

S2Clickをはじめとする Seasarファミリーを活用した アジャイル開発事例のご紹介

NTTデータ先端技術 竹添直樹

自己紹介

竹添 直樹

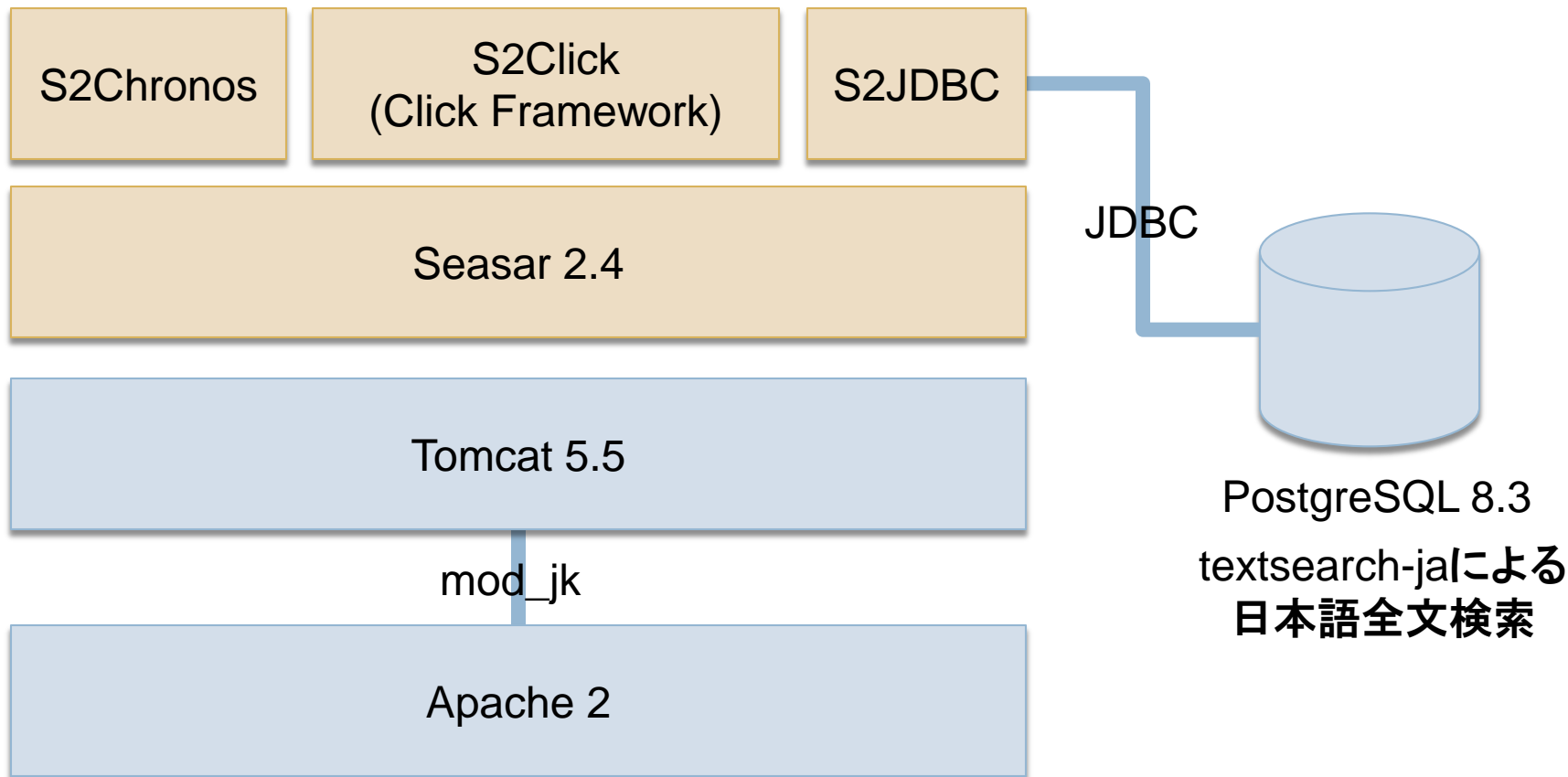
- たけぞう
- NTTデータ先端技術
- OSS活動
 - ▣ Project Amateras
 - ▣ Apache Software Foundation(Apache Click)
 - ▣ Seasarプロジェクト(S2Click, S2JSF Plugin)
- テクニカルライター
 - ▣ 書籍、雑誌、オンラインメディア等での執筆

