

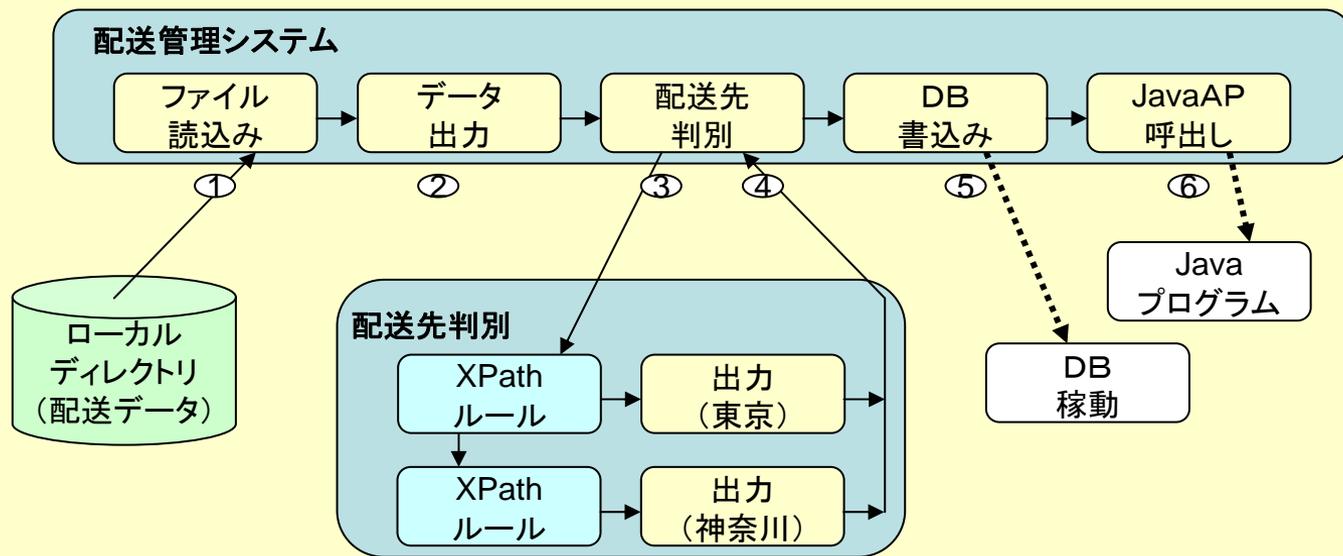
# PolarLake(その1)

「**ノンコーディング**でほとんどの開発が可能」という特徴を検証.

## ■ 評価結果

- ・個々の部品の仕様(利用方法)が比較的わかりやすい.
- ・部品のドラッグ&ドロップと関連属性値の設定を繰り返して、アプリケーションの処理を記述できる.
- ・1週間位の受講で、エンドユーザ(業務の専門家)主導で開発可.  
但し、SEの支援は必要.
- ・XML, XPath, DB, Java などに関して、基本的知識は必要.

