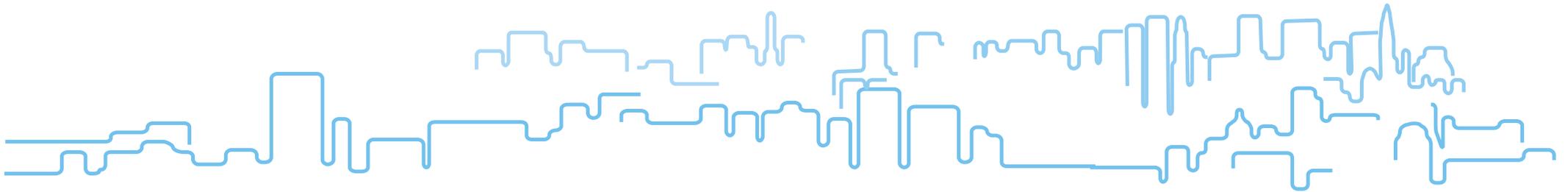




スクエア-freeセミナー 第83回

業務アプリケーションの開発作業を 自動開発ロボットで代替



開発・運営のサイクルの短縮化を強かに支援する
自動開発ロボット「E-Labor」

ワンダフルフライ株式会社



Powered by SaaSForce®



目次

I. 会社概要

II. SaaSForceプラットフォームの特長

III. DevOpsへの弊社からの提案

IV. 他社開発ツールとの違い

V. SaaSForce導入事例と効果（新規導入、マイグレーション）

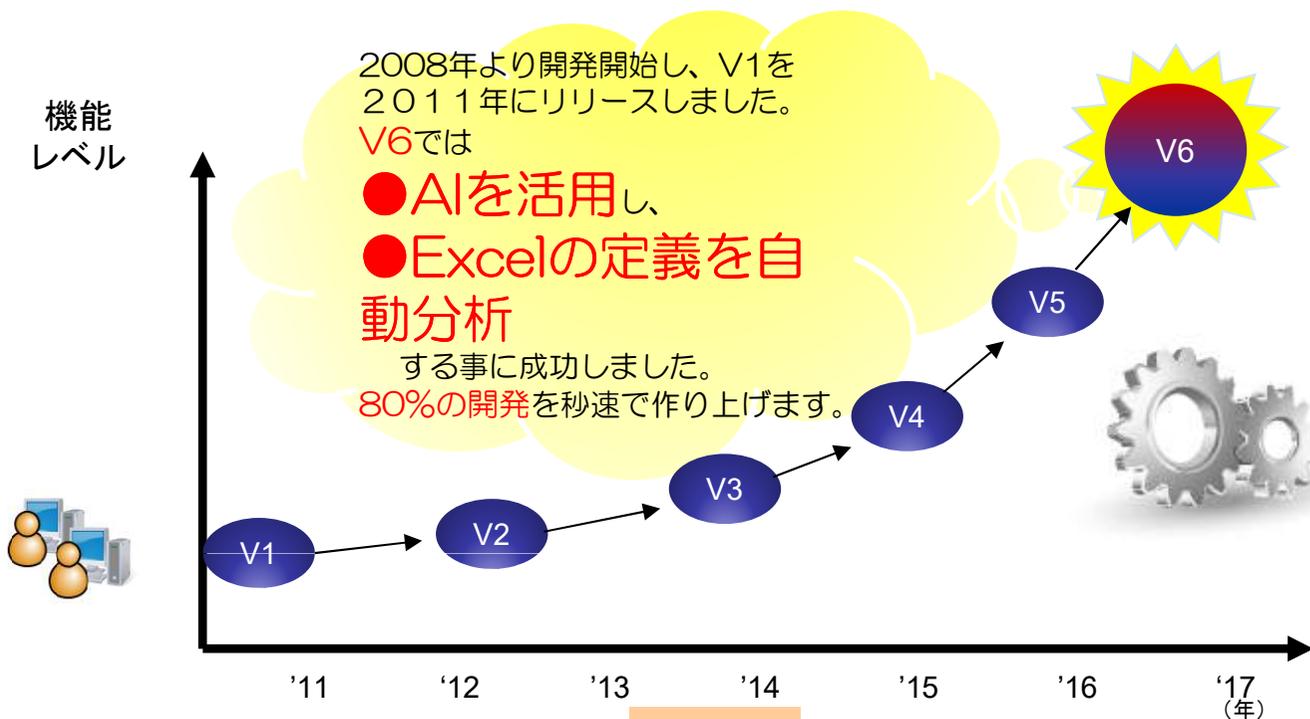
VI. 今後の展望

I. 会社概要



会社名	ワンダフルフライ株式会社
所在地	〒103-0001 東京都中央区日本橋小伝馬町4-2 VORT日本橋本町 TEL : 03-5695-8666 / FAX : 03-5695-8667
設立日	2003年7月25日
資本金	60,000,000円
企業理念	お客様に寄り添い、素晴らしい成功の旅立ちを共に追及・創造するグローバルな集団