プロジェクト概要

プロジェクト概要

- 比較的小規模なWebアプリケーション
 - 100画面程度
 - おおよそ5名×4ヶ月
 - 要件が確定していない状態で開発スタート
- アジャイル開発
 - 要件を機能に落としつつ実装を進めた
 - イテレーション毎に動作するアプリケーションをデモ
 - フィードバックを受けて修正の繰り返し

ソフトウェア構成



Seasarファミリーの活用

Click Frameworkの特徴

- イベント駆動、コンポーネント指向のWebアプリケーションフレームワーク
- シンプル・高速・教育コスト低
- ドキュメントが異常なほど充実
- 日本語翻訳Wikiもあります
- 2008年、Apache Incubatorに移行しました!

Click Framework

ページクラス

```
public class SamplePage extends Page {

    public Form form = new Form("form");
    private TextField userId = new TextField("userId");
    private Submit loginButton = new Submit("login");

    public LoginPage(){
        // 入力フォームを構築
        form.add(userId);
        form.add(loginButton);
        // リスナを登録
        loginButton.addActionListener(
            new ActionListener(){
                public boolean onAction(Control source) {
                    // ボタンがクリックされたときの処理
                }
            });
    }
}
```

ビューテンプレート

```
<html>
  <head>
    <title>Simple Form</title>
    $imports
  </head>
  <body>
    $form
  </body>
</html>
```

**ビューはVelocityで記述
(JSPやFreeMarkerも利用可能)**

**\$formと書くだけでページクラスで
組み立てた入力フォームを
自動レイアウトして表示
(手動でレイアウトすることも可能)**

S2Click

- Click FrameworkをS2と組み合わせて使うためのアダプタ+独自の追加機能
 - HOT deploy対応
 - publicフィールド対応
 - コードを削減するための便利機能
 - Ajax対応コンポーネント
 - SAStruts風のURLパターン
- 最近の新機能
 - ユニットテストのサポート

S2JDBC

- O/Rマッピングフレームワーク
 - ▣ 流れるインターフェース(タイプセーフ)
 - ▣ SQLファイル(2Way-SQL)
- 自動生成ツールS2JDBC-Gen

現時点では最もアジャイル開発に適したO/Rマッピングフレームワーク

S2JDBC 流れるインターフェース

```
public List<Message> getMessages(){
    return jdbcManager
        .from(Message.class)
        .orderBy(desc(messageId()))
        .getResultList();
}
```

メリット:

タイプセーフであるためDB変更時の修正漏れがない

デメリット:

細かい制御や複雑なSQLを発行することができない

S2JDBC 外部SQLファイル

```
SELECT * FROM EMPLOYEE
WHERE
  SALARY >= /*salaryMin*/1000
AND
  SALARY <= /*salaryMax*/2000
```

メリット:

細かい制御や複雑なSQLを発行することができる
SQLをそのまま実行することができる

デメリット:

タイプセーフではないためDB変更時の修正漏れがあり得る

S2Chronos

- S2Container上で動作するスケジューラ
- war内に含めることが可能
- 小規模なアプリケーションのバッチ処理には最適

```
@Task
```

```
@CronTrigger(expression = "0 */1 * * * ?")
```

