外の場合は、エラーの原因となるため、チェックを付けないでください。
4	[データタイプ]	キー値のデータタイプを指定します。 キーフィールドのデータタイプと一致させてください。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。 【注意事項】 Internet Address 型を指定して実行すると、エラーが発生します。
5	[検索文字列]	キー値の書式を設定します。項目名と実際の機能が異なります。 日付データ型 (Data、Timestamp)、数値データ型 (Integer、Number、BigNumber) でだけ有効な項目です。 キーフィールドの書式と合わせてください。
6	[桁区切り文字]	「, (コンマ)」など、桁を区切る文字を指定します。 キーフィールドの桁区切り文字と合わせてください。
7	[数値区切り文字]	「. (ドット)」など、小数点を表す文字を指定します。 キーフィールドの数値区切り文字と合わせてください。
8	[値]	キー値を設定します。
9	[ステップ名]	出力先を設定します。
10	[ステップ名(デフォルト)]	キーフィールドの値が、[値]に設定したすべての値に一致しない場合の出力先を指定します。

注※ 画面の項目名と指定する内容が異なります。

2.6.12 [フロー]空ストリーム探知 (Detect empty stream)

入力ストリームのデータが 0 行の場合は、入力ストリームのフィールド定義ですべての値が NULL のレコードを 1 行出力します。1 行以上の場合は、データを出力しません。このステップは、入力行が 0 件の場合に後続のレコードでエラーを発生させないために使用できます。

図 2-116 [空ストリーム探知]画面

表 2-119 [空ストリーム探知]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

2.7 スクリプト

- [スクリプト]Java スクリプト (Modified Java Script value)
- [スクリプト]Python Executor (Python Executor)
- [スクリプト]SQL 文実行 (Execute SQL script)
- [スクリプト]SQL 文実行(行) (Execute row SQL script)
- [スクリプト]ユーザー定義 Java クラス (User defined java expression)
- [スクリプト]数式 (Formula)
- [スクリプト]正規表現 (Regex evaluation)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.7.1 [スクリプト]Java スクリプト (Modified Java Script value)

データ変換するスクリプトを作成するためのユーザーインターフェースを提供します。画面の JavaScript に設定したコードを、入力ストリームの各レコードに対して実行します。

Pentaho で JavaScript を実行するときに重要な変数を次に示します。

表 2-120 JavaScript 実行時に重要な変数

変数名	説明
trans_Status	次の定数を指定することで、各レコードの出力処理のタイプを変更します。 • CONTINUE_TRANSFORMATION デフォルトのステータスです。入力に対して、入力フィールドに出力フィールドを加えたレコードを後続のステップに出力します。 • SKIP_TRANSFORMATION 入力に対して、後続のステップに結果を出力しません。 • ERROR_TRANSFORMATION このステータスが設定されているレコードを読み込んだ時点で、ステップを実行しているデータ変換を異常終了として停止します。 • ABORT_TRANSFORMATION このステータスが設定されているレコードを読み込んだ時点で、ステップを実行しているデータ変換を正常終了として停止します。
TransformationName	実行しているデータ変換の名前の文字列が格納されています。
step	実行中のステップの Java クラス 「org.pentaho.di.trans.steps.scriptvalues_mod.ScriptValuesMod」のインスタンスが格納されています。
rowMeta	処理中のレコードのメタ情報を格納する Java クラス「org.pentaho.di.core.row.RowMeta」のインスタンスが格納されています。各入力フィールドのデータタイプなどを参照できます。

変数名	説明
row	処理中のレコードのデータ部が Object[]型で格納されています。

図 2-117 [Java スクリプト Mod]画面



表 2-121 [Java スクリプト Mod]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Java スクリプト関数]	[スクリプト]、[定数]、[関数]、[入力フィールド]、および[出力フィールド]のツリービューが含まれています。 [Java スクリプト関数]をダブルクリックすると、[定数]、[関数]、[入力フィールド]、[出力フィールド]のオブジェクトを挿入できます。ドラッグ&ドロップでもオブジェクトを挿入できます。
3	[スクリプト]	[Java スクリプト]に記述されたスクリプトが表示されます。
4	[定数]	trans_Status 変数に設定する定数を[Java スクリプト]に挿入します。ダブルクリックすると次の定数を挿入できます。 • SKIP_TRANSFORMATION • ERROR_TRANSFORMATION • CONTINUE_TRANSFORMATION
5	[関数]	選択した関数を[Java スクリプト]に挿入します。ダブルクリックすると次の関数を挿入できます。 使用できる関数の種類と詳細の参照先を次に示します。 • 文字列関数 • 数学関数 • 日付関数 • 論理関数 • 特殊関数 • ファイル操作関数
6	[入力フィールド]	前ステップで設定したフィールドが表示されます。

項番	項目名	説明
7	[出力フィールド]	出力フィールドが表示されます。
8	[Java スクリプト]	定義領域にコードを記述します。 [Java スクリプト]の定義領域のタブを右クリックすると次のメニューが表示されます。 • [新規追加] 新しいタブを追加します。 • [コピー追加] 既存のスクリプトのコピーを新しいタブに追加します。 • [変換スクリプト設定] 各入力行に対して実行する変換スクリプトを指定します。変換スクリプトを設定しない場合、デフォルトで表示されているタブが変換スクリプトになります。変換スクリプトとして設定できるタブは1つだけです。 • [開始スクリプト設定] 最初の行を処理する前に実行するスクリプトを指定します。 • [スクリプト終点設定] 最後の行が処理された後に実行するスクリプトを指定します。 • [スクリプトタイプの削除] 変換スクリプトを実行しないように指定します。このメニューを実行しても、タブは削除されません。※
9	[互換モード]	バージョン 2.5 の JavaScript エンジンを使用する場合は、この項目にチェックを付けます。 バージョン 3 の JavaScript エンジンを使用する場合は、チェックを付ける必要はありません。
10	[最適化レベル]	JavaScript 最適化のレベルを選択します。選択できる値は次のとおりです。 • 1 JavaScript は解釈モードで実行されます。 • 0 最適化は実行されません。 • 1-9 すべての最適化が実行されます。デフォルトの最適化は9です。9の場合、より高速なスクリプト実行で最適化されますが、コンパイルは遅くなります。
11	[フィールド]	ここで設定したフィールドを次ステップに渡します。
12	[フィールド名]	フィールド名を指定します。
13	[フィールド名の変更]	フィールド名を変更する場合に指定します。指定できるのは[入力フィールド]のフィールド名です。
14	[データタイプ]	フィールドのデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。
15	[長さ]	フィールドの長さを指定します。
16	[精度]	フィールドの精度を指定します。

項番	項目名	説明
17	[フィールド名もしくは、名前変更の値を置換]	[フィールド名]に指定したフィールドの値を別の値に置き換えるか、またはフィールドの名前を変更するかを指定します。詳細は次のとおりです。 • [Y]を指定した場合 [フィールド名]に指定したフィールドの値を変更、または[フィールド名の変更]に指定したフィールド名に変更します。 • [N]を指定した場合 [フィールド名の変更]に指定したフィールド名に変更します。ただし、[フィールド名の変更]に[入力フィールド]のフィールド名が指定されている場合は、[N]を指定してもフィールド名は変更されません。この場合は、別の名前のフィールドが新規に作成されます。 【注意事項】 [互換モード]にチェックが付いている場合、[Y]を指定するとエラーになります。

注※ タブを削除するには、閉じる[X]ボタンをクリックし、表示されたメッセージダイアログで[はい]を選択します。

(1) 文字列関数

関数	説明	引数の値
date2str(<引数 1 >)	ローカライズされた形式を使用して、Date 型を文字列型に変換します。	1. Date
date2str(<引数 1 >,<引数 2 >)	ローカライズされた形式を使用して、Date 型を文字列型に変換します。	1. Date 2. 書式
date2str(<引数 1 >,<引数 2 >,<引数 3 >)	Date 型を文字列型に変換します。	1. Date 2. 書式 3. ISO 言語コード 「JA」「EN」などの値です。
date2str(<引数 1 >,<引数 2 >,<引数 3 >,<引数 4 >)	Date 型を文字列型に変換します。	1. Date 2. 書式 3. ISO 言語コード 「JA」「EN」などの値です。 4. タイムゾーン 「JST」「EST」などの値です。
escapeHtml(<引数 1 >)	HTML データをエスケープします。	1. エスケープする HTML データ
escapeSQL(<引数 1 >)	SQL データをエスケープします。	1. エスケープする SQL データ
escapeXml(<引数 1 >)	XML データをエスケープします。	1. エスケープする XML データ
fillString(<引数 1 >,<引数 2 >)	指定された文字で、指定された長さの文字列を作成します。	1. 1 文字指定 2. 作成する文字列の長さ（文字数）
getOcuranceString(<引数 1 >,<引数 2 >)	指定した文字列内に、指定した検索文字列が何回出現するか、出現回数を返します。	1. 検索される文字列 2. 検索文字列

関数	説明	引数の値
indexOf(<引数 1 >,<引数 2 >)	文字列の先頭から検索し、検索文字列の位置を返します。 文字列の位置とは、先頭の文字の位置を「0」とした場合に何番目の位置に該当するのかを示す数字です。	1. 検索される文字列 2. 検索文字列
indexOf(<引数 1 >,<引数 2 >,<引数 3 >)	文字列内で指定された位置から検索し、検索文字列の位置を返します。 文字列の位置とは、先頭の文字の位置を「0」とした場合に何番目の位置に該当するのかを示す数字です。	1. 検索される文字列 2. 検索文字列 3. 検索開始位置
initCap(<引数 1 >)	文字列中の各単語の先頭文字を大文字に、それ以外の文字を小文字に変換します。	1. 文字列
lower(<引数 1 >)	文字列の文字を全て小文字に変換します。	1. 文字列
lpad(<引数 1 >,<引数 2 >,<引数 3 >)	指定の長さよりも文字列が短い場合、文字列の左側に指定された文字を埋め込むことで、文字列を指定の長さに変換します。	1. 文字列 2. 埋め込む文字 3. 長さ
ltrim(<引数 1 >)	文字列の先頭にある、連続する半角空白または連続するタブ文字を削除します。	1. 文字列
num2str(<引数 1 >)	ローカライズされた形式を使用して、数値型を文字列型に変換します。	1. 文字列
num2str(<引数 1 >,<引数 2 >)	ローカライズされた形式を使用して、数値型を文字列型に変換します。	1. 文字列 2. 書式
num2str(<引数 1 >,<引数 2 >,<引数 3 >)	数値型を文字列型に変換します。	1. 文字列 2. 書式 3. ISO 言語コード 「JA」「EN」などの値です。
protectXMLCDATA(<引数 1 >)	文字列を XML の CDATA セクションの形式に変換します。	1. 文字列
removeCRLF(<引数 1 >)	文字列から「CR」と「LF」を削除します。	1. 文字列
replace(<引数 1 >,<引数 2 >,<引数 3 >,<引数 4 >,<引数 5 >)	文字列中の検索文字列を置換文字列に変換します。検索文字列と置換文字列の組み合わせは複数指定できます。	1. 文字列 2. 1 番目の検索文字列 3. 1 番目の置換文字列 4. 2 番目の検索文字列 5. 2 番目の置換文字列
rpadd(<引数 1 >,<引数 2 >,<引数 3 >)	指定の長さよりも文字列が短い場合、文字列の右側に指定された文字を埋め込むことで、文字列を指定の長さに変換します。	1. 文字列 2. 埋め込む文字 3. 長さ
rtrim(<引数 1 >)	文字列の末尾にある、連続する半角空白または連続するタブ文字を削除します。	1. 文字列
str2RegExp(<引数 1 >,<引数 2 >)	正規表現を持つ文字列を、配列に分割します。	1. 文字列 2. 分割の対象とする正規表現

関数	説明	引数の値
substr(<引数 1>,<引数 2>)	文字列内の指定された位置にある文字列を返します。	1. 文字列 2. 文字列位置（先頭の文字の位置を「0」とした場合の位置）
substr(<引数 1>,<引数 2>,<引数 3>)	文字列内の指定された位置を始点として、指定された長さの文字列を返します。	1. 文字列 2. 文字列位置（先頭の文字の位置を「0」とした場合の位置） 3. 長さ
trim(<引数 1>)	文字列の前後にある、連続する半角空白または連続するタブ文字を削除します。	1. 文字列
unEscapeHtml(<引数 1>)	エスケープされた HTML データをアンエスケープします。	1. 文字列
unEscapeXml(<引数 1>)	エスケープされた XML データをアンエスケープします。	1. 文字列
upper(<引数 1>)	文字列の文字を全て大文字に変換します。	1. 文字列

(2) 数学関数

関数	説明	引数の値
abs(<引数 1>)	指定された数値の絶対値を返します。	1. 数値データ
ceil(<引数 1>)	小数点以下を切り上げた Double 型の値を返します。	1. 数値データ
floor(<引数 1>)	小数点以下を切り下げた数値を返します。	1. 数値データ
str2num(<引数 1>)	文字列を数値に変換します。	1. 文字列
str2num(<引数 1>,<引数 2>)	文字列を数値に変換します。	1. 文字列 2. 書式
trunc(<引数 1>)	小数点以下を切り下げた数値を返します。	1. 数値データ

(3) 日付関数

関数	説明	引数の値
dateAdd(<引数 1>,<引数 2>,<引数 3>)	Date 型データに、指定された項目の値を追加します。	1. Date 型データ 2. 値を追加する Date 型データの項目 y：年 m：月 d：日 w：週 wd：営業日 hh：時 mi：分

関数	説明	引数の値
		ss：秒 3. 追加する値
dateDiff(<引数 1>,<引数 2>,<引数 3>)	2 つの Date 型データの、Date 型の項目の差分を返します。	1. Date 型データ 2. Date 型データ 3. 値を比較する Date 型データの項目 y：年 m：月 d：日 w：週 wd：営業日 hh：時 mi：分 ss：秒
getDayNumber(<引数 1>,<引数 2>)	Date 型データについて、指定された項目の値を返します。	1. Date 型データ 2. Date 型データから取得する項目 y：<引数 1>の年の最初の日から数えた経過日数 m：<引数 1>の月の最初の日から数えた経過日数 w：<引数 1>の週の最初の日から数えた経過日数 wm：<引数 1>の曜日がその月の何回目の曜日か
getFiscalDate(<引数 1>,<引数 2>)	会計開始オフセットに指定された日と月を 1 月 1 日と見なして、Date 型データの会計日付を返します。 例えば、Date 型データが「2019/12/1」、会計開始オフセットが「4/1 (01.04)」の場合、「2020/9/1」の Date 型データを返します。	1. Date 型データ 2. 会計開始オフセットを表す日付と月（指定形式：dd.MM） dd：日 MM：月
getNextWorkingDay(<引数 1>)	指定された Date 型データの、翌営業日を返します。	1. Date 型データ
month(<引数 1>)	指定された Date 型データの、月の値を返します。	1. Date 型データ
quarter(<引数 1>)	指定された Date 型データの、クォーターの値を返します。 例えば、1～3 月は「1」、4～6 月は「2」、7～9 は「3」、10～12 は「4」を返します。	1. Date 型データ
str2date(<引数 1>)	ローカライズされた形式を使用して、文字列型を Date 型に変換します。	1. 文字列
str2date(<引数 1>,<引数 2>)	ローカライズされた形式を使用して、文字列型を Date 型に変換します。	1. 文字列 2. 書式

関数	説明	引数の値
str2date(<引数 1>,<引数 2>,<引数 3>)	文字列型を Date 型に変換します。	1. 文字列 2. 書式 3. ISO 言語コード 「JA」「EN」などの値です。
str2date(<引数 1>,<引数 2>,<引数 3>,<引数 4>)	文字列型を Date 型に変換します。	1. 文字列 2. 書式 3. ISO 言語コード 「JA」「EN」などの値です。 4. タイムゾーン 「JST」「EST」などの値です。
truncDate(<引数 1>,<引数 2>)	指定された情報の精度に従って、Date 型データの情報の一部を切り捨てます。 例えば、指定された Date 型データが「2022/06/08 12:35.58.123」、情報の精度に「5」が指定された場合、「2022/01/01 00:00:00.000」を返します。	1. Date 型データ 2. 切り捨てる情報の種類 5：月以下の詳細情報を切り捨てる 4：日以下の詳細情報を切り捨てる 3：時間以下の詳細情報を切り捨てる 2：分以下以下の詳細情報を切り捨てる 1：秒以下以下の詳細情報を切り捨てる 0：ミリ秒を切り捨てる
week(<引数 1>)	Date 型データの値が示す年月日が、その年の何週目にあたるのかを示す数字を返します。	1. Date 型データ
year(<引数 1>)	Date 型データ中の「年」を示す値を返します。	1. Date 型データ

(4) 論理関数

関数	説明	引数の値
isCodepage(<引数 1>,<引数 2>)	指定した文字列が指定の文字コードに従っているかどうかを判定した結果を返します。 • true：従っている • false：従っていない	1. 文字列 2. 文字コード 「UTF-8」「Shift_JIS」などの値です。
isDate(<引数 1>)	指定したデータが Date 型かどうかを判定した結果を返します。 • true：Date 型である • false：Date 型ではない	1. 任意の型のデータ
isEmpty(<引数 1>)	指定した文字列が空文字かどうかを判定した結果を返します。 • true：空文字である	1. 文字列

関数	説明	引数の値
	• false：空文字ではない	
isMailValid(<引数 1 >)	指定した文字列が email 形式かどうかを判定した結果を返します。 • true：email 形式である • false：email 形式ではない	1. 文字列
isNum(<引数 1 >)	指定したデータが有効な数値かどうかを判定した結果を返します。 • true：有効な値である • false：有効な値ではない	1. 数値
isRegExp(<引数 1 >,<引数 2 >)	文字列中に、指定した正規表現に一致する文字列が何個あるのかをカウントして、その数字を返します。 指定した正規表現に一致する文字列が 1 つもない場合は、「-1」を返します。	1. 文字列 2. 正規表現
isRegExp(<引数 1 >,<引数 2 >,<引数 3 >)	文字列中に、指定した両方の正規表現に一致する文字列が何個あるのかをカウントして、その数字を返します。 指定した両方の正規表現に一致する文字列が 1 つもない場合は、「-1」を返します。	1. 文字列 2. 正規表現 3. 正規表現
isWorkingDay(<引数 1 >)	指定した Date 型データの値が営業日かどうかを判定した結果を返します。 • true：営業日である • false：営業日ではない	1. Date 型データ

(5) 特殊関数

関数	説明	引数の値
Alert(<引数 1 >)	ローカル実行時に、指定した文字列を警告ダイアログに表示します。	1. 文字列
LoadScriptFile(<引数 1 >)	指定した JavaScript ファイルをロードします。 この関数は開始スクリプトで使用する必要があります。 【注意事項】 変換スクリプトで使用すると、JavaScript ファイルが各処理行にロードされます。	1. JavaScript ファイルのパス
LoadScriptFromTab(<引数 1 >)	指定したタブに定義されているスクリプト関数をロードします。 この関数は開始スクリプトで使用する必要があります。 変換スクリプトで使用すると、関数は各処理行にロードされます。	1. タブ名
LuhnCheck(<引数 1 >)	指定したクレジットカードなどのカード番号の形式が正しいかどうかを Luhn アルゴリズムで判定した結果を返します。 • true：正しい	1. カード番号

関数	説明	引数の値
	• false : 誤っている	
appendToFile(<引数 1 >,<引数 2 >)	ファイルパスに指定されたファイルに、文字列を追加します。 指定されたファイルパスのファイルが存在しない場合は、ファイルパスの場所にファイルを作成して、文字列を書き込みます。	1. ファイルパス 2. 追加する文字列
decode(<引数 1 >,<引数 2 >,<引数 3 >)	文字列が検索文字列と一致する場合は、戻り値の文字列を返します。 文字列が検索文字列と一致しない場合は、文字列をそのまま返します。	1. 文字列 2. 検索文字列 3. 戻り値
decode(<引数 1 >,<引数 2 >,<引数 3 >,<引数 4 >)	文字列が検索文字列と一致する場合は、戻り値（一致した場合）の文字列を返します。 文字列が検索文字列と一致しない場合は、戻り値（一致しなかった場合、省略可）を返します。ただし、戻り値（一致しなかった場合、省略可）が指定されていない場合は、検索対象の文字列をそのまま返します。	1. 文字列 2. 検索文字列 3. 戻り値（一致した場合） 4. 戻り値（一致しなかった場合、省略可）
decode(<引数 1 >,<引数 2 >,<引数 3 >,<引数 4 >,<引数 5 >,<引数 6 >)	文字列が 1 番目または 2 番目の検索文字列と一致する場合は、戻り値（一致した場合）の文字列を返します。 文字列が 1 番目の検索文字列と一致しない場合は、2 番目の検索文字列を調べます。 1 番目、2 番目のどちらの検索文字列とも一致しない場合は、戻り値（2 番目の検索文字列と一致しなかった場合、省略可）を返します。ただし、戻り値（2 番目の検索文字列と一致しなかった場合、省略可）が指定されていない場合は、検索対象の文字列をそのまま返します。	1. 文字列 2. 1 番目の検索文字列 3. 1 番目の戻り値（一致した場合） 4. 2 番目の検索文字列 5. 2 番目の戻り値（一致した場合） 6. 戻り値（2 番目の検索文字列と一致しなかった場合、省略可）
execProcess(<引数 1 >)	外部のプログラムを実行した応答結果を返します。	1. プログラムパス
fireToDB(<引数 1 >,<引数 2 >)	SQL の SELECT 文を実行した結果を 2 次元配列で返します。 1 番目の次元は行番号、2 番目の次元は列番号を表すため、1 行目の 1 列目の配列名は[0][0]です。 例： <pre>var strConn = "XXXXX"; var strSQL = "SELECT COUNT(*) FROM yyyyy"; var xArr = fireToDB(strConn, strSQL); f00=xArr[0][0]; Alert(f00);</pre>	1. データベース接続名 2. SQL 文
getDigitsOnly(<引数 1 >)	文字列中の数字だけを抽出して返します。 例えば、文字列「abc123cde4fg5」の場合、「12345」を返します。	1. 文字列

関数	説明	引数の値
getEnvironmentVar(<引数 1 >)	PDI のシステム変数の値を取得します。	1. システム変数名
getProcessCount(<引数 1 >)	現在処理しているデータの中に、処理タイプに指定したタイプの処理の対象となっている行が何行含まれているか、行数を返します。	1. 処理のタイプ u：更新 i：挿入 w：書き込み r：読み込み o：出力
getTransformationName()	データ変換名を返します。	引数なし
getVariable(<引数 1 >,<引数 2 >)	PDI の変数値を取得します。 変数名が存在しない場合、<引数 2 >の文字列を返します。	1. 変数名 2. 変数名が存在しない場合に返す文字列
print(<引数 1 >)	デバックのために、文字列を改行なしで標準出力に出力します。	1. 文字列
println(<引数 1 >)	デバックのために、文字列を改行ありで標準出力に出力します。	1. 文字列
removeDigits(<引数 1 >)	文字列から数字をすべて削除した値を返します。 例えば、文字列「abc123cde4fg5」の場合、「abccdefg」を返します。	1. 文字列
resolveIP(<引数 1 >,<引数 2 >)	IP アドレスまたはホスト名を返します。 - IP アドレスからホスト名を取得する場合 <引数 1 >に IP アドレスを指定し、<引数 2 >に「IP」を指定します。 - ホスト名から IP アドレスを取得する場合 <引数 1 >にホスト名を指定し、<引数 2 >に「IP 以外の文字列」を指定します。	1. IP アドレスまたはホスト名 2. 「IP」または「IP 以外の文字列」
sendMail(<引数 1 >,<引数 2 >,<引数 3 >,<引数 4 >,<引数 5 >)	指定した送信先アドレスにメールを送信します。 【注意事項】 この関数を使用するには、sendmail のソフトウェアが必要です。	1. SMTP サーバ 2. メールを送信元アドレス 3. メールを送信先アドレス 4.メールの件名 5.メールのメッセージテキスト
setEnvironmentVar(<引数 1 >,<引数 2 >)	PDI にシステム変数を設定します。	1. システム変数名 2. 設定値
setVariable(<引数 1 >,<引数 2 >,<引数 3 >)	PDI に変数を設定します。	1. 変数名 2. 設定値 3. スコープ s：system r：root p：parent g：grandparent

関数	説明	引数の値
writeToLog(<引数 1 >)	データ変換実行時のログレベルが次のとおり指定されている場合に、実行結果のログに指定されたメッセージを出力します。 - 詳細なログ - デバッグ - ローレベル（非常に詳細）	1. メッセージ
writeToLog(<引数 1 >,<引数 2 >)	データ変換の実行結果のログのうち、ログタイプで指定したタイプのログに、指定のメッセージを出力します。ただし、データ変換実行時に指定したログレベルによっては、実行結果としてログタイプで指定したタイプのログが出力されません。その場合は、どのログにも指定のメッセージは出力されません。	1. メッセージ 2. ログタイプ d : Debug l : Detail e : Error m : Minimal r : RowLevel

(6) ファイル操作関数

関数	説明	引数の値
copyFile(<引数 1 >,<引数 2 >,<引数 3 >)	コピー元のファイルパスに指定したファイルを、コピー先のファイルパスにコピーします。 コピー先にファイルが存在する場合に、上書きするかどうかを指定します。	1. コピー元のファイルパス 2. コピー先のファイルパス 3. ファイルを上書きするかどうか true : 上書きしてコピーする false : 上書きしない（コピーしない）
createFolder(<引数 1 >)	指定したフォルダを作成します。	1. フォルダパス
deleteFile(<引数 1 >)	指定したファイルを削除します。	1. ファイルパス
fileExists(<引数 1 >)	ファイルパスに指定したファイルが存在するかどうかを判定した結果を返します。 - true : 存在する - false : 存在しない	1. ファイルパス
getFileExtension(<引数 1 >)	指定したファイルの拡張子を返します。	1. ファイルパス
getFileSize(<引数 1 >)	指定したファイルのサイズを long 型で返します。	1. ファイルパス
getLastModifiedTime(<引数 1 >,<引数 2 >)	指定したファイルの更新日時を返します。	1. ファイルパス 2. 書式
getParentFoldername(<引数 1 >)	指定したファイルのフォルダパスを返します。	1. ファイルパス
getShortFilename(<引数 1 >)	指定したファイルのファイル名を返します。	1. ファイルパス
isFile(<引数 1 >)	ファイルパスに指定したファイルが存在し、ファイルかどうかを判定した結果を返します。	1. ファイルパス

関数	説明	引数の値
	• true：ファイルである • false：指定したファイルが存在しない、または存在するがファイルではない	
isFolder(<引数 1 >)	フォルダパスに指定したフォルダが存在し、フォルダかどうかを判定した結果を返します。 • true：フォルダである • false：指定したフォルダが存在しない、または存在するがフォルダではない	1. フォルダパス
loadFileContent(<引数 1 >)	指定したファイルの内容を出力します。	1. ファイルパス
loadFileContent(<引数 1 >,<引数 2 >)	指定したファイルの内容を出力します。	1. ファイルパス 2. 文字コード 「UTF-8」「Shift_JIS」などの値です。
moveFile(<引数 1 >,<引数 2 >,<引数 3 >)	変更前のファイルパスで指定したファイルの名称を、変更後のファイルパスで指定した名称に変更します。 同じフォルダ内に変更後のファイルパスで指定した名称と同じファイルが存在する場合に、ファイル名を変更して上書きするかどうかを指定します。	1. 変更前のファイルパス 2. 変更後のファイルパス 3. 同じフォルダ内に変更後のファイル名と同じ名称のファイルが存在する場合の処理 true：ファイル名を変更して上書きする false：ファイル名を変更しない
touch(<引数 1 >)	指定したファイルを空データで作成します。	1. ファイルパス

2.7.2 [スクリプト]Python Executor (Python Executor)

先行のステップから送信された入力データを Python スクリプトの変数として使用し、Python スクリプトを実行します。また、Python スクリプトの変数を後続のステップへ出力します。

(1) [Script]タブ

Python スクリプトに関する設定をします。

図 2-118 [Python Executor]画面-[Script]タブ

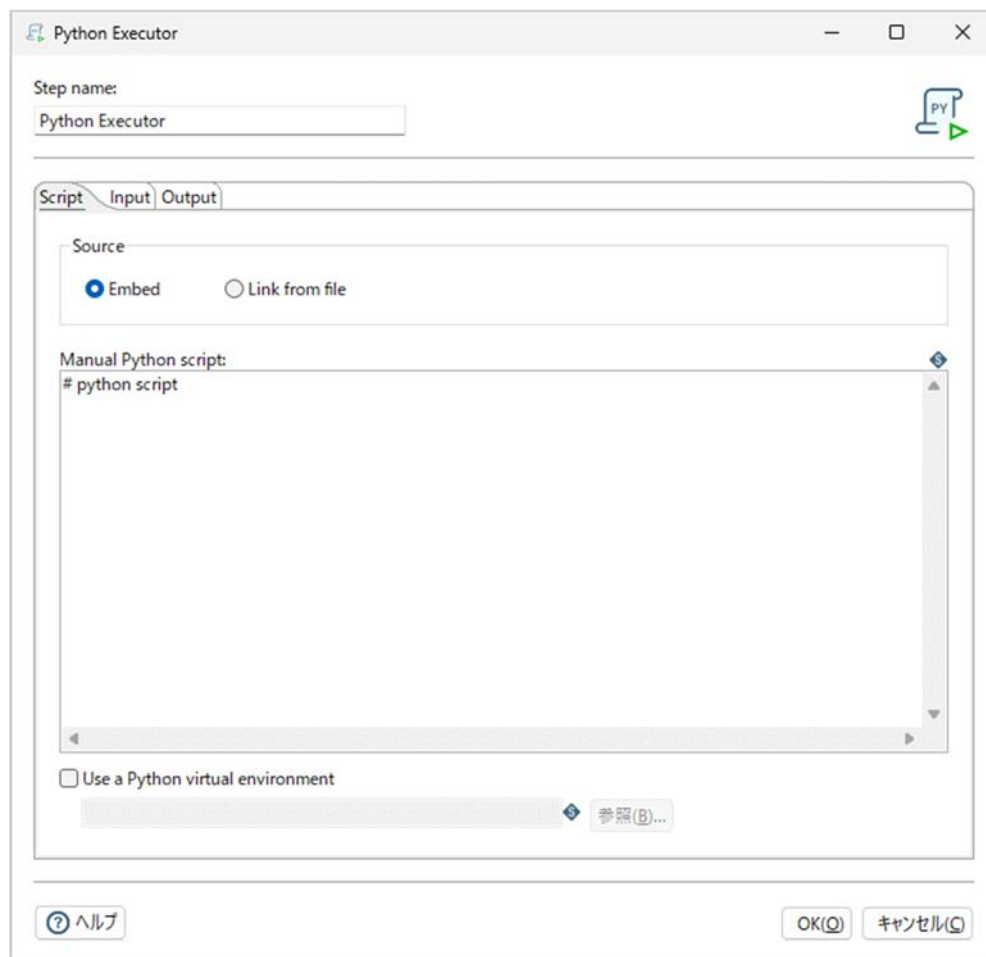


表 2-122 [Python Executor]画面-[Script]タブの項目

項番	項目名	説明
1	[Step name]	ステップ名を指定します。
2	[Embed]	[Manual Python script]に Python スクリプトを記述する場合は、この項目を選択します。
3	[Link from file]	Python スクリプトが記述されているファイルを読み込む場合は、この項目を選択します。
4	[Manual Python script]	定義領域に Python スクリプトを記述します。[Embed]を選択すると指定できます。
5	[Location]	Python スクリプトのファイルが配置されているファイルシステムまたはクラスターを、次の項目から指定します。 • [Local] PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照する場合に、指定します。 • [Hadoop Cluster] Hadoop Cluster で管理されているファイルシステムを参照する場合に、指定します。 • [S3N]

項番	項目名	説明
		• [S3A] Amazon S3 で管理されているファイルシステムを参照する場合に、指定します。 • [HDFS] HDFS 上のファイルを参照する場合に、指定します。 • [MapRFS] MapRFS 上のファイルを参照する場合に、指定します。 • [Azure Data Lake Gen 2] • [Azure Data Lake Gen 1] • [Snowflake Staging] • [HCP] HCP (Hitachi Content Platform) で管理されているファイルシステムを参照する場合に、指定します。 • [Google Cloud Storage] Google Cloud Storage で管理されているファイルシステムを参照する場合に、指定します。 [Link from file]を選択すると指定できます。
6	[File name]	Python スクリプトが記述されているファイルを指定します。[参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。 [Link from file]を選択すると指定できます。
7	[Use a Python virtual environment]	複数のバージョンの Python がインストールされる場合にこの項目にチェックを付けると、テキストボックスに特定のバージョンの実行ファイルを指定できます。[参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。 チェックを付けない場合は、システムのデフォルトで設定されている Python のバージョンが使用されます。

(2) [Input]タブ

先行のステップから入力されたフィールドについて設定します。先行のステップからの入力がない場合、このタブの項目は設定できません。[Get fields]ボタンを押すと、入力データのフィールドと PDI 上のフィールドを入力できます。

図 2-119 [Python Executor]画面-[Input]タブ

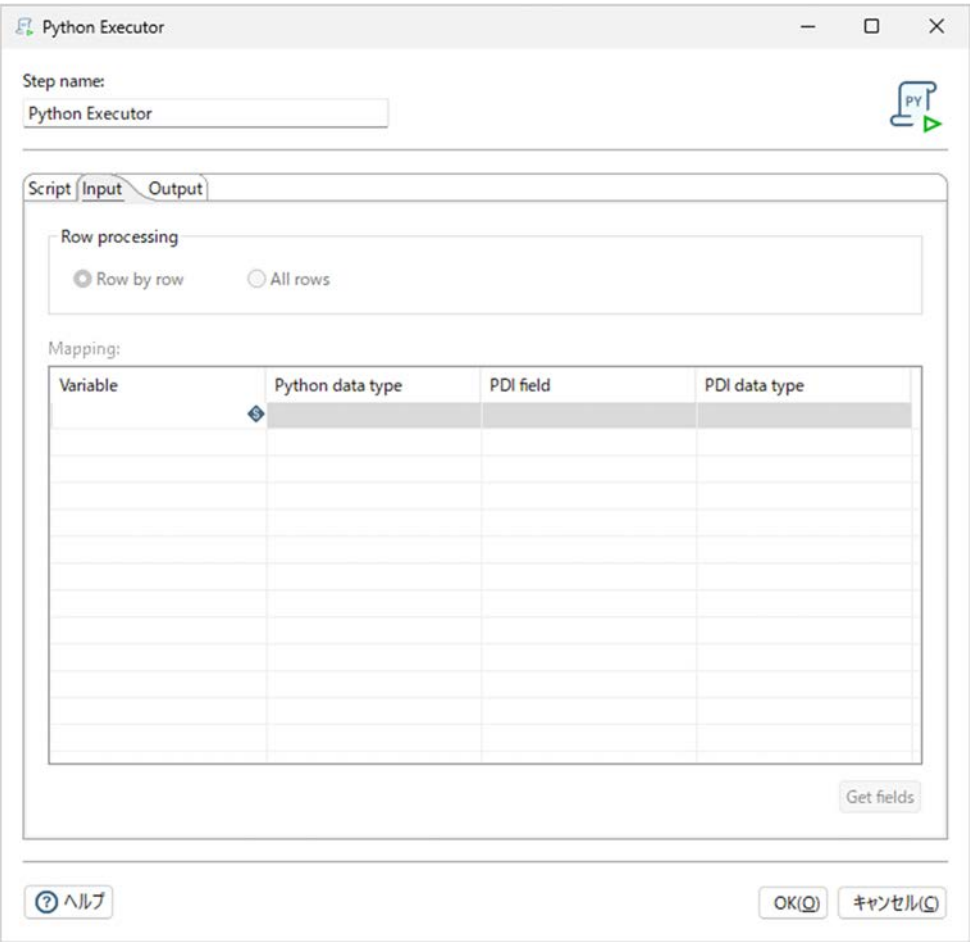


表 2-123 [Python Executor]画面-[Input]タブの項目

項番	項目名	説明
1	[Row by row]	入力データを行ごとに処理する場合は、この項目を選択します。
2	[All rows]	すべての行を一括処理する場合は、この項目を選択します。この項目を選択した場合、入力データは構造化されます。
3	[Variable]	入力データの、Python 変数として使用する変数名を入力します。 [Row by row]を選択すると指定できます。
4	[Python data type]	Python 変数として使用する変数のデータタイプを、次の項目から指定します。 • [str] • [int] • [float] • [datetime] • [bool] [Row by row]を選択すると指定できます。
5	[PDI field]	Python 変数をマッピングするための PDI 上のフィールドを入力します。
6	[PDI data type]	PDI 上のデータタイプを指定します。

項番	項目名	説明
		PDI データタイプと Python データタイプの対応については、「 データタイプの対応表（PDI から Python ヘデータを渡す場合） 」を参照してください。
7	[Available variables]	デフォルトは「variable_1」が登録されています。 [+]アイコンをクリックするとスクリプトで使用する Python 変数を追加します。 [×]アイコンをクリックすると、Python 変数を削除します。 [All rows]を選択すると指定できます。
8	[Variable name]	[Available variables]で選択されている Python 変数の名前を変更します。変更後は、[Available variables]のリストに反映されます。 [All rows]を選択すると指定できます。
9	[Step]	先行のステップのステップの名前を選択します。 先行のステップが複数存在する場合は、どれか 1 つしか選択できません。 [All rows]を選択すると指定できます。
10	[Data Structure]	Python のデータ構造の種類を、次の項目から指定します。 • [Pandas dataframe] 2次元の表形式のデータとして扱う場合に、指定します。 • [NumPy array] 多次元配列のデータとして扱う場合に、指定します。 • [Python List of Dictionaries] 辞書型リストとして扱う場合に、指定します。 [All rows]を選択すると指定できます。
11	[Data structure field]	入力データのフィールド名を入力します。 [All rows]を選択すると指定できます。
12	[Data structure type]	Python 変数として使用する変数のデータタイプを、次の項目から指定します。指定できる値は、[Data Structure]で指定した値によって異なります。 • [object] • [int64] [Data Structure]で[Pandas dataframe]または[Python List of Dictionaries]を選択した場合に指定できます。 • [float64] • [datetime64[ns]] [Data Structure]で[Pandas dataframe]または[Python List of Dictionaries]を選択した場合に指定できます。 • [bool] [All rows]を選択すると指定できます。

表 2-124 データタイプの対応表（PDI から Python ヘデータを渡す場合）

PDI データタイプ	Python データ構造	Python データタイプ	PDI データタイプの説明
BigNumber	Pandas dataframe	float64	任意精度の数値です。

PDI データタイプ	Python データ構造	Python データタイプ	PDI データタイプの説明
Boolean		bool	true または false の真偽値です。
Date		datetime64[ns]	ミリ秒精度の日時値です。
Integer		int64	符号付 64 ビット整数値です。
Number		float64	倍精度浮動小数点値です。
String		object	可変長の無制限のテキストです。
Timestamp		datetime64[ns]	ナノ秒精度の日時値です。
BigNumber	NumPy array	float64	任意精度の数値です。
Boolean		bool	true または false の真偽値です。
Integer		int64	符号付 64 ビット整数値です。
Number		float64	倍精度浮動小数点値です。
BigNumber	Basic Python Data Types	float	任意精度の数値です。
Boolean		bool	true または false の真偽値です。
Integer		int	符号付 64 ビット整数値です。
Number		float	倍精度浮動小数点値です。
String		str	可変長の無制限のテキストです。
Timestamp		datetime	ナノ秒精度の日時値です。

(3) [Output]タブ

後続のステップへ出力するフィールドについて設定します。[Get fields]ボタンを押すとメッセージ「Continuing to "Get fields" will run the Python script. Are you sure you want to continue?」が表示され、[Yes, run]ボタンを押すと Python スクリプトが実行されて Python 変数を追加できます。

図 2-120 [Python Executor]画面-[Output]タブ

Step name:
Python Executor

Script Input Output

Output processing
☒ Variables to fields ☐ Frame to fields

Mapping:

Variable	Python data type	PDI field	PDI data type

Get fields

ヘルプ OK(O) キャンセル(C)

表 2-125 [Python Executor]画面-[Output]タブの項目

項番	項目名	説明
1	[Variables to fields]	Python スクリプトで定義した変数を行ごとに PDI 上のフィールドにマッピングする場合は、この項目を選択します。
2	[Frame to fields]	Python スクリプトで定義した構造化データを PDI 上のフィールドにマッピングする場合は、この項目を選択します。
3	[Variable] ([Mapping]領域)	PDI 上のフィールドにマッピングする Python 変数の変数名を入力します。 [Variables to fields]を選択すると指定できます。
4	[Variable] (テキストボックス)	Python 変数の名前を入力します。 [Frame to fields]を選択すると指定できます。
5	[Python data type]	Python 変数として使用する変数のデータタイプを、次の項目から指定します。 • [str] 文字列 • [int] 整数 • [float] 浮動小数点数

項番	項目名	説明
		• [datetime] 日付 • [bool] ブール [Variables to fields]を選択すると指定できます。 【注意事項】 Python の datetime 型または Python (Pandas dataframe) の datetime64[ns]型から PDI の数値型に変換すると、エラーは発生せず、年号だけが変換されます。 (例) Python の datetime 型の値 2022/10/31 12:34:56 PDI の数値型の値 2022
6	[PDI field]	Python 変数をマッピングするための PDI 上のフィールドを入力します。
7	[PDI data type]	PDI 上のデータタイプを指定します。 PDI データタイプと Python データタイプの対応については、「データタイプの対応表 (Python から PDI へデータを渡す場合)」を参照してください。
8	[Data structure field]	PDI 上のフィールドにマッピングする Python 変数の変数名を入力します。 [Frame to fields]を選択すると指定できます。
9	[Data structure type]	Python 変数として使用する変数のデータタイプを、次の項目から指定します。 • [object] • [int64] • [float64] • [datetime64[ns]] • [bool] [Frame to fields]を選択すると指定できます。 【注意事項 1】 Python の datetime 型または Python (Pandas dataframe) の datetime64[ns]型から PDI の数値型に変換すると、エラーは発生せず、年号だけが変換されます。 (例) Python の datetime 型の値 2022/10/31 12:34:56 PDI の数値型の値 2022 【注意事項 2】 Python (Pandas dataframe) の datetime64[ns]型から PDI の Date 型に変換すると、エラーが発生します。

表 2-126 データタイプの対応表 (Python から PDI ヘデータを渡す場合)

Python データ構造	Python データタイプ	PDI データタイプ	PDI データタイプの説明
Pandas DataFrame	bool	Boolean	true または false の真偽値です。
	datetime64[ns]	Timestamp	ナノ秒精度の日時値です。
	float64	BigNumber	任意精度の数値です。
	int64	Integer	符号付 64 ビット整数値です。
	object	String	可変長の無制限のテキストです。
Basic Python Data Types	bool	Boolean	true または false の真偽値です。
	datetime	Timestamp	ナノ秒精度の日時値です。
	float	BigNumber	任意精度の数値です。
	int	Integer	符号付 64 ビット整数値です。
	str	String	可変長の無制限のテキストです。

2.7.3 [スクリプト]SQL 文実行 (Execute SQL script)

指定した SQL 文を実行します。

図 2-121 [Execute SQL script]画面

Execute SQL script

ステップ名

SQL文実行

データソース名

編集(E)

新規作成(N)...

