

顧客と共に成長する流通次世代ソリューション CoreCenter[®]

CoreCenter[®] to Grow Up with Customers;
Next Generation Solution Package for Distribution Industry

加 藤 友 宏

要 約 日本ユニシス株式会社が提供する CoreCenter[®] シリーズは、これまでの流通業におけるシステム開発経験を集約し次世代に向けたソリューションパッケージである。このパッケージでは、コンセプトである「顧客の期待に沿った品質 (Q), コスト (C), 納期 (D) を実現しつつ変化 (成長) に対応できるシステム」に対して、システム構造 (SOA + コンポーネント指向アーキテクチャ) や導入方法論 (CoreMethod) の中でさまざまな考慮を施し、その要求を実現している。これらの特徴は今後さらに進むと予測される「サービス相互利用」, 「組み合わせ型」でのシステム構築においても最低限必要なシステム要件となってくる。

Abstract CoreCenter[®] series Nihon Unisys, Ltd. has to offer is a solution package for next generation to aggregate system development experience in the distribution industry so far. In this package, for the “system that can support (growth) changes while achieving quality (Q), cost (C), delivery time (D) in line with customer expectations” is a concept, And implementing various considered in system architecture (SOA + component oriented Architecture) and the introduction methodology (CoreMethod), thereby realizing the request. These features become the minimum system requirements of the future “service mutual use”, even in a system built in the “service combination type”.

1. は じ め に

日本ユニシス株式会社（以下、日本ユニシス）は2011年3月に次世代流通ソリューション「CoreCenter[®]」シリーズのリリースを発表した。このパッケージはこれまでの小売業をはじめとする流通業向けシステム開発のノウハウを結集し、新たな共通のシステム基盤上に日本ユニシスにおける流通各業態（「小売」, 「DM (DirectMarketing)」, 「リース」, 「アパレル」および「食品物流」）で導入実績のあるパッケージを再構築してシリーズ化するというものである。

CoreCenter は基盤 (CoreCenter BASE) の部分とその上に展開する小売業向けパッケージ CoreCenter for Retail をはじめとする業務機能部分で構成されており、独立した基盤部分と SOA (Service Oriented Architecture) + コンポーネント指向で編成される業務機能群により、顧客のビジネス戦略の変化にすばやく対応ができる構造になっている。

本稿では2章で CoreCenter の開発に至る背景とコンセプト、3章、4章では CoreCenter の概要とその特徴について記載し、最後に5章で今後期待する効果と展望を記述する。

2. CoreCenter 開発に至る経緯

2.1 開発の背景

CoreCenter のコンセプトを考えるに当たって、まず「お客様の声」, 「これまでのシステム構築や提案事例」, および「マーケット動向」などから業務システムに求められる要件を機能要求, 基盤要求, コスト, 期間および品質要求の体系に整理した。

図1に代表的なキーワードだけ抜粋して記述しているが, 当然のことながら, 導入コスト (C) は削減の方向でありながらより早く (D) 良いものを (Q) 求めるユーザー要求にどのようにすれば答えられるかについて議論を重ねた結果, 「QCD でユーザー要求に答えつつ変化に対応できるシステム基盤」というコンセプトに到達した。

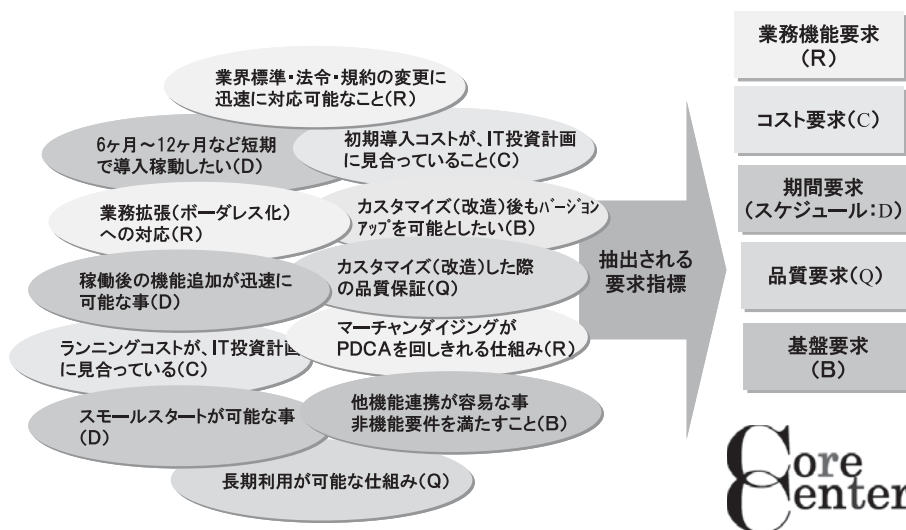


図1 流通業が基幹システムに期待する要件

ただし、早く、安くというニーズと品質良く、変化に対応できるということは相反する部分があり、このバランスをいかに取るか、すなわち作らないで拡張していく仕組みはどのように実現したらよいか最大の検討ポイントであった。

