

.NETでもオープンソースでシステム開発

2008年9月4日(木)

株式会社NTTデータ
技術開発本部
ソフトウェア工学推進センタ
永嶋浩樹



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。



Agenda

- 🌀 はじめに
- 🌀 .NETとオープンソース
- 🌀 支援ツール系オープンソースの紹介
- 🌀 クラスライブラリ系オープンソースの紹介
- 🌀 まとめ



Agenda

- 🌀 はじめに
- 🌀 .NETとオープンソース
- 🌀 支援ツール系オープンソースの紹介
- 🌀 クラスライブラリ系オープンソースの紹介
- 🌀 まとめ



自己紹介

永嶋浩樹(ながしまひろき)

- 株式会社NTTデータ 技術開発本部
ソフトウェア工学推進センタに所属
- 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
 - ◆ DB マガジン
「.NETコントロールまるわかり教室」連載中



私たちの役割

技術開発本部 ソフトウェア工学 推進センタ

ソフトウェア開発の生産性向上のための
ソフトウェア開発プロセス、生産技術、
人材育成の研究開発と普及展開

ソリューションの
提供

NTTデータ

公共

金融

法人

NTTデータグループ会社

🌐 システム開発の総合的ソリューション「**TERASOLUNA[®]**」
の研究開発・普及展開



TERASOLUNAフレームワークラインアップ

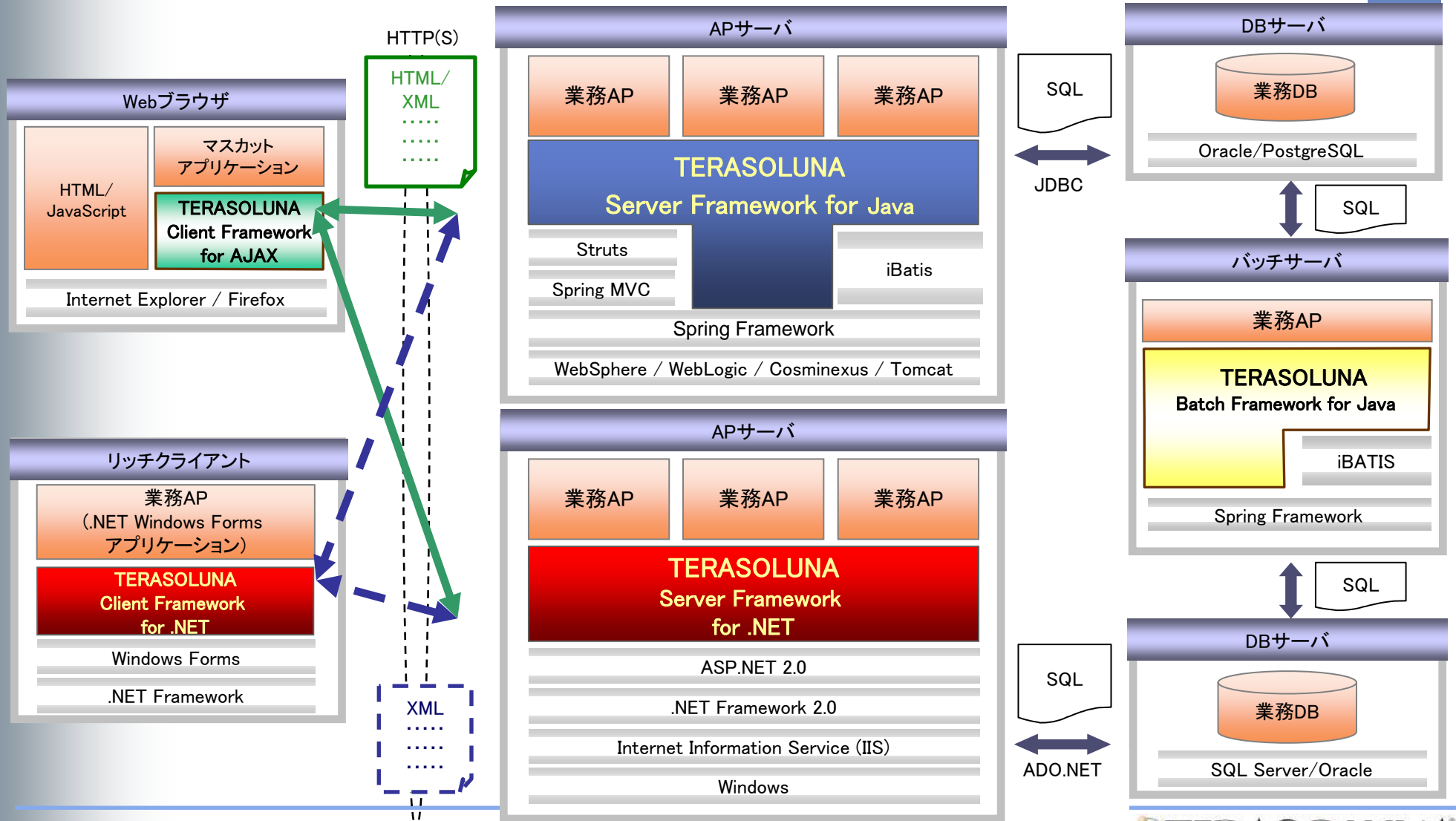
🌐 開発環境(フレームワーク)

