

# IBM PureSystemsの目指すところ

## — 専門家の知見を実装したシステムによるITライフサイクルの変革 —

IBMは、専門家の知見やベスト・プラクティスをシステム内に組み込んだエキスパート・インテグレートド・システムと呼ばれるまったく新しいカテゴリーの製品として IBM PureSystems (以下、PureSystems) を発表しました。この製品は今まで企画、開発、テスト、運用、保守などの各段階で多くの労力とコストが掛かっていた IT のライフサイクルを根本的に変えることを目指したシステムです。具体的には、多様なワークロードを適切に管理・運用するために、設計段階よりサービスに必要なハードウェアとソフトウェア、および管理・運用における知見を定式化したパターンを統合して、最適な IT プラットフォームを提供するものです。本稿では、この PureSystems の誕生の背景とその特長を概括し、この製品に備わる専門家の知見がどのようにその効果をもたらすのか、具体例と共に解説します。

### ① はじめに

近年、保守・運用にかかわる費用が急速に増大し、かつシステム構築においても多くの困難とリスクが高まっています。その背景には、仮想マシンの急激な増加や、ビッグデータに代表されるようなデータ量の急速な伸び、ビジネスの急激な変化への対応、そしてセキュリティーの脅威など、従来のシステムの延長では対応困難な課題が山積していることが挙げられます。

このような課題の解決に向けて、2012年4月11日(日本時間4月12日)、IBMは全世界でエキスパート・インテグレートド・システム(Expert Integrated Systems)というまったく新しいカテゴリーの製品である PureSystems を発表しました。エキスパート・インテグレートド・システムとは、専門家が培った高度な知見をシステムに実装し、今まで個別に対応していた計画から導入、運用に至る煩雑な作業を単純化することを目指したシステムを指します。PureSystemsはその最初の製品として位置付けられており、インフラストラクチャーの管理・運用を最適化する IBM PureFlex System (以下、PureFlex System) とアプリケーション・レベルの管理・運用を最適化する IBM PureApplication System (以下、PureApplication System) の2種類の製品がリリースされています。

従来、新しく導入したサーバーを使えるようにするためには、IT組織を立ち上げて、インフラストラクチャーを構築し、アプリケーションを新たに展開するなどの作業に何カ月もの期間を要していたのですが、PureSystemsではわずか数時間ほどで行えるように設計されています。PureSystemsは、汎用システムの柔軟性、クラウドの弾力性、そしてアプライアンスの単純さを兼ね備えたシステムです [1]。

本稿では、この PureSystems の誕生の背景と、その概要と主な機能を述べ、この製品の特長であるパターン化され

た知見の活用によってどのような効果もたらされるのかについて、幾つかの例を交えながら解説します。

### ② IT環境の変化と課題

本章では、今日の IT 環境にまつわる変化と課題に関して、インフラストラクチャーとアプリケーションの両面で眺めます。

#### 2.1 インフラストラクチャーにおける課題

IDCが行った市場調査の報告 [2] によると、ITの支出の中で、保守・運用にかかわる費用が急速に増大しています。その予測では2013年には全支出の中で保守・運用費用の占める割合が70%を遥かに超える勢いです。この背景には、2000年から2010年間の10年間にサーバー数が6倍に、ストレージ容量が約70倍に増大し、仮想マシンの数が年率50%以上で増加しているという劇的な変化に伴って、ITインフラストラクチャーの維持・管理が非常に複雑になってきている状況がうかがえます。

1990年代、オープン・システムの台頭とともに、サーバー、ストレージ、ネットワーク、OS、ミドルウェアなどのさまざまなコンポーネントを調達して、最適な IT インフラストラクチャーを構築するアプローチが広まりました。構築直後の IT インフラストラクチャーは、当初の目標を達成するために最適な環境を実現してきましたが、時間とともに多様な変化に追従するための更新・変更が繰り返され、その保守・運用だけでも非常に大変な作業になってきました。近年高まるセキュリティー脅威への対応もその1つといえます。全体のセキュリティー・ポリシーを策定するだけではなく、ITインフラストラクチャーを構成するすべてのコンポーネントに対しても、適切なセキュリティー対応が求められています。加えて、仮想マシンの増大も大きな課題であり、サーバー集約・統合により物

