



INTER-Mediator

《三二》勉強会 2017 in 山口

2017年11月25日

山口県周南市コワーキングスペース「カラム」

新居雅行 Masayuki Nii, 博士(工学)

 nii@msyk.net

 [msyk_nii](https://twitter.com/msyk_nii)

 [msyknii](https://www.facebook.com/msyknii)

Agenda

- INTER-Mediatorで作るWebアプリケーション
 - ▶ アプリケーション開発の必要性
 - ▶ アプリケーション開発のためのアーキテクチャ
 - ▶ INTER-Mediatorの開発プロセス
 - ▶ INTER-Mediator開発のために必要な知識
- ハンズオン
 - ▶ 簡単なWebアプリケーションを作ってみます
 - ▶ 適当に休憩を取りながら、一緒に進めます
- INTER-Mediatorの全貌
 - ▶ データベース対応
 - ▶ 認証・認可への対応
 - ▶ 高度なユーザーインタフェース
 - ▶ ロジックの実装と複雑なアプリケーション構築

プレゼン
20分程度

制作作業
2時間程度

プレゼン
20分程度

アプリケーション開発の必 要性

開発が必要になる場面

- 既存のサービスやアプリケーションでは不足する
 - ▶ 例：Facebookでは、ディスカッションがやりにくい
 - ▶ 例：Excelでできなくはないが、操作に長時間かかる
- 独自システムで競争力を持ちたい
 - ▶ 例：通販サイトで商品の見せ方を工夫したい
 - ▶ 例：自社のビジネスのノウハウをシステム化したい
- そもそも、やろうとしていることの独自性が高い
 - ▶ 例：特定のAPIを利用したシステム
 - ▶ 例：音声でやり取りするチャットが必要

開発したいができないジレンマ

- 開発するのは大変
 - ▶ 自分でできない？するとしても多大が学習時間が必要
- 専門会社に依頼するといいいのか？
 - ▶ 非常にコストがかかる
 - ▶ 依頼しても思い通りのものが作成できるか心配
 - ▶ そもそも、依頼すること自体が大変で二の足を踏む
- 現状は？
 - ▶ 予算が確保できる場合において開発を進めることができる
 - ▶ 発注側が開発にタッチできない状況が発生し、要求にそぐわないものが納品される
(丸投げの問題)

開発の問題を解決する1つの方法

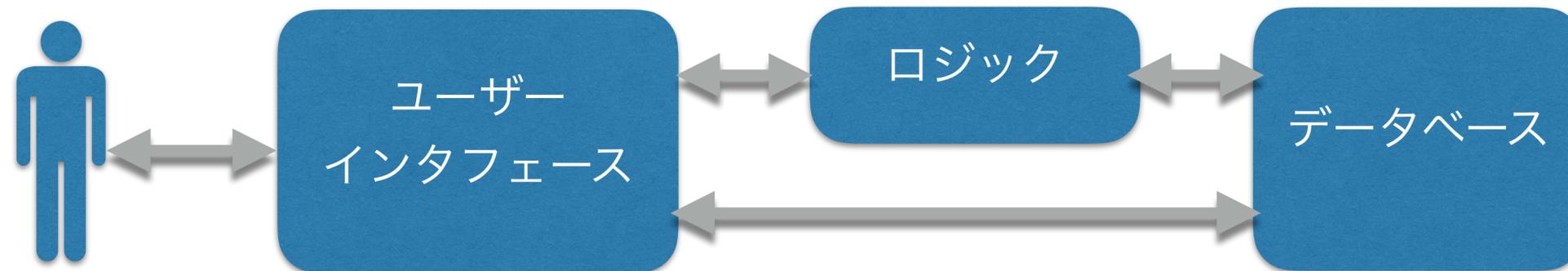
- エンドユーザーが開発に参画する
 - ▶ 利用者が自身の問題として開発を捉える
 - ▶ 業務ノウハウをシステムに組み込むことがよりスムーズ
 - ▶ 業務に合った仕様やテストを計画し、実施可能である
- 今まで、エンドユーザー参画ができなかった理由
 - ▶ 難しいプログラミングやシステム管理が必要であり、専門家でないとは構築作業ができなかった
- 以下のようなエンドユーザーを想定
 - ▶ 自身の仕事内容を熟知している
 - ▶ データベースについての基本概念の理解
 - ▶ HTML/CSSを利用したページ作成ができる