システム稼働後についてはさまざまな顧客事業の変化に対応する必要があることは言うまでもないが、最初にこの変化への対応が必要となるのは、パッケージ導入時における企業ごとの慣習、ビジネスルールの違いすなわち仕様違いへの対応である。これには以下の二つのアプローチで対応することとした。

- 1) これまでの流通業でのシステム構築経験から、企業ごとの変化 (違い) に対応可能なサービス群、部品群を前もって用意しておき、それらを組み合わせ型で提供できる構造とする。
- 2) 変化のすべては予測できないため、新たな業務 (サービス) を作成する際にはできるだけ既存の部品を再利用できる基盤を構築する。

日本ユニシスがあらかじめ用意する機能やルールの中から機能選択してシステムを組み上げていく手法により、セミオーダー的な感覚でシステム構築することができる。また、バリエー

ソリューションに存在しなくても、コンポーネント指向の徹底的な採用と全工程に渡る基盤機能による開発サポートにより、コンポーネント単位でのリユースが可能となり、開発期間の短縮、コストの削減を実現できる。この構造の実現に伴い、パッケージ導入プロセスの見直しも同時に実施し、より効率的に機能選択、組み立てができる方法論も策定することとした。詳細は5.1節で述べる。

2.2 CoreCenter の誕生

前節で述べたコンセプトを受け、この実現方法を図2のように整理した。

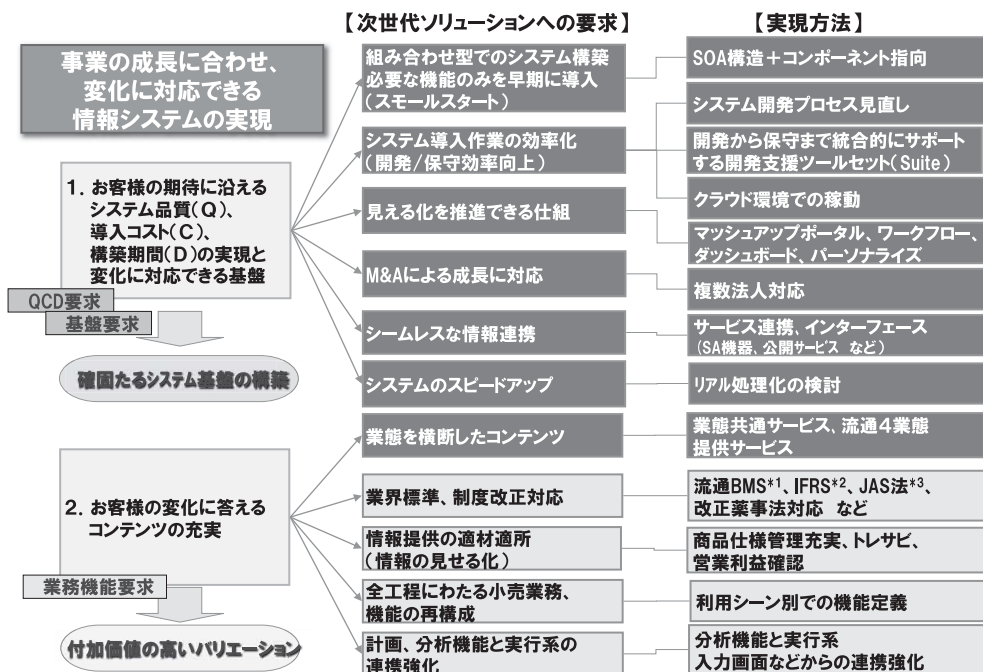


図2 パッケージへの要求に対する実現方法

図2中の次世代ソリューションへの要求と実現方法のうち、主な三つについて以下1)から3)に述べる。なお、2)と3)については、本特集号に収録した論文「次世代統一 AP 基盤 CoreCenter BASE の役割」で詳細を説明している。

1) 組み合わせ型でのシステム構築 (スモールスタート)

SOAの構造により、ユーザーにはより効果が出て本当に必要な機能範囲だけの提供が可能となった。早期導入、初期コストの抑制と段階的なシステム拡張、それによるリスク分散および検証からシステム改善までのサイクルを今までよりも短期間でまわすことが可能となり、ビジネスのめまぐるしい変化にシステムが追従しやすくなった。一般的にSOAでのシステム構築プロジェクトは、少しずつシステムを成長させていく(育てていく)形となり、一つ一つのプロジェクトが中小規模となって反復型の開発に近づいていくと言われている。

長期間のシステム構築プロジェクトでは、仕様凍結期間の増大や、上流工程で定義した要件

やニーズが本番時期には変化してしまうということが頻繁に発生しがちだが、短期サイクルのリリースであれば、ユーザー要求とのギャップについても解消される。図3は従来型の基幹システム導入におけるビジネス要求とシステム機能の時間経過とギャップ状況、図4はスモールスタートによる段階的導入でのビジネス要求に対するシステム機能の追従曲線である。