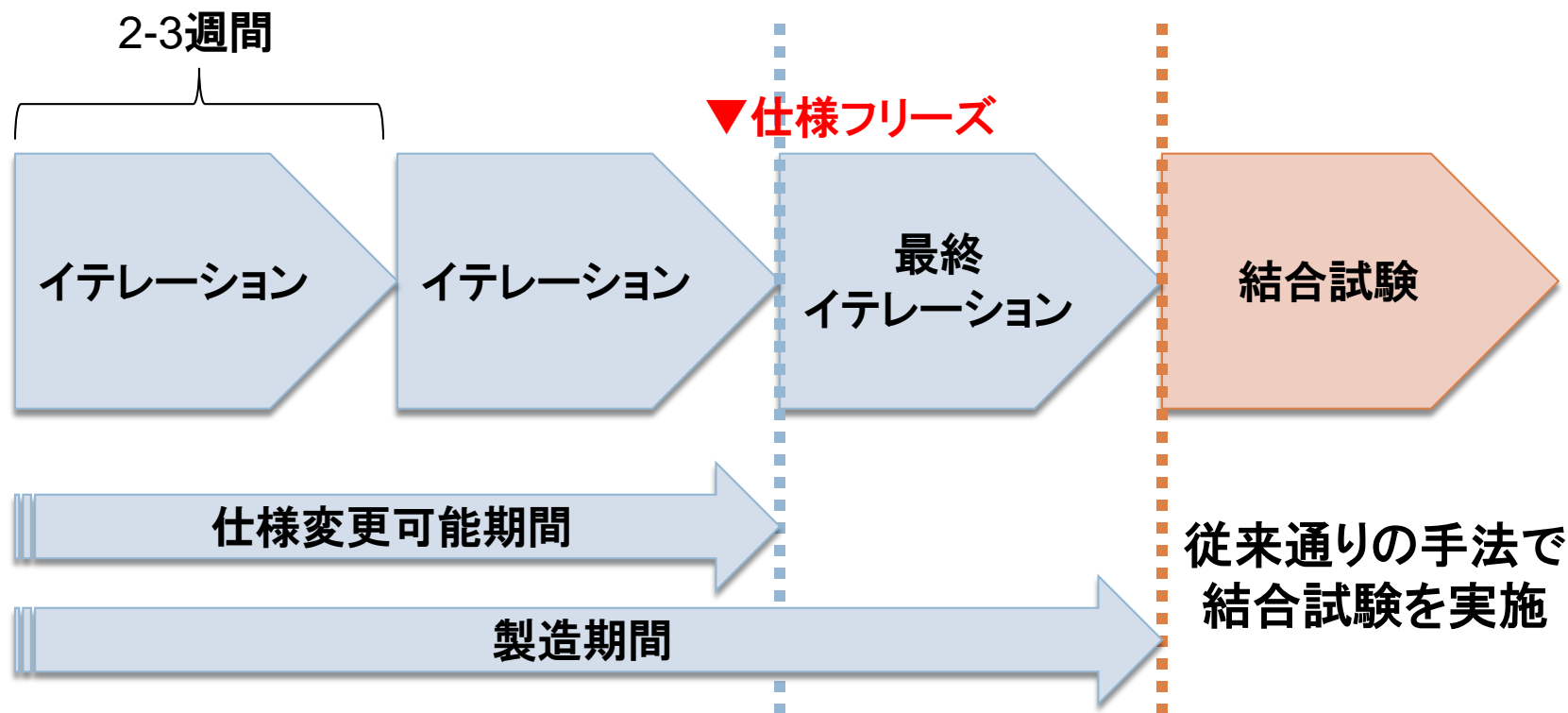
```
public class BasicTask {  
    public void doExecute() {  
        // タスク処理  
    }  
}
```

1分おきに実行

アジャイル開発のための工夫

アジャイル開発のリスクヘッジ

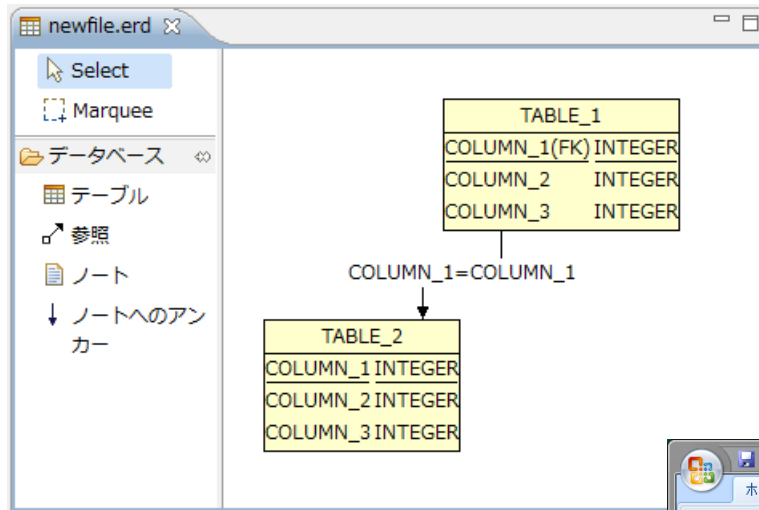
- 仕様が決まらない、仕様変更が止まらない
- 品質の低下(大規模なプロジェクトほど顕著)



変更時のインパクトを極小化する

- フレームワーク・ツール活用でコード量を減らす
 - コード量が少なければ修正も少なくて済む
 - 捨てて作り直してもOKが理想
- タイプセーフで修正漏れを防ぐ
 - 影響範囲の確認が容易
 - 恐れずに変更できる

