

業務システム構築基盤のご紹介 SystemDirector Enterprise for Java

日本電気株式会社

Orchestrating a brighter world

未来に向かい、人が生きる、豊かに生きるために欠かせないもの。
それは「安全」「安心」「効率」「公平」という価値が実現された社会です。

NECは、ネットワーク技術とコンピューティング技術をあわせ持つ
類のないインテグレーターとしてリーダーシップを発揮し、
卓越した技術とさまざまな知見やアイデアを融合することで、
世界の国々や地域の人々と協奏しながら、
明るく希望に満ちた暮らしと社会を実現し、未来につなげていきます。

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
- 補足.エディションの違い

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

付録 リアルタイム開発

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

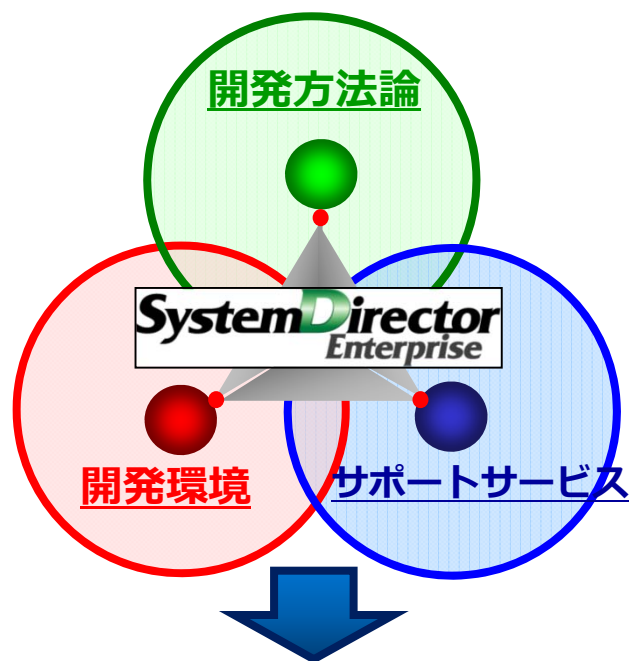
生産性の向上

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構築基盤です。開発方法論、開発環境、サポートサービスによる効率的なシステム構築を支援します。



◇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

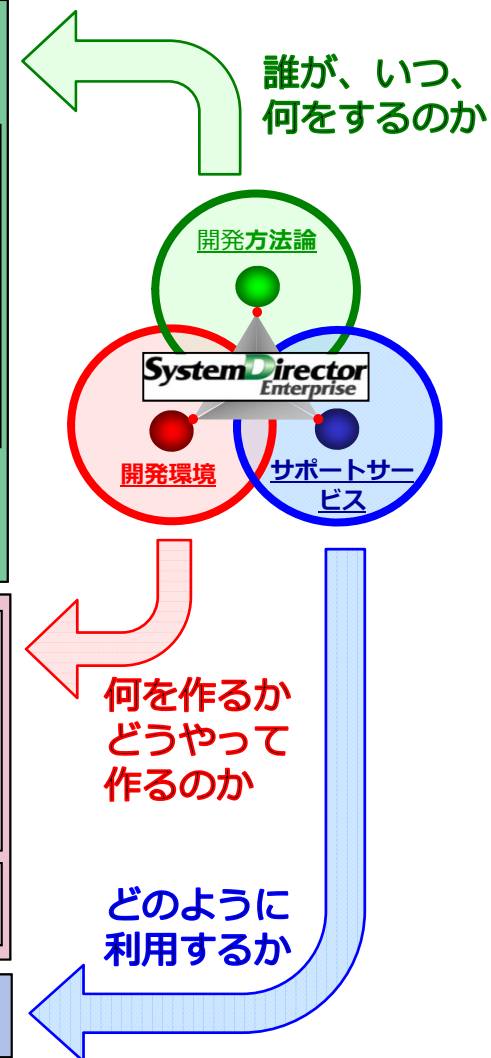
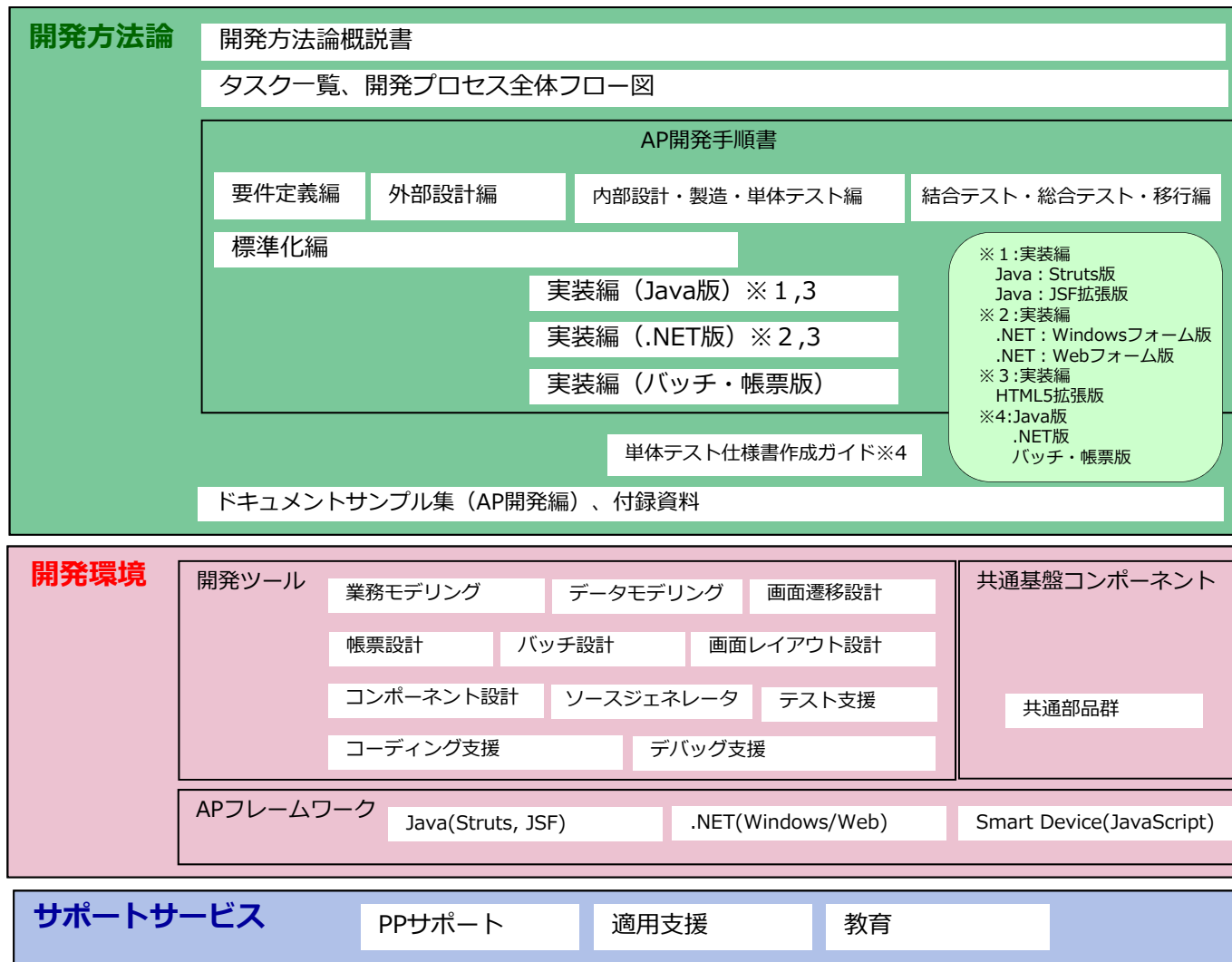
- ・ 開発方法論で標準作業の効率化
- ・ 開発環境で生産性向上

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。



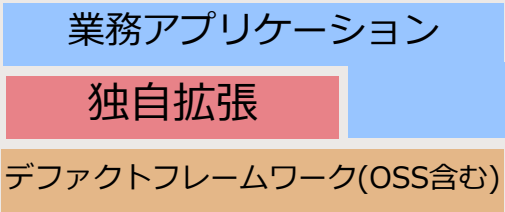
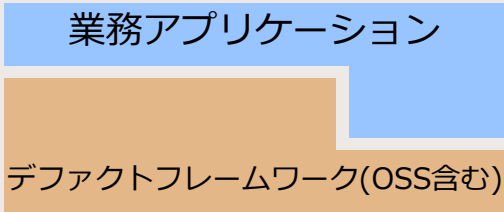
1.4.SystemDirector Enterprise の製品ラインナップ

「オンラインシステム開発基盤」「バッチシステム開発基盤」の製品カテゴリで、お客様のシステム開発を支援



補足.エディションの違い

SystemDirector Enterpriseでは、生産性重視のProfessional Editionと適用柔軟性の高いExpress Editionの2つを提供しており、要望に合わせて選択可能です。

	Professional Edition	Express Edition
製品 コンセプト	生産性重視の開発環境 生産性を高める独自のフレームワーク提供	適用しやすい開発環境 導入コストを下げるためOSS等標準技術のみを利用
独自フレーム ワーク	独自拡張機能を有したフレーム ワークを提供 	独自実装を排除し、ピュアなソース コードで開発 
メリット	<ul style="list-style-type: none">独自ツール、フレームワークにより作業効率化を実現フレームワークによる標準化で工数を削減	<ul style="list-style-type: none">Eclipse、Visual Studioのみの開発より高い生産性部分的な切り離しが容易
デメリット	<ul style="list-style-type: none">独自フレームワークによる一気通貫な機能提供となるため、部分的に切り離しにくい	<ul style="list-style-type: none">開発の自由度が大きいため、標準化工数がかかる実行環境に独自性がないため、サポート提供がない