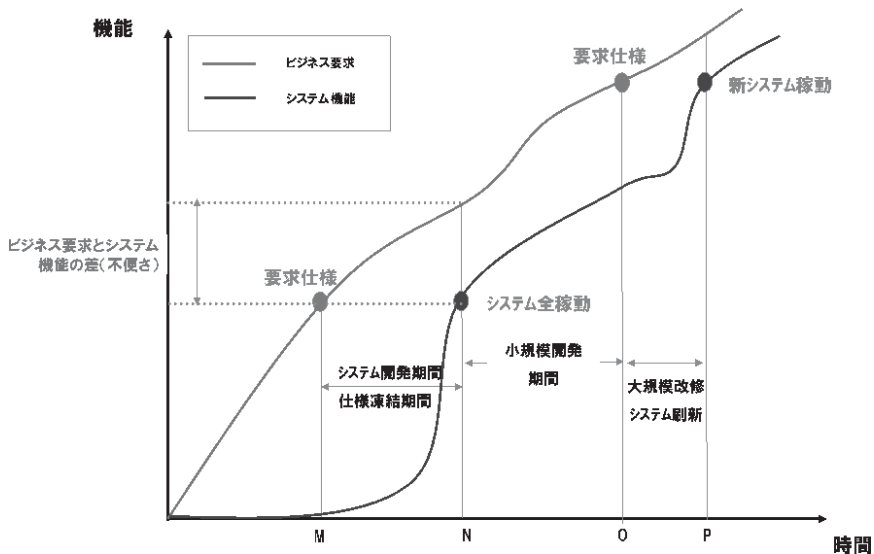


図3 従来型の基幹システム導入におけるビジネス要求とシステム機能の時間経過とギャップ

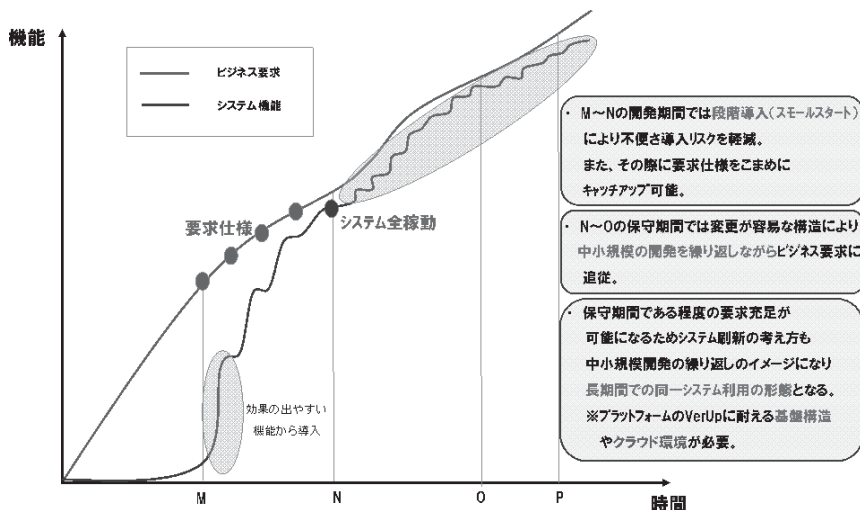


図4 スモールスタートによる段階的導入でのビジネス要求に対するシステム機能の追従曲線

2) 開発、導入作業の効率化

短期サイクルのリリースにおいては部品の開発支援はもとより、設計から保守まで統合管理できる基盤が必要になる。これらは、サービスおよびコンポーネント群の構成、仕様およびバージョンを管理するリポジトリや構成管理、およびリリース管理を包含する必要がある。CoreCenter BASEでは「Suite」というツールセットでその機能を提供している。

3) 見える化推進のための仕組み

これらの組み立て型での構造化が進むと、見える化推進のためのポータルについても各サービス、コンポーネントからのアウトプットを組み合わせでの構築が可能となる。また他システムとの連携も標準化されたインターフェースに統一できる。CoreCenter BASE ではそれぞれ「ポータル基盤」、「インターフェース基盤」としてこれを実現している。

3. CoreCenter の概要

3.1 CoreCenter とは

2章で記載したコンセプトのもとで開発、リリースされた CoreCenter は、図5のような構造となっている。基盤部分とアプリケーション部分を分離することにより、プラットフォームやミドルウェアのバージョンアップに対しては、基盤部分で変更を吸収することでアプリケーションへの影響をより小さくできるように考慮されている。さらに、この基盤部分については日本ユニシスの標準基盤「AtlasBase[®]」*4 に準拠しており、この層でもバージョンアップでの影響を抑えることができる。

また、流通各業態の業務アプリケーションが同一基盤上で展開され稼働することにより、それぞれの業態別パッケージのサービス/部品を共有できるシナジー効果が期待できる。これによりユーザー企業の変化、例えば新業態への対応についても該当する業態のサービス/部品を再利用することによって早期にシステム化対応することが可能になる。

