

既存パッケージの SaaS 化に向けた課題と解決策

前田和俊*
鈴木 剛*

要 旨

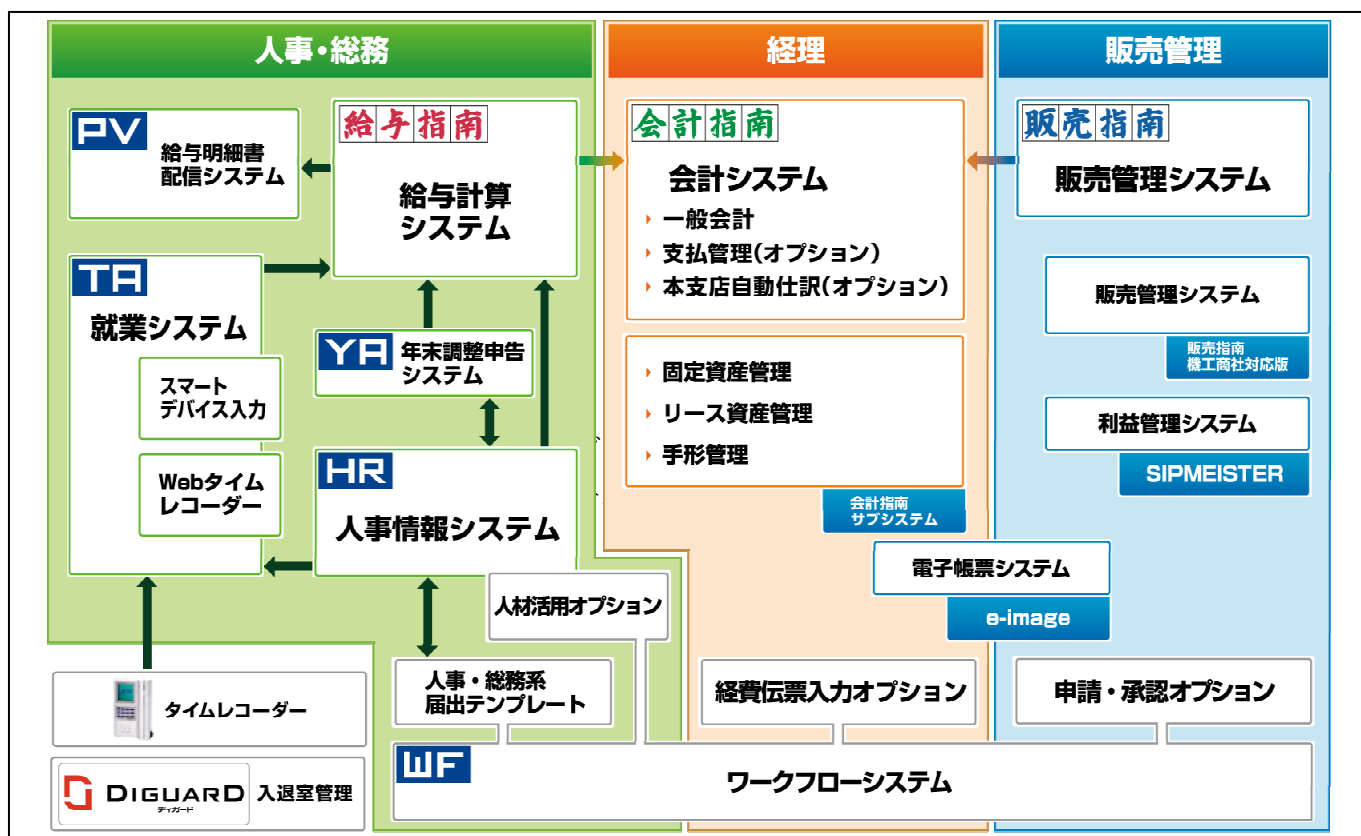
(株)三菱電機ビジネスシステム (MB) では、人事・総務トータルシステムの ALIVE SOLUTION と、基幹系パッケージの指南シリーズを開発・販売している。