理サーバーの数は減少したのに対し、その上で稼働するワークロードの数は増大し続けており、データを含めて管理の負担が急速に増加しています。この仮想マシンの増大に伴って、データセンターにおけるネットワークのトラフィックも劇的に変化しています。従来は、コア・ネットワークとデータセンター内のサーバーとのトラフィックがほとんどでしたが、仮想化が普及するに伴ってデータセンター内のサーバー間のトラフィックが急増しています。これは、仮想マシンからほかのサーバーやストレージにアクセスし、データの移動が増えることに起因しています。このような例はデータセンター内の基本構成にも大きくかわることで、IT インフラストラクチャーやサービスのライフサイクルをいかに適切に管理するかが大きな課題になります。さらに、これらの複雑なインフラストラクチャーを適切に維持・管理するために、高度なスキルを持った多くの要員が必要となり、人的リソースの維持・確保においても多くの課題が表面化してきています。

このような状況を踏まえ、Bathwick Group の Gary Barnett 氏はホワイト・ペーパー「The Road to Smarter Computing」の中で次のように述べています。「時を経て、IT システムは複雑になり管理しにくいものになってしまいました。その結果、本来であればイノベーションを促進すべき IT が、イノベーションの妨げとなっています」[3]

## 2.2 アプリケーションにおける課題

今日の企業は、競争力を維持するために、製品やサービスを市場に送り出すまでの時間、コスト、リスクを今まで以上に減らすことが求められています。言い換えればアプリケーションの開発、テスト、サービスの本番稼働、保守という一連の流れを効率的に実現する仕組みが求められているのです。

しかしながら、現実には従来の開発プロセスにおいて多くの課題があります。例えば、アプリケーションの開発環境を構築する場合、開発プロジェクトごとにサーバー、ネットワーク、OS、ミドルウェアなど必要なコンポーネントを集めシステムを構築すると、多大の時間とコストがかかるだけでなく、テストの局面や本番稼働局面でも同様の構築・管理作業が発生し、各環境の保守、管理のコストが増大します。また、並行して複数のプロジェクトが進む今日において、各プロジェクトの進捗や特性により、各環境を維持することの非効率さも大きな課題になります。

さらには、各アプリケーションの稼働環境もサービス・レベルの定義によって大きく変化し、最大容量の稼働環境をあらかじめ固定的に確保するのは得策ではありません。昨今クラウド・コンピューティングを前提とした開発方法が広まりつつありますが、その流れの中で開発・テスト・運用の各サイクルを融合することにより、アプリケーションの開発効率の向上

と、アプリケーションのデプロイ（展開）のスピードを高め、保守の生産性を高める DevOps [4] が注目されているのは当然の流れです。しかしながら、現実にはその期待に対して、実際のシステムが十分対応できていませんでした。また、その環境を実現するにも多岐にわたる高度の専門的知識とスキルが必要でした。今まで、労働集約的に開発、テスト、保守が行われてきたアプローチから、さらにビジネス・イノベーションにつながるアプローチへの変革が求められています。

## ③ PureSystemsの概要と特長

### 3.1 設計指針

前章では、プロジェクトごとに独立して開発、テスト、本番稼働環境を構築するサイロ型のアプローチの非効率さと課題を述べました。IT のライフサイクルにおける課題を根本的に変えるため、IBM は PureSystems の実現に当たり、以下の設計指針を掲げて取り組みました。

- 多様なワークロードを柔軟にかつ効率的に処理、管理可能にするため、仮想化を徹底し、システム全体の効率とスケラビリティ、および変化に対する俊敏性（Agility）を高める。
- 最新のテクノロジーの採用とともに将来のテクノロジーの進化にも柔軟に対応可能にし、ビジネスの成長が可能なプラットフォームを実現する。
- サービスの基盤となるインフラストラクチャーに求められる信頼性・可用性を実現し、かつ高いセキュリティを実現する。
- インフラストラクチャーとアプリケーションの両面で、シンプルかつ統一的な管理を提供し、運用・管理における煩雑な作業を単純化する。