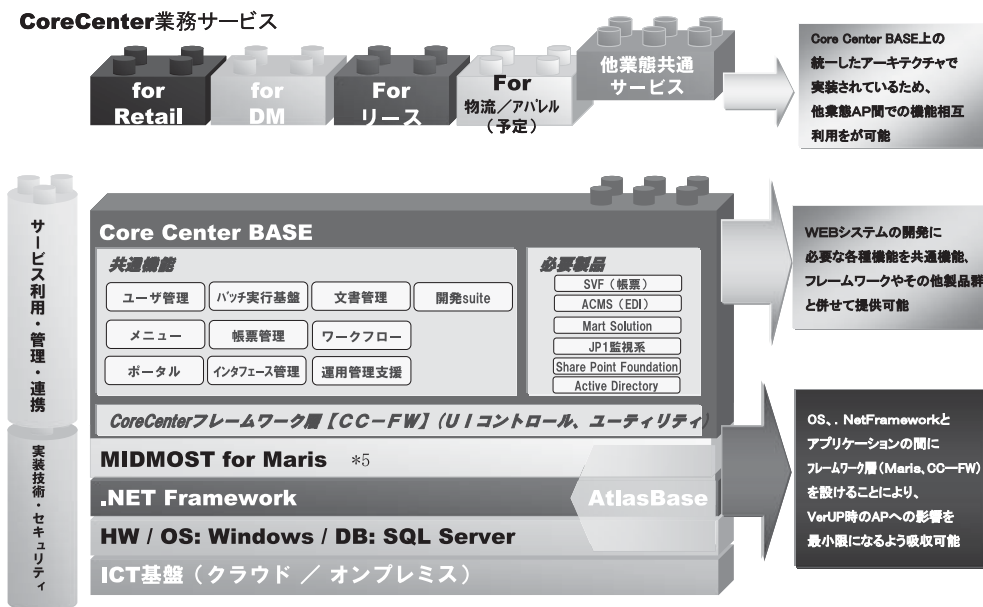


図5 CoreCenter 構成図

3.2 CoreCenter シリーズ

CoreCenter シリーズは日本ユニシスがこれまで流通業界で販売してきた各業務パッケージを新たに CoreCenter BASE 基盤上でシリーズ製品として再編したものである。以下にリリース済製品のラインナップを記述する。各製品の詳細については、本特集号に収録した同製品名

の論文を参照していただきたい。

- 1) CoreCenter BASE 【2011 年 7 月リリース】
→流通業界向けの開発から運用までをサポートする統合基盤
- 2) CoreCenter for Retail 【2012 年 2 月リリース】
→小売業向けの販売管理パッケージ
- 3) CoreCenter for DM 【2014 年 1 月リリース】
→通信販売業向けのフルフィルメントパッケージ
- 4) CoreCenter for Lease (LeaseVision[®]) 【2013 年 10 月リリース】
→リース業向けの基幹パッケージ

それぞれが各業態で多数の導入実績を持つパッケージ製品であるが、今回のシリーズ化にあたって、パッケージを跨った非機能要件、基盤要件の統一、アプリケーション共通機能（管理系業務機能）を検討し、シリーズ製品内の部品共有化を推進している。

4. CoreCenter の特徴

4.1 CoreCenter の利点

CoreCenter のコンセプトである「変化への対応」は、導入するユーザーにとって大きな利点である。これらは以下 1), 2) に整理でき、それぞれに対応した構造、方法論を有している。

- 1) ユーザー企業の成長、ビジネスの変化への対応
 - a) 業務機能の変化への対応
→ SOA + コンポーネント指向による組み合わせ型システム構築への対応①
 - b) 業態、事業の変革【新業態、新事業および M&A】への対応
→業態を跨いでのサービス相互利用（3.1 節参照）、スモールスタート（2.2 節参照）、複数法人に対応したアプリケーション
※上記はあらかじめ予測ができない事象への対応が多い
- 2) ユーザーへのパッケージ導入時の変化（違い）への対応
 - a) 違いが予測可能なものや普遍機能は可能な限り標準機能で対応
→パッケージパラメータ切替方式②、導入方法論によるサポート
 - b) アドオン時も組み合わせ型を利用でき、作り込みを最小限に抑えられる構造の確立
→上記 1). a) コンポーネント指向、b) サービス相互利用で対応
※上記はあらかじめ機能や構造を準備しておくことであるレベルまでの対応は可能

