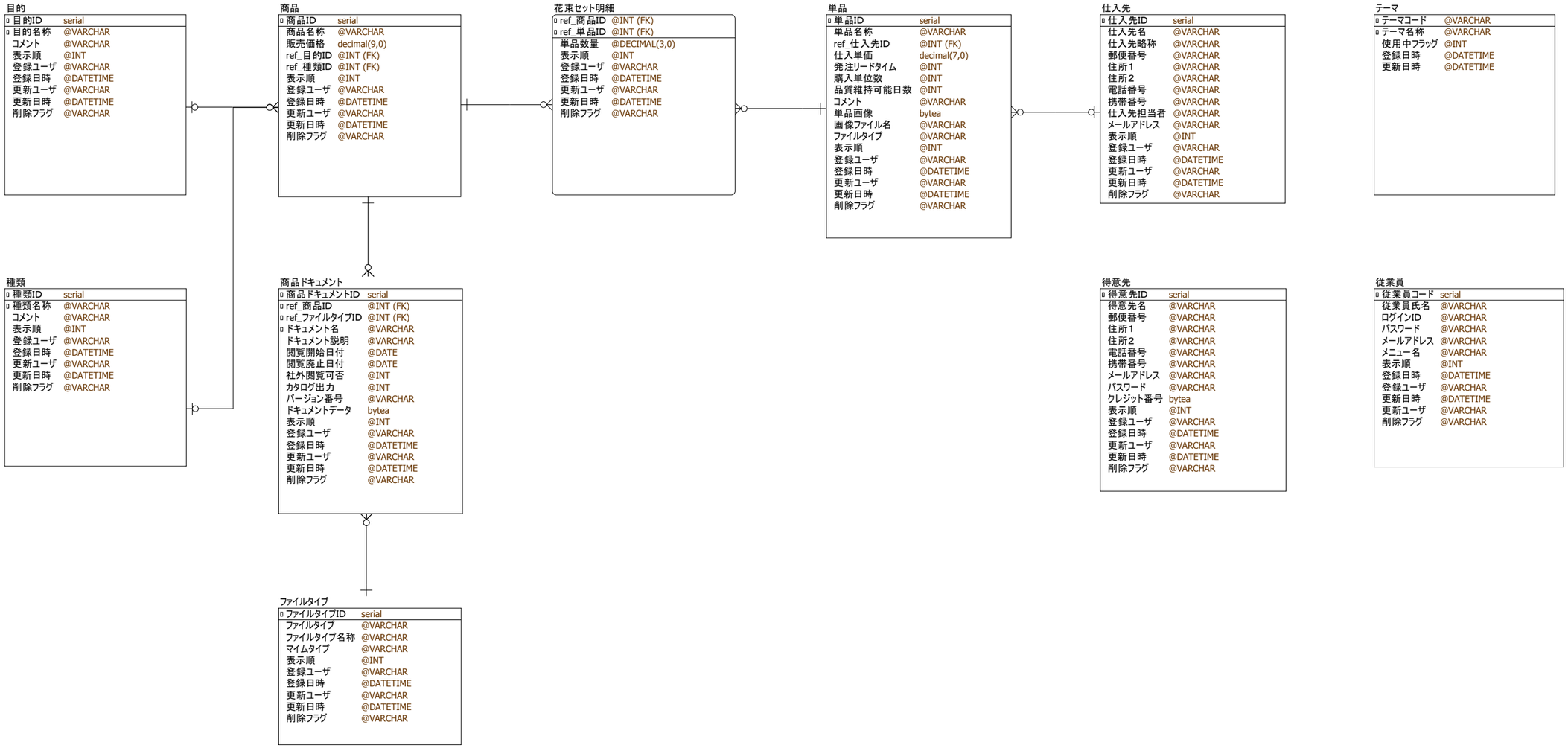
マスター


目的
目的ID serial
目的名称 @VARCHAR
コメント @VARCHAR
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
商品
商品ID serial
商品名称 @VARCHAR
販売価格 decimal(9,0)
ref_目的ID @INT (FK)
ref_種類ID @INT (FK)
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
花束セット明細
ref_商品ID @INT (FK)
ref_単品ID @INT (FK)
単品数量 @DECIMAL(3,0)
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
単品
単品ID serial
単品名称 @VARCHAR
ref_仕入先ID @INT (FK)
仕入単価 decimal(7,0)
発注リードタイム @INT
購入単位数 @INT
品質維持可能日数 @INT
コメント @VARCHAR
単品画像 bytea
画像ファイル名 @VARCHAR
ファイルタイプ @VARCHAR
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
仕入先
仕入先ID serial
仕入先名 @VARCHAR
仕入先略称 @VARCHAR
郵便番号 @VARCHAR
住所1 @VARCHAR
住所2 @VARCHAR
電話番号 @VARCHAR
携帯番号 @VARCHAR
仕入先担当者 @VARCHAR
メールアドレス @VARCHAR
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
テーマ
テーマコード @VARCHAR
テーマ名称 @VARCHAR
使用中フラッグ @INT
登録日時 @DATETIME
