

業務システム構築基盤のご紹介

SystemDirector Enterprise for .NET

日本電気株式会社

本資料では SystemDirector Enterprise for .NET を、「for .NET」と略記することがあります。

目次

第1章 SystemDirector Enterprise とは

- 1.1. SystemDirector Enterprise の狙い
- 1.2. SystemDirector Enterprise の体系
- 1.3. SystemDirector Enterprise の構成要素
- 1.4. SystemDirector Enterprise の製品ラインナップ

第2章 開発方法論

- 2.1. 概要
- 2.2. 一般的な開発方法論との違い
- 2.3. 共通フレーム対応
- 2.4. フロントローディング

第3章 開発環境

- 3.1. 提供機能のポイント
- 3.2. アーキテクチャモデル
- 3.3. フレームワーク
- 3.4. 開発ツール
- 3.5. 動作環境

第4章 サポートサービス

- 4.1. サポートメニュー
- 4.2. 教育メニュー
- 4.3. お問い合わせ先

第 1 章 SystemDirector Enterprise とは

1.1.SystemDirector Enterprise の狙い

環境認識

技術の高度化、
新技術の台頭

短納期・
スピード重視

長期間の
保守メンテナンス

ニーズ

「信頼できるSI」

先端技術の活用
迅速な対応

生産性の向上

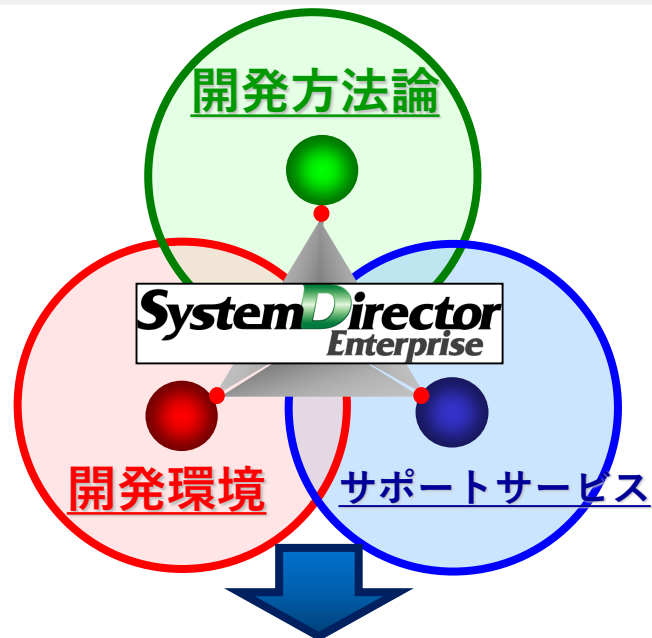
十分な期間の
サポート提供

これらを実践するシステム構築環境を整備
今までのNECグループのシステム構築技術を結集
SystemDirector Enterprise の開発へ

1.2.SystemDirector Enterprise の体系

NECグループ標準の業務システム向け統合開発環境

開発方法論、開発環境、サポートサービスにより、効率的なシステム構築をサポート



◇開発方法論（誰が、いつ、何をするのか）

- 概説書、手順書、ガイド、ドキュメントサンプル集
- 各工程の作業手順をサポート

◇開発環境（何を作るか、どうやって作るのか）

- 業界標準アーキテクチャを採用したフレームワーク
- 開発効率を向上する開発ツールの提供

◇サポートサービス（どのように利用するか）

- システム構築をさまざまな形でサポート
- トータル10年間のソフトウェア製品サポートを提供

先端技術の活用

- ・ 開発環境で取り込み
ガイドも添えて提供

生産性の向上

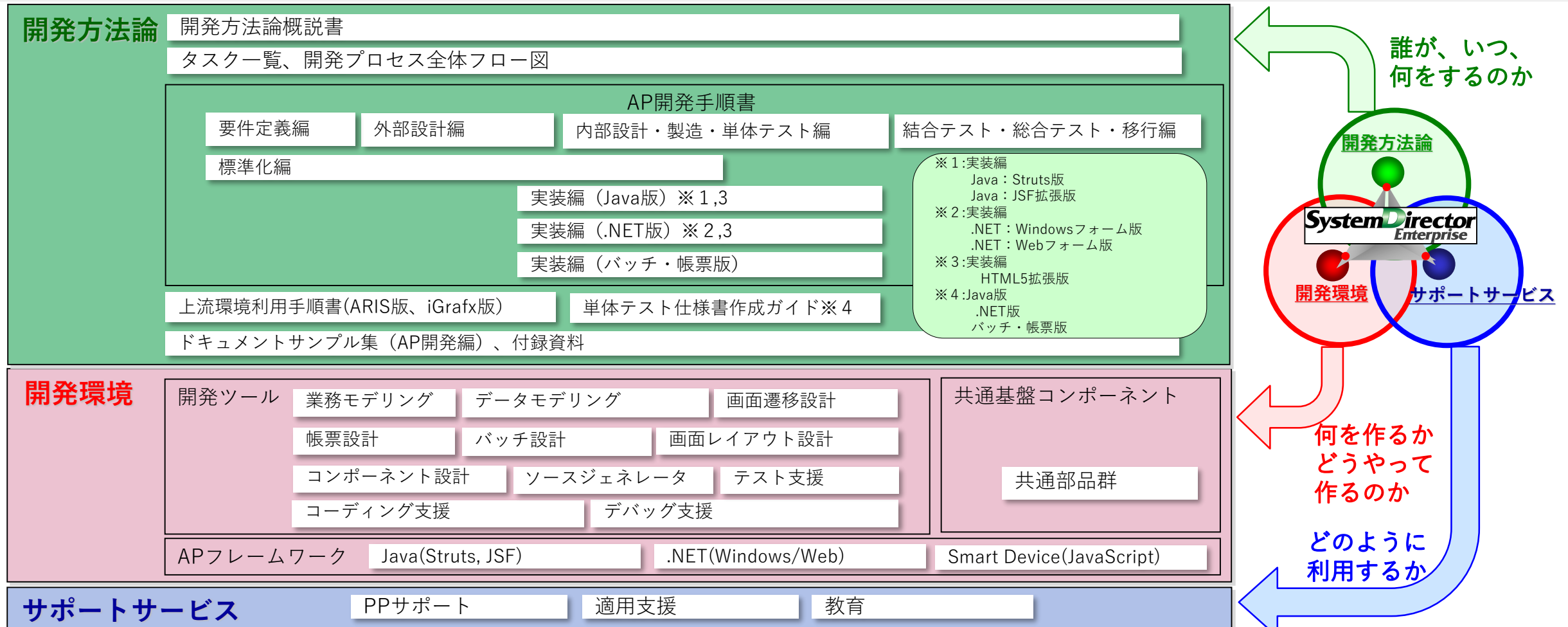
- ・ 開発方法論で標準作業の
効率化
- ・ 開発環境で生産性向上

サポート提供

- ・ 専任の技術サポート
部隊を用意
- ・ 長期サポート保証

1.3.SystemDirector Enterprise の構成要素

業務システムを構築するために必要十分な支援内容を提供します。



1.4.SystemDirector Enterprise の製品ラインナップ

- 要件定義～テストをサポートする業務システム構築基盤と、保守～再構築をサポートする業務アプリケーション分析基盤を提供
- ソフトウェアライフサイクル全般をサポートする製品群として継続強化中

業務システム構築基盤

オンラインシステム開発基盤

SystemDirector Enterprise for Java (オンライン)

SystemDirector Enterprise for .NET

SystemDirector Enterprise for Smart Device

本稿ではこちらの製品について紹介します。

バッチシステム開発基盤

SystemDirector Enterprise for Java (バッチ)

SystemDirector Enterprise for C Batch

業務アプリケーション分析基盤

SystemDirector Enterprise for Java (モダナイゼーション)

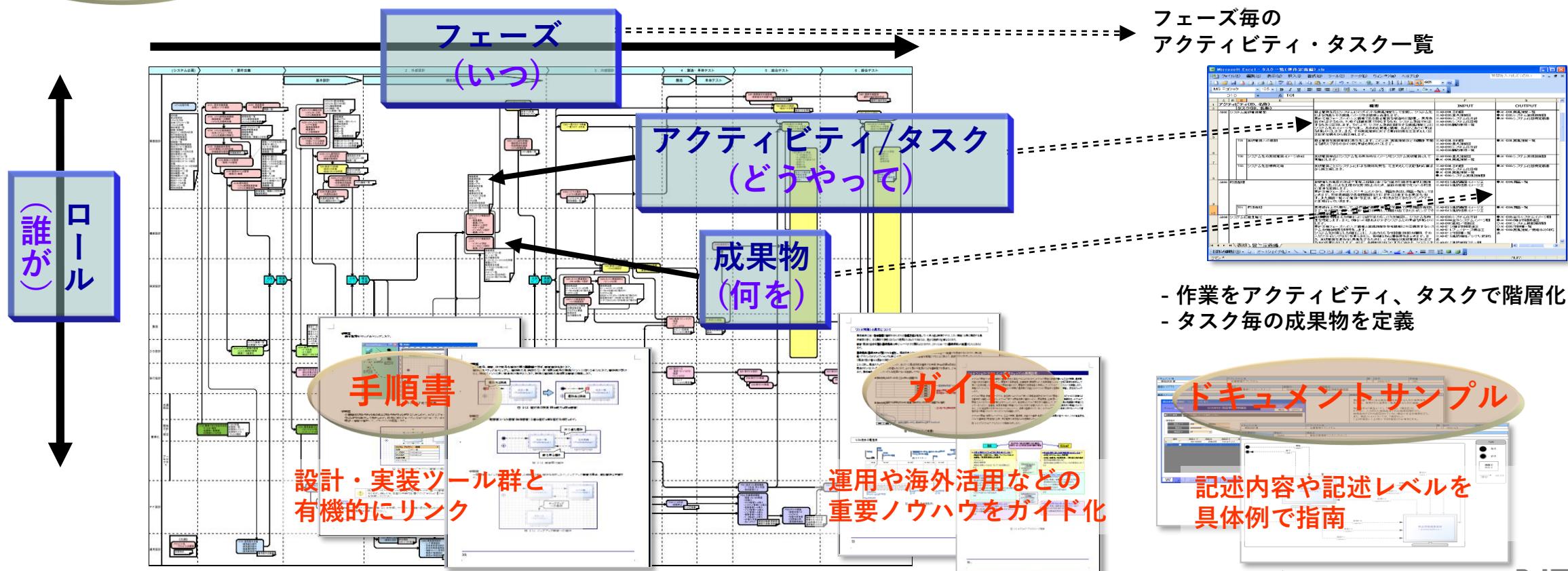
第2章 開発方法論

2.1.概要

開発プロセスに従った手順書、ドキュメントサンプルなどを提供します。

開発プロセス

SI現場の成功・失敗ノウハウを実践的な開発プロセスに結集
SIに必要な「いつ・誰が・何を・どうやって」を漏れなく重複なく体系化



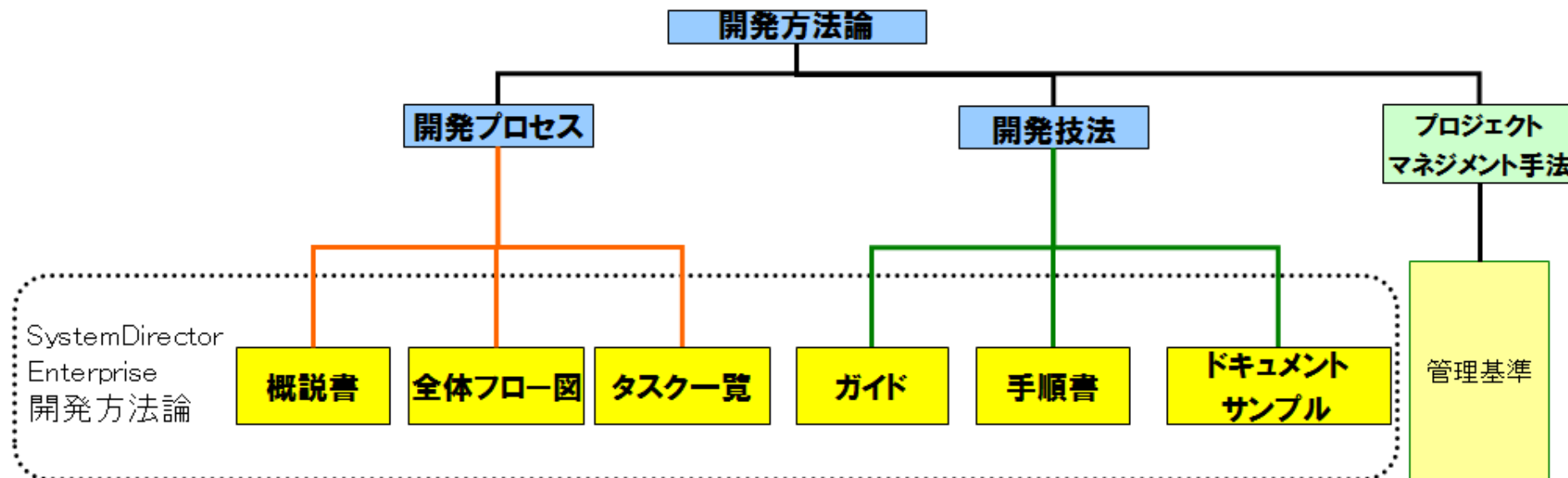
2.2.一般的な開発方法論との違い

役割を意識した、手順、成果物、規約などを体系的に定義します。

一般的に開発方法論は、以下の3つで構成されます。

- オブジェクト指向やデータ中心アプローチなどの具体的な開発手法やツール類を定めた「開発技法」
- 開発をどのような手順で進めるかを定めた「開発プロセス」
- プロジェクトマネジメント手法