HTMLコーダーやWeb
デザイナーの特性に
近い

アプリケーション開発のため のアーキテクチャ

システムを構成する3つの要素

要素	ユーザー インタフェース	ロジック	データベース
動作	システムの利用	複雑な処理の実現	データの保存
目的	利用効率向上 ミスの軽減	複雑な処理の実装ができる	安定したデータ共有
意図	要求の実現 特に非機能要求	要求の実現 特にビジネス的要求	ドメインの実現

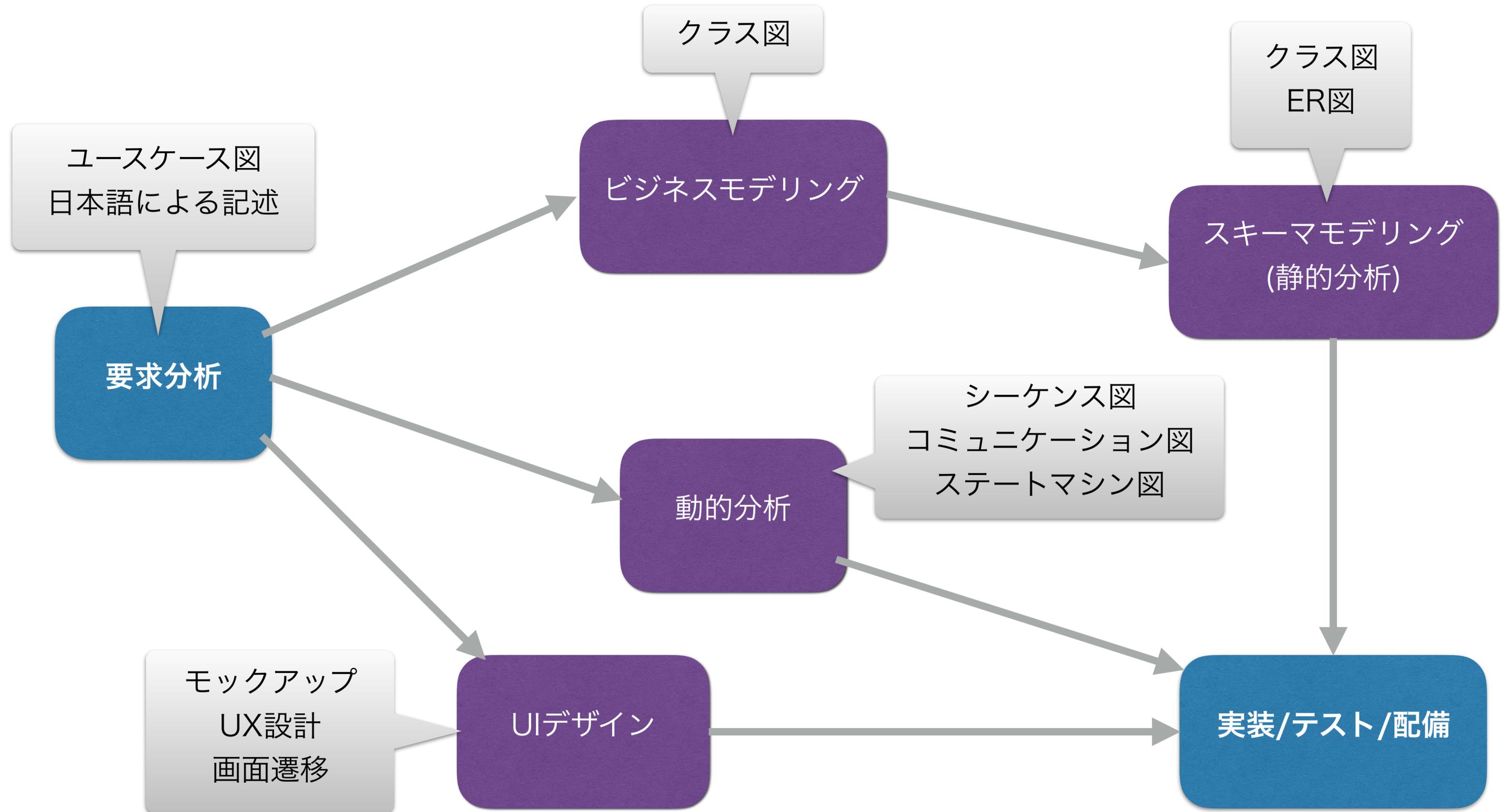


エンドユーザーからの視点

- ユーザーインターフェース
 - ▶ 😊 必要な要素は抽出できる→例えば、HTMLで作成する
 - ▶ 😱 デザイン的な作業は困難
- データベース
 - ▶ 😊 表として扱うという概念は容易に理解できる
 - ユーザーインターフェースに必要な要素の抽出はできる
 - ▶ 😭 リレーションシップを含めた設計は困難