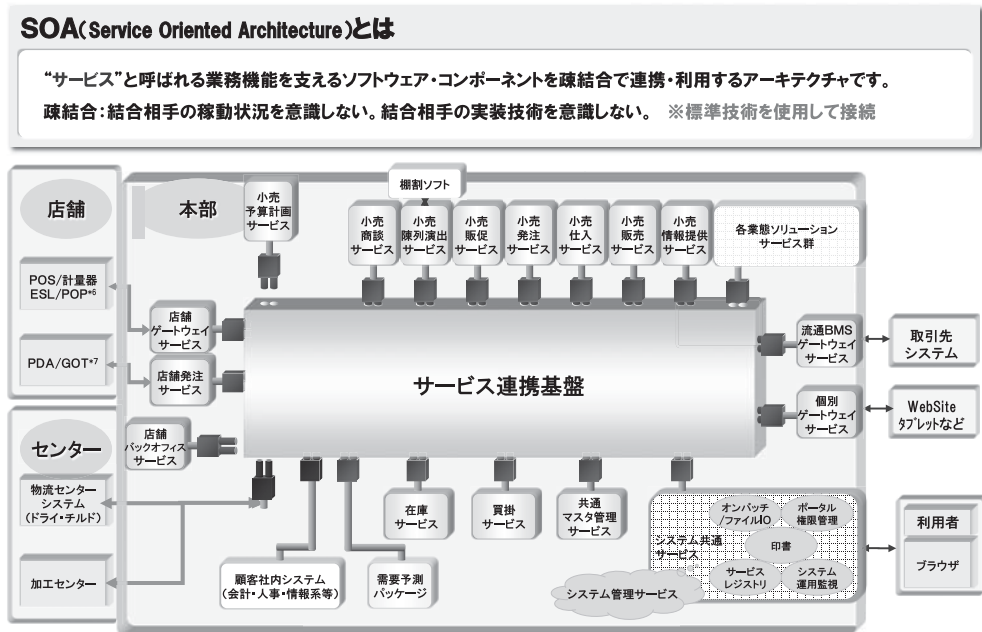
これらの中で特徴的な①「SOA + コンポーネント指向による組み合わせ型でのシステム構築への対応」及び②「パッケージパラメータ切替方式」に関して、次節でそのアーキテクチャの概要を記述する。導入方法論によるサポートについては 5.1 節の導入型ソリューションを参照のこと。

4.2 システムアーキテクチャ概要

1) SOA + コンポーネント指向

一般的に SOA での「サービス」とは「高い疎結合性（独立性）を持ちその再利用のためのインターフェースが明確であること」と定義されている。例えば CoreCenter for Retail

では図6のような単位でサービスを構成する。



図中の「予算計画」、「商談」などがCoreCenter for Retailというサービスの単位である。これらは明確なサービスインターフェースを有し、他のサービスからも独立して稼働可能な設計となっている。このサービスの単位はサブシステムとも呼び、2.2節等で述べたスモールスタートの単位にもなっている。

ただし、再利用性を考慮するとこのサービスの単位だけでは十分ではない。この単位では例えばビジネスルールが一つ異なるだけでそのサービスは再利用できなくなってしまう。そのため、サービスのように他との依存関係がなく完全に独立してはいないが、再利用が可能な部品をコンポーネントと定義し、インターフェースを明確にしてCoreCenter BASEのリポジトリに管理することとした。このことにより、それぞれの変化の度合いによって再利用の粒度を選択できるような構造となった。

(サービス>コンポーネント部品(粒度大)>コンポーネント部品(粒度中)>コンポーネント部品(粒度小)…)

2) パッケージパラメータ切替方式

CoreCenter for Retailではパッケージとして図7のような考え方で顧客の変化すなわち「カスタマイズ」、「アドオン」への対応を可能としている。

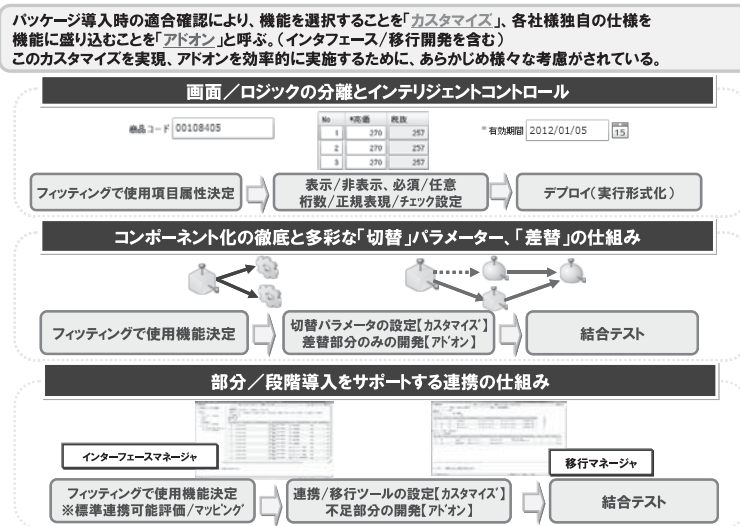


図7 CoreCenter for Retail でのカスタマイズ、アドオンの仕組み

図中に記載のある画面項目の桁数、チェックなどの変更については、画面層とビジネスロジックを完全に分離した上で、桁数などのプロパティ定義やチェックを CoreCenter BASE 機能により AP から外出し定義を可能とした。そして移行ツールについては CoreCenter for Retail 機能に、インターフェース管理/マッピングツールは CoreCenter BASE 機能にそれぞれインクルードしており、短期かつ高品質な導入を実現している。さらには、この中で切替、差替と呼ぶコンポーネントをスイッチするアーキテクチャが、4.1 節 2) で述べた「前もって想定できる範囲での変化」への対応を強力にサポートする。