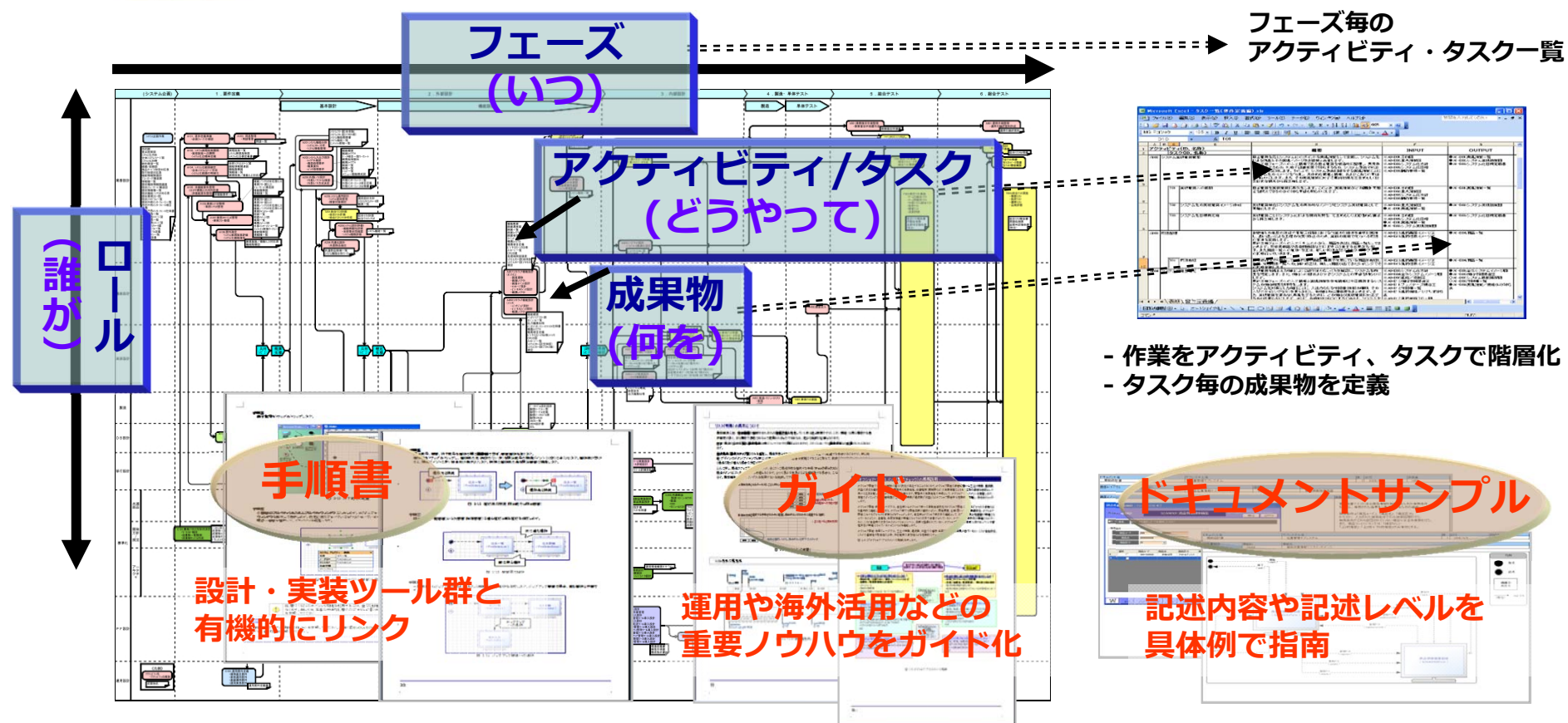
第2章 開発方法論

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



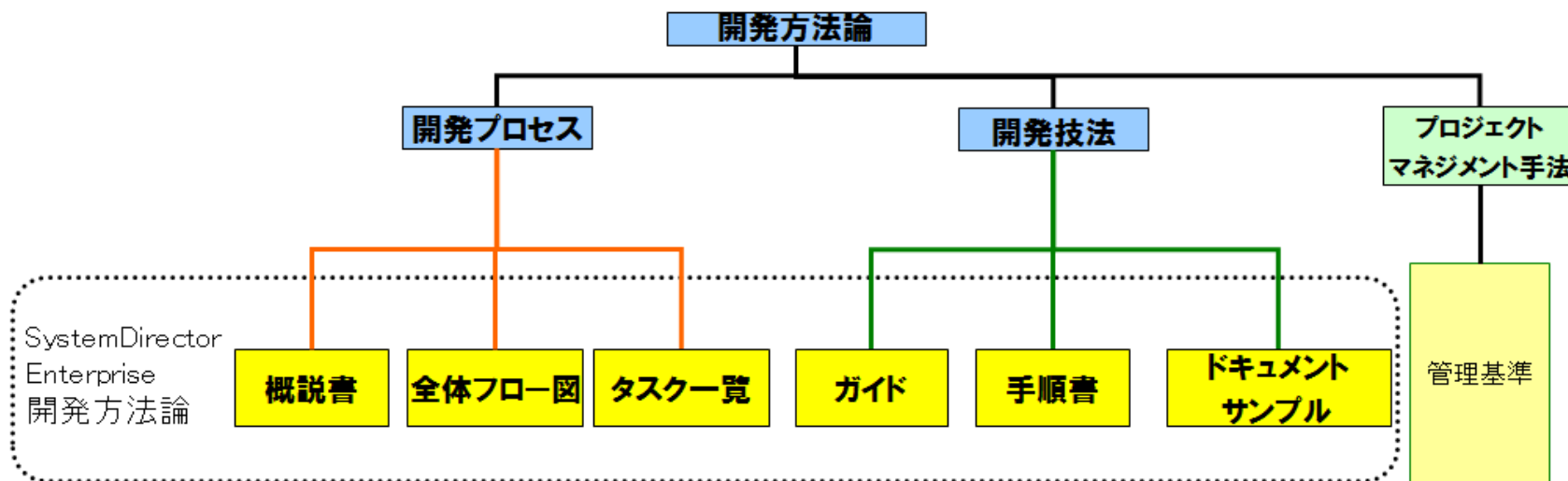
2.2.一般的な開発方法論との違い

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。



2.4.フロントローディング

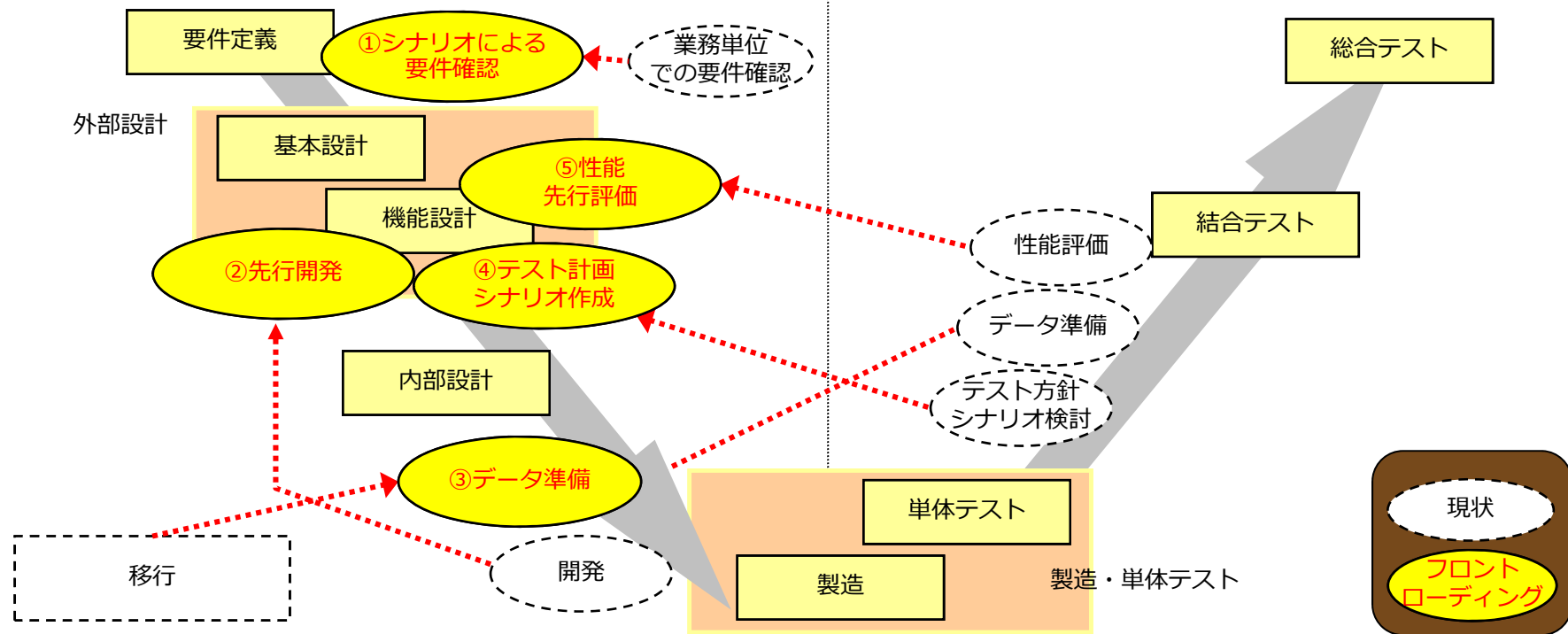
開発の前半で品質を作りこむ手法を取り入れています。

「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

前半の工程で品質を作りこみ

後半の工程での障害を低減



第3章 開発環境

3.1.提供機能のポイント

Webアプリケーションの開発を支援する各種機能を提供します。

Point
1

アーキテクチャモデル

- デファクトスタンダードなフレームワークをベースとしたSystemDirector Enterprise推奨のアーキテクチャモデルを提供することで、スムーズな最新技術への対応やアーキテクチャ選定の検討工数を削減

Point
2

フレームワーク

- Web 3 層システムにおける業務システム構築に必要なフレームワークを提供することで、業務SEは業務ロジックに注力可能。

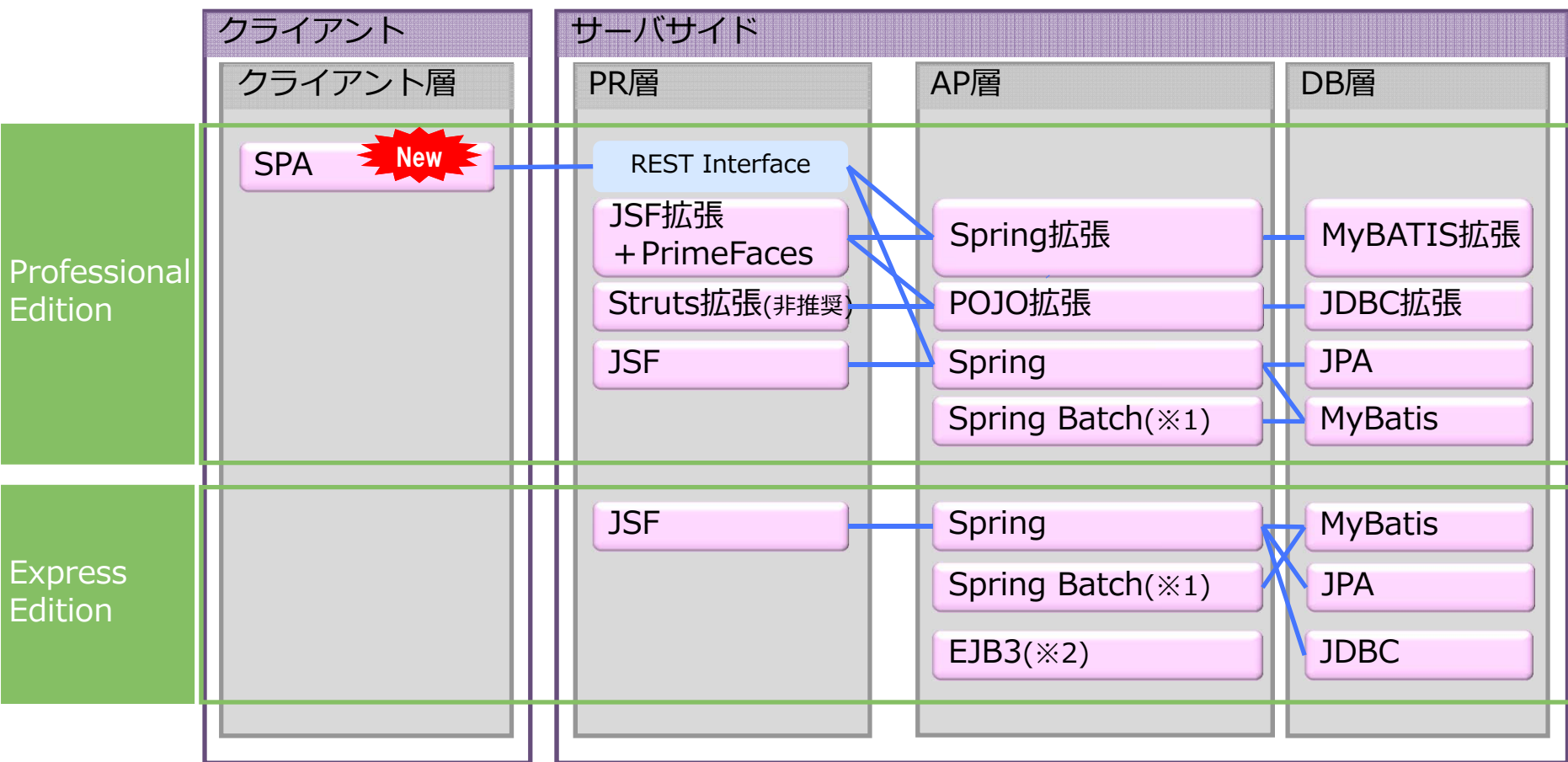
Point
3

開発ツール

- 実績のある SystemDirector Enterprise開発基盤のノウハウを生かし、開発方法論をベースに Web アプリケーションの作成を支援するジェネレータ（設計情報からのソースコード生成）等の開発環境を提供