事業内容	<ul style="list-style-type: none">・クラウドビジネス (SaaSForce[®] / 業務システムソリューション) 事業・各種システム構築支援・ITソリューションサービス事業
役員	代表取締役 平田 雅子

I. 会社概要 (SaaSForce事業)



<その他>

- 大手流通会社様
データ交換EDI管理システム
- グローバル総合機械メーカー様
営業管理・マーケティング分析システム
- グローバルコンサルティングフォーム様
プロジェクト実績管理システム
- 大型店舗グループ様
ECサイト買物代行システム
- 財団法人様
HR日報管理システム
... その他200社以上

事例紹介

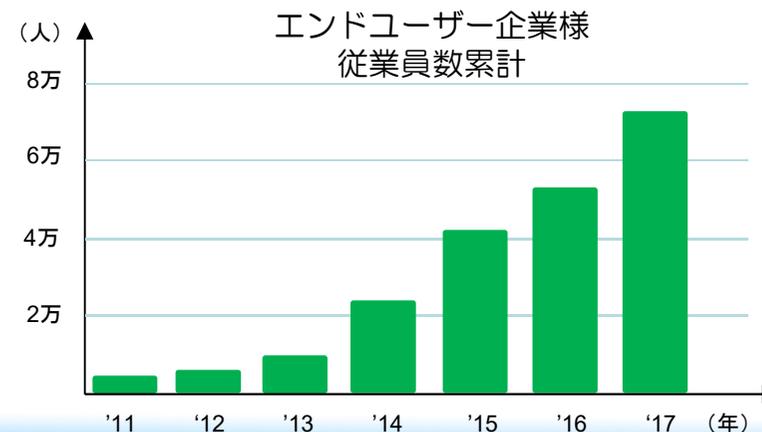
●大手銀行様の資金計画システム

10年前にVBで開発した資金システムをインフラ更改と同時にWEBシステムにリニューアル

システム開発(プログラミング)
工数

	通常開発	SaaSForce
主処理(280KStep)	46.2人月	21.0人月
画面(177画面)	35.4人月	2.2人月
帳票(120帳票)	15.0人月	0.6人月
合計	96.6人月	23.8人月

