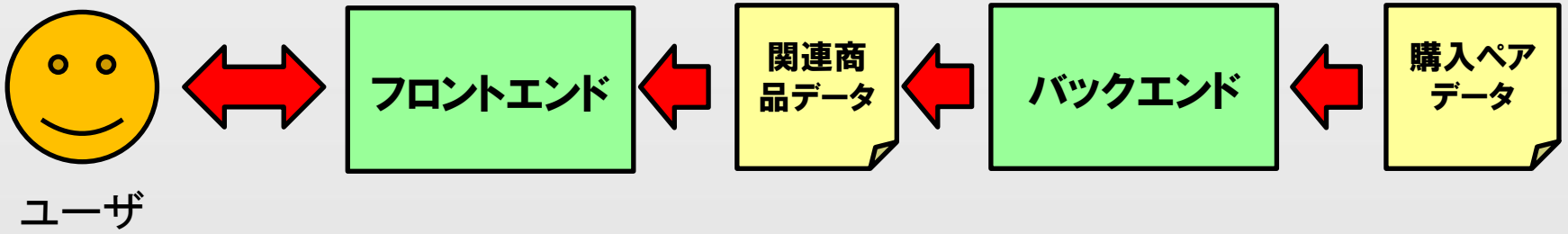
レコメンデーションエンジン システム仕様

システム仕様

- フロントエンド（実装済み）
  - 選択された商品に対してバックエンドが生成する関連商品データをもとに、関連した商品を見つける
  - 関連した商品から関連度の高いものを最大3つ選択し、レコメンドする
- バックエンド
  - 入力には過去に一緒に購入された商品のペアが記録された「**購入ペアデータ**」を用いる（購入ペアデータの詳細は「別紙 レコメンデーションシステムデータ仕様」を参照）
  - 各商品に対して、別な商品が一緒に購入される確率を商品間の「**関連度**」とし、関連度が2.5%以上の商品を**関連商品データに**記録する
  - **ある商品Xと一緒に購入される商品Yの関連度の計算方法は、次の通り**