ウィザード

SQLステートメント

行1列0

行単位で実行

単体構文を実行

変数を使用する

バインドパラメータ

引用文字列

パラメータ

#

フィールド名

1

挿入の統計フィールド名

更新の統計フィールド名

削除の統計フィールド名

読込の統計フィールド名

ヘルプ

OK(O)

キャンセル(C)

フィールドを取得(G)

表 2-127 [Execute SQL script]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	データソース名を指定します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[SQL ステートメント]	実行する SQL 文を指定します。
4	[行単位で実行]	この項目にチェックを付けると、[SQL ステートメント]に指定する SQL 文で、[フィールド名]の値にプレースホルダーを指定できます。プレースホルダーには「? (クエションマーク)」を使用します。 チェックを付けると、次のように動作します。 1. 前のステップからの入力レコードごとに、SQL 文を実行します。

項番	項目名	説明
		2. [SQL ステートメント]に指定した SQL 文にプレースホルダーを使用した場合、前ステップからの入力レコードごとに、パラメータの値を反映して SQL 文を実行します。 3. [バインドパラメータ]および[引用文字列]を使用できます。 チェックを付けない場合、前のステップからの入力レコードに関係なく、[SQL ステートメント]に指定した SQL 文を実行します。
5	[単体構文を実行]	この項目にチェックを付けると、[SQL ステートメント]に指定した複数の SQL 文を、入力レコードごとに単体の SQL 文として実行します。 エラーが発生した場合、入力レコードごとに実行済みの SQL 文をロールバックします。
6	[変数を使用する]	この項目にチェックを付けると、[SQL ステートメント]で変数を使用できます。テーブル名や列名にも変数を使用できます。 変数は、「\${変数}」と記述します。 Ctrl キーとスペースキーを同時に押すと、使用できる変数の一覧が表示されます。
7	[バインドパラメータ]	この項目にチェックを付けると、プリペアドステートメントを使用してパラメータをバインドします。 [行単位で実行]にチェックを付けると指定できます。
8	[引用文字列]	この項目にチェックを付けると、使用するデータベースに応じて文字列の前後に引用符を追加し、CR、LF、引用文字自体などの特殊文字をエスケープします。 前のステップから受け取った入力データの前後に引用符がなくても、引用符を追加して、SQL を実行します。 [行単位で実行]にチェックを付けた場合かつ、[バインドパラメータ]にチェックを付けない場合に指定できます。
9	[フィールド名]	[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。 [SQL ステートメント]に指定したクエリー内の「?」を、この項目に指定したフィールドに格納されている値に置き換えます。 この項目に指定した順序で値を置き換えます。例えば、1 番目に指定したフィールドは、クエリー内の最初の「?」へ置き換え、フィールド名の 2 番目に指定したフィールドは、クエリー内の 2 番目の「?」へ置き換えます。 [SQL ステートメント]に複数のクエリーを指定した場合、次の例のように値を置き換えます。 [SQL ステートメント]に次の 2 つのクエリーを指定する。 <pre>INSERT INTO sample_table VALUES(1, ?, ?); INSERT INTO sample_table VALUES(2, ?, ?);</pre> [フィールド名]に次の値を指定する。 <pre>1 番目 : data1 2 番目 : data2</pre> - クエリーは、次のように値を置き換える。 <pre>INSERT INTO sample_table VALUES(1, [data1の値], [data2の値]); INSERT INTO sample_table VALUES(2, [data1の値], [data2の値]);</pre>

項番	項目名	説明
10	[挿入の統計フィールド名]	実行された SQL 文によって INSERT された行数を格納するフィールドの名前を指定します。
11	[更新の統計フィールド名]	実行された SQL 文によって UPDATE された行数を格納するフィールドの名前を指定します。
12	[削除の統計フィールド名]	実行された SQL 文によって DELETE された行数を格納するフィールドの名前を指定します。
13	[読込の統計フィールド名]	実行された SQL 文によって SELECT された行数を格納するフィールドの名前を指定します。

2.7.4 [スクリプト]SQL 文実行(行) (Execute row SQL script)

前のステップから入力された SQL 文を実行します。または、前のステップから入力された SQL ファイルを読み込んで SQL 文を実行します。

図 2-122 [SQL 文実行 (行)]画面

表 2-128 [SQL 文実行 (行)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	データソース名を指定します。

項番	項目名	説明
		[編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[コミット]	データベースをコミットする行数を指定します。 ただし、指定した行数に関係なく、必ず 1 レコード目でコミットが実行されます。
4	[単一のステートメントを送る]	この項目にチェックを付けると、[SQL フィールド名]で指定したフィールド内の複数の SQL 文を、入力レコードごとに単体の SQL 文として実行します。 エラーが発生した場合、入力レコードごとに実行済みの SQL 文をロールバックします。
5	[SQL フィールド名]	実行する SQL 文が含まれるフィールド名を指定します。 [ファイルから SQL を読む]にチェックを付けた場合、SQL 文が定義されている SQL ファイルパスが含まれるフィールド名を指定します。
6	[ファイルから SQL を読む]	この項目にチェックを付けると、SQL 文が定義されているファイルから SQL 文を実行します。 [SQL フィールド名]で、SQL ファイルパスが含まれるフィールドを指定する必要があります。
7	[insert データフィールド]	実行された SQL 文によって INSERT された行数を格納するフィールドの名前を指定します。
8	[Update データフィールド]	実行された SQL 文によって UPDATE された行数を格納するフィールドの名前を指定します。
9	[Delete データフィールド]	実行された SQL 文によって DELETE された行数を格納するフィールドの名前を指定します。
10	[Read データフィールド]	実行された SQL 文によって SELECT された行数を格納するフィールドの名前を指定します。

2.7.5 [スクリプト]ユーザー定義 Java クラス (User defined java expression)

Java ソースコードを入力し、独自のステップ機能を実行します。ソースコードには、クラス本体 (import、コンストラクター、メソッドなど) が必要です。

ステップ実行時、コンパイル完了後に、Janino プロジェクトライブラリを使用して記述された Java ソースコードを実行します。Java 1.8 を前提としています。

Java のソースコードは、[ユーザー定義 Java クラス]画面の[クラスコード]に定義します。通常は、processRow()メソッドを定義します。processRow()メソッドは、入力ストリームから 1 つの行を読み取り、行の構造やフィールドを変更して、次のステップに渡します。

(1) [フィールド]タブ

このタブに指定したフィールドを次ステップに渡します。

図 2-123 [ユーザー定義 Java クラス]画面-[フィールド]タブ

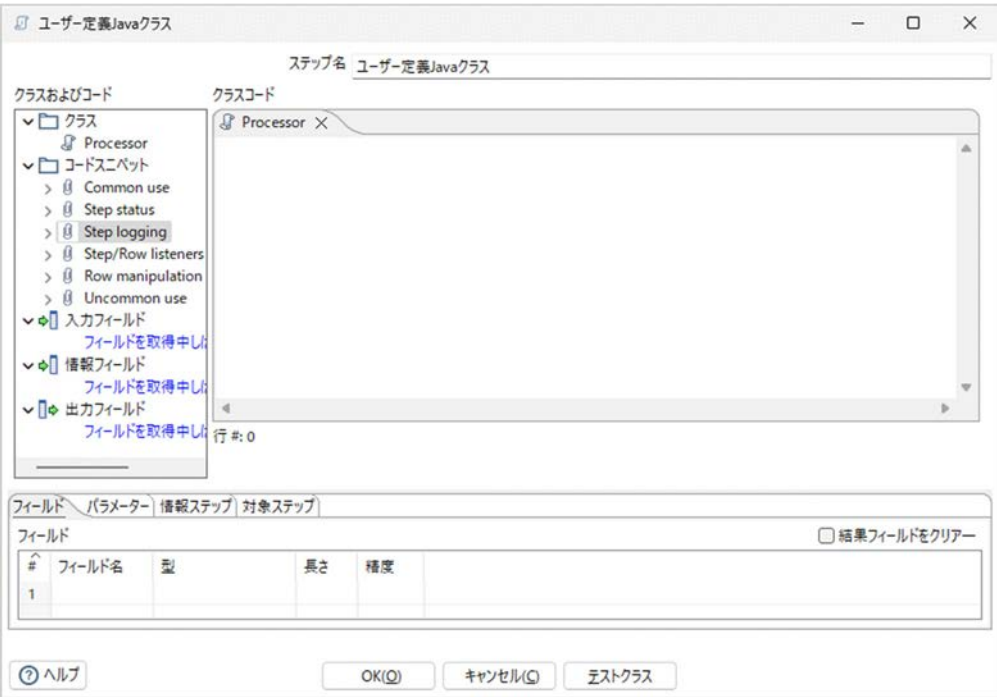


表 2-129 [ユーザー定義 Java クラス]画面-[フィールド]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[クラス]	[クラスコード]に記述したクラスが表示されます。
3	[コードスニペット]	選択したコードテンプレートをダブルクリックすると、そのコードテンプレートが[クラスコード]に挿入されます。
4	[入力フィールド]	前のステップで設定したフィールドが表示されます。
5	[情報フィールド]	[情報ステップ]タブで設定したフィールドが表示されます。
6	[出力フィールド]	[フィールド]タブで設定したフィールドが表示されます。
7	[クラスコード]	クラスコードの[Processor]タブにコードを記述します。右クリックメニューから[新規追加]を選択すると、コードブロックを追加するためのタブを作成します。
8	[結果フィールドをクリアー]	この項目にチェックを付けた場合、このタブに指定したフィールドだけが出力され、入力フィールドのフィールドは出力されません。
9	[フィールド名]	フィールド名を指定します。

項番	項目名	説明
10	[型]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
11	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
12	[精度]	浮動小数点の桁数を指定します。 データタイプが Number と BigNumber の場合に指定できます。

(2) [パラメーター]タブ

このタブで指定した値をクラスコード内で取得できます。次のように指定します。

- 1 行ずつ取得する場合

```
getParameter([タグ名])
```

- 複数行を一括で取得する場合

```
get(Fields.In, [タグ名]);
```

図 2-124 [ユーザー定義 Java クラス]画面-[パラメーター]タブ

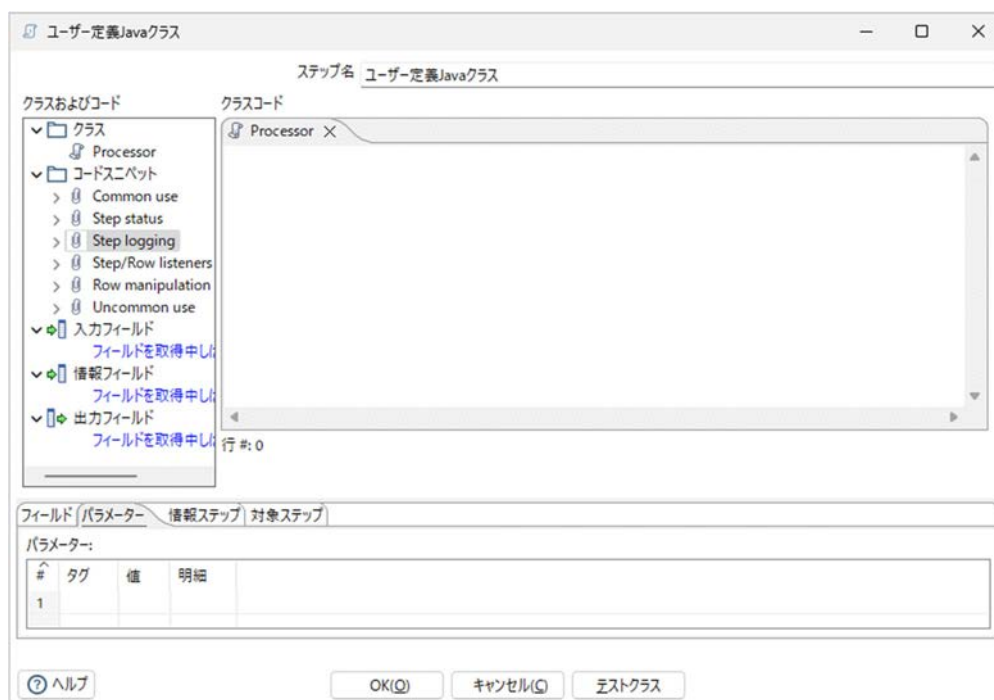


表 2-130 [ユーザー定義 Java クラス]画面-[パラメーター]タブの項目

項番	項目名	説明
1	[タグ]	タグ名を指定します。
2	[値]	値を指定します。
3	[明細]	[タグ]と[値]についての説明を記述します。

(3) [情報ステップ]タブ

検索対象の情報ステップから入力データを取得する際に指定します。

クラスコード内で入力データを取得する際は、次のように指定します。

```
get(Fields.Info, [タグ名])
```

図 2-125 [ユーザー定義 Java クラス]画面-[情報ステップ]タブ

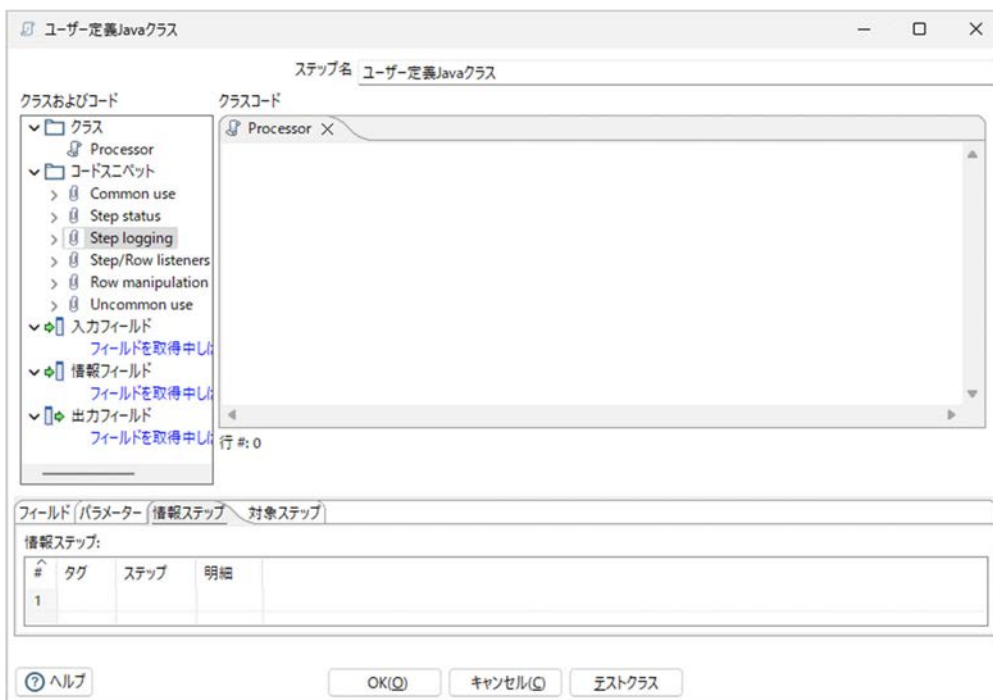


表 2-131 [ユーザー定義 Java クラス]画面-[情報ステップ]タブの項目

項番	項目名	説明
1	[タグ]	タグ名を指定します。
2	[ステップ]	情報ステップを選択します。
3	[明細]	[タグ]と[ステップ]についての説明を記述します。

(4) [対象ステップ]タブ

処理によって、次のステップを振り分ける際に指定します。

クラスコード内では次のように記述します。

```
RowSet rs = findTargetRowSet([タグ名]);
```

図 2-126 [ユーザー定義 Java クラス]画面-[対象ステップ]タブ

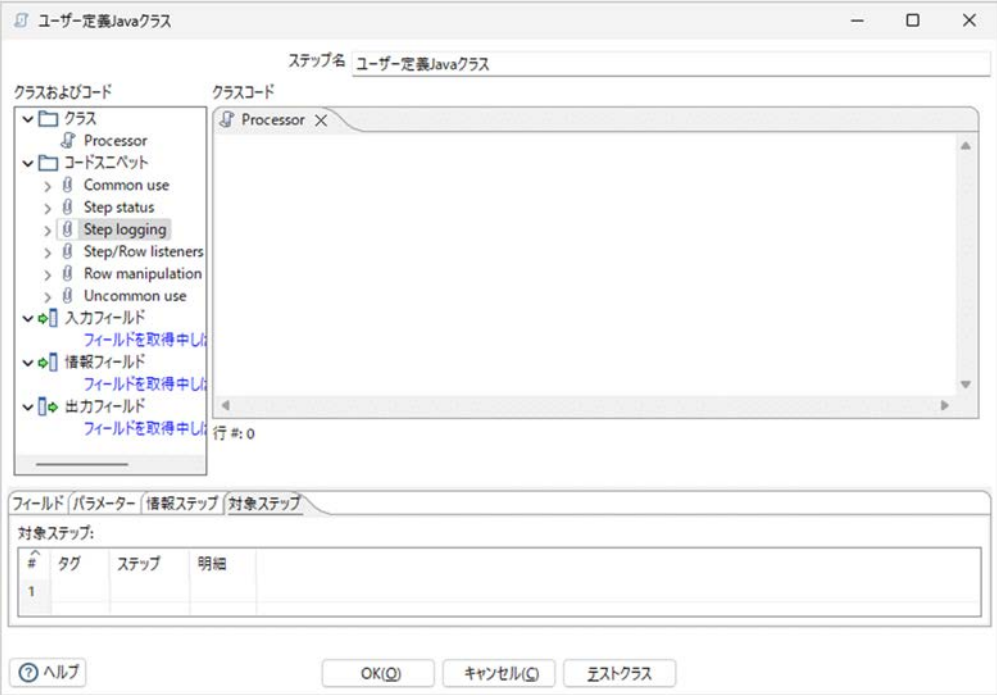


表 2-132 [ユーザー定義 Java クラス]画面-[対象ステップ]タブの項目

項番	項目名	説明
1	[タグ]	タグ名を指定します。
2	[ステップ]	振り分けるステップを選択します。
3	[明細]	[タグ]と[ステップ]についての説明を記述します。

2.7.6 [スクリプト]数式 (Formula)

入力フィールドの値を使用して、設定した数式で計算をします。数値計算だけでなく、文字列処理や、日付データの処理、条件付き演算などもできます。Binary 型、Internet Address 型は正しく計算できないため、数式に使用しないでください。

図 2-127 [数式]画面

fx 数式

ステップ名

数式

フィールド

#	新しいフィールド	数式	データタイプ	長さ	精度	置き換え
1						

ヘルプ

OK(O)

キャンセル(C)

表 2-133 [数式]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[新しいフィールド名]	計算結果を出力するフィールド名を指定します。空欄の場合、エラーが発生します。
3	[数式]	計算に使用する数式を指定します。クリックして表示されるダイアログで数式を指定します。 数式の詳細については、「 数式の詳細 」を参考にしてください。
4	[データタイプ]	出力結果のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
5	[長さ]	データ出力時の長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
6	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
7	[置き換え]	指定したフィールドに値を上書きして、計算結果を出力します。

表 2-134 数式の詳細

項番	カテゴリ	数式	構文	説明
1	Advanced	ARRAYCONCATENATE	ARRAYCONCATENATE({ Array Value }+)	※
2	Advanced	ARRAYCONTAINS	ARRAYCONTAINS(Array array; { Text Value }+)	※

項番	カテゴリ	数式	構文	説明
3	Advanced	ARRAYLEFT	ARRAYLEFT(Array array; Integer numberValues)	※
4	Advanced	ARRAYMID	ARRAYMID(Array array; Integer Start; Integer numberValues)	※
5	Advanced	CSVARRAY	CSVARRAY(Text text; Logical includesQuoting; Text delimiter; Text quoteChar)	※
6	Advanced	CSVTEXT	CSVTEXT(Array array; Logical includesQuoting; Text delimiter; Text quoteChar)	※
7	Basic computation	%	—	百分率を計算します。
8	Basic computation	(	—	式の計算順序を制御するための開きかっこです。
9	Basic computation	)	—	式の計算順序を制御するための閉じかっこです。
10	Basic computation	*	—	掛け算の演算子です。
11	Basic computation	+	—	足し算の演算子です。
12	Basic computation	-	—	引き算の演算子です。
13	Basic computation	/	—	割り算の演算子です。
14	Basic computation	^	—	べき乗の演算子です。
15	Comparisons	<	—	左項の値が右項の値より小さいときに真になる比較演算子です。
16	Comparisons	<=	—	左項の値が右項の値以下のときに真になる比較演算子です。
17	Comparisons	<>	—	左項の値と右項の値が異なるときに真になる比較演算子です。

項番	カテゴリ	数式	構文	説明
18	Comparisons	=	—	左項の値と右項の値が等しいときに真になる比較演算子です。
19	Comparisons	>	—	左項の値が右項の値より大きいときに真になる比較演算子です。
20	Comparisons	>=	—	左項の値が右項の値以上のときに真になる比較演算子です。
21	Database	BEGINSWITH	BEGINSWITH(Text string; Text searchString)	※
22	Database	CONTAINS	CONTAINS(Text string; Text searchString)	※
23	Database	ENDSWITH	ENDSWITH(Text string; Text searchString)	※
24	Database	IN	IN(Text stringToMatch; Text listValue1; Text listValue2; Text listValueN)	※
25	Database	LIKE	LIKE(Text stringToMatch; Text regularExpression)	※
26	Date / Time	DATE	DATE(Integer Year ; Integer Month ; Integer Day)	引数で指定した年、月、日から日付を作成します。
27	Date / Time	DATEIF	DATEDIF(DateParam StartDate ; DateParam EndDate ; Text Format)	引数で指定した日付同士の差を計算します。 引数 Format で指定できる値は、次のとおりです。 • y 年の差を計算します。 • m 月の差を計算します。日付に月がない場合、計算結果は 0 です。 • d 日の差を計算します。 • md 日の差を計算します。なお、年と月の情報は無視されます。 • ym

項番	カテゴリ	数式	構文	説明
				月の差を計算します。なお、年の情報は無視されます。 • yd 日の差を計算します。なお、年の情報は無視されます。
28	Date / Time	DATETIMEVALUE	DATETIMEVALUE(Text D)	※
29	Date / Time	DATEVALUE	DATEVALUE(Text D)	引数で指定した文字列を日付に変換します。
30	Date / Time	DAY	DAY(DateParam Date)	引数で指定した日付の情報から日の情報を取得します。
31	Date / Time	DAYS	DAYS(Date endDate; Date startDate)	※
32	Date / Time	HOUR	HOUR(TimeParam T)	引数で指定した時刻の情報から時の情報を取得します。
33	Date / Time	MINUTE	MINUTE(DateTime)	※
34	Date / Time	MONTH	MONTH(DateParam Date)	引数で指定した日付の情報から月の情報を取得します。
35	Date / Time	MONTHEND	MONTHEND(Date)	※
36	Date / Time	NOW	NOW()	現在の日時を取得します。
37	Date / Time	PARSDATE	PARSEDATE(Text date; Text format; Text locale; Text timezone)	※
38	Date / Time	PREVWEEKDAY	PREVWEEKDAY(Integer weekdayDefinition)	※
39	Date / Time	SECOND	SECOND(DateTime)	※
40	Date / Time	TIME	TIME(Number hours ; Number minutes ; Number seconds)	引数で指定した時、分、秒から時刻を作成します。
41	Date / Time	TIMEEVALUE	TIMEVALUE(Text D)	※
42	Date / Time	TODAY	TODAY()	現在の日付を取得します。
43	Date / Time	WEEKDAY	WEEKDAY(DateParam Date [; Integer Type = 1])	現在の日付から曜日を返します。 引数 Type によって曜日の序数が変わります。 • Type が 1 の場合 日曜日を 1 として始め、土曜日が 7 です。(日月火水木金土) • Type が 2 の場合

項番	カテゴリ	数式	構文	説明
				月曜日を 1 として始め、日曜日が 7 です。(月火水木金土日) • Type が 3 の場合 月曜日を 0 として始め、日曜日が 6 です。(月火水木金土日)
44	Date / Time	YEAR	YEAR(DateParam D)	引数で指定した日付の情報から年の情報を取得します。
45	Date / Time	YESTERDAY	YESTERDAY()	※
46	Information	CHOOSE	CHOOSE(Integer Index ; { Any Value } +)	引数 Index の値を使って、第 2 引数で指定した値のリストから取得する値を決定します。
47	Information	COUNT	COUNT({ Any Value } +)	※
48	Information	COUNTA	COUNTA({ Any Value } +)	※
49	Information	COUNTBLANK	COUNTBLANK({ Any Value } +)	※
50	Information	ERROR	ERROR(Text errorMessage; Text errorCode)	※
51	Information	ISBLANK	ISBLANK(Scalar X)	引数で指定した値が空の場合に真を返します。
52	Information	ISERR	ISERR(Scalar X)	引数で指定した値のタイプが Error かつ NA ではない場合、真を返します。
53	Information	ISERROR	ISERROR(Scalar X)	引数で指定した値のタイプが Error の場合、真を返します。
54	Information	ISEVEN	ISEVEN(Number X)	引数で指定した値が偶数の場合、真を返します。
55	Information	ISLOGICAL	ISLOGICAL(Scalar X)	引数で指定した値のタイプが Logical の場合、真を返します。
56	Information	ISNA	ISNA(Scalar X)	引数で指定した値が NA の場合、真を返します。
57	Information	ISNONTEXT	ISNONTEXT(Scalar X)	引数で指定した値のタイプが Text ではない場合、真を返します。
58	Information	ISNUMBER	ISNUMBER(Scalar X)	引数で指定した値のタイプが Number の場合、真を返します。
59	Information	ISODD	ISODD(Number X)	引数で指定した値が奇数の場合、真を返します。

項番	カテゴリ	数式	構文	説明
60	Information	ISREF	ISREF(Any X)	引数で指定した値のタイプが Reference または ReferenceList の場合、真を返します。
61	Information	ISTEXT	ISTEXT(Scalar X)	引数で指定した値のタイプが Text の場合、真を返します。
62	Information	NA	NA()	定数エラー値 NA を返します。
63	Information	VALUE	VALUE(Text number)	※
64	Logical	AND	AND(condition1 ; condition2 ; condition3)	引数の値の論理積を計算します。
65	Logical	EQUALS	EQUALS(Any param1; Any Param2)	※
66	Logical	FALSE	FALSE()	※
67	Logical	IF	IF(Logical Condition [; [Any IfTrue] [; [Any IfFalse]])	条件を計算します。 • 真の場合 IfTrue で指定された値を返します。 • 偽の場合 IfFalse で指定された値を返します。
68	Logical	IFNA	IFNA(Text string; Text alternateValue)	※
69	Logical	NOT	NOT(Logical L)	引数の値を論理反転します。
70	Logical	OR	OR({ Logical; NumberSequenceList L }+)	引数の値の論理和を計算します。
71	Logical	TRUE	TRUE()	真を返します。
72	Logical	XOR	XOR({ Logical L }+)	引数の値の排他的論理和を計算します。
73	Mathematical	ABS	ABS(Number N)	引数の値の絶対値を返します。
74	Mathematical	ACOS	ACOS(Number N)	※
75	Mathematical	ACOSH	ACOSH(Number N)	※
76	Mathematical	ASIN	ASIN(Number N)	※
77	Mathematical	ATAN	ATAN(Number N)	※
78	Mathematical	ATAN2	ATAN2(Number X; Number Y)	※
79	Mathematical	AVERAGE	AVERAGE({ NumberSequence N }+)	引数の値の平均値を計算します。
80	Mathematical	COS	COS(Number)	※

項番	カテゴリ	数式	構文	説明
81	Mathematical	EVEN	EVEN(Number N)	引数の値を最も近い偶数の整数に切り上げます。
82	Mathematical	EXP	EXP(Number)	※
83	Mathematical	INT	INT(Number N)	引数の値を最も近い整数に切り捨てます。
84	Mathematical	LN	LN(Number)	※
85	Mathematical	LOG10	LOG10(Number)	※
86	Mathematical	MAX	MAX({ NumberSequenceList N })	引数の値の最大値を計算します。
87	Mathematical	MIN	MIN({ NumberSequenceList N })	引数の値の最小値を計算します。
88	Mathematical	MOD	MOD(Number divisor, Number dividend)	※
89	Mathematical	N	N(String)	※
90	Mathematical	ODD	ODD(Number N)	引数の値を最も近い奇数の整数に切り上げます。
91	Mathematical	PI	PI()	※
92	Mathematical	POWER	POWER(Number number, Number power)	※
93	Mathematical	SIN	SIN(Number)	※
94	Mathematical	SQRT	SQRT(Number)	※
95	Mathematical	SUM	SUM({ NumberSequenceList N }+)	引数の値の合計値を計算します。
96	Mathematical	VAR	VAR({ NumberSequence N }+)	※
97	Text	&	Text Left & Text Right	文字列を結合します。
98	Text	ASC	ASC(Text t1)	※
99	Text	CHAR	CHAR(Integer)	※
100	Text	CLEAN	CLEAN(Text)	※
101	Text	CODE	CODE(Text)	※
102	Text	CONCATNATE	CONCATENATE(Text t1 ; Text t2; Text tN)	※
103	Text	DOLLER	DOLLAR(Text text, Integer decimalPlaces)	※

項番	カテゴリ	数式	構文	説明
104	Text	EXACT	EXACT(Text t1 ; Text t2)	大文字と小文字を区別する比較を使用して、2つの文字列が等しいかどうか計算します。
105	Text	FIND	FIND(Text Search ; Text T [; Integer Start = 1])	引数 Search の文字が引数 T 内で登場する位置を計算します。Start は開始位置です。
106	Text	FIXED	FIXED(Text text, Integer decimalPlaces, Boolean omitGroupChar)	※
107	Text	LEFT	LEFT(Text T [; Integer Length])	引数 T の文字列から引数 Length で指定した長さの文字列を左から取得します。
108	Text	LEN	LEN(Text T)	引数の値の長さを計算します。
109	Text	LOWER	LOWER(Text T)	引数の値の大文字をすべて小文字に変換します。 アルファベットだけです。
110	Text	MID	MID(Text T ; Integer Start ; Integer Length)	引数 T の文字列から文字列を抽出します。 引数 Start で抽出個所の先頭を指定します。 引数 Length で抽出する文字列の長さを指定します。
111	Text	PROPER	PROPER(Text T)	※
112	Text	REPLACE	REPLACE(Text T ; Number Start ; Number Len ; Text New)	引数 T の文字列を置き換えます。 置き換えの位置は、引数 Start と引数 Len で指定します。 置き換えたあとの文字列は、引数 New で指定します。
113	Text	REPT	REPT(Text T ; Integer Count)	引数 T の文字列を繰り返します。 繰り返し回数は引数 Count で指定します。
114	Text	RIGHT	RIGHT(Text T [; Integer Length])	引数 T の文字列から引数 Length で指定した長さの文字列を右から取得します。
115	Text	SEARCH	SEARCH(Text findText, Text text, Integer startPosition)	※
116	Text	STRINGCOUNT	STRINGCOUNT(Text text, Text findText)	※
117	Text	SUBSTITUTE	SUBSTITUTE(Text T ; Text Old ; Text	引数 T の文字列を置き換えます。 引数 Old で置き換え対象の文字列を指定します。

項番	カテゴリ	数式	構文	説明
			New [; Number Which])	引数 New で置き換えたあとの文字列を指定します。 引数 Which は、何回目に登場した置き換え対象の文字列を置き換えるかを指定します。
118	Text	T	T(Any X)	引数の値から文字列を作成します。
119	Text	TEXT	TEXT(Scalar X ; Text FormatCode)	引数の値から文字列を作成します。引数 FormatCode でフォーマットを指定できます。
120	Text	TRIM	TRIM(Text T)	引数の値から先頭と末尾のスペースを削除し、内部の複数のスペースをすべて 1 つのスペースに置き換えます。
121	Text	UNICHAR	UNICHAR(Integer)	※
122	Text	UNICODE	UNICODE(Text)	※
123	Text	UPPER	UPPER(Text T)	引数の値の小文字をすべて大文字に変換します。 アルファベットだけです。
124	Text	URLENCODE	URLENCODE(Text text, Text encoding)	※

(凡例) - : 構文はありません。

注※ 説明の詳細は、今後の改訂時に追記される予定です。

2.7.7 [スクリプト]正規表現 (Regex evaluation)

入力フィールドの文字列値を、正規表現で定義されたテキストパターンと照合します。

(1) [設定]タブ

正規化表現に関する設定をします。

図 2-128 [正規表現評価]画面-[設定]タブ

正規表現評価

ステップ名 Regex evaluation

設定

内容

ステップ設定

評価フィールド

結果フィールド名 result

キャプチャグループにフィールドを作成

フィールドを置き換える

正規表現:

正規表現テスト

変数置換を使用する

フィールド (グループ)

#	新規フィールド	タイプ	長さ	精度	書式	グループ	小数	通貨記号	NULL可能	デフォルト	空白除去
1											

ヘルプ

OK

キャンセル

表 2-135 [正規表現評価]画面-[設定]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[評価フィールド]	正規表現で評価したいフィールドを指定します。
3	[結果フィールド名]	結果フィールドの名前を指定します。 このフィールドは出力ストリームに追加され、[評価フィールド]の値が正規表現と一致したかどうかの結果を格納します。一致した場合は「Y」、一致しなかった場合は「N」が出力されます。
4	[キャプチャグループにフィールドを作成]	この項目にチェックを付けると、正規表現によって[評価フィールド]の値が複数に分割される場合（キャプチャグループ）に、その分割フィールド数に基づいて新しい出力フィールドを作成します。 正規表現のキャプチャグループに対応する[評価フィールド]の値の部分文字列が抽出され、新しい出力フィールドに格納されます。 チェックを付ける場合、[フィールド(グループ)]に、必要な数の[新規フィールド]を定義する必要があります。
5	[フィールドを置き換える]	この項目にチェックを付けると、[フィールド(グループ)]で定義した[新規フィールド]が[評価フィールド]に含まれている場合、フィールドの値を置き換えます。 [キャプチャグループにフィールドを作成]にチェックを付けると指定できます。
6	[正規表現]	正規表現を指定します。 正規表現構文のリファレンスドキュメントについては、「java.util.regex.Pattern javadoc (https://docs.oracle.com/javase/jp/8/docs/api/java/util/regex/Pattern.html)」を参照してください。

項番	項目名	説明
7	[正規表現テスト]ボタン	—
8	[変数置換を使用する]	この項目にチェックを付けると、変数参照が正規表現パターンを評価する前に値に展開されます。 正規表現に変数参照が含まれている場合は、チェックを付けてください。
9	[フィールド(グループ)]	入力文字列から正規表現で切り出された部分文字列を格納する新規フィールドを指定します。 [キャプチャグループにフィールドを作成]にチェックを付けている場合、正規表現の各キャプチャグループに対応するフィールドを定義する必要があります。 フィールドの順序は、正規表現のキャプチャグループの順序と同じです。 入力文字列: 「2018/01」、[正規表現]: 「(¥d+)/(¥d+)」 の場合の出力結果の例を次に示します。 出力結果 新フィールド 1: 2018 新フィールド 2: 01 【注意事項】 キャプチャグループは、左右の括弧が一致するペア間の式の一部です。 フィールド定義の数が正規表現のキャプチャグループの数と一致しない場合は、エラーになります。
10	[新規フィールド]	結果を出力するフィールドの名前を指定します。
11	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
12	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
13	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
14	[書式]	出力書式を指定します。
15	[グループ]*	「, (コンマ)」など、桁を区切る文字を指定します。
16	[小数]	「. (ドット)」など、小数点を表す文字を指定します。
17	[通貨記号]	「\$ (ドル)」など、通貨を表す記号を指定します。
18	[NULL 可能]	読み込んだデータがこの項目で指定した値の場合、NULL 値に変換します。 大文字と小文字は区別しません。
19	[デフォルト]	NULL の場合に入力される値を指定します。