SystemDirector Enterprise開発方法論は、オープン環境における業務システム開発をターゲットとして、ソフトウェアライフサイクルプロセス（SLCP）における「要件定義」「外部設計」「内部設計」「製造・単体テスト」「結合テスト」「総合テスト」「移行」の一連の開発プロセスを中心に、役割（ロール）ごとの作業手順、成果物（ドキュメント）、作業規約を体系的に定めた開発方法論です。



2.3.共通フレーム対応

業界標準に準拠した体系をサポートします。

本開発方法論で定義しているフェーズは、業界標準（共通フレーム2013：SLCP-JCF2013）に沿った体系となっています。本開発方法論は、共通フレーム2013の要件定義・開発プロセスをサポートしております。

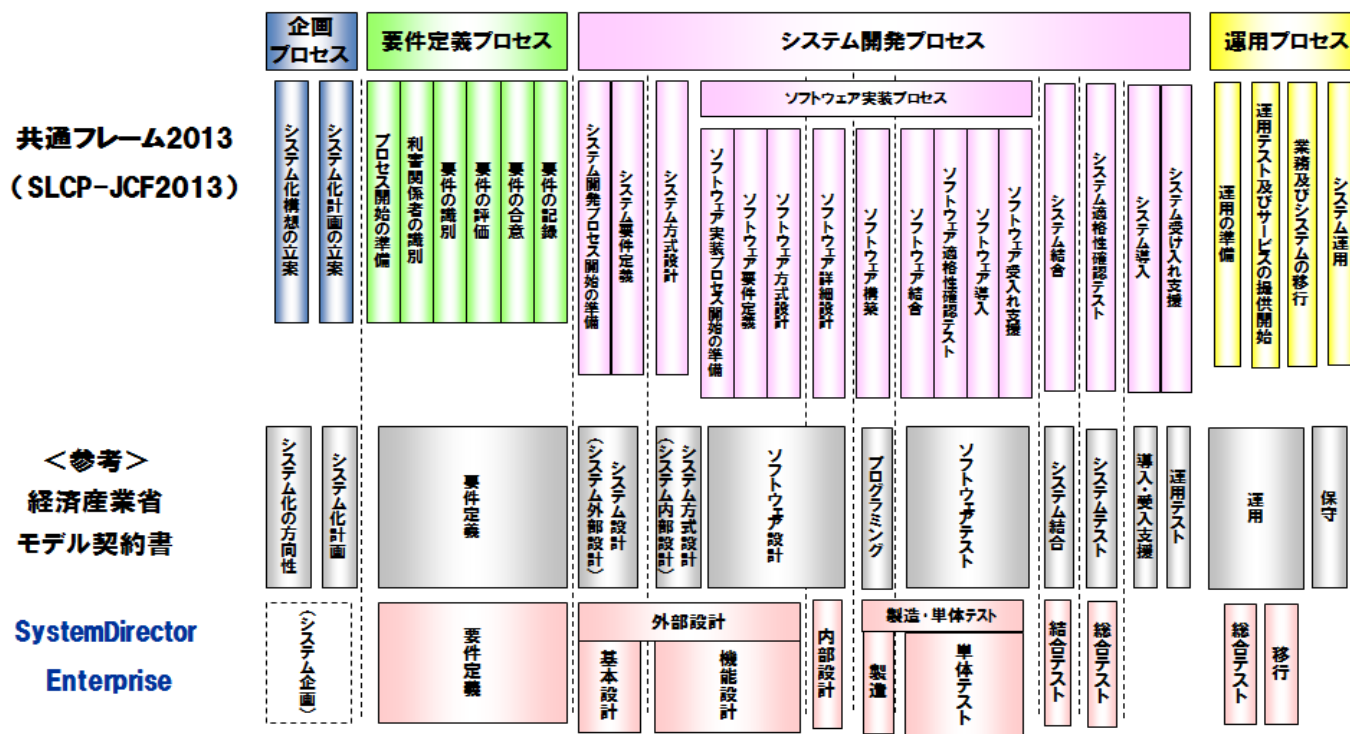


図1 プロセスレベルの対応関係

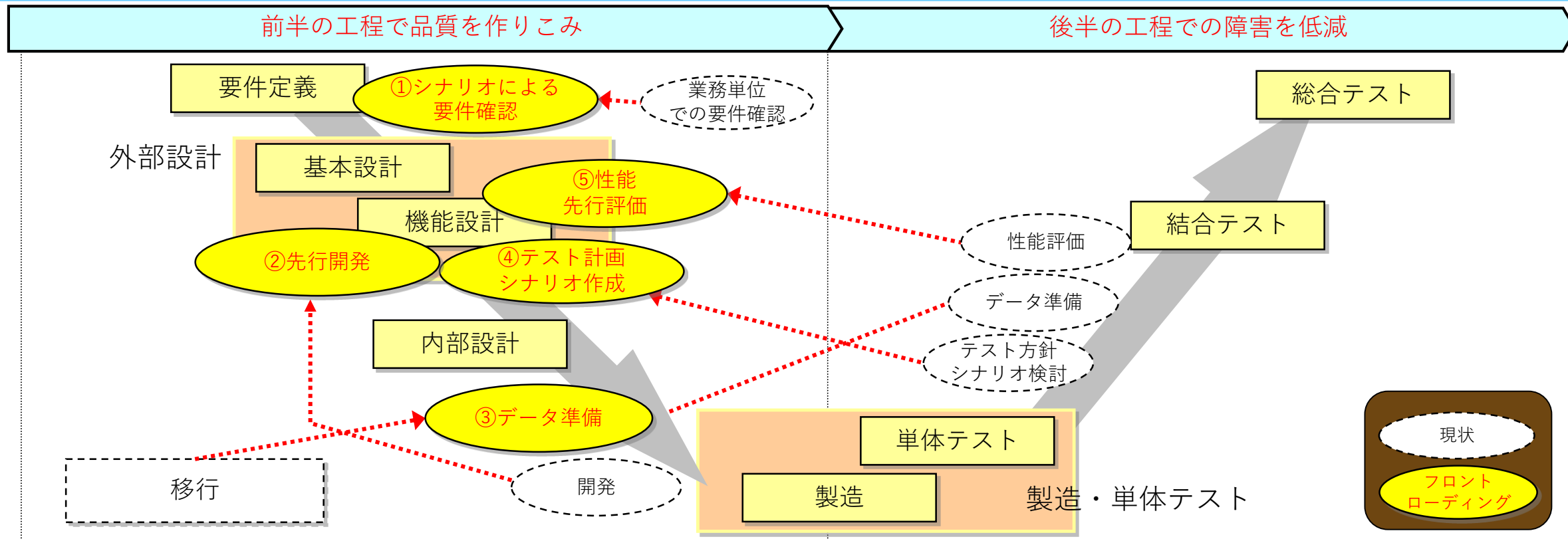
※ 共通フレーム2013との関係はウォーターフォールモデルをベースとし、フェーズ、作業順序を考慮して整理しています。

2.4.フロントローディング

開発の前半で品質を作りこむ手法を取り入れています。

「フロントローディング」とは？

開発の後半で発生する問題を未然に防ぐことを目的に、開発初期でやるべきことを正しく行い品質を作りこむ手法



第 3 章 開発環境

3.1.提供機能のポイント

Windows フォーム、ASP.NET Web フォーム/ASP.NET MVC、SPA を用いた業務システムの構築を総合的にサポートいたします。

Point 1

アーキテクチャモデル

- Microsoft 社ガイドライン「Application Architecture Guide 2.0」に準拠した推奨アーキテクチャを整備。
- スムーズな最新技術対応や、アーキテクチャ選定工数を削減。
- PR 層は Windows フォーム、ASP.NET/ASP.NET MVC、SPA から選択可能。

Point 2

フレームワーク

- Windows フォーム、ASP.NET Web フォーム/ASP.NET MVC を用いた業務システムの構築において、不可欠となる共通機能を提供。

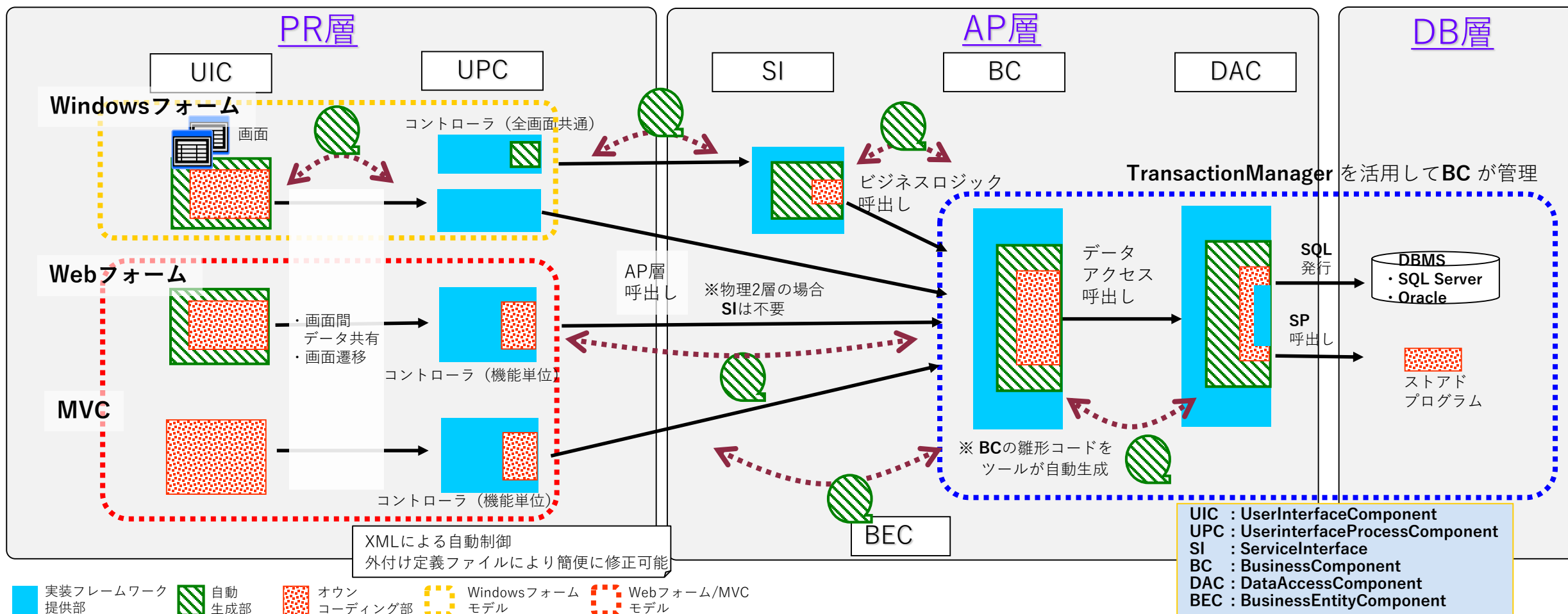
Point 3

開発ツール

- Visual Studio、.NET Framework、HTML5 などを用いた一般的な開発スタイルはそのままに開発を効率化するテンプレートやジェネレータ等の開発環境を提供

3.2.アーキテクチャモデル

Microsoft社ガイドライン「Application Architecture Guide 2.0」に準拠マルチレイヤ構成を採用。均一な品質のアプリケーション開発が可能。