AmaterasERD



ER図を作成するためのEclipseプラグイン

- ・論理情報/物理情報
- ・ダイアグラムのバリデーション
- ・DDL生成
- ・テーブル定義書(HTML/Excel)生成
- ・Oracle、PostgreSQL、MySQL、Derby、hsqldb、H2をサポート

TABLE_1 - Mozilla Firefox

ファイル(E) 編集(E) 表示(V) 履歴(S) ブックマーク(B) ツール(I) ヘルプ(H)

All Tables
TABLE_1
TABLE_2

項番	論理名	物理名	型	Null	初期値
1	カラム1	COLUMN_1	INTEGER	Y	
2	カラム2	COLUMN_2	INTEGER	Y	
3	カラム3	COLUMN_3	INTEGER	Y	

外部キー

Column	Referenced Column
COLUMN_1	TABLE_2.COLUMN_1

完了

test.xls [互換モード] - Microsoft Excel

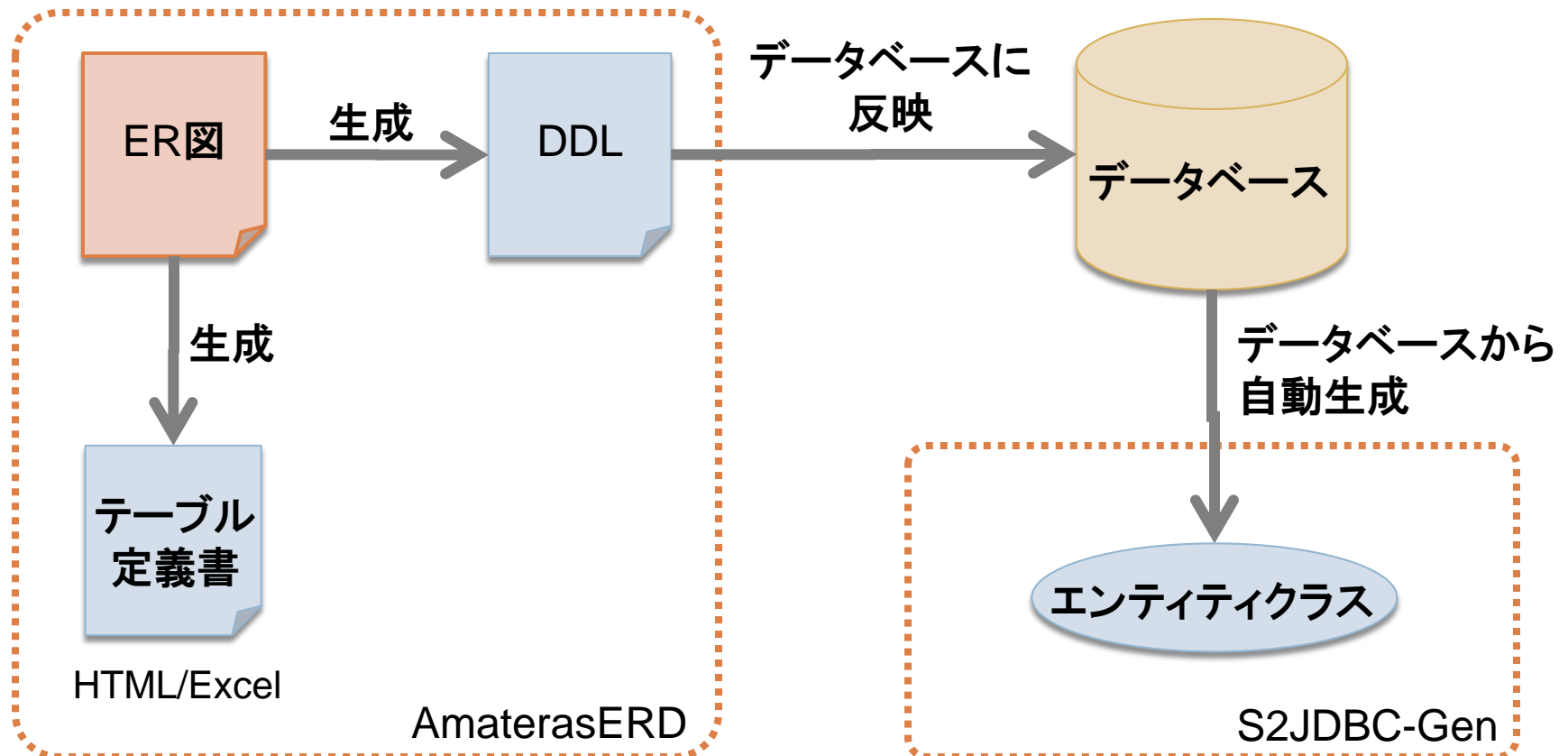
ホーム 挿入 ページレイアウト 数式 データ 校閲 表示 活用しよう! エクセル

