



業務システム開発用フレームワーク TERASOLUNA Framework for .NET

2008年10月18日(土)

presented by tatsumihr

株式会社NTTデータ
技術開発本部
立見博史

記載されている会社名、商品名、又はサービス名は、各社の登録商標又は商標です。



わんくま同盟 東京勉強会 #25

Agenda

- はじめに
- TERASOLUNA for .NET概要
- デモ
- まとめ

Agenda

- はじめに
- TERASOLUNA for .NET概要
- デモ
- まとめ

自己紹介

- tatsumihr
 - TERASOLUNAフレームワーク開発メンバ
 - TERASOLUNAオープンソースプロジェクトメンバ
(<http://terasoluna.sourceforge.jp>)
 - 雑誌執筆
 - @IT
「TERASOLUNA for .NETフレームワーク概説」
 - .NET開発でもオープンソース・フレームワークを使おう
(http://www.atmarkit.co.jp/fdotnet/terasoluna/terasoluna01/terasoluna01_01.html)
 - DB マガジン
「.NETコントロールまるわかり教室」連載中

TERASOLUNAとは？

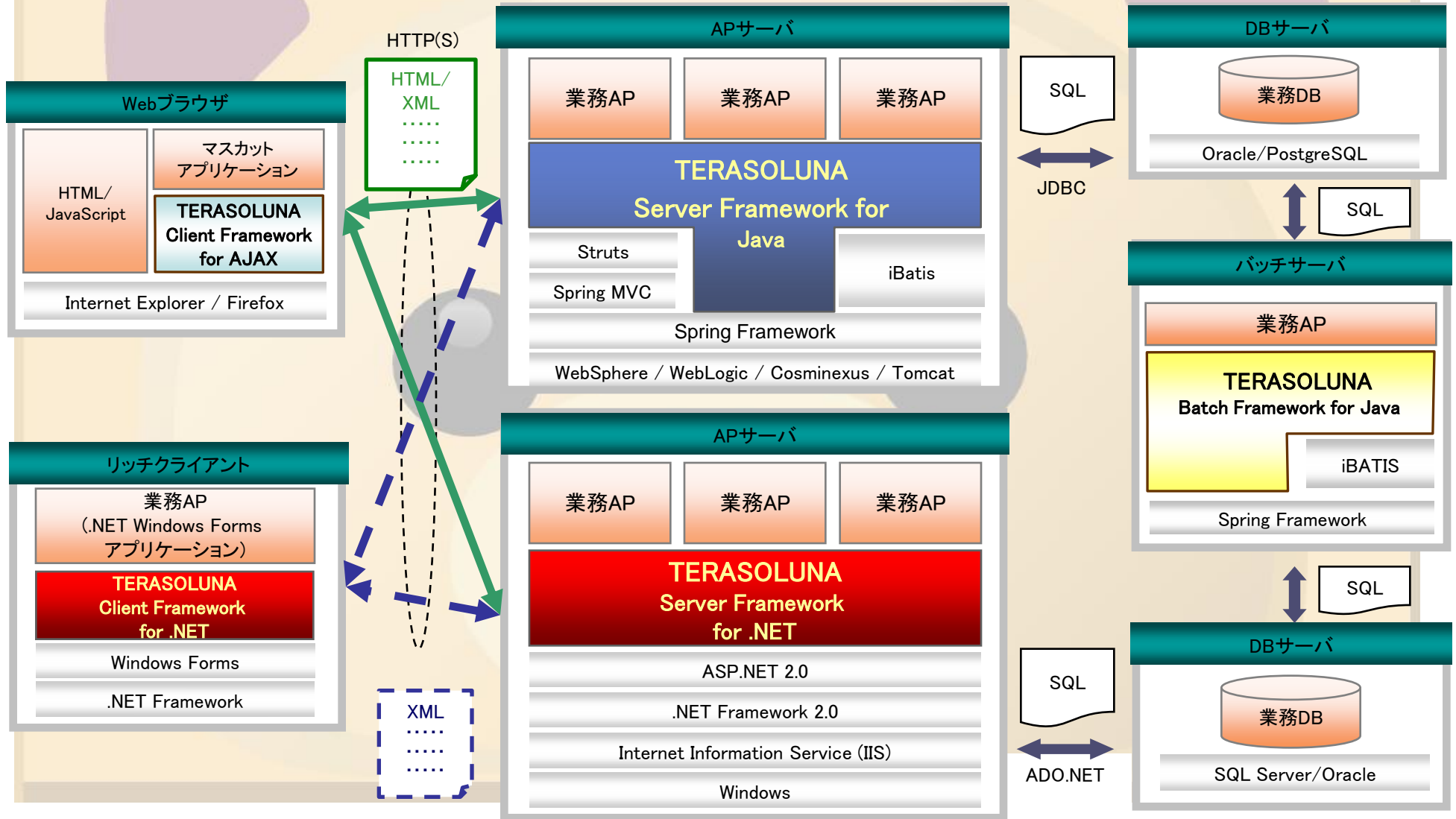
- TERASOLUNA(テラソルナ)とは
 - TERASOLUNAは、NTTデータが提供する「フレームワーク」、「開発プロセス」、「プロジェクト管理」などの技術やノウハウを組合せ、システム開発を包括的にサポートするソリューション
- TERASOLUNAフレームワークとは
 - NTTデータが数多くのシステム開発で培ってきた豊富なノウハウが継続的にフィードバックされたフレームワーク
 - 各種プラットフォームによる、オンライン系/バッチ系ソリューションに対応した複数のフレームワークを用意しており、要件に応じて選択や組み合わせが可能