更新日時 @DATETIME
種類
種類ID serial
種類名称 @VARCHAR
コメント @VARCHAR
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
商品ドキュメント
商品ドキュメントID serial
ref_商品ID @INT (FK)
ref_ファイルタイプID @INT (FK)
ドキュメント名 @VARCHAR
ドキュメント説明 @VARCHAR
閲覧開始日付 @DATE
閲覧廃止日付 @DATE
社外閲覧可否 @INT
カタログ出力 @INT
バージョン番号 @VARCHAR
ドキュメントデータ bytea
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
得意先
得意先ID serial
得意先名 @VARCHAR
郵便番号 @VARCHAR
住所1 @VARCHAR
住所2 @VARCHAR
電話番号 @VARCHAR
携帯番号 @VARCHAR
メールアドレス @VARCHAR
パスワード @VARCHAR
クレジット番号 bytea
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR
従業員
従業員コード serial
従業員氏名 @VARCHAR
ログインID @VARCHAR
パスワード @VARCHAR
メールアドレス @VARCHAR
メニュー名 @VARCHAR
表示順 @INT
登録日時 @DATETIME
登録ユーザ @VARCHAR
更新日時 @DATETIME
更新ユーザ @VARCHAR
削除フラグ @VARCHAR
ファイルタイプ
ファイルタイプID serial
ファイルタイプ @VARCHAR
ファイルタイプ名称 @VARCHAR
マイムタイプ @VARCHAR
表示順 @INT
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
削除フラグ @VARCHAR

