

業務の知識を有するエンドユーザ主導の  
アプリケーション開発技法  
—フレームワーク・ドメインモデル・サービス連携—

2007年 11月 19日(月)

中所武司

明治大学 理工学部 情報科学科

chusho@cs.meiji.ac.jp

<http://www.chusho.jp/>

# 技報の構成

---

1. はじめに
2. 研究の背景
3. 従来 of 技術
4. フレームワークを用いた開発方法
5. モデルベースの開発方法
6. Webサービス連携を用いる開発方法
7. おわりに

# 1. はじめに

---

## ◆ 動向：ソフトウェアのサービス化

- \* インターネットの普及 → Webアプリケーションが増大
- \* ASP (Application Service Provider) → Webサービス
  - SOA (Service-Oriented Architecture)
  - SaaS (Software as a Service)

## ◆ 変化の激しい時代の要請：

### エンドユーザ主導のAP開発・保守

- \* 対象：小さな部門や個人の業務のWebアプリケーション
- \* 特徴：低コストで短期開発，頻繁な機能変更

## 2. 研究の背景



OA化 → ゆとりのない日常生活 → ITによる豊かな生活の実現 (CS-life)

- ◆ 情報処理学会 ソフトウェア工学研究会 1994. 3  
「wwHww:分散オフィスシステムのための  
エンドユーザコンピューティング向きオブジェクト指向モデル」
- ◆ 日本ソフトウェア科学会 コンピュータソフトウェア 1994. 11  
巻頭言 「CS-life」

- ・CS-life vs. CSCW (Computer-Supported Cooperative Work)
- ・生産者中心の視点から利用者中心の視点への転換が必要
- ・仕事の効率より生活を豊かにすることにもっと知恵を絞るべき

# ITによる豊かな生活の実現

## 国の政策

1994 CS-life 「コンピュータによる豊かな生活の実現」

2001 e-Japan 戦略

「すべての国民が情報通信技術(IT)を積極的に活用し、  
かつその恩恵を最大限に享受できる**知識創発型社会の実現**」

2006 IT新改革戦略

「いつでも、どこでも、誰でもITの恩恵を実感できる**社会の実現**」

「使い勝手が利用者の視点に立ったものとなっていない等の理由から、国民・企業等による**電子政府**の利用は進んでおらず、また、住民サービスに直結する**地方公共団体の電子化**が十分ではないなど、国民・企業等利用者が利便性・サービスの向上を実感できていない。

→ (例)パスポートの電子申請

利便さ ≠ 豊かさ

# 豊かな生活の実現手段 '94

エンドユーザ主導開発

PHILOSOPHY: コンピュータによる豊かな生活の実現



POLICY: すべての日常的な業務をコンピュータ化する



★ ルーチンワークからの開放



★ 自由な時間の獲得

\* オフィス: 創造的な仕事

\* 家庭 : 個性的な余暇



業務の専門家が自らの業務を自らコンピュータ化



エンドユーザ主導開発のための情報システム構築技術

# 3. 従来の技術

## エンドユーザ主導開発の位置づけ'91

表-2 ソフトウェア産業の進化過程

ソフト産業の形態	主要な技術職	主要技術
労働集約型産業	プログラマ	自動化 (CASE)
知識集約型産業	設計者	標準化 (パッケージ)
知恵集約型産業	業務専門家	エンドユーザコンピューティング


表-3 プログラミング言語の発展過程

時期	目的	内容
1960年代	量的高級化	記述水準 (機械語への展開率) の向上
1970年代	質的高级化	プログラミング方法論 (構造化技法) の導入
1980年代	パラダイム転換	宣言的記述 (手続き型から非手続き型へ)
1990年代	エンドユーザコンピューティング	脱プログラミング

情報処理 1991年8月  
 エンドユーザコンピューティング  
 -ソフトウェア危機回避のシナリオ

# エンドユーザ主導の開発技術

## －脱プログラミングのミニ歴史－

- 
- 1989** 情報処理 特集「新しいプログラミング環境」  
・視覚的プログラミング環境
- 1991** 情報処理 「エンドユーザコンピューティング」  
・候補: 4GL, 日本語プログラミング,  
ビジュアルプログラミング, AI(ルール, ファジー, ニューロ)
- 1998** コンポーネントウェア(共立出版)  
・部品の組み合わせによるシステム構築ツール  
APPGALLERY, HOLON/VP, IntelligentPad, VisualAge
- 2000** Programming by example 特集(CACM)  
・操作例からユーザの意図を推測し, 類似の操作を自動化
- 2004** End-user development 特集(CACM)  
・IT技術者の支援なしで, ユーザ自身の問題解決ソフトを開発
- 最近** Web2.0, マッシュアップ



# 1989 情報処理 「視覚的プログラミング環境」

## ◆ 概要

ソフトウェア設計やプログラム作成のための視覚的表記を用いる視覚的言語

## ◆ 分類

・グラフ指向言語 ・フォーム指向言語 ・アイコン指向言語

## ◆ 支援対象

- ・データの視覚化
- ・プログラムとその実行の視覚化
- ・上流工程の設計の成果物の視覚化
- ・プログラミング工程の視覚化 ← 脱プログラミングに近い

# 1991 情報処理 「エンドユーザコンピューティング」

- ◆ **4GL** (James Martinの定義の一部)
  - ・非職業的プログラマが使用可能
  - ・非手続的記述形式
  - ・記述量と作成工数がCOBOLより1桁少ない
- ◆ **日本語プログラミング** (課題)
  - ・業務の用語辞書の作成保守機能
  - ・プログラミング言語の知識を不要とする開発保守環境
  - ・あいまい性を避けながら, 自然な表現を保持した文法規則
- ◆ **ビジュアルプログラミング**
  - ・視覚的言語の図的表現形式
- ◆ **AI(ルール, ファジー, ニューロ)**
  - ・簡易プログラミングという観点からの**エキスパートシステム構築ツール**<sub>10</sub>

# 1998 コンポーネントウェア(共立出版)

---

## ◆部品の組み合わせによるシステム構築ツール

### APPGALLERY(日立)

ホームページの公開は、**2003**年11月30日をもって終了

### HOLON/VP(日電)

ホームページ見当たらず

### IntelligentPad(富士通)

販売終了(**2003**年3月末以前) サポート終了(2007年3月31日)

### VisualAge(IBM)

VisualAge for Java End of Support (31 Dec **2003**)

# エンドユーザ主導の開発技術

## － 最近の実例 －

### 長崎県庁：電子県庁システム

出典：「職員が自ら詳細仕様を設計し、IT調達コストを大幅ダウン」

日経コンピュータ 05.7.25

「IT調達改革の星」朝日新聞 07.7.27

#### <役割分担>

- ・職員：画面設計と業務要件決定
- ・ベンダー：DB, アプリケーション設計

1. 職員：画面のラフスケッチ → Web画面開発外注  
業務要件漏れや部門間の誤解の防止
2. DBのテーブル・フォーマット作成  
職員が参画し、SEが作成
3. 詳細仕様書完成  
SEが作成した仕様書に基づき、入札実施

(cf.) 実践的アプローチに基づく要求仕様の発注者ビュー検討会  
→「画面」、「システム振る舞い」、「データモデル」

# ながさきITモデルの概要

手段	狙い	効果
<ul style="list-style-type: none"><li>・職員自らが 詳細な仕様書作成</li><li>・500万円以下に 分割発注</li><li>・システムでの 利用技術を あらかじめ定義</li></ul>	<ul style="list-style-type: none"><li>・ニーズにあった仕様</li><li>・あいまいな仕様 に基づくシステムの リスク回避</li><li>・単価が安い地元 ベンダーの参加</li></ul>	<ul style="list-style-type: none"><li>・コスト削減 (32億円→16億円)</li><li>・満足度向上 (<b>利用者の要望の 迅速な反映</b>)</li><li>・地元のIT産業 振興 (04年度:96件中 73件受注)</li></ul>

# エンドユーザ主導の開発技術

## 情報サービス産業の中での位置づけ

### ■ ITプロフェショナルによる開発

- ・高付加価値の受注ソフトウェアで、高収益の案件
- ・高付加価値の定型業務パッケージ

### ■ エンドユーザ主導開発

- ・投資対効果の不明確な分野
- ・非定型業務で、ユーザ限定の分野  
→ 低コスト, 短期開発, 継続的保守

住み分け  
(競合しない)

- ◆ エンドユーザ主導開発ツールの新市場は大きい。
  - \* ITプロフェショナル向けのツールは市場が小さい。  
しかも、フリーソフトが増加。
  - \* エンドユーザ向けのツールは、市場が大きい。

# エンドユーザ主導開発の研究

## 明大中研の場合

- 1994 分散オフィスシステムのためのEUC向きオブジェクト指向モデル  
- 情報処理学会ソフトウェア工学研究会 (Mar.1994)

wwHww: 窓口業務の**アプリケーションフレームワーク**  
(UI駆動型開発)

- 1995 「ドメインモデル≒計算モデル」を志向した  
アプリケーションソフトウェア開発環境の基本概念、  
- 情報処理学会ソフトウェア工学研究会、(May.1995)

M-base: **モデリング & シミュレーションツール**  
(モデル駆動型開発)

- 2001 絶えざる変化に対応するエンドユーザ主導のサービス連携、  
- 産学戦略的ソフトウェア研究フォーラム、(Apr. 2001).