## ■ 例題アプリケーション(配送管理システム)のシステム構成&処理プロセス



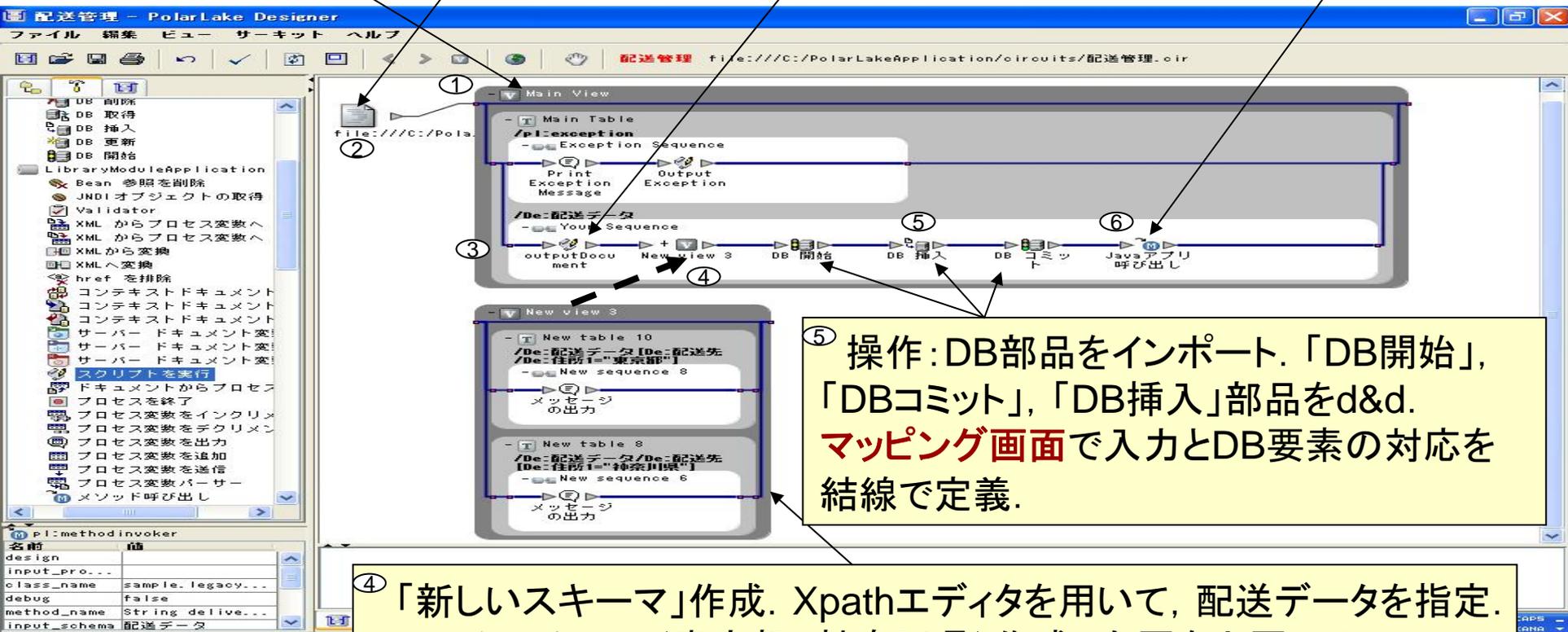
# ■アプリケーション定義手順

② 「ファイル読み込み」部品をd&d. ファイルの場所指定して, 結線.

① データ処理用のテンプレート指定. 処理フローの枠組み設定

⑥ 操作: 「メソッド呼び出し」部品をd&d. クラス名, メソッド名, Jarファイル指定. **マッピング画面**で入力とメソッド引数の対応とメソッド戻り値と出力の対応を結線で定義.

③ 「スクリプト実行」部品をd&d. outputDocument (配送データ出力) を選択.



④ 「新しいスキーマ」作成. Xpathエディタを用いて, 配送データを指定. 2つのシーケンス(東京都, 神奈川県)作成. 左図を上図へd&d. 配送データの「住所」の値により, 上段と下段に分岐.

# ■ マッピング画面で入力とDBテーブル要素の対応を結線で定義.

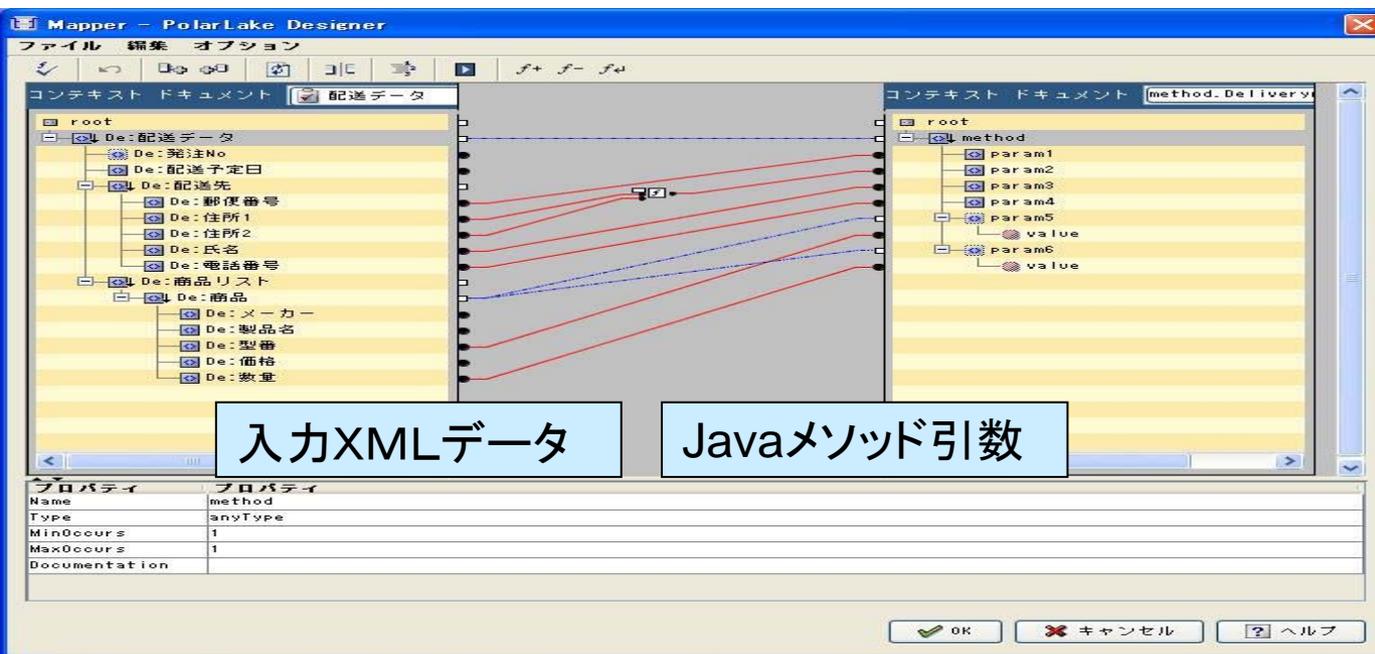
- ・ **青線**は、構造の生成. **赤線**は、データの移動.
- ・ 図中の結線途中の「f」は関数呼び出し.

