

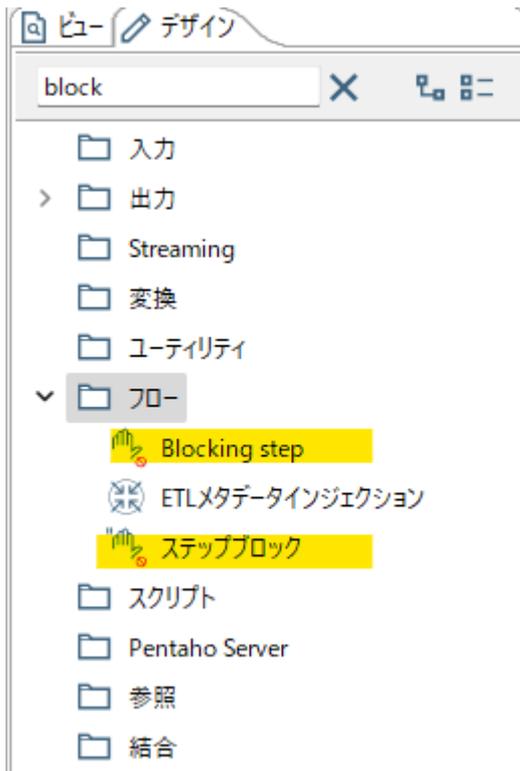
# ステップブロックとブロッキングステップ

2024-02-02

名前も似ているし曖昧に話をされているようにも見たので混乱しないための備忘録

## 見つからない...

片方だけ日本語訳されてるので見つけれない人もいたりするかもblockで検索すると出てくる。



## ステップブロック

使い方としては独立したフロー間の同期を取るのに使うのが主目的だと思われる。

### webの情報

検索で見つかる画像付きの情報 [QiitaPentaho Data Integration Spoonステップ逆引きまとめ](#)

### 使用例

このフローだと実行状況でファイル内容が変わる。テキストファイル出力の実行順序が不定のためID順にテキストを書き出せないし、同時書き込みが発生してファイルが壊れる事も。

定義

実行1回目

ID	DATA
1	
2	ブロッコリー
3	カリフラワー
4	レンコン
5	
6	ほうれん草
7	

実行2回目

ID	DATA
1	
2	ブロッコリー
3	カリフラワー
4	レンコン
5	人参
6	じゃがいも
7	ほうれん草
8	

実行3回目

ID	DATA
1	
2	ブロッコリー
3	カリフラワー
4	レンコン
5	さやえんどう
6	

ステップ「ステップブロック」で前提処理の終了を待ち合わせることができる。

定義

ステップブロック

ステップ名 ステップブロック 2

#	ステップ名	コピー数
1	テキストファイル出力 新規作成 1	0

ヘルプ OK(O) ステップ取得 キャンセル(C)

ステップ名 ステップブロック 3

#	ステップ名	コピー数
1	テキストファイル出力 追加書き 2	0

ヘルプ OK(O) ステップ取得 キャンセル(C)

結果

ID	DATA
1	
2	ブロッコリー
3	カリフラワー
4	レンコン
5	人参
6	じゃがいも
7	ほうれん草
8	長ねぎ
9	筍
10	さやえんどう
11	

繰り返し実行しても変わらない

### データグリッド・ファイル出力

入出力のステップの定義は以下。

### データグリッド

ステップ名 データグリッド 1

#	ID	DATA
1	1	ブロッコリー
2	2	カリフラワー
3	3	レンコン

ステップ名 データグリッド 2

#	ID	DATA
1	4	人参
2	5	じゃがいも
3	6	ほうれん草

ステップ名 データグリッド 3

#	ID	DATA
1	7	長ねぎ
2	8	筍
3	9	さやえんどう

### テキストファイル出力

ステップ名 テキストファイル出力 新規作成 1

ファイル名 E:\pentaho\testdata\食材 参照(B)...

出力をサブプレットに渡す

親フォルダを作成

開始時にファイルを作成しない

先行のステップから値を引き継ぐ

フィールド名

拡張子 csv

ファイル名にステップ番号を含む

ファイル名に区切り番号を含む

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

ステップ名 テキストファイル出力 新規作成 1

既存のファイルに追加する

区切り文字 . タブの挿入(D)

引用符 "

フィールドを引用符で囲む

囲い込みフィックス無効化

ヘッダー

フッター

フォーマット CR+LF terminated (Windows, DOS)

圧縮形式 None

文字コード UTF-8

フィールドの右側を空白で埋める

ダンプ出力する

指定されたレコード数でファイルを分ける

終了レコードを追加する

ステップ名 テキストファイル出力 追加書き 2

既存のファイルに追加する

区切り文字 . タブの挿入(D)

引用符 "

フィールドを引用符で囲む

囲い込みフィックス無効化

ヘッダー

フッター

フォーマット CR+LF terminated (Windows, DOS)

圧縮形式 None

文字コード UTF-8

フィールドの右側を空白で埋める

ダンプ出力する

指定されたレコード数でファイルを分ける

終了レコードを追加する

追加書き 3も同じ。

## ブロッキングステップ

ブロッキングステップの直前のステップが完全終了するまで（全てのレコードを送信し終えるまで）待ち合わせする（と説明にはある）

普通に利用していると使う事はまずないように思う。

### Webの情報

デッドロックが発生している場合に使うらしい。検索するとこんな例が見つかる。 [Transformation Deadlocks](#)