- 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
- すべてオープンソースとして無料でダウンロード
- ソフトウェアアーキテクチャを定義し、品質のばらつきを抑制
- フレームワークの利用で、業務開発者は、業務ロジックの開発のみに集中可能

	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
Java	TERASOLUNA Server Framework for Java	—	TERASOLUNA Batch Framework for Java
.NET	TERASOLUNA Server Framework for .NET	TERASOLUNA Client Framework for .NET	—
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
- 🌀 .NETとオープンソース
- 🌀 支援ツール系オープンソースの紹介
- 🌀 クラスライブラリ系オープンソースの紹介
- 🌀 まとめ



Javaにおけるオープンソース

設計

製造/単体テスト

結合テスト

クラス
ライブラリ

フレームワーク/クラスライブラリ

Struts

Spring

iBATIS

支援
ツール

統合開発環境

Eclipse

製造支援

WTP

Spring IDE

Visual Editor

静的解析

FindBugs

CheckStyle

PMD

単体テスト支援

JUnit

Cobertura

StepCounter

Eclipse Metrics Plugin

回帰テスト支援

Selenium

Marathon

品質管理支援

タスク・故障管理

Trac

構成管理

Subversion



.NETにおけるオープンソース

設計

製造/単体テスト

結合テスト

クラス
ライブラリ

フレームワーク/クラスライブラリ

Enterprise Library