以下の例: 2つの入力値を結合 (concatenation) する処理

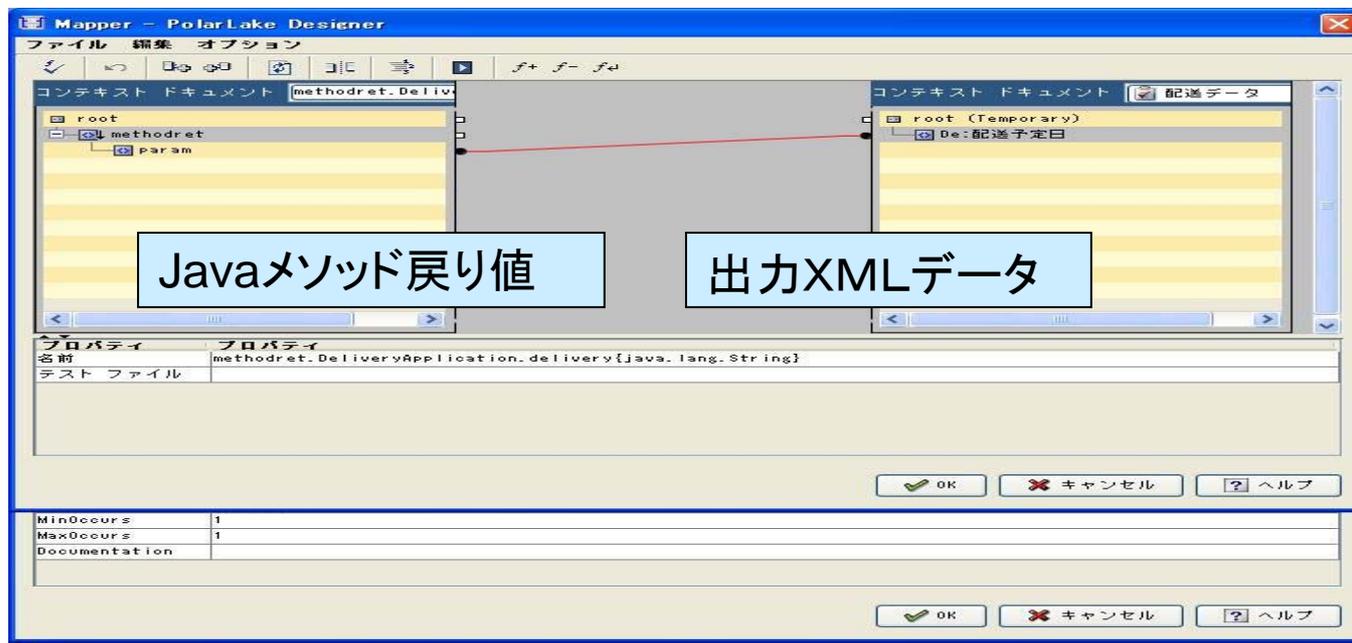
The screenshot shows the Mapper - PolarLake Designer interface. The left pane displays the input XML structure under the context '配送データ'. The right pane displays the database table structure under the context 'dbtxn.insert.DE'. The central area shows a mapping configuration with blue lines for structure generation and red lines for data movement. A function 'f' is used to concatenate two address fields from the input XML into a single address field in the database table.

プロパティ	プロパティ
Id	7
名前	concat
説明	配列内にある値を連結します
Parameter	String []
Parameter	[0] xpath - /De:配送データ/De:配送先/De:住所1
Parameter	[1] xpath - /De:配送データ/De:配送先/De:住所2
出力	/aa:DELIVERY_DATA-list/aa:DELIVERY_DATA/aa:ADDRESS/text()

■ **マッピング画面**でJavaアプリへの入力とメソッド引数の対応を定義.



■ Javaアプリからのメソッド実行結果と出力の対応を結線で定義.