項番	項目名	説明
20	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 • [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 • [左] 文字列の前にある全半角空白、タブ文字、改行を削除します。 • [右] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 • [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。

(凡例) —：説明はありません。

注※ 画面の項目名と指定する内容が異なります。

(2) [内容]タブ

正規化表現の内容を設定します。

図 2-129 [正規表現評価]画面-[内容]タブ

表 2-136 [正規表現評価]画面-[内容]タブの項目

項番	項目名	説明
1	[Unicode エンコードの相違を無視する]	この項目にチェックを付けると、すべて ASCII コードとして文字列を評価します。
2	[大文字小文字を区別しない]	この項目にチェックを付けると、大文字、小文字を区別して評価しません。

項番	項目名	説明
3	[空白とコメント行を許す]	この項目にチェックを付けると、空白または「#」以降の文字列はコメントとして評価しません。
4	[ピリオドを終端にする]	通常、「.」の式は改行文字以外のすべての文字を評価しますが、この項目にチェックを付けると、改行文字を含む文字を評価します。
5	[マルチラインモードを有効にする]	この項目にチェックを付けると、「^」と「\$」という式はそれぞれ、改行文字または入力シーケンスの終わりの直後または直前にマッチします。 【注意事項】 この項目は使用できません。
6	[大文字に変換する]	この項目にチェックを付けると、[大文字と小文字を区別しない]が全角文字にも対応します。
7	[Unix ラインモードを有効にする]	この項目にチェックを付けると、「^」と「\$」という式はそれぞれ、改行文字として「\n」または入力シーケンスの終わりの直後または直前にマッチします。 【注意事項】 この項目は使用できません。

2.8 参照

- [\[参照\]HTTP クライアント \(HTTP Client\)](#)
- [\[参照\]REST クライアント \(REST Client\)](#)
- [\[参照\]ストリーム参照 \(Stream lookup\)](#)
- [\[参照\]テーブル確認 \(Table Exists \(Step\)\)](#)
- [\[参照\]データベース参照 \(Database lookup\)](#)
- [\[参照\]データベース結合 \(Database join\)](#)
- [\[参照\]ファイル確認 \(File exists \(Step\)\)](#)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.8.1 [参照]HTTP クライアント (HTTP Client)

指定した URL に HTTP リクエストを送信し、結果を取得します。

(1) [全般]タブ

HTTP クライアントに関する設定をします。

図 2-130 [HTTP クライアント]画面-[全般]タブ

表 2-137 [HTTP クライアント]画面-[全般]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[URL]	URL を指定します。
3	[フィールドからの URL を許可する]	この項目にチェックを付けると、入力ストリームのフィールドの値を使用して URL を指定できます。
4	[URL フィールド名]	値に URL が設定されているフィールドの名前を指定します。 [フィールドからの URL を許可する]にチェックを付けると指定できます。
5	[文字コード]	データの取得に使用する文字コードを指定します。 MS932 を指定する場合は、[windows-31j]を指定してください。
6	[コネクションタイムアウト]	この機能は動作しません。
7	[ソケットタイムアウト]	サーバからデータを取得できない場合に待機するタイムアウト時間を、ミリ秒単位で指定します。
8	[コネクションをクローズするまでの待ち時間]	サーバから切断できない場合に待機するタイムアウト時間を、ミリ秒単位で指定します。

項番	項目名	説明
9	[結果フィールド名]	サーバから取得した結果を格納するフィールドの名前を指定します。
10	[HTTP ステータスコード フィールド名]	HTTP のステータスコード（例：200、404）を格納するフィールドの名前を指定します。
11	[応答時間（ミリ秒）フィールド名]	HTTP の応答時間（単位：ミリ秒）を格納するフィールドの名前を指定します。
12	[Response header field name]	サーバから取得した HTTP のレスポンスヘッダを格納するフィールドの名前を指定します。
13	[Http ログイン]	HTTP 認証（Basic 認証）に使用するユーザー名を指定します。
14	[Http パスワード]	HTTP 認証（Basic 認証）に使用するユーザーパスワードを指定します。
15	[プロキシホスト]	使用するプロキシサーバのホスト名を指定します。
16	[プロキシポート]	使用するプロキシサーバのポート番号を指定します

(2) [フィールド]タブ

URL にクエリーパラメータを追加したり、HTTP ヘッダーにヘッダーを追加したりします。[フィールド取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

HTTPクライアント

ステップ名 HTTPクライアント

全般 フィールド

パラメーター:

#	フィールド名	パラメータ
1		

フィールド取得

カスタムHTTPヘッダー:

#	フィールド	ヘッダー
1		

フィールド取得

ヘルプ OK(O) キャンセル(C)

項番	項目名	説明
1	[フィールド名]	URL にクエリーパラメータを付与する場合、パラメータの値が格納されているフィールドの名前を指定します。 例えば、次のリクエスト URL の場合、「Value1」が格納されているフィールドの名前を指定します。 http://< URL >?key1=Value1
2	[パラメータ]	URL にクエリーパラメータを付与する場合、パラメータ名を指定します。 例えば、次のリクエスト URL の場合、「key1」を指定します。 http://< URL >?key1=Value1
3	[フィールド]	HTTP ヘッダーにヘッダーを付与する場合、追加するヘッダーの値が格納されているフィールドの名前を指定します。
4	[ヘッダー]	HTTP ヘッダーにヘッダーを付与する場合、追加するヘッダーの名前を指定します。

2.8.2 [参照]REST クライアント (REST Client)

指定した URL に HTTP リクエストを送信し、結果を取得します。おもに、REST API を実行するために使用します。

(1) [全般]タブ

REST クライアントに関する設定をします。

図 2-132 [Rest クライアント]画面-[全般]タブ

表 2-139 [Rest クライアント]画面-[全般]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[URL]	HTTP リクエストの送信先の URL を指定します。
3	[フィールドからの URL を許可する]	この項目にチェックを付けると、入力ストリームのフィールドの値を使用して URL を指定できます。
4	[URL フィールド名]	値に URL が設定されているフィールドの名前を指定します。
5	[HTTP メソッド]	HTTP リクエストの送信に使用する HTTP メソッドを、次の中から指定します。 • GET • PUT • DELETE

項番	項目名	説明
		• POST • HEAD • OPTIONS • PATCH
6	[フィールドからメソッド取得]	この項目にチェックを付けると、入力ストリームのフィールドの値を使用して HTTP メソッドを指定できます。
7	[メソッドフィールド名]	HTTP メソッドを指定したフィールド名を指定します。
8	[Body フィールド]	リクエストボディを指定したフィールド名を指定します。 [HTTP メソッド]に POST、PUT、または PATCH を指定した場合に、リクエストボディを含むフィールドの名前を指定します。
9	[アプリケーションタイプ]	送信する HTTP リクエストの形式として使用するアプリケーションタイプを次の中から指定します。 • TEXT PLAIN • XML • JSON • OCTET STREAM • XHTML • FORM URLENCODED • ATOM XML • SVG XML • TEXT XML
10	[結果フィールド名]	サーバから取得した結果を格納するフィールドの名前を指定します。
11	[HTTP ステータスコード ファイル名]	HTTP のステータスコード（例：200、404）を格納するフィールドの名前を指定します。
12	[レスポンスタイム(ミリ秒) フィールド名]	HTTP の応答時間（単位：ミリ秒）を格納するフィールドの名前を指定します。
13	[Response header field name]	サーバから取得した HTTP のレスポンスヘッダを格納するフィールドの名前を指定します。

(2) [認証]タブ

認証に関する設定をします。

図 2-133 [Rest クライアント]画面-[認証]タブ

Restクライアント

ステップ名

REST client

全般

認証

SSL

ヘッダー

パラメーター

Matrix Parameters

HTTP認証

Httpログイン

Httpパスワード

事前認証

☐

使用するプロキシ

プロキシホスト

プロキシポート

ヘルプ

OK

キャンセル

表 2-140 [Rest クライアント]画面-[認証]タブの項目

項番	項目名	説明
1	[Http ログイン]	ユーザー名を指定します。
2	[Http パスワード]	パスワードを指定します。
3	[事前認証]	—
4	[プロキシホスト]	使用するプロキシサーバのホスト名を指定します。
5	[プロキシポート]	使用するプロキシサーバのポート番号を指定します。

(凡例) —：説明はありません。

(3) [SSL]タブ

SSL 認証に関する設定をします。

図 2-134 [Rest クライアント]画面-[SSL]タブ

Restクライアント

ステップ名

REST client

全般

認証

SSL

ヘッダー

パラメーター

Matrix Parameters

トラストストアファイル

トラストストアファイル

参照(B)...

トラストストアパスワード

Key Store file

Key store file

参照(B)...

Key store password

ヘルプ

OK(O)

キャンセル(C)

表 2-141 [Rest クライアント]画面-[SSL]タブの項目

項番	項目名	説明
1	[トラストストアファイル]	SSL 認証が必要な場合、トラストストアファイルのパスを指定します。
2	[トラストストアパスワード]	SSL 認証が必要な場合、トラストストアのパスワードを指定します。
3	[Key store file]	トラストストアを使用するクライアントを認証するための、キーストアファイルのファイルパスを指定します。
4	[Key store password]	キーストアのパスワードを指定します。

(4) [ヘッダー]タブ

HTTP ヘッダーにヘッダーを追加します。

The screenshot shows the "Restクライアント" (REST client) window. At the top, there's a title bar with standard Windows controls. Below it, a tabbed interface has five tabs: "全般" (General), "認証" (Authentication), "SSL", "ヘッダー" (Header), and "パラメーター Matrix Parameters". The "ヘッダー" tab is currently selected.

Under the "ヘッダー" tab, there's a section titled "ヘッダー" (Header). It contains a table with three columns: "#", "フィールド" (Field), and "名前" (Name). The first row is numbered "1". To the right of the table, there's a button labeled "フィールド取得" (Get Field).

At the bottom of the window, there are three buttons: "? ヘルプ" (Help), "OK(O)", and "キャンセル(C)".

項番	項目名	説明
1	[フィールド]	追加するヘッダーの値が格納されているフィールドの名前を指定します。
2	[名前]	追加するヘッダーの名前を指定します。

URL にクエリーパラメータを追加します。[全般]タブの[HTTP メソッド]で POST、PUT、DELETE、または PATCH を指定した場合に指定できます。

図 2-136 [Rest クライアント]画面-[パラメーター]タブ

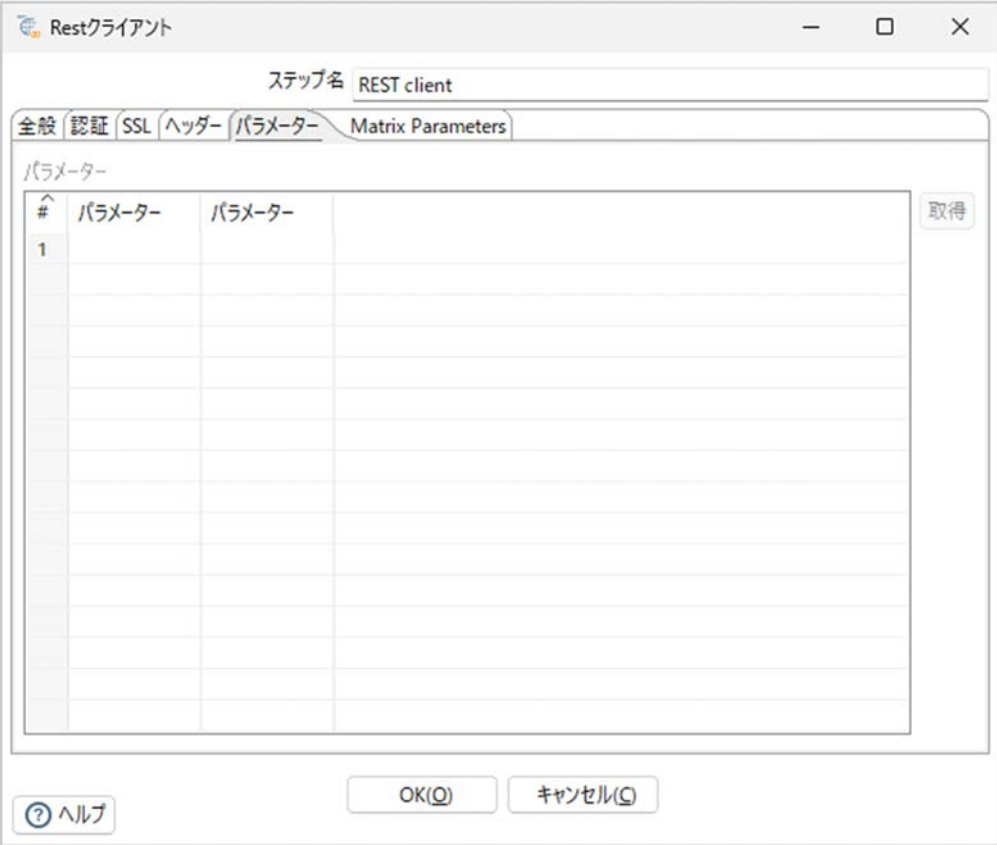


表 2-143 [Rest クライアント]画面-[パラメーター]タブの項目

項番	項目名	説明
1	[パラメーター] (左側)	パラメータの値が格納されているフィールドの名前を指定します。 例えば、次のリクエスト URL の場合、「Value1」が格納されているフィールドの名前を指定します。 http://< URL >?key1=Value1
2	[パラメーター] (右側)	パラメータ名を指定します。 例えば、次のリクエスト URL の場合、「key1」を指定します。 http://< URL >?key1=Value1

(6) [Matrix Parameters]タブ

URL にマトリックスパラメータを追加します。[全般]タブの[HTTP メソッド]に POST、PUT、DELETE、または PATCH を指定した場合に指定できます。

図 2-137 [Rest クライアント]画面-[Matrix Parameters]タブ

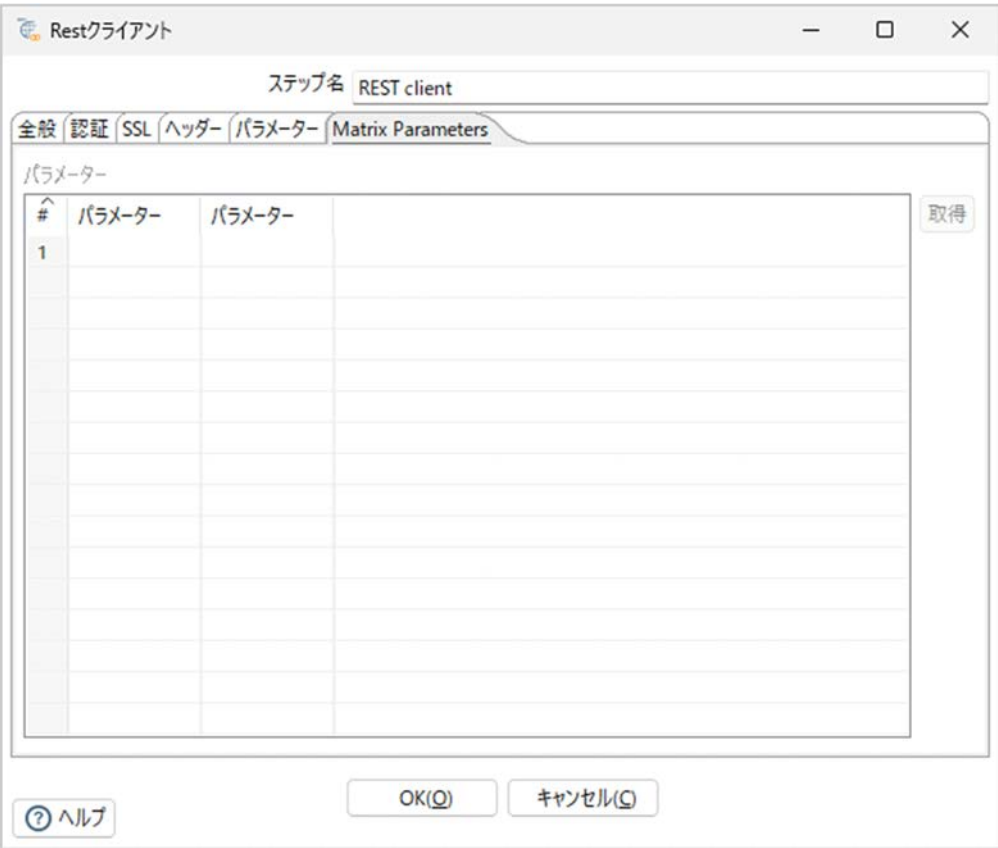


表 2-144 [Rest クライアント]画面-[Matrix Parameters]タブの項目

項番	項目名	説明
1	[パラメーター] (左側)	パラメータの値が格納されているフィールドの名前を指定します。 例えば、次のリクエスト URL の場合、「Value1」が格納されているフィールドの名前を指定します。 http://< URL >/sample;key1=Value1/data
2	[パラメーター] (右側)	パラメータ名を指定します。 例えば、次のリクエスト URL の場合、「key1」を指定します。 http://< URL >/sample;key1=Value1/data

2.8.3 [参照]ストリーム参照 (Stream lookup)

入力ストリームのデータに対して、参照ストリームに入力されたデータと突き合わせ処理を実行し、参照ストリームのデータを取得（検索）します。参照ストリームのデータはいったんすべてメモリに格納され、突き合わせ処理が行われます。

図 2-138 [ストリームルックアップ]画面

ステップ名

ステップ名

フィールド (参照)

#	フィールド名	フィールド名 (ルックアップ)
1		

フィールド (検索)

#	フィールド名 (戻り値)	フィールド名 (新規)	デフォルト	データタイプ
1				

メモリにキャッシュする ☒

メモリへキャッシュされたデータを並替える ☐

ハッシュテーブルを使用する ☐

ヘルプ

OK

キャンセル

フィールドの取得

ルックアップフィールドの取得

表 2-145 [ストリームルックアップ]画面の項目

項番	項目名	説明
1	[ステップ名] (上)	ステップ名を指定します。
2	[ステップ名] (下)	参照ストリームとなるステップを指定します。
3	[フィールド (参照)]	参照キーの設定をします。 [フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
4	[フィールド名]	参照ストリームの参照キーとなるフィールドです。
5	[フィールド名 (ルックアップ)]	入力ストリームの参照キーとなるフィールドです。
6	[フィールド (検索)]	参照データの設定をします。 [ルックアップフィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
7	[フィールド名 (戻り値)]	データを参照するフィールドです。
8	[フィールド名 (新規)]	フィールド名を変更したい場合に、変更後のフィールド名を指定します。 空白の場合、データを参照したフィールド名をそのまま出力します。
9	[デフォルト]	参照データが見つからない場合にデフォルトで出力する値です。 空白の場合 NULL を出力します。
10	[データタイプ]	データタイプをプルダウンから選択します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
11	[メモリにキャッシュする]※	この項目にチェックを付けると、CPU 負荷が大きくなる代わりに、メモリ使用量を抑えられます。

項番	項目名	説明
12	[メモリヘキャッシュされたデータを並替える]	参照フィールドと検索フィールドが Integer の場合にこの項目を選択すると、メモリ使用量を抑えられます。Integer 以外の場合はエラーが発生します。
13	[ハッシュテーブルを使用する]	参照ストリームのレコード長が長い場合にこの項目を選択すると、メモリ使用量を抑えられます。

項番	項目名	説明
5	[フィールド名 (返却値)]	テーブルが存在するかどうかを示す戻り値 (Y または N) を出力するフィールド名を指定します。 Boolean 型のフィールドとして追加されます。テーブルが存在する場合は「Y」が、存在しない場合は「N」が出力されます。

2.8.5 [参照]データベース参照 (Database lookup)

入力ストリームのデータに対して、参照ストリームに入力されたデータベースのデータと突き合わせ処理を実行し、参照ストリームのデータを取得 (検索) します。オプションの設定によって、キャッシュメモリを使用するかどうかを設定できます。キャッシュメモリを使用する場合、次の順序で処理を実行します。

1. キャッシュメモリを検索する。
2. 見つからなければデータベースを検索する。

図 2-140 [データベース参照]画面

表 2-147 [データベース参照]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

項番	項目名	説明
2	[データソース名]	入力テーブルのデータソースを指定します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。[参照]ボタンを押すと、データベース内のスキーマ名を参照できます。
4	[テーブル名]	テーブル名を指定します。[参照]ボタンを押すと、データベース内のテーブル名を参照できます。
5	[キャッシュを使用]	この項目にチェックを付けると、一度参照されたデータをキャッシュメモリに格納します。
6	[キャッシュサイズ]	キャッシュメモリに格納する最大のレコード数を指定します。 0 を指定した場合、レコード数の上限はありません。 数値以外を入力した場合、[OK]ボタンを押すと自動的にデフォルトの値に戻ります。
7	[データをすべて読み込む]	この項目にチェックを付けると、すべての参照元のデータをメモリに格納してから、突き合わせをします。
8	[フィールド (参照)]	[フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。 【注意事項】 XML 制御文字を指定して実行するとエラーが発生します。
9	[列名]	データベースのデータの参照キーとなるフィールドです。
10	[比較演算子]	参照条件となる比較演算子を指定します。
11	[フィールド名 1]	比較演算の第一引数です。
12	[フィールド名 2]	比較演算の第二引数です。
13	[フィールド (戻り値)]	[参照フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
14	[列名]	データを参照するフィールドです。
15	[新しいフィールド名]	フィールド名を変更したい場合に、変更後のフィールド名を指定します。空白の場合、データを参照したフィールド名をそのまま出力します。
16	[デフォルト]	参照データが見つからない場合にデフォルトで出力する値です。 空白の場合 NULL を出力します。
17	[データタイプ]	データタイプをプルダウンから選択します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。

項番	項目名	説明
18	[参照が失敗した場合は中止する]	この項目にチェックを付けると、参照条件に一致するデータがない場合にそのレコードを出力しません。 【注意事項】 レコードが出力されない場合でもエラーは発生しません。
19	[参照が重複した場合は中止する]	【注意事項】 この項目にチェックを付けると、参照条件に一致するデータが複数ある場合にエラーが発生し、データ変換が終了します。
20	[並び替え]	参照条件に一致するデータが複数ある場合に、指定したフィールドを並び替え、先頭のレコードを参照します。

2.8.6 [参照]データベース結合 (Database join)

SQL 文中のプレースホルダーを、先行ステップで指定したパラメータの値に置き換えて、SQL 文を実行し、データを取得します。

[フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-141 [データベース結合]画面

表 2-148 [データベース結合]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	データソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[SQL ステートメント]	データベース結合を形成する SQL クエリーを指定します。プレースホルダーには「? (クエスチョンマーク)」を使用します。 指定したパラメータを使用する場合は「\${パラメータ}」と記述します。
4	[最大レコード数]	読み込むデータのレコード数の最大を指定します。SQL 文の LIMIT 句と同様の処理です。 0 を指定すると、すべてのレコードを読み込みます。
5	[外部結合する]	この項目にチェックを付けると、クエリーが結果を返さない場合に NULL 値のレコードを返します。
6	[パラメータを使用する]	この項目にチェックを付けると、[SQL ステートメント]に指定したパラメータを使用してデータ変換を実行します。
7	[フィールド名]	[SQL ステートメント]に指定したクエリー内の「?」に渡す値を含むフィールド名を指定します。複数のパラメータを指定した場合、[SQL ステートメント]に定義された順でパラメータの値が#1 から割り当てられます。
8	[データタイプ]	フィールドのデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「データタイプの対応表」を参照してください。

2.8.7 [参照]ファイル確認 (File exists (Step))

前のステップで指定されたファイルパスまたはフォルダパスが存在するかどうかを確認します。

図 2-142 [ファイル確認]画面

表 2-149 [ファイル確認]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	存在を確認する対象のファイルパスまたはフォルダパスが含まれるフィールド名を指定します。
3	[フィールド名 (戻り値)]	ファイルパスまたはフォルダパスが存在するかどうかを示す戻り値 (Y または N) を出力するフィールド名を指定します。 Boolean 型のフィールドとして追加されます。ファイルパスまたはフォルダパスが存在する場合は「Y」が、存在しない場合は「N」が出力されます。
4	[ファイル名を結果に含む]	この項目にチェックを付けると、直前のステップのファイル名をメモリに格納します。ファイル取得ステップを使用すると、メモリに格納したファイル名を取得できます。チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [ファイル名を結果に含む]にチェックを付けたステップ名 • comment • timestamp
5	[ファイルタイプを結果に含む]	この項目にチェックを付けると、直前のステップで読み込んだパスのタイプを出力します。
6	[フィールド名]	パスのタイプを出力するフィールドの名前を指定します。出力するパスのタイプは、次のとおりです。

項番	項目名	説明
		• file ファイルパス • folder フォルダパス 【注意事項】 指定したファイルまたはフォルダが存在しない場合は、「imaginary」という文字列ではなく、空白が出力されます。

2.9 結合

- [結合]XML 結合 (XML Join)
- [結合]マージ結合 (Merge join)
- [結合]多方向 Merge Join (Multiway merge join)
- [結合]整列併合 (Sorted merge)
- [結合]行マージ (比較) (Merge rows (diff))
- [結合]行結合 (デカルト積) (Join rows (cartesian product))

上記以外のステップについては、Pentaho Documentation を参照してください。

2.9.1 [結合]XML 結合 (XML Join)

ストリームソース要素を、対象ストリーム要素に追加します。

対象ストリーム要素は、XML 文書として表現された、ただ 1 つのレコードである必要があります。複数レコードの場合でも、先頭レコードだけが、対象ストリーム要素として認識され、その他のレコードは無視されます。

ストリームソース要素は、複数レコードで構成することもできます。これらの行に含まれるタグがすべて対象ストリーム要素に追加されます。

結合 (追加) されると、結果 XML フィールドにただ 1 つのレコードが XML 文書として出力されます。また、対象ストリーム要素のフィールドも出力されます。

なお、対象ストリーム要素、およびストリームソース要素に、文書型宣言は使用できません。

図 2-143 [XML 結合]画面

表 2-150 [XML 結合]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[対象 XML ステップ]	対象ストリーム要素の入力元となるステップ名を指定します。
3	[対象 XML フィールド]	対象ストリーム要素を格納するフィールド名を指定します。
4	[XML ステップソース]	ストリームソース要素の入力元となるステップ名を指定します。
5	[XML フィールドソース]	ストリームソース要素を格納するフィールド名を指定します。
6	[XPath 構文]	ストリームソース要素の対象ストリーム要素への追加ノード位置を示した XPath 構文を指定します。 [複雑な結合]にチェックを付けると、「? (クエスチョンマーク)」がプレースホルダーとして使用されます。プレースホルダーは、[結合比較フィールド]に指定した値に置換されます。 プレースホルダーは複数使用できますが、すべてのプレースホルダーは[結合比較フィールド]に指定した値に置換されるため、同一の値に置換されます。 XPath 構文に誤りがあると、エラーが発生します。

項番	項目名	説明
7	[複雑な結合]	この項目にチェックを付けると、[XPath 構文]で指定したプレースホルダーを[結合比較フィールド]の値で置換した XPath 構文で結合します。
8	[結合比較フィールド]	[XPath 構文]で指定したプレースホルダーと置換される値を格納するフィールドを指定します。 この項目は[複雑な結合]にチェックを付けた場合に有効です。
9	[結果 XML フィールド]	結果ストリーム要素を格納するフィールド名を指定します。
10	[文字コード]	XML 宣言で宣言される文字コードを指定します。結果ストリーム要素に変換する際のエンコーディングでもあります。
11	[XML ヘッダーを除く]	この項目にチェックを付けると、結果ストリーム要素から XML 宣言が除去されます。
12	[XML 結果から NULL 値を除く]	この項目にチェックを付けると、空要素は結果ストリーム要素から完全に除去されます。そのため、結果ストリーム要素ファイルの容量が節約できます。 この項目にチェックを付けない場合、空要素は NULL 値（例：<abc/>）として結果ストリーム要素に追加されます。

2.9.2 [結合]マージ結合 (Merge join)

入力ストリームの 2 つのテーブルを結合します。キーとなるフィールドはソートされている必要があります。

図 2-144 [マージ結合]画面

表 2-151 [マージ結合]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[結合 1 のステップ名]	結合後、左側 (LEFT) に配置されるテーブルを入力するステップを指定します。

項番	項目名	説明
3	[結合 2 のステップ名]	結合後、右側（RIGHT）に配置されるテーブルを入力するステップを指定します。
4	[結合タイプ]	結合タイプを次の中から指定します。 • [INNER] 内部結合します。 • [LEFT_OUTER] 左外部結合します。 • [RIGHT_OUTER] 右外部結合します。 • [FULL_OUTER] 完全外部結合します。
5	[フィールド(結合 1)]の [フィールド名]	キーとなるフィールドの名前を指定します。 最上の行（1 行目）から順にキーが参照されます。 [フィールドを取得]ボタンを押すと、[結合 1 のステップ名]で指定したステップからの入力ストリームのフィールドを自動入力できます。
6	[フィールド(結合 2)]の [フィールド名]	キーとなるフィールドの名前を指定します。 最上の行（1 行目）から順にキーが参照されます。 [フィールドを取得]ボタンを押すと、[結合 2 のステップ名]で指定したステップからの入力ストリームのフィールドを自動入力できます。

2.9.3 [結合]多方向 Merge Join（Multiway merge join）

複数の入力ストリームテーブルを結合します。

キーとなるフィールドがソートされていないと正しく結合できないのでご注意ください。

図 2-145 [多重マージジョイン]画面

表 2-152 [多重マージジョイン]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[Input Step 1]	この項目は、テーブルを入力するステップがある場合に表示されます。 入力ステップが増えると、入力ステップの数に応じて、ステップ番号が増えます。 例：「Input Step 1」、「Input Step 2」、「Input Step 3」

項番	項目名	説明
3	[Join Keys]	この項目は、テーブルを入力するステップがある場合に表示されます。[Select Keys] ボタンを押して、[Join Keys]ウィンドウを表示し、キーとなるフィールドの名前を、入力ステップごとに指定します。 [Input Step 1]から順にキー参照が行われます。
4	[Join タイプ]	結合タイプを次の項目から指定します。 • [INNER] 内部結合 • [FULL_OUTER] 外部結合

2.9.4 [結合]整列併合 (Sorted merge)

複数の入力ストリームのデータに対し、指定したフィールドの値でソートして出力します。各ストリームのデータはフィールドの定義が同じで、かつ指定したフィールドの値で事前にソートされている必要があります。

[フィールド名]に「Key」を指定した場合のソート結果の例を、次に示します。

入力データ

データ1		データ2	
Key	Data	Key	Data
1	aaa	3	xxx
2	bbb	4	yyy
5	ccc	7	zzz
6	ddd		

ソート結果

Key	Data
1	aaa
2	bbb
3	xxx
4	yyy
5	ccc
6	ddd
7	zzz

図 2-146 [整列併合]画面

整列併合

ステップ名 整列併合

フィールド

#	フィールド名	昇順で並替え
1		

ヘルプ

OK(Q)

キャンセル(Q)

フィールドを取得(G)

表 2-153 [整列併合]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	ソートの基準となるフィールド名を指定します。
3	[昇順で並替え]	- 昇順でソートしたい場合 「Y」を指定します。昇順でソート済みの入力データが必要です。 - 降順でソートしたい場合 「N」を指定します。降順でソート済みの入力データが必要です。

2.9.5 [結合]行マージ（比較）（Merge rows (diff)）

フィールドの値を変更する前と変更したあとのデータを入力して比較し、結果を新たなフィールドに出力します。2つの入力データのフィールド定義（名前、列番号）が一致しないとエラーが発生します。

[フィールド(比較 1)]に「Key」を、[フィールド(比較 2)]に「Data」を指定した場合の比較結果の例を、次に示します。

入力データ

変更前[比較1のステップ名]		変更後[比較2のステップ名]	
Key	Data	Key	Data
1	8	1	8
2	10	2	9
3	11	3	14
4	15	4	15
5	18	6	20

比較結果

Key	Data	Result
1	8	identical
2	9	changed
3	14	changed
4	15	identical

5	18	deleted
6	20	new

図 2-147 [行マージ (比較)]画面

表 2-154 [行マージ (比較)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[比較 1 のステップ名]	変更前のデータを入力するステップ名を指定します。
3	[比較 2 のステップ名]	変更後のデータを入力するステップ名を指定します。
4	[フラグとなるフィールド名]	このフィールドには、比較結果として次の値が出力されます。 - identical 値が変更されなかったレコード - changed 値が変更されたレコード - new 新しく追加されたレコード - deleted 削除されたレコード
5	[フィールド(比較 1)]の [フィールド名]	比較時にキーとするフィールドの名前を指定します。
6	[フィールド(比較 2)]の [フィールド名]	比較対象のフィールドの名前を指定します。

2.9.6 [結合]行結合（デカルト積）（Join rows (cartesian product)）

複数ステップから入力されるデータをクロス結合し、条件に合うレコードだけを出力します。

[結合条件]に「Key1 = Key2 AND Data1 > Data2」を指定した場合の出力結果の例を、次に示します。

入力データ

データ1		データ2	
Key1	Data1	Key2	Data2
1	8	1	8
2	9	2	4
3	10	3	5
4	11	4	22
5	12	5	12
6	13	6	13
7	14	7	3

出力結果

Key1	Data1	Key2	Data2
2	9	2	4
3	10	3	5
7	14	7	3

図 2-148 [行結合 (デカルト積)]画面

行結合 (デカルト積)

ステップ名

行結合 (デカルト積) |

一時ディレクトリ

%%java.io.tmpdir%%

参照(B)

一時ファイルの接頭子

out

キャッシュサイズ

500

主に参照するステップ名

結合条件

<field> = <field>
<value>

ヘルプ

OK(Q)

キャンセル(C)

表 2-155 [行結合 (デカルト積)]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[一時ディレクトリ]	一時ファイルを出力するディレクトリのパスを指定します。
3	[一時ファイルの接頭子]	一時ファイルの接頭子を指定します。
4	[キャッシュサイズ]	キャッシュメモリに格納する入力データの最大行数を指定します。

項番	項目名	説明
5	[主に参照するステップ名]	ここで指定したステップがメインストリームとなり、一時ファイルには書き込まれずに処理されます。 ここで指定していないステップからの入力、一時ファイルに書き込まれて処理されます。 入力ステップの中で最もサイズが大きいデータを処理するステップを設定することを推奨します。
6	[結合条件]	出力先を決定するための条件式を指定します。 条件式を記述するとき使用できる論理演算子は、NOT、AND、OR、AND NOT、OR NOT、XOR です。 【注意事項】 - AND と OR は右辺の式に否定が付いた AND NOT と OR NOT です。NAND や NOR ではありませんので、ご注意ください。 - エラーが発生するおそれがあるため、条件式の左辺と右辺のデータタイプは一致させてください。 右辺に値を指定する場合は、[結合条件]の「<value>」をクリックして[入力値]ダイアログを表示します。[入力値]ダイアログでは、[タイプ]、[値]、[変換形式]、[長さ]、[精度]を指定できます。
7	[+]（状態追加）	[+]アイコンをクリックすると新しい結合条件をメインレベルに追加し、既存の結合条件をサブレベルに変換します。 サブレベルの条件をクリックすると条件ツリーの 1 つ下のレベルに移動し、編集できるようになります。

2.10 データウェアハウス

- [\[データウェアハウス\]ディメンジョン 参照/更新 \(Dimension lookup/update\)](#)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.10.1 [データウェアハウス]ディメンジョン 参照/更新 (Dimension lookup/update)

ディメンジョンテーブルの参照と更新ができます。

(1) [キー]タブ

参照または更新に必要な情報を指定します。

図 2-149 [ディメンジョン 参照/更新]画面-[キー]タブ

ステップ名 デフォルト

更新する ☒

データソース名 編集(E) 新規作成(N)... ウィザード

スキーマ名 参照(B)...

テーブル名 dim table name 参照(B)...