### 使用例

以下処理は日本郵便株式会社の公開する全国郵便番号ファイルを使い、各都道府県の自治体数(市区町村数)を算出して47件のレコードを生成してからオリジナルの該当都道府県レコード124,365件に自治体数を付与する。

- デッドロック理由は、124,365件がステップ「行整列2」と同じソートを行っていないため、ステップ「マージ結合」での結合キー不一致により47件のストリームとの結合不可なレコードが発生し、対応レコードの入力待ちになったため、と思われる。
- デッドロック解消理由は、124,365件側にステップ「ブロッキングステップ」を入れることで124,365件すべてに47件との突合せができるようになったため、と思われる。

#### デッドロック

実行結果

ステップ名	コピー	読み込み	書き込み	入力	出力	更新	除外	エラー	状況	時間	速度 (r/s)	入力/出力
1 CSV入力	0	0	20000	10001	0	0	0	0	実行中	48.4s	413	0/10000
2 行整列 2	0	10000	0	0	0	0	0	0	実行中	48.4s	205	0/0
3 グループ化 市区町村数	0	0	0	0	0	0	0	0	実行中	48.4s	0	0/0
4 マージ結合	0	0	0	0	0	0	0	0	実行中	48.4s	0	10000/0
5 選択/名前変更	0	0	0	0	0	0	0	0	実行中	48.4s	0	0/0
6 ダミー (何もしない)	0	0	0	0	0	0	0	0	実行中	48.4s	0	0/0

#### デッドロック解消

実行結果

ステップ名	コピー	読み込み	書き込み	入力	出力	更新	除外	エラー	状況	時間	速度 (r/s)	入力/出力
1 CSV入力	0	0	248730	124365	0	0	0	0	完了	2.0s	125,431	
2 行整列 2	0	124365	124365	0	0	0	0	0	完了	2.1s	59,877	
3 Blocking step	0	124365	124365	0	0	0	0	0	完了	3.5s	35,716	
4 グループ化 市区町村数	0	124365	47	0	0	0	0	0	完了	2.1s	59,676	
5 マージ結合	0	124412	124365	0	0	0	0	0	完了	3.8s	32,948	
6 選択/名前変更	0	124365	124365	0	0	0	0	0	完了	3.8s	32,892	
7 ダミー (何もしない)	0	124365	124365	0	0	0	0	0	完了	3.8s	32,805	

#### ブロッキングステップ

ステップ名: Blocking step

すべての行を許可する

テンポラリディレクトリ: %%java.io.tmpdir%%

テンポラリファイルの接続子: block

キャッシュサイズ: 5000

テンポラリファイルを圧縮する

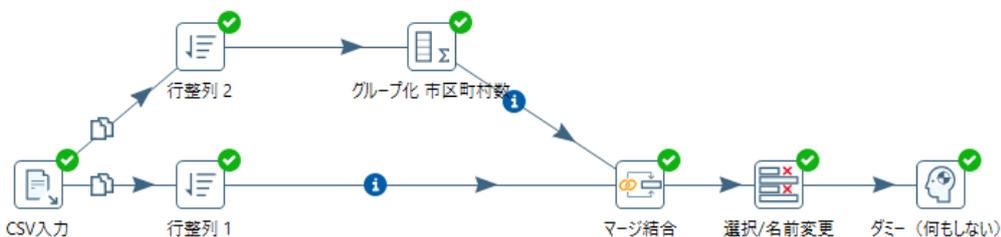
ヘルプ OK キャンセル

### 本来はブロッキングステップ不要

ただし、ほとんどの場合正しく作ってればデッドロックは起きない。ブロッキングステップ導入は本当に最後の手にしておくべきかと思う。

### ちゃんと作ればブロッキングステップは要らない

日本郵便株式会社が提供する全国郵便番号ファイルを入力します



### 実行結果

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

#	ステップ名	コピー	読み込み	書き込み	入力	出力	更新	除外	エラー	状況	時間	速度 (r/s)	入力/出力
1	CSV入力	0	0	248730	124365	0	0	0	0	完了	0.2s	1,480,536	-
2	行整列 2	0	124365	124365	0	0	0	0	0	完了	0.3s	487,706	-
3	行整列 1	0	124365	124365	0	0	0	0	0	完了	0.4s	340,726	-
4	グループ化 市区町村数	0	124365	47	0	0	0	0	0	完了	0.3s	474,676	-
5	マージ結合	0	124412	124365	0	0	0	0	0	完了	0.7s	182,155	-
6	選択/名前変更	0	124365	124365	0	0	0	0	0	完了	0.7s	181,026	-
7	ダミー (何もしない)	0	124365	124365	0	0	0	0	0	完了	0.7s	179,200	-

技術資料, etl, pentaho, ステップブロック, ブロッキングステップ

From:

<https://wiki.hgotoh.jp/> - 努力したWiki

Permanent link:

<https://wiki.hgotoh.jp/documents/etl/pentaho/pentaho-004>

Last update: 2024/02/02 21:20