3.2.アーキテクチャモデル

デファクトスタンダードなJavaフレームワークをベースに独自拡張や推奨の組み合わせを定義したアーキテクチャモデルを提供します。



※各線の接続はツールかガイドで提供
※1 小規模簡易バッチAP開発向け
※2 Deprecated

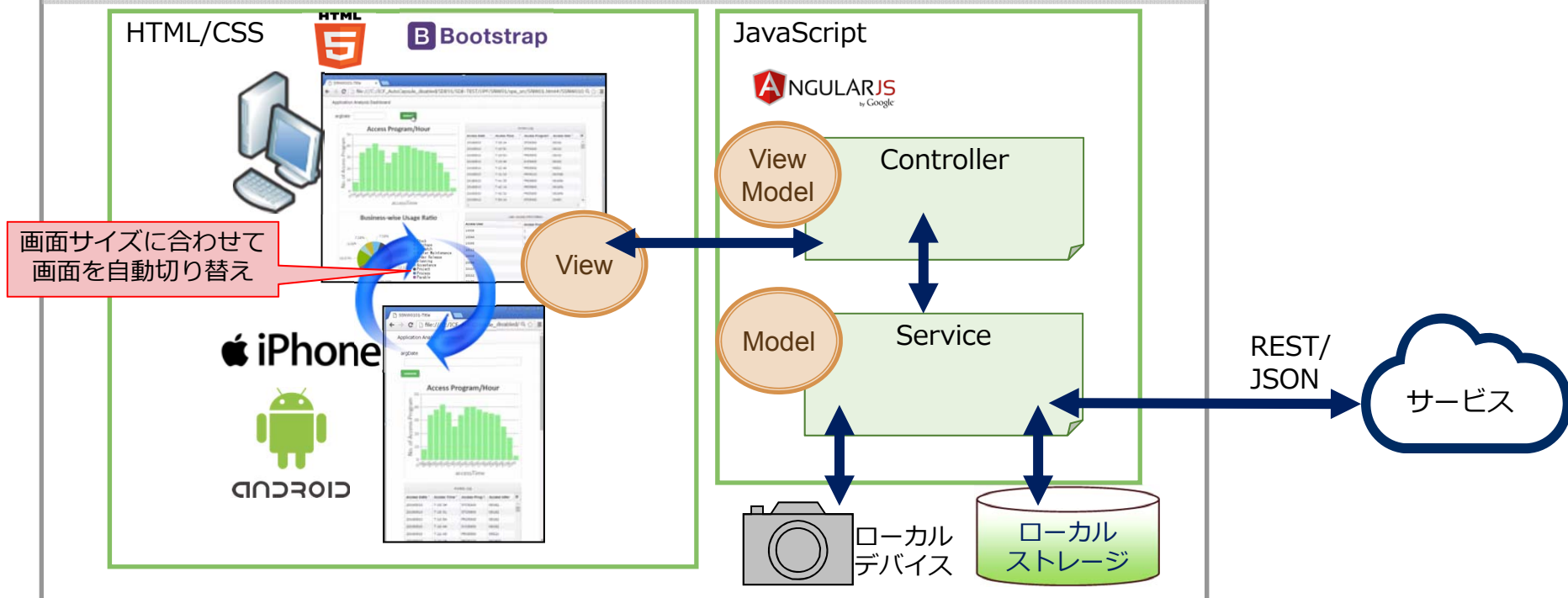
3.2.1. HTML5拡張モデル(SPA)の新設

サーバと疎結合で保守性の良いSPAアプリケーションを自動生成



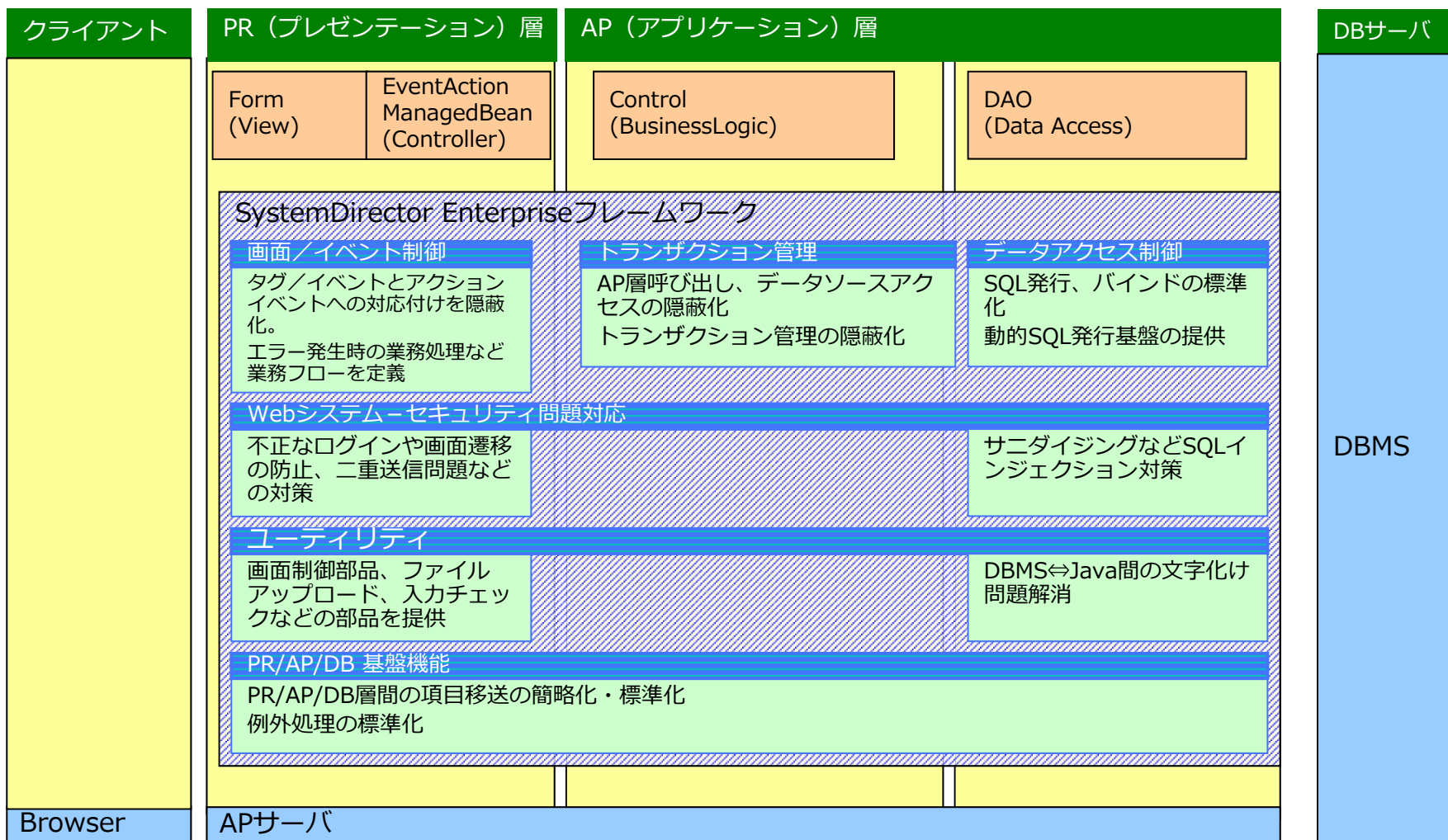
- HTML5、Bootstrapによる表現豊かなユーザインタフェースを提供
- レスポンシブウェブデザインにより、マルチデバイス要件にも対応可能
- Angular、MVVMアーキテクチャによる保守性の高いソースコード
- オフライン要件にも対応可能

クライアント



3.3.フレームワーク

Web 3 層システムにおける、業務システム構築に必要なフレームワークを提供することで、業務SEは業務ロジックに注力可能です。



3.3.1. ユーティリティ（共通部品）

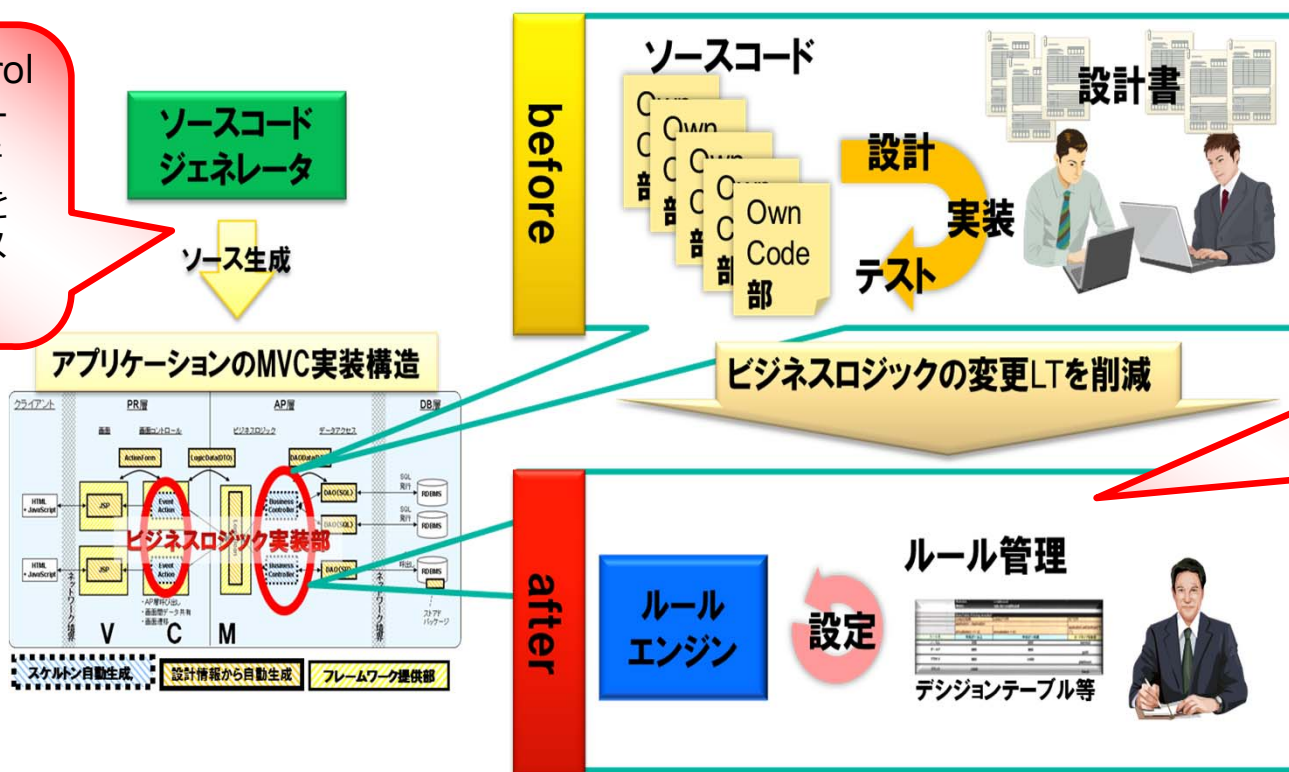
部品の機能一覧

- 算術関数機能
- 統計関数機能
- 文字列操作機能
- SQL文エスケープ機能
- 日付・時刻フォーマット機能
- 数値フォーマット機能
- サニタイジング機能
- 文字列比較機能、文字列変換機能
- 文字種検査・サロゲートペア検査機能
- エンコーディング機能
- メール送信機能
- CSV機能、ファイル操作機能
- GZip圧縮／展開機能、Zip圧縮／展開機能
- 外部プログラム実行機能
- ストップウォッチ機能、タイマー実行機能
- DNS問い合わせ機能
- クライアント情報取得機能
- 非永続マップ機能
- メッセージダイジェスト算出機能、対称鍵暗号機能、バイナリ符号化機能

3.3.2.BRMS実行部品

BRMS適用を効率化する部品であるBRMS実行部品を提供します。BRMSを利用したシステムでは、業務ロジックをデシジョンテーブルとして表現し、ソースコードと分離する事により保守性向上や工数削減の効果が期待できます。