コミットサイズ 100

キャッシュを有効化する ☒

キャッシュをプリロードする ☐

キャッシュサイズ 5000

キー フィールド

キーフィールド

#	ディメンジョンフィールド名	ストリームフィールド名
1		

キーフィールド名 新しいフィールド名

キーの生成

☒ テーブル最大値から+1を使用

☐ シーケンスを使用

☐ 自動増分

バージョンフィールド version

日付フィールド

開始日付フィールド date_from 最小年 1900

代替開始日を使用する ☐ <選択オプション>

終了日付フィールド date_to 最大年 2199

OK(O) キャンセル(C) フィールドの取得(G) SQL(S)

ヘルプ

表 2-156 [ディメンジョン 参照/更新]画面-[キー]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[更新する]	この項目にチェックを付けると、入力ストリームの情報に基づいてディメンジョンテーブルを更新します。 チェックを付けない場合、ディメンジョンテーブルの参照だけを実行し、[キーフィールド名]で指定した内容をストリームに追加します。
3	[データソース名]	ディメンジョンテーブルが存在するデータベース接続を指定します。
4	[スキーマ名]	データベース内のスキーマ名を指定します。
5	[テーブル名]	データベース内のディメンジョンテーブル名を指定します。

項番	項目名	説明
		ディメンジョンテーブルには次のカラムが必要です。 • サロゲートキー[整数型] [キーフィールド名]に指定するカラムです。 • バージョン[整数型] [バージョンフィールド]に指定するカラムです。 • 開始日付[日付型] [開始日付フィールド]に指定するカラムです。 • 終了日付[日付型] [終了日付フィールド]に指定するカラムです。 • 最終 insert/update 日付[日付型] [更新タイプ]に[最終 insert または update 日付(ストリームフィールドを含まない)]を選択した場合に必要なカラムです。 • 最終 insert 日付[日付型] [更新タイプ]に[最終 insert 日付(ストリームフィールドを含まない)]を選択した場合に必要なカラムです。 • 最終 update 日付[日付型] [更新タイプ]に[最終 update 日付 (without stream field as source)]を選択した場合に必要なカラムです。 • 最新バージョン[日付型] [更新タイプ]に[最新バージョン(ストリームフィールドを含まない)]を選択した場合に必要なカラムです。 【注意事項】 参照先テーブルの列名に SQL 演算子が含まれていると、データ変換実行時にエラーが発生することがあります。
6	[コミットサイズ]	コミットサイズを指定します。10 を指定すると、10 回の挿入または更新ごとにコミットを実行します。 [更新する]にチェックを付けると指定できます。
7	[キャッシュを有効化する]	この項目にチェックを付けると、このステップでデータキャッシュを有効にします。
8	[キャッシュをプリロードする]	この項目にチェックを付けると、参照を実行する前にディメンジョンテーブルの内容全体を読み取ることで、パフォーマンスを向上できます。 [更新する]にチェックを付けない場合で、かつ[キャッシュを有効化する]にチェックを付けると指定できます。 【注意事項】 参照先テーブルに Timestamp 型のデータが含まれていると、データ変換実行時にエラーが発生します。
9	[キャッシュサイズ]	メモリに保持される行数のキャッシュサイズです。 [キャッシュを有効化する]にチェックを付けると指定できます。 0 を指定した場合、JVM のメモリがなくなるまでキャッシュされます。
10	[キーフィールド]	参照または更新に必要な情報の一覧です。
11	[ディメンジョンフィールド名]	参照または更新に必要なディメンジョンテーブルのフィールドを指定します。

項番	項目名	説明
12	[ストリームフィールド名]	参照または更新に必要なストリームデータのフィールドを指定します。
13	[キーフィールド名]	ディメンジョンテーブルの主キー（サロゲートキー）を指定します。 例えば、項番のように、それ自体に意味はなくても、ユニークなデータであり、主キーになるものがサロゲートキーです。
14	[新しいフィールド名]	参照時にストリームに追加する[キーフィールド名]の名前を変更する場合に、新しい名前を指定します。 [更新する]にチェックを付けない場合に指定できます。
15	[キーの生成]	キーの生成方法を、次の項目から指定します。 • テーブル最大値から+1 を使用 • シーケンスを使用 • 自動増分
16	[テーブル最大値から+1 を使用]	テーブルの最大キーから新しいキーが作成されます。
17	[シーケンスを使用]	データベースのシーケンスを使用して、キーを生成する場合は、シーケンス名を指定します。
18	[自動増分]	データベースの自動インクリメントを使用して、キーを生成します。
19	[バージョンフィールド]	バージョン情報を管理するフィールド名を指定します。 [更新する]にチェックを付けると指定できます。
20	[日付フィールド]	• 更新の場合 ストリームに最後に変更された日時があれば、そのフィールドを指定します。指定を省略した場合は、システム時間が設定されます。 • 参照の場合 [開始日付フィールド]以降かつ[終了日付フィールド]未満に該当するレコードが検索されます。この項目が指定されない場合は、現在のシステム日付を使用して検索します。
21	[開始日付フィールド]	開始日時のフィールド名です。 ディメンジョンテーブルに追加された日付が設定されます。
22	[最小年]	[開始日付フィールド]のフィールドに設定する開始日時の年数部分を指定します。例えば、「1900」を指定すると、開始日時として「1900/1/1/ 00:00:00」が設定されます。 [更新する]にチェックを付けると指定できます。
23	[代替開始日を使用する]	この項目にチェックを付けると、「[最小年]/1/1 00:00:00」の日時の代わりに指定できます。次の項目から指定します。 • [System date] [終了日付フィールド]にシステム日時を設定します。 1 件追加する場合、[終了日付フィールド]に「[最大年]/12/31/ 23:59:59」が設定されます。2 件目以降の追加の場合に、[終了日付フィールド]にシステム日時が設定されます。 • [Start date if transformation] [開始日付フィールド]に、データ変換の開始時のシステム日時を設定します。

項番	項目名	説明
		• [Empty(null)値] [開始日付フィールド]に null 値を設定します。 最新バージョンの[終了日付フィールド]には「[最大年]/12/31/ 23:59:59」が設定されます。 • [A column value] [A column value]を選択すると、右側の項目が指定できます。指定したフィールドの値を使用して[開始日付フィールド]を指定します。 ストリームのフィールドの値を設定します。指定するフィールド名は、ディメンジョンテーブルのフィールド名と同じにしないとエラーが発生します。 【注意事項】 [A column value]に日付型の列名が指定されていると、データ変換実行時にエラーが発生します。
24	[終了日付フィールド]	終了日時のフィールド名です。
25	[最大年]	[終了日付フィールド]のフィールドに設定する、終了日時の年数部分を指定します。例えば、「2199」を指定すると、終了日時として「2199/12/31/ 23:59:59.999+9」が設定されます。 [更新する]にチェックを付けると指定できます。

(2) [フィールド]タブ (更新の場合)

更新に必要な情報を指定します。

図 2-150 [ディメンジョン 参照/更新]画面-[フィールド]タブ (更新の場合)

表 2-157 [ディメンジョン 参照/更新]画面-[フィールド]タブの項目 (更新の場合)

項番	項目名	説明
1	[ルックアップ/アップデートフィールド]	更新に必要な情報の一覧です。
2	[ディメンジョンフィールド名]	更新に必要なディメンジョンテーブルのフィールドを指定します。
3	[比較フィールド名]	更新に必要なストリームデータのフィールドを指定します。 [更新する]にチェックを付けると指定できます。
4	[更新タイプ]	フィールドごとの更新タイプを選択します。選択した内容で自動的に管理されます。 [更新する]にチェックを付けると指定できます。

項番	項目名	説明
		• [Insert] 比較フィールドのデータが見つかり、1 つ以上の属性が異なっていた場合、テーブルに追加します。 • [Update] 比較フィールドのデータが見つかり、1 つ以上の属性が異なっていた場合、最新のディメンジョンバージョンのこれらの属性を更新します。 • [Punch through] 比較フィールドのデータが見つかり、1 つ以上の属性が異なっていた場合、すべてのディメンジョンバージョンのこれらの属性を更新します。 • [最終 insert または update 日付(ストリームフィールドを含まない)] 最後に insert または update した日時が設定されます。 • [最終 insert 日付(ストリームフィールドを含まない)] 最後に insert した日時が設定されます。 • [最終 update 日付(without stream field as source)] 最後に update した日時が設定されます。 • [最終バージョン(ストリームフィールドを含まない)] 最新バージョンなら true、最新以外なら false が設定されます。

(3) [フィールド]タブ (参照の場合)

参照に必要な情報を指定します。

図 2-151 [ディメンジョン 参照/更新]画面-[フィールド]タブ (参照の場合)

ディメンジョン 参照/更新

ステップ名

ディメンジョン 参照/更新

更新する

☐

データソース名

編集(E)

新規作成(N)...

ウィザード

スキーマ名

参照(B)...

テーブル名

dim table name

参照(B)...

コミットサイズ

100

キャッシュを有効化する

☒

キャッシュをプリロードする

☐

キャッシュサイズ

5000

キー

フィールド

ルックアップ/アップデートフィールド

#	ディメンジョンフィールド名	出力フィールド名	結果フィールド名
1			

キーフィールド名

新しいフィールド名

キーの生成

☒ テーブル最大値から+1を使用

☐ シーケンスを使用

☐ 自動増分

バージョンフィールド

version

日付フィールド

開始日付フィールド

date_from

最小年

1900

代替開始日を使用する

☐

<選択オプション>

終了日付フィールド

date_to

最大年

2199

OK(O)

キャンセル(C)

フィールドの取得(G)

SQL(S)

ヘルプ

表 2-158 [ディメンジョン 参照/更新]画面-[フィールド]タブの項目 (参照の場合)

項番	項目名	説明
1	[ルックアップ/アップデートフィールド]	参照に必要な情報の一覧です。
2	[ディメンジョンフィールド名]	参照に必要なディメンジョンテーブルのフィールドを指定します。
3	[出力フィールド名]	出力フィールド名を指定します。 [更新する]にチェックを付けない場合に指定できます。
4	[結果フィールド名]	データタイプを選択します。 [更新する]にチェックを付けない場合に指定できます。

2.11 データ検証

- [データ検証]データ検証 (Data Validator)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.11.1 [データ検証]データ検証 (Data Validator)

フィールド内のデータのルール（値の範囲、値の個別のリスト、またはデータ長など）を定義し、データを検証します。

図 2-152 [データ検証]画面

表 2-159 [データ検証]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[検証リスト]	データの検証ルールの一覧です。 検証ルールに従い、入力データの検証を行います。 検証エラーが発生すると、次のフィールドが出力されます。 - エラーレコード - エラー詳細 - エラーフィールド - エラーコード 出力されるフィールドは、エラーが発生したステップで事前に設定しています。該当するステップの、右クリックメニューから[エラー処理定義]を選択して設定できます。 検証ルールを作成する場合は、[新規作成]ボタンを押し「バリデーション名（検証ルール名）」を指定します。 リストから削除する場合は、対象の検証ルールを選択し、[リストから削除]ボタンを押します。
3	[全てのエラーを報告する]	この項目にチェックを付けると、すべてのエラーを報告します。 チェックを付けない場合、複数エラーが発生しても最初に発生した検証ルールのエラーだけを報告します。
4	[単体行に出力、:によって区切られたエラー連鎖]	この項目にチェックを付けると、指定した区切り文字を使って、データごとに次のフィールドの文字列を連結し、1行として出力します。 - エラー詳細 - エラーフィールド - エラーコード
5	[説明]	検証ルール作成時に指定したバリデーション名です。この項目を変更すると[検証リスト]に反映されます。
6	[フィールド名]	検証対象のデータが格納されているフィールド名です。
7	[エラーコード]	エラーコードに出力する独自のエラーコードを指定します。 対象データに複数のエラーが発生している場合でも、ここで指定したエラーコードが出力されます。
8	[説明]	エラー詳細に出力するメッセージを指定します。 対象データに複数のエラーが発生している場合でも、ここで指定した説明が出力されます。
9	[データ型を確認する]	この項目にチェックを付けると、入力データのデータタイプの検証を行います。
10	[データタイプ]	PDI上のデータタイプを指定します。 PDIデータタイプとJavaデータタイプの対応については、「データタイプの対応表」を参照してください。 [データ型を確認する]にチェックを付けた場合、ここで指定したデータタイプと[フィールド名]のデータタイプが異なるときに、エラーコード（KVD004）のエラーを出力します。

項番	項目名	説明
		[データ型を確認する]にチェックを付けない場合、[データ検証]に指定するデータ※のデータタイプを指定します。 注※ データとは、[最大値]、[最小値]、[許容値]、[他のステップから値を参照する]で指定した値です。
11	[変換]	[データ検証]に設定するデータ※の書式を、入力データの書式から変更する場合は指定します。 注※ データとは、[最大値]、[最小値]、[許容値]、[他のステップから値を参照する]で指定した値です。
12	[桁区切り文字]※	[. (ドット)] など、小数点を表す文字を指定します。
13	[数値囲み文字]※	[, (コンマ)] など、桁を区切る文字を指定します。
14	[NULL を許容する]	この項目にチェックを付けると、データに NULL 値が含まれていることを許可します。 チェックを付けない場合、データに NULL 値が含まれているとエラーコード (KVD001) のエラーを出力します。
15	[NULL のみを許容する]	この項目にチェックを付けると、データに NULL 値だけを許可します。 データに NULL 値以外が含まれるとエラーコード (KVD015) のエラーを出力します。
16	[数値データのみ]	この項目にチェックを付けると、数値だけを許可します。 データに数値以外の値が含まれるとエラーコード (KVD008) のエラーを出力します。
17	[最大の長さ]	入力したデータが[最大の長さ]より長いときは、エラーコード (KVD002) のエラーを出力します。
18	[最小の長さ]	入力したデータが[最小の長さ]より短いときは、エラーコード (KVD003) のエラーを出力します。
19	[最大値]	入力したデータが[最大値]より大きいときは、エラーコード (KVD006) のエラーを出力します。
20	[最小値]	入力したデータが[最小値]より小さいときは、エラーコード (KVD005) のエラーを出力します。
21	[前方文字列]	入力したデータが[前方文字列]で始まらないとき、エラーコード (KVD009) のエラーを出力します。
22	[後方文字列]	入力したデータが[後方文字列]で終わらないときは、エラーコード (KVD010) のエラーを出力します。
23	[除外する前方の文字列]	入力したデータが[除外する前方の文字列]で始まるときは、エラーコード (KVD011) のエラーを出力します。
24	[除外する後方の文字列]	入力したデータが[除外する後方の文字列]で終わるときは、エラーコード (KVD012) のエラーを出力します。
25	[検索文字列]	入力したデータが[検索文字列] (正規表現) と一致しないときは、エラーコード (KVD013) のエラーを出力します。
26	[除外する検索文字列]	入力したデータが[除外する検索文字列] (正規表現) と一致するときは、エラーコード (KVD014) のエラーを出力します。

項番	項目名	説明
27	[許容値]	許容値のリストです。 許容値を作成する場合は、[追加]ボタンを押し、値を指定後、[OK]ボタンを押します。 許容値を削除する場合は、対象の許容値を選択し、[削除]ボタンを押します。 データがどの許容値とも一致しない場合は、エラーコード（KVD007）のエラーを出力します。
28	[他のステップから値を参照する]	この項目にチェックを付けると、データが指定したステップのフィールドの値と一致しないときは、エラーコード（KVD007）のエラーを出力します。
29	[ステップ名]	参照するステップ名を指定します。
30	[フィールド名]	参照するフィールド名を指定します。

注※ 画面の項目名と指定する内容が異なります。

2.12 統計

- [統計]グループ化 (Group by)
- [統計]サンプル行 (Sample rows)
- [統計]メモリーグループ化 (Memory group by)
- [統計]分析クエリー (Analytic query)
- [統計]単変量統計 (Univariate statistics)
- [統計]貯留層サンプリング (Reservoir sampling)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.12.1 [統計]グループ化 (Group by)

キーでレコードをグループ化して選択した値を集計し、集計結果を出力します。

キーとなるフィールドはソートされている必要があります。

図 2-153 「グループ化」画面

[illegible]

表 2-160 [グループ化]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[すべての行を含む]	この項目にチェックを付けると、グループ化して選択した値を集計し、集計結果を入力行に追加して出力します。
3	[一時ディレクトリ]	出力が 5,000 行を超える場合、一時ファイルに出力されます。一時ファイルを保存するディレクトリを指定します。 [すべての行を含む]にチェックを付けると指定できます。
4	[一時ファイルの接頭子]	一時ファイルの接頭辞を指定します。 [すべての行を含む]にチェックを付けると指定できます。
5	[レコード番号を含む]	この項目にチェックを付けると、レコード番号を付与できます。 シーケンス追加（変化値）ステップのように、キーフィールドの値が変化したときに、カウンタが 1 に戻ります。 [すべての行を含む]にチェックを付けると指定できます。
6	[フィールド名]	レコード番号を出力するフィールド名を指定します。 [レコード番号を含む]にチェックを付けると指定できます。
7	[戻り値を常に返す]	—
8	[フィールド(グループ)]の [フィールド名]	グループ化のキーとなるフィールドを指定します。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
9	[フィールド(集計)]の [フィールド名]	出力フィールドの名前を指定します。 [参照フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
10	[対象]	集計対象となるフィールドを指定します。
11	[演算子]	指定する項目については、「 [演算子]の指定項目（グループ化） 」を参照してください。
12	[値]	次の演算子を指定した場合に、必要な値を指定します。 - パーセンタイル値 - 文字列連結(セパレーターを値で指定) - Percentile (nearest-rank method)

(凡例) —：説明はありません。

表 2-161 [演算子]の指定項目（グループ化）

項目名	説明
[合計]	合計を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[平均(Mean)]	平均を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。

項目名	説明
	数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[中央値]	中央値を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[パーセンタイル値]	パラメータ[値]に入力された数を「q」としたとき、q パーセンタイルを計算します。 [値]に 0~100 の数を入力でき、それ以外を入力するとエラーが発生します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[最低]	最低の値を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 Internet Address 型以外のデータ型を入力できます。
[最高]	最高の値を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 Internet Address 型以外のデータ型を入力できます。
[値の数(N)]	値の数が入力されている行数（NULL 以外）を計算します。 空文字はカウントされません。
[,(コンマ)で分けられた文字列を結合]	データを文字列に変換して「,」区切りで連結して出力します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。
[最初の値(NULL を除く)]	NULL を除いた最初に現れた値を出力します。
[最後の値(NULL を除く)]	NULL を除いた最後に現れた値を出力します。
[最初の値]	NULL を含む最初に現れた値を出力します。
[最後の値]	NULL を含む最後に現れた値を出力します。
[累積合計(すべての行を含む場合のみ)]	各キーでの累積合計を各行に出力します。 Integer 型、Number 型、BigNumber 型、Boolean 型、および String 型を入力できます。
[累積平均(すべての行を含む場合のみ)]	各キーでの累積平均を各行に出力します。 Integer 型、Number 型、および BigNumber 型を入力できます。
[標準偏差]	標準偏差を出力します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 入力された値がすべて NULL の場合、エラーが発生します。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[文字列連結(セパレーターを値で指定)]	データを文字列に変換して、値で指定した文字で区切り、連結して出力します。
[異なった値の数(N)]	ユニークな値が入力されている行数（NULL を除く）を出力します。 Internet Address 型以外のデータ型を入力できます。
[行数(フィールド引数なし)]	指定したキーと一致する行数を出力します。

項目名	説明
[Standard deviation (sample)]	標本標準偏差を出力します（平均からの偏差平方和を n-1 で割った値の平方根。標準偏差は平均からの偏差平方和を n で割った値の平方根）。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 入力された値がすべて NULL の場合、エラーが発生します。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[Percentile (nearest-rank method)]	パラメータ[値]に入力された数を「q」としたとき、q パーセンタイルを計算します。パーセンタイルの計算には Nearest-rank 手法を使用します。 [値]に 0～100 の数を入力でき、それ以外を入力するとエラーが発生します。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。

2.12.2 [統計]サンプル行 (Sample rows)

入力データから指定した行だけを出力します。

図 2-154 [サンプル行]画面

表 2-162 [サンプル行]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[行範囲]	入力データから出力する行の番号を指定します。 1 行目、4 行目、および 15 行目を出力したい場合 「1,4,15」と指定します。 5 行目～10 行目、16 行目～20 行目を出力した場合 「5..10,16..20」と指定します。
3	[行番号フィールド名]	入力データの行番号を出力するフィールド名です。

2.12.3 [統計]メモリーグループ化 (Memory group by)

キーでレコードをグループ化して選択した値を集計し、集計結果を出力します。

キーフィールドがソートされていなくても処理されますが、集計結果の種類が多い場合、メモリ不足のエラーが発生しやすいため、ご注意ください。

処理できるデータ量は JVM に割り当てるメモリに依存します。

図 2-155 [メモリーグループ化]画面

メモリーグループ化

ステップ名

メモリーグループ化

戻り値を常に返す

フィールド (グループ)

#

フィールド名

1

フィールドを取得(G)

フィールド (集計)

#

フィールド名

対象

演算子

値

1

フィールドを取得(L)

ヘルプ

OK(Q)

キャンセル(Q)

表 2-163 [メモリーグループ化]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[戻り値を常に返す]	－
3	[フィールド(グループ)]の [フィールド名]	グループ化のキーとなるフィールドを指定します。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
4	[フィールド(集計)]の [フィールド名]	出力フィールドの名前を指定します。 [フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
5	[対象]	集計対象となるフィールドを指定します。
6	[演算子]	指定する項目については、「 [演算子]の指定項目 (メモリーグループ化) 」を参照してください。
7	[値]	次の演算子を指定した場合に、必要な値を指定します。 - Percentile - 文字列結合

(凡例) －：説明はありません。

表 2-164 [演算子]の指定項目 (メモリーグループ化)

項目名	説明
[合計]	合計を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[平均(Mean)]	平均を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[Median]	中央値を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[Percentile]	パラメータ[値]に入力された数を「q」としたとき、q パーセンタイルを計算します。 [値]に 0～100 の数を入力でき、それ以外を入力するとエラーが発生します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。
[最低]	最低の値を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 Internet Address 型以外のデータ型を入力できます。
[最高]	最高の値を計算します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。 Internet Address 型以外のデータ型を入力できます。
[値の数(N)]	値の数が入力されている行数 (NULL 以外) を計算します。 空文字はカウントされません。
[,(コンマ)で分けられた文字列を結合]	データを文字列に変換して「,」区切りで連結して出力します。 NULL 値は対象外となり、NULL 値のレコードを除いた結果が返ります。
[最初の NULL 以外の値]	NULL を除いた最初に現れた値を出力します。
[最後の NULL 以外の値]	NULL を除いた最後に現れた値を出力します。
[最初の値]	NULL を含む最初に現れた値を出力します。
[最後の値]	NULL を含む最後に現れた値を出力します。
[標準偏差{ひょうじゅんへんさ}]	標準偏差を出力します。 入力された値がすべて NULL の場合、エラーが発生します。 数値データ型または数値に変換できる文字列以外を入力すると、エラーが発生します。 【注意事項】 一部の入力値が NULL の場合にも、エラーが発生してしまうことがあります。
[文字列結合]	データを文字列に変換して、値で指定した文字で区切り、連結して出力します。
[重複しない値の数(N)]	ユニークな値が入力されている行数 (NULL を除く) を出力します。 Internet Address 型以外のデータ型を入力できます。

項目名	説明
[Number of rows (without field argument)]	指定したキーと一致する行数を出力します。

2.12.4 [統計]分析クエリー (Analytic query)

あるフィールドについて、N 行前、または N 行後のレコードの値を参照して付与します。参照する値が存在しない場合は、NULL 値を付与します。グループキーを設定すると、キー値ごとにテーブルを分割し、分割したテーブルごとに処理を行います。

次に例を示します。

グループキーはなし、かつ 2 レコードあとの値を参照した場合

入力データ		出力データ		
Key	Data	Key	Data	Data_2R_LEAD
K1	1	K1	1	3
K1	2	K1	2	4
K1	3	K1	3	5
K2	4	K2	4	6
K2	5	K2	5	7
K2	6	K2	6	<null>
K2	7	K2	7	<null>

グループキーは「Key」、かつ 2 レコードあとの値を参照した場合

入力データ		出力データ		
Key	Data	Key	Data	Data_2R_LEAD
K1	1	K1	1	3
K1	2	K1	2	<null>
K1	3	K1	3	<null>
K2	4	K2	4	6
K2	5	K2	5	7
K2	6	K2	6	<null>
K2	7	K2	7	<null>

※出力データは、Key で分割されます。

図 2-156 [分析クエリー]画面

分析クエリー

ステップ名

分析クエリー

グループを形成するフィールド:

#

グループフィールド

1

フィールドの取得

分析ファンクション:

#

フィールド名(新規)

サブジェクト

タイプ

N

1

参照フィールドの取得

ヘルプ

OK

キャンセル

表 2-165 [分析クエリー]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[グループを形成するフィールド]	グループキーとなるフィールドを選択します。 [フィールドの取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
3	[グループフィールド]	グループキーとなるグループフィールド名です。
4	[分析ファンクション]	前後の値を参照するフィールドを選択します。 [参照フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。
5	[フィールド名(新規)]	出力フィールド名です。
6	[サブジェクト]	前後の値を参照するフィールドです。
7	[タイプ]	前のレコードを参照するか、後ろのレコードを参照するかを指定します。 次の項目から指定します。 • [LEAD "N"行進み、サブジェクトを取得] • [LAG "N"行戻り、サブジェクトを取得]
8	[N]	いくつ前のレコード、またはいくつ後ろのレコードを参照するのかを指定します。前と後ろ、どちらのレコードを参照するのかは、[タイプ]の設定内容によって決まります。なお、整数以外（空白、文字、小数）を入力した場合は、1を入力した場合と同じ処理になります。

2.12.5 [統計]単変量統計（Univariate statistics）

入力フィールドのデータからシンプルな単変量統計を計算できます。

図 2-157 [単変量解析]画面

高 単変量解析

ステップ名 単変量統計

入力フィールドおよび導出統計

#	入力フィールド	N	算術平均	標準偏差	最小	最大	中央値	百分位数	内挿法百分位数
1									

ヘルプ OK(Ⓚ) キャンセル(Ⓚ)

表 2-166 [単変量解析]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[入力フィールド]	統計を計算するための入力フィールドを指定します。 数値である入力フィールドだけがリストに表示されます。
3	[N]	[入力フィールド]で指定したフィールドに含まれるデータの個数を出力するかどうかを指定します。 • [True]：出力します。 • [False]：出力しません。
4	[算術平均]	[入力フィールド]で指定したフィールドに含まれるデータの平均値を出力するかどうかを指定します。 • [True]：出力します。 • [False]：出力しません。
5	[標準偏差]	[入力フィールド]で指定したフィールドに含まれるデータの標準偏差を出力するかどうかを指定します。 • [True]：出力します。 • [False]：出力しません。
6	[最小]	[入力フィールド]で指定したフィールドに含まれるデータの最小値を出力するかどうかを指定します。 • [True]：出力します。 • [False]：出力しません。
7	[最大]	[入力フィールド]で指定したフィールドに含まれるデータの最大値を出力するかどうかを指定します。 • [True]：出力します。 • [False]：出力しません。
8	[中央値]	[入力フィールド]で指定したフィールドに含まれるデータの中央値を出力するかどうかを指定します。 • [True]：出力します。 • [False]：出力しません。
9	[百分位数]	0 から 100 の値を入力することで、入力フィールドで選択したフィールドに含まれるデータについての百分位数を出力します。

項番	項目名	説明
10	[内挿法百分位数]	データ内の隣接する値の間を滑らかに補間する、より高度なパーセンタイル計算を使用するかどうかを指定します。 • [True]：使用します。 • [False]：使用しません。 使用する場合の計算式を次に示します。 内挿法百分位数の計算式：(x1*y1)+(x2*y2) (凡例) x1：(i の整数部分の値) +1-i x2：1.0-x1 y1：データの (i の整数部分の値-1) 番目の値 y2：データの (i の整数部分の値) 番目の値 ここで使用する、i は次の計算式で求めます。 $i = p * (n + 1)$ (凡例) p：0 以上 1 以下の値 n：データの個数 i が 1 以下の場合：データの最小値を代入します。 i が n 以上の場合：データの最大値を代入します。

2.12.6 [統計]貯留層サンプリング (Reservoir sampling)

入力データから指定した行をサンプリングします。

図 2-158 [リザーバサンプリング]画面

表 2-167 [リザーバサンプリング]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[サンプルサイズ(行)]	サンプリングする行数を選択します。 0 を設定すると、すべての行がサンプリングされます。 負の値を設定すると、1 行もサンプリングされません。

項番	項目名	説明
3	[乱数の種]	乱数生成に使用するシードを指定します。

2.13 ジョブ

- [ジョブ]変数取得 (Get Variables)
- [ジョブ]変数設定 (Set Variables)
- [ジョブ]行取得 (Get rows from result)
- [ジョブ]行送信 (Copy rows to result)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.13.1 [ジョブ]変数取得 (Get Variables)

入力ストリームのすべてのレコードに対して、フィールドを追加して出力します。フィールドに設定される値は変数で指定できます。指定できる変数は、データ変換のパラメータ、kettle.properties ファイルに定義された PDI の環境変数などです。

[プレビュー]ボタンを押すと、取得する変数の値が確認できます。[変数取得]ボタンを押すと、指定できる変数を自動入力できます。

図 2-159 [変数取得]画面

表 2-168 [変数取得]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	フィールド名を指定します。
3	[変数名]	該当フィールドに出力する変数の値を指定します。変数と定数の組み合わせも指定できます。 変数は、「\${変数}」と記述します。
4	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。

項番	項目名	説明
		【注意事項】 フィールド値の先頭にスペースが入った状態で Integer、Number、または BigNumber の変数値を変数取得ステップで取得すると、エラーが発生します。次のどちらかの方法で対策できます。 - 変数取得ステップの[空白除去]で[両方]を指定する。 - 入力ステップで Integer、Number、または BigNumber のフィールドを定義する場合は、[書式]および[長さ]を指定する。
5	[書式]	入力書式を指定します。
6	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
7	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。
8	[通貨記号]	「\$（ドル）」など、通貨を表す記号を指定します。
9	[小数]	「.（ドット）」など、小数点の文字を指定します。
10	[グループ]※	「,（コンマ）」など、桁を区切る文字を指定します。
11	[空白除去]	文字列の前後にある全半角空白、タブ文字、改行を削除するかどうかを指定します。 [なし] 文字列の前後にある全半角空白、タブ文字、改行を削除しません。 [右] 文字列の前にある全半角空白、タブ文字、改行を削除します。 [左] 文字列の後ろにある全半角空白、タブ文字、改行を削除します。 [両方] 文字列の前後にある全半角空白、タブ文字、改行を削除します。

注※ 画面の項目名と指定する内容が異なります。

2.13.2 [ジョブ]変数設定 (Set Variables)

1 レコードのフィールドの値を変数に設定します。[変数のスコープ]の設定によって、変数を参照できるジョブの範囲が異なります。

[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。

図 2-160 [環境変数設定]画面

環境変数設定

ステップ名

変数設定

初期化適応

☒

フィールド

#	フィールド名	変数名	変数のスコープ	デフォルト
1				

ヘルプ

OK(O)

キャンセル(C)

フィールドを取得(G)

表 2-169 [環境変数設定]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[初期化適応]	この項目にチェックを付けると、日付や数値などの値を書式の設定に従った形式に初期化します。 チェックを付けない場合、初期化しません（書式は無視されてデフォルトの書式になります）。
3	[フィールド名]	フィールド名を指定します。
4	[変数名]	設定する変数名を指定します。
5	[変数のスコープ]	変数の有効範囲を次の項目から指定します。 [Valid in the Java Virtual Machine] この変数は JVM 上で有効です。 [Valid in the parent job] この変数は親ジョブ、および子データ変換で有効です。 [Valid in the grand-parent job] この変数は親ジョブの親ジョブ、すべての子ジョブ、および子データ変換で有効です。 [Valid in the root job] この変数はルートジョブ（最上位のジョブ）、すべてのジョブ、およびデータ変換で有効です。
6	[デフォルト]	デフォルト値として設定する変数の名前を指定します。

2.13.3 [ジョブ]行取得（Get rows from result）

行送信ステップまたはトランスフォーメーション実行ステップがメモリに格納したレコードを取得します。

図 2-161 [前の結果から列を取得]画面

前の結果から列を取得

ステップ名

行取得

フィールド:

#	フィールド名	タイプ	長さ	精度
1				

ヘルプ

OK(O)

キャンセル(C)

表 2-170 [前の結果から列を取得]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	フィールド名を指定します。
3	[タイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[長さ]	該当フィールドのデータの長さを指定します。 データタイプによって、次の項目を指定します。 - Number、BigNumber、Integer の場合 数値中の有効桁数を指定します。 - String、Date、Timestamp の場合 文字列長を指定します。
5	[精度]	データタイプが Number または BigNumber の場合に、浮動小数点の桁数を指定します。

2.13.4 [ジョブ]行送信 (Copy rows to result)

ジョブの次のデータ変換またはジョブエントリに渡すレコードを、メモリに格納します。メモリに格納されたレコードは、行取得ステップで取得できます。

図 2-162 [結果に列コピー]画面

結果に列コピー

ステップ名

行送信

ヘルプ

OK(O)

キャンセル(C)

表 2-171 [結果に列コピー]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

2.14 マッピング

- [マッピング]シンプルマッピング（サブデータ変換）（Simple Mapping (sub-transformation)）
- [マッピング]マッピング（サブデータ変換）（Mapping (sub-transformation)）
- [マッピング]マッピング入力（Mapping input specification）
- [マッピング]マッピング出力（Mapping output specification）

2.14.1 [マッピング]シンプルマッピング（サブデータ変換）（Simple Mapping (sub-transformation)）

指定したデータ変換を呼び出し、入力ストリームのデータを引き渡して実行して、処理結果を後続のステップへ出力します。シンプルマッピングステップで呼び出されたデータ変換は親データ変換の一部として動作します。

シンプルマッピング（サブデータ変換）ステップでは、入力と出力のステップを 1 つだけ指定できます。

(1) [パラメータ]タブ

子データ変換実行時の変数を指定します。

[illegible]

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[トランスフォーメーション]	実行対象のデータ変換のパスを指定します。
3	[変数名]	変数名を指定します。
4	[値]	変数の値として設定する固定値です。
5	[指定されたデータ変換から変数を引き継ぐ]	この項目にチェックを付けると、親データ変換の変数を子データ変換に引き継ぎます。親データ変換から引き継いだ変数の名称と、この画面で指定した[変数名]が重複している場合は、この画面で指定した内容が優先されます。

(3) [出力]タブ

子データ変換のマッピング出力ステップと、親データ変換の出力先ステップの対応付けを定義します。

[Get Fields]ボタンでフィールドを設定することもできます。

図 2-165 [マッピング (サブデータ変換)]画面-[出力]タブ (シンプルマッピングの場合)

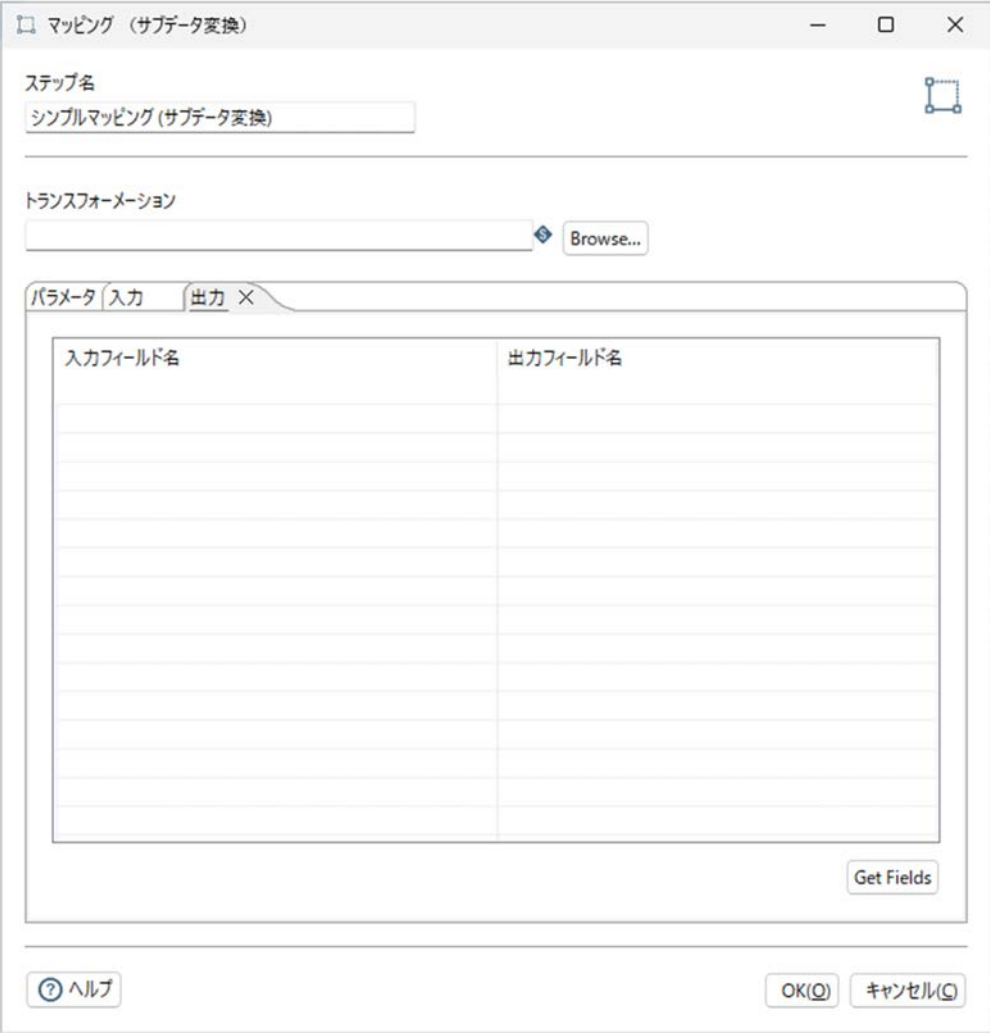


表 2-174 [マッピング (サブデータ変換)]画面-[出力]タブ (シンプルマッピングの場合) の項目

項番	項目名	説明
1	[入力フィールド名]	入力ステップのフィールド名です。
2	[出力フィールド名]	出力ステップのフィールド名です。

2.14.2 [マッピング]マッピング (サブデータ変換) (Mapping (sub-transformation))

指定したデータ変換を呼び出し、入力ストリームのデータを引き渡して実行して、処理結果を後続のステップへ出力します。マッピングステップで呼び出されたデータ変換は親データ変換の一部として動作します。

マッピングステップへの入力および出力は複数ストリームを指定できます。

(1) [パラメータ]タブ

子データ変換実行時の変数を指定します。

図 2-166 [マッピング (サブデータ変換)]画面-[パラメータ]タブ

The screenshot shows a software window titled "マッピング (サブデータ変換)". It contains several sections:

- ステップ名**: A text field containing "マッピング (サブデータ変換)" and a small square icon with dashed lines.
- トランスフォーメーション**: A text field followed by a blue diamond icon and a "Browse..." button.
- パラメータ**: A tabbed interface with three tabs: "パラメータ" (selected), "入力", and "出力". Below the tabs is a table with two columns: "変数名" (Variable Name) and "値" (Value). The table has multiple rows, all currently empty.
- Checkboxes**: At the bottom left, there is a checked checkbox labeled "指定されたデータ変換から変数を引き継ぐ".
- Buttons**: At the bottom right, there are three buttons: "? ヘルプ", "OK(Q)", and "キャンセル(C)".

表 2-175 [マッピング (サブデータ変換)]画面-[パラメータ]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[トランスフォーメーション]	実行対象のデータ変換のパスを指定します。
3	[変数名]	変数名を指定します。
4	[値]	変数の値として設定する固定値です。
5	[指定されたデータ変換から変数を引き継ぐ]	この項目にチェックを付けると、親データ変換の変数を子データ変換に引き継ぎます。親データ変換から引き継いだ変数の名称と、この画面で指定した[変数名]が重複している場合は、この画面で指定した内容が優先されます。

(2) [入力]タブ

マッピングステップへの各入力ストリームと、子データ変換のマッピング入力ステップの対応付けを定義します。

図 2-167 [マッピング (サブデータ変換)]画面-[入力]タブ

マッピング (サブデータ変換)

ステップ名
マッピング (サブデータ変換)

トランスフォーメーション

Browse...

パラメータ

入力

出力

Available inputs: + X

入力

☒ おもなデータ経路にする

入力ステップ名

選択(S)

出力ステップ名

選択(S)

コメント

入力フィールド名

出力フィールド名

☒ 出力時に名前を変更する

マッピング(P)

ヘルプ

OK(O)

キャンセル(C)

表 2-176 [マッピング (サブデータ変換)]画面-[入力]タブの項目

項番	項目名	説明
1	[Available inputs]	対応付けた入力ステップマッピングの一覧です。 [+]アイコンをクリックすると新たな入力の対応付けを定義できます。 削除したい対応付けを選択して[×]アイコンをクリックすると、既存の対応付けを削除できます。
2	[おもなデータ経路にする]	この項目にチェックを付けると、入力ステップ名および出力ステップ名の指定を省略できます。入力ストリームが1つのときだけ有効な設定です。
3	[入力ステップ名]	親データ変換からの入力ステップを指定します。
4	[出力ステップ名]	子データ変換への出力ステップ（マッピング入力ステップ）を指定します。
5	[コメント]	入力ステップマッピングの説明を記載します。
6	[入力フィールド名]	入力ステップのフィールド名です。

2. データ変換

Pentaho 機能説明 PDI 編

312

項番	項目名	説明
7	[出力フィールド名]	出力ステップのフィールド名です。
8	[出力時に名前を変更する]	この項目にチェックを付けると、子データ変換の処理結果をマッピングステップから出力する際に、フィールド名を各出力フィールド名に対応する入力フィールド名に変更して出力します。

(3) [出力]タブ

子データ変換のマッピング出力ステップと、親データ変換の出力先ステップの対応付けを定義します。

図 2-168 [マッピング (サブデータ変換)]画面-[出力]タブ

マッピング (サブデータ変換)

ステップ名

マッピング (サブデータ変換)

トランスフォーメーション

Browse...