- ロジック
 - ▶ 🤔 動作仕様は記述可能である→例えば日本語で記述する
 - ▶ 😭 動作を実装することには困難が発生する

INTER-Mediatorの開発プロセス

一般的なシステム開発の流れ



INTER-Mediatorでの実装作業

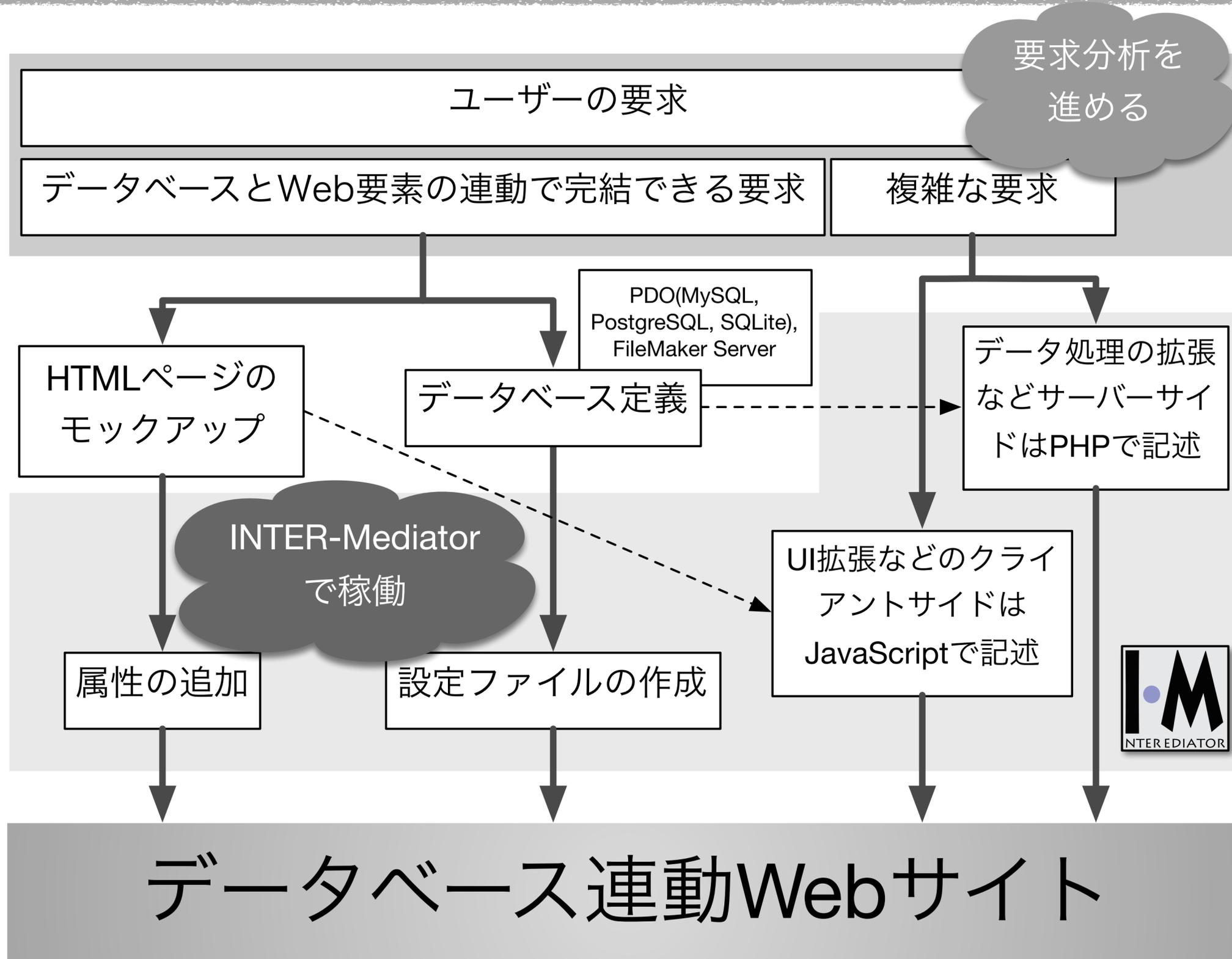
- ユーザーインタフェース
 - ▶ HTMLでUIデザインを行う（モックアップ駆動開発）
 - ▶ データベースと連動する部分を属性でマーキング
 - ▶ UIデザインに応じて定義記述を行い機能を呼び出す
- スキーマモデリング
 - ▶ データベース上に展開する
 - ▶ データベース利用できるように記述
- 動的分析結果
 - ▶ UIに関わる部分など、クライアント側での処理はJavaScriptで実装可能
 - ▶ サーバー処理に関わる部分はPHPで実装可能