TERASOLUNAフレームワークラインアップ

- TERASOLUNAフレームワークラインアップ一覧
 - オンライン系／バッチ系アプリケーションに対応した5種類
 - Java、.NET、Ajaxに対応、最適な組み合わせを選択可能

	サーバ フレームワーク	クライアント フレームワーク	バッチ フレームワーク
.NET	TERASOLUNA Server Framework for .NET	TERASOLUNA Client Framework for .NET	—
Java	TERASOLUNA Server Framework for Java	—	TERASOLUNA Batch Framework for Java
Ajax	—	TERASOLUNA Client Framework for Ajax (マスカット)	—

TERASOLUNAフレームワークラインアップ



Agenda

- はじめに
- TERASOLUNA for .NET概要
- デモ
- まとめ

業務システムを開発するときの共通的な問題点

- 品質の問題

- メンテナンスができないコード

- .NETではUIとビジネスロジックの分離しなくても作れてしまうなど、自由度の高い開発が可能であるため、規約がないと開発者によって実装方法がバラバラになってしまう
 - 後でメンテナンスしようとしても、1つのメソッドが膨大かつ複雑になっていると、どこを直せばよいか、わからなくなってしまう

※クライアント・サーバシステムの悪夢の再来

- コストの問題

- 作りこみが多い

- クライアントアプリケーションだと、非同期処理など複雑な処理の作り込みが多い

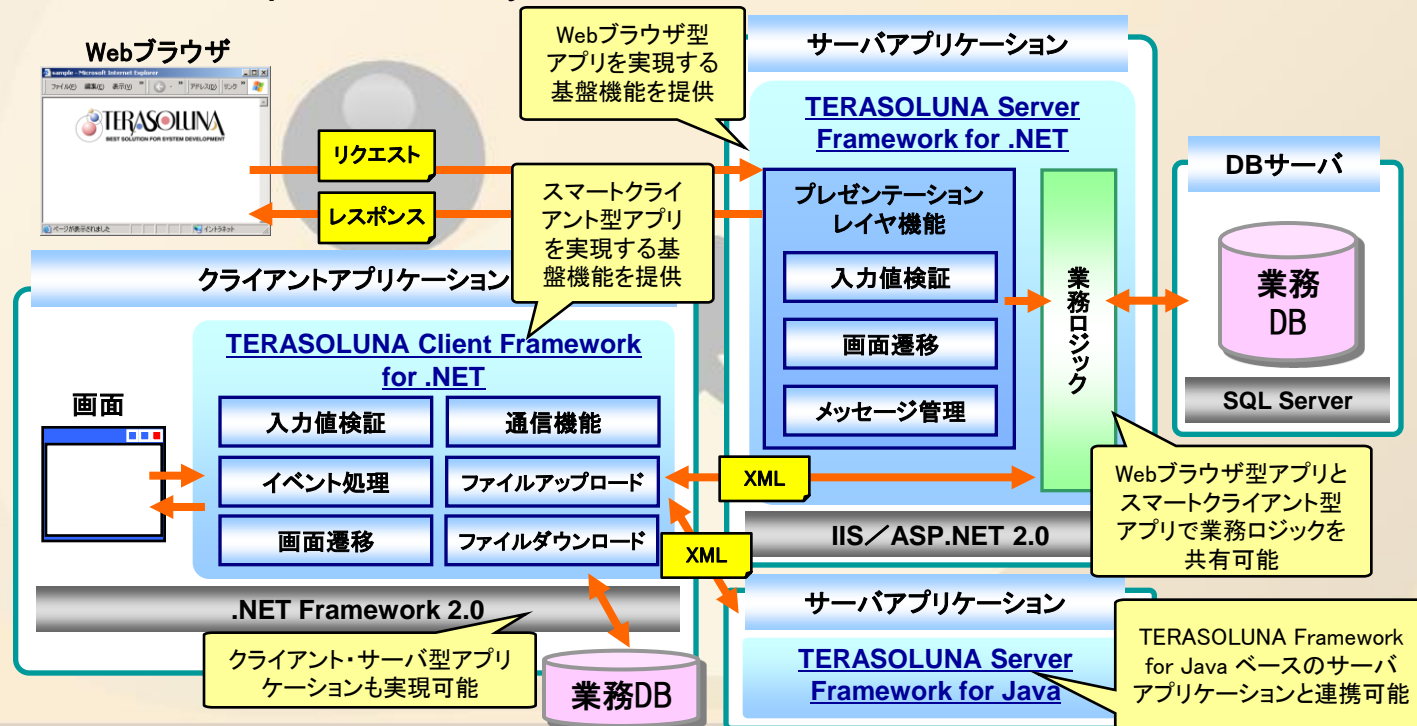
- 強力なWindowsFormsやASP.NETでの開発において、どのプロジェクトでも毎回同じような共通機能を整備

- 画面遷移機能の強化
 - セッション管理機能の整備
 - ログ出力機能の整備

TERASOLUNA for .NETが解決

TERASOLUNA Framework for .NET概要

- TERASOLUNA Server/Client Framework for .NET