Webサービスによる統合 (**Webサービス連携**)  
(フォーム駆動型開発)

## 4. フレームワークを用いた開発方法

### ◆ フレームワーク

- ・ ソフトウェアアーキテクチャの基本的な枠組みと,  
その枠組みを構成するクラスライブラリを備えたもの
- ・ 特定業務に特化
  - 既に業務の知識が業務コンポーネントとして用意され,  
カスタマイズは比較的容易
- ・ 特定の情報処理内容に特化
  - + フロントエンド系 (Struts など)
  - + バックエンド系 (Hibernate など)
  - 業務処理 (業務ロジック) を付加する必要があるが,  
業務に横断的に適用可能
- ・ 個々のアプリケーションは, フレームワークをカスタマイズして作成
  - 短期開発と低コストと高品質を実現

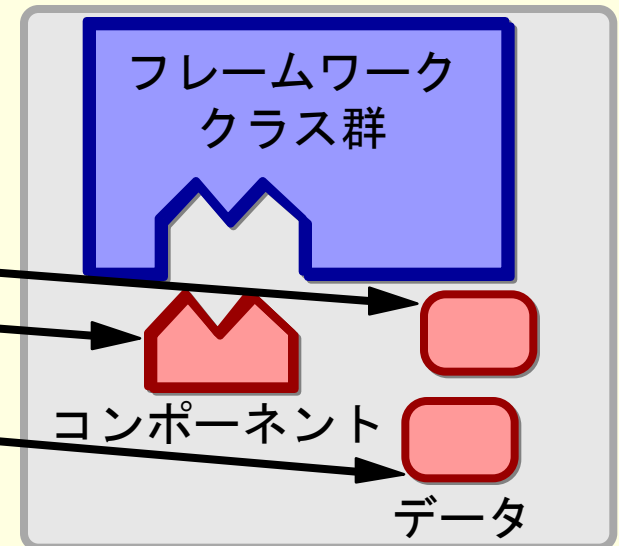
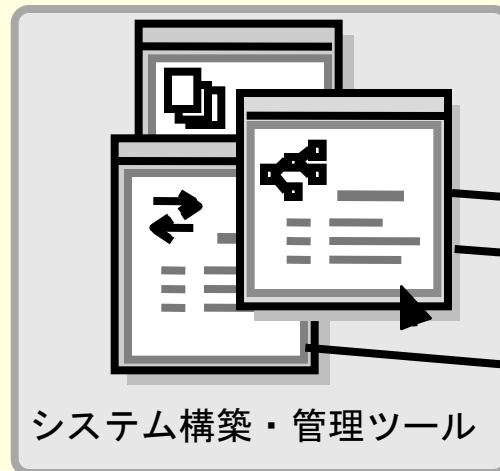
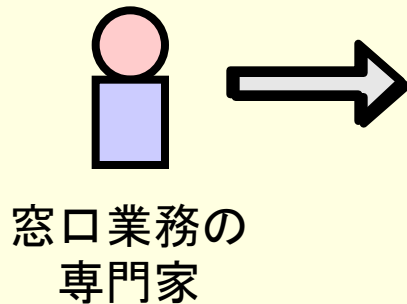