エンドユーザーが対応可能

複雑なものは
専門的な知識が必要

基本的に専門的な知識が必要

INTER-Mediatorでの開発プロセス



専門家による援助が前提になるが…

- INTER-Mediatorで開発する場合…
- エンドユーザーが開発の全てを担えない
 - ▶ ビジネスモデリングやスキーマは難しい領域
 - ▶ 手続き的なプログラミングは専門的な知識が必要
 - ▶ システム全体を見渡した上での判断が必要な場合もある
 - ▶ **専門家が関わらないとできない領域は必ずある**
- エンドユーザーが主体的に関われる場面
 - ▶ 要求を文書化するなどの要求定義
 - ▶ HTMLでUIデザインを行う
 - ▶ 機能を呼び出すための設定（宣言的な記述）を行う
 - ▶ **宣言的な記述の部分は多大な学習コストをかけなくてもエンドユーザーは習得可能**
 - コーダーやデザイナーなど、Web業界のスタッフの多くは同様に关われる

他のフレームワークとの違い

- INTER-Mediatorでは、手続き的なプログラミングをしなくても実装や改変が可能な箇所が多い
 - ▶ UI実現において、他のフレームワークで手続き的なプログラミングが必要な箇所でも、INTER-Mediatorでは設定を行うだけで可能
 - ▶ モデリング、動的な動作の実現は、INTER-Mediatorも他のフレームワークでも専門的な知識が前提
- 「エンドユーザー向けWebアプリケーションが持つべき機能*」のほとんどを、宣言的な記述で実装可能
 - ▶ 他のフレームワークに比べて、エンドユーザーによる作成や改変できる範囲が広い
- 現実のシステムは、UIの実現に多大な労力がかかっている
 - ▶ UI周りの開発効率を高めるのがINTER-Mediatorの狙い

* Rode, J. and Rosson, M. B.: Programming at Runtime: Requirements and Paradigms for Nonprogrammer Web Application Development, HCC '03

INTER-Mediator開発のため に必要な知識

稼働させるために必要なこと

- 前提条件
 - ▶ PHPが動作するWebサーバー
 - ▶ データベースには、スキーマが適用されたデータベースがすでに存在する
- データベース連動ページに必須のもの
 - ▶ ページファイル
 - HTMLファイルで記述し、一部に独自の属性を記述
 - ▶ 定義ファイル
 - PHPファイルで作成するが、キーに対する値を記述するの中心で、それ以外は定型的な記述である
 - 従って、PHPではあるが、手続き的なプログラミングではなく、宣言的な記述で行っていると言える
- 宣言的な記述だけで、データベース連動したページの骨格を構築できる