MB では SaaS 化に向けた様々な技術検討を実施しており、本稿では、両シリーズのユーザインタフェースを実現するために開発したフレームワークと共通ソフトウェアコンポーネントを紹介し、SaaS 化に向けた当該フレームワークの課題と解決策について述べる。

ALIVE SOLUTION で使用するフレームワーク radish には、ユーザインタフェースを受け持つ各種のコンポーネント群

があるが、パッケージとしてイントラネット環境での使用を前提としている。事前検証により、ネットワークトラフィックに関する課題があり、パッケージをそのまま SaaS 化する事は困難であることを確認した。解決策としてコンポーネントの軽量化に取り組んでおり、その試作評価結果について紹介する。

また、指南シリーズは操作の応答性を重視して、クライアントサーバ方式を採用している。クライアントサーバ方式の SaaS 化に向けた課題と取組みを紹介する。



ALIVE SOLUTION・指南シリーズの構成

幅広い業種・業務で培ってきたノウハウと技術力を基盤に、ユーザーの抱える現場のニーズを解決するシステムを提供している。効率的・機能的な業務改善を目的としたパッケージであり、パッケージ単独の導入はもちろん、必要なパッケージを自由に組み合わせることで最適なシステムにカスタマイズすることも可能である。

1. ま え が き

ALIVE SOLUTION は、総務系の業務を対象とした人事・総務のトータルシステムで、2002年から販売を開始した。

指南シリーズは、経理系の業務を対象とした会計・給与計算システムと、販売管理業務を対象とした販売管理システムで、1998年から販売を開始した。

両シリーズとも各種機能追加、法改正対応、オープン化を進めるための開発言語の変更を行い、顧客ニーズにあわせ商品性を向上してきた。

本稿では、既存パッケージの ALIVE SOLUTION と、指南シリーズを概観し、ALIVE SOLUTION の SaaS 提供に向けた課題と解決策を中心に、両パッケージの SaaS 化への取り組みについて述べる。

2. パッケージの概要

2.1. ALIVE SOLUTION

ALIVE SOLUTION は、人事・総務部門の業務効率化と、戦略的人材活用をサポートする業種パッケージである。

ALIVE SOLUTION の構成は以下の通り。

① TA | 就業システム

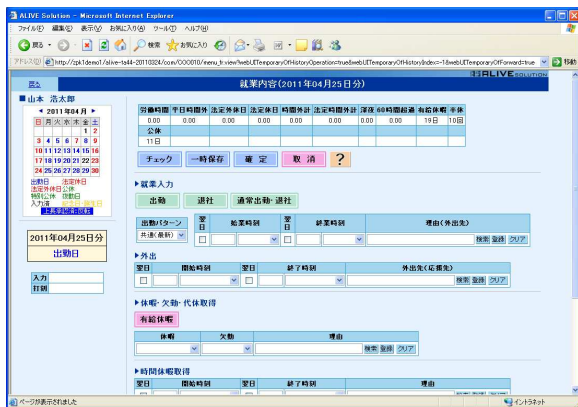


図 1. TA - 就業入力画面

- ② HR | 人事情報システム
- ③ WF | ワークフローシステム
- ④ PV | 給与明細書配信システム
- ⑤ YA | 年末調整申告システム

2.2. 指南シリーズ

指南シリーズは、経営活動をサポートする基幹系パッケージである。

指南シリーズの構成は以下の通り。

- ① 会計指南 | 会計システム
- ② 給与指南 | 給与計算システム
- ③ 販売指南 | 販売管理システム

3. ALIVE SOLUTION の SaaS 化

本章では、ALIVE SOLUTION のシステム概要、ALIVE SOLUTION

で利用するフレームワーク radish、及び SaaS 化における radish の課題と解決策について述べる。

3.1. ALIVE SOLUTION の構成

ALIVE SOLUTION は、社内イントラネットを前提とした Web システムである。社内のサーバにインストールした OS 上で ALIVE SOLUTION を稼働させ、利用者はブラウザから HTTP プロトコルで ALIVE SOLUTION にアクセスする。