# アプリケーションフレームワーク

業務の専門家による構築手順

1. サービスの定義
2. フォームの作成
3. 処理方式の設定
4. サーバへの登録

- ★固定部分(再利用)
- ★新規作成
  - ・差分コンポーネントのプラグイン
  - ・プロパティデータの設定



# 窓口業務フレームワーク

## ◆ 窓口業務の課題

担当者は、毎日同じような質問を受け、ルーチンワーク化した判断処理をしている。

## ◆ 解決方法

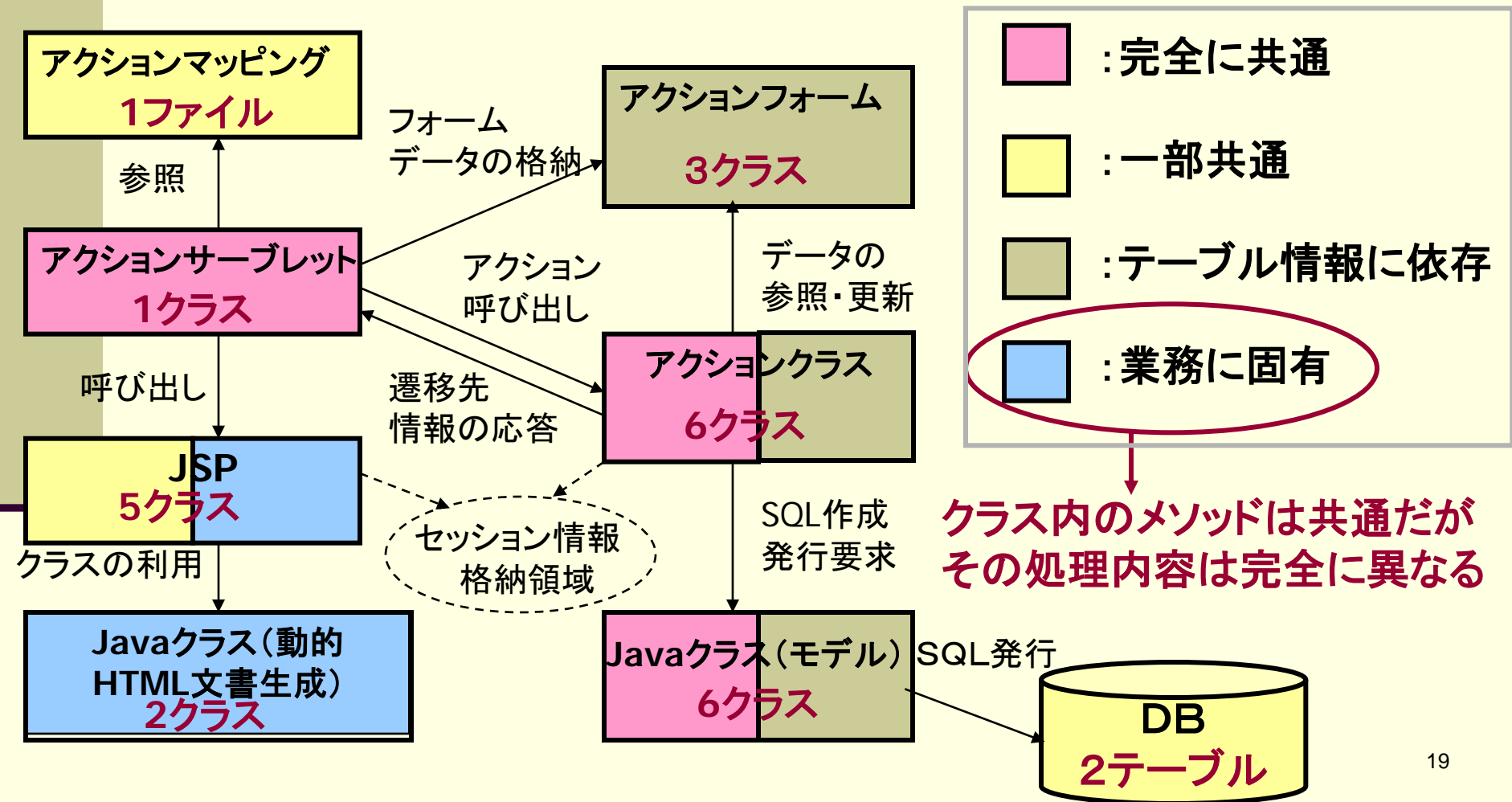
ルーティンワークのコンピュータ化によるサービスの向上や費用の低減

## ◆ 波及効果

急速な高齢化社会の到来と若年労働者の減少への対応

# 試作した予約業務フレームワーク

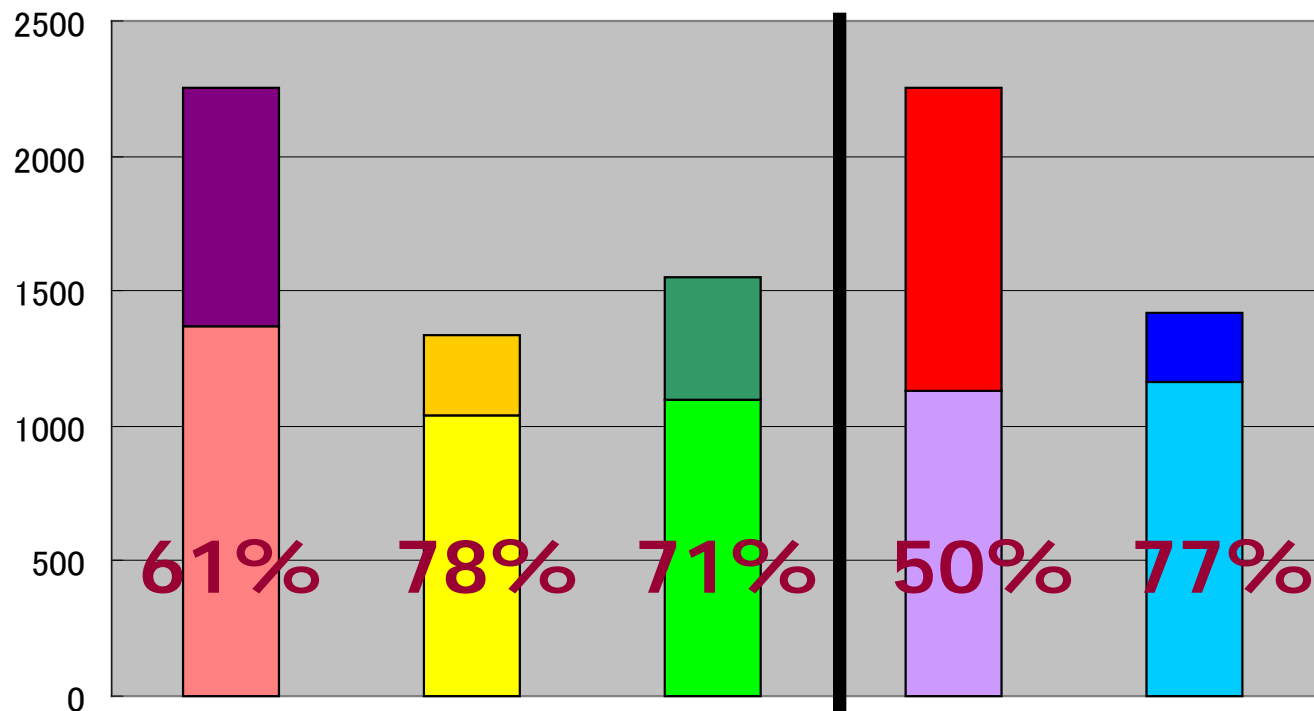
## 4つのレベルによる分類



# 適用結果

様々な予約業務に対して幅広く適用可能

ステップ



会議室

図書

商品

会議室

座席

(ツール適用前)

# アプリケーションフレームワーク

## 開発プロセス上の課題

- ◆ UI定義: UI遷移を含み, 要求定義は可.
  - \* UI遷移→UI定義
  - \* 実装→JSP(SEによる実装支援)
  
- ◆ モデル定義: MVCフレームワークでほぼ決定.
  - \* 実装→フレームワーク化  
(SEによる固有のビジネスロジックの実装支援)
  
- ◆ DB定義: 設計・実装上の課題
  - \* テーブル設計
  - \* 実装→ツール化

## 5. モデルベースの開発方法

### ■ 対象ソフトウェア

オフィスなどにおける小規模な業務アプリケーション

### ■ 開発・保守形態

開発はSEの支援を受けるが保守はエンドユーザのみ  
\* 稼働後の機能変更(モデル変更)への対応

### ■ 基本コンセプト

「ドメインモデル≒計算モデル」

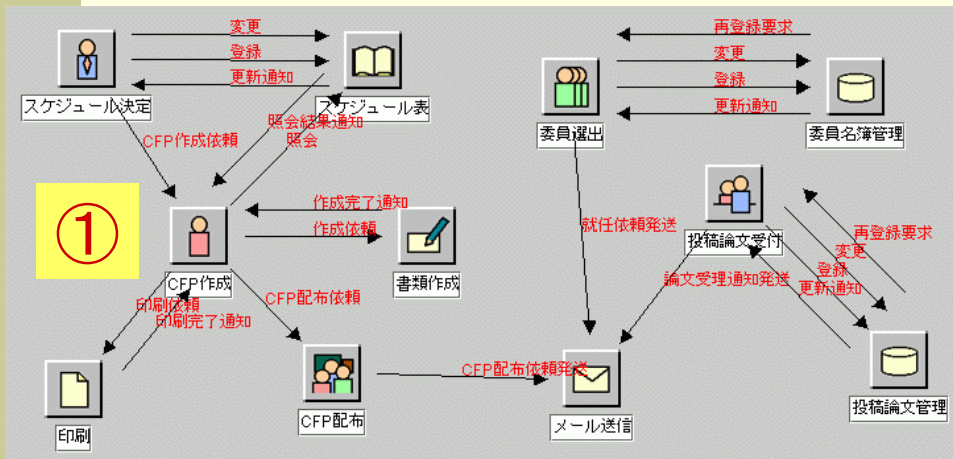
「分析≒設計≒プログラミング」

「ソフト開発=モデリング+シミュレーション」

アジャイル開発  
反復開発

# モデリング & シミュレーション 業務の専門家による構築手順

1. ドメインモデル
2. ルール記述
3. UI自動生成
4. シミュレーション



### ルール追加

優先度

条件部  
作業継続 == true  
印刷の有無 == true

実行部  
印刷依頼

### UIの確認

時  分

主催

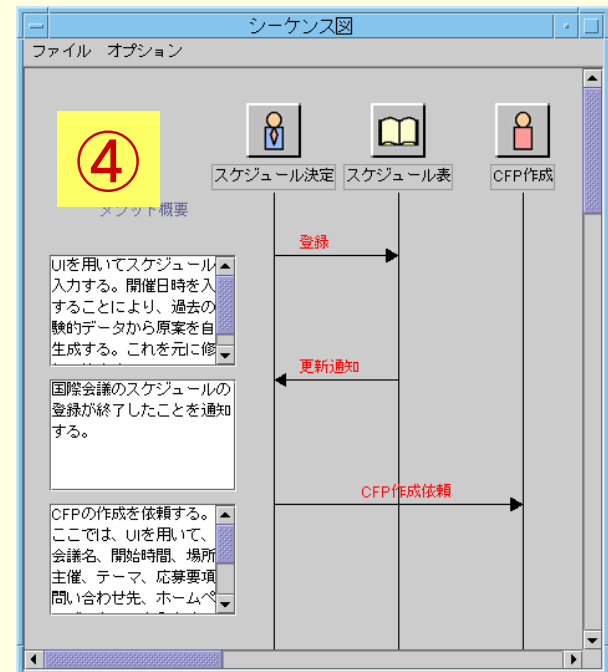
テーマ

応募要項

問い合わせ先

ホームページアドレス

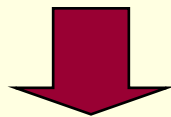
参加費



# モデリング & シミュレーション

## 開発プロセス上の課題

- ◆コンポーネントの事前の品揃えは無理.  
非定形的な業務では新規コンポ要の可能性大
- ◆必要になったら発注  
エンドユーザにコンポーネントの要求仕様書  
が書けるか？（実装技術の知識が必要）