パラメータ

入力

出力

Available outputs: + X

出力

☒ おもなデータ経路にする

ステップ名 (入力)

選択(S)

ステップ名 (出力)

選択(S)

コメント

入力フィールド名

出力フィールド名

マッピング(P)

ヘルプ

OK(O)

キャンセル(C)

表 2-177 [マッピング (サブデータ変換)]画面-[出力]タブの項目

項番	項目名	説明
1	[Available outputs]	対応付けた出力ステップマッピングの一覧です。 [+]アイコンをクリックすると新たな出力の対応付けを定義できます。

2. データ変換

Pentaho 機能説明 PDI 編

313

項番	項目名	説明
		削除したい対応付けを選択して[×]アイコンをクリックすると、既存の対応付けを削除できます。
2	[おもなデータ経路にする]	この項目にチェックを付けると、入力ステップ名および出力ステップ名の指定を省略できます。 出力ストリームが1つのときだけ有効な設定です。
3	[ステップ名（入力）]	子データ変換からの入力ステップ（マッピング出力ステップ）を指定します。
4	[ステップ名（出力）]	親データ変換への出力ステップを指定します。
5	[コメント]	出力ステップマッピングの説明を記載します。
6	[入力フィールド名]	入力ステップのフィールド名です。
7	[出力フィールド名]	出力ステップのフィールド名です。

2.14.3 [マッピング]マッピング入力 (Mapping input specification)

このステップが配置されたデータ変換が、マッピングステップで呼び出された際の入力となります。

図 2-169 [マッピング入力]画面

表 2-178 [マッピング入力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[フィールド名]	フィールド名を指定します。
3	[データタイプ]	PDI 上のデータタイプを指定します。 PDI データタイプと Java データタイプの対応については、「 データタイプの対応表 」を参照してください。
4	[長さ]	データの長さを指定します。 データタイプによって、次の項目を指定します。 • Number、BigDecimal、Integer の場合

項番	項目名	説明
		数値中の有効桁数を指定します。 • String、Date、Timestamp の場合 文字列長を指定します。
5	[精度]	浮動小数点の桁数を指定します。 データタイプが Number と BigInteger の場合に指定できます。
6	[未定義のフィールドを含む]	この項目にチェックを付けると、入力データとして定義していないフィールドが入力された場合に、そのまま後続のステップへデータを出力します。 チェックを付けない場合は、このステップで定義したフィールドだけを出力します。

2.14.4 [マッピング]マッピング出力 (Mapping output specification)

このステップが配置されたデータ変換が、マッピングステップで呼び出された際の出力となります。

図 2-170 [マッピング出力]画面

表 2-179 [マッピング出力]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。

2.15 バルクロード

- [バルクロード]PostgreSQL バルクロード (PostgreSQL bulk loader)

上記以外のステップについては、Pentaho Documentation を参照してください。

2.15.1 [バルクロード]PostgreSQL バルクロード (PostgreSQL bulk loader)

PostgreSQL データベースのテーブルにデータを書き込みます。

図 2-171 [PostgreSQL バルクロード]画面

PostgreSQLバルクロード

ステップ名

PostgreSQL bulk loader

データソース名

編集(E)

新規作成(N)...

ウィザード

スキーマ名

テーブル名

target tablename

参照(B)...

ロードアクション

挿入

優先するデータベース名

引用符

"

デリミッター

;

Stop on error

☐

フィールド:

#	テーブルフィールド	ストリームフィールド	日付マスク	
1				

フィールドを取得(F)

マッピング編集

ヘルプ

OK(O)

キャンセル(C)

SQL(S)

表 2-180 [PostgreSQL バルクロード]画面の項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[データソース名]	データソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。

項番	項目名	説明
		[ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。[参照]ボタンを押すと、データベース内のスキーマ名を参照できます。
4	[テーブル名]	テーブル名を指定します。[参照]ボタンを押すと、データベース内のテーブル名を参照できます。
5	[ロードアクション]	データ挿入時の動作を指定します。 • [挿入] 挿入だけを実行します。 • [切り捨て] 挿入前にテーブルの内容を削除します。
6	[優先するデータベース名]	データソース名のデータベース接続設定と異なるデータベースに優先して接続する場合、接続先のデータベース名を指定します。
7	[引用符]	psql コマンド内で使用する引用符を指定します。
8	[デリミッター]	psql コマンド内で使用する区切り文字を指定します。
9	[Stop on error]	この項目にチェックを付けると、psql コマンドがエラーになった場合に、このステップをエラーにします。
10	[フィールド]	[フィールドを取得]ボタンを押すと、入力ストリームのフィールドを自動入力できます。 【注意事項】 Internet Address 型のデータを PostgreSQL バルクロードで挿入すると、エラーが発生します。
11	[テーブルフィールド]	出力先のテーブルのフィールドを指定します。
12	[ストリームフィールド]	入力元の前ステップのフィールドを指定します。
13	[日付マスク]	入力データを PostgreSQL にどのように渡すかを指定します。 • パススルー 入力データをそのままテーブルへ渡します。 • 日付マスク 日付マスクを設定した入力データをテーブルへ渡します。 (例) 入力データ(2015/10/12 12:13:14) テーブル(2015/10/12 00:00:00) • 日付/時間マスク 日付と時間（時間はミリ秒まで）マスクした入力データをテーブルへ渡します。 (例) 入力データ(2015/10/12 12:13:14.123456789) テーブル(2015/10/12 12:13:14.123)

項番	項目名	説明
14	[マッピング編集]ボタン	このボタンを押すと、[マッピング入力]画面を表示します。この画面では、入力ストリームのフィールドと、データベースのカラムのマッピングをします。
15	[SQL]ボタン	このボタンを押すと、SQL を生成します。

2.16 新機能

- [新機能]SFTP put (SFTP put)

2.16.1 [新機能]SFTP put (SFTP put)

ローカルのファイルシステムから FTP サーバへ、SFTP を使用してファイルを転送します。

(1) [一般]タブ

SFTP サーバの情報および拡張設定をします。

図 2-172 [SFTP put]画面-[一般]タブ

The screenshot shows the 'SFTP put' dialog box with the '一般' (General) tab selected. The 'ステップ名' (Step Name) field is set to 'SFTP put'. The 'サーバー設定' (Server Settings) section includes fields for 'サーバー' (Server), 'ポート' (Port) set to 22, 'ユーザー名' (Username), and 'パスワード' (Password). There are checkboxes for 'キーファイルを使用' (Use Key File) and 'キーファイルパスワード' (Key File Password), with a '参照(B)...' (Browse...) button next to the key file field. Below these are fields for 'プロキシタイプ' (Proxy Type), 'プロキシホスト' (Proxy Host), 'プロキシポート' (Proxy Port), 'プロキシユーザー名' (Proxy Username), and 'プロキシパスワード' (Proxy Password). A 'テスト接続' (Test Connection) button is located at the bottom right of the settings section. The '圧縮' (Compression) dropdown is set to 'none'. At the bottom of the dialog are buttons for 'ヘルプ' (Help), 'OK(O)', and 'キャンセル(C)' (Cancel).

表 2-181 [SFTP put]画面-[一般]タブの項目

項番	項目名	説明
1	[ステップ名]	ステップ名を指定します。
2	[サーバー]	SFTP サーバ名または IP アドレスを指定します。
3	[ポート]	SFTP サーバのポート番号を指定します。
4	[ユーザー名]	SFTP サーバに接続するためのユーザー名を指定します。
5	[パスワード]	SFTP サーバに接続するためのパスワードを指定します。
6	[キーファイルを使用]	秘密鍵ファイルを使用する場合に指定します。
7	[キーファイル]	秘密鍵ファイルのパスを指定します。 [キーファイルを使用]にチェックを付けると指定できます。
8	[キーファイルパスワード]	パスフレーズが必要な場合にパスワードを入力します。 [キーファイルを使用]にチェックを付けると指定できます。
9	[プロキシタイプ]	プロキシサーバのプロキシタイプとして、次のどちらかを指定します。 • HTTP • SOCKS5
10	[プロキシホスト]	SFTP プロキシサーバのホスト名を指定します。
11	[プロキシポート]	SFTP プロキシサーバのポート番号を指定します。
12	[プロキシユーザー名]	SFTP プロキシサーバに接続するためのユーザー名を指定します。
13	[プロキシパスワード]	SFTP プロキシサーバに接続するためのパスワードを指定します。
14	[圧縮]	ファイル転送時の圧縮形式として、次のどちらかを指定します。 • none • zlib

(2) [ファイル]タブ

配置するファイルの情報を設定します。

図 2-173 [SFTP put]画面-[ファイル]タブ

表 2-182 [SFTP put]画面-[ファイル]タブの項目

項番	項目名	説明
1	[入力ストリーム]	この項目にチェックを付けると、[ソースファイル名フィールド]に指定したファイルの内容を、メモリから読み取ります。この項目にチェックを付けた場合、[リモートファイル名]を指定する必要があります。 チェックを付けない場合、[ソースファイル名フィールド]に指定したファイルをそのまま読み取ります。
2	[ソースファイル名フィールド]	前のステップから転送元のファイルパスを含むフィールドを指定します。
3	[ファイル名を結果に加える]	ファイル転送に使用したファイルパスをメモリに格納します。ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。 [入力ストリーム]にチェックを付けている場合、この項目は非活性になります。
4	[SFTP Put の後]	[何もしない]を指定してください。それ以外は使用しないでください。
5	[移動先フォルダ]	この機能は使用しないでください。

項番	項目名	説明
6	[保存先フォルダを作成]	この機能は使用しないでください。
7	[リモートディレクトリ]	前のステップからファイルを配置する SFTP サーバ上のディレクトリを含むフィールドを指定します。
8	[フォルダ作成]	宛先フォルダが存在しない場合、フォルダを作成する必要があるときに指定します。
9	[リモートファイル名]	前のステップからリモートファイル名を含むフィールドを指定します。 リモートディレクトリに配置するファイルの名称を指定できます。 [入力ストリーム]にチェックを付けている場合、確認できます。

3

ジョブ

この章では、PDI クライアント（Spoon）の[データ変換]パースペクティブの[ジョブ]画面を使用してジョブを定義する際に、使用できるジョブエントリーについて説明します。

3.1 ジョブエントリーの一覧

PDI クライアント (Spoon) で使用できるジョブエントリーの一覧を次の表に示します。

表 3-1 PDI クライアント (Spoon) で使用できるジョブエントリーの一覧

カテゴリ	ジョブエントリー名	説明
全般	START (Start)	ジョブ実行の開始点を定義します。
	Dummy (Dummy)	入力ストリームのデータに対して、何も処理をしないで後続のジョブエントリーに出力します。
	ジョブ (Job)	定義されたジョブ (サブジョブ) を実行します。
	データ変換 (Transformation)	指定したデータ変換を呼び出し、入力ストリームのデータに応じた設定でデータ変換を実行します。
	変数設定 (Set variables (job entry))	ジョブエントリーやデータ変換を実行する場合の変数を設定します。
	成功 (Success)	エラー状態のジョブを強制的に成功状態にします。エラー情報はすべて削除されます。
メール	メール (Mail)	電子メールを送信します。
ファイル管理	HTTP (HTTP)	HTTP プロトコルを使用して Web サーバからファイルを転送します。
	ファイルへ書き込み (Write to file)	ファイルにテキストデータを書き込みます。
	ファイルコピー (Copy Files)	指定したファイルを、指定した出力先にコピーします。
	ファイル作成 (Create file)	空のファイルを作成します。
	ファイル削除 (Delete file)	指定した 1 ファイルだけを削除します。
	ファイル削除 (複数) (Delete files)	指定したファイルをすべて削除します。
	ファイル圧縮 (Zip file)	参照元のファイルとディレクトリを圧縮します。
	ファイル待機 (Wait for a file)	設定した時間内に、指定したファイルが存在するかどうかを確認します。
	ファイル移動 (Move files)	ファイルを移動します。
	ファイル解凍 (Unzip file)	1 つまたは複数の zip ファイルを解凍します。
	フォルダー作成 (Create a folder)	指定したフォルダを作成します。
	フォルダー削除 (Delete folders)	指定したフォルダを削除します。指定したフォルダ内にファイルがある場合は、フォルダ内のファイルもすべて削除します。
条件	テーブル検証 (Evaluate rows number in a table)	テーブルの内容を評価します。

カテゴリ	ジョブエントリ名	説明
	テーブル確認 (Table exists)	指定したテーブルがデータベースに存在するかどうかを確認します。
	データベース接続確認 (Check Db Connections)	データベースへの接続を確認します。
	ファイル確認 (File Exists)	指定した 1 ファイルが存在するかどうかを確認してから、後続のジョブエントリに遷移します。
	ファイル確認 (複数) (Checks if files exists)	指定した複数のファイルまたはディレクトリが存在するかどうかを確認してから、後続のジョブエントリに遷移します。
	待機 (Wait for)	次のジョブエントリの実行を待機します。
	簡易検証 (Simple evaluation)	行送信したフィールドや変数の値を検証します。
スクリプト	JavaScript (JavaScript)	真偽値を計算する式を入力し、式の結果で次に実行するエントリを決定します。
	SQL (SQL)	[SQL ステートメント]に指定した SQL 文、または SQL ファイルに記載された SQL 文を実行します。
	シェル (Shell)	定義したシェルスクリプトを、ジョブが実行されているホストで実行します。
ユーティリティー	Ping 送信 (Ping a host)	ICMP プロトコルを使用してホストに ping を送信します。
	ログ書き込み (Write to log)	指定した件名およびメッセージをログに出力します。
	停止 (Abort job)	ジョブを停止する場合に使用します。
ファイル転送	FTP ファイル取得 (Get a file with FTP)	FTP サーバ上のファイルを取得して、ローカルのファイルシステムへ転送します。ファイル転送時のプロトコルには FTP を使用します。
	FTP ファイル追加 (Put a file with FTP)	ローカルのファイルシステム上のファイルを FTP サーバへ転送 (追加) します。ファイル転送時のプロトコルには FTP を使用します。
	FTP 削除 (FTP delete)	プロトコルに FTP、FTPS、SFTP、または SSH のどれかを使用して、サーバ上のファイルを削除します。
	SFTP ファイル取得 (Get a file with SFTP)	SFTP サーバ上のファイルを取得して、ローカルのファイルシステムへ転送します。ファイル転送時のプロトコルには SFTP を使用します。
	SFTP ファイル追加 (Put a file with SFTP)	ローカルのファイルシステム上のファイルを SFTP サーバへ転送 (追加) します。ファイル転送時のプロトコルには SFTP を使用します。

この表に記載されていないジョブエントリについては、Pentaho Documentation を参照してください。

3.1.1 ジョブエントリーの説明について

変数について

変数は「\${変数}」と記述します。テーブル名や列名にも変数を使用できます。Ctrl キーとスペースキーを同時に押すと、使用できる変数の一覧が表示されます。

3.2 全般

- [全般]START (Start)
- [全般]Dummy (Dummy)
- [全般]ジョブ (Job)
- [全般]データ変換 (Transformation)
- [全般]変数設定 (Set variables (job entry))
- [全般]成功 (Success)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.2.1 [全般]START (Start)

ジョブ実行の開始点を定義します。ジョブには 1 つの Start エントリーが必要です。

図 3-1 [ジョブスケジュール]画面

Job entry name: Start

繰り返し: ☐

実行間隔: スケジュール無し

秒: 0

分: 60

毎時: 12 0

曜日: 月曜日

日: 1

ヘルプ OK(O) キャンセル(C)

表 3-2 [ジョブスケジュール]画面の項目

項番	項目名	説明
1	[Job entry name]	ジョブ名を指定します。
2	[繰り返し]	この項目にチェックを付けると、ジョブが繰り返し実行されます。 【注意事項】 この項目にチェックを付けることは推奨しません。ジョブ実行のスケジュールは、PDI クライアント (Spoon) の[Scheduler]パースペクティブで設定してください。
3	[実行間隔]	ジョブの実行契機を次の項目から指定します。 • [スケジュール無し]

項番	項目名	説明
		[繰り返し]にチェックを付けた場合、停止操作※が行われるまで繰り返しジョブを実行します。 [繰り返し]にチェックを付けない場合、ジョブを1回実行します。 • [間隔] [繰り返し]にチェックを付けた場合、[秒]と[分]で指定した実行間隔で、停止操作※が行われるまで繰り返しジョブを実行します。 [繰り返し]にチェックを付けない場合、[秒]と[分]で指定した時間後にジョブを1回実行します。 • [毎日] [繰り返し]にチェックを付けた場合、[毎時]で指定した時間に、停止操作※が行われるまで毎日繰り返しジョブを実行します。 [繰り返し]にチェックを付けない場合、[毎時]で指定した時間にジョブを1回実行します。 指定した時間を過ぎている場合は、[繰り返し]のチェックの有無に関係なく翌日に実行します。 • [毎週] [繰り返し]にチェックを付けた場合、[曜日]で指定した曜日の[毎時]で指定した時間に、停止操作※が行われるまで毎週繰り返しジョブを実行します。 [繰り返し]にチェックを付けない場合、[曜日]で指定した曜日の[毎時]で指定した時間にジョブを1回実行します。 指定した曜日の指定した時間を過ぎている場合は、[繰り返し]のチェックの有無に関係なく翌週に実行します。 • [毎月] [繰り返し]にチェックを付けた場合、[日]で指定した日にちの[毎時]で指定した時間に、停止操作※が行われるまで毎月繰り返しジョブを実行します。 [繰り返し]にチェックを付けない場合、[日]で指定した日にちの[毎時]で指定した時間にジョブを1回実行します。 指定した日にちの指定した時間を過ぎている場合は、[繰り返し]のチェックの有無に関係なく、翌月に実行します。 存在しない日付（例えば2/30）を指定した場合、実行されません。また、毎月31日は指定できません。 注※ 停止操作とは、例えば、PDI クライアント（Spoon）の[停止]ボタンを押したり、REST API でジョブを停止させたりする操作のことです。 【注意事項】 この項目を設定することは推奨しません。ジョブ実行のスケジュールは、PDI クライアント（Spoon）の[Scheduler]パースペクティブで設定してください。
4	[秒]	何秒後に実行するかを指定します。 【注意事項】 この項目を設定することは推奨しません。ジョブ実行のスケジュールは、PDI クライアント（Spoon）の[Scheduler]パースペクティブで設定してください。
5	[分]	何分後に実行するかを指定します。

項番	項目名	説明
		【注意事項】 この項目を設定することは推奨しません。ジョブ実行のスケジュールは、PDI クライアント（Spoon）の[Scheduler]パースペクティブで設定してください。
6	[毎時]	何時何分 to 実行するかを指定します。 【注意事項】 この項目を設定することは推奨しません。ジョブ実行のスケジュールは、PDI クライアント（Spoon）の[Scheduler]パースペクティブで設定してください。
7	[曜日]	何曜日に実行するかを指定します。 【注意事項】 この項目を設定することは推奨しません。ジョブ実行のスケジュールは、PDI クライアント（Spoon）の[Scheduler]パースペクティブで設定してください。
8	[日]	何日に実行するかを指定します。 【注意事項】 存在しない日付（例えば 2/30）を指定した場合、実行されません。また、毎月 31 日は指定できません。 この項目を設定することは推奨しません。ジョブ実行のスケジュールは、PDI クライアント（Spoon）の[Scheduler]パースペクティブで設定してください。

3.2.2 [全般]Dummy (Dummy)

入力ストリームのデータに対して、何も処理をしないで後続のジョブエントリーに出力します。

図 3-2 [Dummy]画面

表 3-3 [Dummy]画面の項目

項番	項目名	説明
1	[Job entry name]	ジョブ名を指定します。

3.2.3 [全般]ジョブ (Job)

定義されたジョブ（サブジョブ）を実行します。

(1) [Options]タブ

実行対象のジョブ（サブジョブ）に関する設定をします。

図 3-3 [Job]画面-[Options]タブ



表 3-4 [Job]画面-[Options]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[Job]	実行するジョブを指定します。 [参照]ボタンを押すと、リポジトリに接続していない場合は PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。リポジトリに接続している場合はリポジトリを参照できます。
3	[Run configuration]	ジョブを実行する場所を指定します。
4	[ループ処理を行う]	この項目にチェックを付けると、先行エントリーのデータ変換において、行送信で送信したデータの入力行数分、データ変換をループ実行します。

項番	項目名	説明
5	[ジョブをスレーブサーバにエクスポート]	この機能の動作は不明です。
6	[サーバーの子ジョブと子トランスフォーメーションを展開する]	この機能の動作は不明です。
7	[リモートでのジョブ完了を待つ]	この機能の動作は不明です。
8	[ジョブが失敗した場合はローカルで実行する]	この機能の動作は不明です。

(2) [ログ設定]タブ

ログファイルに関する設定をします。

図 3-4 [Job]画面-[ログ設定]タブ

表 3-5 [Job]画面-[ログ設定]タブの項目

項番	項目名	説明
1	[ログファイルを設定する]	この項目にチェックを付けると、ログファイルの詳細を設定できます。
2	[ログファイル名]	ログファイルのファイルパスを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[拡張子]	ログファイルの拡張子を指定します。
4	[ログレベル]	出力するログのログレベルを次の項目から指定します。 • [ログなし] ログを出力しません。

項番	項目名	説明
		• [エラーログのみ] エラーログだけを出力します。 • [最小限のログ] 最小限のログだけを出力します。 • [基本のログ] デフォルトのログレベルです。 • [詳細なログ] 詳細なログを出力します。 • [デバッグ] 非常に詳細なログを出力します。目的がデバッグの場合に設定します。 • [ローレベル (非常に詳細)] 大量のログデータを出力します。
5	[既存のログファイルに追加する]	この項目にチェックを付けると、ログファイルがある場合に、ログファイルに追記する形でログデータを出力します。
6	[親フォルダーを作成]	この項目にチェックを付けると、[ファイル名]で指定したパスの中のディレクトリがない場合に、そのディレクトリを自動作成します。 チェックを付けない場合、[ファイル名]で指定したパスの中のディレクトリがないときは、エラーが発生します。
7	[ファイル名に日付を付ける]	この項目にチェックを付けると、ログデータを出力したときの日付（形式：yyyyMMdd）をファイル名の末尾に追加します。
8	[ファイル名に時刻を付ける]	この項目にチェックを付けると、ログデータを出力したときの時刻（形式：HHmmss）をファイル名の末尾に追加します。

(3) [Arguments]タブ

引数に関する設定をします。


Job

ジョブ名

ジョブ






Job:

参照(B)

Options ログ設定 Arguments パラメータ

☐ 先行のジョブエントリから値を引き継ぐ

Argument

ヘルプ

OK(O)

キャンセル(C)

項番	項目名	説明
1	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行エントリーのジョブエントリーまたはデータ変換エントリーから行送信ステップで渡された結果を、ジョブの引数として渡します。
2	[Argument]	ジョブに渡す引数を指定します。

パラメータに関する設定をします。

図 3-6 [Job]画面-[パラメータ]タブ

表 3-7 [Job]画面-[パラメータ]タブの項目

項番	項目名	説明
1	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行のジョブエントリーの実行結果を、パラメータとして呼び出し先のジョブに渡します。
2	[サブジョブに全てのパラメーターを渡す]	この項目にチェックを付けると、このジョブのすべてのパラメータを呼び出し先のジョブに渡します。
3	[パラメータ]	呼び出し先のジョブに渡すパラメータ名を指定します。
4	[フィールド名]	先行のジョブエントリーの結果からパラメータに設定する値をフィールド名で指定します。
5	[値]	パラメータに設定する値を指定します。

項番	項目名	説明
6	[パラメーター取得]ボタン	指定したジョブのパラメータを自動入力します。

3.2.4 [全般]データ変換（Transformation）

指定したデータ変換を呼び出し、入力ストリームのデータに応じた設定でデータ変換を実行します。

(1) [Options]タブ

データ変換対象のジョブに関する設定をします。

図 3-7 [この変換ジョブの明細]画面-[Options]タブ

この変換ジョブの明細:

ジョブ名
データ変換

Transformation:
参照(B)

Options ログ設定 Arguments パラメータ

Run configuration:
Pentaho local

Execution

- ☐ ループ処理を行う
- ☐ 実行前にレコードのリストを消去する
- ☐ 実行前にファイルのリストを消去する
- ☒ リモートでのジョブ完了を待つ
- ☐ ジョブが失敗した場合はローカルで実行する
- ☐ Suppress result data from remote transformation

ヘルプ OK(O) キャンセル(C)

表 3-8 [この変換ジョブの明細]画面-[Options]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[Transformation]	実行対象のデータ変換のパスを指定します。 [参照]ボタンを押すと、リポジトリに接続していない場合は PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。リポジトリに接続している場合はリポジトリを参照できます。
3	[Run configuration]	データ変換を実行する場所を指定します。
4	[ループ処理を行う]	この項目にチェックを付けると、先行エントリーのデータ変換において、行送信で送信したデータの入力行数分、データ変換をループ実行します。
5	[実行前にレコードのリストを 消去する]	この項目にチェックを付けると、結果レコードを削除してから、データ変換を実行します。
6	[実行前にファイルのリストを 消去する]	この項目にチェックを付けると、結果ファイルを削除してから、データ変換を実行します。
7	[リモートでのジョブ完了を待つ]	この機能の動作は不明です。
8	[ジョブが失敗した場合はローカルで実行する]	この機能の動作は不明です。
9	[Suppress result data from remote transformation]	この機能の動作は不明です。

(2) [ログ設定]タブ

ログファイルに関する設定をします。

図 3-8 [この変換ジョブの明細]画面-[ログ設定]タブ

この変換ジョブの明細:

ジョブ名
データ変換

Transformation:
参照(B)

Options ログ設定 Arguments パラメータ

☐ ログファイルを設定する

Logfile

ログファイル名
参照

拡張子
参照

ログレベル
基本のログ

☐ 既存のログファイルに追加する
☐ 親フォルダーを作成
☐ ファイル名に日付を付ける
☐ ファイル名に時刻を付ける

ヘルプ OK(O) キャンセル(C)

表 3-9 [この変換ジョブの明細]画面-[ログ設定]タブの項目

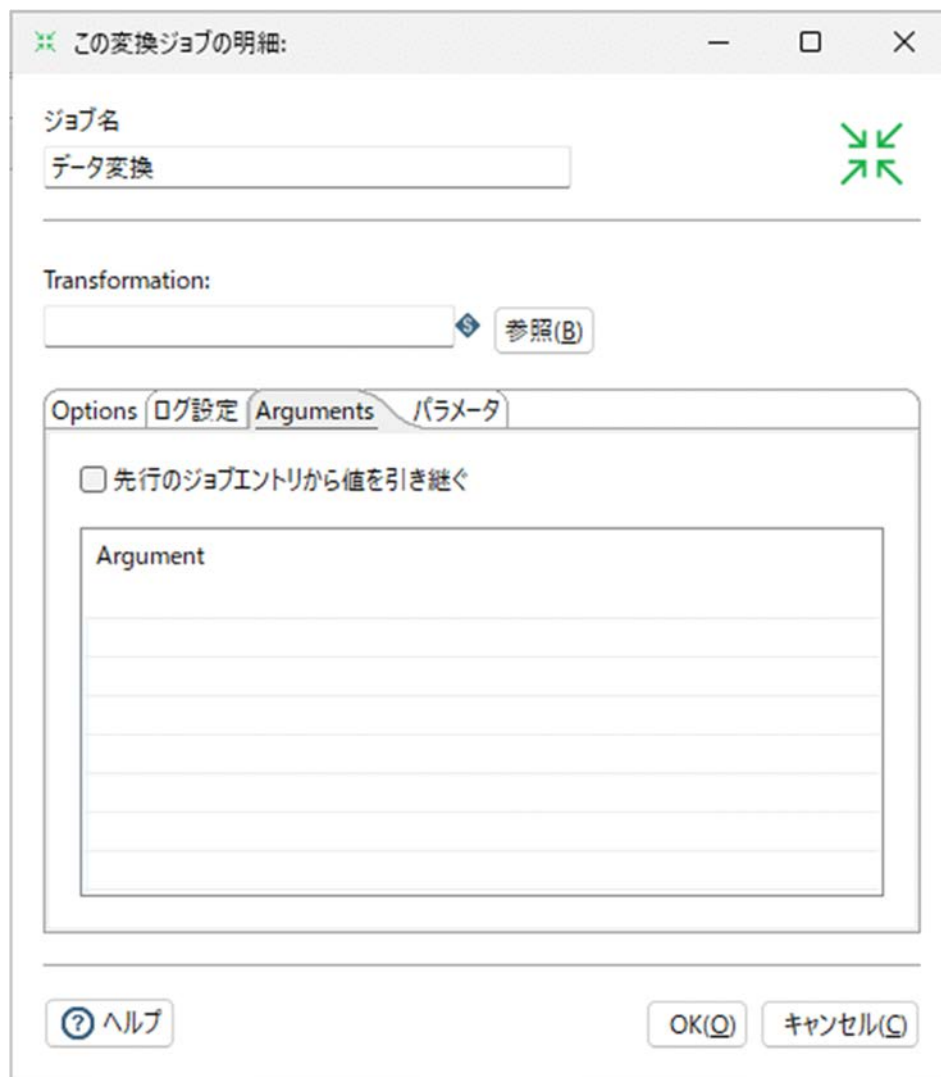
項番	項目名	説明
1	[ログファイルを設定する]	この項目にチェックを付けると、ログファイルの詳細を設定できます。
2	[ログファイル名]	ログファイルのファイルパスを指定します。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
3	[拡張子]	ログファイルの拡張子を指定します。
4	[ログレベル]	出力するログのログレベルを次の項目から指定します。 • [ログなし]

項番	項目名	説明
		ログを出力しません。 • [エラーログのみ] エラーログだけを出力します。 • [最小限のログ] 最小限のログだけを出力します。 • [基本のログ] デフォルトのログレベルです。 • [詳細なログ] 詳細なログを出力します。 • [デバッグ] 非常に詳細なログを出力します。目的がデバッグの場合に設定します。 • [ローレベル (非常に詳細)] 大量のログデータを出力します。
5	[既存のログファイルに追加する]	この項目にチェックを付けると、ログファイルがある場合に、ログファイルに追記する形でログデータを出力します。
6	[親フォルダーを作成]	この項目にチェックを付けると、[ファイル名]で指定したパスの中のディレクトリがない場合に、そのディレクトリを自動作成します。 チェックを付けない場合、[ファイル名]で指定したパスの中のディレクトリがないときは、エラーが発生します。
7	[ファイル名に日付を付ける]	この項目にチェックを付けると、ログデータを出力したときの日付（形式：yyyyMMdd）をファイル名の末尾に追加します。
8	[ファイル名に時刻を付ける]	この項目にチェックを付けると、ログデータを出力したときの時刻（形式：HHmmss）をファイル名の末尾に追加します。

(3) [Arguments]タブ

引数に関する設定をします。

図 3-9 [この変換ジョブの明細]画面-[Arguments]タブ



この変換ジョブの明細:

ジョブ名
データ変換

Transformation:

Options ログ設定 **Arguments** パラメータ

☐ 先行のジョブエントリから値を引き継ぐ

Argument

表 3-10 [この変換ジョブの明細]画面-[Arguments]タブの項目

項番	項目名	説明
1	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行エントリーのジョブエントリーまたはデータ変換エントリーから行送信ステップで渡された結果を、データ変換の引数として渡します。
2	[Argument]	データ変換に渡す引数を指定します。

(4) [パラメータ]タブ

パラメータに関する設定をします。

図 3-10 [この変換ジョブの明細]画面-[パラメータ]タブ

この変換ジョブの明細:

ジョブ名
データ変換

Transformation:
参照(B)

Options ログ設定 Arguments **パラメータ**

☐ 先行のジョブエントリから値を引き継ぐ
☒ サブ変換に全ての値を渡す

パラメータ	フィールド名	値

パラメーター取得

ヘルプ OK(O) キャンセル(C)

表 3-11 [この変換ジョブの明細]画面-[パラメータ]タブの項目

項番	項目名	説明
1	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行のジョブエントリーの実行結果を、パラメータとしてデータ変換に渡します。
2	[サブ変換に全ての値を渡す]	この項目にチェックを付けると、このジョブのすべてのパラメータをデータ変換に渡します。
3	[パラメータ]	データ変換に渡すパラメータ名を指定します。
4	[フィールド名]	先行のジョブエントリーの結果からパラメータに設定する値をフィールド名で指定します。
5	[値]	パラメータに設定する値を指定します。
6	[パラメーター取得]ボタン	指定したデータ変換のパラメータを自動入力します。

3.2.5 [全般]変数設定 (Set variables (job entry))

ジョブエントリーやデータ変換を実行する場合の変数を設定します。

図 3-11 [変数を設定]画面

変数を設定

Job entry name: 変数設定

プロパティファイル

プロパティファイルの名前

変数のスコープ 有効なJava仮想マシン

設定

変数を置き換える ☒

変数

#	変数名	Value	変数のスコープ
1			

ヘルプ

OK(O)

キャンセル(C)

表 3-12 [変数を設定]画面の項目

項番	項目名	説明
1	[Job entry name]	ジョブ名を指定します。
2	[プロパティファイルの名前]	プロパティファイルのファイルパスを指定します。 通常は、kettle.properties ファイルを指定します。
3	[変数のスコープ]	変数の有効範囲を次の項目から指定します。 [有効な Java 仮想マシン] 変数は JVM 上で有効です。 [有効な現在のジョブ] 変数は親ジョブだけで有効です。 [有効な親ジョブ] 変数は親ジョブ、すべての子ジョブ、およびデータ変換で有効です。 [有効な管理ジョブ] 変数はルートジョブ、すべての子ジョブ、およびデータ変換で有効です。
4	[変数を置き換える]	この項目にチェックを付けると、同じ変数名が存在する場合に、その変数名を[Value]の値に置き換えます。
5	[変数名]	変数名を指定します。

項番	項目名	説明
6	[Value]	変数の値を指定します。
7	[変数のスコープ]	変数の有効範囲を次の項目から指定します。 • [有効な Java 仮想マシン] この変数は JVM 上で有効です。 • [有効な現在のジョブ] この変数は親ジョブだけで有効です。 • [有効な親ジョブ] この変数は親ジョブ、すべての子ジョブ、および変換で有効です。 • [有効な管理ジョブ] この変数はルートジョブ、すべての子ジョブ、およびデータ変換で有効です。

3.2.6 [全般]成功 (Success)

エラー状態のジョブを強制的に成功状態にします。エラー情報はすべて削除されます。

図 3-12 [成功ジョブ]画面

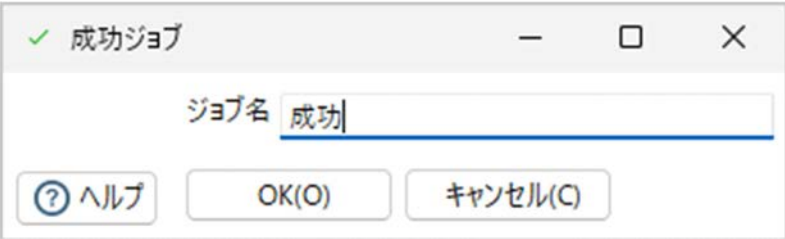


表 3-13 [成功ジョブ]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。

3.3 メール

- [メール]メール (Mail)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.3.1 [メール]メール (Mail)

電子メールを送信します。

(1) [アドレス]タブ

宛先や差出人について設定します。

図 3-13 [ジョブメール詳細]画面-[アドレス]タブ

ジョブメール詳細

ジョブ名 Mail

アドレス

サーバー

メッセージ

添付ファイル

宛先設定

宛先

Cc:

Bcc:

差出人情報

名前

メールアドレス

返信先

返信名

電話番号

ヘルプ

OK(O)

キャンセル(C)

表 3-14 [ジョブメール詳細]画面-[アドレス]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。

項番	項目名	説明
2	[宛先]	宛先のメールアドレスを指定します。 スペースで区切ると複数のアドレスを指定できます。
3	[Cc]	メールアドレスを指定します。 スペースで区切ると複数のアドレスを指定できます。 メッセージのコピーが送信され、他の受信者のメールにもここで指定したメールアドレスが表示されます。
4	[BCc]	メールアドレスを指定します。 スペースで区切ると複数のアドレスを指定できます。 他の受信者のメールにはメールアドレスが表示されません。
5	[名前]	送信者の名前を指定します。
6	[メールアドレス]	送信者のメールアドレスを指定します。メールヘッダーの「From」に該当します。
7	[返信先]	返信先のメールアドレスを指定します。「Reply」の送信先に該当します。
8	[返信名]	メールに記載する担当者の名前を指定します。「Contact information」の「Person to contact」に該当します。
9	[電話番号]	メールに記載する連絡先の電話番号を指定します。「Contact information」の「Telephone number」に該当します。

(2) [サーバー]タブ

使用する SMTP サーバについて設定します。

図 3-14 [ジョブメール詳細]画面-[サーバー]タブ

ジョブメール詳細

ジョブ名 Mail

アドレスサーバーメッセージ添付ファイル

サーバー設定

SMTPサーバー

ポート番号

認証設定

認証を使用する

ユーザ名

パスワード

セキュリティで保護された認証を使用する

タイプ SSL

ヘルプ

OK(O)

キャンセル(C)

表 3-15 [ジョブメール詳細]画面-[サーバー]タブの項目

項番	項目名	説明
1	[SMTP サーバー]	SMTP サーバアドレスを指定します。
2	[ポート番号]	SMTP サーバのポート番号を指定します。
3	[認証を使用する]	この項目にチェックを付けると、SMTP サーバへの認証の使用を有効にします。
4	[ユーザ名]	SMTP ユーザのアカウント名を指定します。 [認証を使用する]にチェックを付けると指定できます。
5	[パスワード]	SMTP ユーザのパスワードを指定します。 [認証を使用する]にチェックを付けると指定できます。
6	[セキュリティで保護された 認証を使用する]	この項目にチェックを付けると、認証時にコネクションタイプ（SSL または TLS） を使用できます。
7	[タイプ]	コネクションタイプを次の項目から指定します。 • [SSL] • [TLS]（STARTTLS） [セキュリティで保護された認証を使用する]にチェックを付けると指定できます。

(3) [メッセージ]タブ

メールのテキスト内容について設定します。

図 3-15 [ジョブメール詳細]画面-[メッセージ]タブ

ジョブメール詳細

ジョブ名 Mail

アドレス | サーバー | **メッセージ** | 添付ファイル

メール設定

メッセージに日付を含む

コメントのみ送信

HTMLメールを使用する

文字コード UTF-8

レベルを管理

優先度 普通

緊急度 普通

Sensitivity Normal

メッセージ

件名

本文

ヘルプ

OK(O)

キャンセル(C)

表 3-16 [ジョブメール詳細]画面-[メッセージ]タブの項目

項番	項目名	説明
1	[メッセージに日付を含む]	この項目にチェックを付けると、メッセージに日付を含めます。「Job」の「Message date」が追加されて、日時が入ります。
2	[コメントのみ送信]	この項目にチェックを付けると、メールにはジョブの実行に関する情報は含まれません。 チェックを付けない場合、メールにはジョブの実行に関する情報が含まれます。
3	[HTML メールを使用する]	この項目にチェックを付けると、メッセージは HTML 形式で送信されます。「Context-Type」は「text/html」です。
4	[文字コード]	文字コードを指定します。 [HTML メールを使用する]にチェックを付けると指定できます。
5	[レベルを管理]	この項目にチェックを付けると、「優先度」「緊急度」「重要度」を選択できるようになります。
6	[優先度]	優先度を次の項目から指定します。

項番	項目名	説明
		• [低] • [普通] • [高] 選択した優先度によって、メールヘッダーの「X-Priority」に次の値が表示されます。 • 低を選択した場合：3 • 普通を選択した場合：2 • 高を選択した場合：1
7	[緊急度]	緊急度を次の項目から指定します。 • [低] • [普通] • [高] 選択した緊急度によって、メールヘッダーの「Importance」に次の値が表示されます。 • 低を選択した場合：low • 普通を選択した場合：normal • 高を選択した場合：high
8	[重要度]	重要度を次の項目から指定します。 • [Normal] • [Personal] • [Private] • [Confidential] 選択した重要度によって、メールヘッダーの「Sensitivity」に次の値が表示されます。 • Normal を選択した場合：normal • Personal を選択した場合：personal • Private を選択した場合：private • Confidential を選択した場合：company-confidential
9	[件名]	メールの件名を指定します。
10	[本文]	メール本文の内容を指定します。

(4) [添付ファイル]タブ

添付ファイルについて設定します。

図 3-16 [ジョブメール詳細]画面-[添付ファイル]タブ

ジョブメール詳細

ジョブ名 Mail

アドレス | サーバー | メッセージ | 添付ファイル

添付ファイル設定

ファイルを添付する ☐

タイプ

一般
ログ
エラー行
エラー
警告

圧縮アーカイブする ☐

圧縮アーカイブファイル名

埋め込み画像

ファイル名

追加

ファイル参照

コンテンツID

埋め込み画像

#	画像	コンテンツID
1		

削除

編集

ヘルプ

OK

キャンセル

表 3-17 [ジョブメール詳細]画面-[添付ファイル]タブの項目

項番	項目名	説明
1	[ファイルを添付する]	この項目にチェックを付けると、メールにファイルを添付できます。
2	[タイプ]	送信するファイルタイプを次の項目から指定します。 • [一般] • [ログ] • [エラー行] • [エラー] • [警告] [ファイルを添付する]にチェックを付けると指定できます。
3	[圧縮アーカイブする]	この項目にチェックを付けると、添付ファイルを zip 形式で作成できます。 [ファイルを添付する]にチェックを付けると指定できます。
4	[圧縮アーカイブファイル名]	zip 形式のファイル名を指定します。 [ファイルを添付する]にチェックを付けると指定できます。
5	[ファイル名]	メールに組み込む画像ファイルを指定します。 メール本文中に次の文字列を指定します。「myimageid」はコンテンツ ID です。

項番	項目名	説明
		[メッセージ]タブの[HTML メールを使用する]にチェックを付けると指定できます。
6	[画像]	[追加]ボタンを押すと、ファイル名に指定した内容が[埋め込み画像]の欄に反映されます。
7	[コンテンツ ID]	[ファイル名]に画像ファイルを指定すると、自動で設定されます。 [メッセージ]タブの[HTML メールを使用する]にチェックを付けると指定できます。

3.4 ファイル管理

- [ファイル管理]HTTP (HTTP)
- [ファイル管理]ファイルへ書き込み (Write to file)
- [ファイル管理]ファイルコピー (Copy Files)
- [ファイル管理]ファイル作成 (Create file)
- [ファイル管理]ファイル削除 (Delete file)
- [ファイル管理]ファイル削除 (複数) (Delete files)
- [ファイル管理]ファイル圧縮 (Zip file)
- [ファイル管理]ファイル待機 (Wait for a file)
- [ファイル管理]ファイル移動 (Move files)
- [ファイル管理]ファイル解凍 (Unzip file)
- [ファイル管理]フォルダー作成 (Create a folder)
- [ファイル管理]フォルダー削除 (Delete folders)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.4.1 [ファイル管理]HTTP (HTTP)

HTTP プロトコルを使用してファイルを転送します。

(1) [全般]タブ

転送対象のファイルに関する設定をします。

図 3-17 [HTTP を利用してファイルを転送]画面-[全般]タブ

表 3-18 [HTTP を利用してファイルを転送]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[URL]	ダウンロードするファイルの URL、またはアップロードするファイルの格納先ディレクトリの URL を指定します。
3	[先行のジョブ登録からの値を URL として使用する]	この項目にチェックを付けると、先行ジョブの実行結果を URL として使用します。
4	[フィールド名]	先行ジョブの実行結果を URL として使用する場合、先行のジョブエントリーまたはデータ変換エントリーから行送信された、ファイルの URL の指定を含んだフィールドを指定します。
5	[Input field which contains upload file name]	先行ジョブの実行結果を URL として使用する場合、先行のジョブエントリーまたはデータ変換エントリーから行送信された、アップロードするローカルファイルの指定を含んだフィールドを指定します。
6	[Input field which contains destination file name]	先行ジョブの実行結果を URL として使用する場合、先行のジョブエントリーまたはデータ変換エントリーから行送信された、アップロード先の指定を含んだフィールドを指定します。

項番	項目名	説明
7	[ユーザ名]	認証が必要な場合、ログイン時のユーザー名を指定します。
8	[パスワード]	認証が必要な場合、ログイン時のユーザーパスワードを指定します。
9	[プロキシホスト名]	URL 接続時に使用するプロキシサーバのホスト名を指定します。
10	[プロキシポート番号]	URL 接続時に使用するプロキシサーバのポート番号を指定します。
11	[無視するプロキシホスト名]	プロキシを経由しないで接続する HTTP サーバのホスト名または IP アドレスを指定します。 複数の HTTP サーバを指定する場合、「 」（ストローク）」で区切って指定します。ワイルドカードも使用できます。 補足：項目名と動作が不一致ですが、「無視する HTTP サーバホスト名」と読み替えてお使いください。
12	[参照元ファイル名]	アップロードするファイルのファイルパスを指定します。 [参照] ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
13	[保存先ファイル名]	ダウンロードするファイルのファイルパスを指定します。 [参照] ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
14	[既存のファイルに追加する]	この項目にチェックを付けると、同じ名前のファイルが存在する場合、既存ファイルの末尾に新しいデータを追加します。
15	[ファイル名に日付と時刻を付ける]	この項目にチェックを付けると、ファイルを転送したときの日付と時刻（形式：yyyyMMdd_HHmmss）をファイル名の末尾に追加します。
16	[拡張子]	ファイルの拡張子を指定します。
17	[結果ファイルにファイル名を追加]	この項目にチェックを付けると、[保存先ファイル名]のファイル情報をメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納した[保存先ファイル名]のファイル情報を取得できます。 【注意事項】 HTTP エントリーをリポジトリに保存している場合、この項目にチェックを付けても、ファイル取得ステップでファイル情報を取得できません。

(2) [ヘッダー]タブ

HTTP ヘッダーを設定します。

[illegible]

項番	項目名	説明
1	[名前]	HTTP ヘッダーの名前を指定します。
2	[値]	HTTP ヘッダーの値を指定します。

ファイルにテキストデータを書き込みます。

図 3-19 [ファイルへ書き込み]画面

ファイルへ書き込み...