このような指針の下、IBM はインフラストラクチャーの基盤 (IaaS: Infrastructure as a Services) として PureFlex System を、アプリケーションの基盤 (PaaS: Platform as a Services) として PureApplication System を開発しました。この両者は、共通のコンポーネントとして Flex System と呼ばれる、シャース、X86/POWER プロセッサを含むコンピュータ・ノード、HDD および SSD を内蔵するストレージ・ノード、ネットワーキング・ノード、そして統合システム管理アプライアンスなどのモジュールから成り立ちます。PureFlex System と PureApplication System はエキスパート・インテグレーション・システムとして、あらかじめ必要なモジュールを最適に構成した状態で出荷されるのに対して、IBM Flex System (以下、Flex System) はお客様の環境に合わせてシステム構築を行うために、各モジュール単位で提供されます [5]。そ

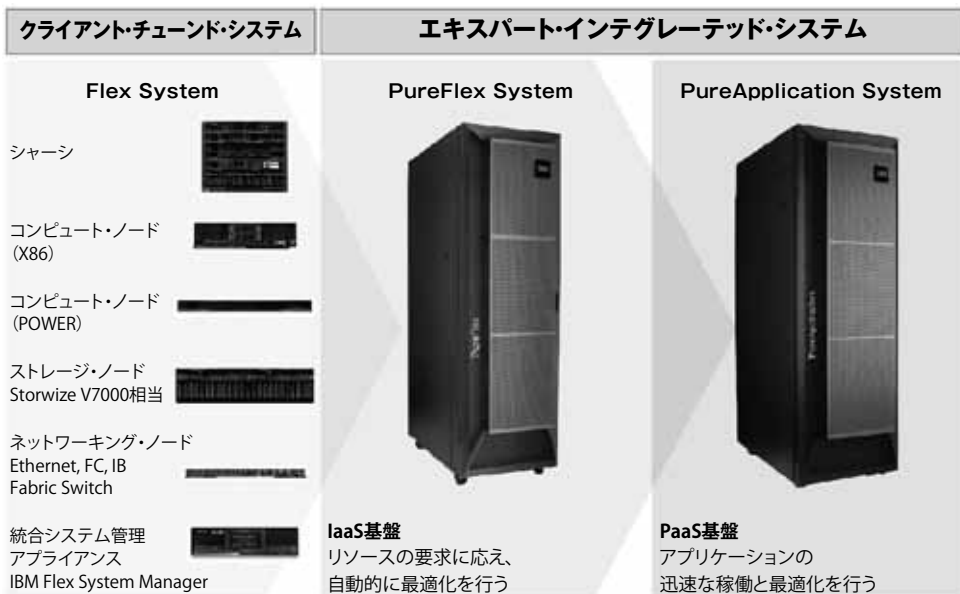


図1. PureSystemsファミリーの製品群

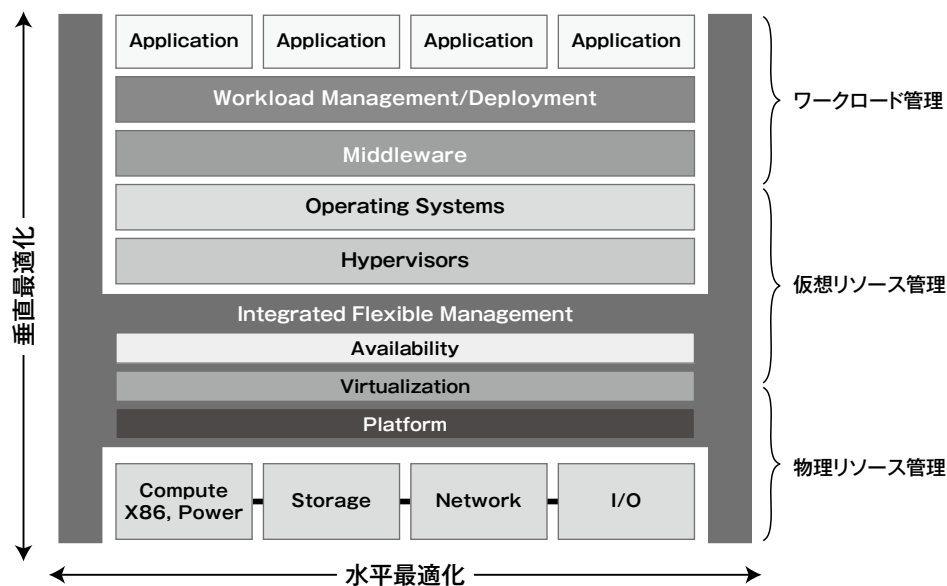


図2. ワークロードを主体に据えたPureSystemsのアーキテクチャー・フレームワーク

の関係を図1に示しました。

### 3.2 基本アーキテクチャー

PureSystemsでは、ワークロードを中心に据えて、ワークロードの管理、仮想リソースの管理、そして物理リソースの管理の3階層を統合する管理フレームワークを導入しました。