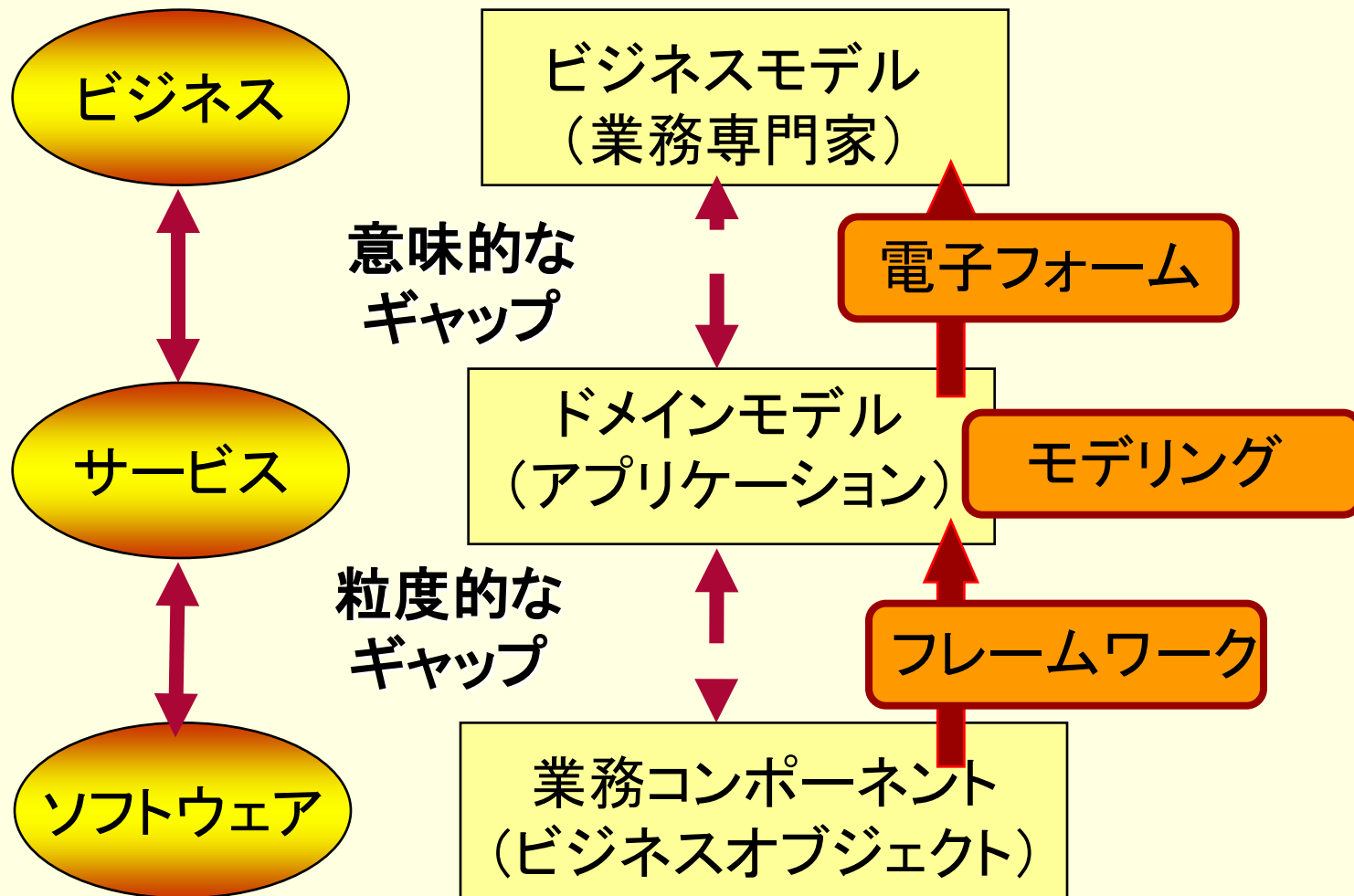


- ◆Webサービス化  
インタフェースが簡単で、明確な要求仕様可.  
フォーム定義主体（ここからUI駆動型に回帰）



# 6. Webサービス連携を用いる開発方法

## エンドユーザ主導のシステム構築技術



# Webサービスによる統合

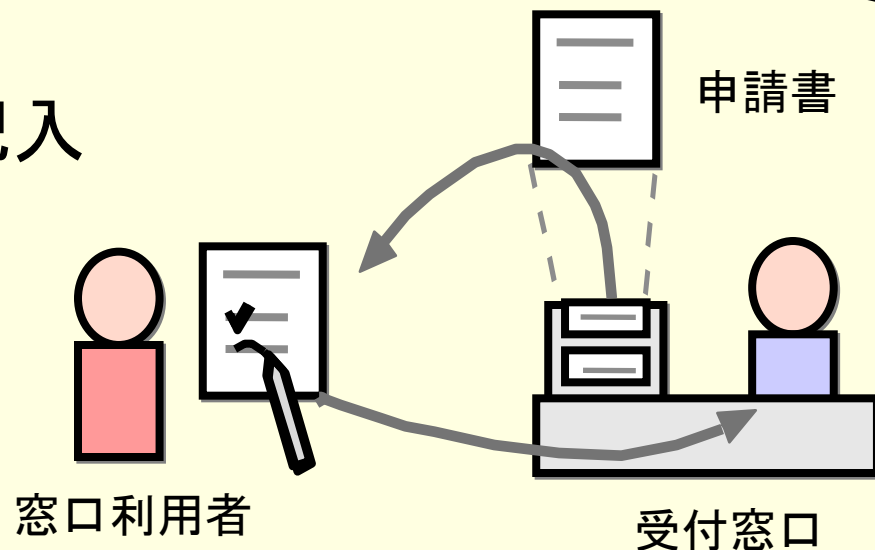
## サービス授受のメタファー

### 「サービス授受＝窓口業務」

＜インタフェース：フォーム＞

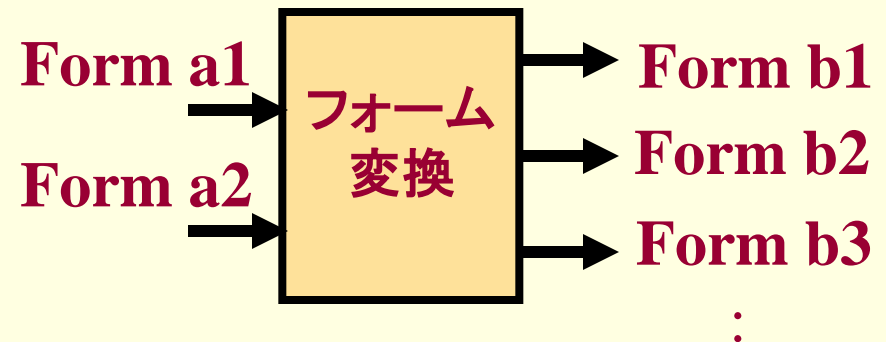
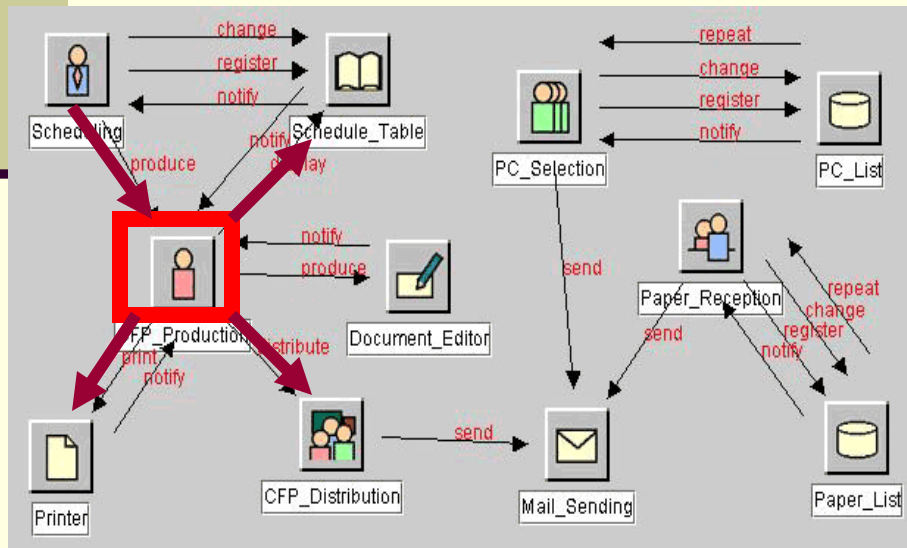
エンドユーザ  
(業務の専門家)  
に理解容易

1. 受付窓口の検索
2. フォームの取り寄せと記入
3. 受付窓口への提出



# Webサービスによる統合 フォーム変換

- <ワークフロー> → < Webサービス >
- ・ 業務コンポーネント → Web サービス
  - ・ メッセージフロー → フォームフロー
  - ・ メッセージ変換 → フォーム変換