# ■実行開始:「配送管理」をコンソール画面にドラッグ & ドロップ

クリックでDB起動

クリックでDBマネージャ起動

d&dで, 実行開始

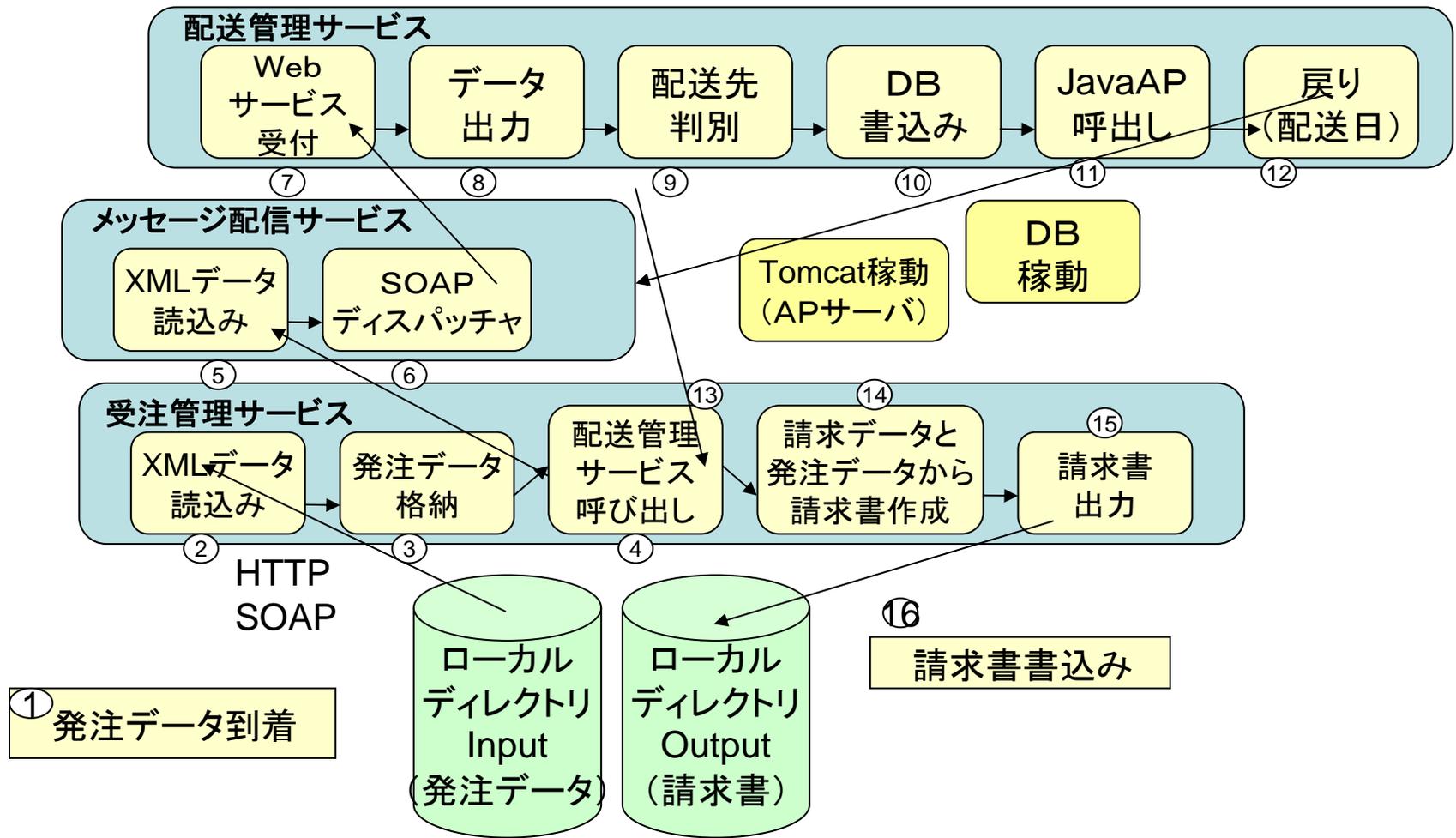
右クリックでサーバ起動

The screenshot displays the PolarLake Management Console interface. The main window is titled "PolarLake Management Console" and contains a menu bar (ファイル, ツール, Web, アプリケーション, ウィンドウ, ヘルプ) and a toolbar. On the left, a tree view shows the project structure under "PolarLake サーバー", including "MyPolarLakeServer" and various circuit components like "ランタイム ブート サーキット" and "com.polar.lake.circuit". The "配送管理" component is highlighted in the tree. A dashed arrow indicates it is being dragged towards the console output area. The console output area on the right shows XML data for a delivery record and a log message from the database module. The log message includes details such as the thread ID, database operation, and delivery address information. The Windows taskbar at the bottom shows the Start button and several open applications, including Internet Explorer, Windows Explorer, and Java.

