

INTER-Mediator Developer's Meet-UP#1

2012/3/14

KDDI Seminar Room

Agenda



開発報告と開発予定

事例紹介

コンテキスト指向開発のポイント

IMのソースをベース開発をする

ディスカッション

What's IM...はありません

開発報告と開発予定

この集会・平日夜、KDDIさんを基本

3～4ヶ月に1回程度開催予定

- 従って、次回は6～7月くらい

主として現状の確認集会

のちのちは勉強会的なものにしたい

- 発表をしてください

これまでの経緯



Early Story

- 2009/12/30：最初のリリース
- 2010/1：mixiのWeb勉強会で発表／2：FM-Tokyo、Cocoa勉強会で発表
- 2010/5：FFNでプレゼンテーション／リストラクチャー開始
- 2011/4～7：すごい勢いで開発（笑）

Ver.1.0.0 (2011/10/19)

- データベースとWebページの連携のための基本的な仕組み
- サーバ・クライアント双方でのプログラムを追加する仕組み
- Mac OS X Server勉強会で発表、FileMaker Conferenceで話題に

Ver.1.0.2 (2011/12/4)

- トランザクション／Server Night!でプレゼンテーション

Ver.1.0.5 (2012/2/5)

- PDOでの認証

機能追加アンケート結果！

先々の開発計画について、アンケートにしてみました。現在、認証をがんばっているところで、さらに次は、アスペクト指向によるコンテキストの分離を行います。その先、どの機能の組み込みに移るか、みなさんの要望を聞きたいと思います。複数選択もOKですので、よろしくお願いします。

- バリデーション (値のチェックと警告) 
 - COMMIT/ROLLBACKを使った書き込みデータの確定 
 - ユーザによるパスワードの変更 
 - 検索結果のローカルキャッシュ 
 - FileMakerスタイルの検索機能 
 - クラウドデータベースのサポート 
 - ユーザやグループを管理するためのページ 
 - 画像やファイルのハンドリング
 - ローカルデータベースサポート
 - FileMakerの「チェックボックスセット」的な動作をする選択肢
- + オプションを追加...

質問者

14票



新居 雅行 ▶ INTER-Mediator(Japanese)
約1週間前 ·  ·  オプションを編集 · 削除

今後の開発に関する予定

近々の予定

- FileMaker Serverでの認証
- アクセス権のデバッグ、ネイティブユーザ認証

トップエスーの修了制作

- サーバ・クライアントサイドでのプログラム追加をスムーズにできるような仕組みの発見を模索中

さらに先は

- バリデーションかな

バリデーションの実装について

IM_Entryの第2引数ではなく、第1引数で指定する
ruleに「値を返すJavaScriptの記述」を記載する

コンテキスト

```
validation=>[{
  field=>'price',
  rule=>'valCheck(value, target)',
  message=>'Seriously?'}...]
```

生成されるプログラム

```
addEvent( 'change', function(){
  var target = self;
  var value = INTERMediatorLib.getValue(target);
  var result = valCheck(value, target);
  if ( ! result ) { alert( "...message..." ); }
  return result; } );
```


バリデーションの例

コンテキスト validation=>[
 field=>'price',
 rule=>'value>=0']...

生成されるプログラム

```
addEvent( 'change', function(){
  var target = self;
  var value = INTERMediatorLib.getValue(target);
  var result = value>=0;
  if ( ! result ) { alert( “既定のメッセージ” );
  return result;} );
```