# 方式1: XSLTを用いたフォーム変換

## ■ 概要

- フォームをXML化し, XSLTで変換
- Webサービス連携

## ■ 例題

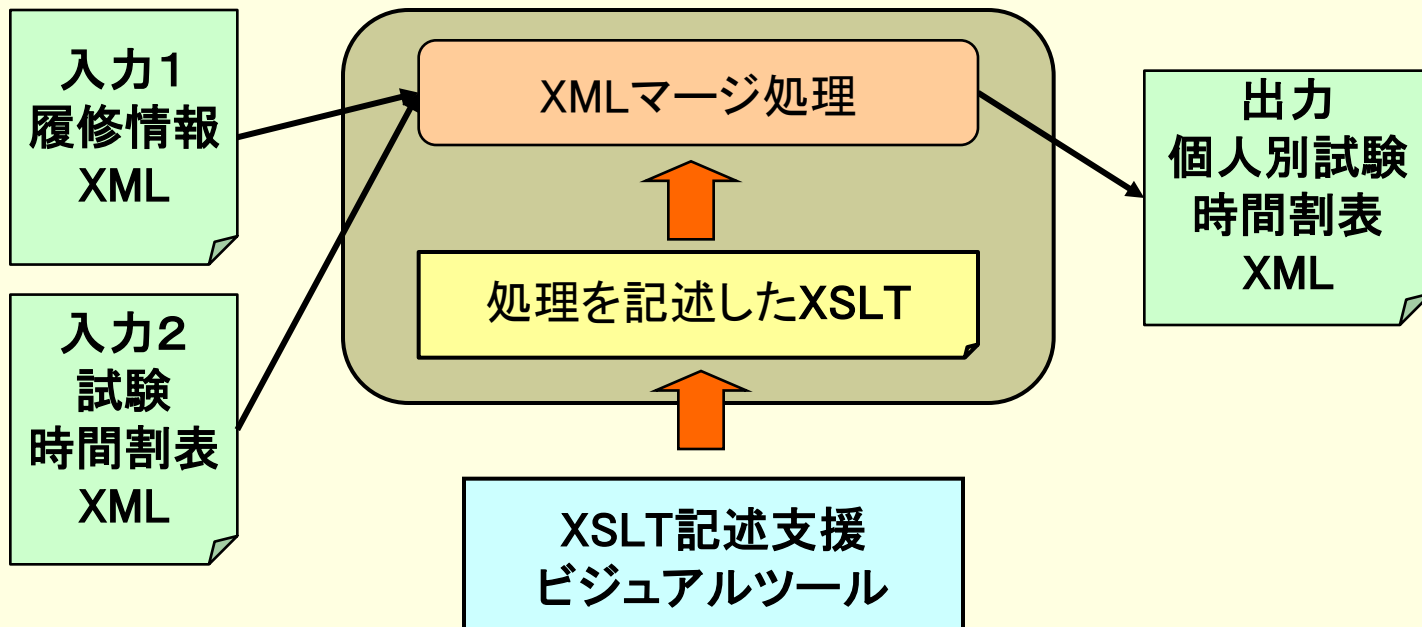
- 例題1: 個人別時間割表作成システム
- 例題2: 卒業条件判定システム

## ■ 課題

- エンドユーザ(業務の専門家)主導の  
XSLT作成

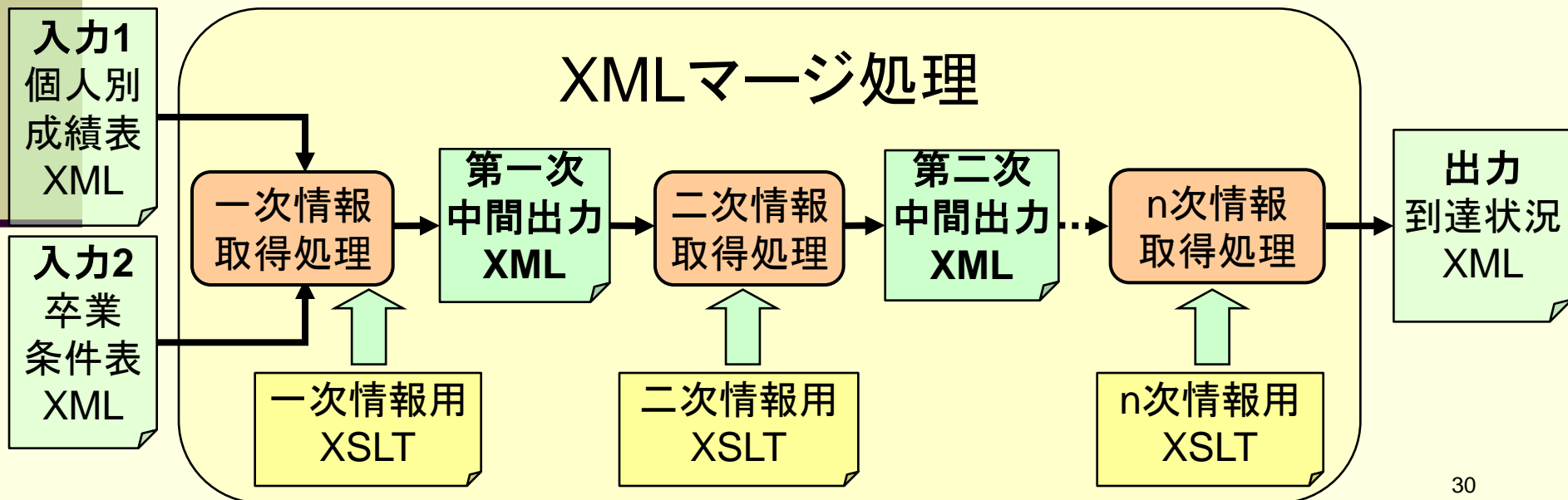
# 基本的なマージ処理を必要とする例題

- 個人別試験時間割表システム
  - 各学生に特化した個人別試験時間割表を提供する



# 複雑なマージ処理のスキ립ト化

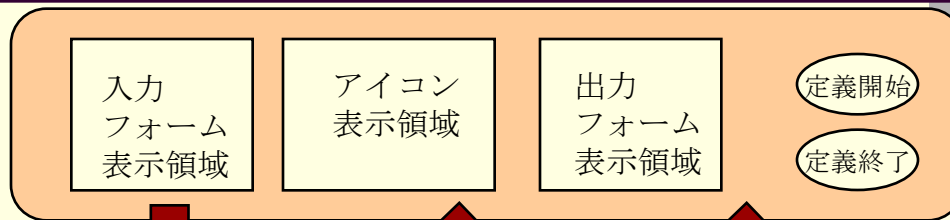
- マージ処理の多段階化
  - 回数ごとにXSLTを用意し、中間出力XMLを出力



## 方式2:ビジュアルツールを用いた フォーム変換定義

- 概要
  - ビジュアルツール
  - エンドユーザにXML, XSLTを見せない.
- 例題
  - ホテル予約システム
    - \* 入力: 宿泊申込書, レンタカー申込書
    - \* 出力: 請求書, 領収書

# ビジュアルツールを用いた フォーム変換定義方法



宿泊申込書

ホテル名

料金  円

人数  人

レンタカー申込書

会社名

保険料  円

レンタル料  円

台数  台

初期化

( )