## ■ Webサービス連携の例題

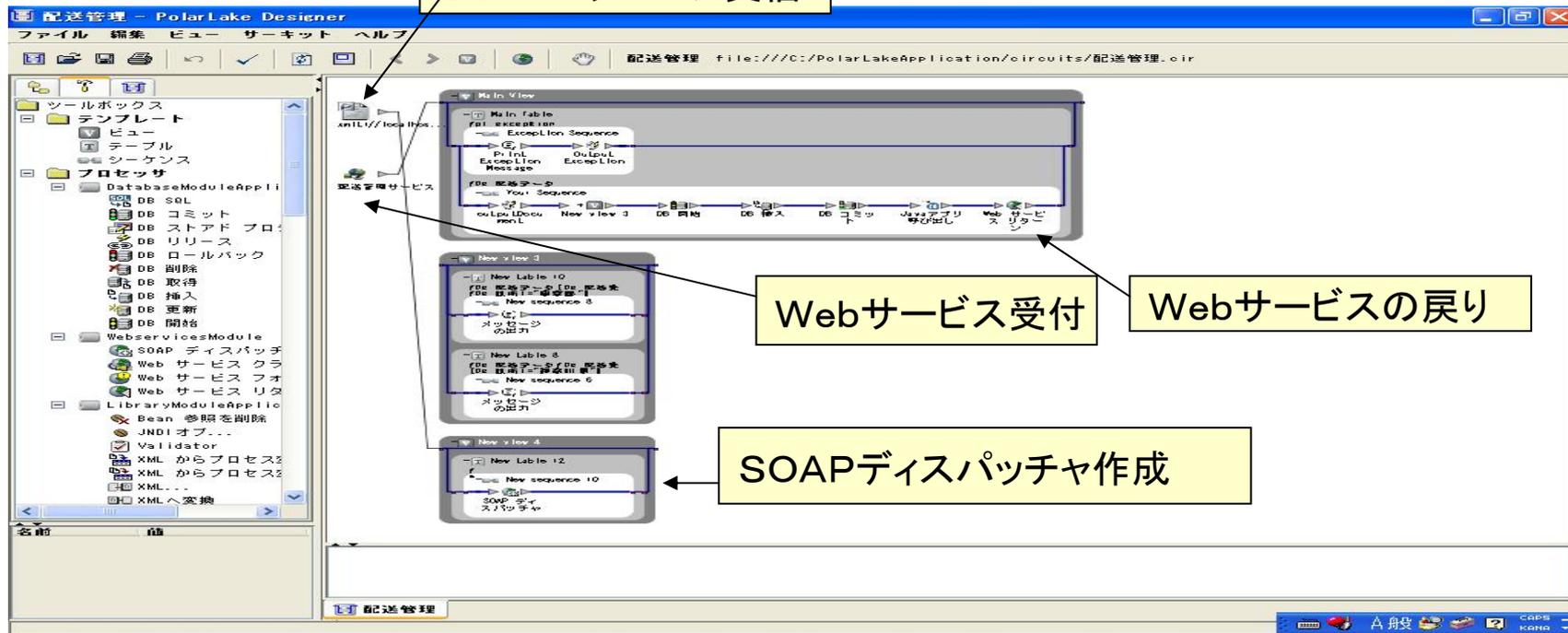
- ・作成済みの配送管理アプリケーションをWebサービス化して、新たに作成する受注管理サービスから呼び出す。