ジョブ名

ファイルへ書き込み

ファイル

ファイル名

参照(B)...

親フォルダの作成

ファイルの添付

内容

エンコーディング

テキスト

ヘルプ

OK(O)

キャンセル(C)

表 3-20 [ファイルへ書き込み]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ファイル名]	データを書き込むファイルのファイルパスを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[親フォルダーの作成]	この項目にチェックを付けると、[ファイル名]で指定したパスの中のディレクトリがない場合に、そのディレクトリを自動作成します。 チェックを付けない場合、[ファイル名]で指定したパスの中のディレクトリがないときは、エラーが発生します。
4	[ファイルの添付]	この項目にチェックを付けると、[ファイル名]で指定したファイルに、テキストデータを追記します。 チェックを付けない場合、[ファイル名]で指定したファイルに、テキストデータを上書きします。
5	[エンコーディング]	ファイルに書き込むテキストデータの文字コードを指定します。
6	[テキスト]	ファイルに書き込むテキストデータを指定します。

3.4.3 [ファイル管理]ファイルコピー（Copy Files）

指定したファイルを、指定した出力先にコピーします。

(1) [Files]タブ

コピー対象のファイルに関する設定をします。

図 3-20 [ファイルをコピー]画面-[Files]タブ

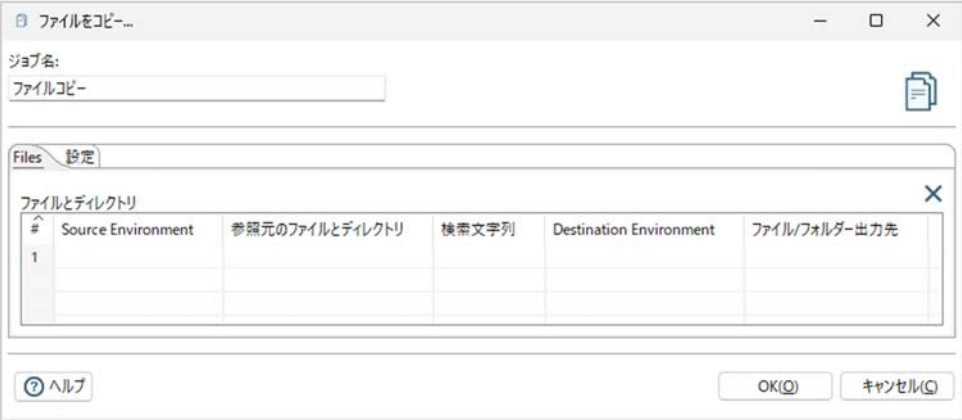


表 3-21 [ファイルをコピー]画面-[Files]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[Source Environment]	[参照元のファイルとディレクトリ]に指定するファイルまたはディレクトリの環境を、次の項目から指定します。 • [Local] [参照元のファイルとディレクトリ]にローカルのファイルシステム上のファイルまたはディレクトリを指定したい場合に、指定します。 • [<Static>] [参照元のファイルとディレクトリ]にパスをコピー＆ペーストしてファイルまたはディレクトリを直接指定したい場合に、指定します。
3	[参照元のファイルとディレクトリ]	コピーするファイルのファイルパス、またはコピーするファイルが格納されているディレクトリのパスを指定します。
4	[検索文字列]	[参照元のファイルとディレクトリ]にディレクトリのパスを指定した場合に、検索文字列と完全一致するファイルだけをコピーします。 ディレクトリ内の複数ファイルを指定したい場合は、正規表現で入力してください。
5	[Destination Environment]	[ファイル/フォルダー出力先]に指定するファイルまたはディレクトリの環境を、次の項目から指定します。 • [Local] [ファイル/フォルダー出力先]にローカルのファイルシステム上のファイルまたはディレクトリを指定したい場合に、指定します。 • [<Static>] [ファイル/フォルダー出力先]にパスをコピー＆ペーストしてファイルまたはディレクトリを直接指定したい場合に、使用します。
6	[ファイル/フォルダー出力先]	コピー先のファイルパス、またはコピー先のディレクトリのパスを指定します。

(2) [設定]タブ

ファイルをコピーする場合の条件を設定します。

図 3-21 [ファイルをコピー]画面-[設定]タブ

ファイルをコピー...

ジョブ名:
ファイルコピー

Files 設定

- ☐ サブ・ディレクトリを含む
- ☐ 移動先がファイル
- ☒ 空のディレクトリにコピー
- ☐ 移動先のディレクトリを作成する
- ☐ 既存のファイルを置き換え
- ☐ 移動後に参照元のファイルを削除
- ☐ 先行のジョブ・エントリから値を引き継ぐ
- ☐ ファイル名を結果に含む

ヘルプ OK(O) キャンセル(C)

表 3-22 [ファイルをコピー]画面-[設定]タブの項目

項番	項目名	説明
1	[サブ・ディレクトリを含む]	この項目にチェックを付けると、サブディレクトリ内のファイルも含め、サブディレクトリごとコピーします。
2	[移動先がファイル]	[Files]タブの[ファイル/フォルダー出力先]で指定したパスがファイルの場合、この項目にチェックを付けないとエラーが発生します。
3	[空のディレクトリにコピー]	この項目にチェックを付けると、ディレクトリが空の場合でも空のディレクトリがコピーされます。 この項目にチェックを付けるには、次の条件を満たす必要があります。 • [サブ・ディレクトリを含む]にチェックを付ける • [Files]タブの[検索文字列]を空白にする
4	[移動先のディレクトリを作成する]	この項目にチェックを付けると、[Files]タブの[ファイル/フォルダー出力先]で指定したディレクトリがコピー先に存在しない場合に、ディレクトリを自動作成します。
5	[既存のファイルを置き換え]	この項目にチェックを付けると、コピー先に同じ名前のファイルが存在する場合、既存ファイルを上書きします。
6	[移動後に参照元のファイルを削除]	この項目にチェックを付けると、ファイルをコピーした後、コピー元のファイルを削除します。
7	先行のジョブ・エントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリーの実行結果を引数に設定します。
8	[ファイル名を結果に含む]	この項目にチェックを付けると、コピー先ファイルのファイルパスをジョブの実行結果に追加します。

3.4.4 [ファイル管理]ファイル作成 (Create file)

空のファイルを作成します。

図 3-22 [ファイル作成]画面

ファイル作成

ジョブ名:

ファイル名:

参照(B)...

ファイルが存在する場合はエラーとする: ☒

ファイル名を結果に含む ☐

ヘルプ

OK(O)

キャンセル(C)

表 3-23 [ファイル作成]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ファイル名]	作成するファイルのファイルパスを指定します。ファイルには何も出力されません。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
3	[ファイルが存在する場合はエラーとする]	この項目にチェックを付けると、同じ名前のファイルが存在する場合に、エラーが発生します。
4	[ファイル名を結果に含む]	この項目にチェックを付けると、作成したファイルのファイルパスをジョブの実行結果に追加します。

3.4.5 [ファイル管理]ファイル削除 (Delete file)

指定した 1 ファイルだけを削除します。

図 3-23 [ファイル削除]画面

ファイル削除

ジョブ名:

ファイル名:

参照(B)...

ファイルが存在しない場合はエラーとする: ☐

ヘルプ

OK(O)

キャンセル(C)

表 3-24 [ファイル削除]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ファイル名]	削除するファイルのファイルパスを指定します。

項番	項目名	説明
		[参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
3	[ファイルが存在しない場合はエラーとする]	この項目にチェックを付けると、削除するファイルが存在しない場合に、エラーが発生します。

3.4.6 [ファイル管理]ファイル削除（複数）（Delete files）

指定したファイルをすべて削除します。ディレクトリを指定して、ディレクトリ内のすべてのファイルを削除することもできます。

図 3-24 [ファイル削除（複数）]画面

表 3-25 [ファイル削除（複数）]画面の項目

項番	項目名	説明
1	[ジョブエントリー名]	ジョブ名を指定します。
2	[サブディレクトリを含む]	この項目にチェックを付けると、サブディレクトリ内のファイルも削除します。
3	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリーの実行結果を引数に設定します。
4	[ファイル/ディレクトリ]	削除するファイルのファイルパス、または削除するファイルが格納されているディレクトリのパスを指定します。
5	[検索文字列]	[ファイル/ディレクトリ]にディレクトリのパスを指定した場合に、検索文字列と完全一致するファイルだけを削除します。 ディレクトリ内の複数ファイルを指定したい場合は、正規表現で入力してください。

3.4.7 [ファイル管理]ファイル圧縮 (Zip file)

参照元のファイルとディレクトリを圧縮します。

(1) [全般]タブ

参照ファイルと圧縮ファイルに関する設定をします。

[ファイル名の参照]ボタンを押すと、出力ファイルリストの一覧が表示されます。

図 3-25 [zip ファイルの作成]画面-[全般]タブ

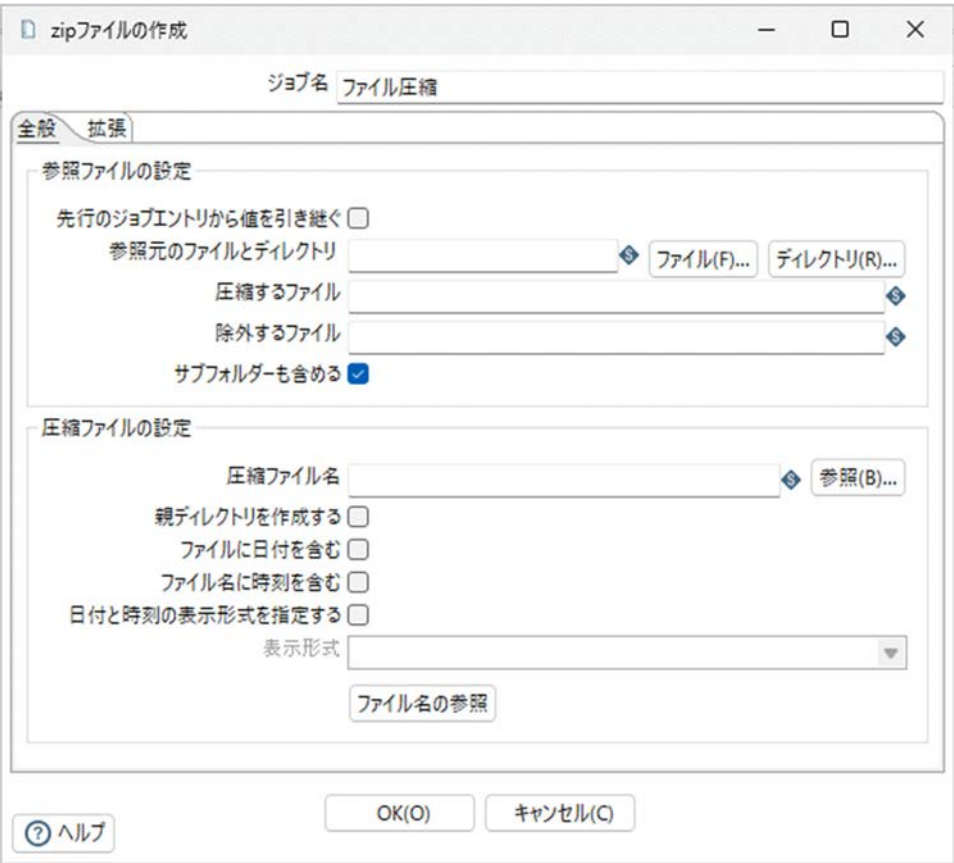


表 3-26 [zip ファイルの作成]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリーのパス情報をジョブ情報として使用できます。 先行ジョブエントリーとしては、例えば、データグリッドステップと行送信ステップを組み合わせたサブデータ変換が使用できます。この場合、データグリッドステップの1番目に[参照元のファイルとディレクトリ]、2番目に[圧縮するファイル]、3番目に[除外するファイル]、4番目に[圧縮ファイル名]を指定してください。 なお、この項目にチェックを付けた場合、[参照元のファイルとディレクトリ]を使用したデータ読み込みはできません。

項番	項目名	説明
3	[参照元のファイルとディレクトリ]	圧縮するファイルまたはディレクトリを指定します。 次の場合はエラーが発生します。 - 指定したファイルまたはディレクトリが存在しない - ファイルまたはディレクトリにアクセス権限がない
4	[圧縮するファイル]	指定した文字列で検索し、[圧縮するファイル]と完全一致するファイルだけを入力ファイルリストに追加します。 正規表現で入力する必要があります。 [参照元のファイルとディレクトリ]にディレクトリを指定した場合に指定できます。
5	[除外するファイル]	指定した文字列で検索し、[除外するファイル]と完全一致するファイルだけを入力ファイルリストから除外します。 正規表現で入力する必要があります。 [参照元のファイルとディレクトリ]にディレクトリを指定した場合に指定できます。 [圧縮するファイル]と同じファイルを指定した場合は、[除外するファイル]が優先されます。
6	[サブフォルダーも含める]	[参照元のファイルとディレクトリ]にディレクトリを指定した場合、この項目にチェックを付けると、指定したディレクトリのサブフォルダも圧縮されます。 【注意事項】 [参照元のファイルとディレクトリ]にディレクトリを指定してこの項目にチェックを付けて実行すると、圧縮の際はサブディレクトリを含めずにファイルが移動先へ移動します。例を次に示します。 【圧縮前】 圧縮対象のディレクトリとファイルの構成にサブディレクトリ src1 が存在します。 <pre>source2 -test01.txt -abc123.txt -src1 test02.txt</pre> 【圧縮後】 指定したパスに圧縮対象のディレクトリとファイルが移動しますが、サブディレクトリ src1 は存在しません。ディレクトリ構成は次のとおりとなります。 <pre>-test01.txt -test02.txt -abc123.txt</pre>
7	[圧縮ファイル名]	作成する圧縮ファイルの名称とパスを指定します。
8	[親ディレクトリを作成する]	この項目にチェックを付けると、[圧縮ファイル名]に指定したパスの中のディレクトリが存在しない場合でも、自動でディレクトリを作成します。 チェックを付けない場合、[圧縮ファイル名]に指定したパスの中のディレクトリが存在しないとエラーが発生します。
9	[ファイルに日付を含む]	この項目にチェックを付けると、ジョブエントリを実行したときの日付（形式：yyyyMMdd）を圧縮後のファイルのファイル名の末尾に追加します。

項番	項目名	説明
10	[ファイル名に時刻を含む]	この項目にチェックを付けると、ジョブエントリを実行したときの時刻（形式：HHmmssSSS）を圧縮後のファイルのファイル名の末尾に追加します。
11	[日付と時刻の表示形式を指定する]	この項目にチェックを付けると、ジョブエントリを実行したときの日付と時刻を[表示形式]で選択した形式で圧縮後のファイルのファイル名の末尾に追加します。 チェックを付けると、[ファイルに日付を含む]および[ファイル名に時刻を含む]の設定は無効になります。
12	[表示形式]	圧縮後のファイルのファイル名の末尾に追加する日付と時刻の形式を選択します。

(2) [拡張]タブ

拡張設定をします。

図 3-26 [zip ファイルの作成]画面-[拡張]タブ

表 3-27 [zip ファイルの作成]画面-[拡張]タブの項目

項番	項目名	説明
1	[圧縮タイプ]	圧縮タイプを次の項目から指定します。 [圧縮しない] zip 形式のファイルを作成しますが、圧縮は行いません。 [デフォルト圧縮]

項番	項目名	説明
		zip 形式のファイルを作成し、圧縮します。 • [ベストの圧縮率] zip 形式のファイルを作成し、デフォルト圧縮よりやや大きい圧縮率で圧縮します。 • [ベストの速度] zip 形式のファイルを作成し、デフォルト圧縮や、ベストの圧縮率と比べて小さい圧縮率で圧縮します。圧縮速度はデフォルト圧縮やベストの圧縮率より速くなります。
2	[ファイルが存在した場合の処理]	圧縮先のディレクトリに同じ名前のファイルが存在した場合の処理を次の項目から指定します。 • [ユニークな名前で新規ファイルを作成] 拡張子の前に実行時の日付と時刻「_yyyyddMM_HHmssSSS」が追加されます。 「_yyyyddMM_HHmssSSS」が付与されたファイルと同じ名前のファイルが存在する場合、圧縮ファイルが上書きされます。 • [既存のファイルに追加] 圧縮先のファイルを追加して圧縮します。 zip ファイルに同じ名前のファイルがある場合、エラーが発生し、zip ファイルの中身が空になります。 • [何もしない] ファイル圧縮処理を中止して、正常終了します。 • [失敗] ファイル圧縮処理を中止して、異常終了します。
3	[圧縮後の処理]	圧縮後に圧縮元のファイルに対する処理を次の項目から指定します。 • [何もしない] 何もしないで、正常終了します。 • [ファイルを削除] [全般]タブの[参照元のファイルとディレクトリ]にファイルを指定した場合、指定したファイルが圧縮後に削除されます。 [全般]タブの[参照元のファイルとディレクトリ]にディレクトリを指定して、さらに[サブフォルダーも含める]にチェックを付けた場合、指定したディレクトリの配下にあるファイルとサブフォルダの中にあるファイルが削除されます。ディレクトリは削除されません。 [全般]タブの[参照元のファイルとディレクトリ]にディレクトリを指定して、さらに[サブフォルダーも含める]にチェックを付けない場合、指定したディレクトリの配下にあるファイルだけ削除され、サブフォルダの中にあるファイルは削除されません。ディレクトリも削除されません。 • [ファイルを移動] [全般]タブの[参照元のファイルとディレクトリ]にファイルを指定した場合、指定したファイルが圧縮後に移動します。 [全般]タブの[参照元のファイルとディレクトリ]にディレクトリを指定して、さらに[サブフォルダーも含める]にチェックを付けた場合、指定したディレクトリの配下にあるファイルとサブフォルダの中にあるファイルが移動します。ディレクトリは移動しません。 [全般]タブの[参照元のファイルとディレクトリ]にディレクトリを指定して、さらに[サブフォルダーも含める]にチェックを付けない場合、指定したディレクトリの配下にあ

項番	項目名	説明
		るファイルだけ移動し、サブフォルダの中にあるファイルは移動しません。ディレクトリも移動しません。
4	[移動先のディレクトリ]	[圧縮後の処理]で[ファイルを移動]を指定した場合、移動先のディレクトリを指定します。次の場合はエラーが発生します。 - 指定したディレクトリが存在しない - 指定したディレクトリにアクセス権限がない
5	[ディレクトリを作成する]	[圧縮後の処理]で[ファイルを移動]を指定した場合、移動先のディレクトリが存在しないとき、この項目にチェックを付けると、ディレクトリを自動作成し、ファイルが移動します。[移動先のディレクトリ]に存在するディレクトリを指定し、この項目にチェックを付けた場合、指定したディレクトリにファイルが移動します。 ディレクトリを作成する権限がないなどで、ディレクトリの作成に失敗した場合はエラーが発生します。
6	[Stored source path depth]	[全般]タブの[先行のジョブエントリから値を引き継ぐ]にチェックを付けている場合、この機能を使用できます。 - 先行ジョブの第 1 引数にファイルを指定した場合 [Stored source path depth]に指定した数字やパスの数だけ、zip ファイル内に圧縮元のパスが引き継がれます。 例：Stored source path depth が 3 の場合 先行ジョブの第 1 引数： C:¥PentahoTest¥trans_P06_S94¥01_inputdata¥test¥test01.txt 先行ジョブの第 2 引数：null 先行ジョブの第 3 引数：null 先行ジョブの第 4 引数： C:¥PentahoTest¥trans_P06_S94¥11_outputdata¥out¥test1.zip test1.zip の内容は次のようになります。 test1.zip -01_inputdata -test -test01.txt - 先行ジョブの第 1 引数にディレクトリを指定した場合 zip ファイル内に圧縮元のパスは引き継がれません。 例：Stored source path depth が 3 の場合 先行ジョブの第 1 引数：C:¥PentahoTest¥trans_P06_S94¥01_inputdata¥test 先行ジョブの第 2 引数：null 先行ジョブの第 3 引数：null 先行ジョブの第 4 引数： C:¥PentahoTest¥trans_P06_S94¥11_outputdata¥out¥test1.zip test1.zip の内容は次のようになります。 test1.zip -test01.txt -test02.txt -test03.txt 9 以上の値を指定しても、動作します。

項番	項目名	説明
7	[ファイル名を結果に含む]	この項目にチェックを付けると、データ出力に用いたファイルパスをメモリに格納します。ファイル取得ステップを使用することで、メモリに格納したファイルパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。

3.4.8 [ファイル管理]ファイル待機 (Wait for a file)

設定した時間内に、指定したファイルが存在するかどうかを確認します。

図 3-27 [ファイル待機]画面

表 3-28 [ファイル待機]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ファイル名]	存在を確認するファイルのファイルパスを指定します。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
3	[タイムアウト]	ファイルの存在を確認するタイムアウト時間を、秒単位で指定します。 0 を指定すると、ファイルの存在が確認されるまで処理が継続されます。 【注意事項】 0 を指定したジョブを停止する場合は、ジョブを強制終了してください。
4	[更新間隔 (秒)]	ファイルの存在を確認する間隔を、秒単位で指定します。 ファイルの存在は、まず実行開始時に確認されます。その後、[タイムアウト]に設定した時間に達するまで、この項目で指定した時間ごとに確認されます。
5	[タイムアウトを成功とする]	この項目にチェックを付けると、[タイムアウト]に設定した時間に達した時点でファイルが存在しない場合でも、成功状態にします。

項番	項目名	説明
6	[ファイルサイズをチェックする]	この項目にチェックを付けると、ファイルが存在することが確認できたあと、[更新間隔（秒）]で指定した時間内にファイルサイズが更新されると、再度ファイルが存在するかどうかの確認処理を実行します。
7	[ファイル名を結果に含む]	この項目にチェックを付けると、確認したファイルのファイルパスをジョブの実行結果に追加します。

3.4.9 [ファイル管理]ファイル移動（Move files）

ファイルを移動します。

(1) [全般]タブ

移動対象のファイルに関する設定をします。

図 3-28 [ファイル移動]画面-[全般]タブ

ファイル移動

ジョブ名 ファイル移動

全般 移動先 拡張

設定

☐ サブディレクトリを含む
☐ 空のディレクトリを含む
☐ シミュレーション
☐ 先行のジョブエントリから値を引き継ぐ

参照元のファイルとディレクトリ 追加(A) ファイル(F)... ディレクトリ(R)...
 移動先のファイルとディレクトリ ファイル(F)... ディレクトリ(R)...
 検索文字列

ファイルとディレクトリ

#	参照元のファイルとディレクトリ	移動先のファイルとディレクトリ	検索文字列
1			

削除(D) 編集(E)

ヘルプ OK(O) キャンセル(C)

表 3-29 [ファイル移動]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[サブディレクトリを含む]	この項目にチェックを付けると、サブディレクトリのファイルも移動します。
3	[空のディレクトリを含む]	この項目にチェックを付けると、空のディレクトリも移動します。
4	[シミュレーション]	この項目にチェックを付けると、ファイルを移動できるかどうかを確認します。実際にはファイル移動は実行しません。
5	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリの実行結果を引数に設定します。
6	[参照元のファイルとディレクトリ]	移動するファイルのファイルパス、またはファイルが格納されているディレクトリのパスを指定します。 [追加]ボタンを押すと、ファイルとディレクトリのリストに登録されます。 [ファイル]ボタンまたは[ディレクトリ]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。 (ファイルパスを指定する場合は[ファイル]ボタン、ディレクトリパスを指定する場合は[ディレクトリ]ボタンを押してください。)
7	[移動先のファイルとディレクトリ]	移動先のディレクトリのパス、または移動先がファイルの場合はそのファイルのファイルパスを指定します。 移動先がファイルの場合、[移動先]タブの[ファイルへ移動する]にチェックを付けてください。 [ファイル]ボタンまたは[ディレクトリ]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。 (ファイルパスを指定する場合は[ファイル]ボタン、ディレクトリパスを指定する場合は[ディレクトリ]ボタンを押してください。)
8	[検索文字列]	検索文字列と完全一致するファイルだけを移動します。正規表現も入力できます。
9	[削除]ボタン	このボタンを押すと、選択対象をファイルとディレクトリのリストから削除します。
10	[編集]ボタン	このボタンを押すと、選択対象のファイルまたはディレクトリのパスを編集します。

(2) [移動先]タブ

ファイルの移動先に関する設定をします。

図 3-29 [ファイル移動]画面-[移動先]タブ

ファイル移動

ジョブ名ファイル移動

全般移動先拡張

ファイル移動の設定

ディレクトリを作成する

ファイルへ移動する

ディレクトリツリーを維持しない

ファイル名に日付を付ける

ファイル名に時刻を付ける

日付と時刻の表示形式を指定する

表示形式

日付と時刻を拡張子にする

ファイルが存在した場合の処理何もしない

ディレクトリ移動の設定

移動先のディレクトリ

参照(B)...

ディレクトリを作成する

ディレクトリ名に日付を含む

ディレクトリ名に時刻を含む

日付の表示形式を指定する

表示形式

日付をディレクトリ名に付ける

ディレクトリが存在した場合の処理何もしない。

ヘルプ

OK(O)

キャンセル(C)

表 3-30 [ファイル移動]画面-[移動先]タブの項目

項番	項目名	説明
1	[ディレクトリを作成する]	この項目にチェックを付けると、[全般]タブの[移動先のファイルとディレクトリ]で指定したディレクトリが移動先に存在しない場合にディレクトリを自動作成します。
2	[ファイルへ移動する]	[全般]タブの[移動先のファイルとディレクトリ]にファイルパスを指定する場合、この項目にチェックを付けてください。 補足：項目名と動作が不一致ですが、「移動先がファイルの場合」と読み替えてお使いください。
3	[ディレクトリツリーを維持しない]	この項目にチェックを付けると、移動先にサブディレクトリが存在しない場合に、移動元ファイルを移動先ディレクトリ直下に移動します。 この項目にチェックを付けたい場合は、[全般]タブの[サブディレクトリを含む]にチェックを付けてください。 例えば、移動元にサブディレクトリ yyy があり、移動先ディレクトリ zzz 内にサブディレクトリ yyy がない場合、次のようになります。 - 移動元ファイル /xxx/a.txt /xxx/yyy/b.txt - 移動先ディレクトリ /zzz - 実行後の移動先

項番	項目名	説明
		/zzz/a.txt /zzz/b.txt
4	[ファイル名に日付を付ける]	この項目にチェックを付けると、ジョブエントリを実行したときの日付（形式：yyyyMMdd）を移動後のファイルのファイル名の末尾に追加します。
5	[ファイル名に時刻を付ける]	この項目にチェックを付けると、ジョブエントリを実行したときの時刻（形式：HHmmssSSS）を移動後のファイルのファイル名の末尾に追加します。
6	[日付と時刻の表示形式を指定する]	この項目にチェックを付けると、ジョブエントリを実行したときの日付と時刻を[表示形式]で選択した形式で移動後のファイルのファイル名の末尾に追加します。
7	[表示形式]	移動後のファイルのファイル名の末尾に追加する日付と時刻の形式を選択します。
8	[日付と時刻を拡張子にする]	この項目にチェックを付けると、日付と時刻をファイル拡張子の前に追加します。 補足：項目名と動作が不一致ですが、「日付と時刻を拡張子の前に付ける」と読み替えてお使いください。 例えば、[ファイル名に日付を付ける]にチェックを付けた場合、ファイル名は次のようになります。 - 移動前：<ファイル名>.txt - 移動後：<ファイル名>_yyyyMMdd.txt
9	[ファイルが存在した場合の処理]	移動先のディレクトリに同じ名前のファイルが存在した場合の処理を次の項目から指定します。 [何もしない] ファイル移動処理を中止して、正常終了します。 [移動先のファイルを上書き] 移動先のファイルを上書きします。 [ユニーク名でファイルを作成] 移動先にユニークな名称のファイルとして作成、保存します。 [元ファイルを削除] 移動元のファイルを削除します。 [ソースファイルをフォルダに移動] [ディレクトリ移動の設定]領域の項目を設定できるようになります。 [失敗] ファイル移動処理を中止して、異常終了します。
10	[移動先のディレクトリ]	移動先ディレクトリを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
11	[ディレクトリを作成する]	この項目にチェックを付けると、移動先ディレクトリが移動先に存在しない場合にディレクトリを自動作成します。
12	[ディレクトリ名に日付を含む]	この項目にチェックを付けると、ジョブエントリを実行したときの日付（形式：yyyyMMdd）を移動後のファイルのファイル名の末尾に追加します。 補足：項目名と動作が不一致ですが、「ファイル名に日付を付ける」と読み替えてお使いください。

項番	項目名	説明
13	[ディレクトリ名に時刻を含む]	この項目にチェックを付けると、ジョブエントリを実行したときの時刻（形式：HHmmssSSS）を移動後のファイルのファイル名の末尾に追加します。 補足：項目名と動作が不一致ですが、「ファイル名に時刻を付ける」と読み替えてお使いください。
14	[日付の表示形式を指定する]	この項目にチェックを付けると、ジョブエントリを実行したときの日付と時刻を[表示形式]で選択した形式で移動後のファイルのファイル名の末尾に追加します。 補足：項目名と動作が不一致ですが、「日付と時刻の表示形式を指定する」と読み替えてお使いください。
15	[表示形式]	移動後のファイルのファイル名の末尾に追加する日付と時刻の形式を選択します。
16	[日付をディレクトリ名に付ける]	この項目にチェックを付けると、日付と時刻をファイル拡張子の前に追加します。 補足：項目名と動作が不一致ですが、「日付と時刻を拡張子の前に付ける」と読み替えてお使いください。 例えば、[ファイル名に日付を付ける]にチェックを付けた場合、ファイル名は次のようになります。 - 移動前：＜ファイル名＞.txt - 移動後：＜ファイル名＞_yyyyMMdd.txt
17	[ディレクトリが存在した場合の処理]	移動先のディレクトリに同じ名前のファイルが存在した場合の処理を次の項目から指定します。 補足：項目名と動作が不一致ですが、「ファイルが存在した場合の処理」と読み替えてお使いください。 [何もしない] ファイル移動処理を中止して、正常終了します。 [ファイルを上書き] 移動先のファイルを上書きします。 [ユニーク名] 移動先にユニークな名称のファイルとして作成、保存します。 [失敗] ファイル移動処理を中止して、異常終了します。

(3) [拡張]タブ

ファイル移動の処理条件に関する設定をします。

図 3-30 [ファイル移動]画面-[拡張]タブ

ファイル移動

ジョブ名ファイル移動

全般移動先拡張

処理条件の設定

条件全て正常に実行された場合に成功

ファイル制限10

オプション

ファイル名を結果に含む

ヘルプ

OK(O)

キャンセル(C)

表 3-31 [ファイル移動]画面-[拡張]タブの項目

項番	項目名	説明
1	[条件]	処理条件を次の項目から指定します。 [全て正常に実行された場合に成功] すべてのファイルを正常に移動できた場合、成功状態にします。 [指定数以上のファイルを移動した場合は成功] [ファイル制限]で指定した値以上のファイル数を移動した場合に、成功状態にします。 [エラーが指定回数より少ない場合は成功] [ファイル制限]で指定した値未満のファイル数を移動した場合に、成功状態にします。
2	[ファイル制限]	[条件]で選択した処理条件に合わせて、値を指定します。
3	[ファイル名を結果に含む]	この項目にチェックを付けると、移動したファイルのファイルパスをジョブの実行結果に追加します。

3.4.10 [ファイル管理]ファイル解凍 (Unzip file)

1 つまたは複数の zip ファイルを解凍します。

(1) [全般]タブ

解凍するファイルの情報を設定します。

図 3-31 [解凍ファイル]画面-[全般]タブ

解凍ファイル

ジョブ名 ファイル解凍

全般

拡張

参照元ファイル

先行のジョブ・エントリから値を引き継ぐ

圧縮ファイル名

ファイル(F)...

ディレクトリ(R)...

検索文字列

解凍先ファイル

圧縮ファイル名をルート・ディレクトリとする

解凍先のディレクトリ

ディレクトリ(R)...

ディレクトリを作成する

解凍するファイル

除外するファイル

ファイル名に日付を付ける

ファイル名に時刻を付ける

日付と時刻の表示形式を指定する

表示形式

オリジナルtimestampを追加

修正日をオリジナルに設定

ファイルが存在していた場合の処理

スキップ

解凍後の処理

何もしない

移動先のディレクトリ

ディレクトリ(R)...