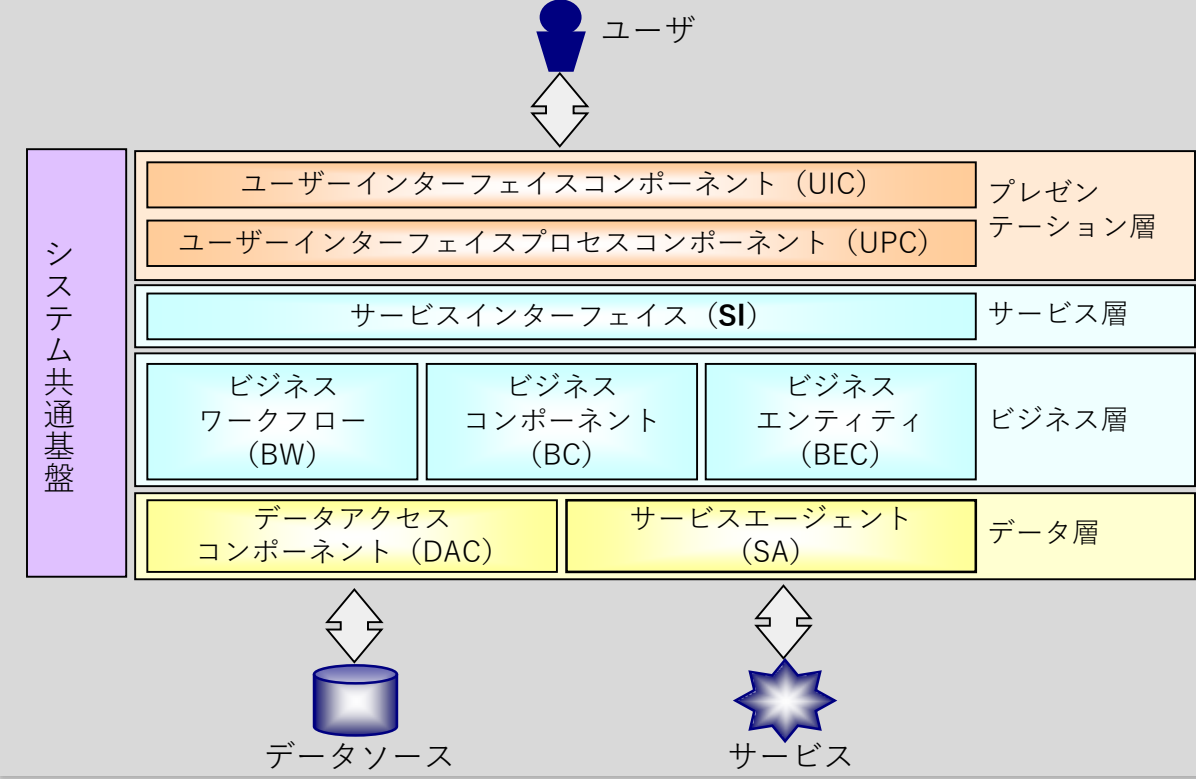


補足.「AAG(Application Architecture Guide)」とは

Microsoft社が提供する.NETアプリケーションのアーキテクチャ策定と設計のためのガイドンス

- アプリケーションの論理階層の分け方、各階層で実現すべき機能を解説
- 複数のハードウェアシステムに分散させやすいアプリケーションを構築可能

AAGが規定しているリファレンスアーキテクチャの一部

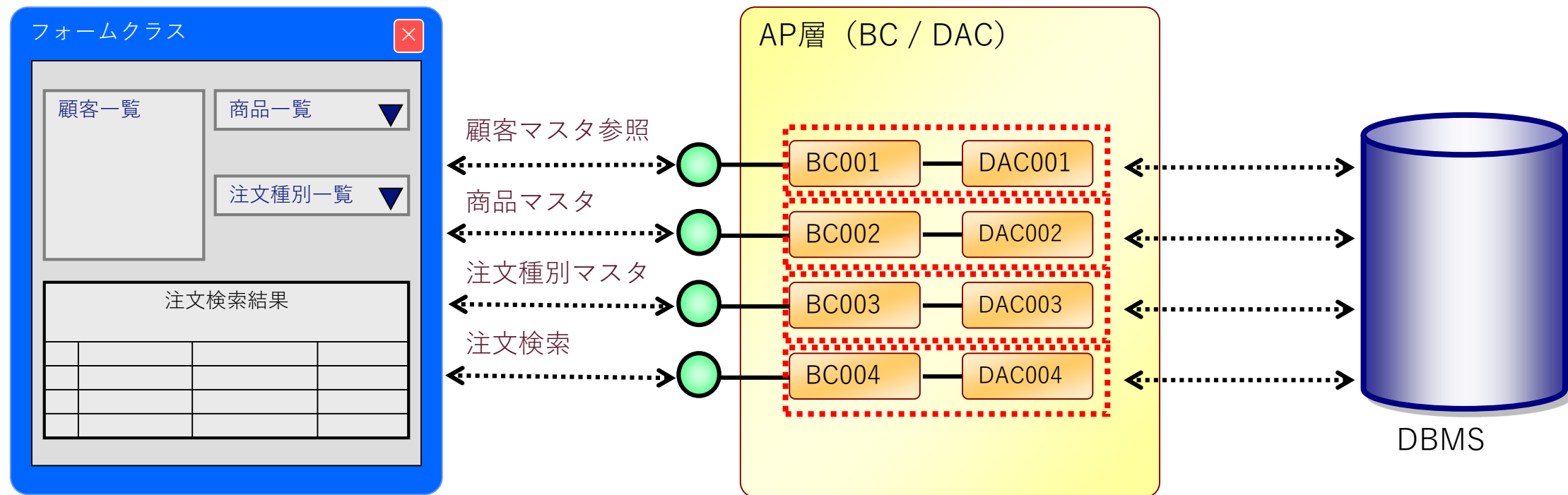


	コンポーネント	主な役割と位置付け
PR層	UIC	ユーザとの対話、データの表示とそれに関連するロジック、入力受付、データ入力チェック、フォーマット機能、Undo、クリップボードサポート、表示キャッシュ、ページング制御など
	UPC	画面遷移制御、入力仕掛けデータの保持、一時保存機能など
AP層	SI	BCに対する通信ラッパ機能、データ形式変換、通信制御、認証など
	BW	ビジネス・プロセスの定義
	BC	ビジネスロジックやトランザクション制御
	BEC	レイヤ間で受け渡すビジネスデータ
DB層	DAC	データソースに対するアクセス制御
	SA	他システムとの連携
システム共通基盤		セキュリティ、運用管理、通信などのシステム共通の機能

3.2.1. クラス構造の統一化

BCクラスの構造をあらかじめ規定し、成果物の品質を均一化

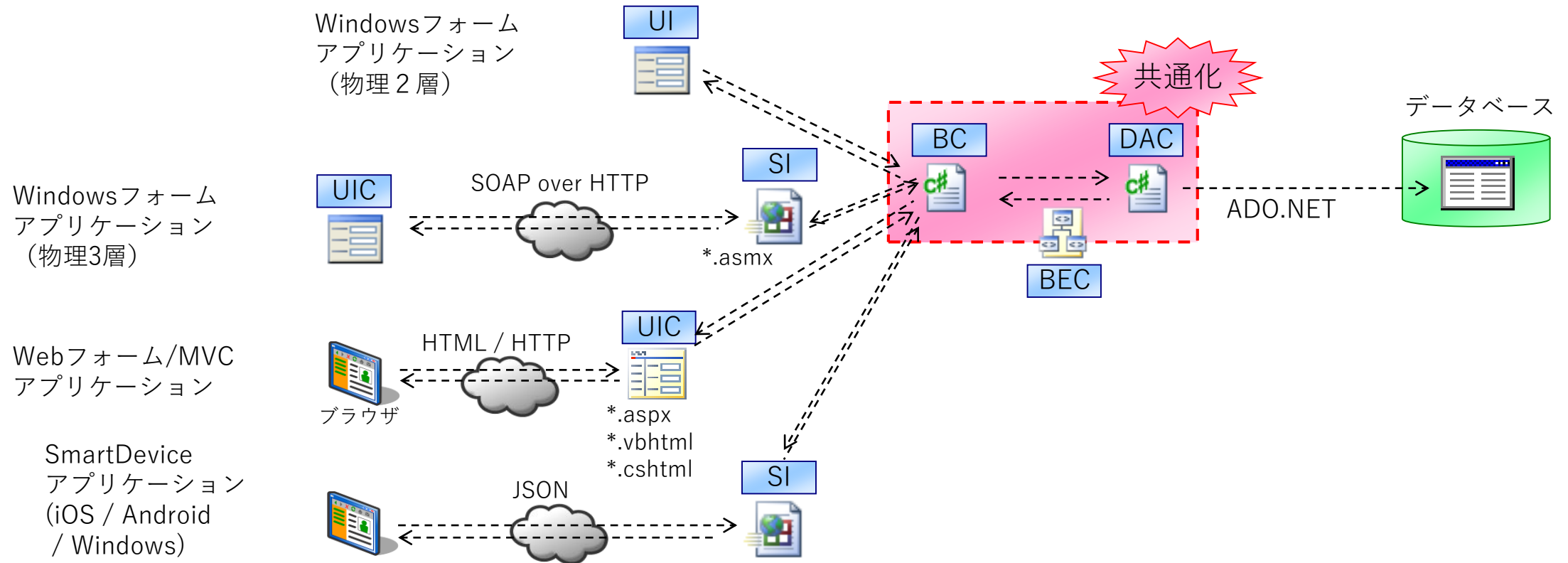
- BCクラスが持つメソッドは、Executeの1つのみ
- BCのメソッドは、1回の呼び出しで処理が完結するサービスとして設計
- トランザクション制御コードの記述は不要（BusinessComponentBase利用時）
- クラス設計とトランザクション設計が直結し、レビュー・保守効率が向上