バリデーションの仕様



対象

- INPUT/SELECT/TEXTAREA

手法

- changeイベントにチェックメソッドを追加

判定

- ruleの式を織り込んだ関数の返り値がfalseならメッセージを出す

メッセージ

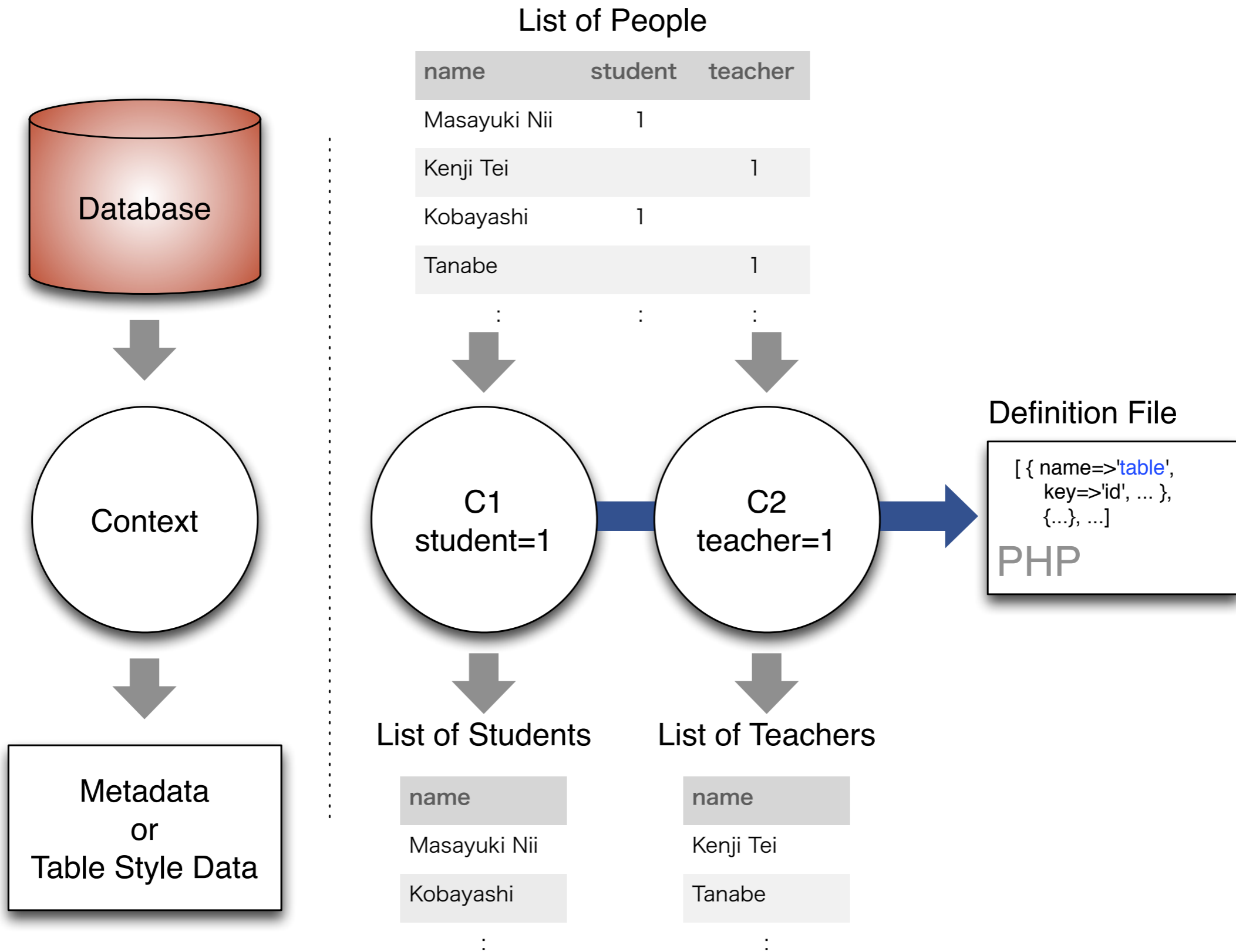
- 指定すれば、独自のメッセージ。指定がなければ既定のメッセージを出す

事例紹介

* * * * *

コンテキスト指向開発の ポイント

コンテキストとは



Specifying the Context

キー名	設定する内容
name	コンテキスト名
view	参照時のデータベースエンティティ
table	更新時のデータベースエンティティ
key	キーフィールド名
query	検索条件
sort	ソート条件
default-values	新規レコード作成時のフィールド値
relation	他のコンテキストとの関連（外部キー）
records	先頭から何レコード分を利用するか
repeat-control	レコード追加や削除のユーザインタフェース
authentication	認証とアクセス権に関する設定

グレイのボックスは、動的に指定が可能

コンテキストの特性

コンテキストの属性

- 前のページの表、さまざまな「条件」とも言える

コンテキストに対する操作

- CRUDに対応するオペレーションが可能

Readオペレーションに対して

- メタデータが得られる
- あるいは、表形式のデータが得られる

Createオペレーションにより

- 新規レコードのキー値が得られる

コンテキスト指向開発



コンテキストを中心に考える

コンテキストを抽出する

コンテキストを定義する

コンテキストによりメタデータが得られる

コンテキストを変更するプログラムを記述する

プログラムの中でコンテキストを利用できる

IMのソースをベースに 開発する

INTER-Mediatorのソース



GitHubを使って公開

専用アプリケーションを使えばかなり楽になる

ハンズオン



アカウントの作成（当然、無料）

アプリケーションをダウンロード

レポジトリのダウンロード

変更結果のアップロード

ディスカッション