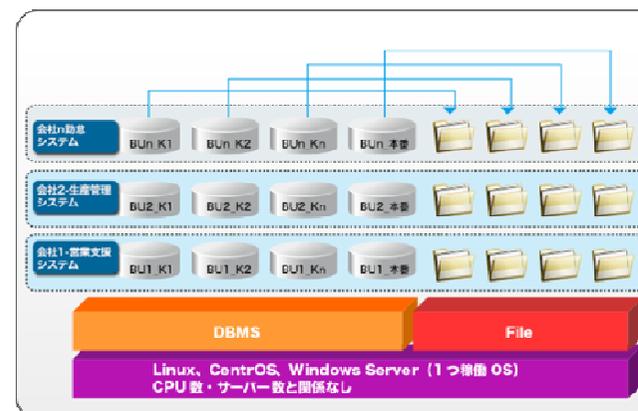
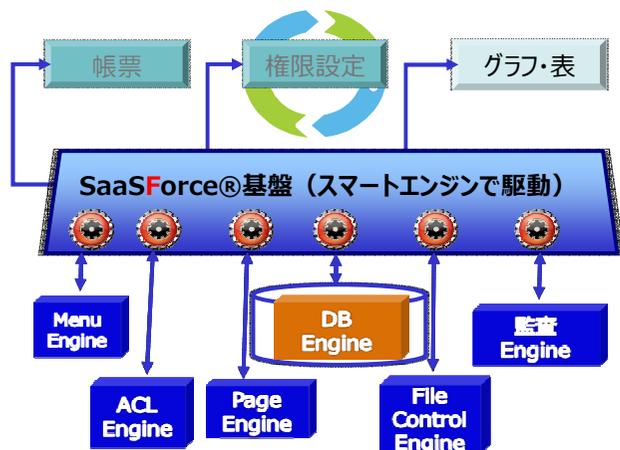
72.8人月(75.4%)削減!!



II. SaaSForceプラットフォームの特長



■ 業務アプリケーションの開発基盤(マルチテナント型)



- ・標準機能はモジュールを呼び出すだけ
- ・プログラミングも可能
- ・類似システムはコピー感覚で横展開
- ・運用しながら追加機能を開発

■ 変化する要求に即応し続けられる業務ソリューション

- ・マルチプラットフォームのサポート
- ・進化するデバイス連携(カメラ、センサー、GPS)
- ・ExcelによるWebアプリケーションの設計書を
E-Labor(**電脳ロボット**)導入で自動開発



Ⅱ. SaaSForceプラットフォームの特長



動作環境を選びません

オープンソース（JAVA）で開発された基盤は、ネットワーク、OS、DBの種類に制限されることなく、お望みの環境で稼働できます。



パブリッククラウド



プライベートクラウド



自社サーバ

非業務機能（ユーザー管理、権限管理、セキュリティポリシー、メール送信機能）を標準装備。

Server OS : Windows、Linux
データベース : MySQL、SQLServer、ORACLE他
Webブラウザ : Chrome（推奨）、IE11、Edge

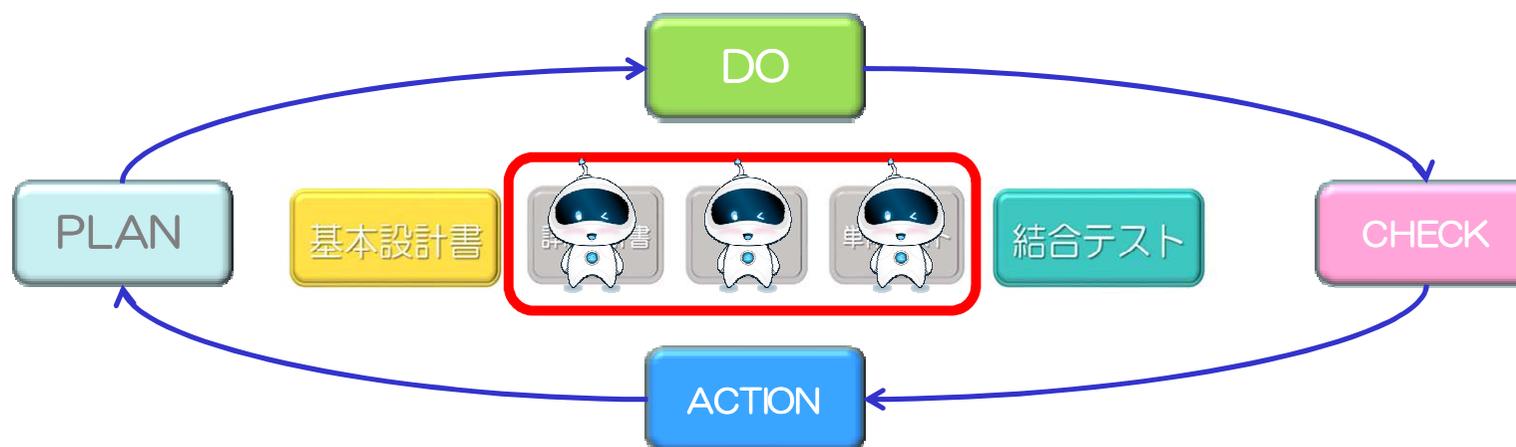
※本資料の内容は予告なく変更となる場合があります。記載されている会社名、製品名は各社の登録商標または商標です。