トランザクション

受注ヘッダー
受注ID serial
受注日付 @DATE
ref_得意先ID @INT
ref_商品ID @INT
受注金額 @DECIMAL(9,0)
お届け日 @DATE
贈り主氏名 @VARCHAR
お届け先郵便番号 @VARCHAR
お届け先住所1 @VARCHAR
お届け先住所2 @VARCHAR
お届け先氏名 @VARCHAR
お届け先電話番号 @VARCHAR
メッセージ要不要 @INT
お届けメッセージ @VARCHAR
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
出荷ヘッダー
出荷ID serial
受注日付 @DATE
ref_得意先ID @INT
ref_商品ID @INT
出荷金額 @DECIMAL(9,0)
お届け日 @DATE
贈り主氏名 @VARCHAR
お届け先郵便番号 @VARCHAR
お届け先住所1 @VARCHAR
お届け先住所2 @VARCHAR
お届け先氏名 @VARCHAR
お届け先電話番号 @VARCHAR
メッセージ要不要 @INT
お届けメッセージ @VARCHAR
送り状番号 @VARCHAR
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
単品在庫明細
在庫日付 @DATE
ref_単品ID @INT
ref_入荷ID @INT
ref_行番号 @INT
前日在庫数量 @DECIMAL(7,0)
入荷数量 @DECIMAL(7,0)
出荷数量 @DECIMAL(7,0)
廃棄数量 @DECIMAL(7,0)
当日在庫数量 @DECIMAL(7,0)
品質維持期限 @DATE
単品予定在庫
在庫日付 @DATE
ref_単品ID @INT
ref_発注ID @INT
ref_行番号 @INT
前日在庫数量 @DECIMAL(7,0)
入荷予定数量 @DECIMAL(7,0)
加工予定数量 @DECIMAL(7,0)
廃棄予定数量 @DECIMAL(7,0)
当日在庫数量 @DECIMAL(7,0)
品質維持期限 @DATE
受注明細
ref_受注ID @INT (FK)
ref_単品ID @INT
受注数量 @DECIMAL(7,0)
出荷予定日付 @DATE
予定在庫更新フラグ @VARCHAR
出荷明細
ref_出荷ID @INT (FK)
ref_単品ID @INT
出荷数量 @DECIMAL(7,0)
出荷日付 @DATE
単品在庫計算用ワーク
在庫日付 @DATE
ref_単品ID @INT
ref_入荷ID @INT
ref_行番号 @INT
前日在庫数量 @DECIMAL(7,0)
入荷数量 @DECIMAL(7,0)
出荷数量 @DECIMAL(7,0)
廃棄数量 @DECIMAL(7,0)
当日在庫数量 @DECIMAL(7,0)
品質維持期限 @DATE
単品予定在庫計算用ワーク1
在庫日付 @DATE
ref_単品ID @INT
ref_発注ID @INT
ref_行番号 @INT
前日在庫数量 @DECIMAL(7,0)
入荷予定数量 @DECIMAL(7,0)
加工予定数量 @DECIMAL(7,0)
廃棄予定数量 @DECIMAL(7,0)
当日在庫数量 @DECIMAL(7,0)
品質維持期限 @DATE
単品予定在庫計算用ワーク2
在庫日付 @DATE
ref_単品ID @INT
ref_発注ID @INT
ref_行番号 @INT
前日在庫数量 @DECIMAL(7,0)
入荷予定数量 @DECIMAL(7,0)
加工予定数量 @DECIMAL(7,0)
廃棄予定数量 @DECIMAL(7,0)
当日在庫数量 @DECIMAL(7,0)
品質維持期限 @DATE
発注ヘッダー
発注ID serial
発注日付 @DATE
ref_仕入先ID @INT
希望納入日付 @DATE
伝票出力フラグ @VARCHAR
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
入荷ヘッダー
入荷ID serial
発注日付 @DATE
ref_仕入先ID @INT
入荷日付 @DATE
納品書番号 @VARCHAR
登録ユーザ @VARCHAR
登録日時 @DATETIME
更新ユーザ @VARCHAR
更新日時 @DATETIME
発注明細
ref_発注ID @INT (FK)
行番号 @INT
ref_単品ID @INT
発注数量 @DECIMAL(7,0)
単価 @DECIMAL(7,0)
品質維持期限 @DATE
予定在庫更新フラグ @VARCHAR
入荷明細
ref_入荷ID @INT (FK)
行番号 @INT
ref_単品ID @INT
入荷数量 @DECIMAL(7,0)
単価 @DECIMAL(7,0)
品質維持期限 @DATE

単品在庫明細　更新処理

単品在庫明細について
・日付別に単品の
前日在庫数量・入荷数量・出荷数量・廃棄数量・当日在庫数量を表わす。
過去日付けの在庫を管理する。

・在庫日付、ref_単品ID、ref_入荷ID、ref_行番号毎にデータを持つ
入荷明細をもとに、在庫日付を付与して作成する。
当日在庫数量が0になるまで、翌日に繰越される。
したがって、同じref_単品ID、ref_入荷ID、ref_行番号のデータが、複数の在庫日付に
またがって存在することになる。
理由は、入荷ID毎に、品質維持期限が設定され期限がきた単品は使用できなくなり、
廃棄されるため。