CoreCenter for Retail の場合、これまでのパッケージ導入で経験した様々な機能群（例：特売種類別の発注単位採用ルール、在庫評価手法など）は、切替可能なコンポーネントのバリエーションとして前もって実装しておき、パラメータでそのコンポーネントをスイッチ（切替）する。この場合重要となるのはスイッチする粒度でコンポーネントを構成することである。CoreCenter for Retail では過去の小売業へのシステム構築経験から変更になる箇所（カスタマイズポイント）の傾向を分析しこの粒度を決定した。

一方で、これまでの経験のないロジックの開発が必要になった場合は、コンポーネントを組み上げていき、必要な箇所（ロジック）のみ差替を可能とする構造を持っている。CoreCenter ではこれらのコンポーネント間のつながりは DI（Dependency Injection：依存性の注入）という形態で CoreCenter BASE 機能として構築し実現している（図8）。これにより、インターフェースの互換性を持たせて新しく開発したコンポーネントと既存コンポーネントの差替が容易となる。再利用性の向上と作り込み部分を減らすことにより、テスト箇所も局所化でき、品質の確保につながっている。これらの構造の具体的事例については、本特集号に収録した論文「CoreCenter の採用技術」で述べている。

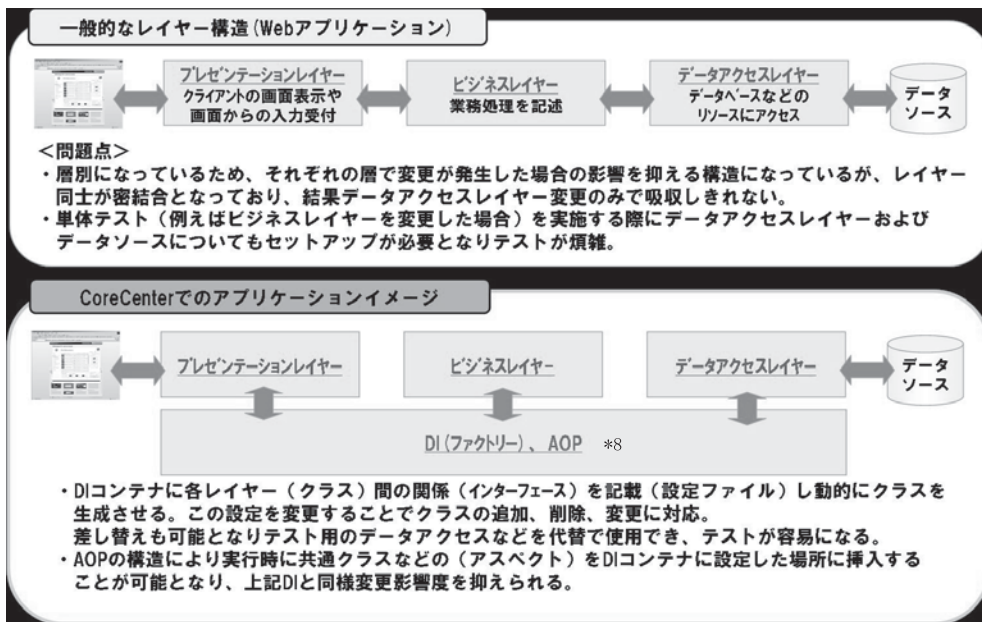


図8 CoreCenter BASEでのDI（ファクトリー）機能

5. CoreCenterの期待効果と今後の展望

5.1 導入型ソリューション

日本ユニシスの中期経営計画（2012 → 2014）の重点施策「コアビジネスの拡大」に、「ソリューションサービスの強化」という項目があり、顧客へのソリューション提供の形態を「テンプレート型^{*9}から導入型へ転換する」ことを掲げている。導入型ソリューションとは、次のような業務パッケージを指す。CoreCenterシリーズも導入型ソリューションとして開発されている。

- 1) 複数顧客を想定した汎用ソリューションで、複数顧客共通の標準部品を持つ。
- 2) 短期間のカスタマイズ・導入が可能で、そのための仕組みを持つ。
- 3) 顧客の現状業務に合わせてカスタマイズするのではなく、顧客の業務をソリューションに合わせることを原則とする
- 4) ソースコードを日本ユニシスで一元管理する。

4章に記載のとおり、CoreCenterでは上記1), 2)に対応するために部品化、再利用促進のための構造をとっている。また4)についてもCoreCenter BASEのリポジトリ、構成管理のなかでソースコードの一元管理を実施している。さらには、3)の顧客の業務をパッケージに合わせることで、他社でのベストプラクティスを共有しながら、業務改善さらには低コストでの導入を実現するという利点を享受いただくために、パッケージ導入方法論（CoreMethod）を準備している。このCoreMethodは大きく三つのフェーズ（Profiling, Fitting, Testing）に分かれている。以下にCoreCenter for Retailでの事例を記述する。