3.2.2.さまざまなクライアント形態に対応

AP層はクライアント形態にかかわらず共通のアーキテクチャ

- AP 層のコンポーネントは、独立性が高く、様々な形態のクライアントに対応
- 設定情報の変更により、クライアントアーキテクチャをシームレスに移行可能



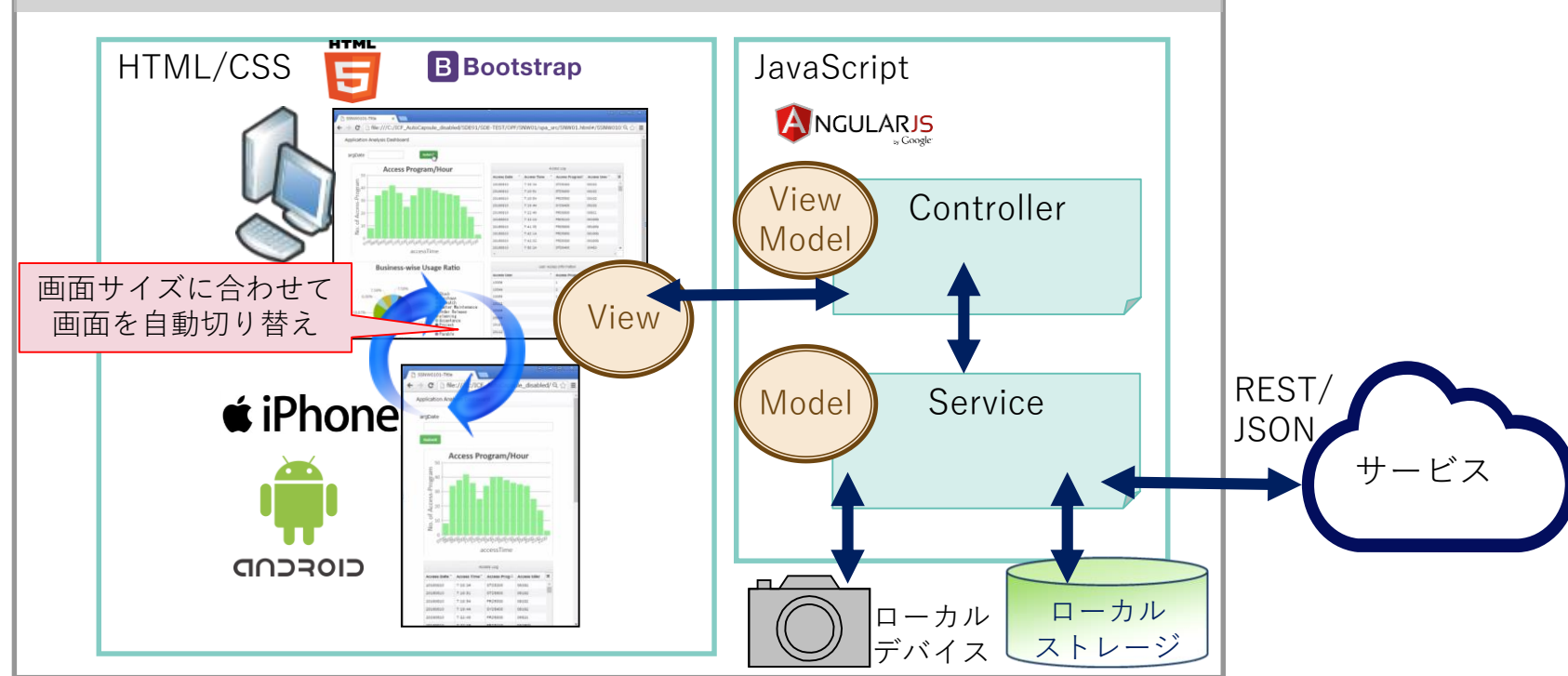
3.2.3. HTML5拡張モデル(SPA)の新設

サーバと疎結合で保守性の良いSPAアプリケーション構築を支援



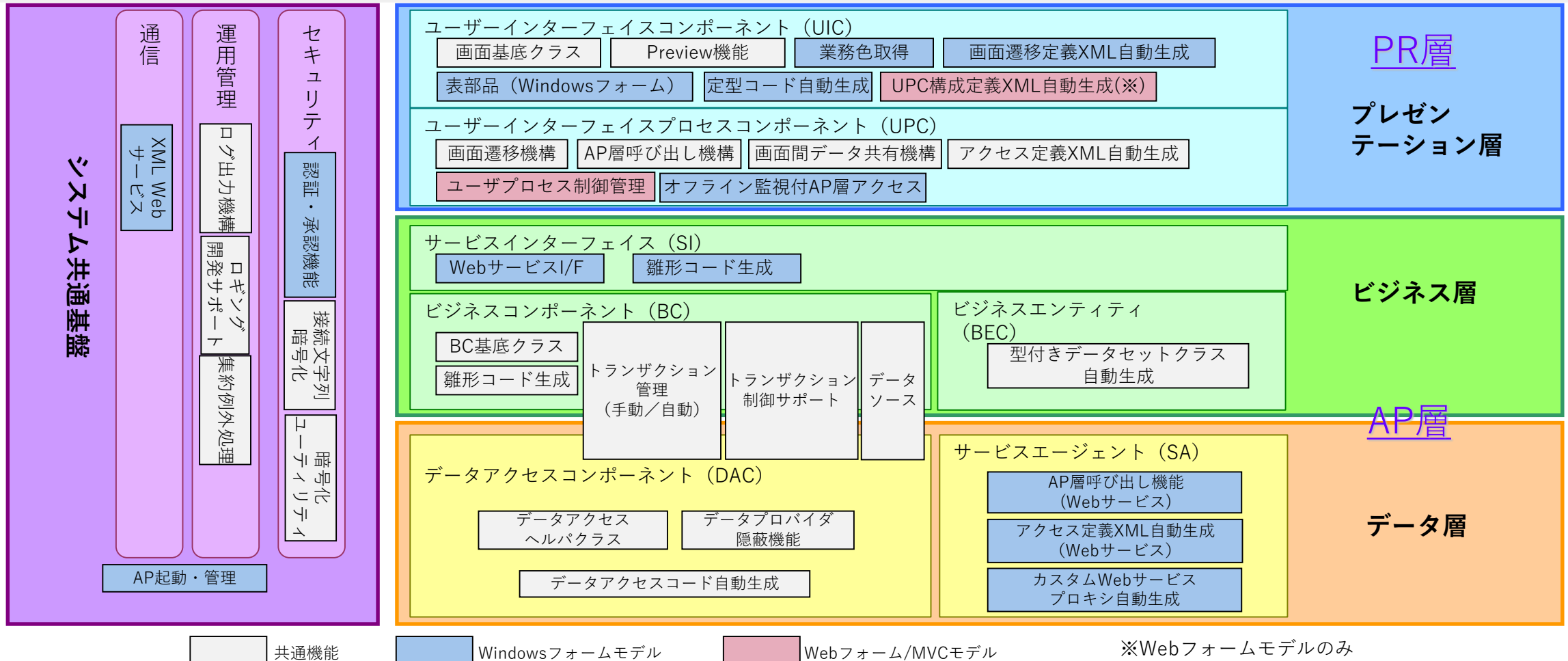
- HTML5、Bootstrapによる表現豊かなユーザインタフェースを提供
- レスポンシブウェブデザインにより、マルチデバイス要件にも対応可能
- Angular、MVVMアーキテクチャによる保守性の高いソースコード
- オフライン要件にも対応可能

クライアント



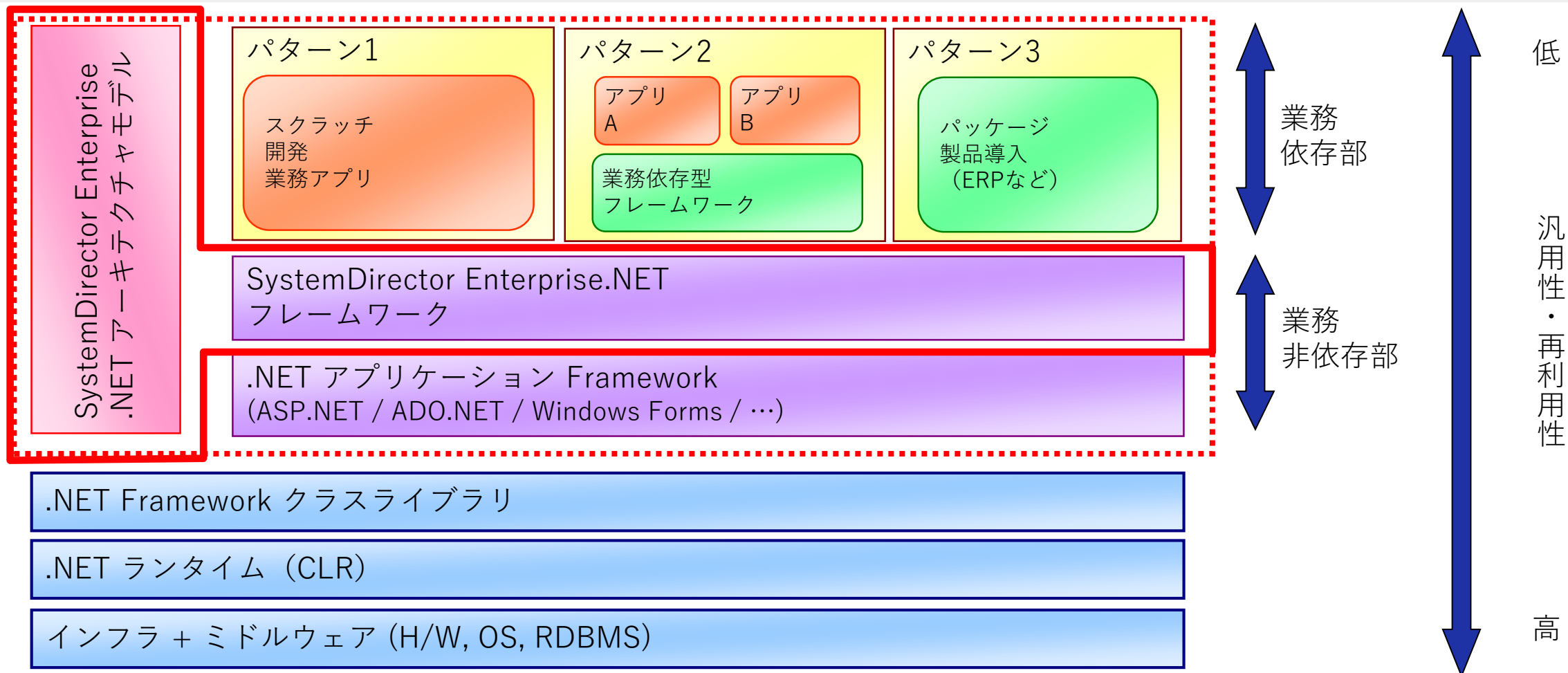
3.3.フレームワーク

アーキテクチャモデルに合わせ、実装効率を高めるための機能を提供



3.3.1.NET Frameworkとの関係

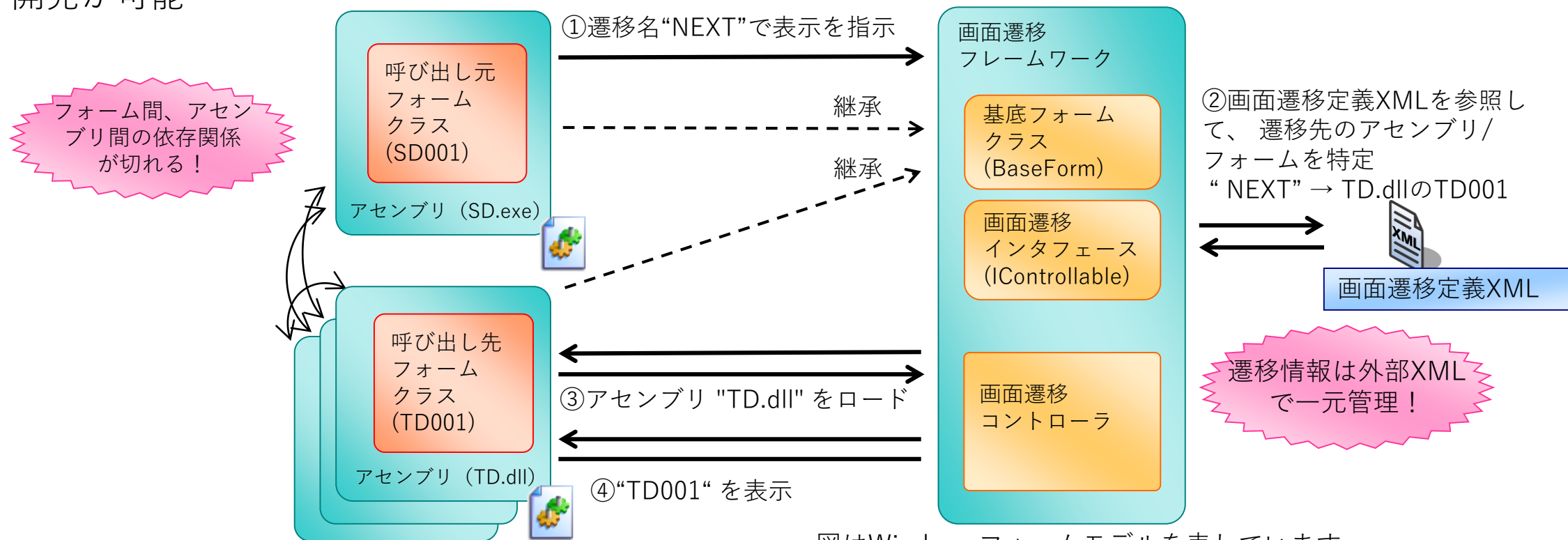
.NET Frameworkをベースとし、NEC独自の強化を行った業務非依存の「アーキテクチャモデル」と「フレームワーク」を提供



3.3.2.画面遷移機構

各モデルの特性に合わせた画面遷移オペレーションの管理機能を提供

- 遷移情報は、外部のXMLファイルで一元管理し、高い保守性を実現
- 画面間の依存関係が切れるため、呼び出し先フォームを含むDLLを参照設定する必要がなく、分離並行開発が可能