・更新タイミング
業務終了後のバッチ処理（当日の入荷・出荷データを使用して更新する）

在庫更新処理　バッチ起動

起動処理
1．バッチ処理キーの作成　前処理1
単品在庫明細・出荷明細・入荷明細から単品IDを抽出する。
抽出条件：単品在庫明細・出荷明細・入荷明細全件
抽出項目：DISTINCT 単品ID

2．バッチ処理キーの作成　前処理2
単品在庫明細から最大在庫日付を取得する。

3．バッチ処理キーの作成
バッチ処理キー作成用のRuby関数をコールする。
上記1・2の出力データを使用して、単品ID・在庫抽出日・在庫日付の
バッチ用キーを作成する。

4．在庫更新バッチを起動する。

在庫更新処理　バッチ

1．バッチ処理キーの取得
単品ID・在庫抽出日・在庫日付を1件取得し、バッチ用リクエストJSONのバッチ用キーにセットする。
この処理はフレームワークが自動で行う。

2．単品在庫計算用ワークをクリア（削除）する
削除条件：ref_単品ID　＝　単品ID（バッチ処理キー）

3．単品在庫明細から繰越データを抽出し、単品在庫計算用ワークへ出力する
抽出条件：単品ID　＝　単品ID（バッチ処理キー）
　AND　在庫日付　＝　在庫抽出日（バッチ処理キー）
出力編集：在庫日付　　←　在庫日付（バッチ処理キー）
　　前日在庫数量　←　当日在庫数量（単品在庫明細）
　　ref_単品ID・ref_入荷ID・ref_行番号・当日在庫数量・品質維持期限は単品在庫明細の項目を設定する。
　　入荷数量・出荷数量・廃棄数量は0を設定する。

4．入荷明細を抽出し、単品在庫計算用ワークへ出力する
抽出条件：ref_単品ID　＝　単品ID（バッチ処理キー）
　AND　入荷日付　＝　在庫日付（バッチ処理キー）
出力編集：在庫日付　←　在庫日付（バッチ処理キー）
　　入荷数量　←　入荷数量（入荷明細）
　　当日在庫数量　←　入荷数量（入荷明細）
　　ref_単品ID・ref_入荷ID・ref_行番号・当品質維持期限は入荷明細（と入荷ヘッダー）の項目を設定する
　　前日在庫数量・出荷数量・廃棄数量は0を設定する。

5．出荷明細を抽出し、出荷数量をサマリーする（同時に単品マスターから品質維持可能日数を取得しておく）
抽出条件：ref_単品ID　＝　単品ID（バッチ処理キー）
　AND　出荷日付　＝　在庫日付（バッチ処理キー）

注：品質維持可能日数は、繰越データ・入庫データが無く、出荷データのみが発生した時の為に取得しておく。
　この時、出荷データから、単品在庫計算用ワークを作成する必要がある。

6．単品在庫計算用ワークから、繰越データと入荷データを抽出する
抽出条件：ref_単品ID　＝　単品ID（バッチ処理キー）
　AND　在庫日付　＝　在庫日付（バッチ処理キー）
ソート順：ref_入荷ID, ref_行番号

7．Ruby　在庫引当処理関数を呼び出し
　処理1：サマリー出荷数量　＞　当日在庫数量　のとき
　　当日出荷数量　←　当日在庫数量
　　当日在庫数量　←　0
　　単品在庫計算用ワークを更新（キー：在庫日付・単品ID・ref_入荷ID・ref_行番号）

　　サマリー出荷数量　＝　サマリー集荷数量　－　当日出荷数量
　　処理1へ（次の行を処理する）
　　ただし、最終行の時は、処理1ではなく、処理2を行う

　処理2：サマリー出荷数量　＜＝　当日在庫数量
　　当日出荷数量　←　サマリー出荷数量
　　当日在庫数量　←　当日在庫数量　－　サマリー出荷数量
　　単品在庫計算用ワークを更新（キー：在庫日付・単品ID・ref_入荷ID・ref_行番号）
　　Rubyの処理終わり