AJAX Control Toolkit

統合開発環境

Visual Studio 2008

製造支援

Sandcastle

静的解析

コード分析
機能

単体テスト支援

単体テスト

NUnit

回帰テスト支援

Webテスト

品質管理支援

カバレッジ計測機能

タスク・故障管理

Trac

Team Foundation Server

構成管理

Subversion

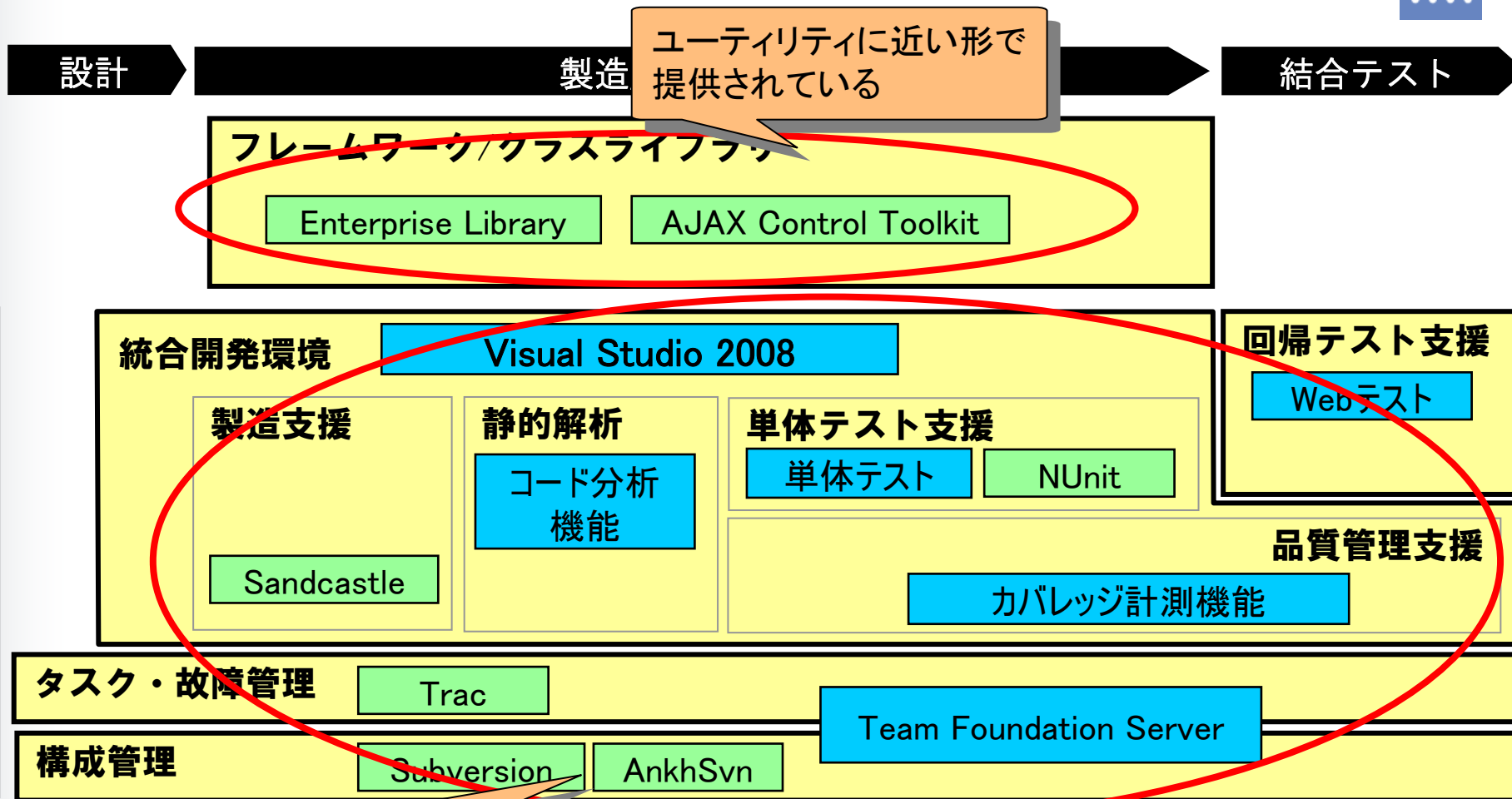
AnkhSvn

■ : Visual Studio ■ : オープンソース

支援
ツール



.NETにおけるオープンソース



■ : Visual Studio ■ : オープンソース

Visual Studio に豊富な機能が存在するが、上位Editionにしか利用できない機能を下位Editionで利用できるようにする形で提供されている



Agenda

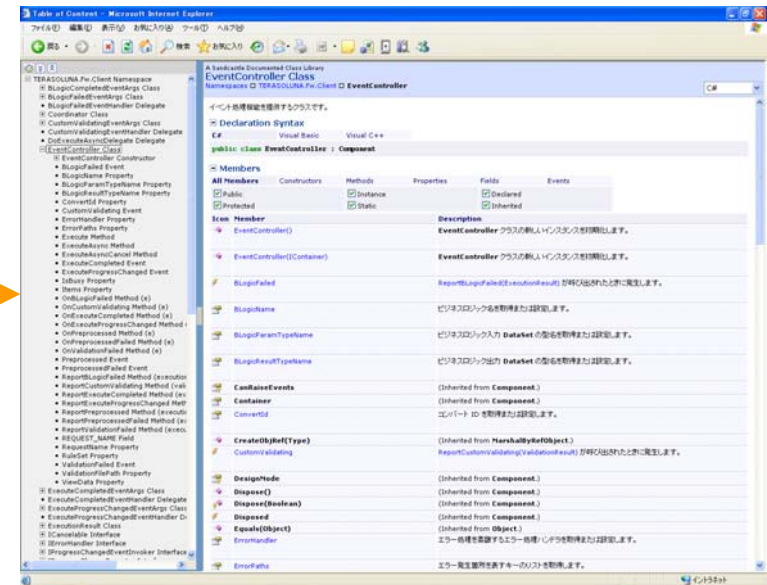
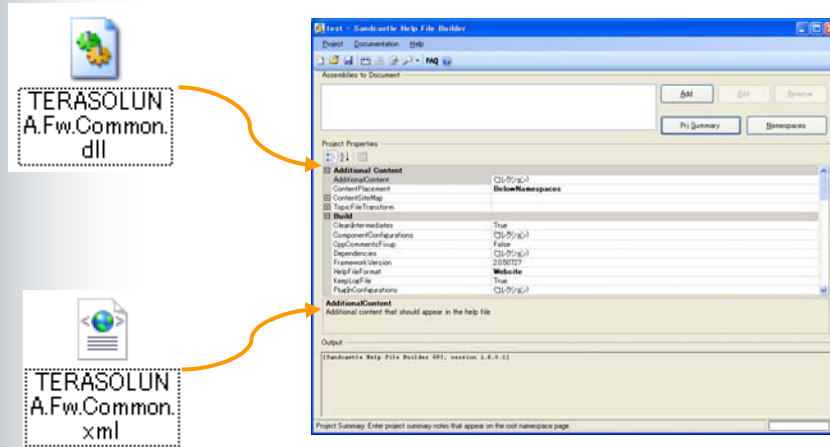
- 🌀 はじめに
- 🌀 .NETとオープンソース
- 🌀 支援ツール系オープンソースの紹介
- 🌀 クラスライブラリ系オープンソースの紹介
- 🌀 まとめ



APIリファレンス作成


 Sandcastle (<http://www.codeplex.com/Sandcastle>)

- MSDN風のAPIリファレンスを作成するツール
- JavaのJavadocに相当



Sandcastle Help File Builderを使って
APIリファレンスを作成




 NUnit (<http://www.nunit.org/>)

- 単体テストの自動実行を支援するためのツール
- JavaのJUnitに相当



構成管理、タスク・問題管理

 Trac (<http://trac.edgewall.org/>)