+ -

\* /

x

y

z

請求書

ホテル名

宿泊料  円

会社名

レンタル料 (保険込み)  円

合計請求金額  円

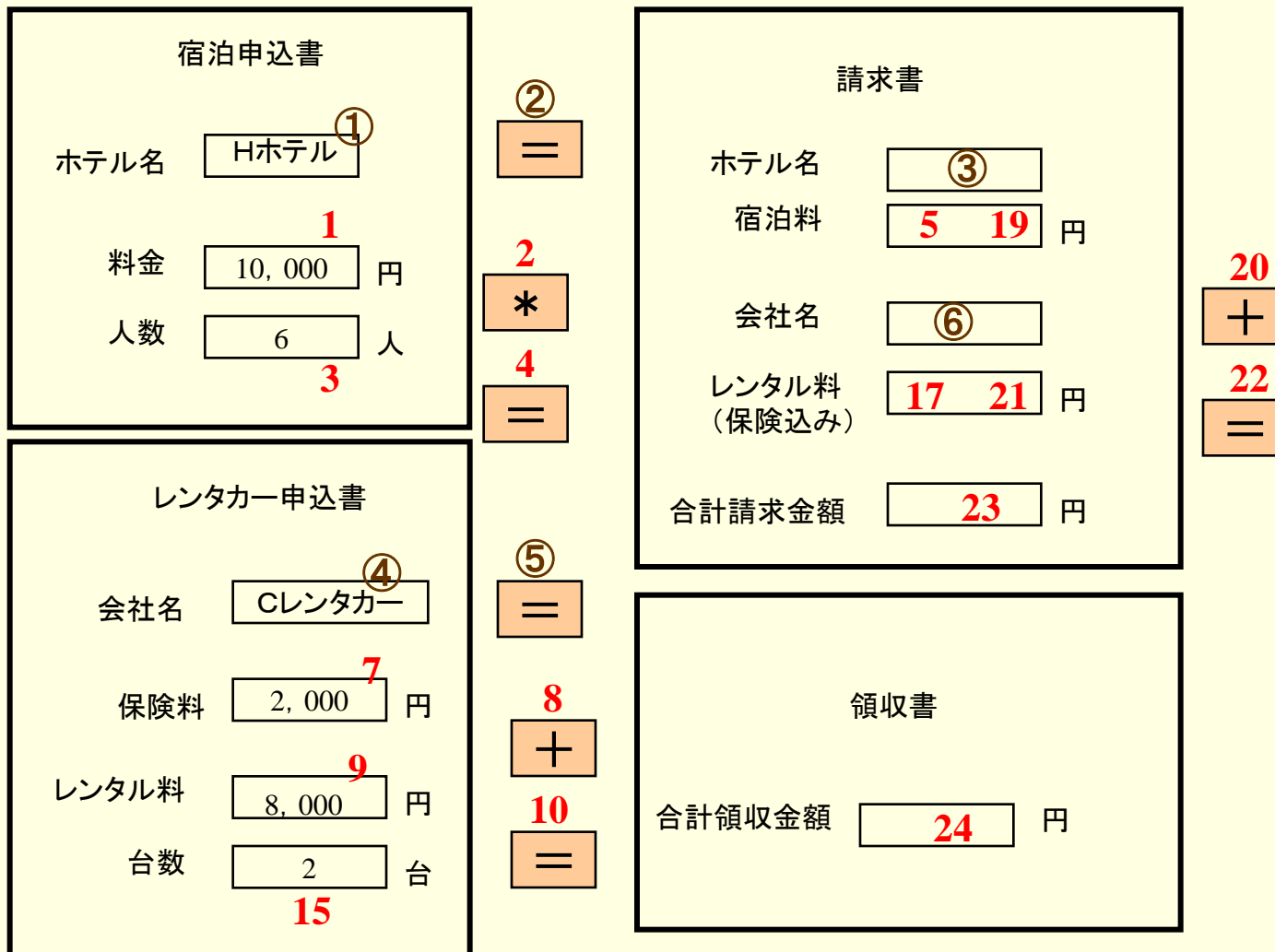
領収書

合計領収金額  円

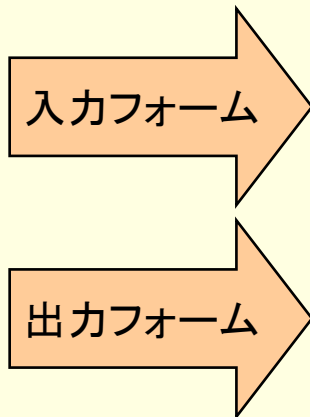


# 変換手順の定義の操作例

変換手順の定義の操作例



# フォーム変換定義の例



C:\Documents and Settings\chusho\Local Settings\Temp\def.zip の一時ディレクトリ 1...

ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

戻る 検索 お気に入り 移動 リンク

アドレス(C) C:\Documents and Settings\chusho\Local Settings\Temp\def.zip の一時ディレ

**宿泊申込書**

ホテル名

料金  円

人数  人

---

**請求書**

ホテル名

宿泊料

削除

初期化

( )

+ -

\* /

x y

z =

ページが表示されました

マイ コンピュータ

入力フォーム

出力フォーム

変換手順

C:\Documents and Settings\chusho\Local Settings\Temp\def.zip の一時ディレクトリ 1...

ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)

戻る 検索 お気に入り 移動 リンク

アドレス(D) and Settings\chusho\Local Settings\Temp\def.zip の一時ディレクトリ 1\def.html

### 宿泊申込書

ホテル名

料金  円

人数  人

---

### 請求書

ホテル名

宿泊料

削除

初期化

( )

+ -

\* /

x y

z =

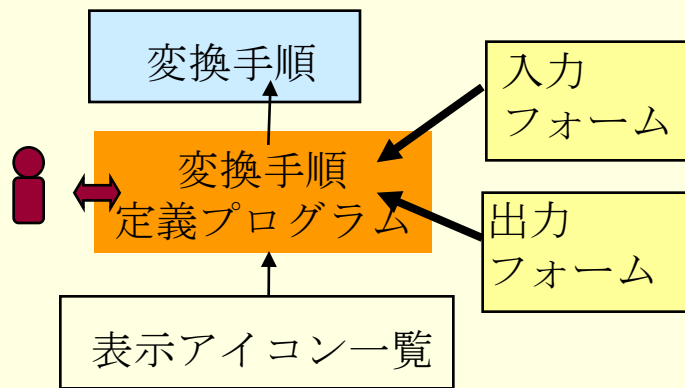
番号	内容
1	宿泊申込書.ホテル名
2	EQUAL
3	請求書.ホテル名
4	INIT
5	宿泊申込書.料金
6	MULT
7	宿泊申込書.人数
8	EQUAL
9	請求書.宿泊料