定義ファイルの例

定義ファイルエディタでの設定

定義ファイルの中身はPHPのコード

Contexts											
追加											
name	person			table	person			view	person		
	key	id		sequence				paging	true		
	repeat-control	confirm-insert confirm-d		navi-control				records	1		
	maxrecords	100		cache				soft-delete			
Query											
追加											
Sorting											
追加											
Relationship											
追加											
Options											
theme				media-root-dir				media-context			

```
<?php
require_once('INTER-Mediator/INTER-Mediator.php');

IM_Entry(array (
  0 =>
  array (
    'name' => 'person',
    'table' => 'person',
    'view' => 'person',
    'records' => 1,
    'maxrecords' => 100,
    'paging' => true,
    'key' => 'id',
    'repeat-control' => 'confirm-insert confirm-delete',
    'validation' =>
    array (
      0 =>
      array (
        'field' => 'name',
        'rule' => 'length(address)>0',
        'message' => '空欄はだめです!',
        'notify' => 'inline',
      ),
    ),
  ),
),
array (
),
array (
  'db-class' => 'PDO',
  'dsn' => 'mysql:host=localhost;dbname=test_db;charset=utf8mb4',
),
false);
```

ページファイルの例

```
1 <!DOCTYPE html>
2 <html lang="ja">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <title></title>
7   <script type="text/javascript" src="def03.php"></script>
8 </head>
9 <body>
10 <div id="IM_NAVIGATOR"></div>
11 <table>
12   <tbody>
13     <tr><th>id</th><td data-im="person@id"></td></tr>
14     <tr>
15       <th>name</th>
16       <td><input type="text" data-im="person@name"></td>
17     </tr>
18     <tr>
19       <th>mail</th>
20       <td><input type="text" data-im="person@mail"></td>
21     </tr>
22   </tbody>
23 </table>
24
25 <table>
26   <tbody data-im-control="post">
27     <tr>
28       <th>name</th>
29       <td><input type="text" data-im="person@name"></td>
30     </tr>
31     <tr>
```

純粋なHTMLファイル、独自拡張なし

定義ファイルをscriptタグで読み込み

ページネーションをこのIDの要素に生成

コンテキスト名@フィールド名でバインド

様々なコントロール、ここは入力専用モード

ハンズオン

資料のURL

- バーチャルマシンの利用
 - ▶ <http://inter-mediator.info/ja/for-novices/vm.html>
 - ▶ VirtualBox 5.2で必要な機能の呼び出し方法が一部変化しています
- ハンズオン
 - ▶ <http://inter-mediator.info/ja/for-novices/handson.html>
 - ▶ 一緒に進めましょう！

INTER-Mediatorの全貌

対応するデータベース

- 対応データベース
 - ▶ FileMaker Server／Ver.12以降だが、13以降を推奨
 - Custom Web Publishing（XML共有）とFX.phpを利用したアクセス
 - FileMaker REST APIを利用したアクセス（ほぼ稼働）
 - ▶ MySQL 5.7、MariaDB
 - ▶ PostgreSQL 8/9
 - ▶ SQLite
 - ▶ Microsoft SQL Server／PHP7以降のドライバを利用
 - いずれも、PDOをベースに対応している
- Webアプリケーションの動作環境
 - ▶ PHP 5.x/7.xが稼働するWebサーバー
- 対応ブラウザ
 - ▶ HTML5準拠／IEはVer.9以降だが、現在はVer.11でのみチェック

基本機能

- バインド
 - ▶ データベースのフィールドと、ページ上の要素を論理的に結合
 - ▶ フィールドの値が要素の値やその属性値として設定可能
 - ▶ フォーム要素は修正するとデータベース更新
- 繰り返し
 - ▶ 1レコード分のページ要素を、レコードの数だけ繰り返して、リスト表示
 - ▶ ページネーションの自動生成
- リレーションシップ
 - ▶ 内包する領域に、関連レコードをさらに展開（何段階でも内包可能）
- レコード作成、削除、複製
 - ▶ 設定により適切な位置に自動的にボタンが作成される