 - Webアプリケーション、スマートクライアントアプリケーション開発に必要な機能を備えた汎用フレームワーク
 - 一部Enterprise Library 3.1を利用・拡張して機能を提供



TERASOLUNA Framework for .NETの狙い

- TERASOLUNA Framework for .NETの狙い
 1. **アーキテクチャの統一**
 2. **学習コストの最小化**
 3. **.NET Frameworkに足りない機能の強化**

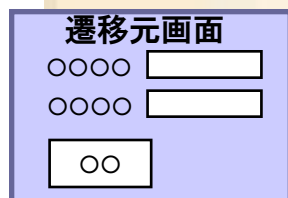
1. アーキテクチャの統一

・ アーキテクチャの統一

– クライアント業務画面からサーバ業務処理を呼び出す処理の定型化

- サーバへ送信するデータの入力値検証
- サーバとの通信処理
- サーバ業務処理呼び出し中のエラーハンドリング

画面上の情報を利用してサーバの業務処理を呼び出し、結果を取得する



①サーバ業務
処理の呼び出し

FA-02 拡張フォーム機能

FB-01 イベント処理機能

CM-02
入力値
検証機能

入力値検証
設定ファイル

FB-02
データセット
変換機能

データセット変
換設定
ファイル

CM-04
ビジネスロジック
生成機能

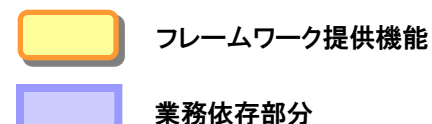
ビジネスロジック
設定ファイル

FC-01 XML通信機能
FC-02 ファイルアップロード機能
FC-03 ファイルダウンロード機能



APサーバ

サーバ
業務処理



2. 学習コストの最小化(1/2)

学習コストの最小化

- Visual Studioでの従来の開発スタイルを崩さない形で、フレームワークとして提供

① ツールボックスから、EventControllerを画面に追加

② EventControllerのプロパティを設定

③ EventControllerのExecuteメソッドを呼ぶだけ

```
// イベントコントローラを用いてイベント処理を実行する  
ExecutionResult result = eventController1.Execute();
```