図 2 に、現状のイントラネット構成を示す。

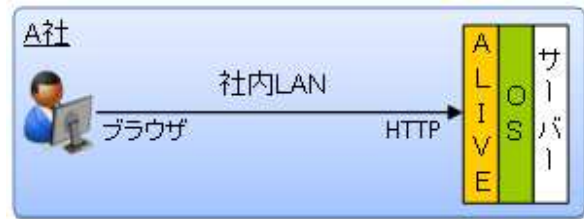


図 2. ALIVE SOLUTION のイントラネット構成

SaaS では、インターネット経由の利用が前提となる。データセンターの仮想マシンにインストールした OS の上で ALIVE SOLUTION を稼働させる。利用者はブラウザからインターネットを経由し ALIVE SOLUTION にアクセスする。ALIVE SOLUTION では人事データを取り扱うため、通信経路上での盗聴等を防止する必要があり、HTTPS で通信を行う。

図 3 に、SaaS の構成を示す。

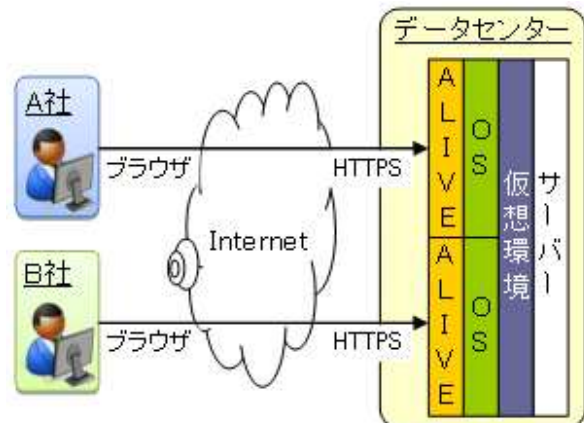


図 3. ALIVE SOLUTION の SaaS 構成

3.2. ALIVE SOLUTION システム基盤 (radish)

radish は、ALIVE SOLUTION で使用している自社開発の Java (注 1) フレームワークである。

radish の中に、Web システムのユーザインタフェースを受け持つ radish Web UI コンポーネント群があり、その中の 1 つに入力補助機能を提供する入力コンポーネントがある。HTML の Input タグのテキストボックスは入力値の型や範囲をチェックする機能を持たないが、radish Web UI の入力コンポーネントにより、JavaScript による入力値チェックやフォーマット変換の機能を付加する。

入力値のチェックは Ajax (Asynchronous JavaScript and XML: サーバとの非同期通信) にてサーバで処理を行っている。そのため、サーバ側に固有のチェック処理を記述し、テキストボックスから呼び出すことで、ビジネスロジックに関わる

チェック処理を、HTML に記述する事なく実現可能としている。

図4に、時間入力コンポーネントの例を示す。

No	翌日	時刻(始業・終業)	
1	<input type="checkbox"/>	8:53	出勤 <input type="button" value="▼"/>
2	<input type="checkbox"/>	08:53	<input type="button" value="▼"/>
3	<input type="checkbox"/>	8:61	<input type="button" value="▼"/>
4	<input type="checkbox"/>	abc	<input type="button" value="▼"/>

図4. 時間入力コンポーネント

1行目は入力途中の状態、8時53分の出勤を入力するために「8.53」と入力している。この状態でフォーカスを移動すると、2行目のように「08:53」と自動でフォーマット変換し表示する。3行目の「8.61」、4行目の「abc」のように時間表現としては不正な入力をした場合、テキストボックスの背景色を変え、入力値の不正を表現する。

(注1) Java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登録商標です。

3.3. SaaS化に向けた課題

SaaS化に向けた検証の一例として、TAをインターネット経由で利用した時と、イントラネット内で利用した時の性能比較、パケット量の調査結果を紹介する。

検証環境は、図3の構成で、仮想マシンはCPU:1Core、メモリ:4GBを割り当てた。インターネット経由の利用はMB社内から、イントラネット利用はデータセンター内の別仮想マシンからアクセスする事で、アクセス経路のみが異なる環境を構築した。