- Pythonベースのバグトラッキングシステム
- Wiki機能やSubversion連携機能もついている

 Subversion (<http://subversion.tigris.org/>)



- プログラムのソースコードなどを管理するバージョン管理システム
- HTTPでのアクセスや、チェンジセット単位でのコミットが可能

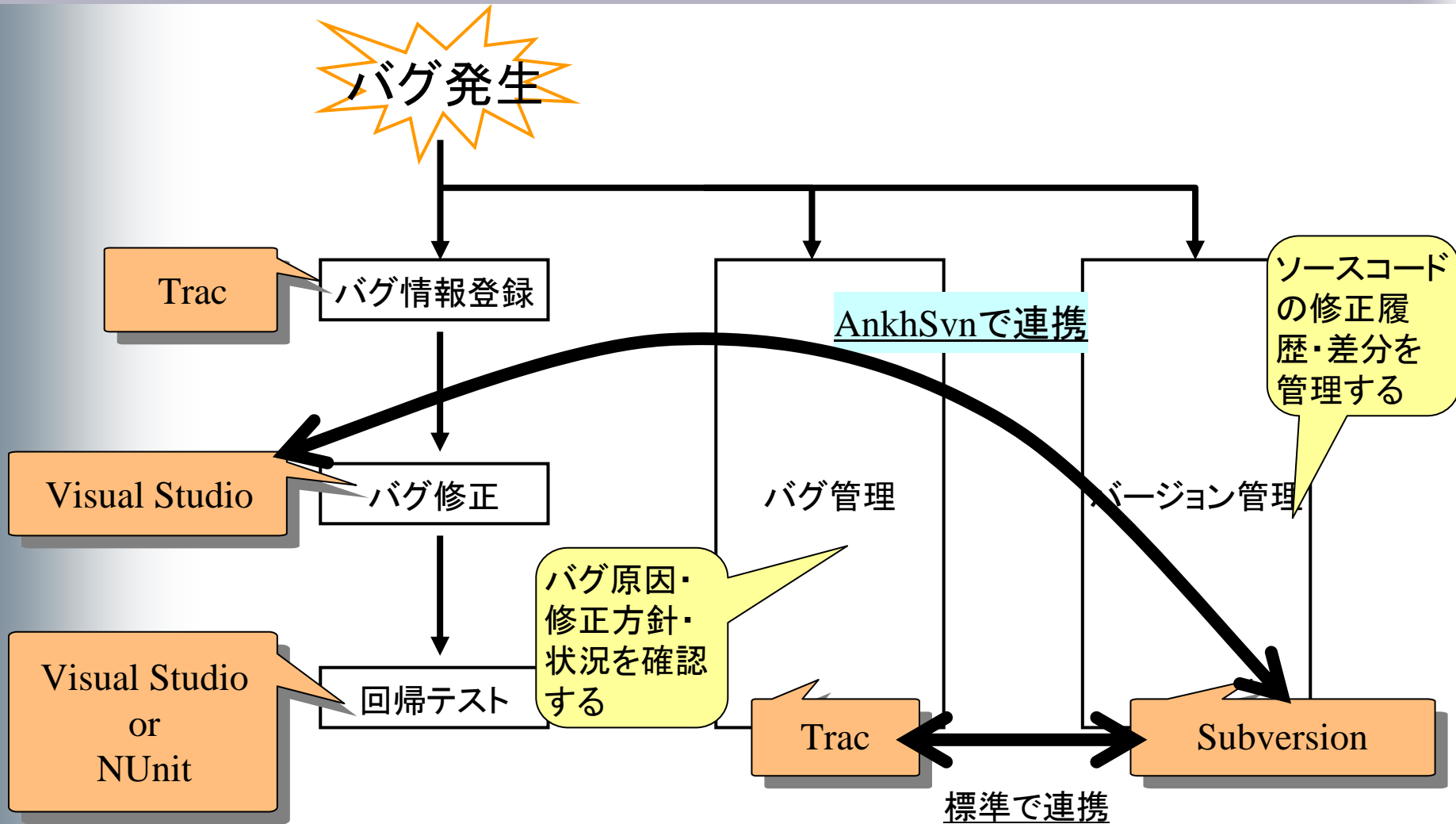
 AnkhSvn (<http://ankhsvn.open.collab.net/>)