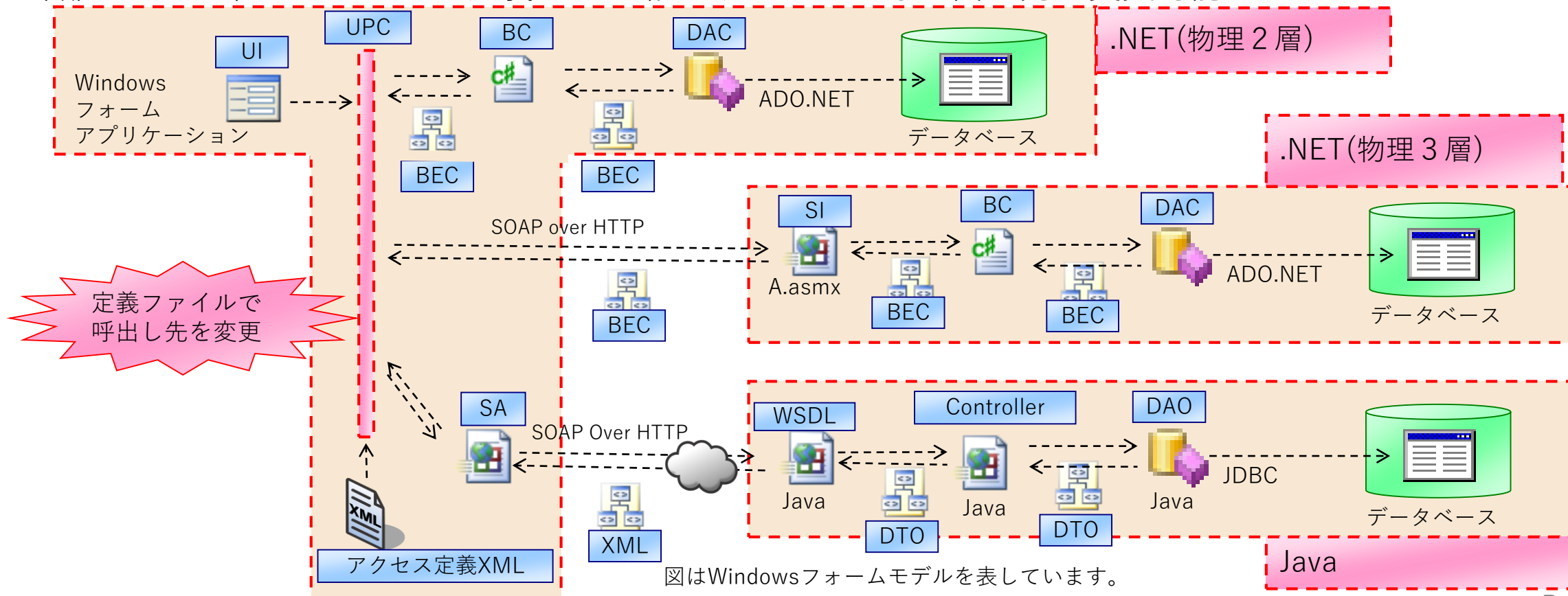


図はWindowsフォームモデルを表しています。

3.3.3. A P 層呼出し機構

PR層／AP層間を疎結合にするとともに通信手段を隠蔽

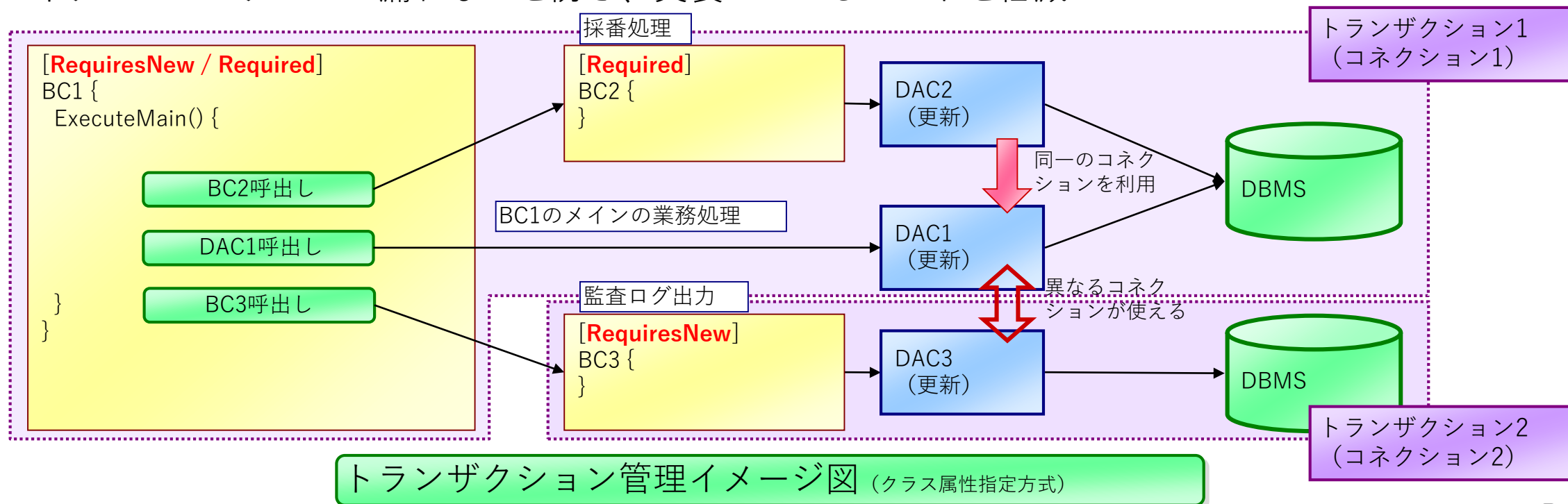
- PR層はAP層の配置先を意識することなく単一のAPIで呼び出すことが可能
- 外部ファイル化したアクセス定義XMLを修正するだけで呼び出し先を変更可能



3.3.4. トランザクション管理機能

トランザクション管理に係わる共通処理をフレームワークで提供

- トランザクションを管理するための共通機能を提供します
- データベース接続・コミット・ロールバック・データベース切断
- データソース管理機構やデータアクセスヘルパークラスとシームレスに連携
- コネクションのクローズ漏れなどを防ぎ、実装にかかるコストを低減



3.4.開発ツール

Visual Studioをベースに、開発プロセスの各工程を効率化するテンプレートやジェネレータを整備

外部設計

ソフトウェア機能設計

画面遷移
エディタ

設計書テンプレート
(画面一覧
画面遷移設計 など)

画面エディタ

設計書生成ツール

内部設計

オンライン実装設計

インタラクショ
ンエディタ

設計書テンプレート
(クラス設計
インタクシ
ョン設計
エンティティ設計 など)

DBアクセス
コマンド設計

製造・単体T

製造・モジュールテスト

ソースコード生成機能
(ジェネレータ)

プログラミング
環境

CDIツール

単体テスト

単体テスト
ツール

カバレッジ

凡例：

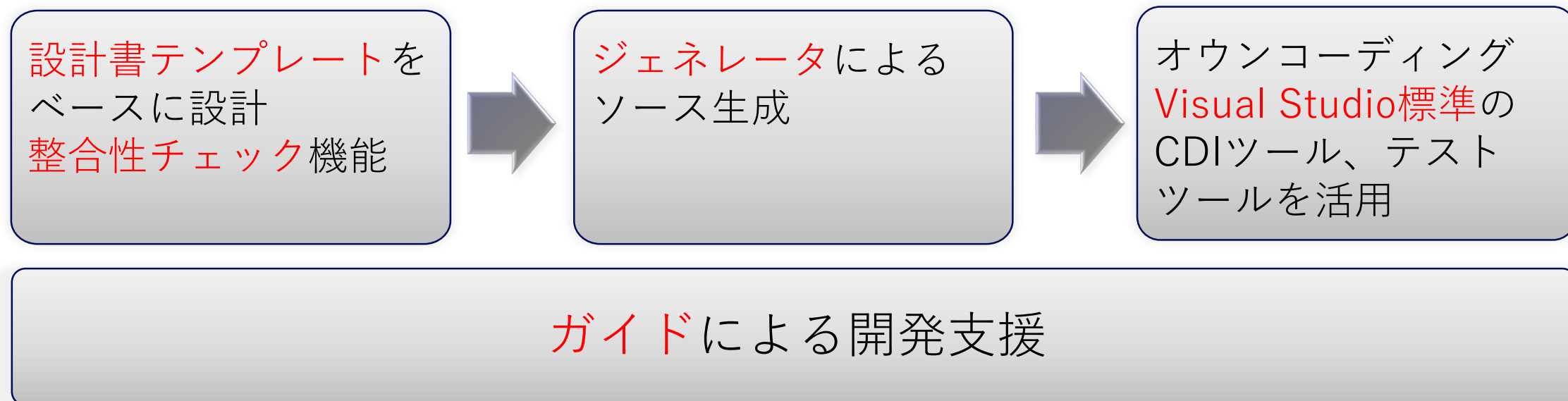
SystemDirector Enterpriseで提供

Visual Studioを利用

3.4.1.開発ツールの特徴（1 / 2）

Visual Studio + .NET Frameworkによる一般的な開発スタイルをそのままに、開発効率化するための設計書テンプレートとジェネレータ群を提供

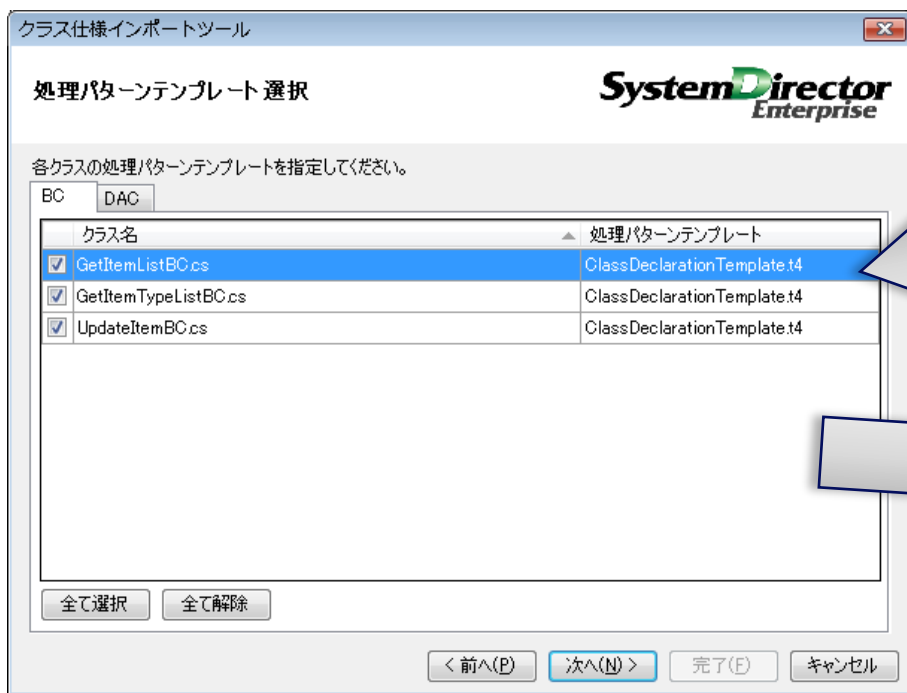
- Excelベースの設計書テンプレートを提供。
- 設計情報をインプットに、推奨アーキテクチャとなるソースを生成
- ソース生成後はVisual Studio標準の開発方式でコーディング。
- Visual Studio標準のCDIツールやテストツールも制約なく利用可能