3.3.1. ネットワークトラフィックの課題

就業入力画面をインターネット経由で利用した場合、今回検証した環境では、イントラネット環境に比べ、最大2.5倍の応答時間を要した。

インターネット経由で応答時間が長くなる原因として、回線速度やサーバ性能のほか、ネットワークトラフィックが多いこともあげられる。

現行のALIVE SOLUTIONは3.2に示したとおり、コンポーネントにてHTML標準のテキストボックスに機能を付加している。機能の実装にはJavaScriptを多用しているため、HTMLが巨大化している。

表1に、テキストボックスのみを表示したサンプルHTMLに関して、radishによる変換処理前、処理後のHTML容量を示す。

テキストボックス数	処理前 (byte)	処理後 (byte)
1個	161	3,266
10個	684	9,895
20個	1,274	17,305

表1. HTML容量

テキストボックスが1個あるHTMLの容量は161byteであるが、このHTMLをradish上で実行すると、テキストボックスのほか、formタグに対する初期処理等が追加され、3,266byteとなる。

JavaScriptにて機能を付加したHTMLをクライアントに転送するネットワークトラフィック減少が、SaaS提供する際の応答性能を維持するための課題となる。

3.3.2. ユーザビリティの課題

SaaSでは、システム導入作業が現状のパッケージ販売とは異なるため、導入時の操作指導に関しても対応が必要となる。より理解しやすいマニュアルを整備するとともに、より直感的に利用可能なユーザビリティへと改善する必要がある。

3.4. 解決策

3.4.1. ネットワークトラフィック低減の解決策

SaaS提供するために、radish Web UIコンポーネントが生成するJavaScriptや機能の見直しをすることで、ネットワークトラフィックの低減を目指しており、radishの改善版を試作評価した。

現行版のradishでは、コンポーネントの振る舞いに必要なJavaScriptを各コンポーネントが個別に生成している。この動的に生成しているJavaScriptのうち、共有可能な処理を関数として1つ生成し、各コンポーネントから呼び出すことで、JavaScriptの生成量を減らしHTMLサイズを縮小した。

表2に、radishにより機能付加したHTMLに関して、クライアントに出力される容量を示す。

テキストボックス数	現行版 HTML量 (byte)	改善版 HTML量 (byte)	削減量 (byte)	削減効果 (%)
1個	3,266	4,972	-1,706	-52
10個	9,895	6,731	3,164	32
20個	17,305	8,701	8,604	50

表2. radish改善によるHTML削減効果

テキストボックスが1個の場合、改善版のHTMLは現行版の-52%と悪化している。しかし、テキストボックスが10個の場合は32%、20個の場合は50%と、テキストボックスの数が多くなるほど、共有化による効果が現れる。

実際のHTMLにはデザインに関連する記述や、表示文字列等含まれるため、一概には言えないが、項目数の少ない画面で転送量が問題になることは少なく、改善による効果は大きい。

3.4.2. ユーザビリティの解決策

具体例として、ガジェットのようなシンプル機能を用意することで、SaaS導入時における操作指導などの利用者の負担を減らすことも可能となる。TAパッケージの最新版では、就業入力ガジェットを追加している。

本機能は、Windows Vista(注2)以降のクライアントOSにて、デスクトップに表示可能なガジェットとして、出勤・退勤を1クリックで入力可能な機能である。



図 5. 就業入力ガジェット

(注 2) Windows Vista は、米国 Microsoft Corporation の米国、日本および/またはその他の国における登録商標または商標です。

4. 指南シリーズの SaaS 化

本章では、指南シリーズで採用している入力コンポーネント MBInput と SaaS 化に向けた取組みを紹介する。

4.1. 指南シリーズの構成

指南シリーズは、クライアントサーバ方式を採用しており、利用端末毎にソフトウェアをインストールする。

4.2. 指南シリーズ共通コンポーネント (MBInput)