Profiling フェーズではすでに準備されているパッケージが前提としている業務フローや業務機能（画面、帳票）の説明を実施し、ビジネスルールが記載されている「業務説明書」をベースにその中に記述されている業務（機能）のバリエーションをセッション内や顧客内部での検

討により仮決め（選択）していく。

Fitting 工程では Profiling フェーズで仮決めしたパラメータを設定し顧客ごとの CoreCenter for Retail バージョン 0 をデプロイ（実行形式化）しリリースする。このときに基本的なマスタ（店舗/取引先/分類）については簡単な移行を実施し、より実地に近い環境を整える。その後は、準備されている確認シナリオに基づいて実機で「見て、触って、動かしながら」Profiling フェーズでの議論の検証とパッケージへの理解を深めていく。このフェーズについてはパラメータ設定→検証のサイクル（反復：イテレーション）を 2 回実施し、パラメータ精度を向上させる。最後に Testing で外部接続や移行データとの結合テストを実施し、稼働という流れとなる。

Profiling, Fitting 工程を通じてパッケージに対する理解度を徐々に高め、Fitting 終了時には「人に教育できるレベル」への到達を目指す。また、早い段階で実機確認が可能なため（Fitting での実機確認はプロジェクト開始から 3～5 ヶ月後）、顧客は安心感を得られ、認識齟齬を早い段階で焙り出して、後工程への影響を少なくできるとともに、パッケージの機能および前提としている業務を理解したうえでパッケージに自社の運用を合わせるという選択が可能となる（図 9）。

方法論の詳細については、本特集号の収録論文「CoreCenter の採用技術」を、また具体的な事例については「小売業 A 社への CoreCenter for Retail および CoreMethod 適用事例」を参照のこと。

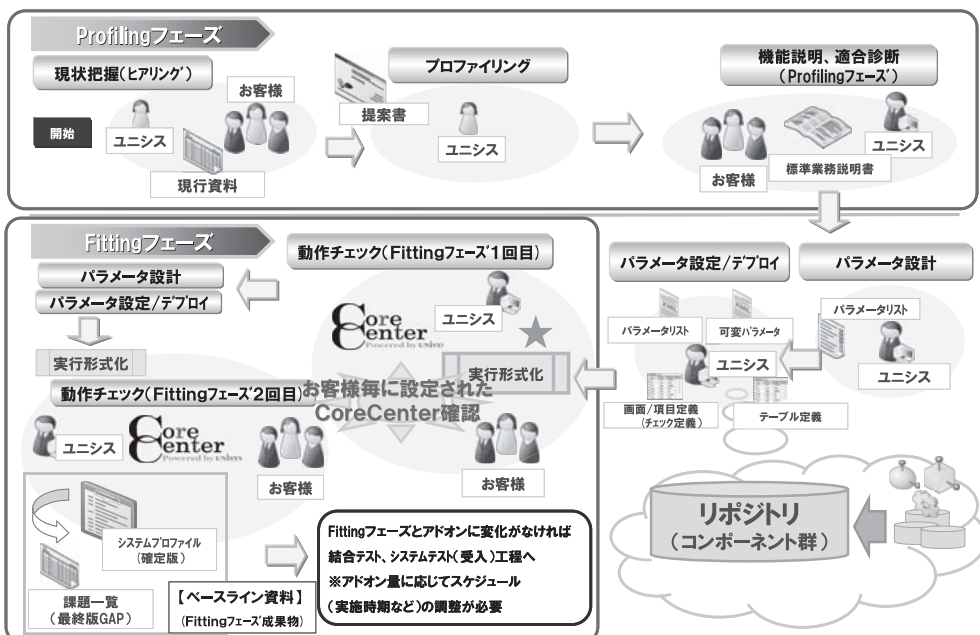


図 9 CoreMethod の流れ (例: CoreCenter for Retail)

これまで記載してきた SOA + コンポーネント指向をベースにした部品化と、その部品群を組み上げ、切替/差替を実現できるアーキテクチャ、さらには顧客の要件により体系的にアジャイル方式でパラメータを設定して部品を組み上げる方法すなわち業務そのものを組み立ててい

ける方法論により、CoreCenter ではこれまでにない短期間、低コストでの導入を実現している（CoreCenter for Retail では最短で6ヶ月モデル）。このことにより当初のコンセプトであった「顧客の期待する QCD に応えつつ変化に対応できるシステム」を実現できた。また、コンテンツ（業務機能）について、これまで日本ユニシスが培ってきた内容があらかじめ具備されており、それらが選択可能になっていることも短納期、高品質に繋がっている。

5.2 今後の展望

CoreCenter シリーズは導入型ソリューションとして開発されたことで、これまでと違う導入アプローチでのパッケージ適用が可能となった。今後は、日本ユニシスが保持するこれらの部品群（リポジトリ）を成長させながら、顧客の変化に対して「作らずに」対応できる部分を増やしていき、顧客企業内のみならず、外部を巻き込んだ形でのサービス相互利用が進むと期待される（図 10）。