8．廃棄処理（単品在庫計算用ワークを下記条件で更新）
更新条件：ref_単品ID　　＝　単品ID（バッチキー）
　AND　品質維持期限　<=　在庫日付（バッチ処理キー）
　AND　当日在庫数量　>0
更新編集：廃棄数量　　←　当日在庫数量
　　当日在庫数量　←　0

9．単品在庫明細削除処理（この削除処理は、同じ日に在庫更新バッチを二度実行された時の為）
削除キー：ref_単品ID　＝　単品ID（バッチキー）
　AND　品質維持期限　<=　在庫日付（バッチ処理キー）

10．単品在庫計算用ワークを抽出し、単品在庫明細へ追加する
抽出条件：ref_単品ID　＝　単品ID（バッチ処理キー）
　AND　入荷日付　＝　在庫日付（バッチ処理キー）

単品予定在庫　更新処理1

単品予定在庫について
・日付別に単品の
前日在庫数量・入荷予定数量・加工予定数量・廃棄予定数量・当日在庫数量を表わす。
単品在庫明細は過去日付の在庫を管理するが、単品予定在庫は、未来日付の予定を管理する。

・更新タイミング1
日次処理の単品在庫明細更新後、続けて、予定在庫を作成
・・・発注データと受注データを全件処理（総洗い替え）

・更新タイミング2
在庫予定更新処理の起動画面（随時処理）
・・・発注データを入力後、発注データに含まれる単品IDを洗い替え

・更新タイミング3
受注入力時、在庫予定更新処理（受注）を起動
・・・受注データを入力後、受注データに含まれる単品IDを洗い替え

単品予定在庫更新処理1
単品在庫明細・発注明細・受注明細を使用して、新たに単品予定在庫を作成する。
毎日、単品在庫明細更新後に更新を行う。

起動処理
1. 単品予定在庫を全件削除する

2. 単品予定在庫計算用ワーク1を全件削除する

3. 単品在庫明細の最大在庫日付を抽出する

4. 単品在庫明細を抽出し、単品予定在庫を作成する
抽出条件：在庫日付　＝　引当開始日

注：引当開始日は、Ruby関数を使用して設定する
3の最大在庫日付とシステム日付を比較し、小さい日付を引当開始日とする。

5. バッチ処理キー作成　前処理1
発注明細・受注明細から単品IDを抽出する。
抽出条件：発注明細・受注明細全件
抽出項目：DISTINCT 単品ID

6. バッチ処理キー作成　前処理2
発注ヘッダー・受注明細から引当終了日を設定する。
抽出条件：発注ヘッダー・受注明細全件
抽出項目：引当終了日

7. バッチ処理キーの作成
バッチ処理キー作成用のRuby関数をコールする。
4の引当開始日と5・6の出力データを使用して、単品ID・引当日の
バッチ用キーを作成する。

8. 単品予定在庫更新バッチを起動する。

単品予定在庫更新バッチ
1. バッチ処理キーの取得
単品ID・引当日を1件取得し、バッチ用リクエストJSONのバッチ用キーにセットする。
この処理はフレームワークが自動で行う。

2. 単品予定在庫を抽出し、単品予定在庫計算用ワーク1へ出力（前日繰越データ作成）
抽出条件：ref_単品ID　＝　ref_単品ID（バッチキー）
AND　在庫日付　＝　引当日（バッチキー）　－　1
AND　当日在庫数量　＞　0
出力編集：在庫日付　←　引当日（バッチキー）
前日在庫数量　←　当日在庫数量
品質維持期限　←　品質維持期限
ref_単品ID・ref_入荷ID・ref_行番号・当日在庫数量はそのまま移送する。
入庫数量・出庫数量・廃棄数量に0を設定する。