- Visual StudioとSubversionを連携させるためのプラグイン



構成管理、タスク・問題管理



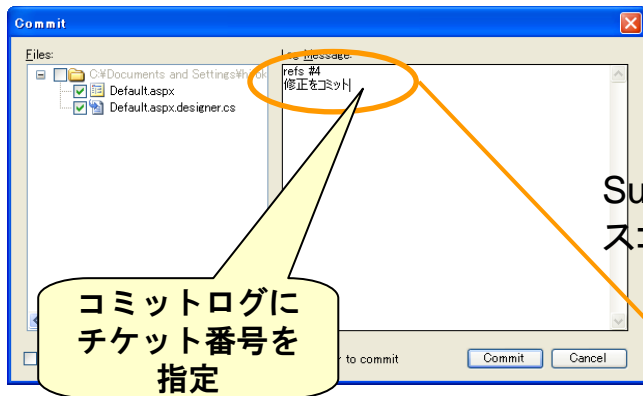


構成管理、タスク・問題管理

AnkhSvn

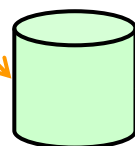
Trac

Visual Studio
から起動

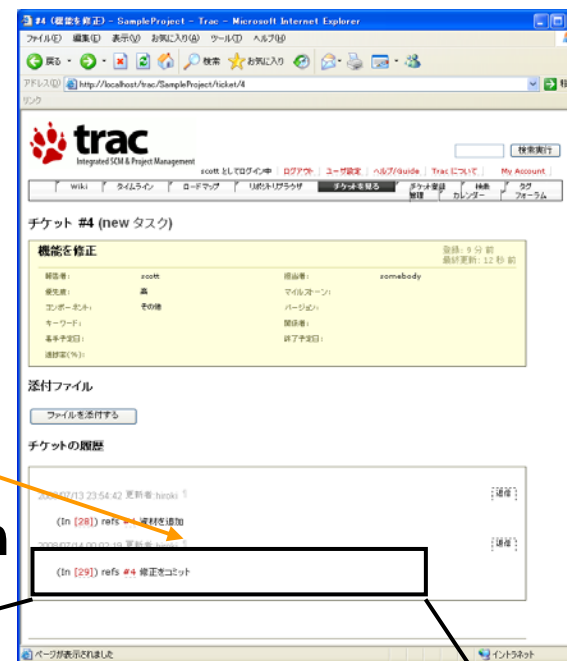


コミットログに
チケット番号を
指定

Subversionにソー
スコードをコミット



Subversion



対応するチケット*にコメントが自動
追加される

2008/07/14 00:02:19 更新者:hiroki
(In [29]) refs #4 修正をコミット

変更セットへの
リンク

※TracとSubversionの連携機能

*チケット:
タスクやバグ情報をあらわすTracの最小単位。
Team Foundation Serverの「作業項目」に相当



Agenda

- 🌀 はじめに
- 🌀 .NETとオープンソース
- 🌀 支援ツール系オープンソースの紹介
- 🌀 クラスライブラリ系オープンソースの紹介
- 🌀 まとめ



Enterprise Library (<http://www.codeplex.com/entlib>)

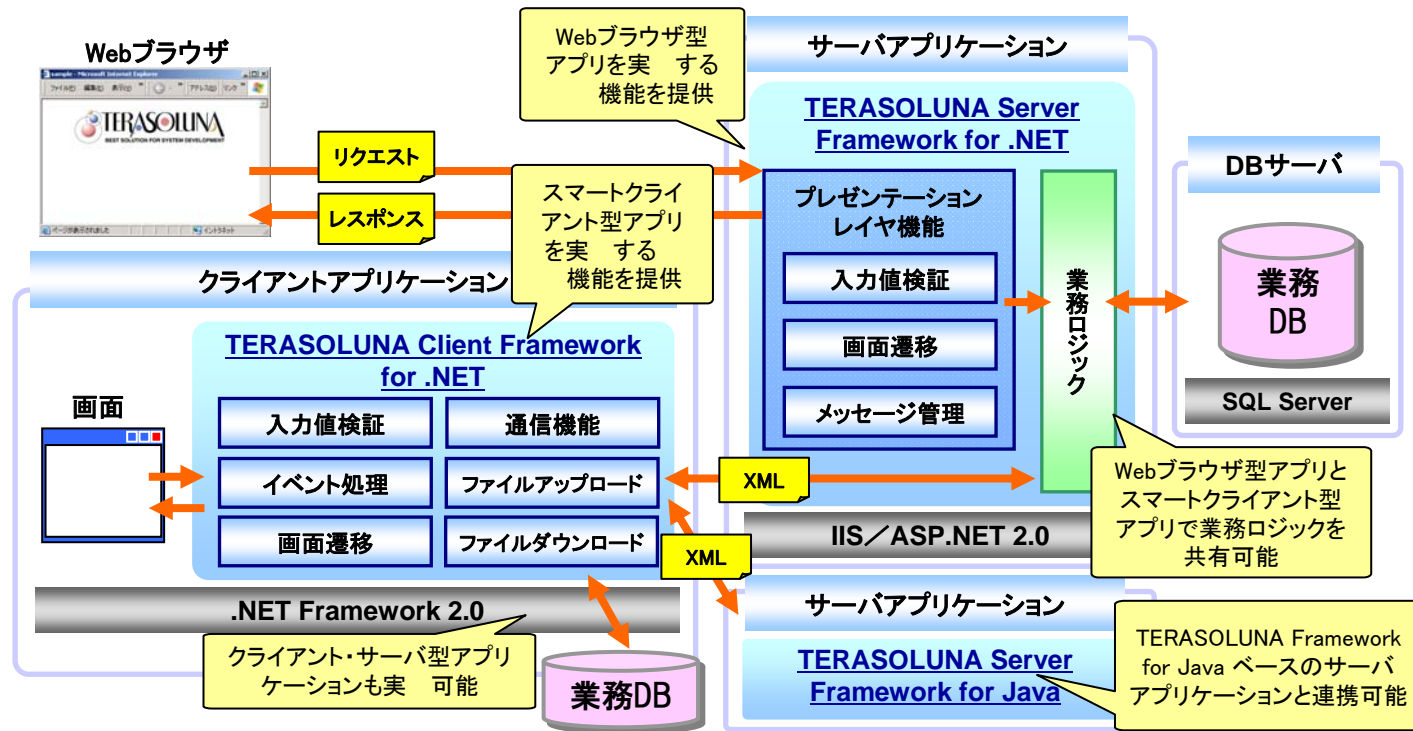
- 企業向けアプリケーション開発に利用できる再利用性の高いクラスライブラリ
 - ◆ ログ出力機能、入力値検証機能など