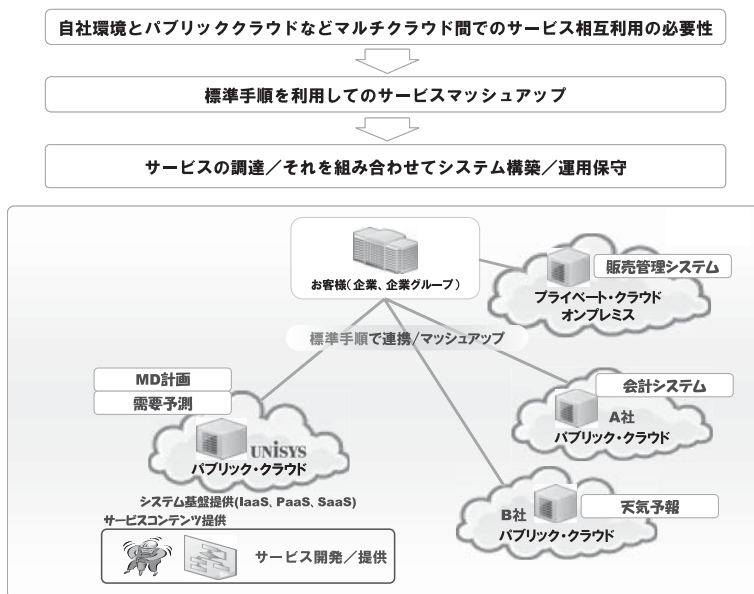


図 10 サービス相互利用図

図 10 に示すような自社環境とパブリッククラウドなどマルチクラウド間でのサービス相互利用の環境下におけるシステム基盤および基幹システムは、パブリッククラウドでの他のシステム基盤、利用端末の多様化に伴うマルチデバイスなどさまざまなチャネルからのサービス利用（インターフェース）も視野に入れる必要がある。一方で、これら CoreCenter が保有するサービスコンテンツの提供だけでなく、顧客ニーズに合わせ外部からサービスを調達し、それらを組み合わせるシステム構築を実施、さらにはその導入効果の評価、運用保守、およびその後の拡張提案などを行う「サービスコンシェルジュ」（サービスインテグレータ）としての役割が、より重要になる。

6. お わ り に

情報システムに期待される事項としては、従来の「コスト削減」(事務処理の軽減)から「利益の創出」(マーケティングツール)へと担うべき役割が変わりつつあると言われている。そのような環境の中で、日本ユニシスがこれまで構築を繰り返してきた、いわゆる「基幹システム」について今後どのように変革し、成長をさせるべきか、正しい答えを導き出すことは容易ではない。ただし、マーケティングに必要な商品、販売、在庫および顧客情報について「基幹システム」が全ての基となるデータを保持していることは間違いなく、今後の広がりにおいても CoreCenter シリーズの「変化への対応」(部品化、サービス相互利用)のコンセプトは最低限必要な要件と思われる。

日本ユニシスは今後も流通業の変化に追従していきながら CoreCenter シリーズを成長させ、顧客価値の向上に寄与していく。

-
- * 1 「流通ビジネスメッセージ標準 (Business Message Standards)」の略で、流通事業者(メーカー、卸、小売)が統一的に利用できる EDI (Electric Data Interchange) の標準仕様。
 - * 2 International Financial Reporting Standards, 国際財務報告基準。
 - * 3 農林物資の規格化及び品質表示の適正化に関する法律。この法律で定められたルールに従い、食品などには JAS マークや原産地などの表示が付いている。
 - * 4 日本ユニシスが提供するシステム基盤の総称。インフラ環境や開発支援フレームワークおよびツール群を指す。
 - * 5 日本ユニシスが提供する .NET アプリケーション開発標準。
 - * 6 Point Of Sale/Electronic Shelf Label (電子棚札)/Point of Purchase
 - * 7 Personal Digital Assistant/Graphical Order Terminal
 - * 8 Aspect Oriented Programming, ソフトウェアの特定の振る舞いを「アスペクト」として分離し、モジュール化するプログラミング技法。
 - * 9 適用時にパッケージの改造を加味しての適用を前提とした導入方法

- 参考文献** [1] 情報サービス産業協会 REBOK 企画 WG, 要求工学知識体系 第1版, 近代科学社, 2011年7月
- [2] 石川零・千葉滋, アスペクト指向プログラミングと Dependency Injection の融合, 情報処理学会論文誌プログラミング (PRO), 情報処理学会, PRO029 2006年5月
- [3] 浅海智晴, 上流工程 UML モデリング, 日経 BP 社, 第一版, 2008年8月
- [4] 大場克哉他, 仕事の流れで理解する 実践! SOA モデリング, 翔泳社, 2007年4月

執筆者紹介 加藤 友宏 (Tomohiro Kato)

1993年日本ユニシス(株)入社。1995年より流通小売業のシステム開発及びパッケージ開発、顧客適用業務に従事。2009年より CoreCenter の企画、開発のプロジェクトマネージャーを担当し、現在 CoreCenter for Retail 主管及び顧客適用を担当。