最適なITインフラストラクチャーあるいはアプリケーション稼働環境を迅速に提供するために、設計段階からプロセッサ、ネットワーク、ストレージなどのハードウェア・モジュールとハイパーバイザー、OS、ミドルウェア、管理ツールなどを完全に統合しています。また、各モジュール間のオーバーヘッド

を減らす工夫などにより、処理速度の向上を図っています。すなわちシステム・スタック全体として、垂直方向の最適化を行っています。

その一方、多様なワークロードに柔軟に対応するため、適切なアーキテクチャーの選択を可能にする水平方向の最適化も行えるようにしました。例えば、複数のプロセッサ・アーキテクチャーやハイパーバイザー、OSを1つのシステム内に共存させて、統合的に管理することで、リソースの利用効率を高め、管理のための労力を削減できます。

このように、水平と垂直の2つの最適化が可能なアーキテクチャーを採用することにより、コンポーネント・レベルからシステム・レベルまで、ワークロードの特性に合わせ、必要なリソースでの最適な処理が可能になり、かつ統合された管理フレームワークによる統一的な管理・運用を実現するシステムになりました(図2)。

### 3.3 PureFlex System

PureFlex Systemは、プライベート・クラウド(IaaS)の機能をシステムに統合した

もので、仮想マシン実行環境として最適化されています。VMware、KVM、Hyper-V、PowerVMという多様なハイパーバイザー環境を最適に管理するとともに、シンプルな操作で仮想稼働環境の導入、構築、運用が可能になります。これは、サーバー、ストレージ、ネットワークを設計段階で統合すること、さらにその管理、運用に関して専門家の知見とベスト・プラクティスを適用することで実現されます。従来システム構成変更や仮想環境のデプロイを行うに当たり、広範かつ専門的な知識と経験を基に、複雑な操作を人手で行っていましたが、このPureFlex Systemの場合、管理コンソールのメニューから目的の操作を選択・実行することにより容易

に仮想環境が提供可能になります。また、複数の異なるアーキテクチャーの仮想環境を同一システム内で共存・管理可能なため、スムーズなアプリケーション移行作業が可能になります。さらに、システム内サーバー障害予兆検知を起点とした仮想マシン退避やファームウェアの更新対応機能もシステム内に備わっているため、運用面でも可用性を高めます。

これらの機能は PureFlex System に搭載されたシステム管理モジュールである IBM Flex System Manager (以下、FSM) がシャーシ内の各モジュールと密接に連携することにより実現されています。FSM はハードウェアの管理のみならず、ドライバーの更新やハイパーバイザーの仮想化機能までを、系統的に透過的に運用・管理することを可能にします。特に FSM は複数アーキテクチャーのハイパーバイザーを統一的に管理可能であるため、今までハイパーバイザーごとに管理が異なっていた複雑さを解消しています。

### 3.4 PureApplication System

一方、PureApplication System は、アプリケーション実行環境として最適化されており、プライベートクラウド (PaaS) の機能をシステムに統合しています。PureApplication System は、前述の PureFlex System をベースにして、構成済みの OS や IBM WebSphere、IBM DB2、IBM Workload Deployer などのミドルウェア環境を実装することにより、アプリケーション実行環境を提供します。IBM によってこれらの多くのコンポーネントがあらかじめ最適に統合されているため、ユーザーはアプリケーションの開発、テスト、運用に専念することができます。この効率を高めるために、

PureApplication System にはさまざまな用途に最適化されたアプリケーション・パターンが用意されています。これらのパターンから利用目的に応じて必要なものを選択することにより、アプリケーションの構成や運用要件を容易に定めることができ、アプリケーションのライフサイクルの適切な管理が可能になります。