3. 発注明細を抽出し、単品予定在庫計算用ワーク1へ出力（入庫予定データ作成）
抽出条件：ref_単品ID　＝　ref_単品ID（バッチキー）
AND　希望納入日付　＝　引当日（バッチキー）
出力編集：在庫日付　←　引当日（バッチキー）
入荷予定数量　←　発注数量
当日在庫数量　←　発注数量
ref_単品ID・ref_入荷ID・ref_行番号はそのまま移送する。
前日在庫数量・加工予定数量・廃棄予定数量に0を設定する。

4. 受注明細を抽出し、受注数量（受注合計数量）をサマリーする（同時に単品マスターから品質維持可能日数を取得しておく）
抽出条件：ref_単品ID　＝　単品ID（バッチ処理キー）
AND　出荷予定日付　＝　引当日（バッチキー）

注：品質維持可能日数は、繰越データ・発注データが無く、受注データのみが発生した時の為に取得しておく。
この時、受注ータから、単品予定在庫計算用ワーク1を作成する必要がある。

5. 単品予定在庫計算用ワーク1を抽出する（前日繰越データと入荷予定データ）
抽出条件：ref_単品ID　＝　ref_単品ID（バッチキー）
AND　在庫日付　＝　引当日（バッチキー）
ソート順：ref_発注ID・ref_行番号

6. Ruby　在庫引当処理関数を呼び出し
処理1：受注合計数量　＞　当日在庫数量　のとき
受注合計数量　＝　受注合計数量　－　当日在庫数量

加工予定数量　＝　当日在庫数量
当日在庫数量　←　0
単品予定在庫計算用ワーク1を更新（キー：在庫日付・単品ID・ref_発注ID・ref_行番号）
処理1へ（次の行を処理する）
ただし、最終行の時は、処理1ではなく、処理2を行う

処理2：受注合計数量　＜＝　当日在庫数量
加工予定数量　←　受注合計数量
当日在庫数量　←　当日在庫数量　－　受注合計数量
単品予定在庫計算用ワーク1を更新（キー：在庫日付・単品ID・ref_発注ID・ref_行番号）
Ruby処理終わり

7. 単品予定在庫計算用ワーク1の廃棄予定数量を更新
更新条件：ref_単品ID　＝　ref_単品ID（バッチキー）
AND　品質維持期限　＜＝　引当日（バッチキー）
AND　当日在庫数量　＞　0
更新編集：廃棄予定数量　←　当日在庫数量
当日在庫数量　←　0

8. 単品予定在庫ワーク1を単品予定在庫へ追加する
追加条件：ref_単品ID　＝　ref_単品ID（バッチキー）
AND　在庫日付　＝　引当日（バッチキー）

注：このバッチ処理は、予定在庫更新日次処理、予定在庫更新随時処理、受注入力（登録・訂正・削除）から
起動される共通のバッチ処理となる。

単品予定在庫　更新処理2

単品予定在庫更新処理2
　当日入力した発注データを使用して、単品予定在庫を更新する。
　発注入力を行い、必要都度実行する。

起動処理
1．バッチ処理キー作成　前処理1
　当日入力した発注データから単品ID毎に希望納入日付の最小日付と最大日付を取得する
　抽出条件：更新日時　＝　システム日付
　抽出項目：単品ID・MIN（希望納入日付）・MAX（希望納入日付）
　グループ：単品ID