2. 学習コストの最小化(2/2)

– WindowsForms/ASP.NETのアーキテクチャをそのまま踏襲

– WindowsForms/ASP.NETでの開発で本当に必要な共通機能のみをユーティリティとして提供しているため、TERASOLUNAフレームワークとしてのアーキテクチャを始めから学習する必要がない

**.NETの学習さえすれば、
すぐにTERASOLUNAの機能を利用した
アプリケーション開発ができる**

3. .NET Frameworkに足りない機能の強化(1/3)

・ .NET Frameworkに足りない機能の強化

- エンタープライズ向けアプリケーション開発に不足している機能をユーティリティとして提供
 - ユーティリティとしての機能提供のため、プロジェクトで本当に必要な一部の機能のみを利用できる
 - 非同期での業務処理呼び出しのサポート (Client FW)
 - 入力値検証における日本語のルールの強化
 - その他、ログ出力・メッセージ管理など

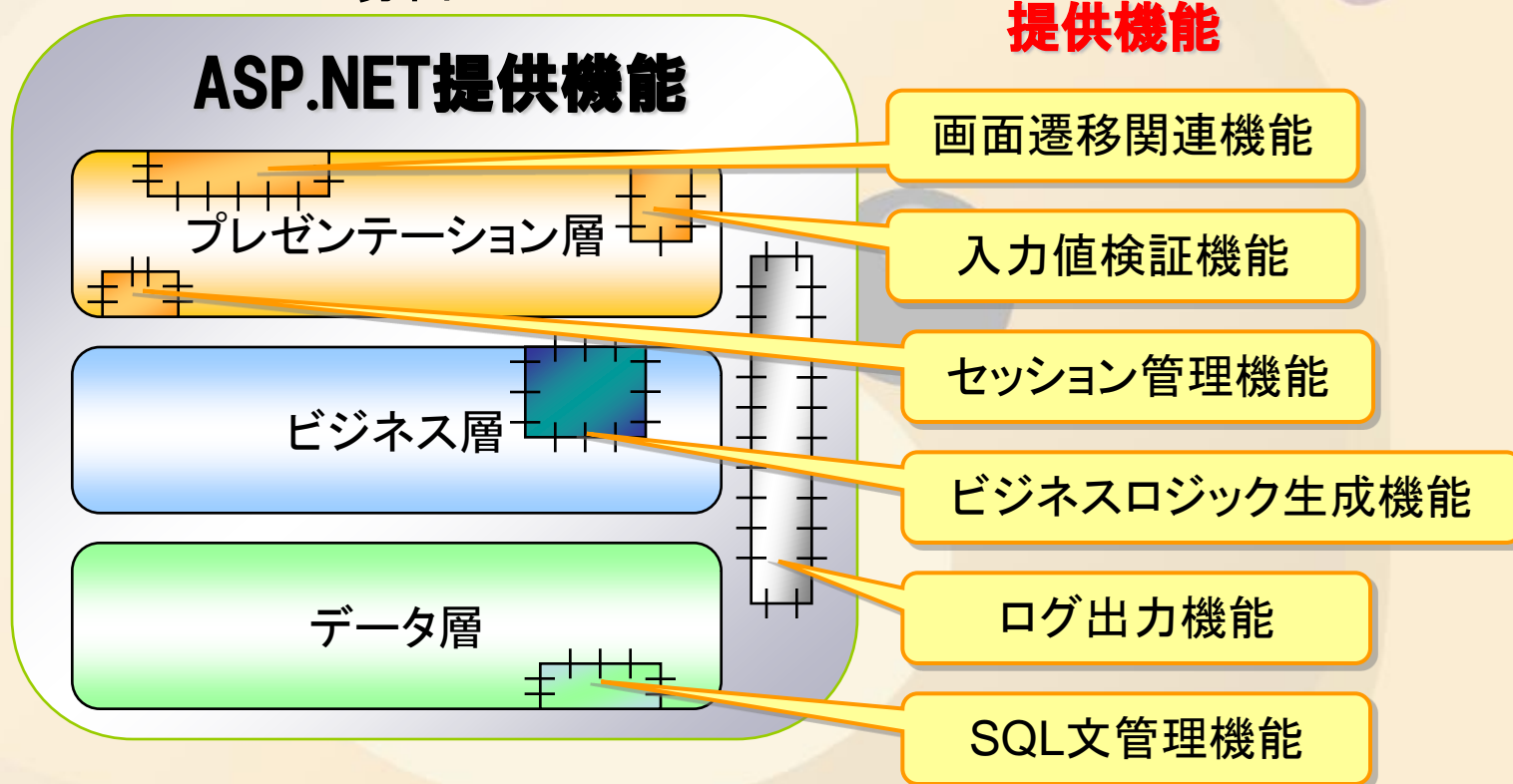
**プロジェクトで独自に整備した
共通機能との共存が可能！**

**TERASOLUNAの機能を一部利用する際にも、
WindowsForms/ASP.NETのアーキテクチャに変更は発生しない**



3. .NET Frameworkに足りない機能の強化(2/3)

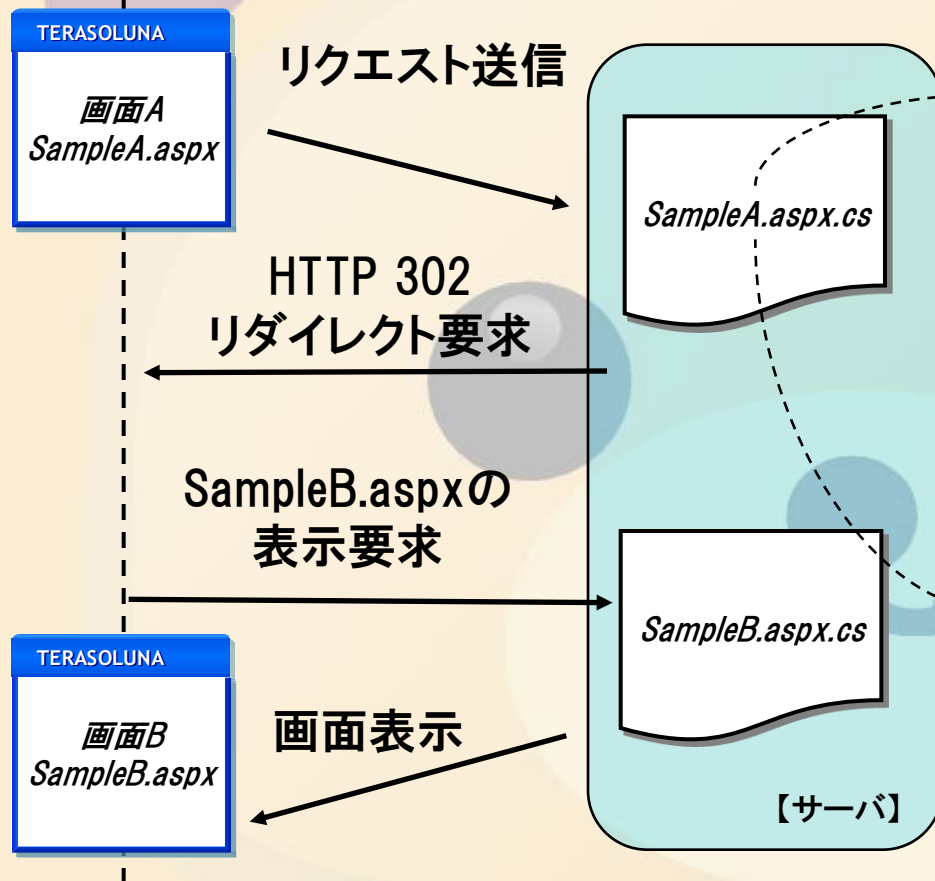
- TERASOLUNA提供機能のイメージ
 - 例: Server FWの場合



ASP.NETの穴はTERASOLUNAが埋める！！

3. .NET Frameworkに足りない機能の強化(3/3)