例えば、Web アプリケーションの導入には通常 18 以上の手順が必要となりますが、PureApplication System に備わる Web アプリケーション・パターンを利用することにより、1 つの手順だけで導入可能になります。また、データベース・パターンを使用すると、トランザクション (On Line Transaction Processing: OLTP)、またはデータマート・ワークロードに対応して構成・最適化されたプライベート・クラウドに、新しいデータベースを数分間で導入できます。さらに、PureApplication System にはインフラストラクチャーとアプリケーションの両者を束ねて統一的な監視・管理を可能とするビューが備わっているため、運用がシンプルになります。

### 3.5 専門家の知見をパターンとして展開

PureSystems では、専門家の知見とベスト・プラクティスを展開するために、必要な定義情報を「パターン」として定式化しています。このパターンの採用により、迅速にビジネスの価値へつなげ、システム構築のリスクとコストを低減し、前提となるスキル要求レベルを下げる事が可能になります。このパターンには、プロセッサ、ストレージ、ネットワーク、仮想化などに関する知見を対象としたインフラストラクチャー・パターン、インフラストラクチャーとミドルウェア構成などの知見を

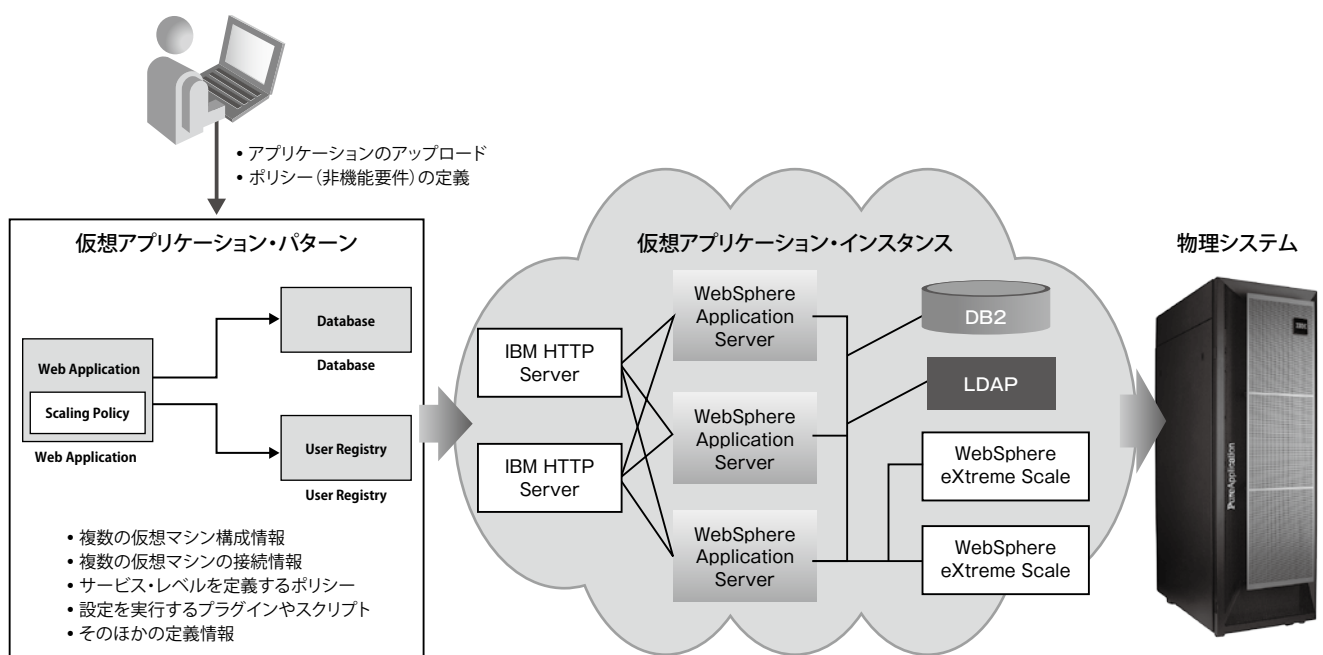


図3. アプリケーション・パターンのデプロイ

対象としたプラットフォーム・パターン、そしてビジネス・アプリケーションの振る舞いに関する知見を対象としたアプリケーション・パターンの3種類が備わっています。

例えば、ある業務を実行するアプリケーション・サーバーを実現することを考えてみましょう。従来であれば、業務要件を分析し、それを基に稼働環境を実現するために、データベースやクラスター構成、Webサーバーのキャッシュの設定など、多くのパラメーターを詳細に設定・調整するために、幅広い専門知識と経験を必要としていました。PureApplication Systemの場合には、IBM Virtual Application Builderと呼ばれるツールを介して、定義されたパターンに従い、必要なリソースやコンポーネントをメニューから選択し、応答時間などの運用要件としてのスケーリングの情報を入力することで、アプリケーションの導入展開の手順が定められ、稼働環境の実現が可能になります。