AP層SubControl
の自動生成コード
にBRMS実行
部品の呼出しを
ガイドするコメ
ントを出力

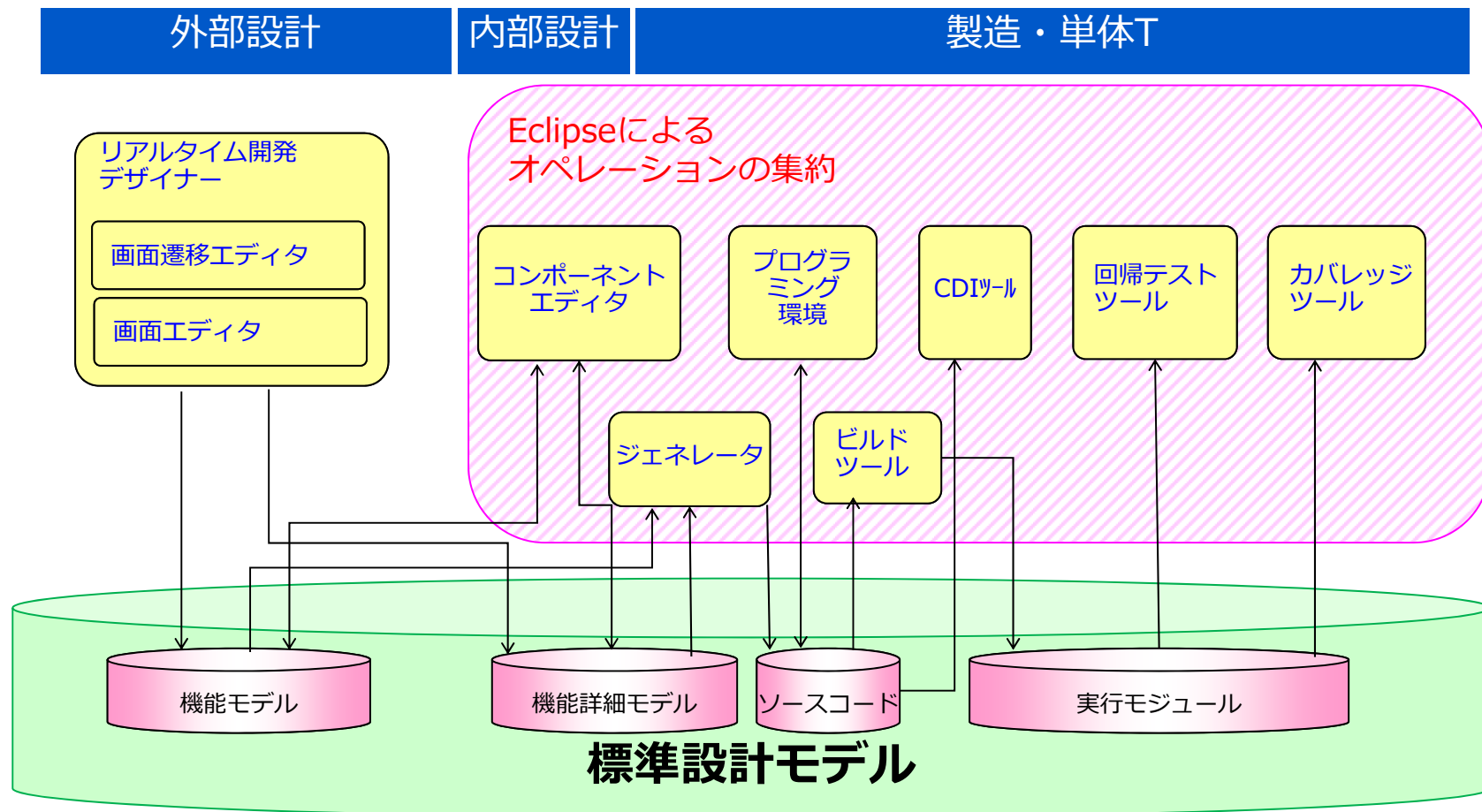


BRMS適用
を効率化す
る部品であ
るBRMS実
行部品を提
供

3.4.開発ツール

外部設計から製造・単体テストまでを対象に開発ツールを提供します。

各フェーズの成果物は標準設計モデル(リポジトリ)として格納され次工程と連携します。



3.4.1.リアルタイム開発デザイナー

要件定義のその場でA Pの作成を支援するツールです。テンプレートを活用しドラッグ&ドロップで開発でき、開発期間・開発コストを削減します。

Step1: 事前準備
テンプレート作成

レイアウト共通化も
実現可能

・画面、画面遷移、業務ロジックの自動生成のためのプロジェクト用ひな形整備

リアルタイム開発デザイナー

Step2 : 画面の遷移情報や処理概要を整理

機能概要入力

画面処理説明入力

画面遷移定義

画面エディタを起動

画面レイアウト編集

プロパティ設定

部品選択

ドラッグ&ドロップで自由に画面をデザイン

HTML5タグも利用可能

JavaScript編集

Step3 : ドラッグ&ドロップでユーザーと画面をデザイン

部品定義ファイル

プロジェクトカスタマイズ可能

Step4 :
ワンクリックで設計書とソースを自動生成

SystemDir

設計書

ソースコード

Excel
表示用データ

APサーバ

Step5 :
実際に動かしながらユーザーと要件を最終確認

3.4.2.設計エディタ・ソースコードジェネレータ

- 主に詳細設計作業を支援する設計エディタ（コンポーネントエディタ）を提供し、データ定義やマッピング設計などを支援します。
- 設計情報をもとにソースコードを自動生成することで、品質・生産性向上を実現します（全コードの15%～35%程度を自動生成します）

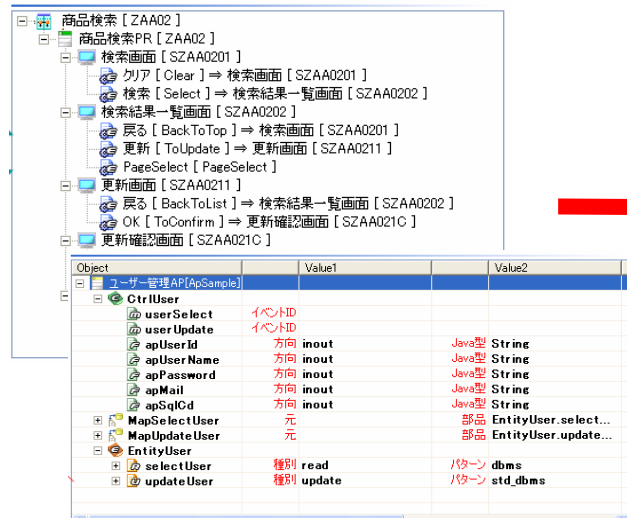
可読性の高い
コードを生成

雛形ファイル

```
package ${packageName};  
public class ${name}Action {  
    public void execute() {  
    }  
}
```

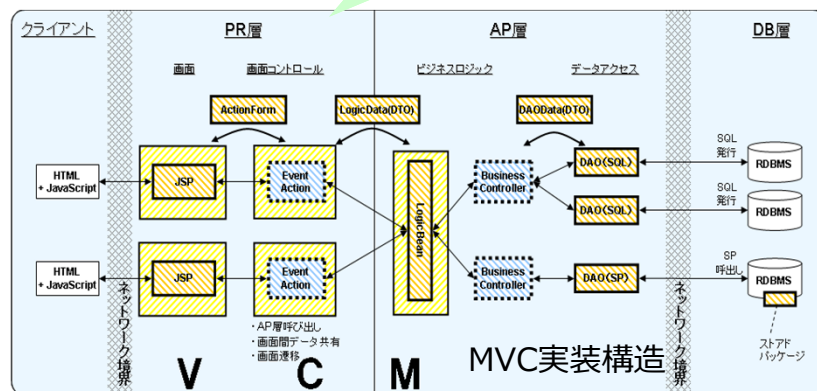
業務ロジック追記型で
柔軟性と生産性を確保

コンポーネントエディタ



ソースコード生成

ソースコード群



追記した業務ロジックは保護され
繰り返し自動生成が可能

スケルトン自動生成

設計情報から自動生成

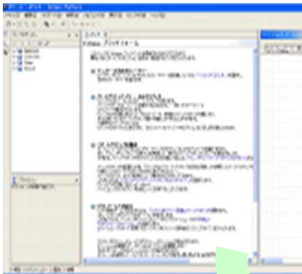
フレームワーク提供部

3.4.3.テスト支援ツール

■ オウンコーディング部分の品質を高めるCDIツール、テスト品質の向上を支援するカバレッジツールを同梱し、ガイドと合わせて提供します。

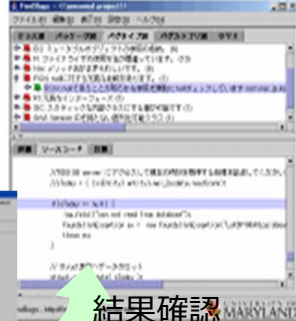
CDIツール コーディング規約をチェックしてソースを均一化

各業務開発者の環境



CDIの実行

指摘箇所のソース修正



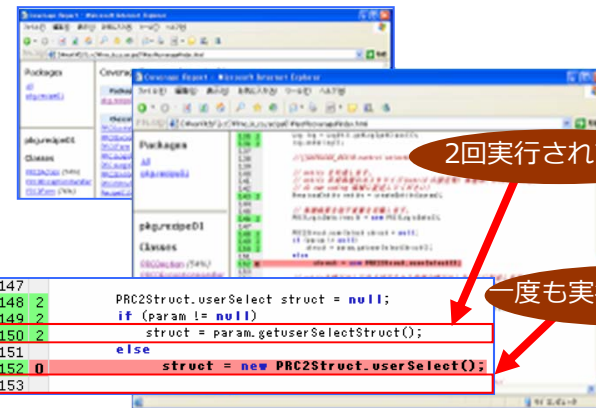
結果確認

CDI : CoDeInspection

カバレッジツール

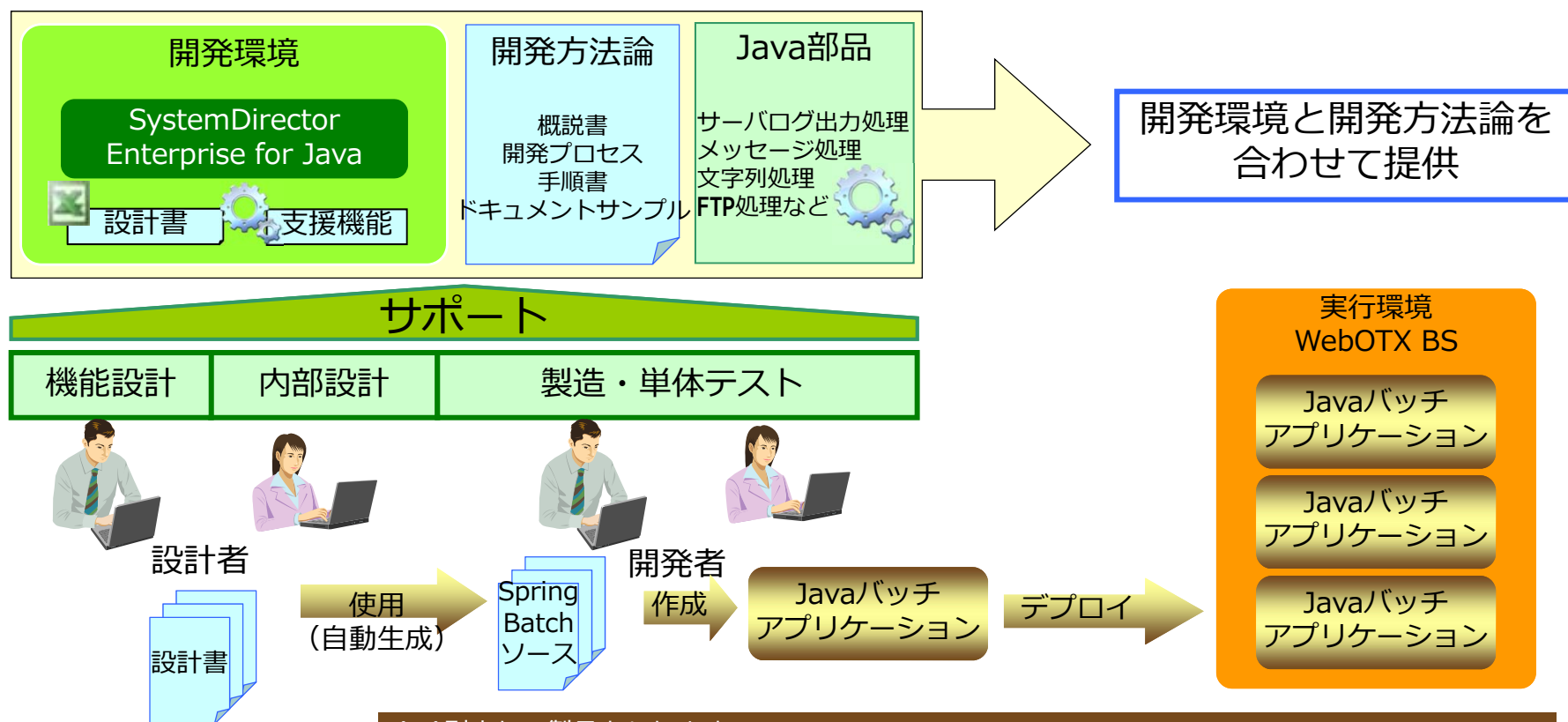
網羅性を保証し確証としても活用

テストカバレッジ率測定確認ツールを利用し、アプリケーションにおけるテスト網羅率を把握することができます。



3.4.4.バッチアプリケーションの開発支援機能

Webアプリケーションのみでなく、簡易なバッチアプリケーションの開発も可能です。開発方法論、WebOTX Batch Server(※) および Spring Batch に対応したソース生成機能により、煩雑なソースファイル、XML 定義ファイルの作成工数を削減します。



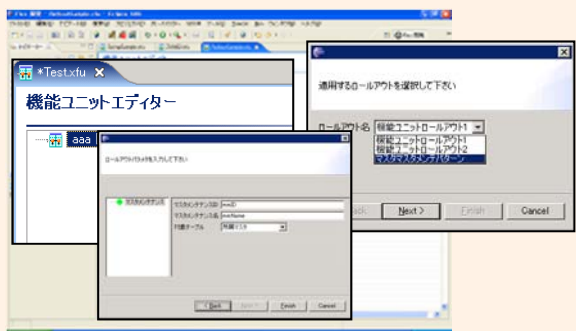
(※)別売りの製品となります。

(注)大規模なバッチ開発の場合は、SystemDirector Enterprise for Batchの適用を推奨します。

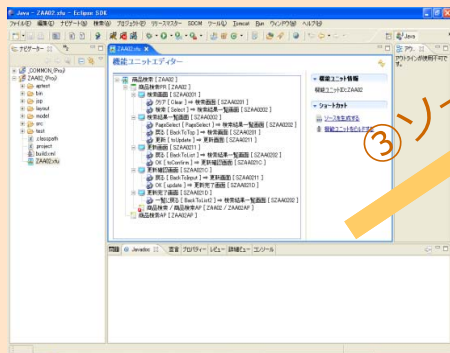
3.4.5.ルールアウト機能

あらかじめ業務画面のパターンをテンプレート化しておくことで、画面レイアウト、画面遷移制御、入力コントロール、DBアクセス処理、RESTサービスなどを一気に自動生成できます。