TERASOLUNA Framework for .NET

TERASOLUNA 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





.NETのみのクライアント・アプリ開発の問題点

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



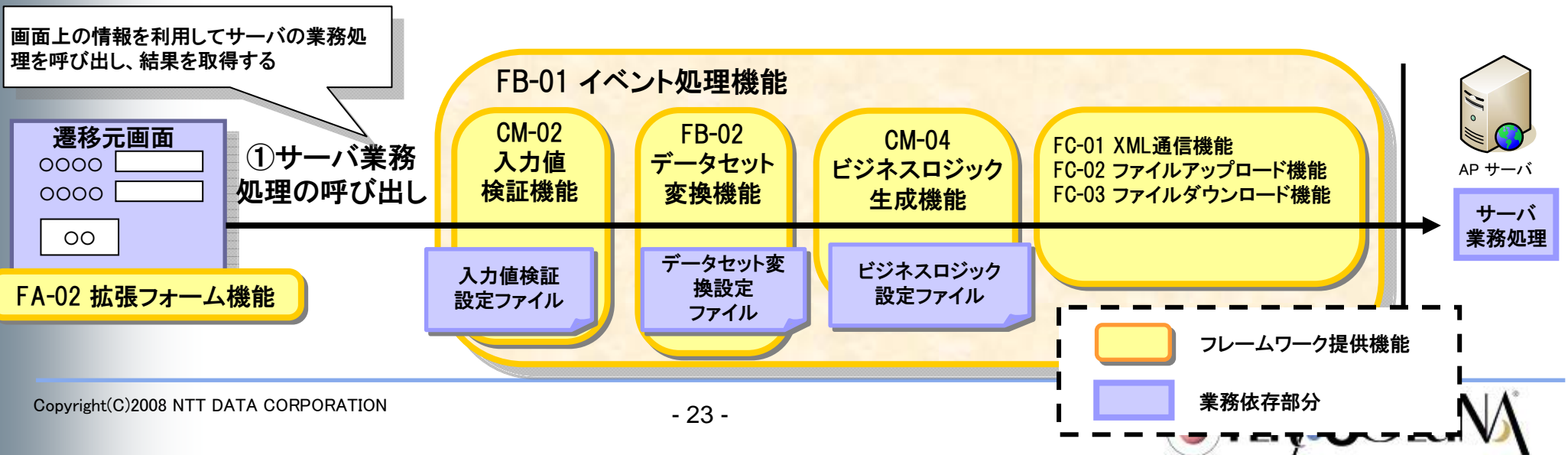
- 🌐 TERASOLUNA Framework for .NETの狙い
 1. **アーキテクチャの統一**
 2. **学習コストの最小化**
 3. **.NET Frameworkに足りない機能の強化**