この際には物理的なシステム構成やミドルウェアのパラメーターの詳細な設定などを気にせず目的が達成可能となります。これはPureApplication Systemがパターンとして記述された定義情報を解釈し、非機能要件のポリシーに従い、システムのさまざまな設定を自律的に調整することで実現しています。図3にはWebアプリケーションを例にした仮想アプリケーション・パターンのデプロイ（展開）の流れを示しました。

この例から分かるように、パターンをシステム内に実装することで、サービス立ち上げの俊敏性（Agility）の向上やシステム構築と運用の人員の削減、スキル要件の単純化、そしてリスクや問題発生<sup>しきいち</sup>の低減をもたらすことが可能です。

## 4 IBM PureSystemsにおける専門家の知見の活用

この章では、IBM PureSystemsに備わるパターンの機能を用いて、どのような効果が発揮できるかについて、具体例を交えながら解説します。

### 4.1 サービス・レベルを維持する自律運用

システムの負荷が高くなると、アプリケーションの応答時間が長くなり、その結果サービス・レベルの低下を招きます。この問題に対処するために、多くの場合システム状態を常に監視し、システム・リソースの配分と構成の変更を手作業で行ってききました。

PureApplication Systemに備わるアプリケーション・パターンには、サービス・レベルを規定するためのポリシーとして「スケーリング」などの設定メニューが備わり、CPUやリソース追加・縮退の閾値を選択することができます。システムは常にインフラストラクチャーとアプリケーションの状態を総合的に監視し、過負荷を検知した場合、これらのパラメーターに応じて自動的に仮想マシンを追加し、人手を介さずに所定のサービス・レベルを維持することができます。このような自律的な監視・応答の仕組みは、システムに障害が発生した場合にも当てはまります。万が一アプリケーション・サーバーのノードに障害が起きた場合、その障害を検知したシステムが、ポリシーに従って自動的に仮想マシンを追加し、サービス・レベルを維持します。