①パターンを選択しパラメータ入力

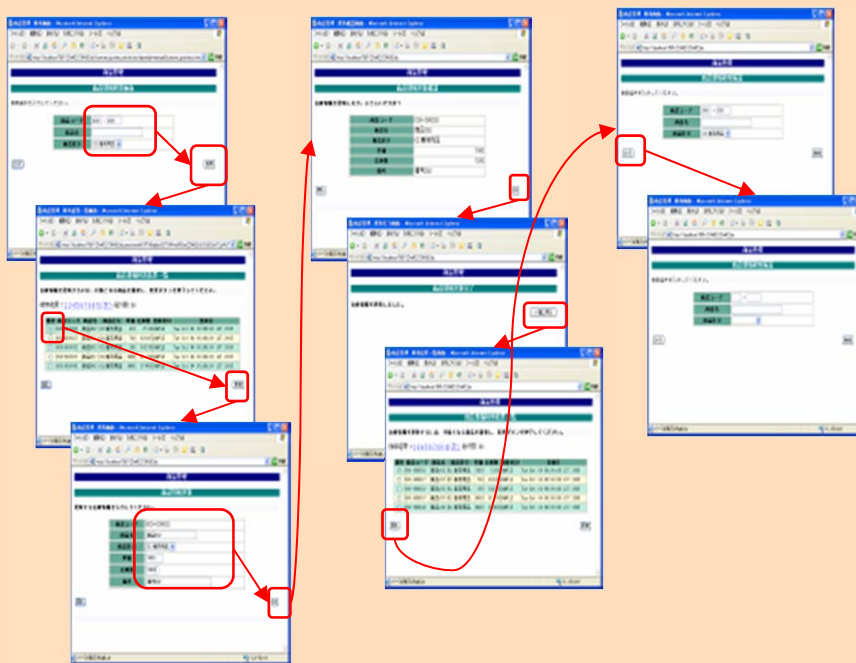


②内部設計データを一括生成



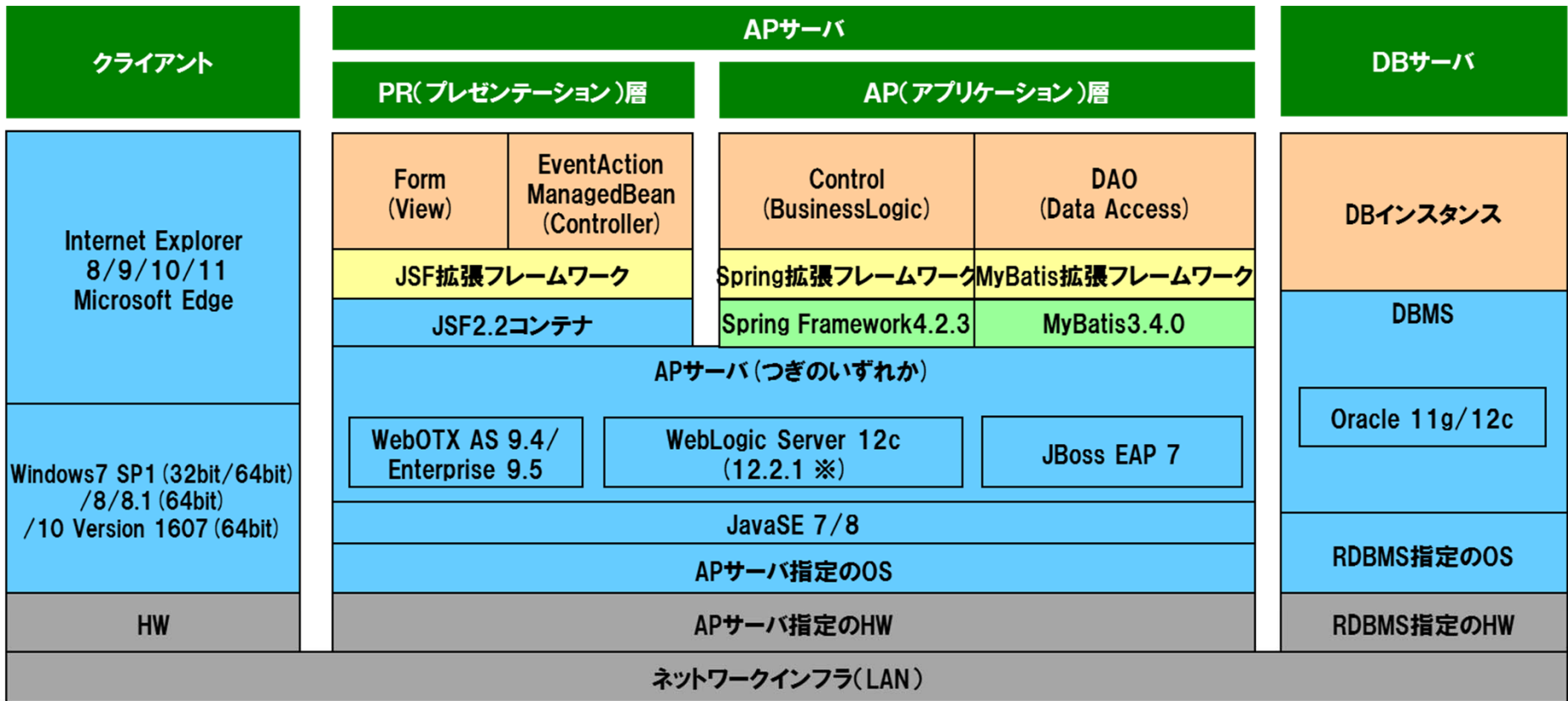
③ソース生成&ビルド

ビジネスロジックの追加のみで
アプリケーション実行可能



3.5.動作環境

JSF拡張/Spring拡張/MyBatis拡張モデル



凡例

ハードウェア

調達品ソフト

SystemDirector Enterprise for Java OSSライブラリ

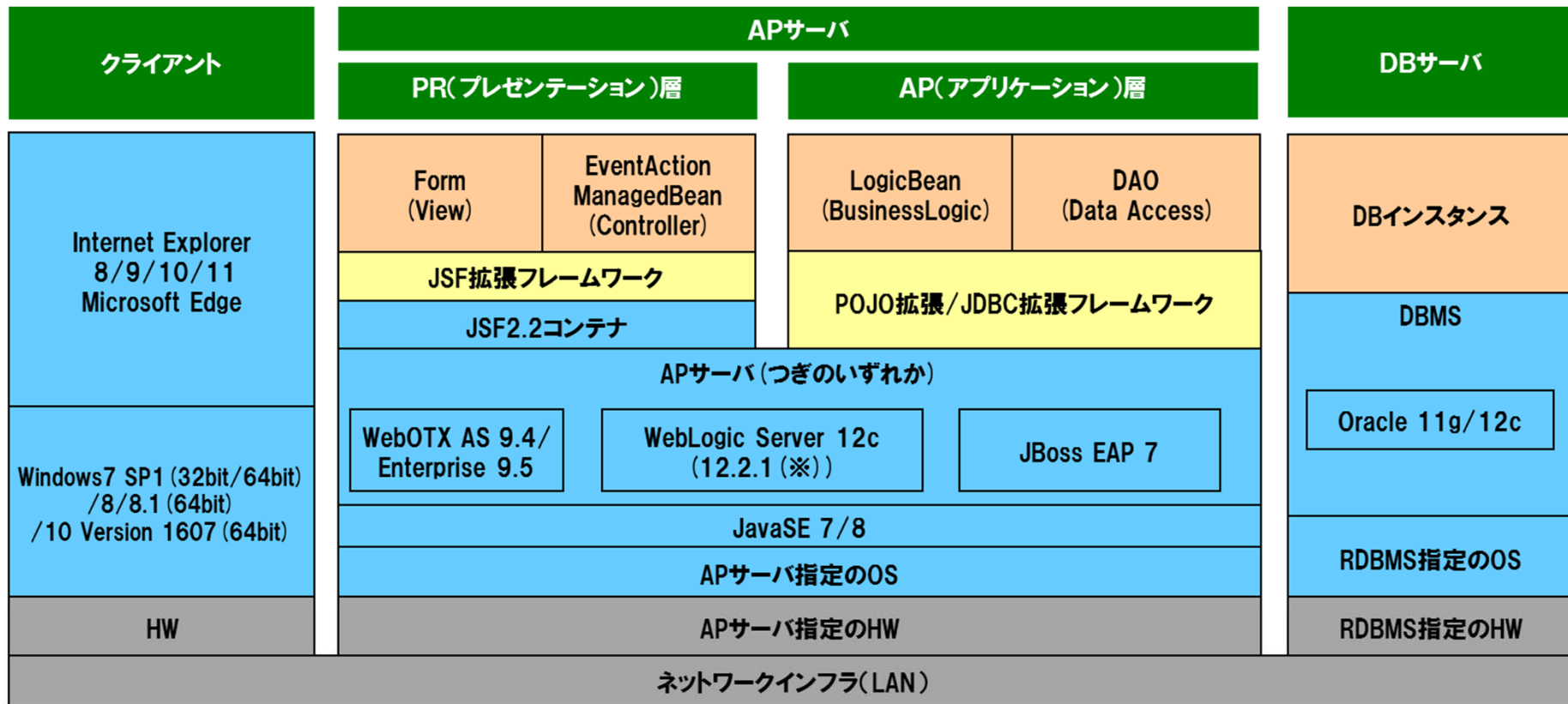
SystemDirector Enterprise for Java
自製ライブラリ

プロジェクトが開発した資産

※: WebLogic Server 12.2.1は、Oracleが正式にサポートする64bitのみの対応となります。

3.5.動作環境

JSF拡張/POJO拡張/JDBC拡張モデル



凡例

ハードウェア

調達品ソフト

SystemDirector Enterprise for Java
OSSライブラリ

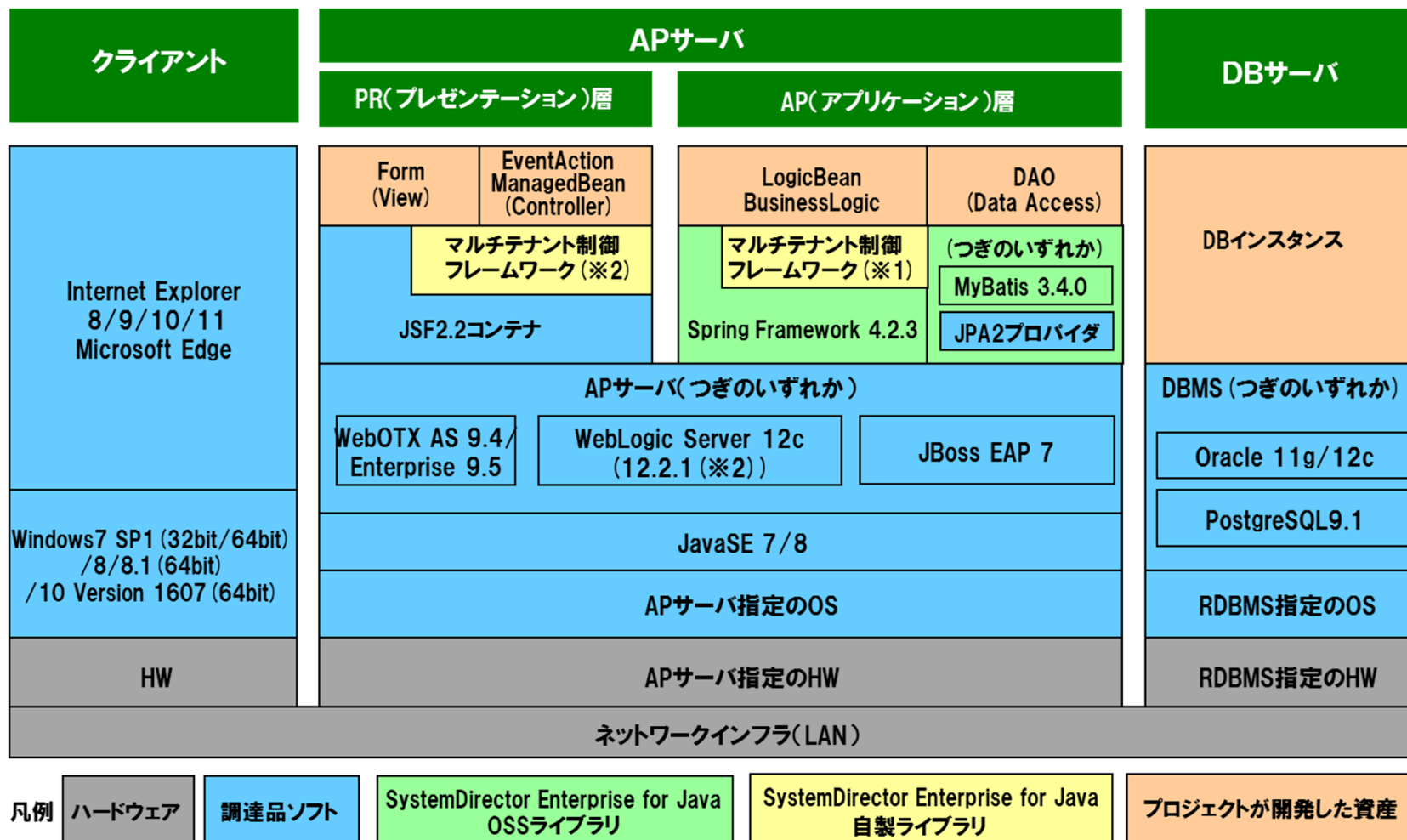
SystemDirector Enterprise for Java
自製ライブラリ