ページが表示されました

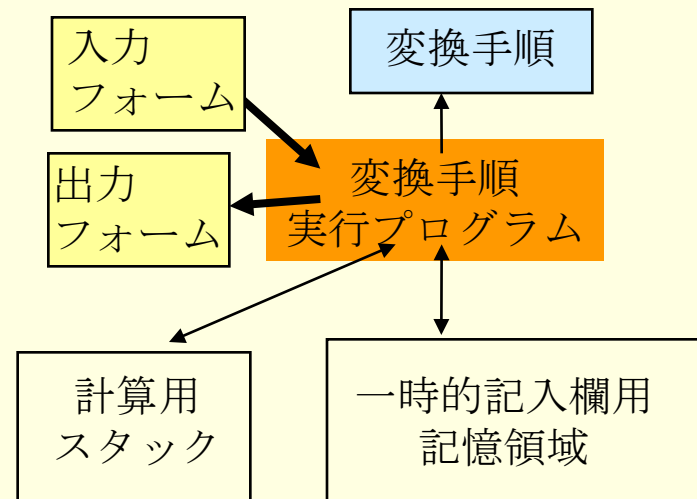
マイ コンピュータ

# 変換手順の定義・実行方式

## 変換手順の定義方式



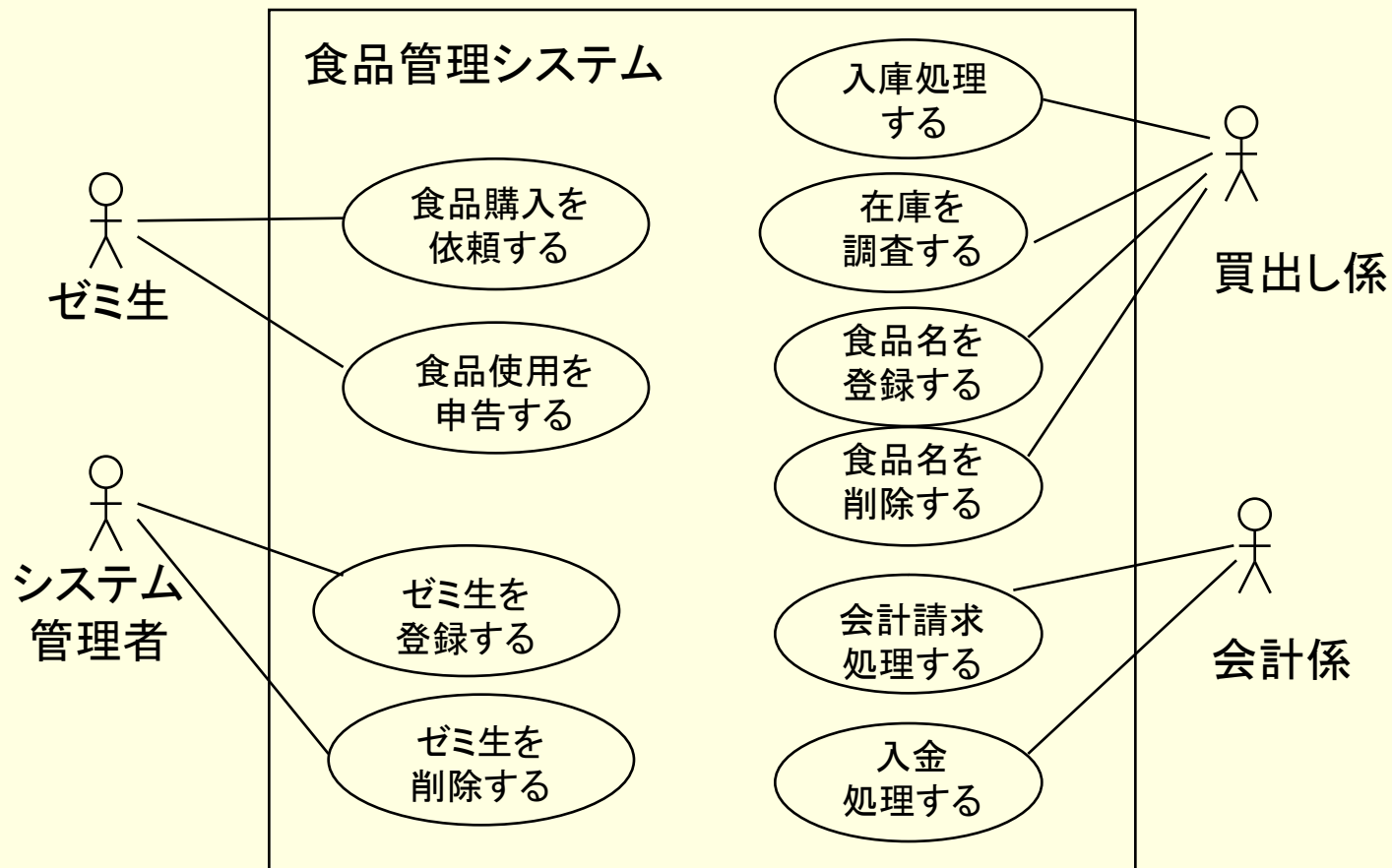
## 変換手順の実行方式



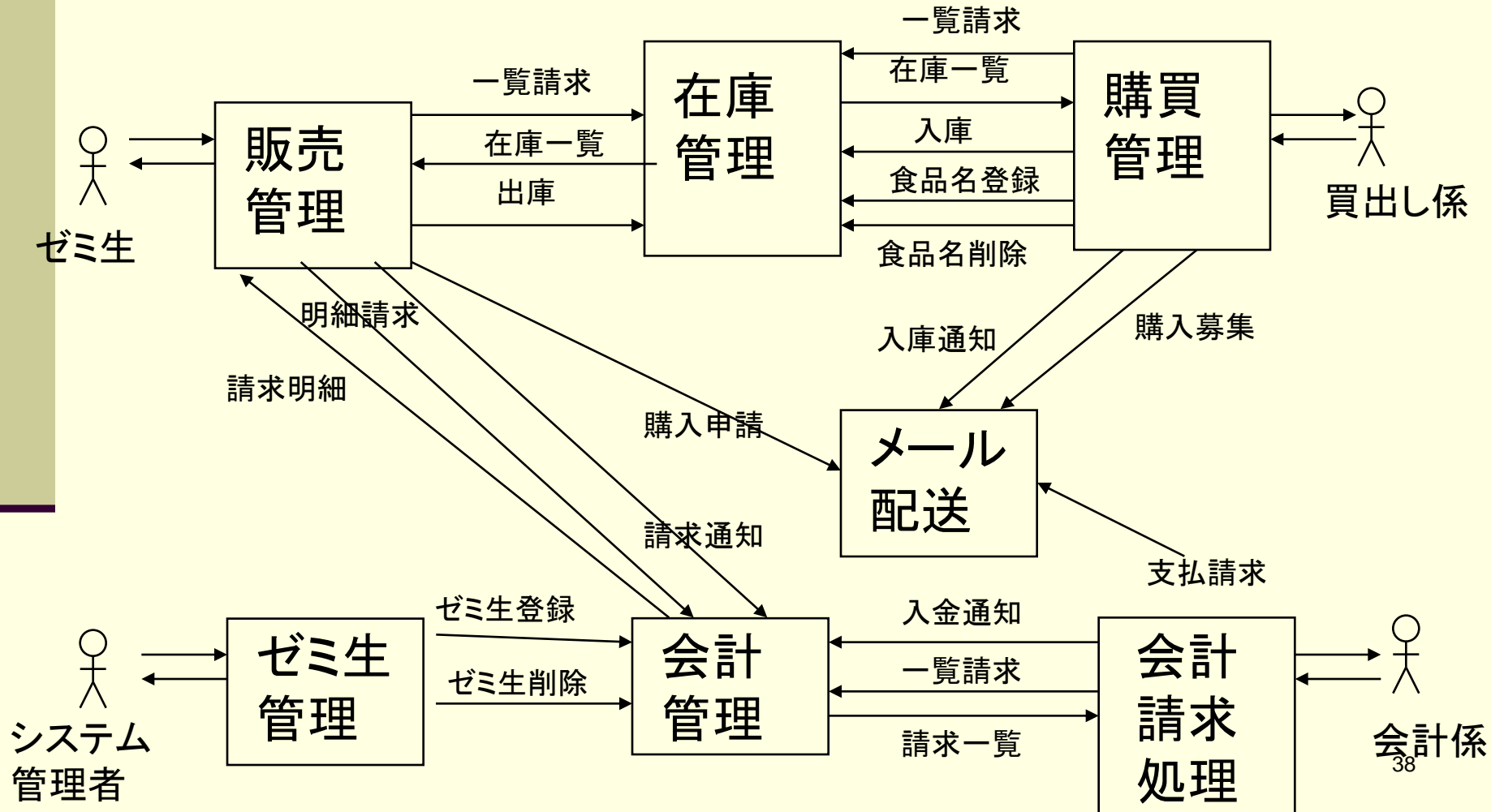
- ◆ 課題: ビジネスロジックの記述機能の拡張

# 実用的な例題： 食品管理システム (FOOD)

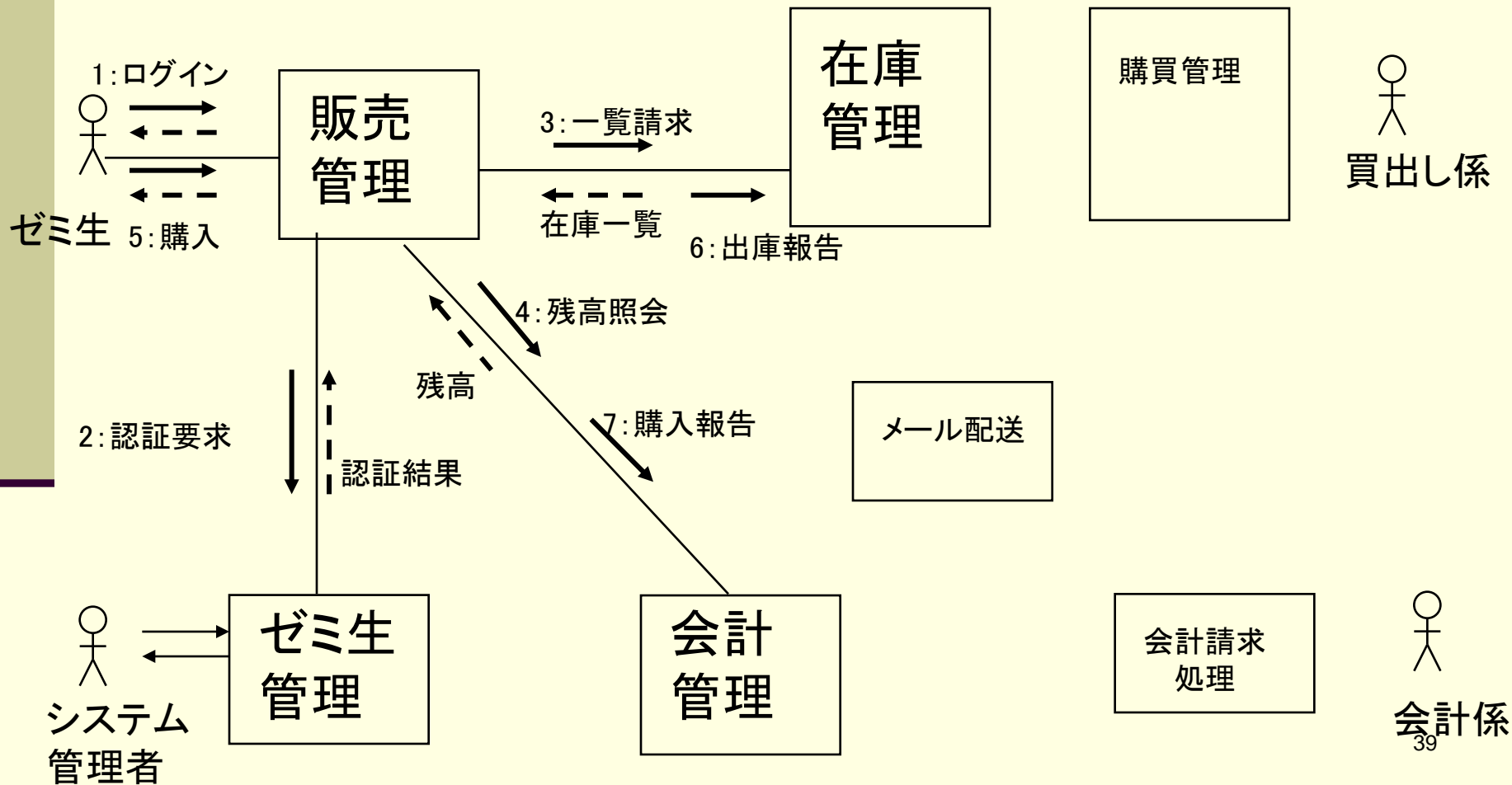
## ユースケース図



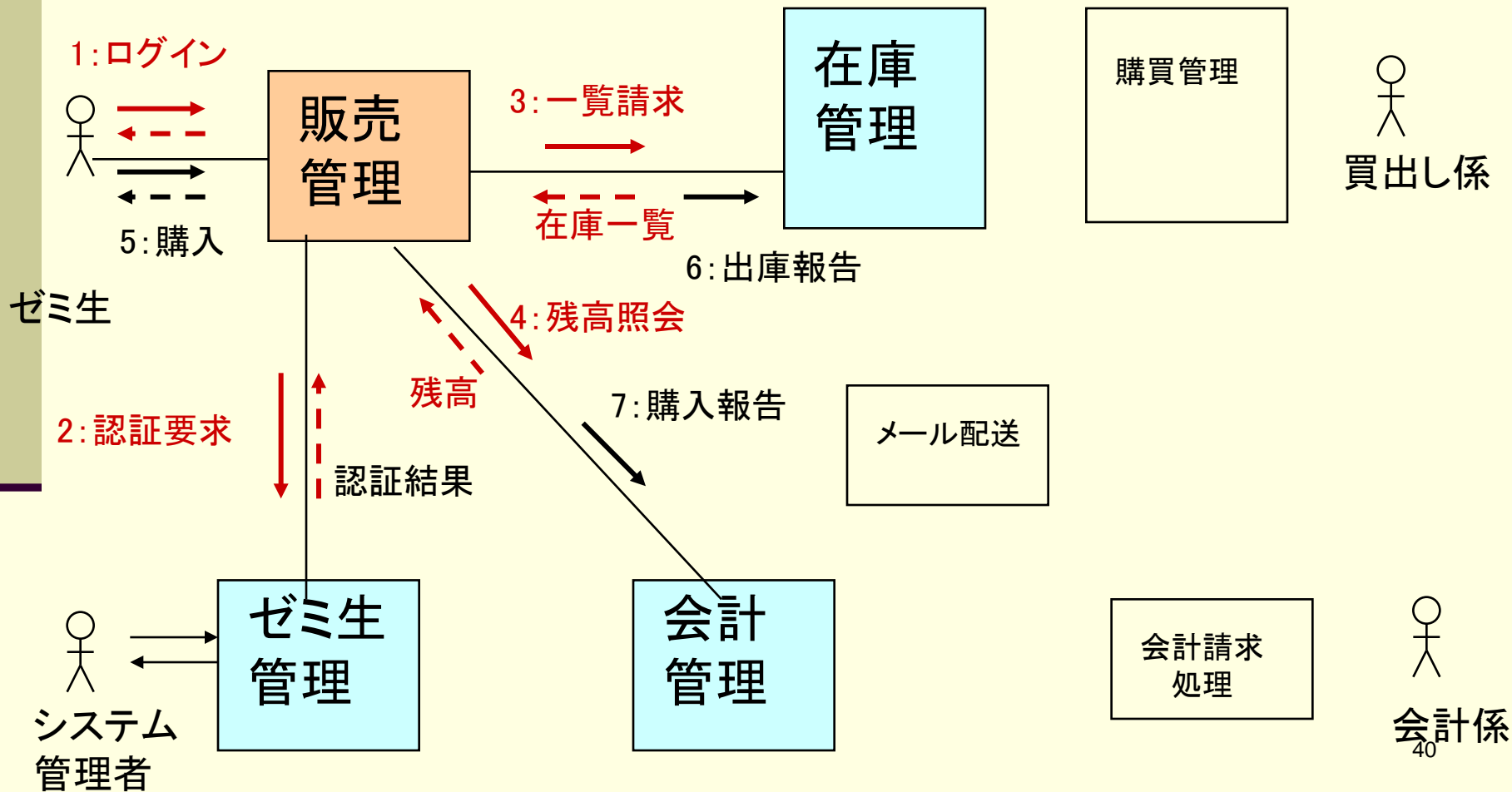
# コラボレーション図 (UML 1.\*風)



# コミュニケーション図 (販売管理関連:UML 2.0 風)



# コミュニケーション図 (ログイン & 購入)





# Webサービスによる統合

## フォーム変換による要求定義

明大  
中研

◆UI駆動型要求分析は左端のみを対象とした

◆このフォームは抽象フォームで、エンドユーザ向けのインタフェース表記に利用