Ⅲ. DevOpsへの弊社からの提案



DevOps：開発と運用（PDCAフロー）との連携協力

システムジェネレータの活用

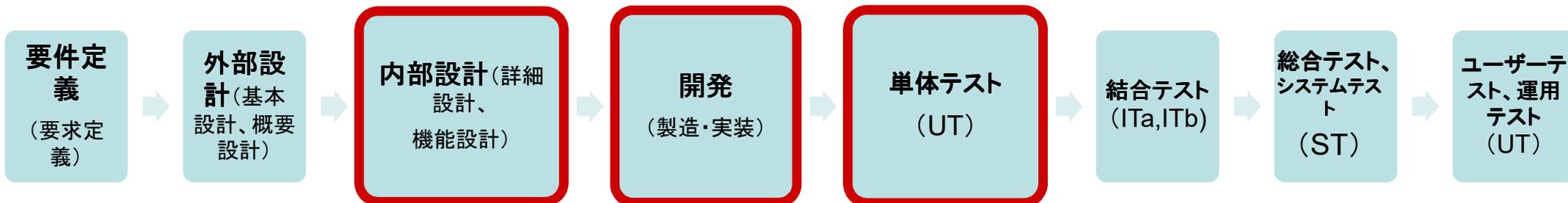


IT人材に関連する多くの課題は、開発基盤へのロボットの導入により解消します

開発業務：従来の設計手法を大きく変えずに、開発工程の自動化を図れる。

運用：Excel帳票等、慣れ親しんだ運営の良さを大きく変えずにPDCAを回せます。

Ⅲ . DevOpsへの弊社からの提案



モック画面

Excel上で画面配置出来る高い自由度

業務ロジックを日本語で記述

ExcelやPDF形式の帳表出力



修正したい部分だけでも繰り返し自動実行するよ！

コンパイル操作不要
すぐに動作確認できる

日本語記述の業務処理はコメント化されてソース生成

他システムとのCSV連携維持可能

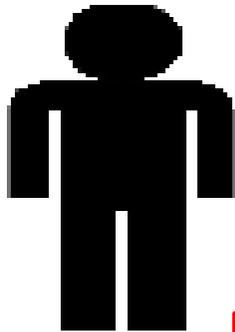
他事業や類似業務へのシステム展開が容易



ロボットが開発する意味

- 人材不足
- 人によるバラツキ
 - スキル
 - 勤務態度
 - 仕上がりに個性

長期なほど要員変更や追加要員対応等、**想定外のコストが掛かるケースが多い**



人
(JAVA技術者)

人件費



▲4割以上Off

ロボット費用



AI自動開発
ロボット

- 調達コスト不要
- 品質安定 (同一設計書)
- 残業なし
- 賃金? 算定単価は1ユニット
1ユニット=10機能分
 - ・画面数
 - ・帳票数
 - ・インポート種別数
 - ・エクスポート種別数 の合計

開発ロボットをIT要員として迎えるメリット

- ・人手不足対策 (社員のスキルマッチ、社外からのスキル要員調達)
※派遣法改正を含む要員調達コストの削減
- ・働き方改革を支援 (残業の削減、業務負荷の平準化、動機の維持)

IV. 他社開発ツールとの違い



他社ソリューションが抱える課題

本質的に、開発者の負荷は変わらない

設計書を見ながらの**再設定業務**

設計書を見ながら、再度、データモデリングを作成し、画面の設定や、それ以外の複雑な設定または開発が必要である。漏れがある場合も？

画面・帳票開発の限界

画面・帳票の細かな部分の作成は大変苦勞する。または、ツールではやらない。

モデルドリブンな開発手法を取り入れる必要がある場合がある

従来型開発手法でなくUML（モデル化）等、新たな手法を取り入れる必要がある場合あり。見積方法、フェーズの分け方、成果物、フェーズアウト・チェックポイントなど大幅な変更が必要。または、案件規模や技術者の習熟度で分ける場合あり。