## ■ システム構成 & 処理プロセス



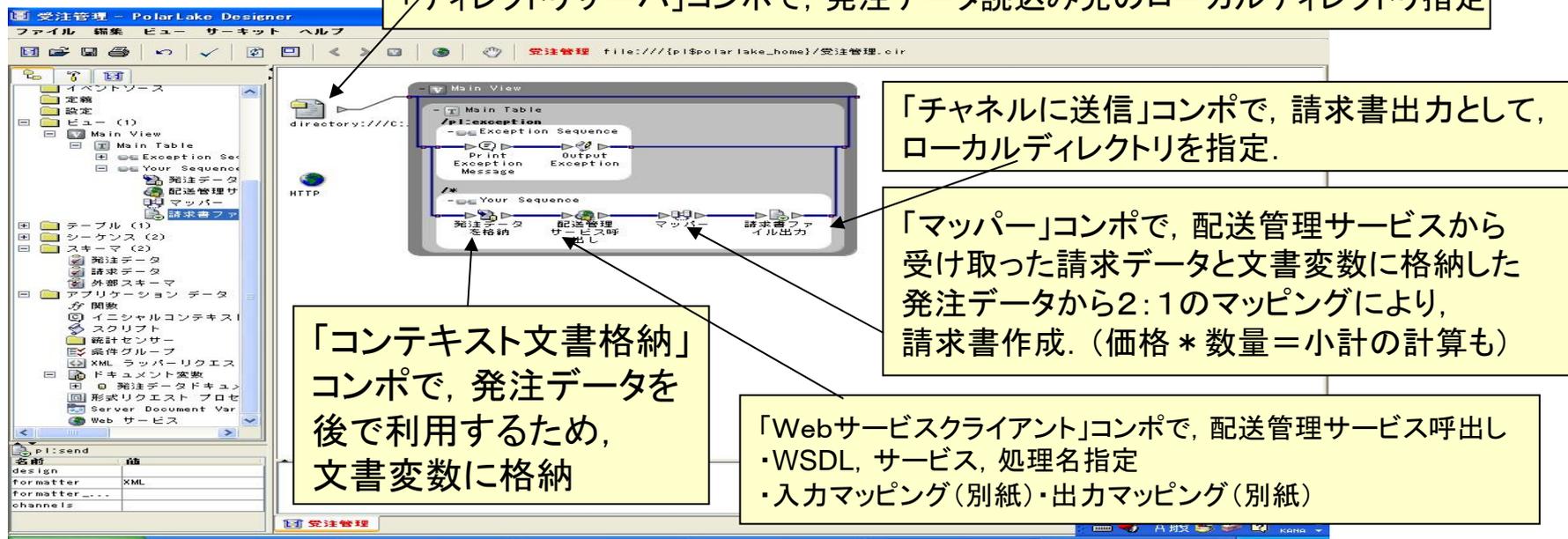
## ■ 配送管理サービス

## SOAPメッセージ受信

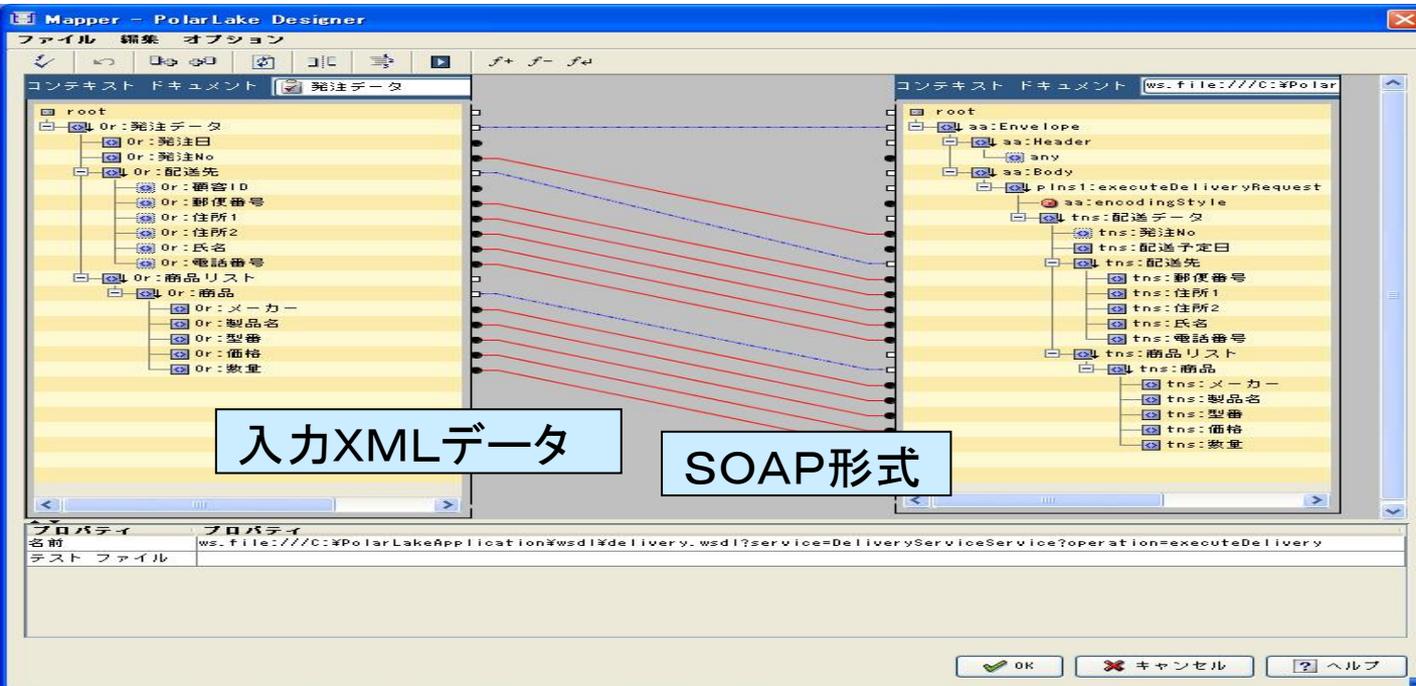


## ■ 受注管理サービス

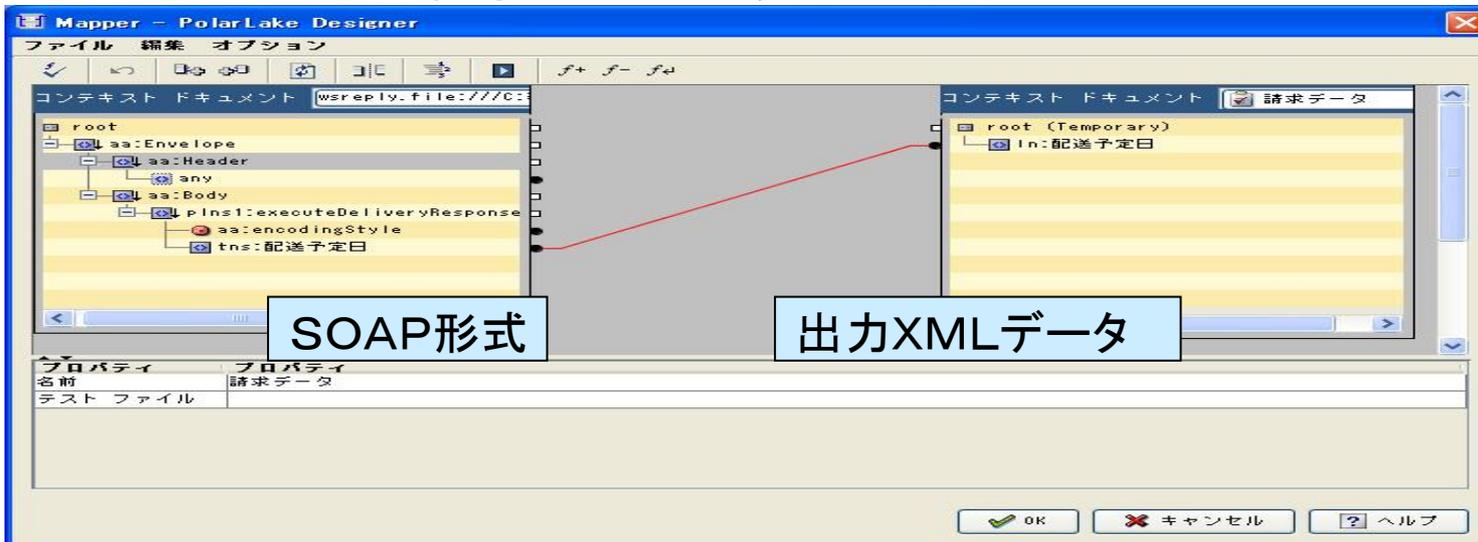
「ディレクトリサーバ」コンポで、発注データ読み込み先のローカルディレクトリ指定



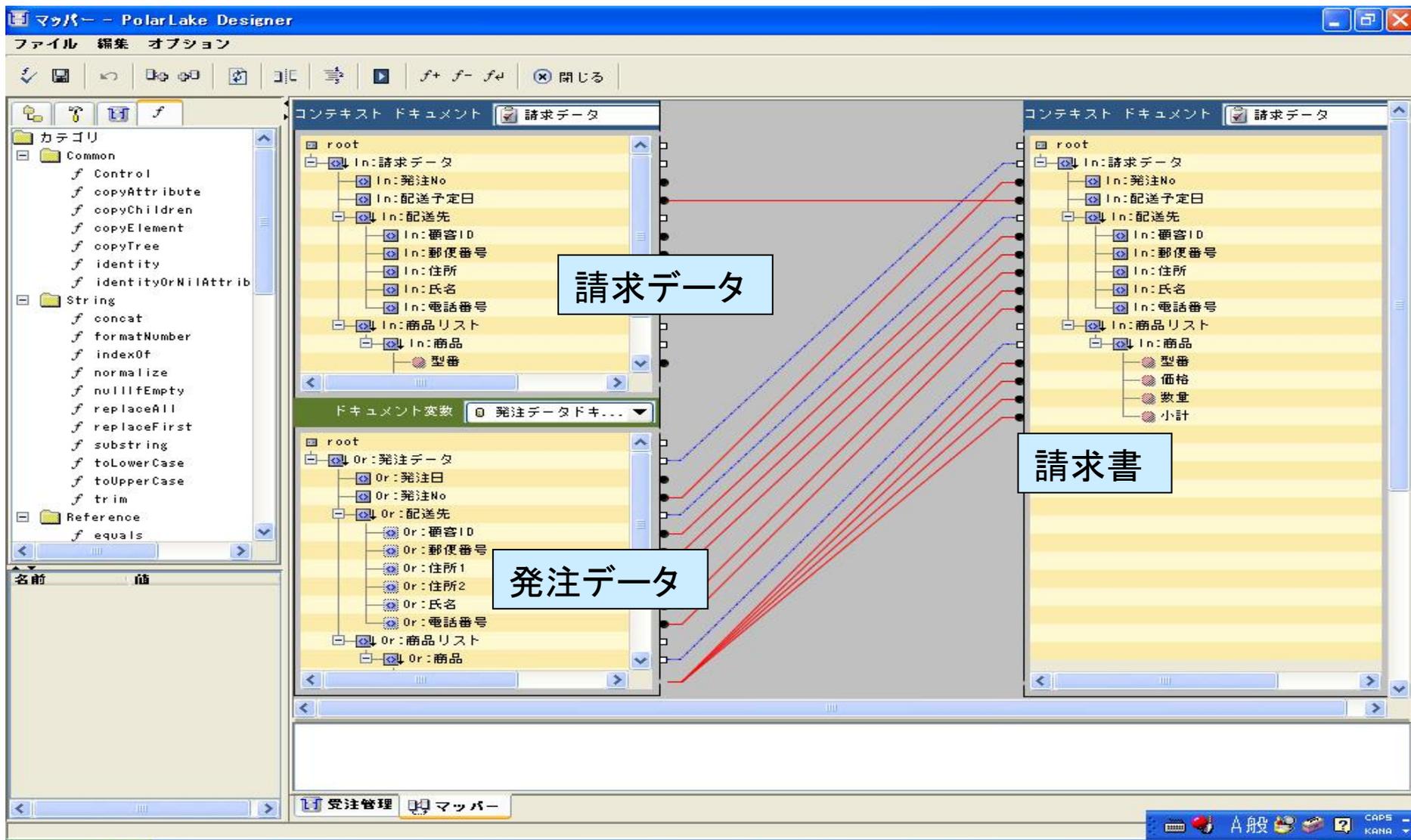
# ■ マッピング画面で発注データと配送管理サービスの入力への対応



# ■ 配送管理サービスの出力(SOAP)から請求データへの要素の対応



# ■ マッピング画面で請求データと発注データから請求書への対応



(注) 価格 \* 数量 = 小計を関数を用いて定義(ここでは省略)

# PolarLake: アプリケーションの実行

## ■ 入力の発注データ

```
<?xml version="1.0" encoding="Shift_JIS" ?>