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





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メソッドを呼びだけ

```
// 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！**

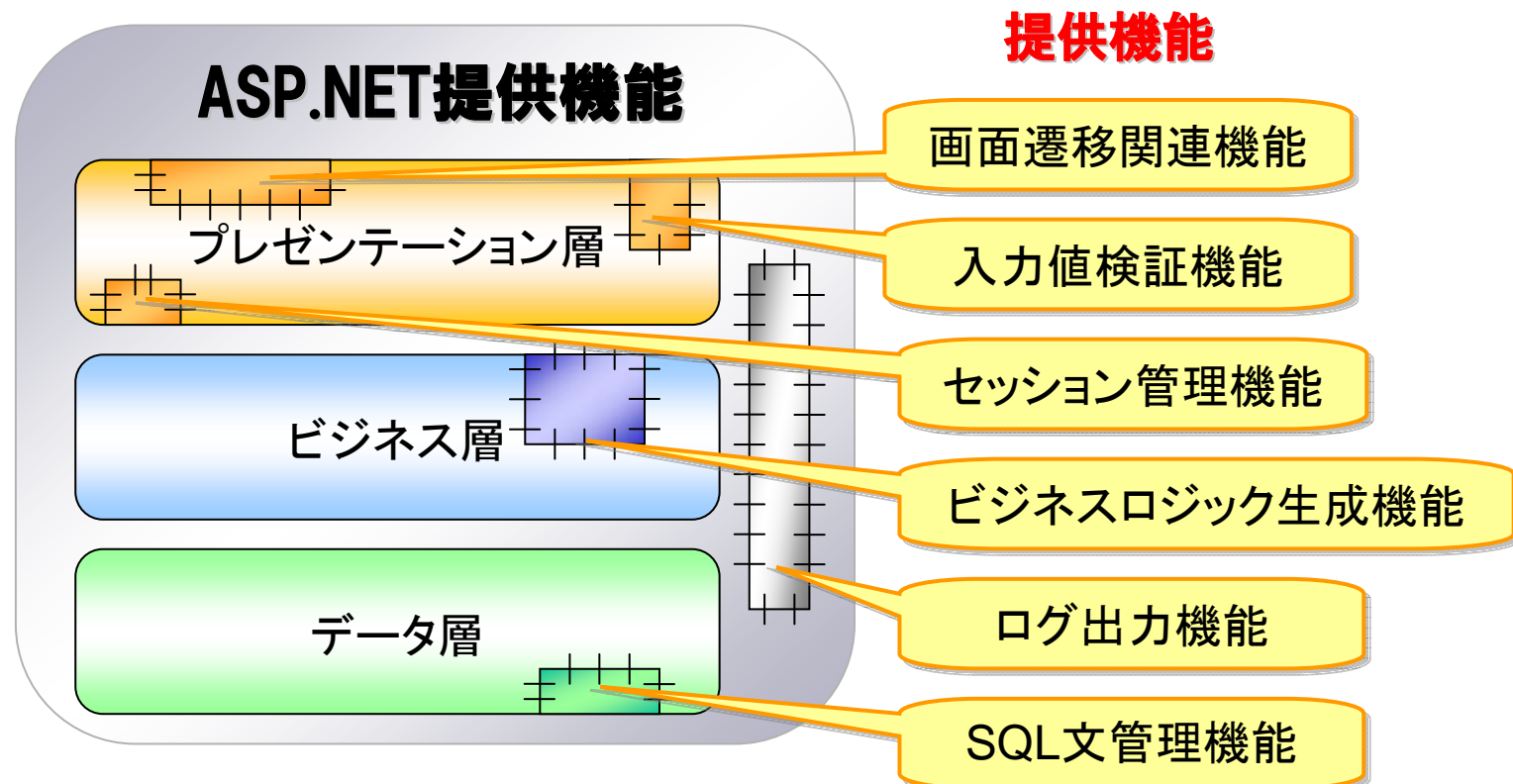
**TERASOLUNAの機能を一部利用する際にも、
WindowsForms/ASP.NETのアーキテクチャに変更は発生しない**



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

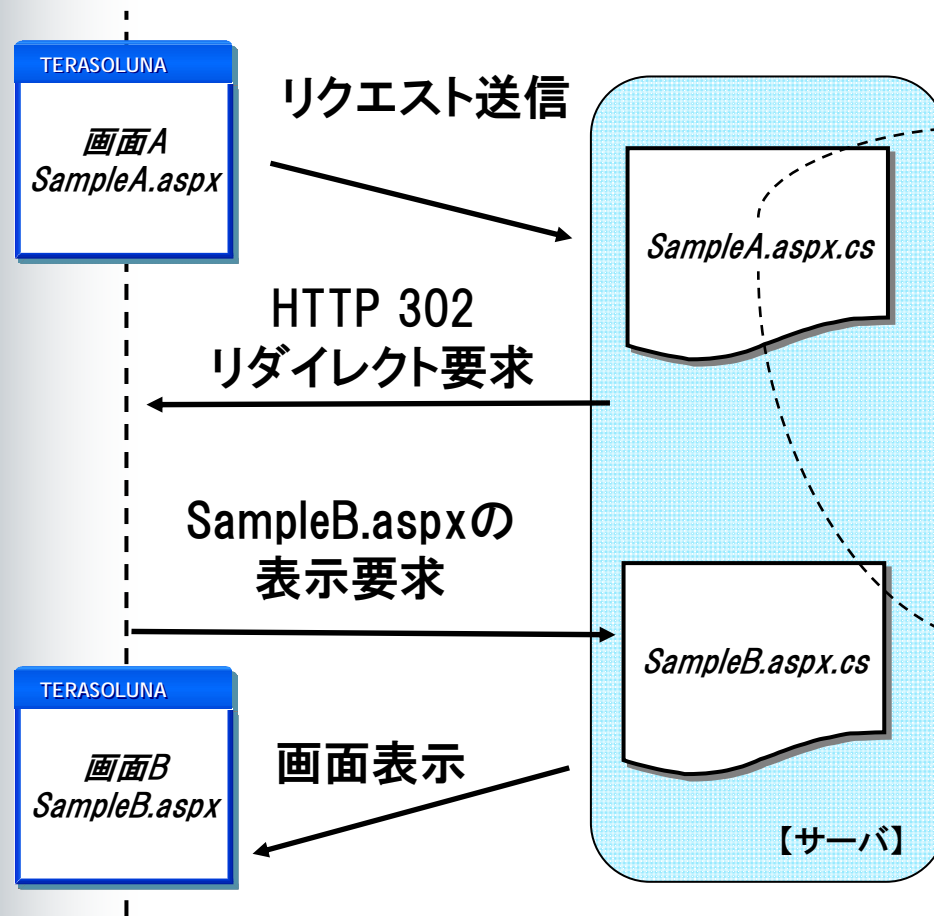


ASP.NETの穴はTERASOLUNAが埋める！！



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。



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

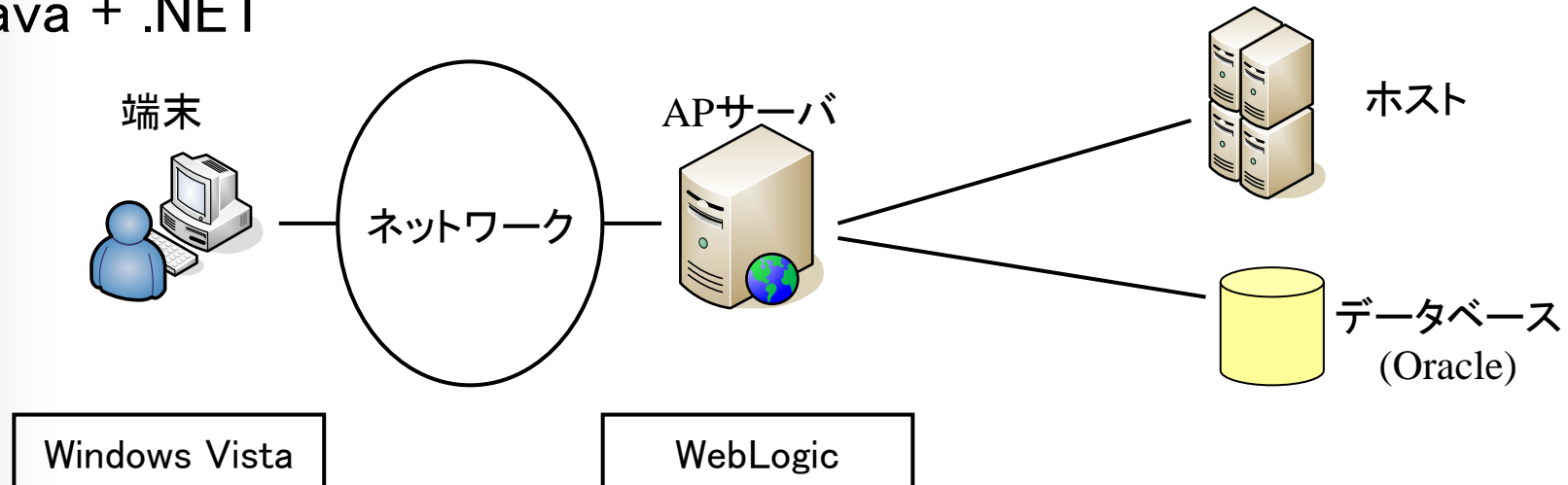
Server 機能
画面遷移管理機能
画面遷移保証機能
二重押下防止機能
エラー画面遷移機能
リクエストコントローラ機能*
ファイルアップロード機能*
ファイルダウンロード機能*
セッション管理機能
SQL文管理機能

* リッチクライアントAP用機能



導入事例

Java + .NET



■ ポイント

- ◆ サーバは非MS製品 → サーバはJava
- ◆ クライアントでデバイス操作を行う → クライアントを.NET

■ 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
- 🌀 .NETとオープンソース
- 🌀 支援ツール系オープンソースの紹介
- 🌀 クラスライブラリ系オープンソースの紹介
- 🌀 まとめ



ツール系

- オープンソースを積極的に利用して、上位Editionの Visual Studioで出来る作業を実施する
 - ◆ 単体テストや構成管理を必ず実施

ライブラリ系

- .NETのアーキテクチャ・標準開発スタイルを踏襲し、 Visual Studioの開発生産性をうまく利用できるように、機能を組み合わせる
 - ◆ ユーティリティ系を取捨選択
 - ◆ アーキテクチャの統一



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>





ご清聴ありがとうございました。