3.4.1.開発支援ツールの特徴（2 / 2）

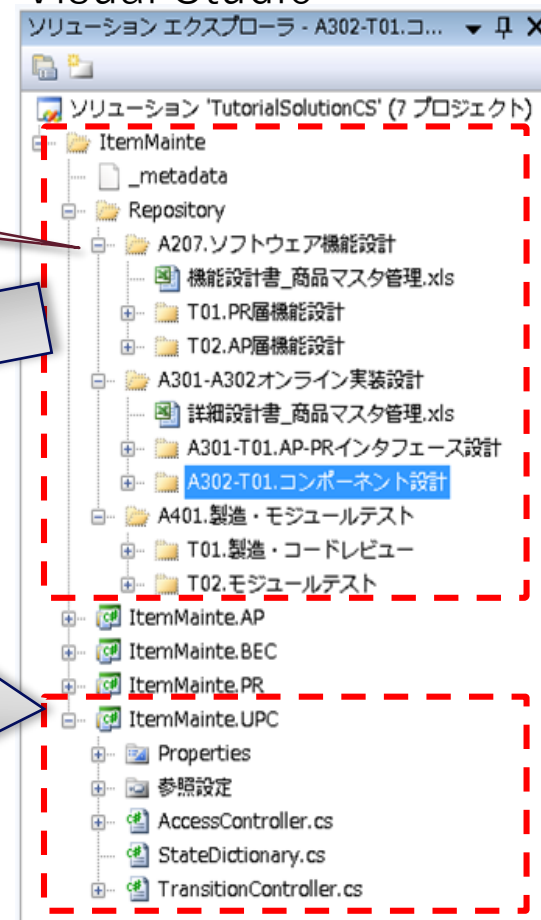
Visual Studio上で、各工程の成果物の管理を実現

開発支援ツール



設計情報や各開発ツールの入出力成果物は、Visual Studioのソリューションエクスプローラ上で管理

Visual Studio



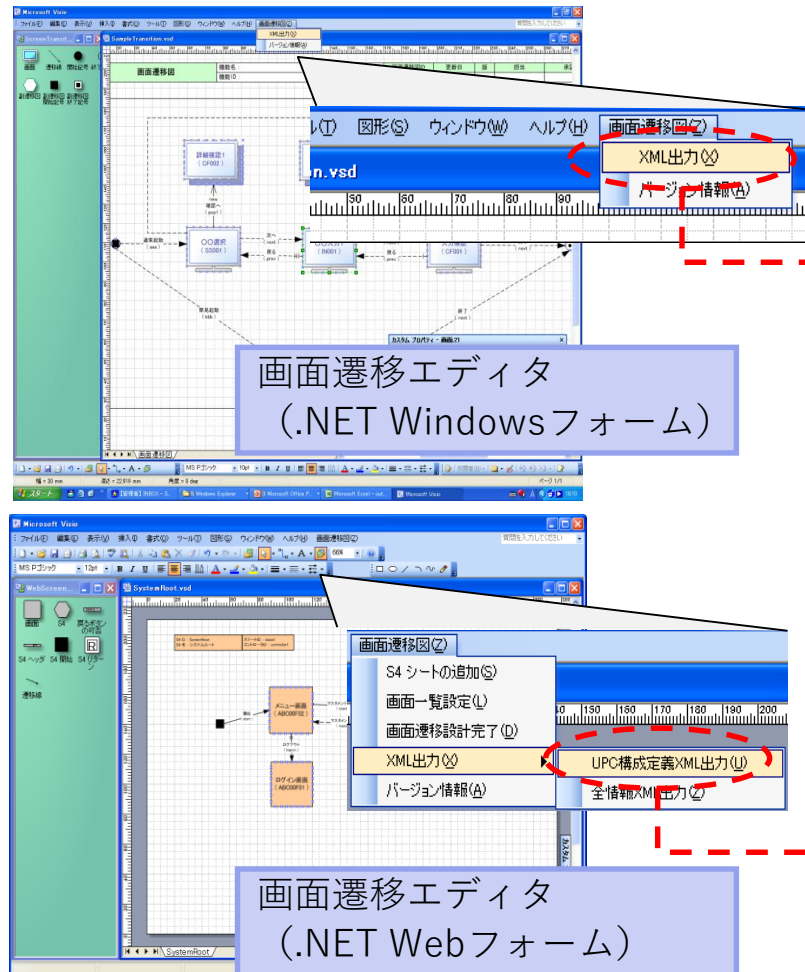
3.4.2.開発ツール一覧

開発工程は「SystemDirector Enterprise開発方法論」に準拠。おもに、機能設計～製造・単体テストまでを開発ツールでサポート

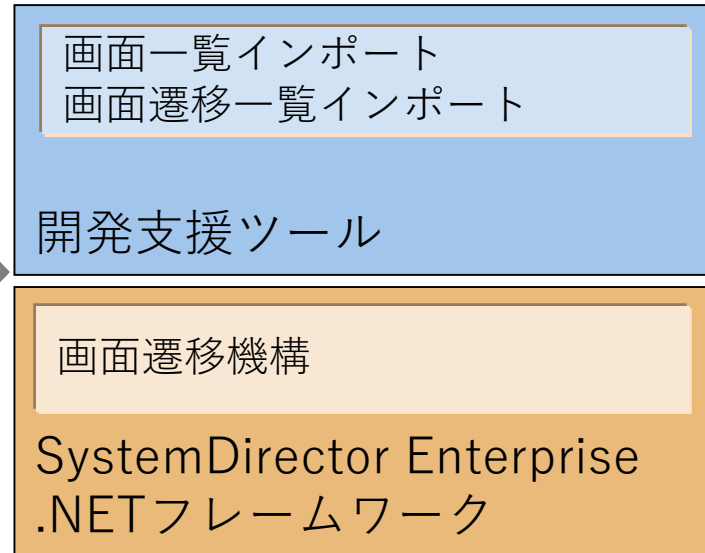
ツール／テンプレート		概要
設計書テンプレート	画面遷移設計、クラス設計、インタラクション設計 他	ジェネレータのインプットとなる設計書テンプレート
	画面遷移エディタ	画面遷移を設計
ジェネレータ	インタラクションエディタ	レイヤ間のインタラクションを設計
	画面一覧インポート	フォーム雛型を一括生成
	画面遷移一覧インポート	画面遷移コードを生成
	クラス仕様インポート	BC/DACクラスを一括生成
	インタラクション仕様インポート	AP層呼び出しコードを生成
	データセット仕様インポート	型付きデータセットを一括生成
	ViewModel仕様インポート	ViewModelクラスを一括生成
	データコントラクト仕様インポート	データコントラクトを一括生成
その他	DACビルダ	データベースアクセスコード生成
	デザインリフレクタ	UI情報をExcelに一覧出力
	キーマネージャ	接続文字列を暗号化
	データソースXML生成ツール	キーマネージャと連携し、暗号化したデータソースXMLを生成

3.4.3.画面遷移エディタ

画面遷移の妥当性を設計時点で検証。遷移情報はXMLファイルとして出力



- ◆ 不正な設計情報を整合チェックにより検出することで、設計時の品質向上が図れます。
- ◆ 設計情報をxmlファイルに出力し、後工程に引き継ぐことが可能です。
- ◆ Excel帳票で遷移を一覧情報として設計し、遷移図自体は任意のツールで作成することも可能です。

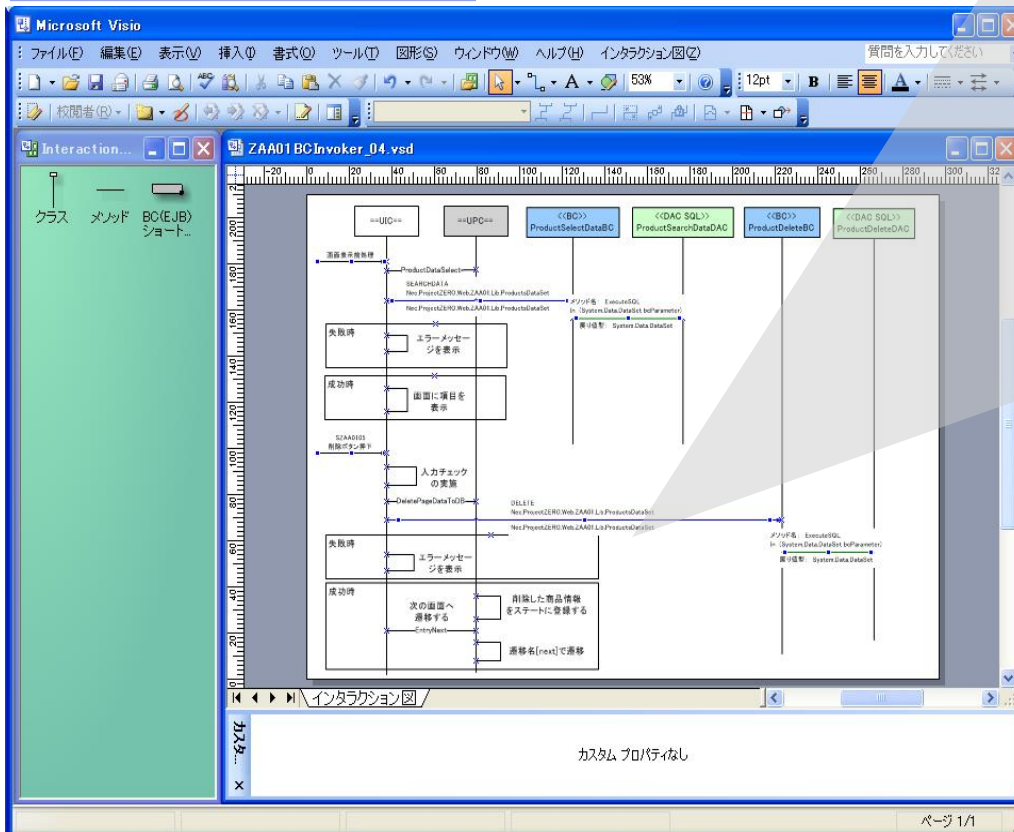


※ MVCモデルは未対応

3.4.4.インタラクションエディタ

AP層のコンポーネント構成をビジュアルなモデルとして設計
ソースコードの雛形やAP層アクセスに必要なXML情報を自動生成

インタラクション図



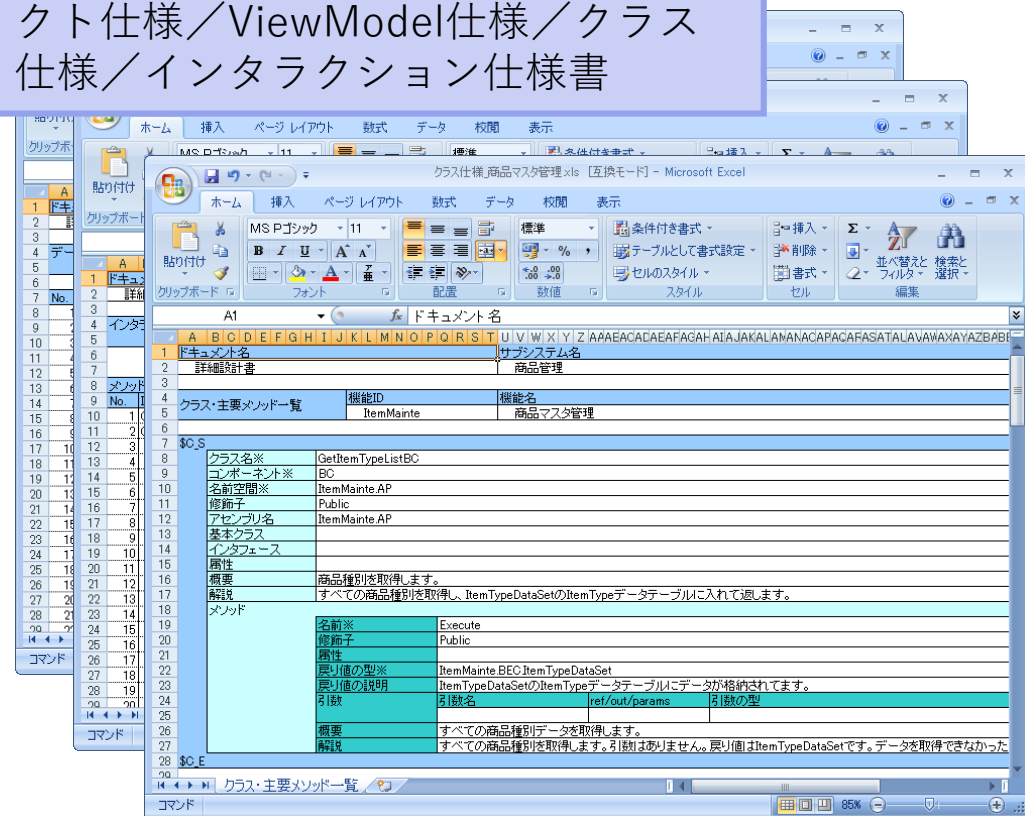
コンポーネント
プロパティ

- ◆ Visioのテンプレートとしてインタラクションエディタを提供。PR層とAP層のインタラクション図が作成できます。
- ◆ 記述したインタラクション図に基づいて「AP層コードの雛形」、「アクセス定義XML」、「仮想アクセス定義XML」を生成します。
- ◆ Excelの仕様書を用いて設計することも可能です。

3.4.5.ジェネレータ

内部設計で作成した仕様書(*.xls)からエンティティクラスやBC/DACクラスのスケルトンを一括生成

データセット仕様／データコントラクト仕様／ViewModel仕様／クラス仕様／インタラクション仕様書



The screenshot shows an Excel spreadsheet titled 'クラス仕様_商品マスタ管理.xls [互換モード] - Microsoft Excel'. The table contains the following data:

クラス名※	GetItemListBC
コンポーネント※	BC
名前空間※	ItemMainte.AP
修飾子	Public
アセンブリ名	ItemMainte.AP
基本クラス	
インタフェース	
属性	商品種別を取得します。
概要	すべての商品種別を取得し、ItemTypeDataSetのItemTypeデータテーブルに入れて返します。
メソッド	
名前※	Execute
修飾子	Public
属性	
戻り値の型※	ItemMainte.BC.ItemTypeDataSet
戻り値の説明	ItemTypeDataSetのItemTypeデータテーブルにデータが格納されています。
引数	引数名 ret/out/params 引数の型
概要	すべての商品種別データを取得します。
解説	すべての商品種別を取得します。引数はありません。戻り値はItemTypeDataSetです。データを取得できなかった

- ◆ データセット仕様／データコントラクト仕様／ViewModel仕様／クラス仕様／インタラクション仕様をExcel帳票ベースで作成します。
- ◆ 設計書フォーマットはPJ毎にカスタマイズが可能です。
- ◆ 作成した設計書の内容を元に、コードを自動生成することで、手作業によるミスの軽減と作業効率を向上します。（仕様書毎の個別生成も可能）