プロジェクトが開発した資産

※:WebLogic Server 12.2.1は、Oracleが正式にサポートする64bitのみの対応となります。

3.5.動作環境

JSF/Spring/MyBatis・JPAモデル

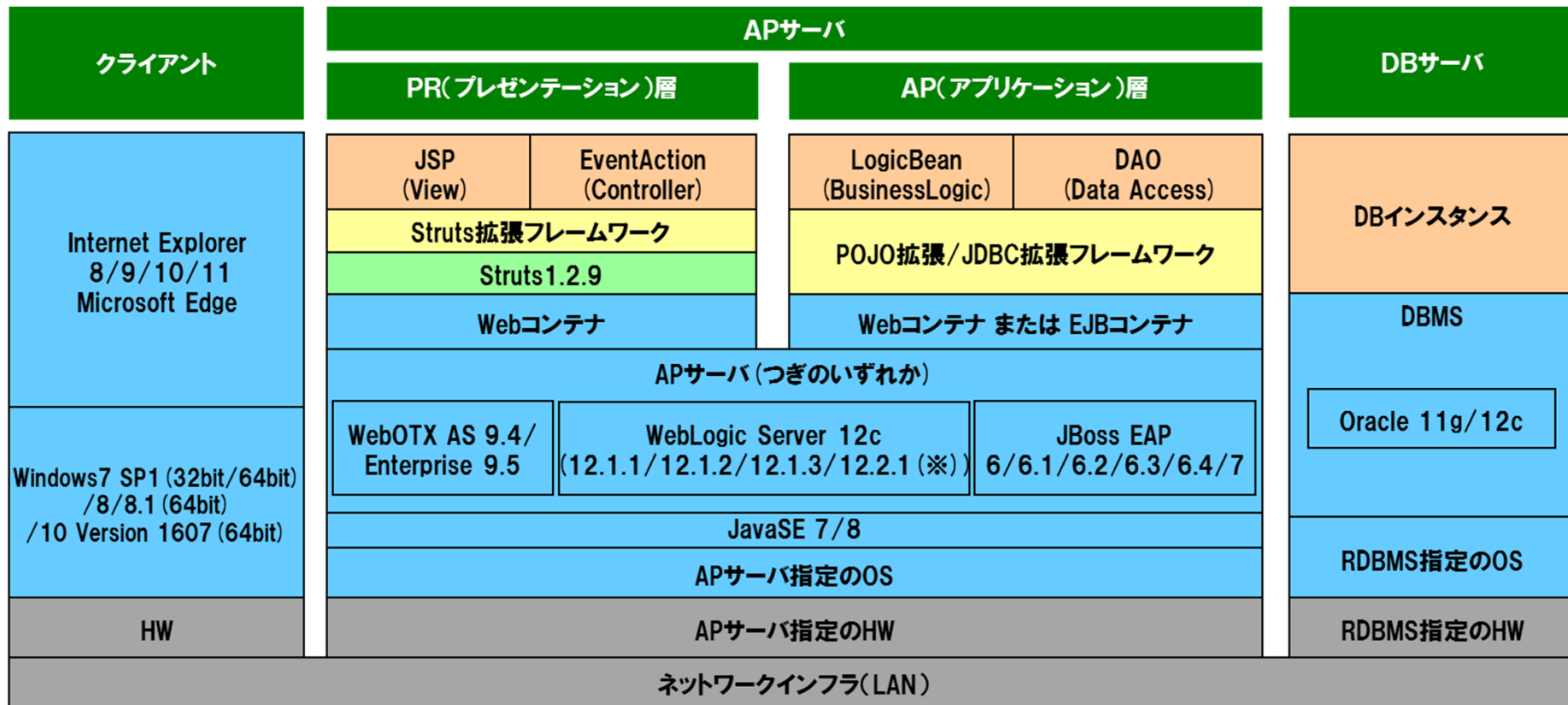


※1: マルチテナントアプリケーション開発時に適用。Facelets利用時に一部未対応

※2: WebLogic Server 12.2.1は、Oracleが正式にサポートする64bitのみの対応となります。

3.5.動作環境

Struts拡張/POJO拡張/JDBC拡張モデル



凡例

ハードウェア

調達品ソフト

SystemDirector Enterprise for Java
OSSライブラリ

SystemDirector Enterprise for Java
自製ライブラリ

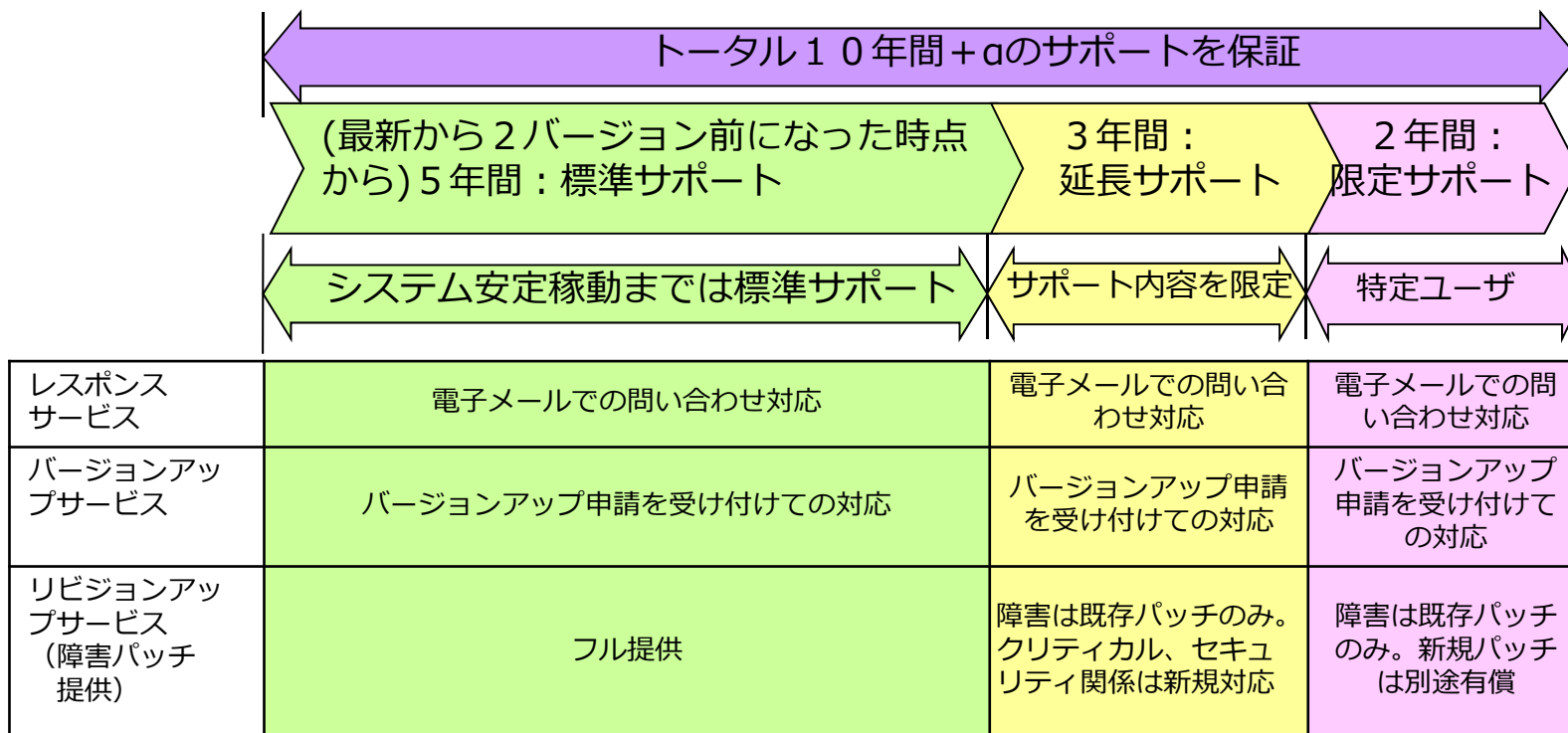
プロジェクトが開発した資産

※:WebLogic Server 12.1.3、12.2.1は、Oracleが正式にサポートする64bitのみの対応となります。

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 －サブレット・JSP－	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

ご購入前のお問い合わせ

NEC SystemDirector Enterprise お問い合わせ窓口

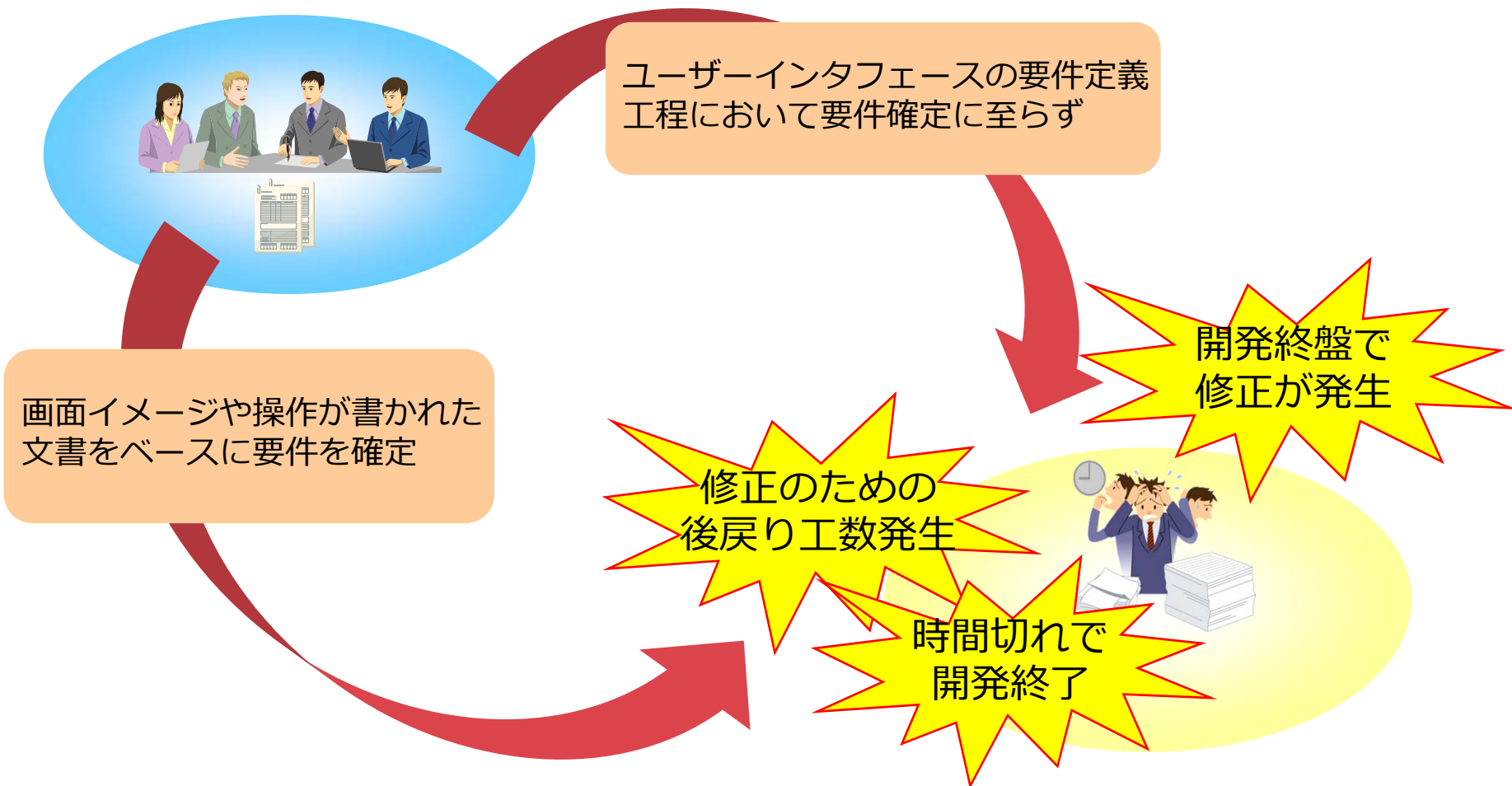
Web

<http://jpn.nec.com/SystemDirectorEnterprise/contact.html>

記載されている会社名、製品名は、各社の登録商標または商標です。

付録 リアルタイム開発

要件定義工程の時点では仕様が曖昧で、開発後半に認識の相違やイメージが異なっていることが判明し無駄な工数、費用が発生



UIの要件確定を上流の早い段階で短期に実施

発注者、利用者の
CSが向上します

利用者

発注者
(レビュー)