高度なユーザーインタフェースを簡単に実現

- ページネーション
- マスター/ディテール形式のUI
 - ▶ 切り替わり、左右に配置
- テーマ機能
 - ▶ 既定のスタイルでも十分に視認性の高いページが作成される
- ファイルアップロード
 - ▶ 独自UIによるドラッグ&ドロップや、ファイル選択
 - ▶ JQuery-FileUploadを利用したプログレス表示機能付きコンポーネント
- 同時参照時の即時更新
 - ▶ クライアント間の同期、Pusherを利用

検索フィールド、ソートボタン

- 検索フィールド

- ▶ `<input type="text" data-im="_@condition:postalcode:f3,f7,f8,f9:*match*">`
- ▶ 入力してリターンを押すと…
 - ページ上に展開しているpostalcodeコンテキストを更新
 - その時に、フィールドf3, f7, f8, f9の値について、OR検索
 - テキストフィールドの値が「含まれる (*match*)」レコードに絞り込む

- ソートボタン

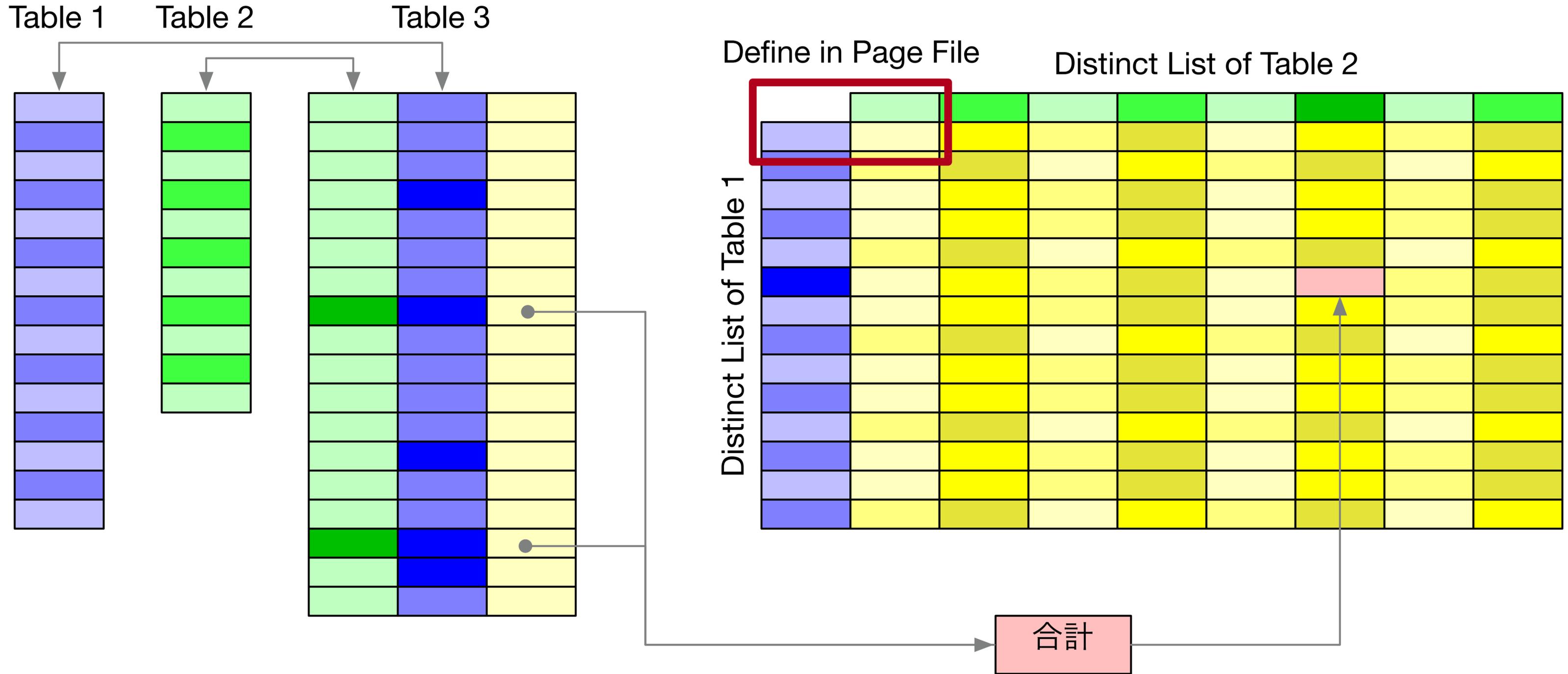
- ▶ `▲`
- ▶ 三角をクリックすると…
 - ページ上に展開しているpostalcodeコンテキストを更新
 - その時に、フィールドf3を昇順でソートする設定を含めてデータベース検索