BC/DACクラス

エンティティクラス

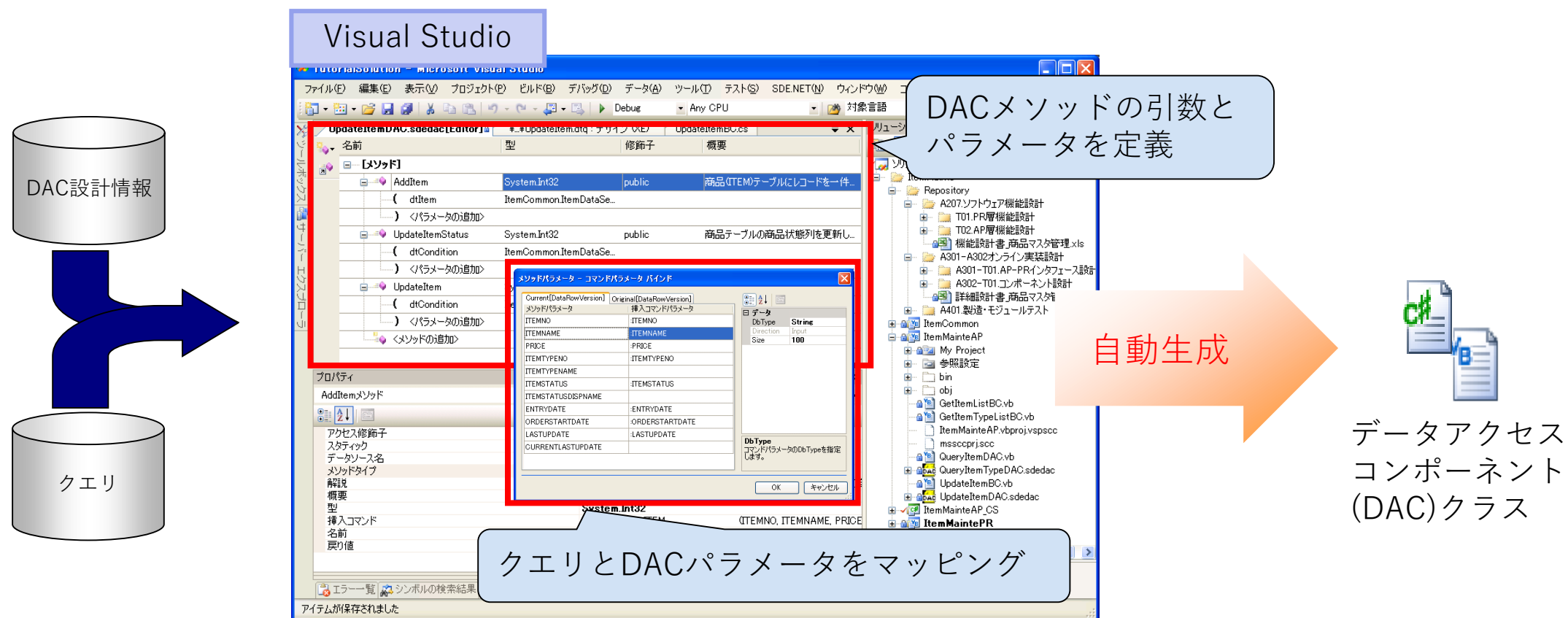
画面遷移コード

AP層呼び出しコード

3.4.6.DACビルダ

Visual Studioデザイナー上で、クエリとDAC設計情報をマッピングして、ソースコードを自動生成

- データアクセスコンポーネントは、生成後必要に応じて自由にカスタマイズ可能。
- 設計情報の変更による再生成も、それまでに施したカスタマイズの内容を保護。

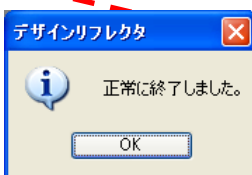
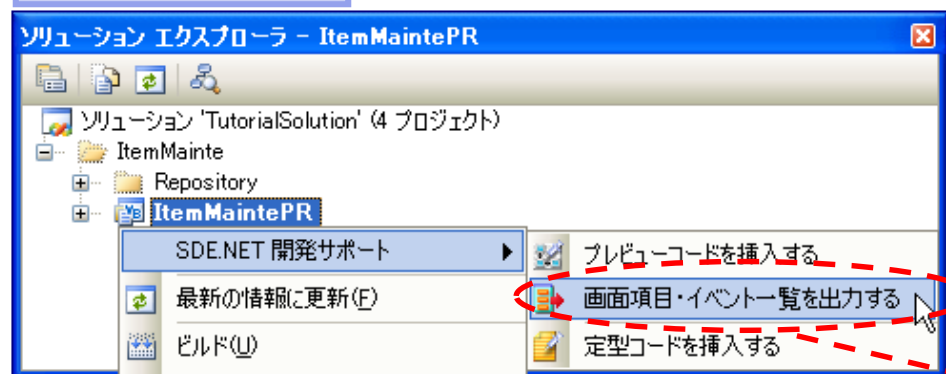


3.4.7.デザインリフレクタ

画面レイアウトから、画面とイベントの設計情報をExcelファイルに出力

- ◆ 画面項目情報とイベント情報を取得し、Excelファイルとして出力します。
- ◆ 出力項目/ExcelフォーマットはPJ毎にカスタマイズが可能です。

Visual Studio



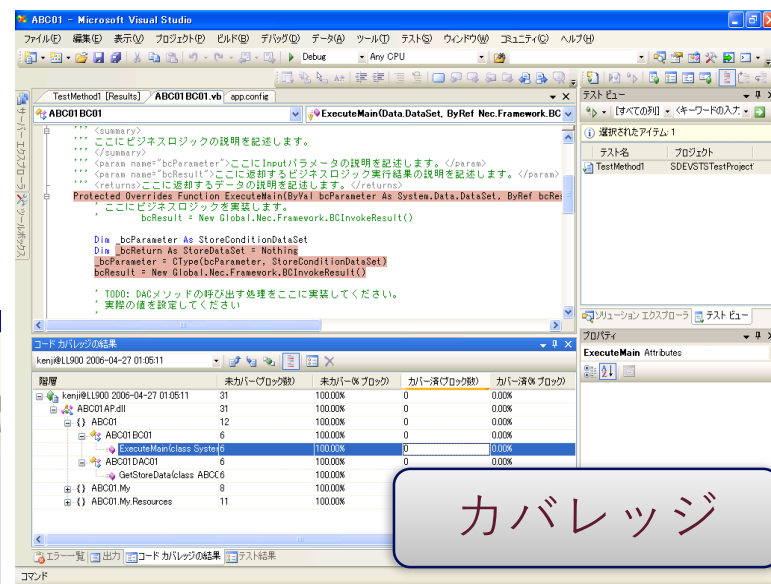
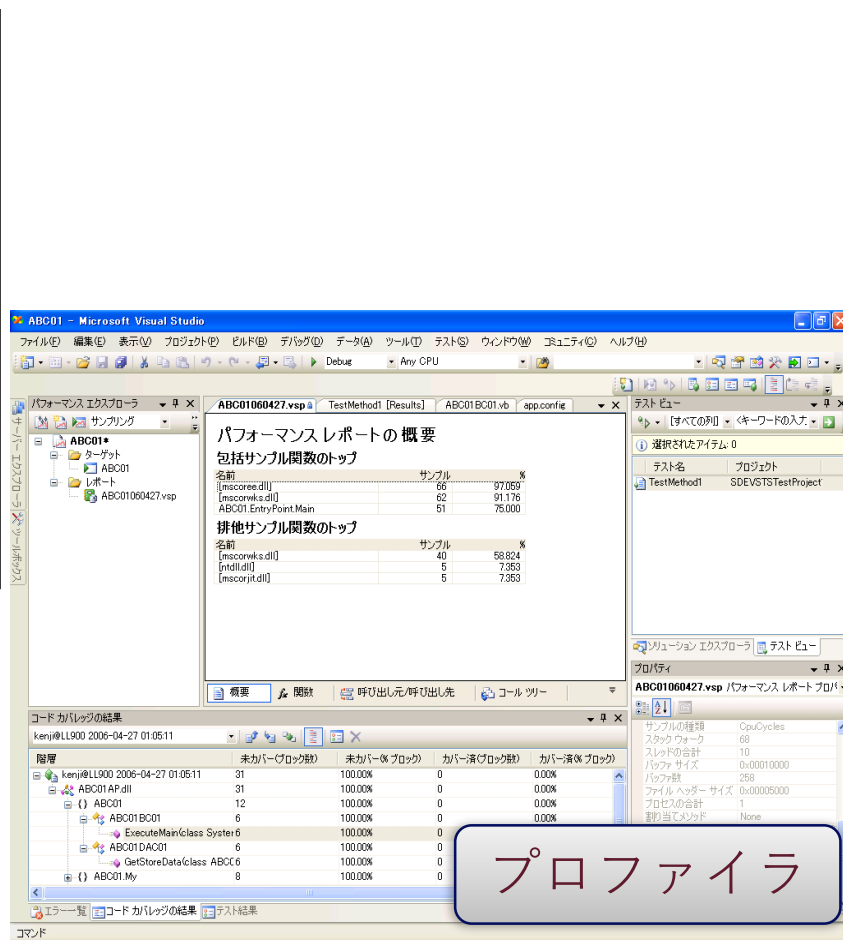
設計情報

Microsoft Excel - 画面項目定義_frmSe.xls

No.	項目名	オブジェクト名	種類	I/O	桁数	属性	必須	初期値	チェック	フォーマット	テーブル名
1	ドキュメント名										サブシステム名
2	0										0
4	画面項目定義	機能ID	機能名								画面ID
5		0	0								frmSearchMain
7											
8	1	txtPRODUCT_CD	TextBox		10	0					
9	2	txtPRODUCT_NN	TextBox		30	1					
10	2	btnSEARCH	Button			2			検索		
11	2	btnUPDATE	Button			3			更新		
12	2	Label1	Label			5			商品コード		
13	2	Label2	Label			8			商品名		
14	2	grdPRODUCT	DataGridView			7					
15	2										