開発者



・動くAPを利用者に
渡して要件確認

- ・実際に動くAPにデータを表示して合意
- ・設計書も出力してレビュー
- ・微調整はその場で修正・再作成して確認

リアルタイム開発の流れ

事前準備

画面エディタ（リアルタイム開発デザイナー）

動作確認

Step0

共通的に利用できる
AP雛型の作成

Step1

画面の遷移情報や処理概要を整理

Step2

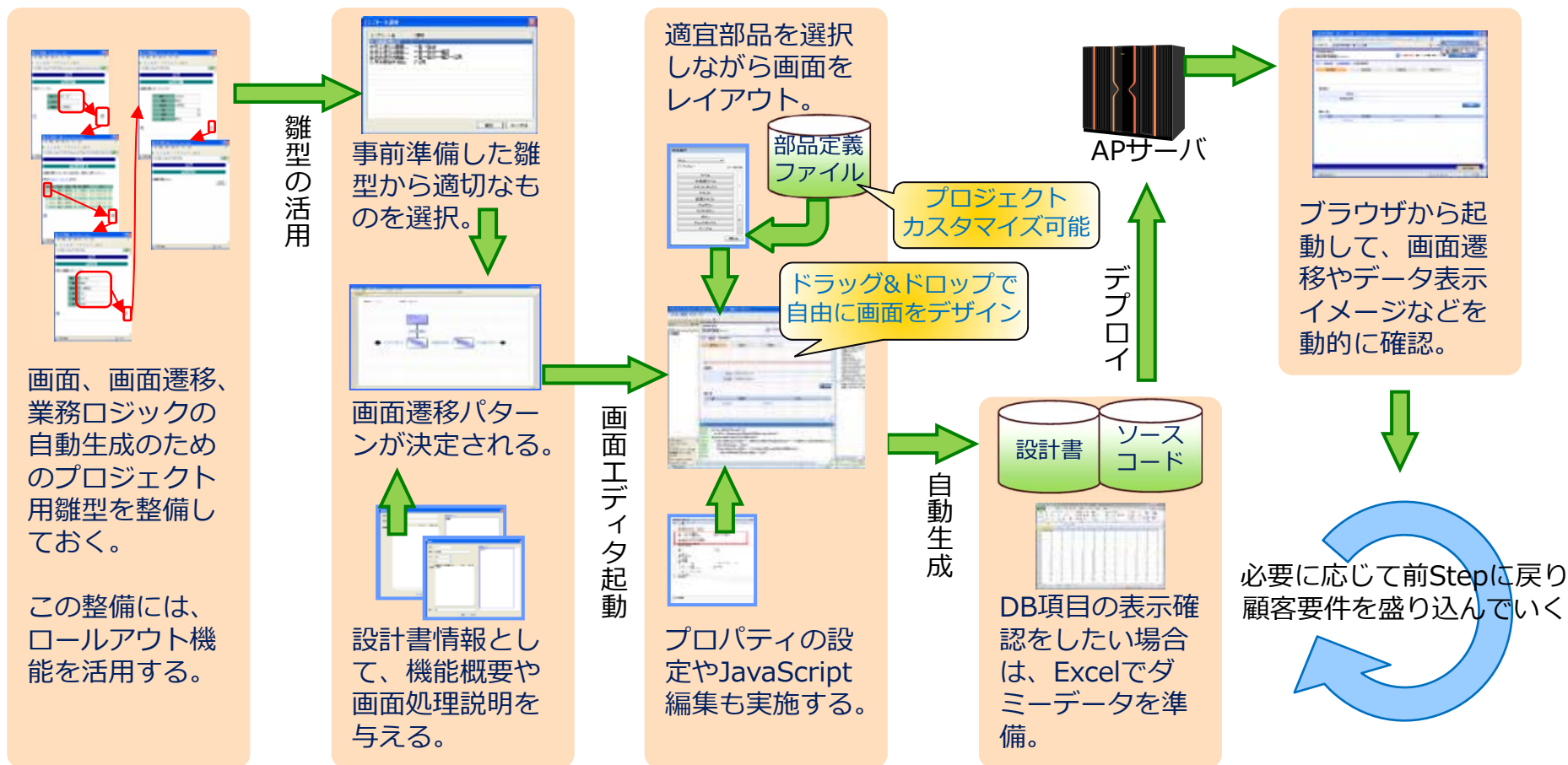
ドラッグ&ドロップで
ユーザと画面をデザイン

Step3

ワンクリックで設計書
とソースを自動生成

Step4

実際に動かしながらユーザと要件を最終確認



Step1. テンプレートによる画面遷移・画面パターンの選択・決定

- 既存システムを分析し、代表的な画面遷移パターンと画面レイアウトパターンをあらかじめテンプレート化（パラメータ化しており自由に変更可能）。
- **テンプレートをベースに、画面デザインを行うことで画面仕様決定の迅速化、標準化を促進**

テンプレートでは埋められない業務固有項目や、レイアウトや遷移の微修正は、画面デザイナーから追加定義可能。

複雑な業務要求、画面構成にも対応可能

画面デザイナーからテンプレートの選択・読込が可能。

GUIベースでイメージが見られるため、ユーザーとの認識齟齬を大幅削減。

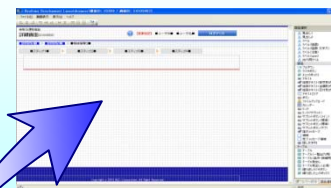
代表画面テンプレートから選択して画面定義のベースを決定。

画面検討を
検討型 ⇒ 選択型 ⇐



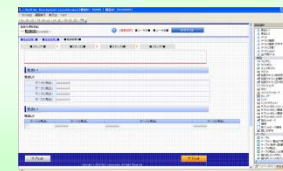
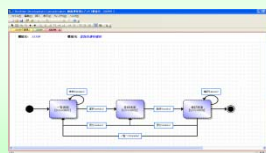
業務固有項目は
追加でカスタマイズ

リアルタイム開発デザイナー



選択

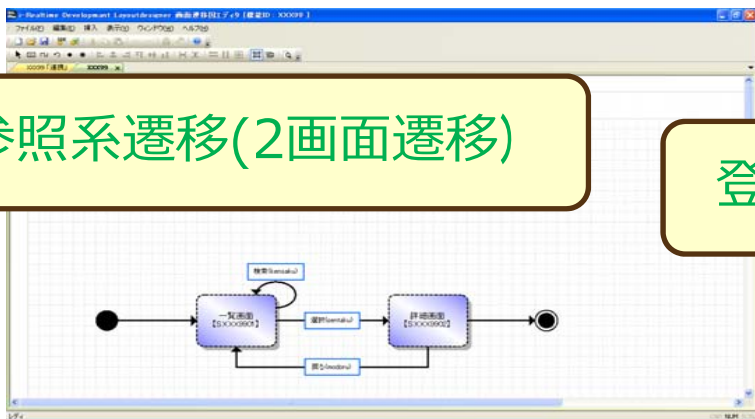
代表画面遷移・デザインテンプレート



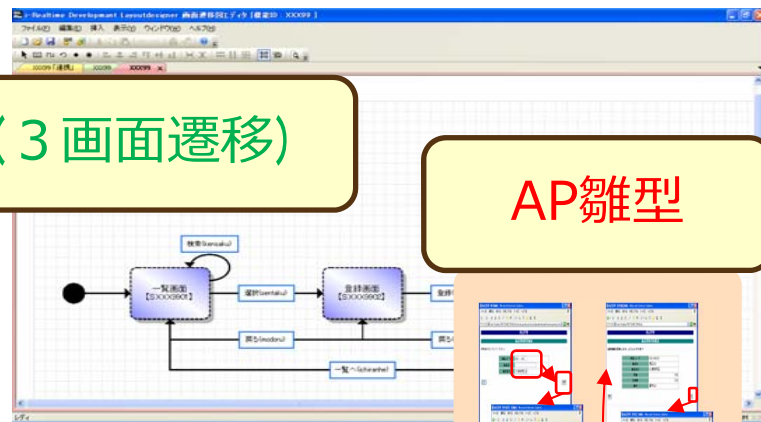
Step1. 事前準備～AP雛型の作成

対象システムの要件を分析し、画面、画面遷移、データファイル読込を実装したAP雛型を作成

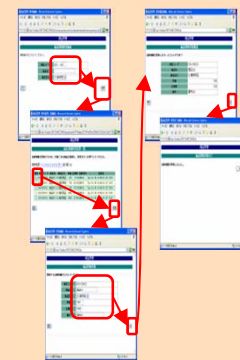
参照系遷移(2画面遷移)



登録系遷移(3画面遷移)



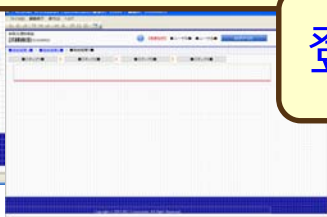
AP雛型



一覧画面テンプレート



詳細画面テンプレート



登録画面テンプレート



Step2. 要件確認～画面遷移設計

■ 事前に作成した画面遷移パターンから適切なものを選択

The screenshot displays the I-Realtime Development Layout Designer interface. On the left, a 'テンプレート選択' (Template Selection) dialog box is open, showing a list of templates. A blue arrow points from the '登録系遷移(4画面...)' template to the main editor area. The main editor area shows a screen transition diagram with the following flow:

```
graph LR; Start(( )) --> S1[支店検索画面 [S DEMO1 01]]; S1 -- 検索(kensaku) --> S1; S1 -- 選択(sentaku) --> S2[支店詳細画面 [S DEMO1 02]]; S2 -- 詳細(syousai) --> S3[社員詳細画面 [S DEMO1 03]]; S3 --> End((( )); S3 -- 戻る(modoru) --> S2; S2 -- 戻る(modoru) --> S1;
```

The diagram illustrates a workflow starting from a start node, moving to '支店検索画面 [S DEMO1 01]', then to '支店詳細画面 [S DEMO1 02]', then to '社員詳細画面 [S DEMO1 03]', and finally to an end node. There are also feedback loops labeled '戻る(modoru)' from the detailed screens back to the search screen.

Step3. 要件確認～画面デザインのプレビュー実行

- 事前に作成した画面パターンから適切なものを選択
- レイアウト変更(ボタンを移動、色を変更) 、コンポーネントをドラッグ&ドロップで配置

部品を選択して配置

JavaScriptコードを記述

JavaScriptコードを記述

```
00001 // エラーメッセージにメッセージを出力
00002 function createErrorMessage(element, msg) {
00003     var flg = true;
00004     var info = document.getElementById('message_area_e');
00005     for (var i = info.childNodes.length-1; i >= 0; i--) {
00006         if (info.childNodes[i].id == element.id + "_err") {
00007             flg = false;
00008         }
00009     }
00010 }
```

Step3. 要件確認～画面デザインのプレビュー実行

画面デザイナーからプレビュー実行してレイアウト・画面遷移・JavaScript実行結果を確認

The image shows a web application preview in a browser window and its corresponding design in a screen editor. A blue arrow points from the screen editor to the browser preview.

Screen Editor (Left): The interface shows a DOM tree on the left and a design canvas in the center. The canvas displays a form with search conditions and a list of branches. The design is for a page titled "社員情報検索機能 支店検索画面(SDEM0101)".