- ポップアップメニューから1ページのレコード数指定も可能

認証・認可

- データベースの内容をやりとする段階で認証を行う
 - ▶ ページファイル自体のやりとりでの認証はWebサーバーに依存
- 独自のログインパネル
 - ▶ ユーザーを管理するテーブルを利用、グループ管理も持つ
 - ▶ データベースシステム自体のアカウントを利用することもできる
 - ▶ LDAP、OAuth2にも対応（Active DirectoryやGoogleアカウント対応実績あり）
- 認可の設定
 - ▶ コンテキスト／データベース操作／レコード単位での認可が可能
- その他の機能
 - ▶ 画像やPDFファイルなどのファイルに対するアクセス権の設定も可能
 - ▶ パスワードポリシーの設定

クロス集計



計算プロパティ バリデーション

- 計算プロパティ = FileMakerの計算フィールド
 - ▶ 計算式を与えて、計算結果が得られる新たなフィールドを定義する
 - ▶ 同じレコードだけでなく、ページ内に展開している他のテーブルも参照可能
 - ▶ 文字列、数値だけでなく、日付や時刻、リストの計算処理関数も用意
- バリデーション
 - ▶ フィールド単位に式で判断し、ダイアログボックスやページ上にエラーメッセージを表示する
 - ▶ 複数のフィールドにまたがる判断は、JavaScriptを利用する
 - ▶ ユーザーインタフェース側のみで処理をしており、サーバーへの通信上では処理をしていない点に注意が必要／将来バージョンでサポート

そのほかの機能

- メール送信
 - ▶ データベース処理後にメール送信、テンプレートやフィールドの値の利用
- ブラウザー判別
 - ▶ 非対応ブラウザの場合にメッセージを表示
- 集計処理を含むSQLステートメントの記述
 - ▶ GROUP-BYやJOIN等を含むSQLによるコンテキストの取得
- 論理削除
 - ▶ 特定のフラグとなるフィールドを利用して、レコード削除をしないで削除をしたかのように振る舞う手法（ソフトデリート）

ロジックの実装と複雑なア プリケーション構築

クライアントサイドのプログラミング

- JavaScriptで実装
- APIの例
 - ▶ ページに展開したデータを取得、展開データを変更してサーバーに反映
 - ▶ データベース処理を直接行う
- 様々な実行タイミングの用意
 - ▶ ページ生成前後
 - ▶ 1テーブルの展開直後、1レコードの展開直後
 - ▶ バリデーションの前後
- ユーザーインタフェースコンポーネント
 - ▶ ファイルアップロードや、HTMLエディタなどを用意
 - ▶ 独自に開発可能

サーバーサイドのプログラミング

- PHPで実装する
- 既存実装機能の拡張や変更
 - ▶ フィールド単位の「データコンバータ」、DBアクセスのためのクラスの自作
- データベース処理の前後にプログラムを追加
 - ▶ アスペクト指向的な動作
 - ▶ データベース処理と絡めて追加の処理を行う
 - ▶ 例：APIを叩くシステムの場合
 - クライアントでボタンを押して、データベースに新規レコードを作るようにする
 - 新規レコード前の処理にAPI呼び出しを行う
 - API呼び出しした結果をフィールドに追加することで、新規レコードにはAPI呼び出しの結果が残る

利用実績

実績：ふち無しはがき印刷本舗

- FileMaker Server
- オンデマンド印刷
 - ▶ 顧客ごとに変わる注文内容を管理
 - ▶ CRM機能により、修正の依頼や素材の受け渡しがスムーズに