<発注データ xmlns="http://www.hitachi-
system.co.jp/seminar">
  <発注日>2004/10/18</発注日>
  <発注No>No000777</発注No>
  <配送先>
    <顧客ID>10001</顧客ID>
    <郵便番号>108-8250</郵便番号>
    <住所1>東京都</住所1>
    <住所2>港区港南2-18-1</住所2>
    <氏名>日立太郎</氏名>
    <電話番号>03-6718-7777</電話番号>
  </配送先>
  <商品リスト>
  <商品>
    <メーカー>日立</メーカー>
    <製品名>Prius400</製品名>
    <型番>T100001</型番>
    <価格>198000</価格>
    <数量>2</数量>
  </商品>
  <商品>
    <メーカー>日立</メーカー>
    <製品名>Prius500</製品名>
    <型番>T100002</型番>
    <価格>218000</価格>
    <数量>1</数量>
  </商品>
</商品リスト>
</発注データ>
```

## ■ 出力の請求書

```
<?xml version="1.0" encoding="utf-8" ?>
<ln:請求データ xmlns:ln="http://www.hitachi-system.co.jp/seminar">
  <ln:発注No>No000777</ln:発注No>
  <ln:配送予定日 />
  <ln:配送先>
    <ln:顧客ID>10001</ln:顧客ID>
    <ln:郵便番号>108-8250</ln:郵便番号>
    <ln:住所>東京都</ln:住所>
    <ln:氏名>日立太郎</ln:氏名>
    <ln:電話番号>03-6718-7777</ln:電話番号>
  </ln:配送先>
  <ln:商品リスト>
    <ln:商品 型番="T100001" 価格="198000" 数量="2" 小計="198000" />
    <ln:商品 型番="T100002" 価格="218000" 数量="1" 小計="218000" />
  </ln:商品リスト>
</ln:請求データ>
```

(注) 結線ミスで、小計の計算ミス