テーブル名	論理名	物理名	型	サイズ	NULL	初期値	参照先
テーブル1	カラム1	COLUMN_1	INTEGER				TABLE_2.COLUMN_1
テーブル1	カラム2	COLUMN_2	INTEGER				
テーブル1	カラム3	COLUMN_3	INTEGER				
テーブル2	カラム1	COLUMN_1	INTEGER				
テーブル2	カラム2	COLUMN_2	INTEGER				
テーブル2	カラム3	COLUMN_3	INTEGER				

テーブル定義書

AmaterasERDとS2JDBC-Gen

AmaterasERDによるERDセントリックな開発スタイル



今後の課題

今後の課題

- 開発はかなりこなれて効率化されている
- テストの効率の悪さが目立つ
 - ▣ データベース周りのユニットテスト(データの作成)
 - ▣ UIのテスト(手動)

データベース関連のテストの効率化

- S2TestCase
 - ▣ Excelを使ったデータ投入、アサーションが可能
 - ▣ 初期値、期待値のExcelを作るのが面倒
 - ▣ DB変更のたびにExcelをメンテナンスしなければならない
- S2ClickServiceTestCase
 - ▣ 初期値Excelがあれば自動的にインサート
 - ▣ DBの内容から結果比較用Excelを自動生成

Excel自動生成の例 その1

```
@Assert(tables = @Table(name="MESSAGE"))  
public void testInsertTx() throws Exception {  
    Message message = new Message();  
    message.name = "たけぞう";  
    message.message = "こんにちは";  
    service.insert(message);  
}
```

@Assertアノテーション

データベースの内容を期待値Excelと比較、Excelが存在しない場合はデータベースの内容からExcelを自動生成

Excel自動生成の例 その2

```
@GenerateExcel(  
    tables = @Table(name="MESSAGE"))  
public void testGetMessages2Tx() {  
    List<Message> result = service.getMessages();  
  
    assertBeanListEquals("期待値と一致しません。",  
        getExpectDataSet(), result);  
}
```

@GenerateExcelアノテーション

期待値Excelが存在しない場合はデータベースの内容からExcelを自動生成(アサートはテストケース側で行う)

UIテストの自動化

- WebDriver
 - JavaからWebブラウザを制御
 - JUnitに組み込み可能
 - Seleniumと違いセキュリティ制約を受けない

WebDriverの使用例

```
WebDriver driver = new FirefoxDriver();  
driver.get("http://www.google.co.jp/");
```

// 検索を実行

```
WebElement searchBox = driver.findElement(By.name("q"));  
searchBox.sendKeys("マイコミジャーナル");  
searchBox.submit();
```

// 検索画面のタイトルを取得

```
System.out.println(driver.getTitle());
```

// 検索結果のURLを取得

```
WebElement link = driver.findElement(By.linkText("マイコミジャーナル"));  
System.out.println(link.getAttribute("href"));
```

**さらなる開発の
効率化を
目指します！**

参考URL

参考URL

- Apache Click
 - <http://incubator.apache.org/click/>
- Click日本語翻訳Wiki
 - <http://amateras.sourceforge.jp/cgi-bin/fswiki/wiki.cgi/click>
- S2Click
 - <http://s2click.sandbox.seasar.org/>
- S2JDBC / S2JDBC-Gen
 - <http://s2container.seasar.org/2.4/ja/s2jdbc.html>
 - http://s2container.seasar.org/2.4/ja/s2jdbc_gen/index.html
- S2Chronos
 - <http://s2chronos.sandbox.seasar.org/ja/>
- textsearch-ja(PostgreSQLで日本語全文検索)
 - <http://textsearch-ja.projects.postgresql.org/index-ja.html>
- AmaterasERD
 - <http://amateras.sourceforge.jp/cgi-bin/fswiki/wiki.cgi?page=AmaterasERD>
- ホワイトボード(アスクル)
 - <http://www.askul.co.jp/p/131488/>
- Web Driver
 - <http://code.google.com/p/webdriver/>

ご清聴ありがとうございました