初期画面

ユーザID

パスワード

FTFT(1:1)

販売管理

BL:型チェック

FTFT(1:2)

販売管理

BL:結果NG  
なら別処理

FTFT(2:1)

販売管理

◆ビジネスロジック(BL)を定義する方法が課題

認証要求

ユーザID

パスワード

認証結果

ユーザID

結果

一覧問合せ

個別残高問合せ

ユーザID

在庫一覧

商品名	価格	在庫数

個別残高

ユーザID

残高

問合せ

問合せ

問合せ

DB

DB

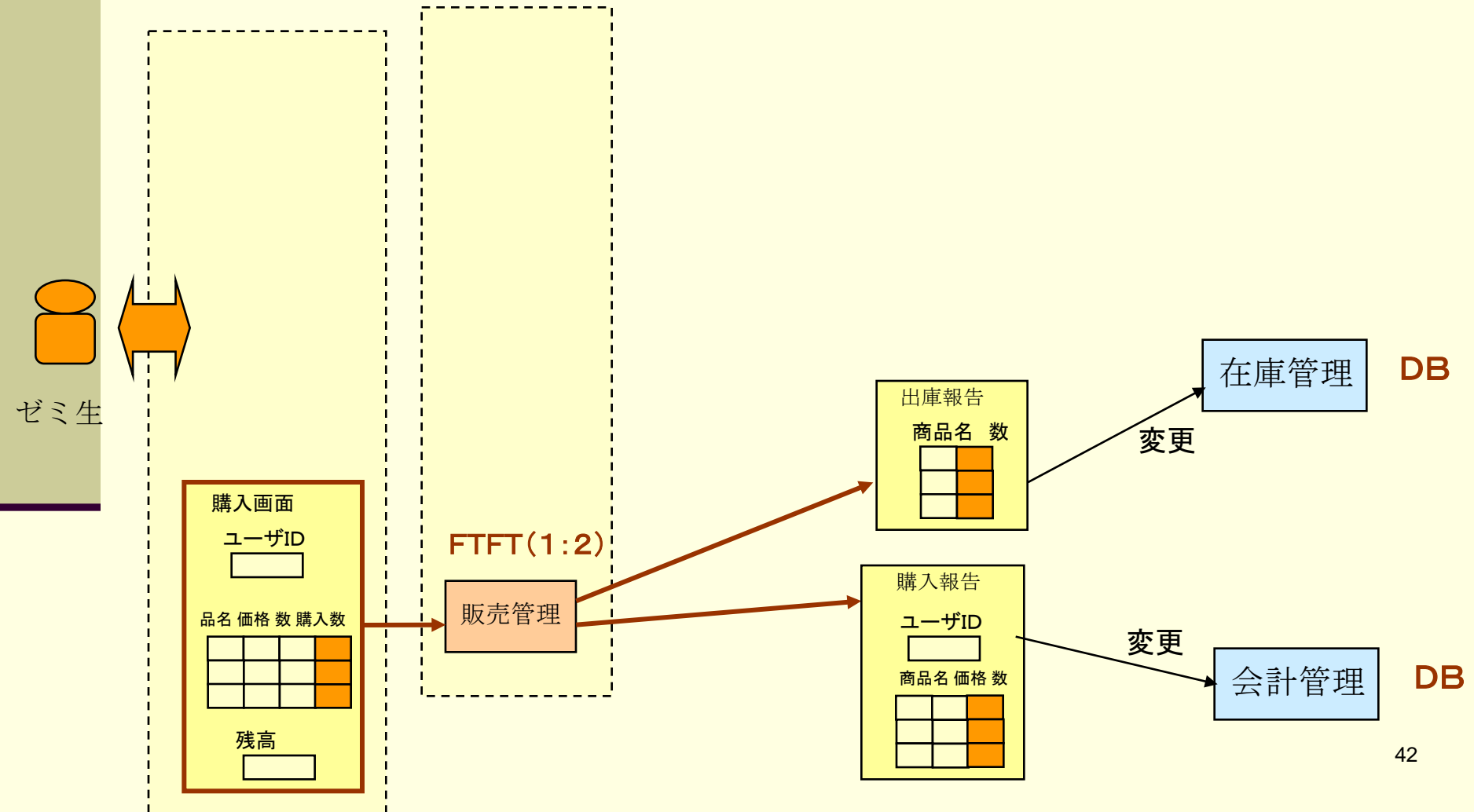
DB



客

# Webサービスによる統合(続)

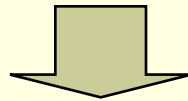
## フォーム変換による要求定義



# まとめ

- エンドユーザ主導開発の実現性
  - 業務の知識を有するエンドユーザ(業務の専門家)が、自らの業務をコンピュータ化する技法

\* アプリケーションフレームワーク  
\* ビジュアルモデリング



\* フォーム変換によるWebサービス連携

ご清聴ありがとうございます.