MBInput は、業務アプリケーションにおける項目入力の標準化を目的として自社開発した、指南シリーズ共通の入力コンポーネントである。売上入力で使用される入力コンポーネントには、伝票情報を片手にテンキーのみの操作できる機能と、1 伝票を数秒で登録可能とする性能が求められる。MBInput のコンポーネントは、上記の様な入力業務で求められる機能と性能の両方を兼ね備えており、ユーザインタフェースにおいて重要な役割を担っている。

MBInput の特長を以降に示す。

- ・ 複数の入力項目の形式を 1 つのコンポーネントでサポート
- ・ 指定された型に対応した入力値制御、入力桁数・範囲指定が可能
- ・ Enter キーによるフォーカス移動
- ・ フォーカス取得時の背景色変更機能
- ・ 優れた入力応答性

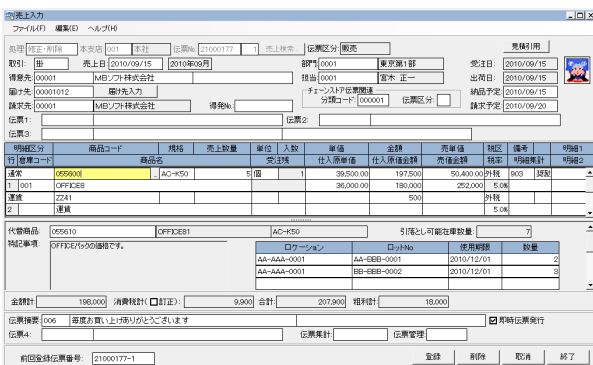


図 6. 販売指南 - 売上入力画面

図 6 に、販売指南の売上入力の画面イメージを示す。入力

項目は全て MBInput で作成している。

4.3. 課題と解決に向けた取組み

指南シリーズは、4.2 に示したとおり入力時のユーザインタフェースが重要であり、Web システムとして作り直す場合も、MBInput が持つ機能・性能の全てを実現する必要がある。このため ALIVE SOLUTION より SaaS 化の課題が多く、以下で述べるように、早期提供開始、セキュリティ、コスト、機能・性能等の要件を満たすことができる提供方式を検討している。

SaaS では、インターネット経由でソフトウェアを利用するため、セキュリティや提供コストを考慮すると Web システムの方が都合がよい。リッチクライアント方式を採用すれば機能・性能を満たせるが、対応開発のための期間とコストを要する。

機能・性能要件を満たした上で、顧客の SaaS 化ニーズに俊敏に対応するため、現時点では、既存のクライアントサーバ方式の資産を活かし SaaS 提供する方向で技術課題を抽出中である。

5. ソフトウェア以外の課題

SaaS サービスを提供する際に、ソフトウェア以外の大きな課題として以下が挙げられ、これらについても取組みを進めている。

5.1. サーバのサイジング

TA 等の業務処理が特定の時間に集中する場合に、同一の物理サーバに幾つの仮想環境を動作させることが可能か検証し、ガイドラインを設定する必要がある。

5.2. サービス環境構築の迅速化

認証認可の設定、ポート、Firewall など、仮想ネットワークの設定や仮想サーバを容易に設定できる環境やツールを準備し、サービス開始までの導入期間、コストを削減する必要がある。

6. むすび

SaaS 提供には、本稿に述べた課題のほか、アプリケーションのマルチテナント対応、マルチテナントで SaaS 提供する場合のセキュリティ課題、インターネット回線使用時の性能確保等、既存のパッケージとは異なる技術課題が存在する。今後、これらの課題に取り組み、変化の激しい市場環境において、ユーザー・ニーズに俊敏に応えられる製品、サービスを提供していく所存である。

2011 年 8 月 初版執筆 SE 本部 SE 管理部 PKG 企画

2018 年 2 月 改訂版 1 生産技術本部 技術管理課