ツールの制限がある

専用環境での運用が必要

運用時には、ツールの環境を利用する必要がある

開発元が国外の場合、対応スピード、柔軟性に課題が？

問い合わせ対応の遅延

グローバル開発拠点からの問い合わせに対し、レスポンスが遅い

価格交渉の柔軟性への課題

グローバルな価格戦略の中で、柔軟性に課題がある場合がある

次期開発機能要求の優先順位付けへの柔軟性の課題

グローバルな優先順位の中で決定

最新のテクノロジーに遅れている場合がある

Cobol言語には合うが、**JAVA対応**などに遅れている場合がある

SaaSForceの特長

国内開発実績から生まれた手法であり、確実に負荷を軽減する

設計書を元に自動化

設計書を元に、再度設定は不要。ロボットが設計書を読み込むことで、詳細設計から開発・単体テストを省く。基本設計から結合テストまで一気通貫で可能。整合性チェックも自動化。

自由度の高い画面・帳票開発

Excelにて設計した画面・帳票をそのまま取込み、既に用意されているコンポーネントで自由なUIを作ることが可能

従来型開発手法の中でも活用

従来型開発手法の中でも、その手法を変更することなく、詳細設計から単体テストフェーズまでそのまま適用できます。

OSS環境もサポートしている

OSS環境での運用も**可能**

ソース出力機能で、運用環境にSaaSForceを利用せず、OSS環境にても運用が可能

日本国産であり、柔軟で、対応が速い

問い合わせ対応が速く柔軟である

開発元は日本国内であるため、速く、柔軟に対応できる

価格戦略の共有

Win Winの戦略を、価格面で柔軟に共有することが可能

技術戦略の共有

日本国内で、今後、必須となるテクノロジーをお互いに共有することが可能
(フィンテックAPI、IOT API、ブロックチェーン、ERP API、、、、)

安定且つ実績がある新しいテクノロジーを採用

AI、自動化、予測、自然言語分析など積極的に採用

IV. 他社開発ツールとの違い



E-Labor（電脳ロボット）の分析処理

現在のExcel設計書に定義する内容（抜粋）：自然言語（中文、日本文）を自動判断

①処理指定（業務ロジック）：

例。「解除」をクリックすると以下の処理を実施。

A3より日付を取得。「終了日」を取得。「終了日」が空白なら

1. 「A4」に<A3>を設定
 2. その他の場合、「A4」に「終了日」を設定
- 処理終了。

例。[請求書コピー]ボタンの項目処理定義（検索一致する値を抽出してコピー実行）

```
5.「コピー請求書NO」に<請求書NO>を設定する。
6.データ操作2:テーブル「請求書明細」によるテーブル「請求書明細」にデータ登録,
「請求書明細」テーブルの項目「業務名、技術者名、数量内、単価内、金額内、数量外、
単価外、金額外、経費、合計」はテーブル「請求書明細」から取得するデータ項目「業務
名、技術者名、数量内、単価内、金額内、数量外、単価外、金額外、経費、合計」、
条件は「請求書NO」が<D23>と一致する、
「請求書NO」が<請求書NO>、「連番」が自動採番を設定(ループ開始)。
検索結果ループ終了。
7.「コピーLINK」ボタンを実行する。
処理終了
```

```
115
116 /*「コピー請求書NO」に<請求書NO>を設定する。*/
117 ActStr+="$( '#AB25' ).val( '' +SEIKYUUSHONO+ '' );";
118
119 /*データ操作2:テーブル「請求書明細」によるテーブル「請求書明細」にデータ登録,
120 row = triggerID.split('_')[1];
121
122 sqlStr.append("SELECT SYOUMUMEI,GIJUTSUSHAMEI,SUURYONAI,TANKANNAI,KINGAKUNAI,
123 param[paramIndex] = D23;
124 paramIndex++;
125
126
127 ay = exequeryarraylist(sqlStr.toString(),param);
128 sqlStr.setLength(0);
129
130 paramIndex = 0;
131 param = [];
132
133 if (ay != null && ay.size() > 0) {
134 for(var i=0;i<ay.size();i++){
135     SYOUMUMEI = ay.get(i).get(0);
136
```