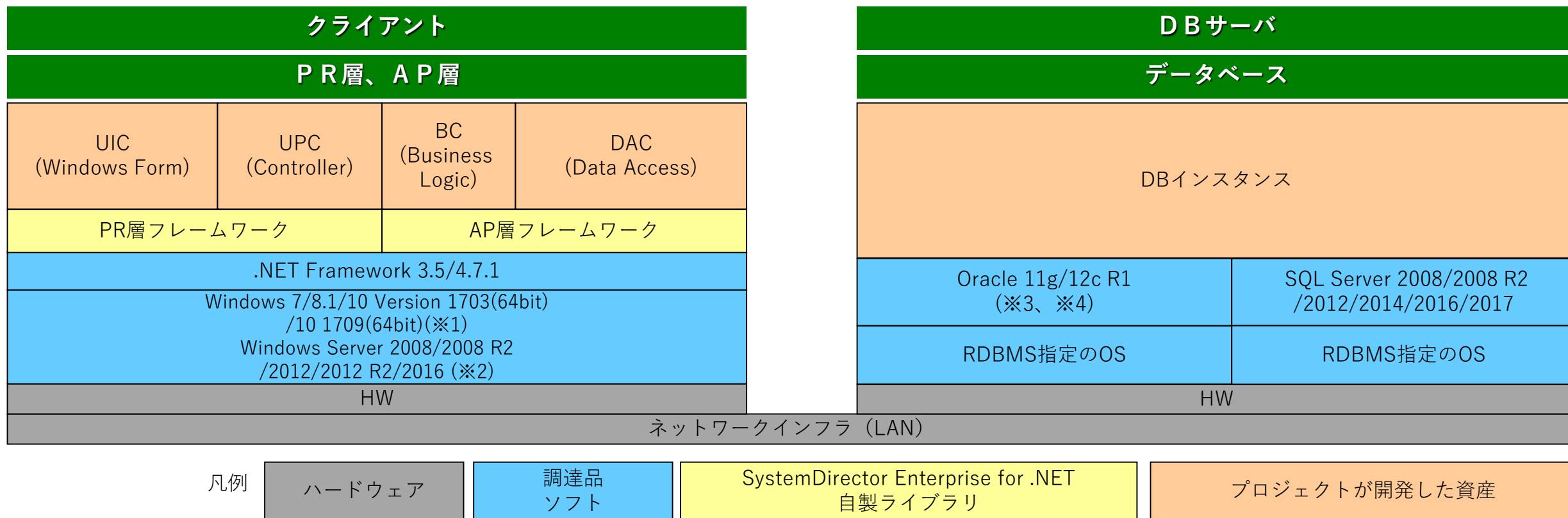
3.4.8.テスト機能連携

Visual Studioのテスト機能と連携し、テスト効率と品質をさらに向上



3.5.動作環境

Windowsフォームモデル（2層）



※1：特に指定がない場合、32bit/64bitに対応

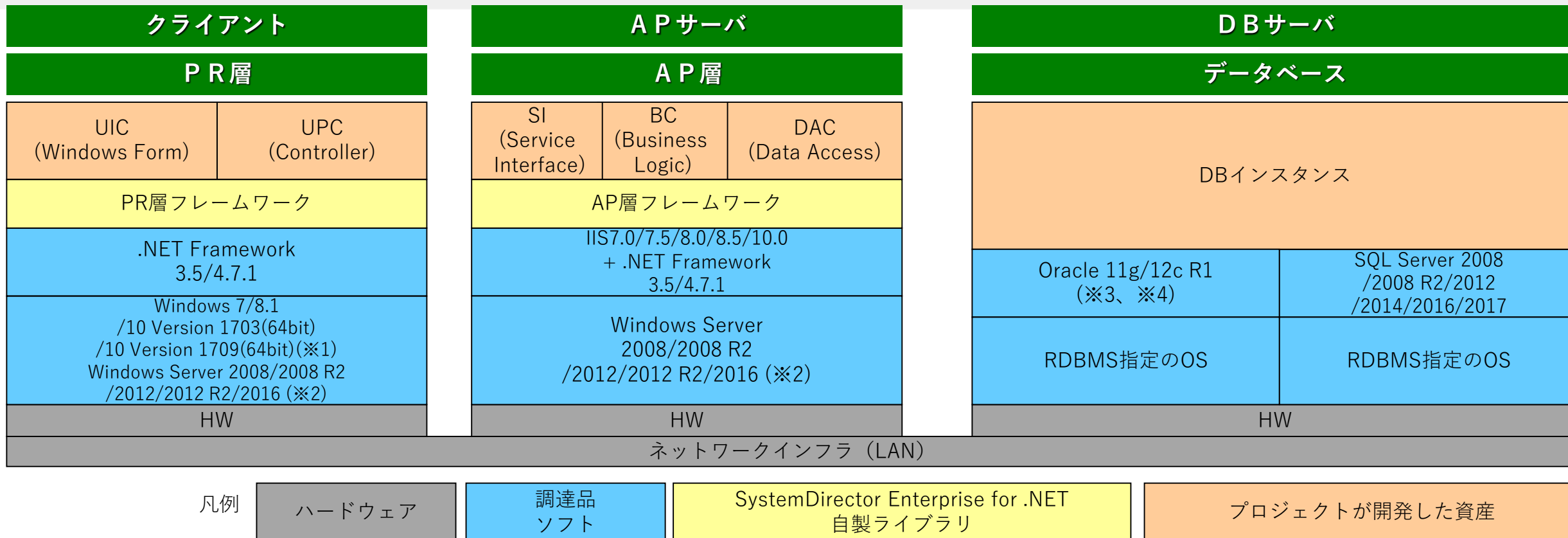
※2：Windows Server 2008は32bit/64bitに対応、2008 R2以降は64bitに対応

※3：2018年3月現在Oracle社製データプロバイダーが.NET Framework 4.7.1に未対応であるため、.NET Framework 4.7.1環境からOracle Databaseへの接続は未対応

※4：.NET Framework 3.5SP1環境は、Oracle 12cR2への接続に未対応

3.5.動作環境

Windowsフォームモデル（3層）



※1：特に指定がない場合、32bit/64bitに対応

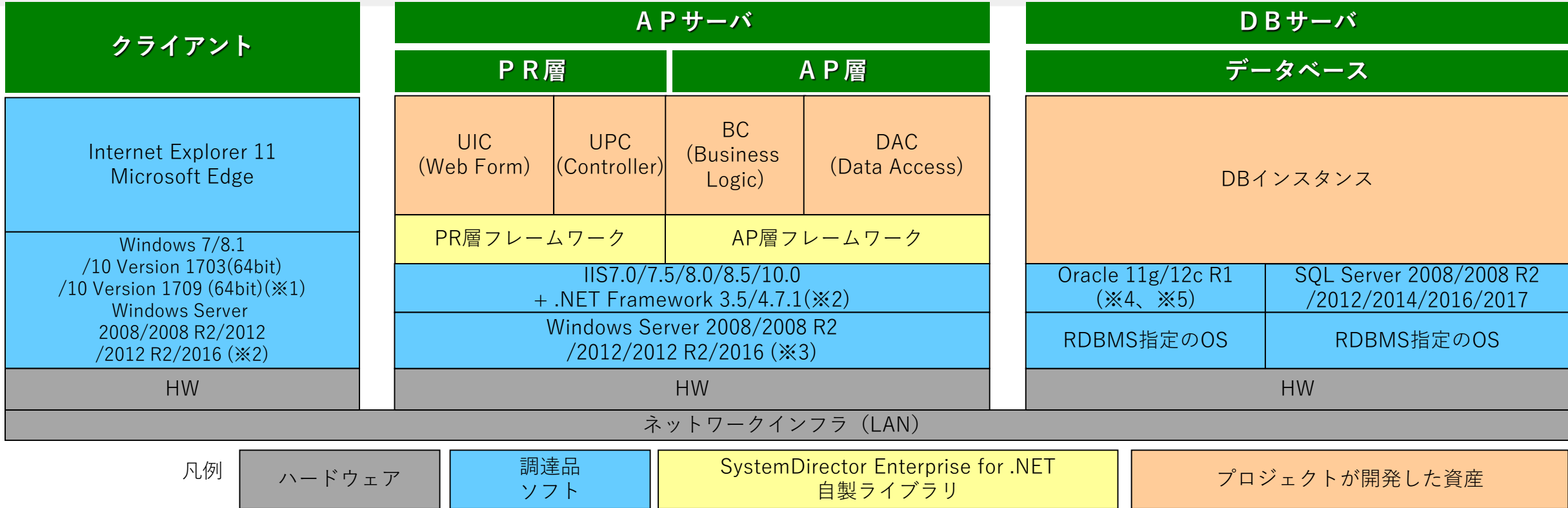
※2：Windows Server 2008は32bit/64bitに対応、2008 R2以降は64bitに対応

※3：2018年3月現在Oracle社製データプロバイダーが.NET Framework 4.7.1に未対応であるため、.NET Framework 4.7.1環境からOracle Databaseへの接続は未対応

※4：.NET Framework 3.5SP1環境は、Oracle 12cR2への接続に未対応

3.5.動作環境

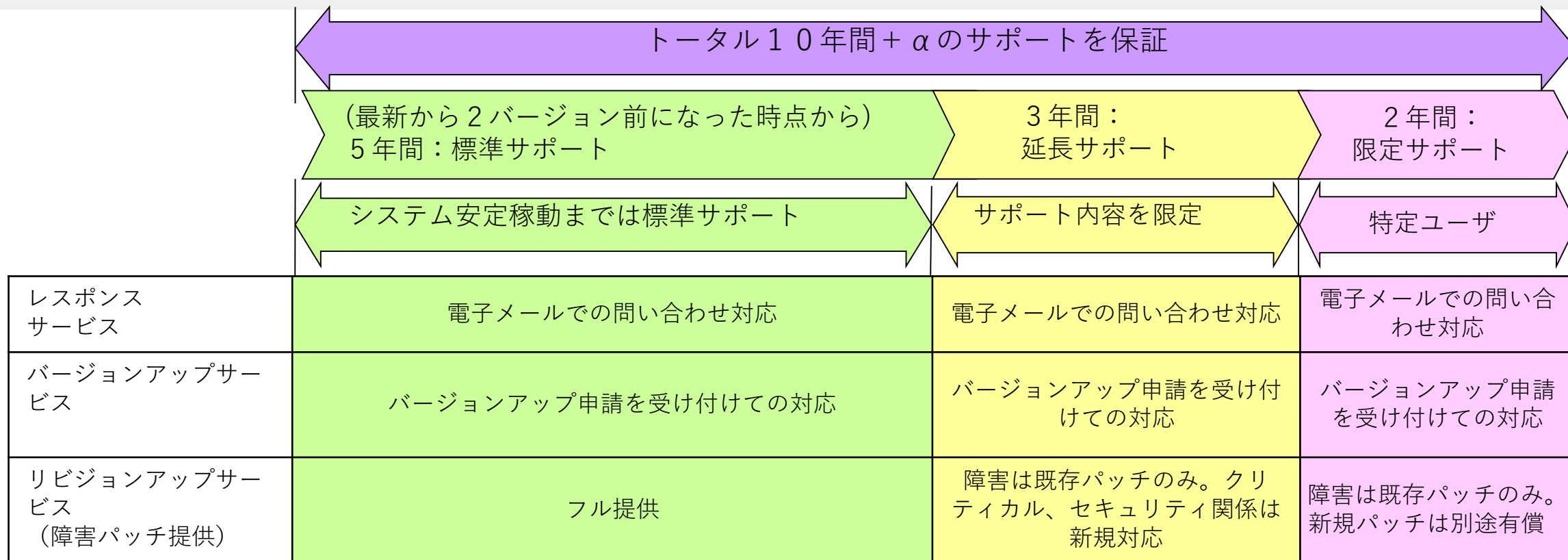
Webフォーム/MVCモデル（3層）



第4章 サポートサービス

4.1.サポートメニュー

サービスメニューとして、以下のサービスを提供します。お客様のご要望に応じて内容を組み立てさせていただくことも可能です。



◆ 基本サービスに含まれないものでも、ご要望があれば調整可能です。

- 例) SystemDirector Enterpriseを新規に導入する場合の環境構築／プロジェクト立ち上げ教育といった導入支援
- 例) お客様専用窓口の設置によるホットラインサポート（メールに加えて、電話・現地での問い合わせ対応なども実施）

4.2.教育メニュー

- ◆ SystemDirector Enterpriseの全体像が把握できる研修メニューを、NECマネジメントパートナーのラーニング事業と連携して提供します。ニーズに合わせたカスタマイズメイド教育として実施します。
- ◆ 現在、定期開催されているコースはございませんが、ご要望いただければ時期・内容・場所など含めて開催調整させていただきます。お気軽にご相談ください。
- ◆ なお、SystemDirector Enterpriseのベースとなっている一般知識は前提となります。前提知識を習得するための教育コースもございますので、合わせてご検討ください。

NECマネジメントパートナー ラーニング事業サイト

<https://www.neclearning.jp/>

	Java	.NET	Smart Device
前提 コース	JavaによるWebアプリケーション開発1 ーサーブレット・JSPー	Visual Studio による Windowsプログラミング	HTML5/CSS3基礎
	Javaによるデータベースアクセス	Webアプリケーション開発入門 (ASP.NET編)	JavaScriptプログラミング
	Javaプログラミング基礎 (入門、基礎1、基礎2)		

4.3.お問い合わせ先



SystemDirector Enterprise サポート窓口では
業務システム構築基盤に関する様々なご質問やご相談に
お応えします。例えば・・・

新しい技術をどのように適用すれば良いか
業務アプリケーションの開発や保守を効率化するにはどうしたら良いか
既存資産をどう活用したら良いか

などお気軽にご相談ください



ご購入前のお問い合わせ

SystemDirector ご相談窓口

 Eメール contact@systemdirector.jp.nec.com

- SystemDirector Enterprise, NeoFace, WebOTX は日本電気株式会社の登録商標です。
- Windows, Windows Server, Excel, SQL Server, Visual Studio, .NET Framework, Internet Explorer, Microsoft Edge は Microsoft Corporationの米国およびその他の国における登録商標です。 また、Windows の正式名称は Microsoft Windows Operating System です。
- Oracle, Java, JavaScript, JDK, WebLogicは、Oracle Corporationおよびその子会社、関連会社の米国およびその他の国における商標または登録商標です。
- Eclipseは Eclipse Foundation, Inc. の米国およびその他の国における登録商標です。
- Android は Google LLC の商標です。
- iPad, Mac OSは、米国および他の国々で登録されたApple Inc.の商標です。
- iOSは、米国Ciscoのライセンスに基づき使用されています。
- Tomcat は、Apache Software Foundation の登録商標または商標です。
- Docker は、Docker Inc. の米国およびその他の国における登録商標もしくは商標です。
- PostgreSQL はPostgreSQL の米国およびその他の国における登録商標もしくは商標です。
- Linux はLinus Torvalds 氏の米国およびその他の国における登録商標または商標です。
- Red Hat Enterprise Linuxは、米国およびその他の国における Red Hat, Inc. の登録商標です。
- Amazon Web Services, “Powered by Amazon Web Services”ロゴ、その他のAWS商標はAmazon.com, Inc.の米国およびその他の国における登録商標です。
- その他、記載されている会社名および製品名は、各社の商標または登録商標です。

Orchestrating a brighter world

NECは、安全・安心・公平・効率という社会価値を創造し、
誰もが人間性を十分に発揮できる持続可能な社会の実現を目指します。

\Orchestrating a brighter world

NEC