The screenshot shows the website interface for 'ふち無しはがき印刷本舗'. The main page features a navigation menu with '印刷メニュー' (Print Menu) and 'お助け情報' (Help Information). The '印刷メニュー' includes options like 'ふち無し年賀状', 'ふち無し暑中見舞い', and 'ビジネス挨拶はがき'. The 'お助け情報' section includes '無料画像・テンプレート集', '年賀状の書き方辞典', and '暑中見舞いの書き方辞典'. A central banner advertises a '6.28 > 7.15' sale for 'ふち無しはがき印刷 白フチ'. The right side of the page shows a detailed order page for '暑中見舞いフォト宛名 (商品番号: 00266)'. The order progress is shown as 'STEP2 ご入稿' (Order Submission), with 'STEP1 申し込み' (Application) completed on 2013/06/13. The order details include: 'ご注文名: 暑中見舞いフォト宛名', '商品番号: 00266', '会社名: 3360923', 'お客様名: 新居雅行', 'フリガナ: にいまさうき', '郵便番号: 222-0003', '住所: 神奈川県横浜市港北区大曽根', 'メール: msyk@msyk.net', '選択商品: 暑中見舞い_フォトフレーム', '印刷枚数: 200', '宛名印刷枚数: 100'. The 'ご請求金額' (Request Amount) table is as follows:

商品名	単価	個数	金額
暑中見舞い (かもめ〜る)	50	200	10,000
暑中見舞いはがき印刷	60	200	12,000
宛名印刷	18	100	1,800
宛名データ処理費	2,000	1	2,000
佐川急便	840	1	840
合計金額: ¥26,640			
うち消費税: ¥1,269 (5%)			

The total amount is ¥26,640, including ¥1,269 (5% tax). The page also shows a 'STEP2: ご入稿' (Order Submission) section with the heading '1. お客様作業のご案内' (Customer Work Instructions). A note states: 'ご注文では、はがきデザイン用のデータをご入稿ください。以下の作業が必要です。' (For your order, please submit data for postcard design. The following work is required.)

実績：倉敷芸術科学大学 研究業績DB

- MySQL
- 教員が自身の業績を入力する部分で利用
 - ▶ 外部からのサイトは別フレームワークで実装
- Excel入力をメールで集める手順から大きな進歩

倉敷芸術科学大学 教育研究業績データベース

このサイトでは、倉敷芸術科学大学の教員・研究者の教育研究業績について情報公開しています。

倉敷芸術科学大学 教育研究業績データベース

学校法人加計学園 倉敷芸術科学大学の教育研究業績データベース
このサイトは倉敷芸術科学大学の教員・研究者の教育研究業績について情報公開しています。

教育研究業績を調べる

学部・学科から調べる

学部に所属する教員一覧の中から教育研究業績を調べたい教員を選んでください。

芸術学部
産業科学技術学部
生命科学部

倉敷芸術科学大学 教育研究業績データベース

公開画面 データベース保守 ログアウト

基本情報 Basic Info 教育活動 Education 研究活動 Research 学内外の活動 Activities その他 Misc

データ管理 TOP > 教育活動 ようこそ、管理人さん

対象レコード：倉敷 太郎（事務局 事務局）
倉敷先生の教育活動について、入力してください。

1. 担当授業について

※ 科目名の間は全角の「、」で区切ってご入力ください。
※ 大学院開講科目は、科目名の接頭字として「院）」をつけてください。

追加		削除
	年度 担当授業の科目名を「、（全角の区点）」で区切って列挙してください 大学院の担当授業は「院）」に続けて授業名を続けてください	削除
平成 27年	基礎化学Ⅰ、物理学実験、基礎化学Ⅱ、科学総論、院）特別講義Ⅱ	削除
平成 26年	基礎化学Ⅰ、物理学実験、基礎化学Ⅱ、科学総論、卒業研究、院）特別講義	削除

2. 授業以外の教育活動について

※ 入力時の改行は業績表示ページに反映されません。

追加	年度	年度ごとの活動内容	削除
		項目 活動内容	
		オフィスアワー 水曜4時限目	
		リメディアル教育 科学Ⅰ履修者のうち希望者を対象に前期毎週水曜日5時限目に勉強会を実施。	
		教育実習 大学の授業 科学総論にてアクティブラーニングの手法を用いた授業を実施	

キーワード検索