ディレクトリを作成する

ヘルプ

OK(O)

キャンセル(C)

表 3-32 [解凍ファイル]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリーの情報をジョブの引数として使用できます。 先行ジョブエントリーとしては、例えば、データグリッドステップと行送信ステップを組み合わせたサブデータ変換が使用できます。この場合、データグリッドステップの1番目に[圧縮ファイル名]、2番目に[検索文字列]を指定してください。 なお、この項目にチェックを付けた場合、[圧縮ファイル名]、および[検索文字列]は非活性になります。
3	[圧縮ファイル名]	解凍する zip ファイル名またはディレクトリ名を指定します。 次の場合はエラーが発生します。 指定したファイル名またはディレクトリが存在しない

項番	項目名	説明
		指定したファイルの親ディレクトリ、または指定したディレクトリにアクセス権限がない 【注意事項】 圧縮ファイルの中にマルチバイトを含む名前のファイルが含まれている場合、ジョブを実行するとファイルが解凍できず、エラーが発生します。
4	[検索文字列]	[圧縮ファイル名]にディレクトリを指定した場合、解凍する zip ファイル名の検索文字列を正規表現で指定します。
5	[圧縮ファイル名をルート・ディレクトリとする]	この項目にチェックを付けると、[解凍先のディレクトリ]に指定した場所に、圧縮ファイル名と同じ名前のディレクトリが作成され、その配下にファイルが解凍されます。 ディレクトリ名に圧縮ファイルの拡張子は含まれません。 [圧縮ファイル名]にディレクトリを指定した場合は、そのディレクトリ配下にあるすべての圧縮ファイル（[検索文字列]を指定した場合は一致したすべての圧縮ファイル）ごとに圧縮ファイルと同じ名称のディレクトリが作成され、その配下にそれぞれファイルが解凍されます。
6	[解凍先のディレクトリ]	解凍先のディレクトリを指定します。 次の場合はエラーが発生します。 - 指定したディレクトリが存在しない - 指定したディレクトリにアクセス権限がない
7	[ディレクトリを作成する]	この項目にチェックを付けると、[解凍先のディレクトリ]で指定したディレクトリが存在しない場合にディレクトリを自動作成します。 [解凍先のディレクトリ]で指定したディレクトリが存在する場合でも、エラーにはなりません。 ディレクトリを作成する権限がないなどで、ディレクトリの作成に失敗した場合はエラーとなります。
8	[解凍するファイル]	zip ファイルに格納されている、解凍の対象とするファイル名の検索文字列を正規表現で指定します。
9	[除外するファイル]	zip ファイルに格納されている、解凍の対象外とするファイル名の検索文字列を正規表現で指定します。 [解凍するファイル]と[除外するファイル]に同じファイル名を指定した場合は、[除外するファイル]が優先されます。
10	[ファイル名に日付を付ける]	この項目にチェックを付けると、システム日付（形式：yyyyMMdd）を解凍後のファイルのファイル名の末尾に追加します。 複数のファイルが解凍された場合は、すべてのファイルのファイル名の末尾にシステム日付が追加されます。
11	[ファイル名に時刻を付ける]	この項目にチェックを付けると、システム時刻（形式：HHmmssSSS）を解凍後のファイルのファイル名の末尾に追加します。 複数のファイルが解凍された場合は、すべてのファイルのファイル名の末尾にシステム時刻が追加されます。
12	[日付と時刻の表示形式を指定する]	この項目にチェックを付けると、システム日付とシステム時刻を[表示形式]で選択した形式で解凍後のファイルのファイル名の末尾に追加します。 デフォルトの表示形式は「yyyyMMdd'_'HHmmssSSS」です。

項番	項目名	説明
13	[表示形式]	日付表示形式をプルダウンから指定します。
14	[オリジナル timestamp を追加]	[ファイル名に日付を付ける]、[ファイル名に時刻を付ける]、[日付と時刻の表示形式を指定する]のどれかにチェックを付けた場合、この項目にチェックを付けると、元のファイルの最終更新日時をファイル名に追加します。 [オリジナル timestamp を追加]の形式は[表示形式]で指定した形式となります。
15	[修正日をオリジナルに指定]	この項目にチェックを付けると、解凍されたファイルの更新日時が元のファイルと同じになります。
16	[ファイルが存在していた場合の処理]	解凍先に同じ名前のファイルが存在した場合の処理を次の項目から指定します。 • [スキップ] ファイル解凍処理を行いません。 • [上書き] ファイルを上書きします。 • [ユニークなファイル名] 別の名前を付けて解凍します。 【注意事項】 この項目を指定してもユニークなファイル名で解凍されません。解凍先に同じ名前のファイルがある場合、上書きされます。 • [失敗] 異常終了します。 • [Overwrite if size different] 解凍後のファイルサイズが、解凍先に存在するファイルのサイズと異なる場合は上書きします。 • [Overwrite if size equal] 解凍後のファイルサイズが、解凍先に存在するファイルのサイズと同じ場合は上書きします。 • [圧縮ファイルサイズが大きい場合の上書き] 解凍後のファイルサイズが、解凍先に存在するファイルのサイズよりも大きい場合は上書きします。 • [圧縮ファイルサイズが大きい場合と同サイズの場合の上書き] 解凍後のファイルサイズが、解凍先に存在するファイルのサイズよりも大きい、または同じ場合は上書きします。 • [圧縮ファイルサイズが小さい場合の上書き] 解凍後のファイルサイズが、解凍先に存在するファイルのサイズよりも小さい場合は上書きします。 • [圧縮ファイルサイズが小さい場合と同サイズの場合の上書き] 解凍後のファイルサイズが、解凍先に存在するファイルのサイズよりも小さい、または同じ場合は上書きします。
17	[解凍後の処理]	[圧縮ファイル名]で指定した zip ファイルの解凍後の処理を次の項目から指定します。 • [何もしない] 何もしないで、正常終了します。 • [ファイル削除] [圧縮ファイル名]で指定した zip ファイルを削除します。

項番	項目名	説明
		• [ファイル移動] [圧縮ファイル名]で指定した zip ファイルを移動します。
18	[移動先のディレクトリ]	[解凍後の処理]で[ファイル移動]を指定した場合、zip ファイルの移動先のディレクトリを指定します。 次の場合はエラーが発生します。 • 指定したディレクトリが存在しない • 指定したディレクトリにアクセス権限がない
19	[ディレクトリを作成する]	[解凍後の処理]で[ファイル移動]を指定した場合、この項目にチェックを付けると、zip ファイルの移動先のディレクトリを作成します。 [移動先のディレクトリ]で指定したディレクトリが存在する場合はエラーは発生しません。 ディレクトリを作成する権限がないなどで、ディレクトリの作成に失敗した場合はエラーが発生します。

(2) [拡張]タブ

拡張設定をします。

図 3-32 [解凍ファイル]画面-[拡張]タブ

解凍ファイル

ジョブ名

ファイル解凍

全般

拡張

オプション

ファイル名を結果に含む ☐

処理条件の設定

条件

全て正常に実行された場合に成功

ファイル制限

10

ヘルプ

OK(O)

キャンセル(C)

表 3-33 [解凍ファイル]画面-[拡張]タブの項目

項番	項目名	説明
1	[ファイル名を結果に含む]	この項目にチェックを付けると、データ入力に使用したファイルパスをメモリに格納します。 ファイル取得ステップを使用すると、メモリに格納したファイルパスを取得できます。チェックを付けない場合、メモリ使用量を削減できます。 取得できるデータは、次のとおりです。 • type • filename 先行ジョブのファイル名 • path 先行ジョブのファイルパス • parentorigin 先行ジョブのジョブ名 • origin [ファイル名を結果に含む]を[Y]にしたステップ名 • comment

項番	項目名	説明
		• timestamp
2	[条件]	処理条件を次の項目から指定します。 • [全て正常に実行された場合に成功] すべてのファイルを正常に移動できた場合、成功状態にします。 • [指定数以上のファイルを解凍した場合は成功] [ファイル制限]で指定した値以上のファイル数を解凍した場合に、成功状態にします。 • [エラーが指定数以下の場合は成功] [ファイル制限]で指定した値以下のエラー数の場合は、成功状態にします。
3	[ファイル制限]	[条件]で選択した処理条件に合わせて、値を指定します。[条件]で[指定数以上のファイルを解凍した場合は成功]または[エラーが指定数以下の場合は成功]のどちらかを選択した場合に、指定できます。

3.4.11 [ファイル管理]フォルダー作成 (Create a folder)

指定したフォルダを作成します。

図 3-33 [フォルダー作成]画面

表 3-34 [フォルダー作成]画面の項目

項番	項目名	説明
1	[ジョブエントリー名]	ジョブ名を指定します。
2	[ディレクトリ名]	作成するディレクトリのパスを指定します。 [参照]ボタンを押すと、PDI クライアント (Spoon) を起動しているマシンのファイルシステムを参照できます。
3	[ディレクトリが存在する場合はエラーとする]	この項目にチェックを付けると、同じ名前のファイルが存在する場合に、エラーが発生します。

3.4.12 [ファイル管理]フォルダー削除 (Delete folders)

指定したフォルダを削除します。指定したフォルダ内にファイルがある場合は、フォルダ内のファイルもすべて削除します。

図 3-34 [フォルダー削除]画面

フォルダー削除

ジョブ名: フォルダー削除

設定

先行のジョブエントリから値を引き継ぐ

処理条件の設定

条件 エラーなし

フォルダー数/エラー数 10

ディレクトリ

追加(A) ディレクトリ(R)...

ディレクトリ:

#	ディレクトリ
1	

削除(D) 編集(E)

ヘルプ OK(Q) キャンセル(Q)

表 3-35 [フォルダー削除]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[先行のジョブエントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリーの実行結果を引数に設定します。
3	[条件]	削除処理が成功と見なされる条件を次の項目から指定します。 [エラーなし] エラーが発生しなかった場合に削除処理を終了し、成功状態にします。 エラーが発生した場合はその時点で削除処理を終了します（成功状態にはなりません）。 [x フォルダー以上処理された場合] [フォルダー数/エラー数]で指定した値以上のフォルダが処理された場合に削除処理を終了し、成功状態にします。 指定した値未満のフォルダが処理された場合はエラーになります（成功状態にはなりません）。 [エラーが x 未満の場合] [フォルダー数/エラー数]で指定した値未満の件数のエラーが発生した場合に削除処理を終了し、成功状態にします。 指定した値と同じ件数以上のエラーが発生した場合は削除処理を終了します（成功状態にはなりません）。
4	[フォルダー数/エラー数]	[条件]で[x フォルダー以上処理された場合]を選択した場合のフォルダ数、または[エラーが x 未満の場合]を選択した場合のエラー件数を指定します。
5	[ディレクトリ]	削除するディレクトリのパスを指定します。 [追加]ボタンを押すと、ディレクトリのリストに登録されます。 [ディレクトリ]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
6	[削除]ボタン	このボタンを押すと、選択対象をディレクトリのリストから削除します。

項番	項目名	説明
7	[編集]ボタン	このボタンを押すと、選択対象のディレクトリのパスを編集します。

3.5 条件

- [条件]テーブル検証 (Evaluate rows number in a table)
- [条件]テーブル確認 (Table exists)
- [条件]データベース接続確認 (Check Db Connections)
- [条件]ファイル確認 (File Exists)
- [条件]ファイル確認 (複数) (Checks if files exists)
- [条件]待機 (Wait for)
- [条件]簡易検証 (Simple evaluation)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.5.1 [条件]テーブル検証 (Evaluate rows number in a table)

テーブルの内容を評価します。

図 3-35 [テーブル検証]画面

テーブル検証

ジョブエントリー名

テーブル検証

データソース名

編集(E)

新規作成(N)...

ウィザード

ターゲットスキーマ

テーブル名:

参照(B)...

処理条件

条件式

大きい

最大レコード数

0

カスタムSQL

カスタムSQL

☐

パラメータを設定する

☐

実行する前に前回の結果を消去する

☒

実行した結果を追加する

☐

SQLステートメント:

SQLの選択(S)

ヘルプ

OK(O)

キャンセル(C)

表 3-36 [テーブル検証]画面の項目

項番	項目名	説明
1	[ジョブエントリー名]	ジョブ名を指定します。
2	[データソース名]	検証するテーブルのデータソースを選択します。

項番	項目名	説明
		[編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[ターゲットスキーマ名]	スキーマ名を指定します。 [カスタム SQL]にチェックを付けていない場合に、指定できます。
4	[テーブル名]	検証するテーブルの名前を指定します。 [カスタム SQL]にチェックを付けていない場合に、指定できます。 [参照]ボタンを押すと、データベース内のテーブル名を参照できます。
5	[条件式]	[最大レコード数]で設定した値と比較する条件式を、次の項目から指定します。 • [等しい] • [異なる] • [小さい] • [小さい/等しい] • [大きい] • [大きい/等しい]
6	[最大レコード数]	条件式を評価するための最大レコード数を指定します。
7	[カスタム SQL]	この項目にチェックを付けると、検証するテーブルを SQL 文で指定できます。
8	[パラメータを設定する]	この項目にチェックを付けると、ジョブ実行時に指定したパラメータを[SQL ステートメント]で使用できます。 [カスタム SQL]にチェックを付けると指定できます。
9	[実行する前に前回の結果を 消去する]	この項目にチェックを付けると、SQL 文を実行する前に、結果行の内部リストを消去します。 [カスタム SQL]にチェックを付けると指定できます。
10	[実行した結果を追加する]	SQL 文を実行して返されたすべての行を、結果行の内部リストに追加します。内部リストに追加された行は、後続のジョブエントリに引き継ぎます。 [カスタム SQL]にチェックを付けると指定できます。
11	[SQL ステートメント]	検証するテーブルを特定するための SQL 文を入力します。 ジョブ実行時に指定したパラメータを使用する場合は、「\${パラメータ}」と記述します。 [カスタム SQL]にチェックを付けると指定できます。 [SQL の選択]ボタンを押すと、データベースエクスプローラーが表示されます。テーブルを選択することで SQL 文を自動生成できます。

3.5.2 [条件]テーブル確認 (Table exists)

指定したテーブルがデータベースに存在するかどうかを確認します。

図 3-36 [テーブル確認]画面

テーブル確認

ジョブ名

テーブル確認

データソース名

編集(E)

新規作成(N)...

ウィザード

スキーマ名

参照(B)...

テーブル名

参照(B)...

ヘルプ

OK(O)

キャンセル(C)

表 3-37 [テーブル確認]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[データソース名]	存在を確認するテーブルのデータソースを選択します。 [編集]ボタン 作成済みのデータソース定義を編集します。 [新規作成]ボタン [データベース接続]画面で新規にデータソースを定義します。 [ウィザード]ボタン [データベース接続ウィザード]画面で新規にデータソースを定義します。
3	[スキーマ名]	スキーマ名を指定します。 [参照]ボタンを押すと、データベース内のスキーマ名を参照できます。
4	[テーブル名]	存在を確認するテーブルの名前を指定します。 [参照]ボタンを押すと、データベース内のテーブル名を参照できます。

3.5.3 [条件]データベース接続確認 (Check Db Connections)

データベースへの接続を確認します。

図 3-37 [データベース接続確認]画面

データベース接続確認

ジョブ名

データベース接続確認

接続

#	接続	待機	時間ユニット
1			

接続を取得

削除

ヘルプ

OK(O)

キャンセル(C)

表 3-38 [データベース接続確認]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[接続]	[データベース接続]画面で定義したデータベースを設定します。 [接続を取得]ボタンを押すと、[データベース接続]画面で定義したすべてのデータベースへの接続情報を取得します。 [削除]ボタンを押すと、選択対象を接続のリストから削除します。
3	[待機]	データベース接続開始後の待機時間を指定します。
4	[時間ユニット]	[待機]で指定した待機時間の単位を、次の項目から指定します。 • [Milliseconds]：ミリ秒 • [Seconds]：秒 • [Minutes]：分 • [Hours]：時間

3.5.4 [条件]ファイル確認 (File Exists)

指定した 1 ファイルが存在するかどうかを確認してから、後続のジョブエントリーに遷移します。

- 確認結果が True：指定したファイルが存在する
- 確認結果が False：指定したファイルが存在しない

図 3-38 [ファイル確認]画面

ファイル確認

ジョブ名

ファイル確認

ファイル名

参照(B)...

ヘルプ

OK(O)

キャンセル(C)

表 3-39 [ファイル確認]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ファイル名]	存在を確認する対象のファイルのファイルパスを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。

3.5.5 [条件]ファイル確認（複数）（Checks if files exists）

指定した複数のファイルまたはディレクトリが存在するかどうかを確認してから、後続のジョブエントリーに遷移します。

- 確認結果が True：指定したファイルまたはディレクトリがすべて存在する
- 確認結果が False：指定したファイルまたはディレクトリのうち、1 つ以上のファイルまたはディレクトリが存在しない

図 3-39 [ファイル確認（複数）]画面

表 3-40 [ファイル確認（複数）]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ファイル/ディレクトリ]	存在を確認する対象のファイルのファイルパスまたはディレクトリのパスを指定します。 [追加]ボタンを押すと、ファイルとディレクトリのリストに登録されます。 [ファイル]ボタンまたは[ディレクトリ]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 （ファイルパスを指定する場合は[ファイル]ボタン、ディレクトリパスを指定する場合は[ディレクトリ]ボタンを押してください。）
3	[削除]ボタン	このボタンを押すと、選択対象をファイル/ディレクトリのリストから削除します。
4	[編集]ボタン	このボタンを押すと、選択対象のファイルまたはディレクトリのパスを編集します。

3.5.6 [条件]待機 (Wait for)

次のジョブエントリーの実行を待機します。

図 3-40 [待機]画面

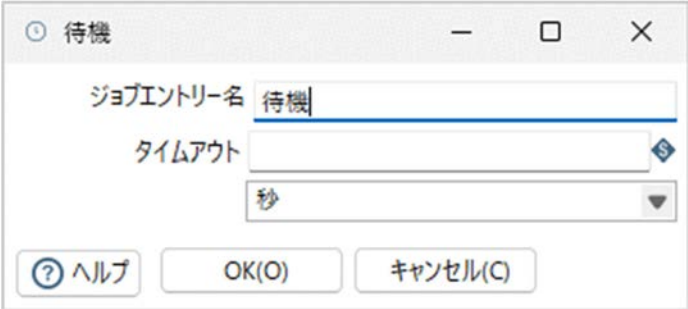


表 3-41 [待機]画面の項目

項番	項目名	説明
1	[ジョブエントリー名]	ジョブ名を指定します。
2	[タイムアウト]	ジョブエントリーの実行を待機するタイムアウト時間を指定します。
3	(単位)	[タイムアウト]で指定した時間の単位を次の項目から指定します。 • [秒] • [分] • [時間]

3.5.7 [条件]簡易検証 (Simple evaluation)

行送信したフィールドや変数の値を検証します。検証方法は、条件式で指定します。

(1) [全般]タブ

検証する対象や条件を設定します。

図 3-41 [簡易評価]画面-[全般]タブ

表 3-42 [簡易評価]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[条件式]	次のどちらかの条件式を指定します。 • [前の結果からのフィールド] 行送信したフィールドを検証します。 • [値] 変数の値を検証します。
3	[フィールド名]	この項目は、[条件式]で[前の結果からのフィールド]を指定した場合に表示されます。 検証する対象のフィールド名を指定します。
4	[変数値]	この項目は、[条件式]で[値]を指定した場合に表示されます。 検証する対象の変数名を指定します。
5	[データ・タイプ]	データタイプを次の項目から指定します。 • [文字列] • [数値型] • [日付時間] • [ブール型]
6	[日付表示形式]	この項目は、[データ・タイプ]で[日付時間]を指定した場合に表示されます。 日付と時刻の形式を選択します。

項番	項目名	説明
7	[値が設定されている場合は成功]	この項目は、[条件式]で[値]を指定した場合に表示されます。 この項目にチェックを付けると、変数が設定されている場合に正常終了します。
8	[条件式]	検証する条件を次の項目から指定します。 [データ・タイプ]に[文字列]を指定した場合 • [値が表記と等しい場合] 検証対象の値が、指定した[値]と等しい場合に正常終了します。 • [値が表記と異なる場合] 検証対象の値が、指定した[値]と異なる場合に正常終了します。 • [値が表記を含む場合] 検証対象の値が、指定した[値]を含む場合に正常終了します。 • [値が表記を含まない場合] 検証対象の値が、指定した[値]を含まない場合に正常終了します。 • [値が表記から開始の場合] 検証対象の値が、指定した[値]から開始する場合に正常終了します。 • [値が表記から開始しない場合] 検証対象の値が、指定した[値]から開始しない場合に正常終了します。 • [値が表記から開始する場合] 検証対象の値が、指定した[値]で終了する場合に正常終了します。 補足：項目名と動作が不一致ですが、「値が表記で終了する場合」と読み替えてお使いください。 • [値が表記で終了していない場合] 検証対象の値が、指定した[値]で終了しない場合に正常終了します。 • [値が正規表現と合致する場合] 検証対象の値が、指定した[値]の正規表現と一致する場合に正常終了します。 • [if value in list] 検証対象の値が、指定した[値]のリストデータに存在する場合に正常終了します。 • [if value not in list] 検証対象の値が、指定した[値]のリストデータに存在しない場合に正常終了します。 [データ・タイプ]に[数値型]または[日付時間]を指定した場合 • [値が表記と等しい場合] 検証対象の値が、指定した[値]と等しい場合に正常終了します。 • [値が表記と異なる場合] 検証対象の値が、指定した[値]と異なる場合に正常終了します。 • [値が小さい場合] 検証対象の値が、指定した[値]より小さい場合に正常終了します。 • [値が表記より小さいか等しい場合] 検証対象の値が、指定した[値]以下の場合に正常終了します。 • [値が表記より大きい場合] 検証対象の値が、指定した[値]より大きい場合に正常終了します。 • [もし値が大きい/等しい場合]

項番	項目名	説明
		検証対象の値が、指定した[値]以上の場合に正常終了します。 • [値が表記値の間に有る場合] 検証対象の値が、指定した[値]の最小値と最大値の範囲内にある場合に正常終了します。 • [if value in list] 検証対象の値が、指定した[値]のリストデータに存在する場合に正常終了します。 • [if value not in list] 検証対象の値が、指定した[値]のリストデータに存在しない場合に正常終了します。 [データ・タイプ]に[ブール型]を指定した場合 • [値が TRUE の場合] 検証対象の値が TRUE の場合に正常終了します。 • [値が FALSE] 検証対象の値が FALSE の場合に正常終了します。
9	[値]	条件式を評価するための値を指定します。
10	[最小値]	この項目は、[条件式]で[値が表記値の間に有る場合]を指定した場合に表示されます。値の最小値を指定します。
11	[最大値]	この項目は、[条件式]で[値が表記値の間に有る場合]を指定した場合に表示されます。値の最大値を指定します。
12	[条件]	この項目は、[データ・タイプ]で[ブール型]を指定した場合に表示されます。次のどちらかを指定します。 • [値が TRUE の場合] • [値が FALSE]

3.6 スクリプト

- [スクリプト]JavaScript (JavaScript)
- [スクリプト]SQL (SQL)
- [スクリプト]シェル (Shell)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.6.1 [スクリプト]JavaScript (JavaScript)

真偽値を計算する式を入力し、式の結果で次に実行するエントリを決定します。

図 3-42 [JavaScript]画面

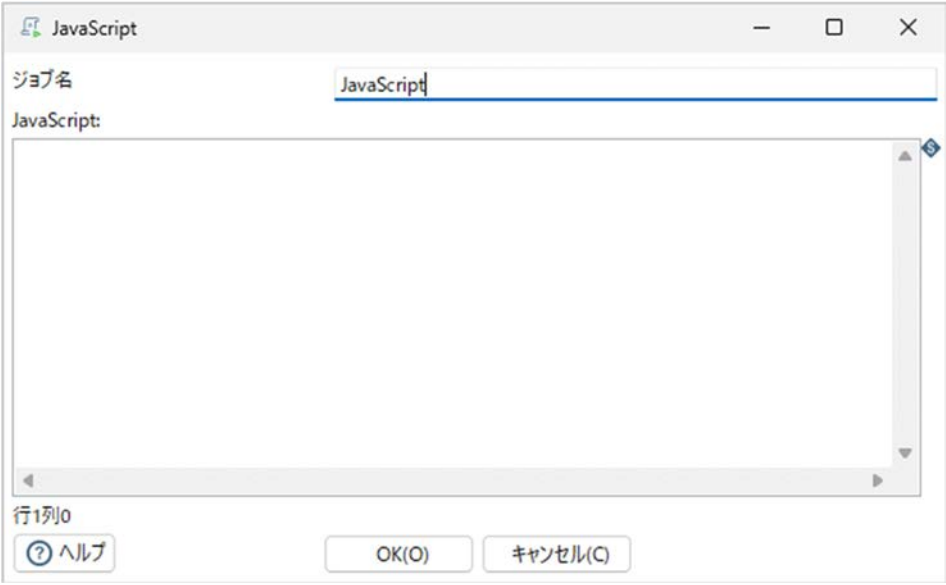


表 3-43 [JavaScript]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[JavaScript]	JavaScript を記載します。 スクリプトは Boolean 型を返すよう実装します。 使用できる変数および関数については、「表 3-44 JavaScript エントリー変数・関数一覧」を参照してください。

表 3-44 JavaScript エントリー変数・関数一覧

項番	カテゴリ	変数・関数名	型	変数・関数説明
1	変数	errors	Long	前のジョブエントリーのエラー数

項番	カテゴリ	変数・関数名	型	変数・関数説明
2		lines_input	Long	データベースまたはファイルから読み取られた行数
3		lines_output	Long	データベースまたはファイルに書き込まれた行数
4		lines_updated	Long	データベーステーブルで更新された行数
5		lines_read	Long	前の変換ステップで読み取られた行数
6		lines_written	Long	前の変換ステップで書き込まれた行数
7		files_retrieved	Long	FTP サーバから取得したファイルの数
8		exit_status	int	シェルスクリプト（整数）の終了ステータス
9		nr	Long	直前のジョブエントリのジョブエントリ番号
10		is_windows	Boolean	ジョブが Windows で実行されたかどうか
11		parent_job	(object)	現在のジョブエントリの親ジョブ
12		_entry_	(object)	現在のジョブエントリ
13	関数 (公開されている previous_result tを使用した関 数)	previous_result.getResult()	Boolean	前のジョブエントリの実行結果を取得します。
14		previous_result.getExitSta tus()	int	前のシェルスクリプトジョブエントリの終了ステータスを取得します。 取得値は、変数「exit_status」と同じです。
15		previous_result.getEntry Nr()	int	前のジョブエントリのジョブエントリ番号を取得します。 取得値は、変数「nr」と同じです。
16		previous_result.getNrErro rs()	Long	前のジョブエントリのエラー数を取得します。 取得値は、変数「errors」と同じです。
17		previous_result.getNrLine sInput()	Long	ファイルまたはデータベースから読み取られた行数を取得します。 取得値は、変数「lines_input」と同じです。
18		previous_result.getNrLine sOutput()	Long	ファイルまたはデータベースに書き込まれた行数を取得します。 取得値は、変数「lines_output」と同じです。
19		previous_result.getNrLine sRead()	Long	前のステップから読み取られた行数を取得します。 取得値は、変数「lines_read」と同じです。
20		previous_result.getNrLine sUpdated()	Long	ファイルまたはデータベースで更新された行数を取得します。 取得値は、変数「lines_updated」と同じです。
21		previous_result.getNrLine sWritten()	Long	前の変換ステップで書き込まれた行数を取得します。 取得値は、変数「lines_written」と同じです。

項番	カテゴリ	変数・関数名	型	変数・関数説明
22		previous_result.getNrLinesDeleted()	Long	前の変換ステップで削除された行数を取得します。
23		previous_result.getNrLinesRejected()	Long	エラー処理を介して別のステップに渡された行数を取得します。
24		previous_result.getRows()	List<RowMetaAndData>	前のステップからの結果行を取得します。 取得リストの例 [foo, baa]
25		previous_result.isStopped()	Boolean	前の前のジョブエントリが停止したかどうかを取得します。
26		previous_result.getResultFilesList()	List<ResultFile>	前のジョブエントリ、または複数のエントリで使用されたすべてのファイルのリストを取得します。 取得リストの例 [file:///C:/PentahoTest/Test/SampleOut.txt - 一般 - 2021/04/06 11:04:02.058 - テキストファイル出力 - JavaScript_lineinput_sub, file:///C:/PentahoTest/Test/sample_text.txt - 一般 - 2021/04/06 11:04:02.197 - テキストファイル入力.0 - JavaScript_lineinput_sub]
27		previous_result.getNrFilesRetrieved()	int	FTP、SFTP などから取得したファイルの数を取得します。 取得値は、変数「files_retrieved」と同じです。
28		previous_result.getLogText()	String	前のジョブエントリとその子ジョブ、子データ変換の実行のログテキストを取得します。
29		previous_result.getLogChannelId()	String	前のジョブエントリのログチャンネルの ID を取得します。

3.6.2 [スクリプト]SQL (SQL)

[SQL ステートメント]に指定した SQL 文、または SQL ファイルに記載された SQL 文を実行します。

図 3-43 [SQL]画面

SQL

ジョブ名

SQL

データソース名

編集(E)

新規作成(N)...

ウィザード

ファイルを入力する

SQLファイル名

参照(B)...

単体のステートメントとしてSQLを発行

変数の値を置き換え

SQLステートメント

行1列0

ヘルプ

OK(O)

キャンセル(C)

表 3-45 [SQL]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[データソース名]	データソースを選択します。
3	[ファイルを入力する]	この項目にチェックを付けると、[SQL ファイル名]が指定できます。 この項目にチェックを付けない場合、[SQL ステートメント]に SQL 文を定義してください。
4	[SQL ファイル名]	ファイルから SQL 文を実行する場合、SQL ファイルを指定します。
5	[単体のステートメントとして SQL を発行]	この項目にチェックを付けると、複数のステートメントを1つのステートメントとして扱います。 途中でエラーが発生した場合、実行済みのステートメントもすべてロールバックします。
6	[変数の値を置き換え]	この項目にチェックを付けると、変数参照は SQL を実行する前に値に展開されます。SQL 文に変数参照が含まれている場合は、この項にチェックを付けてください。
7	[SQL ステートメント]	実行する SQL 文を定義します。 複数のステートメントを実行したい場合は、セミコロンで区切ることで定義できます。 [ファイルを入力する]にチェックを付けている場合、この項目は非活性になります。

3.6.3 [スクリプト]シェル (Shell)

定義したシェルスクリプトを、ジョブが実行されているホストで実行します。

先行ジョブのデータ変換から、行送信ステップが送信したデータごとにシェルスクリプトを実行できます。

(1) [全般]タブ

実行するシェルスクリプトに関する設定をします。

図 3-44 [シェルスクリプトを実行]画面-[全般]タブ

シェルスクリプトを実行

ジョブ名 シェル

全般

スクリプト

スクリプトを挿入

スクリプトファイル名

作業ディレクトリ

参照(B)...

ログ設定

ログファイルを設定する

既存のファイルに追加する

ログファイル名

拡張子

ファイル名に日付を付ける

ファイル名に時刻を付ける

ログレベル

先行のジョブ・エントリから値を引き継ぐ

ループ処理を行う

フィールド

#

引数名

1

ヘルプ

OK(O)

キャンセル(C)

表 3-46 [シェルスクリプトを実行]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[スクリプトを挿入]	この項目にチェックを付けると、[スクリプト]タブの定義領域にシェルスクリプトを記述できます。
3	[スクリプトファイル名]	実行するシェルスクリプトのファイルパスを指定します。 ファイル名だけを指定した場合は、システムプロパティ「\${user.dir}」のパスでファイルを探します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
4	[作業ディレクトリ]	実行するシェルスクリプトの作業ディレクトリのパスを指定します。 何も指定しない場合、または指定した作業ディレクトリが使用できない場合は、システムプロパティ「\${user.dir}」を作業ディレクトリのパスとして使用します。

項番	項目名	説明
5	[ログファイルを設定する]	この項目にチェックを付けると、シェルスクリプト実行時にログファイルが生成されます。ログファイルの名称や拡張子などの詳細も設定できるようになります。
6	[既存のファイルに追加する]	この項目にチェックを付けると、既存にログファイルが存在する場合にログファイルに追記する形でログデータを出力します。
7	[ログファイル名]	ログファイルの名前を指定します。
8	[拡張子]	ログファイルの拡張子を指定します。
9	[ファイル名に日付を付ける]	この項目にチェックを付けると、ジョブエントリを実行したときの日付（形式：yyyyMMdd）ファイル名の末尾に追加します。 [ファイル名に日付を付ける]と[ファイル名に時刻を付ける]の両方の項目にチェックを付けた場合は、日付と時刻（形式：_YYYYMMDD_HHMMSS）をファイル名の末尾に追加します。
10	[ファイル名に時刻を付ける]	この項目にチェックを付けると、ジョブエントリを実行したときの時刻（形式：HHmmss）をファイル名の末尾に追加します。 [ファイル名に日付を付ける]と[ファイル名に時刻を付ける]の両方の項目にチェックを付けた場合は、日付と時刻（形式：_YYYYMMDD_HHMMSS）をファイル名の末尾に追加します。
11	[ログレベル]	出力するログのログレベルを次の項目から指定します。 • [ログなし] ログを出力しません。 • [エラーログのみ] エラーログだけを出力します。 • [最小限のログ] 最小限のログだけを出力します。 • [基本のログ] デフォルトのログレベルです。 • [詳細なログ] 詳細なログを出力します。 • [デバッグ] 非常に詳細なログを出力します。目的がデバッグの場合に設定します。 • [ローレベル (非常に詳細)] 大量のログデータを出力します。 【注意事項】 ジョブ実行時、ここで指定したよりも詳細度が低いログを出力する指定をすると、ログは出力されません。例えば、ここで[基本のログ]を指定した場合、ジョブ実行時に[最小限のログ]を指定するとログは出力されません。
12	[先行のジョブ・エントリから値を引き継ぐ]	この項目にチェックを付けると、先行ジョブエントリの実行結果を引数に設定します。
13	[ループ処理を行う]	この項目にチェックを付けると、先行ジョブが結果を返した場合、行ごとにシェルスクリプトを1回実行します。
14	[引数名]	シェルスクリプトの引数として使用する文字列を指定します。

(2) [スクリプト]タブ

実行するシェルスクリプトを直接入力する場合の設定をします。

図 3-45[シェルスクリプトを実行]画面-[スクリプト]タブ

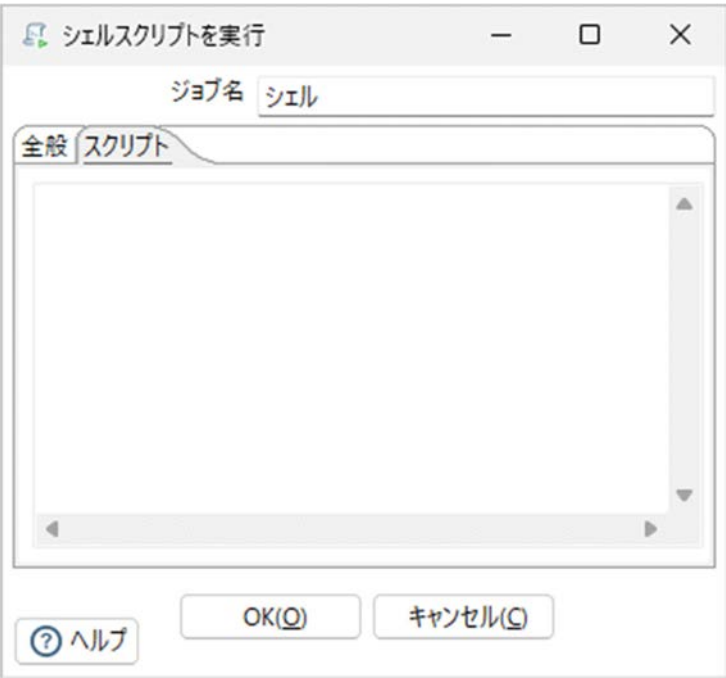


表 3-47 [シェルスクリプトを実行]画面-[スクリプト]タブの項目

項番	項目名	説明
1	(定義領域)	実行するシェルスクリプトを記述します。

3.7 ユーティリティー

- [ユーティリティー]Ping 送信 (Ping a host)
- [ユーティリティー]ログ書き込み (Write to log)
- [ユーティリティー]停止 (Abort job)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.7.1 [ユーティリティー]Ping 送信 (Ping a host)

ICMP プロトコルを使用してホストに ping を送信します。

図 3-45 [ピン送信]画面

表 3-48 [ピン送信]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ホスト名]	ping 送信先のホスト名または IP アドレスを指定します。
3	[タイプ]	ping のタイプを次の項目から指定します。 • [クラシックピン送信] オペレーティングシステムのコマンドを使用します。 • [システムピン送信] Java で実装されているメソッドを使用します。 • [システムピン送信の後にクラシックピン送信] 上記 2 つのオプション両方を実施します。システムピン送信のあとにクラシックピン送信をします。
4	[タイムアウト]	タイムアウト値を ms 単位で指定します。デフォルトは 3000 です。 [タイプ]で[システムピン送信]を選択したときだけ指定できます。 【注意事項】 [タイムアウト]に 0 を指定して実行した場合、エラーになります。

項番	項目名	説明
5	[送出力ケット数]	送信するパケットの数を指定します。デフォルトは2です。 [タイプ]で[クラシックピン送信]を選択したときだけ指定できます。 【注意事項】 [送出力ケット数]に大きい数値（例えば100）を指定してローカル実行した場合、ジョブが終了しなくなります。

3.7.2 [ユーティリティー]ログ書き込み（Write to log）

指定した件名およびメッセージをログに出力します。

図 3-46 [ログ書き込み]画面

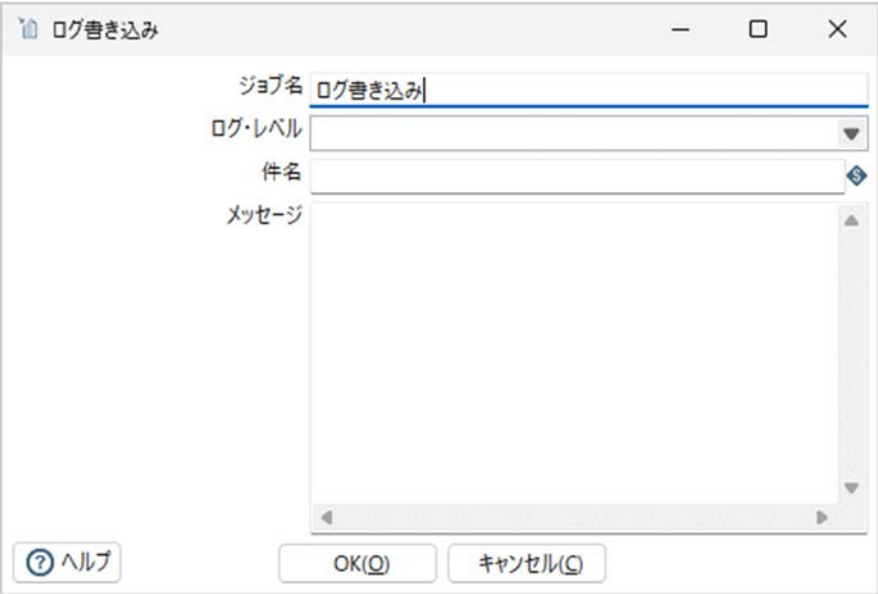


表 3-49 [ログ書き込み]画面の項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ログ・レベル]	出力するログのログレベルを次の項目から指定します。 • [ログなし] ログを出力しません。 • [エラーログのみ] エラーログだけを出力します。 • [最小限のログ] 最小限のログだけを出力します。 • [基本のログ] デフォルトのログレベルです。 • [詳細なログ] 詳細なログを出力します。

項番	項目名	説明
		• [デバッグ] 非常に詳細なログを出力します。目的がデバッグの場合に設定します。 • [ローレベル (非常に詳細)] 大量のログデータを出力します。 【注意事項】 ジョブ実行時、ここで指定したよりも詳細度が低いログを出力する指定をすると、ログは出力されません。例えば、ここで[基本のログ]を指定した場合、ジョブ実行時に[最小限のログ]を指定するとログは出力されません。
3	[件名]	ログに出力する件名を指定します。次の形式でログに出力されます。 YYYY/MM/DD hh:mm:ss - <[件名]に指定した文字列> - <[メッセージ]に指定した文字列>
4	[メッセージ]	ログに出力するメッセージを指定します。次の形式でログに出力されます。 YYYY/MM/DD hh:mm:ss - <[件名]に指定した文字列> - <[メッセージ]に指定した文字列>

3.7.3 [ユーティリティー]停止 (Abort job)

ジョブを停止する場合に使用します。

図 3-47 [停止ジョブ]画面

表 3-50 [停止ジョブ]画面の項目

項番	項目名	説明
1	[停止ジョブ]	ジョブ名を指定します。
2	[メッセージ]	ログに出力するメッセージを指定します。

3.8 ファイル転送

- [ファイル転送]FTP ファイル取得 (Get a file with FTP)
- [ファイル転送]FTP ファイル追加 (Put a file with FTP)
- [ファイル転送]FTP 削除 (FTP delete)
- [ファイル転送]SFTP ファイル取得 (Get a file with SFTP)
- [ファイル転送]SFTP ファイル追加 (Put a file with SFTP)

上記以外のジョブエントリーについては、Pentaho Documentation を参照してください。

3.8.1 [ファイル転送]FTP ファイル取得 (Get a file with FTP)

FTP サーバ上のファイルを取得して、ローカルのファイルシステムへ転送します。ファイル転送時のプロトコルには FTP を使用します。

(1) [一般]タブ

FTP サーバに関する設定をします。

図 3-48 [FTP ファイル取得]画面-[一般]タブ

FTPファイル取得

ジョブ名: Get a file with FTP

一般

ファイル

拡張

Socksプロキシ

サーバー

FTPサーバ名 / IPアドレス:

サーバーポート: 21

ユーザ名:

パスワード:

プロキシホスト:

プロキシポート:

プロキシユーザ名:

プロキシパスワード:

テスト接続

拡張設定

バイナリモード? ☐

タイムアウト: 0

常時接続する ☐

文字コード: ISO-8859-1

ヘルプ

OK(O)

キャンセル(C)

表 3-51 [FTP ファイル取得]画面-[一般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[FTP サーバ名 / IP アドレス]	FTP サーバ名または IP アドレスを指定します。
3	[サーバーポート]	FTP サーバのポート番号を指定します。
4	[ユーザ名]	FTP サーバ接続時に使用するユーザー名を指定します。
5	[パスワード]	FTP サーバ接続時に使用するユーザーパスワードを指定します。
6	[プロキシホスト]	FTP プロキシサーバのホスト名または IP アドレスを指定します。
7	[プロキシポート]	FTP プロキシサーバのポート番号を指定します。
8	[プロキシユーザ名]	FTP プロキシサーバ接続時に使用するユーザー名を指定します。
9	[プロキシパスワード]	FTP プロキシサーバ接続時に使用するユーザーパスワードを指定します。
10	[テスト接続]ボタン	このボタンを押すと、FTP サーバに接続できるかどうかをテストします。
11	[バイナリモード?]	この項目にチェックを付けると、バイナリモードでファイルを転送します。

項番	項目名	説明
		チェックを付けない場合、ASCII モードでファイルを転送します。
12	[タイムアウト]	FTP サーバのタイムアウト時間を秒単位で指定します。
13	[常時接続する]	アクティブモードを使用して FTP サーバに接続する場合は、この項目にチェックを付けてください。 チェックを付けた場合は、FTP クライアントからのプロキシポートへの接続を受け入れられるようにファイアウォールを設定してください。 チェックを付けない場合、パッシブモードで FTP サーバに接続してください。
14	[文字コード]	FTP サーバで使用する文字コードを次の項目から指定します。 • [US-ASCII] • [ISO-8859-1] • [UTF-8] • [UTF-16BE] • [UTF-16LE] • [UTF-16]

(2) [ファイル]タブ

FTP サーバに格納されている取得ファイルに関する設定をします。

図 3-49 [FTP ファイル取得]画面-[ファイル]タブ

FTPファイル取得

ジョブ名: Get a file with FTP

一般 ファイル 拡張 Socksプロキシ

リモート

リモートディレクトリ: ディレクトリの確認

検索文字列:

転送後にファイルを削除する ☐

転送後にファイルを移動する ☐

移動ディレクトリ ディレクトリの確認

ディレクトリを作成 ☐

ローカル

ローカルディレクトリ: 参照...