$$\text{関連度} = \frac{\text{商品Xと商品Yを含む購入ペアの総数}}{\text{商品Xを含む購入ペアの総数}}$$



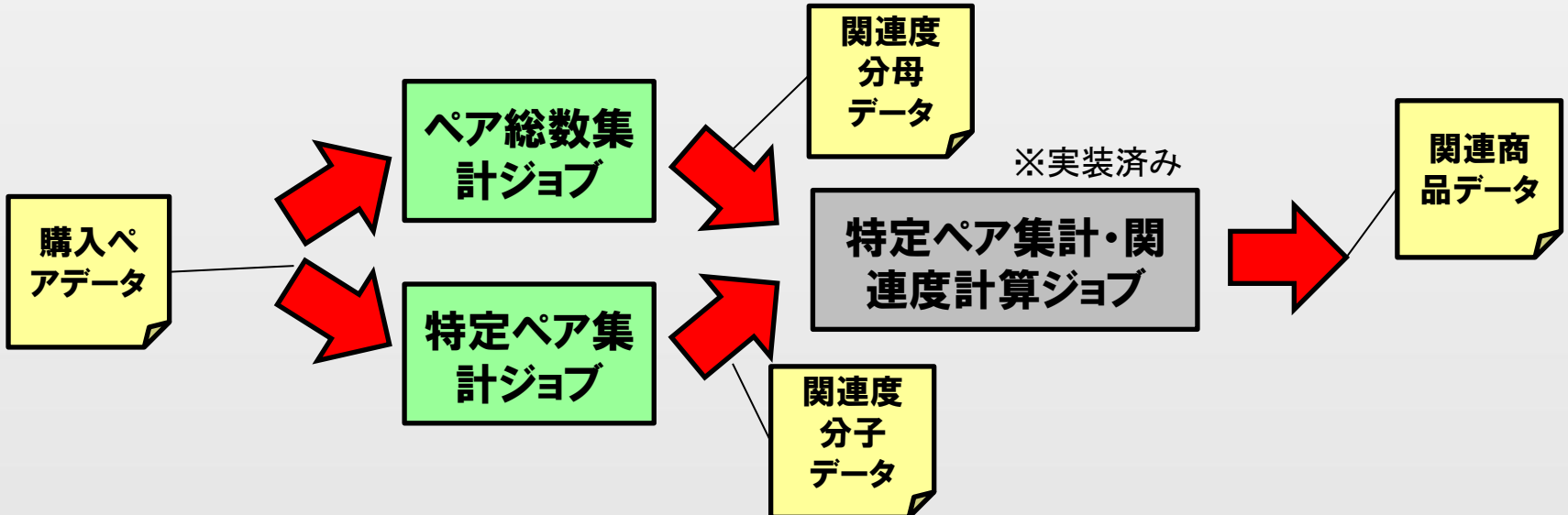
バックエンドジョブ仕様

① 関連度の計算のうち、分母を計算するジョブ（ペア総数集計ジョブ）
- 計算した分母のデータは、**関連度分母データに出力する**（関連度分母データの仕様は、別紙データ仕様の関連度分母データの仕様」を参照）

② 関連度の計算のうち、分子の計算を行うジョブ（特定ペア集計ジョブ）
- 計算した分子データは、**関連度分子データ**に出力する（関連度分子データの仕様は、別紙データ仕様の「関連度分子データの仕様」を参照）

③ ①と②で求めた分母と分子のデータから、関連度を求めるジョブ（関連度計算ジョブ）
- **このジョブはすでに実装済みである**

■ 各ジョブの流れ、および入出力データは下図の通り



ペア総数集計ジョブ

- 商品ごとに、購入データペアに出現した回数を集計し、「関連度分母データ」を生成する
- Mapタスクでの処理
  - mapメソッドに入力されたレコードを分解し、次の中間データを生成する
    - Key:総数を計算する商品名
    - Value:1
  - カンマで区切られた2つの商品の両方について中間データを生成する必要があることに注意
- Reduceタスクでの処理
  - reduceメソッドには商品名をKeyとし、複数個の「１」が記録されたIterableオブジェクトがValueとしてわたってくる。これを利用し、特定の商品が購入ペアデータに出現した回数を求め、Keyが示す商品の分母データとして出力する
  - reduceタスクの出力は、KeyとValueの区別は特にないので、どちらかにカンマ区切りの分母データを出力し、もう片方には空の出力を指定する

特定ペア集計ジョブ

- 特定の商品のペアが、購入ペアデータにどのくらい出現したかを計算し、「関連度分子データ」に記録する
- Mapタスクでの処理
  - mapメソッドには「購入データペア」のレコードが渡される。このレコードを分解し、次のように中間データを生成する
    - Key:商品データのペア
    - Value:1
  - 中間データのKeyは、ペアの商品名をカンマ区切りにした文字列とし、商品名が昇順に並ぶように記録すること
- Reduceタスクの処理
  - reduceメソッドのKeyには、カンマ区切りでペアになった商品名が文字列として渡される。またValueには複数個の「１」が格納されたIterableオブジェクトが渡されるので、これらをもとに特定の商品ペアの出現回数を集計する

関連度計算ジョブ（参考）

- 前二つのジョブで求めた分子データと分母データをもとに、購入ペアデータに出現する商品ペアの関連度を求める
- Mapタスクでの処理
  - mapメソッドには「関連度分子データ」か「関連度分母データ」のいずれかのレコードが入力される。あらかじめsetupメソッド内で、入力されたレコードには入力されたレコードがどのファイルのものかを判定し、分子データの場合ペアの1つ目の商品名をKeyとし残りをValueとして中間データを出力する。分母データの場合、商品名+”#d（識別子）”をKeyとし、残りをValueとして中間データを出力する。
- Reduceタスクでの処理
  - reduceメソッドには、分母の商品名がKeyとして渡され、Valueには先頭の1要素を除き、分子の商品ペアと出現回数が並んだIterableオブジェクトが渡される。このIterableオブジェクトの先頭の要素は分母の商品の出現回数となるようにする
- その他
  - 中間データをReduceタスクに振り分ける際、中間データのKeyの商品名の部分が同じデータを同じReduceタスクに振り分ける。

※このジョブには、講義の範囲外の手法（「レコードが属する入力ファイルのパスの取得」と、「セカンダリソート」と呼ばれる手法）を利用している