IV. 他社開発ツールとの違い



E-Labor（電脳ロボット）の分析処理

- ②システム設定： [一般定義]システム名／初期表示画面／検索結果取得件数
 [セキュリティ関連]パスワード制限内容 [ログイン関連]ID数他（左下図）

一般定義	システム名	請求書管理システム
	初期表示画面	メニュー
	SaaSForceメニュー表示	非表示
	画面中央揃え	中央
セキュリティ関連	検索結果件数	100
	パスワード最少桁数(桁)	6
	パスワードの有効期間(日)	999
	ログイン時の失敗制限回数	3
ログイン関連	複雑さレベル(数字)	1

フォルダ	利用グループ	アクセス権限			
		表示	新規	更新	削除
全部	全部	あり	あり	あり	あり
オペレータ機能	テストグループ	あり	あり	あり	あり
管理機能	テストグループ	あり	あり	あり	あり
マスタ機能	テストグループ	あり	あり	あり	あり

- ③グループ別権限設定：全体／オペレータ／管理者／マスタ管理（右上図）
 ④テーブル設計：テーブルID／項目／ID／属性・桁数／必須／主キー／採番方法 **必須ではない？**

テーブルID	KOKYAKU	テーブル名	顧客マスタ	画面作成	インポート画面作成	
No	項目名	項目ID	属性	必須	主キー	値内容
4	会社名		文字列			
5	支払サイト		整数			
6	都道府県		文字列			
7	住所		文字列			Y

- ⑤ビュー設計：参照先テーブル&フィールド／結合情報[例. 営業担当=社員マスタID]
 ⑥画面設計：レイアウト／ボタンや入力項目／項目処理／検索指定／遷移先リンク条件／印刷／
 入出力指定／グラフ指定／**必須ではない？**
 ⑦帳票設計：Excelシート雛形（セル指定）／

IV. 他社開発ツールとの違い



業務アプリの更新配布が容易（サーバ内）

作成した業務アプリは開発環境から、テスト環境や本番環境への移送を3ステップで容易に移送

The screenshot displays a web-based interface for system migration management. On the left is a navigation menu with categories like 'テンプレート', '画面作成', '権限', 'ツール', '環境管理', 'システム', 'ユーザーとグループ', and 'セキュリティ'. The '環境管理' (Environment Management) section is expanded, and 'サーバ内移送' (In-server migration) is selected. The main content area is titled 'システム移送管理' (System Migration Management) and is divided into three steps:

- ① 移送先指定 (Destination Specification):** This step is highlighted with a red box. It contains a '移送元' (Source) field, a '移送環境元' (Source Environment) dropdown menu set to '0 開発環境' (Development Environment), and a '移送元バージョン' (Source Version) field set to '1.0'. To the right, a '移送先' (Destination) field is followed by a '移送環境先' (Destination Environment) dropdown menu with a '多環境設定' (Multi-environment setting) button, and a '移送先バージョン' (Destination Version) field.
- ② 移送準備（差分分析指定） (Migration Preparation (Difference Analysis Specification)):** This step is also highlighted with a red box. It features a 'ステップ別実行' (Execute by step) section with three buttons: '1: 移送準備' (1: Migration Preparation), '2: 差分分析' (2: Difference Analysis), and '3: リリース' (3: Release). To the right, there is a '一括実行' (Execute all) section with an '実行' (Execute) button.
- ③ リリース (Release):** This step is indicated by a red circle and is part of the overall migration process.