ファイル名に日付を付ける ☐

ファイル名に時刻を付ける ☐

日付と時刻の表示形式を指定する ☐

表示形式

日付と時刻を拡張子にする ☐

ファイルを上書きしない ☐

ファイルが存在していた場合の処理

ファイル名を結果に含む ☒

ヘルプ OK(O) キャンセル(C)

表 3-52 [FTP ファイル取得]画面-[ファイル]タブの項目

項番	項目名	説明
1	[リモートディレクトリ]	取得するファイルが格納されている FTP サーバ上のディレクトリを指定します。 [ディレクトリの確認]ボタンを押すと、指定したディレクトリが存在するかどうかを確認できます。
2	[検索文字列]	取得するファイル名を正規表現で入力してください。
3	[転送後にファイルを削除する]	この項目にチェックを付けると、ファイル転送後に FTP サーバ上のファイルを削除します。ただし、ファイルを削除するのは、すべてのファイル転送が成功した場合だけです。
4	[転送後にファイルを移動する]	この項目にチェックを付けると、転送後に FTP サーバ上のファイルを移動します。
5	[移動ディレクトリ]	移動先のディレクトリを指定します。 [ディレクトリの確認]ボタンを押すと、指定したディレクトリが存在するかどうかを確認できます。
6	[ディレクトリを作成]	この項目にチェックを付けると、[移動ディレクトリ]で指定したディレクトリが存在しない場合にディレクトリを自動作成します。
7	[ローカルディレクトリ]	取得するファイルを格納するローカルディレクトリを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
8	[ファイル名に日付を付ける]	この項目にチェックを付けると、ジョブエントリを実行したときの日付（形式：yyyyMMdd）を転送後のファイルのファイル名の末尾に追加します。 [ファイル名に日付を付ける]と[ファイル名に時刻を付ける]の両方の項目にチェックを付けた場合は、日付と時刻（形式：_YYYYMMDD_HHMMSS）をファイル名の末尾に追加します。
9	[ファイル名に時刻を付ける]	この項目にチェックを付けると、ジョブエントリを実行したときの時刻（形式：HHmmss）を転送後のファイルのファイル名の末尾に追加します。 [ファイル名に日付を付ける]と[ファイル名に時刻を付ける]の両方の項目にチェックを付けた場合は、日付と時刻（形式：_YYYYMMDD_HHMMSS）をファイル名の末尾に追加します。
10	[日付と時刻の表示形式を指定する]	この項目にチェックを付けると、ジョブエントリを実行したときの日付と時刻を[表示形式]で選択した形式で転送後のファイルのファイル名の末尾に追加します。
11	[表示形式]	転送後のファイルのファイル名の末尾に追加する日付と時刻の形式を選択します。
12	[日付と時刻を拡張子にする]	この項目にチェックを付けると、日付と時刻をファイル拡張子の前に追加します。 補足：項目名と動作が不一致ですが、「日付と時刻を拡張子の前に付ける」と読み替えてお使いください。 例えば、[ファイル名に日付を付ける]にチェックを付けた場合、ファイル名は次のようになります。 • 移動前：＜ファイル名＞.txt • 移動後：＜ファイル名＞_yyyyMMdd.txt
13	[ファイルを上書きしない]	この項目にチェックを付けると、[ローカルディレクトリ]で指定したディレクトリに同じ名前のファイルが存在した場合にファイルを上書きしません。

項番	項目名	説明
14	[ファイルが存在していた場合の処理]	[ローカルディレクトリ]で指定したディレクトリに同じ名前のファイルが存在した場合の処理を次の項目から指定します。 • [スキップ] ファイル取得処理を行いません。 • [ローカルファイルにユニークな名前をつけてください] 転送先にユニークな名称のファイルとして作成、保存します。 • [失敗] ファイルを取得しないで、異常終了します。
15	[ファイル名を結果に含む]	この項目にチェックを付けると、取得したファイルのファイルパスをジョブの実行結果に追加します。

(3) [拡張]タブ

ファイル転送の処理結果に関する条件を設定します。

図 3-50 [FTP ファイル取得]画面-[拡張]タブ



表 3-53 [FTP ファイル取得]画面-[拡張]タブの項目

項番	項目名	説明
1	[条件]	転送処理が成功と見なされる条件を次の項目から指定します。 • [全て正常動作] エラーが発生しなかった場合に転送処理を終了し、成功状態にします。 エラーが発生した場合はその時点で転送処理を終了します（成功状態にはなりません）。 • [x ファイル以上取得] [ファイル数/エラー数]で指定した値以上のファイルが処理された場合に転送処理を終了し、成功状態にします。 指定した値未満のファイルが処理された場合はエラーになります（成功状態にはなりません）。 • [エラー数が x 以下]

項番	項目名	説明
		[ファイル数/エラー数]で指定した値未満の件数のエラーが発生した場合に転送処理を終了し、成功状態にします。 指定した値と同じ件数以上の件数のエラーが発生した場合は転送処理を終了します（成功状態にはなりません）。
2	[ファイル数/エラー数]	[条件]で[x ファイル以上取得]を選択した場合のファイル数、または[エラー数が x 以下]を選択した場合のエラー件数を指定します。

(4) [Socks プロキシ]タブ

SOCKS プロキシサーバを使用する場合の設定をします。

図 3-51 [FTP ファイル取得]画面-[Socks プロキシ]タブ

The screenshot shows a window titled 'FTPファイル取得' (FTP File Acquisition). The 'ジョブ名' (Job Name) field contains 'Get a file with FTP'. Below this are four tabs: '一般' (General), 'ファイル' (File), '拡張' (Extension), and 'Socksプロキシ' (Socks Proxy), with the last one being active. Under the 'Socksプロキシ' tab, there is a section labeled 'プロキシ' (Proxy) containing four input fields: 'ホスト' (Host), 'ポート' (Port) with the value '1080', 'ユーザ名' (Username), and 'パスワード' (Password). Each field has a small blue icon to its right. At the bottom of the window are three buttons: 'ヘルプ' (Help) with a question mark icon, 'OK(O)', and 'キャンセル(C)' (Cancel).

表 3-54 [FTP ファイル取得]画面-[Socks プロキシ]タブの項目

項番	項目名	説明
1	[ホスト]	SOCKS プロキシサーバのホスト名または IP アドレスを指定します。
2	[ポート]	SOCKS プロキシサーバのポート番号を指定します。
3	[ユーザ名]	SOCKS プロキシサーバ接続時に使用するユーザー名を指定します。
4	[パスワード]	SOCKS プロキシサーバ接続時に使用するユーザーパスワードを指定します。

3.8.2 [ファイル転送]FTP ファイル追加（Put a file with FTP）

ローカルのファイルシステム上のファイルを FTP サーバへ転送（追加）します。ファイル転送時のプロトコルには FTP を使用します。

(1) [全般]タブ

FTP サーバに関する設定をします。

図 3-52 [FTP 経由でファイルを転送]画面-[全般]タブ

FTP 経由でファイルを転送 ...

ジョブ名Put a file with FTP

全般

ファイル

Socks プロキシ

サーバ設定

ホスト名

ポート番号21

ユーザー名

パスワード

プロキシホスト名

プロキシポート番号

プロキシユーザー名

プロキシパスワード

テスト接続

拡張設定

バイナリモード

タイムアウト0

常時接続する

文字コードISO-8859-1

ヘルプ

OK(O)

キャンセル(C)

表 3-55 [FTP 経由でファイルを転送]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ホスト名]	FTP サーバ名または IP アドレスを指定します。
3	[ポート番号]	FTP サーバのポート番号を指定します。
4	[ユーザー名]	FTP サーバ接続時に使用するユーザー名を指定します。
5	[パスワード]	FTP サーバ接続時に使用するユーザーパスワードを指定します。
6	[プロキシホスト名]	FTP プロキシサーバのホスト名または IP アドレスを指定します。

項番	項目名	説明
7	[プロキシポート番号]	FTP プロキシサーバのポート番号を指定します。
8	[プロキシユーザー名]	FTP プロキシサーバ接続時に使用するユーザー名を指定します。
9	[プロキシパスワード]	FTP プロキシサーバ接続時に使用するユーザーパスワードを指定します。
10	[テスト接続]ボタン	このボタンを押すと、FTP サーバに接続できるかどうかをテストします。
11	[バイナリモード]	この項目にチェックを付けると、バイナリモードでファイルを転送します。 チェックを付けない場合、ASCII モードでファイルを転送します。
12	[タイムアウト]	FTP サーバのタイムアウト時間を秒単位で指定します。
13	[常時接続する]	アクティブモードを使用して FTP サーバに接続する場合は、この項目にチェックを付けてください。 チェックを付けた場合は、FTP クライアントからのプロキシポートへの接続を受け入れられるようにファイアウォールを設定してください。 チェックを付けない場合、パッシブモードで FTP サーバに接続してください。
14	[文字コード]	FTP サーバで使用する文字コードを次の項目から指定します。 • [US-ASCII] • [ISO-8859-1] • [UTF-8] • [UTF-16BE] • [UTF-16LE] • [UTF-16] ファイル名に日本語が含まれている場合は、適切な文字コードを指定してください。

(2) [ファイル]タブ

FTP サーバに転送（追加）するファイルに関する設定をします。

図 3-53 [FTP 経由でファイルを転送]画面-[ファイル]タブ

FTP 経由でファイルを転送 ...

ジョブ名Put a file with FTP

全般

ファイル

Socks プロキシ

ローカル設定

ローカルディレクトリ参照(B)...

検索文字列

転送後にファイルを削除☐

ファイルを上書きしない☐

リモート設定

リモートディレクトリディレクトリの確認

ヘルプ

OK(O)

キャンセル(C)

表 3-56 [FTP 経由でファイルを転送]画面-[ファイル]タブの項目

項番	項目名	説明
1	[ローカルディレクトリ]	転送するファイルが格納されているローカルディレクトリを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
2	[検索文字列]	転送するファイル名を正規表現で入力してください。
3	[転送後にファイルを削除]	この項目にチェックを付けると、ファイル転送後にローカルのファイルシステムに格納されているファイルを削除します。ただし、ファイルを削除するのは、すべてのファイル転送が成功した場合だけです。
4	[ファイルを上書きしない]	この項目にチェックを付けると、[ローカルディレクトリ]で指定したディレクトリに同じ名前のファイルが存在した場合にファイルを上書きしません。
5	[リモートディレクトリ]	転送先の FTP サーバ上のディレクトリを指定します。 [ディレクトリの確認]ボタンを押すと、指定したディレクトリが存在するかどうかを確認できます。

(3) [Socks プロキシ]タブ

SOCKS プロキシサーバを使用する場合の設定をします。

図 3-54 [FTP 経由でファイルを転送]画面-[Socks プロキシ]タブ

FTP 経由でファイルを転送 ...

ジョブ名Put a file with FTP

全般

ファイル

Socks プロキシ

プロキシ

ホスト:

ポート:

1080

ユーザー名:

パスワード:

ヘルプ

OK(O)

キャンセル(C)

表 3-57 [FTP 経由でファイルを転送]画面-[Socks プロキシ]タブの項目

項番	項目名	説明
1	[ホスト]	SOCKS プロキシサーバのホスト名または IP アドレスを指定します。
2	[ポート]	SOCKS プロキシサーバのポート番号を指定します。
3	[ユーザー名]	SOCKS プロキシサーバ接続時に使用するユーザー名を指定します。
4	[パスワード]	SOCKS プロキシサーバ接続時に使用するユーザーパスワードを指定します。

3.8.3 [ファイル転送]FTP 削除 (FTP delete)

プロトコルに FTP、FTPS、SFTP、または SSH のどれかを使用して、サーバ上のファイルを削除します。

(1) [全般]タブ

サーバの設定をします。

図 3-55 [FTP を通してファイルを削除]画面-[全般]タブ

FTPを通してファイルを削除

ジョブ名 FTP delete

全般 ファイル Socks プロキシ

サーバ設定

プロトコル FTP

ホスト名 (またはIPアドレス)

ポート番号 21

ユーザ名

パスワード

コネクションタイプ FTP

プロキシを使用する ☐

プロキシホスト名

プロキシポート番号

プロキシユーザ名

プロキシパスワード

公開鍵を使用する ☐

公開鍵ファイル名 参照(B)...

パスフレーズ

テスト接続

ヘルプ OK(O) キャンセル(C)

表 3-58 [FTP を通してファイルを削除]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[プロトコル]	サーバへ接続するプロトコルを次の項目から指定します。 • FTP • FTPS • SFTP • SSH
3	[ホスト名 (または IP アドレス)]	サーバのホスト名または IP アドレスを指定します。
4	[ポート番号]	サーバのポート番号を指定します。
5	[ユーザ名]	サーバ接続時に使用するユーザー名を指定します。
6	[パスワード]	サーバ接続時に使用するユーザーパスワードを指定します。
7	[コネクションタイプ]	[プロトコル]で[FTPS]を指定した場合のコネクションタイプを、次の項目から指定します。

項番	項目名	説明
		• [FTP] 通信を暗号化しません。 • [暗黙 SSL] SSL で暗号化されている通信を使用します。 • [Authentication SSL] AUTH SSL コマンドで明示的に暗号化通信を行います。 • [暗黙 SSL を使用した FTP 接続] [暗黙 SSL]に加え、データコネクションポートも暗号化します。 • [認証 TLS] AUTH TLS コマンドで明示的に暗号化通信を行います。 • [暗黙 TLS] TLS で暗号化されている通信を使用します。 • [暗黙 TLS と暗号化された FTP 接続] [暗黙 TLS]に加え、データコネクションポートも暗号化します。
8	[プロキシを使用する]	この項目にチェックを付けると、プロキシサーバを利用できます。 【注意事項】 プロキシサーバを利用すると、ジョブ実行時にエラーが発生します。
9	[プロキシホスト名]	プロキシサーバのホスト名を指定します。
10	[プロキシポート番号]	プロキシサーバのポート番号を指定します。
11	[プロキシユーザ名]	プロキシサーバ接続時に使用するユーザー名を指定します。
12	[プロキシパスワード]	プロキシサーバ接続時に使用するユーザーのパスワードを指定します。
13	[公開鍵を使用する]	この項目にチェックを付けると、[プロトコル]で[SSH]を指定した場合、公開鍵ファイルを使用します。 【注意事項】 公開鍵を利用すると、ジョブ実行時にエラーが発生します。
14	[公開鍵ファイル名]	公開鍵ファイルを指定します。
15	[パスフレーズ]	公開鍵のパスフレーズを指定します。

(2) [ファイル]タブ

サーバ接続の拡張設定、削除するファイルおよび成功条件の設定をします。

図 3-56 [FTP を通してファイルを削除]画面-[ファイル]タブ

FTPを通してファイルを削除

ジョブ名 FTP delete

全般 ファイル Socks プロキシ

拡張設定

タイムアウト 0

常時接続する ☐

リモート設定

先行のジョブエントリから値を引き継ぐ ☐

リモートディレクトリ

検索文字列

フォルダ確認

処理条件の設定

条件 すべて正常

ファイル制限 10

ヘルプ OK(O) キャンセル(C)

表 3-59 [FTP を通してファイルを削除]画面-[ファイル]タブの項目

項番	項目名	説明
1	[タイムアウト]	サーバのタイムアウト時間を秒単位で指定します。
2	[常時接続する]	この項目にチェックを付けると、FTP、または SFTP プロトコルを使用してアクティブ (PORT) モードでサーバに接続します。
3	[先行のジョブエントリから値を引き継ぐ]	削除するファイルが別の変換やジョブエントリーによって動的に生成される場合、または削除するファイルを特定する場合に、この項目にチェックを付けてください。
4	[リモートディレクトリ]	削除するファイルが存在するサーバ側のディレクトリを指定します。 【注意事項】 マルチバイト文字を指定すると、ジョブ実行時にエラーが発生します。
5	[検索文字列]	削除するファイル名の検索文字列を正規表現で指定します。 【注意事項】 マルチバイト文字を指定してジョブを実行してもファイルは削除されません。
6	[条件]	成功条件を次の項目から指定します。 • [すべて正常]

項番	項目名	説明
		すべてのファイルを正常に削除できた場合、成功状態にします。 • [最低 x ファイルを削除] 最低 x ファイルを削除できた場合に、成功状態にします。 • [エラーが指定した数より少ない場合] エラー数が指定した数より少ない場合に、成功状態にします。
7	[ファイル制限]	[条件]で選択した成功条件のしきい値を指定します。

(3) [Socks プロキシ]タブ

SOCKS プロキシサーバを使用する場合の設定をします。

[全般]タブの[プロトコル]に[FTP]を指定した場合に設定します。

図 3-57 [FTP を通してファイルを削除]画面-[Socks プロキシ]タブ

FTP を通してファイルを削除

ジョブ名 FTP delete

全般 ファイル Socks プロキシ

プロキシ

ホスト:

ポート: 1080

ユーザー名:

パスワード:

ヘルプ OK(O) キャンセル(C)

表 3-60 [FTP を通してファイルを削除]画面-[Socks プロキシ]タブの項目

項番	項目名	説明
1	[ホスト]	SOCKS プロキシサーバのホスト名を指定します。 【注意事項】 SOCKS プロキシサーバを利用してジョブを実行するとジョブは終了しません。 また、停止処理を実行してもジョブは終了できません。
2	[ポート]	SOCKS プロキシサーバのポート番号を指定します。
3	[ユーザ名]	SOCKS プロキシサーバ接続時に使用するユーザー名を指定します。
4	[パスワード]	SOCKS プロキシサーバ接続時に使用するユーザーパスワードを指定します。

3.8.4 [ファイル転送]SFTP ファイル取得 (Get a file with SFTP)

SFTP サーバ上のファイルを取得して、ローカルのファイルシステムへ転送します。ファイル転送時のプロトコルには SFTP を使用します。

(1) [全般]タブ

SFTP サーバに関する設定をします。

図 3-58 [セキュア FTP でファイルを取得]画面-[全般]タブ

セキュアFTPでファイルを取得

ジョブ名

Get a file with SFTP

全般

ファイル

サーバ設定

ホスト名

ポート番号

22

ユーザ名

パスワード

プライベートキーファイル名を使用

プライベートキーファイル名

参照(B)...

キーバスフレーズ

プロキシタイプ

プロキシ ホスト

プロキシ ポート

プロキシユーザ名

プロキシパスワード

テスト接続

圧縮

none

ヘルプ

OK(O)

キャンセル(C)

表 3-61 [セキュア FTP でファイルを取得]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ホスト名]	SFTP サーバ名または IP アドレスを指定します。
3	[ポート番号]	SFTP サーバのポート番号を指定します。
4	[ユーザ名]	SFTP サーバ接続時に使用するユーザー名を指定します。
5	[パスワード]	SFTP サーバ接続時に使用するユーザーパスワードを指定します。
6	[プライベートキーファイル名を使用]	この項目にチェックを付けると、プライベートキーファイルを使用します。
7	[プライベートキーファイル名]	プライベートキーファイルを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 【注意事項】 - プライベートキーファイルは PEM ファイル形式である必要があります。 - PuTTY で生成されたプライベートキーは PuTTYgen で OpenSSH のフォーマットに変換する必要があります。

項番	項目名	説明
8	[キーパスフレーズ]	プライベートキーを使用するためのパスフレーズを指定します。
9	[プロキシタイプ]	SFTP プロキシサーバのプロキシタイプを次の項目から指定します。 • [HTTP] • [SOCKS5]
10	[プロキシ ホスト名]	SFTP プロキシサーバのホスト名または IP アドレスを指定します。
11	[プロキシ ポート番号]	SFTP プロキシサーバのポート番号を指定します。
12	[プロキシユーザー名]	SFTP プロキシサーバ接続時に使用するユーザー名を指定します。
13	[プロキシパスワード]	SFTP プロキシサーバ接続時に使用するユーザーパスワードを指定します。
14	[テスト接続] ボタン	このボタンを押すと、SFTP サーバに接続できるかどうかをテストします。
15	[圧縮]	ファイル転送で使用する圧縮形式を次の項目から指定します。 • [none] • [zlib]

(2) [ファイル]タブ

SFTP サーバに格納されている取得ファイルに関する設定をします。

図 3-59 [セキュア FTP でファイルを取得]画面-[ファイル]タブ

表 3-62 [セキュア FTP でファイルを取得]画面-[ファイル]タブの項目

項番	項目名	説明
1	[先行のジョブ・エントリから値を引き継ぐ]	この項目にチェックを付けると、取得するファイルのファイルパスは、先行ジョブのデータ変換またはジョブエントリーによって動的に生成されます。

項番	項目名	説明
2	[リモートディレクトリ]	取得するファイルが格納されている SFTP サーバ上のディレクトリを指定します。 [ディレクトリの確認]ボタンを押すと、指定したディレクトリが存在するかどうかを確認できます。
3	[検索文字列]	取得するファイル名を正規表現で入力してください。
4	[転送後にファイルを削除する]	この項目にチェックを付けると、ファイル転送後に SFTP サーバ上のファイルを削除します。ただし、ファイルを削除するのは、すべてのファイル転送が成功した場合だけです。
5	[ローカル・ディレクトリ]	取得するファイルを格納するローカルディレクトリを指定します。 [参照]ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
6	[ディレクトリを作成する]	この項目にチェックを付けると、[ローカル・ディレクトリ]で指定したディレクトリが存在しない場合にディレクトリを自動作成します。
7	[ファイル名を結果に含む]	この項目にチェックを付けると、取得したファイルのファイルパスをジョブの実行結果に追加します。

3.8.5 [ファイル転送]SFTP ファイル追加（Put a file with SFTP）

ローカルのファイルシステム上のファイルを SFTP サーバへ転送（追加）します。ファイル転送時のプロトコルには SFTP を使用します。

(1) [全般]タブ

SFTP サーバに関する設定をします。

図 3-60 [セキュア FTP でファイルを転送]画面-[全般]タブ

セキュアFTPでファイルを転送

ジョブ名Put a file with SFTP

全般

ファイル

サーバ設定

ホスト名

ポート番号22

ユーザ名

パスワード

プライベートキーファイルを使用

プライベートキーファイル名参照(B)...

キーパスフレーズ

プロキシタイプ

プロキシ ホスト

プロキシ ポート

プロキシ ユーザ名

プロキシ パスワード

テスト接続

圧縮none

ヘルプ

OK(O)

キャンセル(C)

表 3-63 [セキュア FTP でファイルを転送]画面-[全般]タブの項目

項番	項目名	説明
1	[ジョブ名]	ジョブ名を指定します。
2	[ホスト名]	SFTP サーバ名または IP アドレスを指定します。
3	[ポート番号]	SFTP サーバのポート番号を指定します。
4	[ユーザ名]	SFTP サーバ接続時に使用するユーザー名を指定します。
5	[パスワード]	SFTP サーバ接続時に使用するユーザーパスワードを指定します。
6	[プライベートキーファイル を使用]	この項目にチェックを付けると、プライベートキーファイルを使用します。
7	[プライベートキーファイル 名]	プライベートキーファイルを指定します。 [参照] ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。 【注意事項】 プライベートキーファイルは PEM ファイル形式である必要があります。
8	[キーパスフレーズ]	プライベートキーを使用するためのパスフレーズを指定します。
9	[プロキシ タイプ]	SFTP プロキシサーバのプロキシタイプを次の項目から指定します。 • [HTTP] • [SOCKS5]

3. ジョブ

Pentaho 機能説明 PDI 編

417

項番	項目名	説明
10	[プロキシ ホスト]	SFTP プロキシサーバのホスト名または IP アドレスを指定します。
11	[プロキシ ポート]	SFTP プロキシサーバのポート番号を指定します。
12	[プロキシ ユーザ名]	SFTP プロキシサーバ接続時に使用するユーザー名を指定します。
13	[プロキシ パスワード]	SFTP プロキシサーバ接続時に使用するユーザーパスワードを指定します。
14	[テスト接続] ボタン	このボタンを押すと、SFTP サーバに接続できるかどうかをテストします。
15	[圧縮]	ファイル転送で使用する圧縮形式を次の項目から指定します。 • [none] • [zlib]

(2) [ファイル]タブ

SFTP サーバに転送（追加）するファイルに関する設定をします。

図 3-61 [セキュア FTP でファイルを転送]画面-[ファイル]タブ

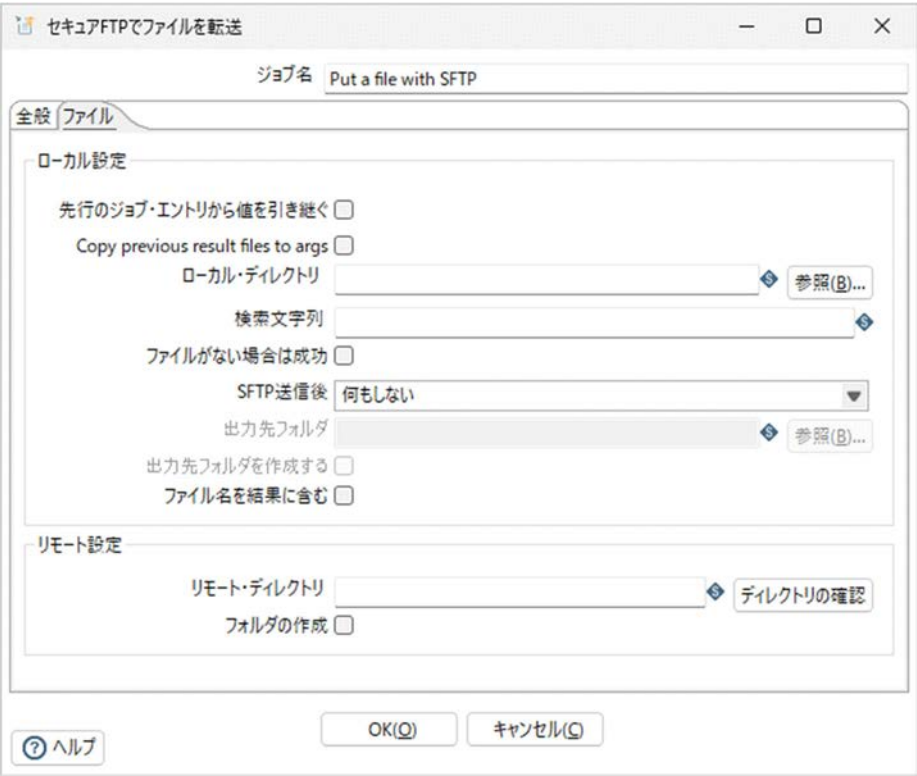


表 3-64 [セキュア FTP でファイルを転送]画面-[ファイル]タブの項目

項番	項目名	説明
1	[先行のジョブ・エントリから値を引き継ぐ]	この項目にチェックを付けると、取得するファイルのファイルパスは、先行ジョブのデータ変換またはジョブエントリーによって動的に生成されます。 【注意事項】 • 引き継ぐフィールド名の条件はありません。

項番	項目名	説明
		先頭のフィールドだけが引き継がれます。
2	[Copy previous result files to args]	この機能は動作しません。
3	[ローカル・ディレクトリ]	転送するファイルが格納されているローカルディレクトリを指定します。 [参照] ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
4	[検索文字列]	転送するファイル名を正規表現で入力してください。
5	[ファイルがない場合は成功]	この項目にチェックを付けると、転送するファイルが存在しない場合は正常終了、存在する場合は異常終了します。
6	[SFTP 送信後]	ファイル転送後の処理を次の項目から指定します。 [何もしない] ファイル転送処理を中止して、正常終了します。 [ファイル削除] 転送元のファイルを削除します。 [ファイルを移動] 転送元のファイルを移動します。
7	[出力先フォルダ]	[SFTP 送信後]で[ファイルを移動]を指定した場合、ファイルの移動先ディレクトリを指定します。 [参照] ボタンを押すと、PDI クライアント（Spoon）を起動しているマシンのファイルシステムを参照できます。
8	[出力先フォルダを作成する]	この項目にチェックを付けると、[SFTP 送信後]で[ファイルを移動]を指定した場合、ファイルの移動先に作成するフォルダを指定します。
9	[ファイル名を結果に含む]	この項目にチェックを付けると、転送したファイルのファイルパスをジョブの実行結果に追加します。
10	[リモート・ディレクトリ]	転送先の SFTP サーバ上のディレクトリを指定します。 [ディレクトリの確認] ボタンを押すと、指定したディレクトリが存在するかどうかを確認できます。
11	[フォルダの作成]	この項目にチェックを付けると、[リモート・ディレクトリ]で指定したディレクトリが存在しない場合にディレクトリを自動作成します。

索引

記号

- [Kettle オプション] 20
- [Scheduler]パースペクティブ 18
- [データ変換]パースペクティブ (ジョブ) 16, 323
- [データ変換]パースペクティブ (データ変換) 12, 24

A

- Abort (データ変換) 203
- Abort job (ジョブ) 398
- Access 入力 32
- Add a checksum 125
- Add constants 140
- Add sequence 127
- Add value fields changing sequence 129
- Analytic query 296
- Append streams 192

B

- Blocking step 184
- Block this step until steps finish 192

C

- Calculator 153
- Check Db Connections 382
- Checks if files exists 384
- Clone row 182
- Concat fields 135
- Copy Files 355
- Copy rows to result 304
- Create a folder 377
- Create file 358
- CSV file input 38
- CSV 入力 38

D

- Database join 265
- Database lookup 263

- Data grid 79
- Data Validator 286
- Delete 118
- Delete file 358
- Delete files 359
- Delete folders 377
- Detect empty stream 206
- De-serialize from file 62
- Dimension lookup/update 278
- Dummy (do nothing) (データ変換) 193
- Dummy (ジョブ) 329

E

- ESRI shapefile reader 41
- ESRI Shapefile Reader 41
- Evaluate rows number in a table 380
- Excel 入力 42
- Execute a process 169
- Execute row SQL script 231
- Execute SQL script 228

F

- File exists (Step) (データ変換) 266
- File Exists (ジョブ) 383
- Filter rows 201
- Fixed File input 90
- Formula 236
- FTP delete 408
- FTP 削除 408
- FTP ファイル取得 399
- FTP ファイル追加 404

G

- Generate rows 92
- Get a file with FTP 399
- Get a file with SFTP 413
- Get data from XML 50

Get file names 87
Get rows from result 303
Get system info 63
Get Variables 301
Group by 290

H

HTTP 351
HTTP Client 250
HTTP クライアント 250

I

If field value is null 165
Insert / update 120

J

Java Filter 185
JavaScript 389
Java スクリプト 208
Java フィルター 185
Job 329
Job Executor 186
Join rows (cartesian product) 275
JSON input 56
JSON Input 56
JSON output 94

L

Load file content in memory 81

M

Mail (ジョブ) 344
Mail (データ変換) 171
Mapping (sub-transformation) 310
Mapping input specification 314
Mapping output specification 315
Memory group by 293
Merge join 271
Merge rows (diff) 274

Metadata structure of stream 170
Microsoft Access input 32
Microsoft Excel input 42
Microsoft Excel writer 98
Modified Java Script value 208
Move files 366
Multiway merge join 272

N

Null if 181
NULL 値定数変換 165
Number range 141

P

Pentaho reporting output 103
Pentaho レポート出力 103
Ping a host 396
Ping 送信 396
PostgreSQL bulk loader 316
PostgreSQL バルクロード 316
Prioritize streams 204
Process files 167
Put a file with FTP 404
Put a file with SFTP 416
Python Executor 220

R

Regex evaluation 245
Replace in string 146
Reservoir sampling 299
REST Client 254
REST クライアント 254
Row denormaliser 151
Row Flattener 148
Row normaliser 150

S

Sample rows 293
Select values 159

Serialize to file 108
Set field value 130
Set field value to a constant 134
Set variables (job entry) (ジョブ) 342
Set Variables (データ変換) 302
SFTP put 319
SFTP ファイル取得 413
SFTP ファイル追加 416
Shell 392
Simple evaluation 385
Simple Mapping (sub-transformation) 306
Sorted merge 273
Sort rows 148
Split fields 131
Split field to rows 133
SQL 391
SQL 文実行 228
SQL 文実行(行) 231
START 327
Stream lookup 260
String operations 143
Strings cut 145
Success 343
Switch / case 205

T

Table Exists (Step) (データ変換) 262
Table exists (ジョブ) 382
Table input 78
Table output 113
Text file input 67
Text file output 109
Transformation 336
Transformation executor 194

U

Unique rows 162
Unique rows (HashSet) 163
Univariate statistics 297

Unzip file 371
Update 122
User defined java expression 232

V

Value mapper 139

W

Wait for 385
Wait for a file 365
Write to file 354
Write to log (ジョブ) 397
Write to log (データ変換) 179

X

XML Join 269
XML output 104
XML 結合 269

Z

Zip file 360

あ

値マッピング 139

か

空ストリーム探知 206
簡易検証 385

き

キューブ出力 108
キューブ入力 62
行結合 (デカルト積) 275
行取得 303
行正規化 150
行生成 92
行整列 148
行送信 304
行非正規化 151
行複製 182

行フラット化 148
行マージ (比較) 274

く

グループ化 290

け

計算 153
結合 269

こ

更新 122
固定幅ファイル入力 90

さ

削除 118
参照 250
サンプル行 293

し

シーケンス追加 127
シーケンス追加 (変化値) 129
シェル 392
システムデータ取得 63
出力 94
条件 380
条件分岐 (Switch/case) 205
ジョブ 301, 329
ジョブ実行 186
新機能 319
シンプルマッピング (サブデータ変換) 306

す

数学関数 213
数式 236
数値範囲 141
スクリプト 208, 389
ステップブロック 192
ストリーム結合 192

ストリーム参照 260

せ

正規表現 245
成功 343
整列併合 273
選択/名前変更 159
全般 327

そ

挿入/更新 120

た

待機 385
多方向 Merge Join 272
ダミー (何もしない) (データ変換) 193
単変量統計 297

ち

重複行削除 162
重複行削除(HashSet) 163
貯留層サンプリング 299

て

停止 (ジョブ) 398
停止 (データ変換) 203
定数 NULL 変換 181
定数追加 140
ディメンジョン 参照/更新 278
データウェアハウス 278
データグリッド 79
データ検証 286
データタイプの対応表 (Java の場合) 31
データタイプの対応表 (PDI から Python ヘデータを渡す場合) 224
データタイプの対応表 (Python から PDI ヘデータを渡す場合) 228
データベース結合 265
データベース参照 263
データベース接続確認 382

データ変換 336
テーブル確認 (ジョブ) 382
テーブル確認 (データ変換) 262
テーブル検証 380
テーブル出力 113
テーブル入力 78
テキストファイル出力 109
テキストファイル入力 67

と

統計 290
特殊関数 216
トランスフォーメーション実行 194

に

入力 32

は

バルクロード 316

ひ

日付関数 213

ふ

ファイル圧縮 360
ファイル移動 366
ファイル解凍 371
ファイル確認 (ジョブ) 383
ファイル確認 (データ変換) 266
ファイル確認 (複数) 384
ファイル管理 351
ファイルコピー 355
ファイル削除 358
ファイル削除 (複数) 359
ファイル作成 358
ファイル処理 167
ファイル操作関数 219
ファイル待機 365
ファイル転送 399

ファイルへ書き込み 354
ファイル名取得 87
ファイルメモリロード 81
フィールド結合 135
フィールド値設定 130
フィールド定数設定 134
フィールド分割 131
フィールド分割 (行) 133
フィルター 201
フォルダー削除 377
フォルダー作成 377
フロー 184
プロセス実行 169
分析クエリー 296

へ

変換 125
変数取得 301
変数設定 (ジョブ) 342
変数設定 (データ変換) 302

ま

マージ結合 271
マッピング 306
マッピング (サブデータ変換) 310
マッピング出力 315
マッピング入力 314

め

メール 344
メール (ジョブ) 344
メール (データ変換) 171
メタデータ構造 170
メモリーグループ化 293

も

文字列オペレーション 143
文字列カット 145
文字列関数 211

文字列置換 [146](#)

ゆ

ユーザー定義 Java クラス [232](#)

優先順位 [204](#)

ユーティリティ (データ変換) [165](#)

ユーティリティー (ジョブ) [396](#)

ろ

ログ書き込み (ジョブ) [397](#)

ログ書き込み (データ変換) [179](#)

論理関数 [215](#)