2．バッチ処理キー作成　前処理2
　単品予定在庫から最大在庫日付を取得する。

3．バッチ処理キーの作成
　バッチ処理キー作成用のRuby関数をコールする。
　1・2の出力データを使用して、単品ID・引当日のバッチ用キーを作成する。

4．単品予定在庫を削除する
　削除条件：ref_単品ID　＝　ref_単品ID（バッチ処理キー）
　　AND　在庫日付　＞＝　最小希望納入日付（上記1）

5．単品予定在庫計算用ワーク1を削除する
　削除条件：ref_単品ID　＝　ref_単品ID（バッチ処理キー）

6．単品予定在庫更新バッチを起動する。

単品予定在庫　更新処理3

単品予定在庫更新処理3
　当日入力した受注データを使用して、単品予定在庫を更新する。
　受注入力（登録・変更・削除）毎に起動処理が実行される。

起動処理
1．バッチ処理キー作成　前処理1
　当日入力した受注データから単品ID毎に希望納入日付の最小日付と最大日付を取得する
　抽出条件：更新日時　＝　システム日付
　抽出項目：単品ID・MIN（出荷予定日付）・MAX（出荷予定日付）
　グループ：単品ID

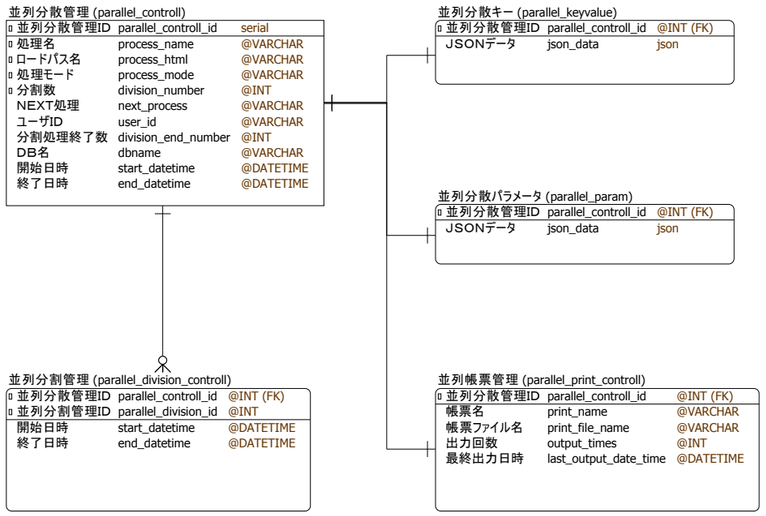
2．バッチ処理キー作成　前処理2
　単品予定在庫から最大在庫日付を取得する。

3．バッチ処理キーの作成
　バッチ処理キー作成用のRuby関数をコールする。
　1・2の出力データを使用して、単品ID・引当日のバッチ用キーを作成する。

4．単品予定在庫を削除する
　削除条件：ref_単品ID　＝　ref_単品ID（バッチ処理キー）
　　AND　在庫日付　＞＝　出荷予定日付（上記1）

5．単品予定在庫計算用ワーク1を削除する
　削除条件：ref_単品ID　＝　ref_単品ID（バッチ処理キー）

6．単品予定在庫更新バッチを起動する。


並列分散管理 (parallel_controll)
並列分散管理ID parallel_controll_id serial
処理名 process_name @VARCHAR
ロードパス名 process_html @VARCHAR
処理モード process_mode @VARCHAR
分割数 division_number @INT
NEXT処理 next_process @VARCHAR
ユーザID user_id @VARCHAR
分割処理終了数 division_end_number @INT
DB名 dbname @VARCHAR
開始日時 start_datetime @DATETIME
終了日時 end_datetime @DATETIME
並列分散キー (parallel_keyvalue)
並列分散管理ID parallel_controll_id @INT (FK)
JSONデータ json_data json
並列分散パラメータ (parallel_param)
並列分散管理ID parallel_controll_id @INT (FK)
JSONデータ json_data json
並列分割管理 (parallel_division_controll)
並列分散管理ID parallel_controll_id @INT (FK)
並列分割管理ID parallel_division_id @INT
開始日時 start_datetime @DATETIME
終了日時 end_datetime @DATETIME
並列帳票管理 (parallel_print_controll)
並列分散管理ID parallel_controll_id @INT (FK)
帳票名 print_name @VARCHAR
帳票ファイル名 print_file_name @VARCHAR
出力回数 output_times @INT
最終出力日時 last_output_date_time @DATETIME