「画面遷移管理」の例



ASP.NET標準の画面遷移

```
Response.Redirect("~/UI/SampleB.aspx");
```

リダイレクト先URLを文字列として指定しなければならないため、画面間の依存関係が強くなる。

TERASOLUNAの画面遷移

```
WebUtils.Transit("SampleB");
```

ページIDを指定し、画面間の依存関係を疎にしている。遷移先画面のURLは設定ファイルに外出しているため、遷移先画面の変更も容易。

(参考)機能一覧

Server & Client共通機能

メッセージ管理機能
入力値検証機能
ログ出力機能
ビジネスロジック生成機能

Client 機能

画面遷移機能
拡張フォーム機能
イベント処理機能
データセット変換機能
XML通信機能
ファイルアップロード機能
ファイルダウンロード機能

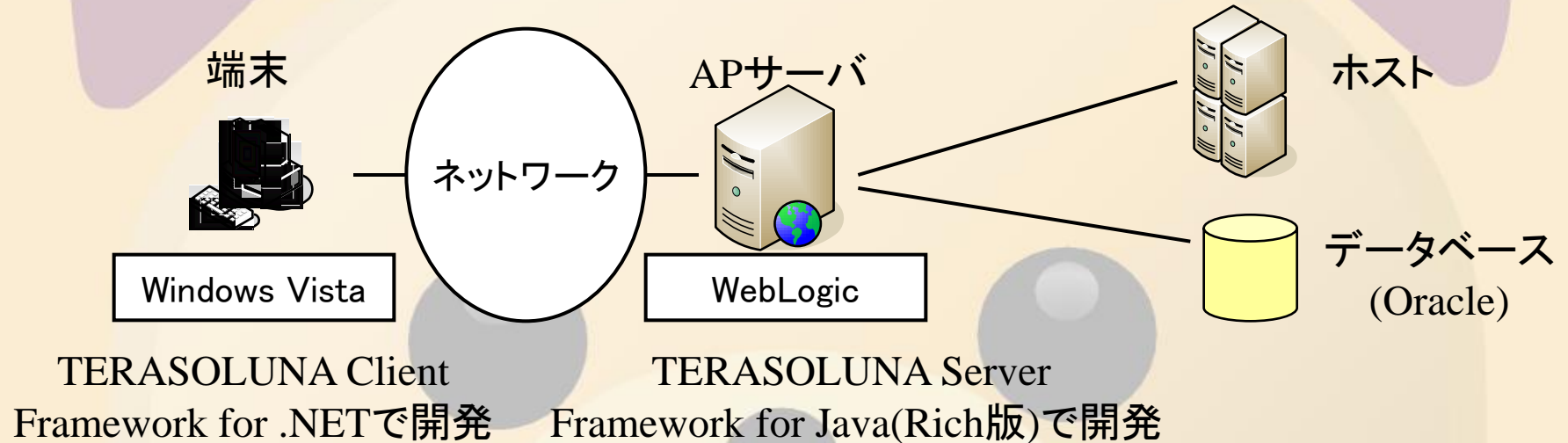
Server 機能

画面遷移管理機能
画面遷移保証機能
二重押下防止機能
エラー画面遷移機能
リクエストコントローラ機能*
ファイルアップロード機能*
ファイルダウンロード機能*
セッション管理機能
SQL文管理機能

* リッチクライアントAP用機能

導入事例

- クライアント.NET + サーバJavaによるスマートクライアントシステム



– ポイント

- クライアントでデバイス操作を行うため.NETを選択
- サーバ側のアーキテクチャを既存システムに合わせるため、Javaでの開発を選択

– 製造上の工夫点

- サーバ・クライアント間で通信に用いるXMLデータ設計情報を共有し、DTO(データ転送オブジェクト)クラスをおのこの自動生成するツールを事前に用意

Agenda

- はじめに
- TERASOLUNA for .NET概要
- デモ
- まとめ

デモ

- TERASOLUNA Server/Client Framework for .NETによるスマートクライアント開発のデモ

Agenda

- はじめに
- TERASOLUNA for .NET概要
- デモ
- まとめ

まとめ

- TERASOLUNA Framework for .NET
 - 「アーキテクチャの統一」「学習コストの最小化」「.NET Frameworkに足りない機能の強化」を狙って開発したフレームワーク
 - NTTデータが数多くのシステム開発で培ってきた豊富なノウハウが継続的にフィードバック
 - 一部の機能をユーティリティ的に利用することも可能

業務システム開発に是非ご活用ください！

社外 Web サイト、フレームワーク公開サイトのお知らせ

- 2008年7月14日にTERASOLUNA Server/Client Framework for .NET をオープンソース化
 - サイト
<http://www.terasoluna.jp/>
 - SourceForge.JP TERASOLUNAフレームワーク プロジェクト
<http://sourceforge.jp/projects/terasoluna>