IV. 他社開発ツールとの違い



業務アプリの更新配布が容易（サーバ間移送）

作成した業務アプリは開発環境から定義体を丸ごと出力して他サーバへ移送する事ができます。

The screenshot shows a web application interface for server migration. On the left is a navigation menu with categories like 'テンプレート', '画面作成', '権限', 'ツール', '環境管理', 'サーバ内移送', 'サーバ間移送', '開発内容バックアップ', 'システム', 'ユーザーとグループ', and 'セキュリティ'. The 'サーバ間移送' (Server-to-server migration) option is highlighted with a red box. The main content area is titled 'サーバ間移送' and contains two tabs: 'リリースファイルエクスポート' and 'リリースファイルインポート'. The 'リリースファイルエクスポート' tab is active and highlighted with a red box. Two red arrows point to the tabs with labels: '① 定義体出力' (Definition body output) pointing to the export tab, and '② 移送先サーバにて定義体入力' (Definition body input at destination server) pointing to the import tab. The export form includes fields for 'リリース元環境' (Release source environment) set to '0 開発環境', 'エクスポートパターン' (Export pattern), and 'リリース対象選択' (Release target selection). It also has buttons for 'エクスポートパターン作成' (Create export pattern), 'エクスポートパターン削除' (Delete export pattern), '全システム情報' (All system information), and '個別機能選択' (Select individual functions). Below these are fields for 'ファイルパスワード' (File password) and 'ファイルエンコード' (File encoding), with a 'ファイルエクスポート' (File export) button. A note at the bottom states: '※「ファイルエクスポート」ボタンをクリックすると、選択されているシステム情報をダウンロードできます。「インポート機能」を利用して別のサーバに移送することが可能です。' (When you click the 'File export' button, you can download the system information selected. It is possible to migrate to another server using the 'Import function').

V. SaaSForce導入事例と効果（新規導入、マイグレーション）



II-1 マイグレーション事例 大手都市銀行の資金計画システム

VBによる拠点毎システムをWebアプリとして統一。
VBからJavaの変換ツールを活用し、運用開始まで1年。

システム開発（プログラミング）工数比較（テスト工程含まず）

	通常開発	SaaSForce
主処理（280KStep）	46.2人月	21.0人月
画面（193画面）	35.4人月	2.2人月
帳票（132帳票）	15.0人月	0.6人月
合計	96.6人月	23.8人月

72.8人月（75.4%）削減!!

運用改善：一括管理を行えるようになったことで、全社での資金予測、関連会社との資金調達スケジュールの調整が可能となり、1人当たりの1日の平均作業時間が1時間削減、外部連携コストも約半分削減された。

V. SaaSForce導入事例と効果（新規導入、マイグレーション）



II-2 新規導入事例 コンサルファームの大規模プロジェクトの工程管理に導入

プロジェクト全体管理ではなく

①要員の作業時間を一括管理

- ・各拠点のパフォーマンスの平準化を図る目的

②進捗報告資料

- ・発注元へのレポートの効率的な準備

II-3 その他導入事例

※既製品や既存の開発手法では実現が困難：コスト／納期／体制

- ・財団法人様のHR日報管理システムのマイグレーション（要件が細かい）
- ・大手家電量販店EDIシステムのマイグレーション（要件変更が頻繁）
- ・地方のホームセンターのECサイトリニューアルによる買物弱者改革（新発想）

VI. 今後の展望（ワンダフルフライの飛び先は）

システム開発の未来は？

人類の歴史は、

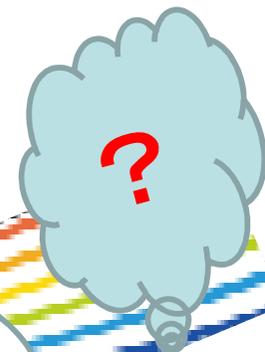
最初の一歩を踏み出すことから、
宇宙旅行へ



システム開発も同様

キーボードに一文字一文字入力してプログラムすることから、
ロボット自動開発まで進化しました

さらに将来は想像もできないことになります！





ご清聴いただきまして、真にありがとうございました。