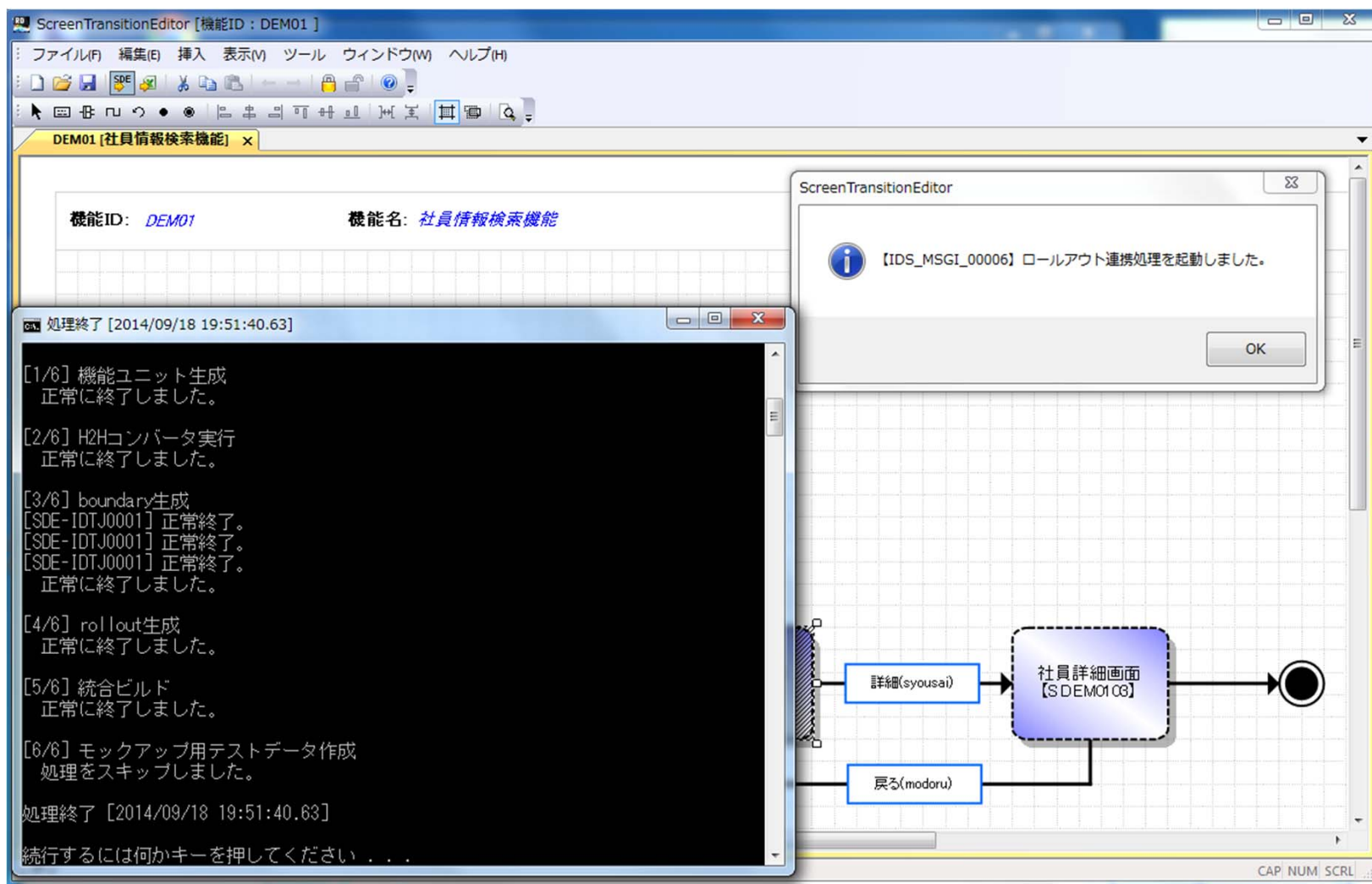
Browser Preview (Right): The browser shows the rendered web application. The page title is "社員情報検索機能 支店検索画面(SDEM0101)". The page includes a search bar, a search button, and a table of branches. The table has columns for "選択" (Select), "支店番号" (Branch Number), and "支店名" (Branch Name). The table contains one row with the values "C", "XXXXXXXXXX", and "XXXXXXXXXX".

JavaScript Code (Bottom): The code snippet shows a function `createErrMsg` that outputs an error message to the page.

```
00001 // エラーメッセージにメッセージを出力
00002 function createErrMsg(element, msg) {
00003     var_flg = true;
00004     var info = document.getElementById('message_area_e');
00005     for (var i = info.childNodes.length-1; i >= 0; i--) {
00006         if (info.childNodes[i].id == element.id + "_err") {
00007            _flg = false;
00008         }
00009     }
00010 }
```


Step4. ソースコードの自動生成（ジェネレーター）

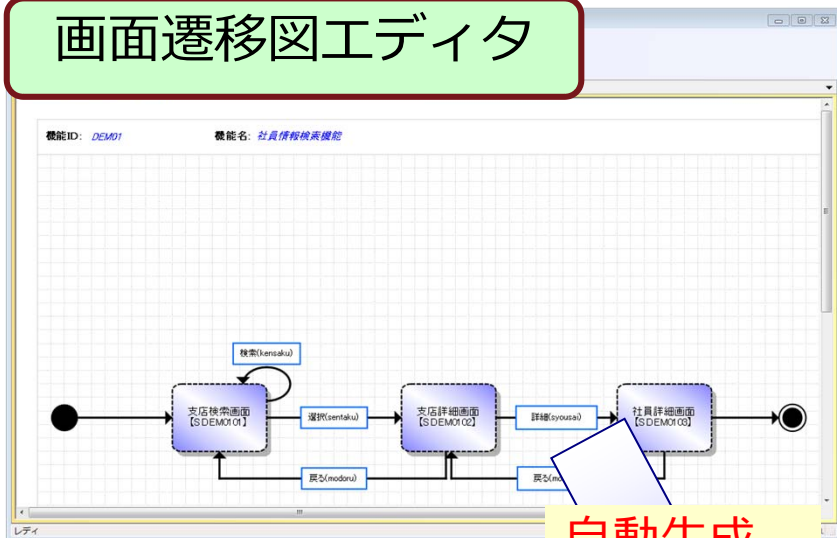
リアルタイム開発から実装コードを自動生成



Step4. 設計書の自動生成

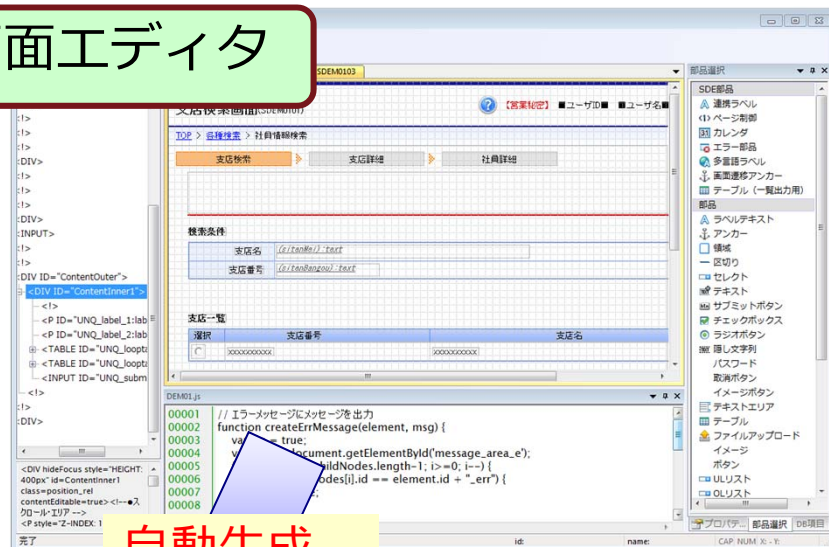
要件確定後、画面遷移図・レイアウト・項目定義など入力した内容を元に、
機能設計書を自動生成

画面遷移図エディタ



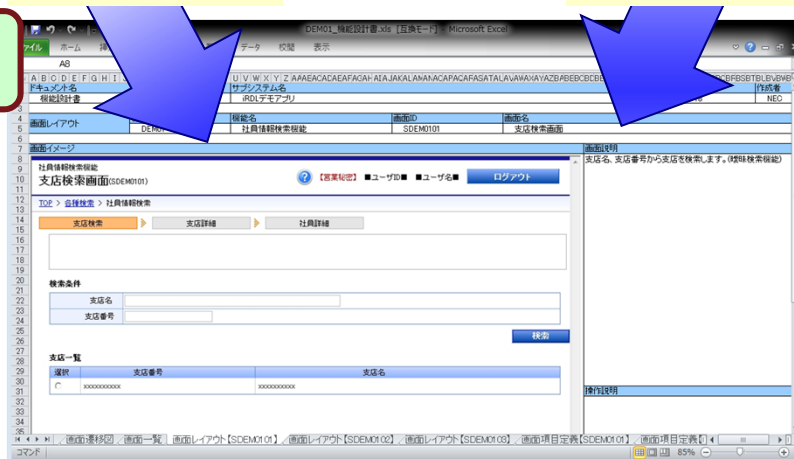
自動生成

画面エディタ



自動生成

機能設計書



Step5. AP実行～フォーマット、データ表示を確認

The screenshot shows a web application interface for branch search and an Excel spreadsheet with mock data. The web application is titled "社員情報検索機能 支店検索画面(SDEM0101)" and includes a search form with fields for "支店名" (Branch Name) and "支店番号" (Branch Number). Below the search form is a table titled "支店一覧" (Branch List) with columns for "選択" (Select), "支店番号" (Branch Number), and "支店名" (Branch Name). The table contains five rows of data. A red box highlights the search form and the table, with a green callout box stating "レイアウトを確認し、選択・支店番号の表示幅を調整" (Check the layout and adjust the display width of the selection and branch number). A blue arrow points from the "支店名" column header to the Excel spreadsheet, which is titled "DEM01APS..." and shows a table with columns "SITEN_CD" and "SITEN_NM". The Excel table contains the same five rows of data as the web application table. A green callout box at the bottom right states "モック用のExcelデータを画面に表示" (Display mock Excel data on the screen).

社員情報検索機能
支店検索画面(SDEM0101)

検索条件

支店名
支店番号

支店一覧

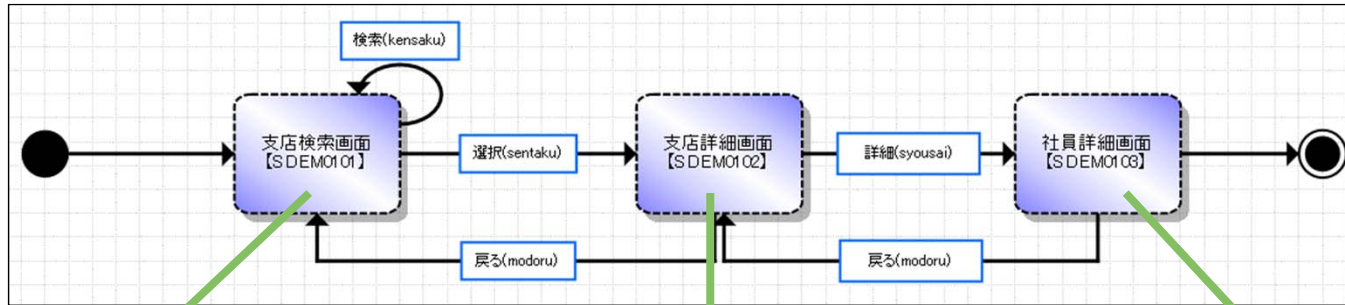
選択	支店番号	支店名
<input type="radio"/>	0000000001	東京
<input type="radio"/>	0000000002	大阪
<input type="radio"/>	0000000003	名古屋
<input type="radio"/>	0000000004	福岡
<input type="radio"/>	0000000005	札幌

レイアウトを確認し、選択・支店番号の表示幅を調整

モック用のExcelデータを画面に表示

	A	B
1	SITEN_CD	SITEN_NM
2	0000000001	東京
3	0000000002	大阪
4	0000000003	名古屋
	0000000004	福岡
	0000000005	札幌

Step5. AP実行～画面遷移を確認



ボタンを実行し、
画面遷移を確認

- SystemDirector Enterprise, InfoFrame は日本電気株式会社の登録商標です。
- SVF, SVFX-Designerは、ウイングアーク 1 s t 株式会社の登録商標です。
- Windows, Office, Excel, Visual Studio, .NET Framework 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。
- JavaはOracle Corporation の米国およびその他の国における登録商標です。
- 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。
- Log4jはThe Apache Software Foundationの登録商標です。
- 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 Inc. の米国およびその他の国における登録商標です。
- iOS はCisco Systems G.K.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