このような自動的にサービス・レベルを維持する運用は、ハードウェアとOS、ミドルウェア、アプリケーションが一体となり、

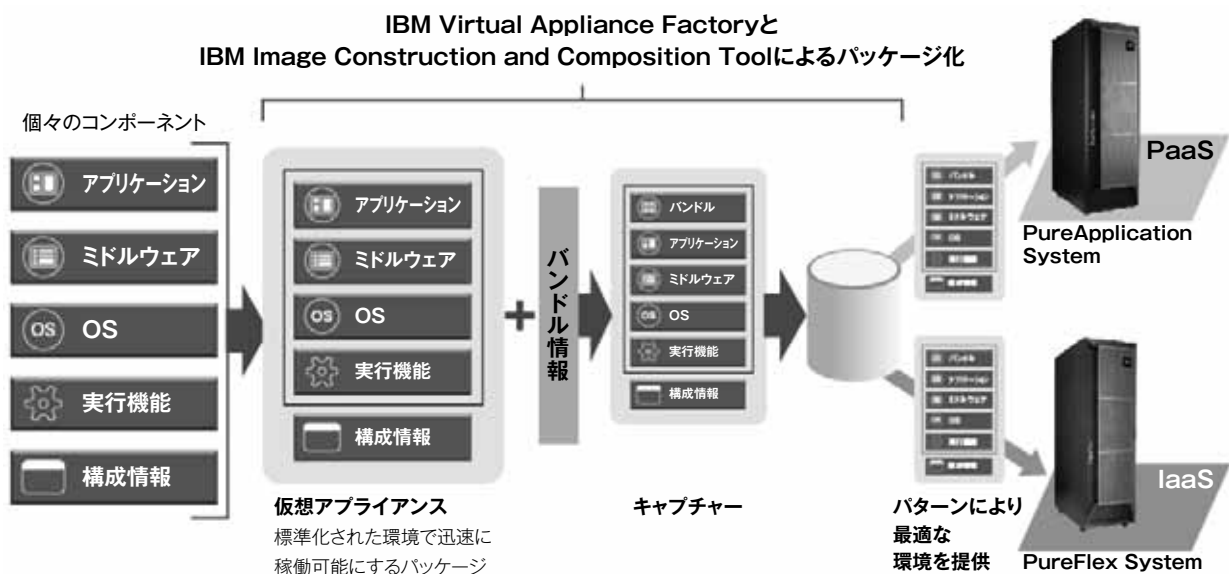


図4. 仮想アプライアンスのクラウド環境への展開

整合性を持ったシステムであるが故に実現できるものです。この結果、サービスイン後の人的作業とリスクを低減し、保守・管理コストを抑制する効果をもたらします。

#### 4.2 仮想アプライアンスのデリバリー・モデル

アプリケーション・パターンは、ビジネス・アプリケーションの専門知識と知見をパターン化したもので、ソフトウェア・ベンダーやお客様が独自のアプリケーションを提供することをサポートします。その1つの例に、アプリケーションの流通、展開を迅速かつ容易に実現する「仮想アプライアンス」があります。これは、OS、ミドルウェア、複数のアプリケーションを1つのパッケージに集約したソフトウェア・スタックで、すべてのコンポーネントを自動的にインストールし、構成することが可能になります。このパターンにより、従来のデプロイメント手法に比べ、より短期間でのクラウドへの移行を実現するとともに、自動化された操作により、高度な専門スキルを必要とせずに対応可能になります。PureFlex Systems では業界標準の DMTF (Distributed Management Task Force) の OVF (Open Virtualization Format) 仕様 [6] に準拠した仮想アプライアンスをサポートするため、この規格をサポートするほかの環境とのインターオペラビリティを持つことを可能としています。すなわち、クラウド環境に対応したアプライアンスを実現するのです。この仮想アプライアンスのアプローチは、アプリケーションの市場機会の拡大ももたらし、アプリケーションの価値を高めることができます。

IBM は、この仮想アプライアンス作成を支援するツールを提供しており、IBM Virtual Appliance Factory (VAF) および IBM Image Construction and Composition Tool (ICCT) はその代表的なものです。これらは KVM 環境や PowerVM 環境にデプロイするアプリケーションソリューションのパッケージ化に有効です。また、パッケージとしてデプロイするだけでなく、システム内に展開されたイメージを取得して、リポジトリに集め、再利用を可能にしています。こ

れにより、迅速かつ効率的なサービスの立ち上げが可能になります。図 4 には個々のコンポーネントから仮想アプライアンス・パッケージを生成し、ライセンス・キーやデータベース接続情報など、パッケージを利用可能にするバンドル情報を添え、IaaS もしくは PaaS 環境に展開される一連の流れを示しました。この仮想アプライアンスの詳細に関しては、参考文献 [7] を参照してください。

#### 4.3 パターンを基にしたライフサイクル管理

今まではインフラストラクチャーの構築、アプリケーションの開発・テスト、サービスの運用・保守の各段階において、必要なスキルとリソースを集め、人手を中心とした作業を行ってきました。生産性向上のため、開発やテストにおいて多様なツールが用いられるようになりましたが、その効果を上げるためには多大な労力と時間が必要でした。特に開発環境とテスト環境、そして本番のサービス運用環境が個別に構築・管理されてきたために、インストールや構成変更作業などには人的作業が多く、その過程でエラー発生の可能性もあり、プロジェクト全体で高いリスクを抱えていました。

仮想化技術の進歩によりハードウェアとソフトウェアのコストとリソースの利用効率を高めることは可能になりましたが、開発作業や運用・保守の段階では依然として人手に頼ることが多く、人的コストとリスクの低減は十分に実現できていないのが実態です。

先に述べたように、PureSystems ではワークロード最適化を主目的として、パターンを単位にインフラストラクチャー

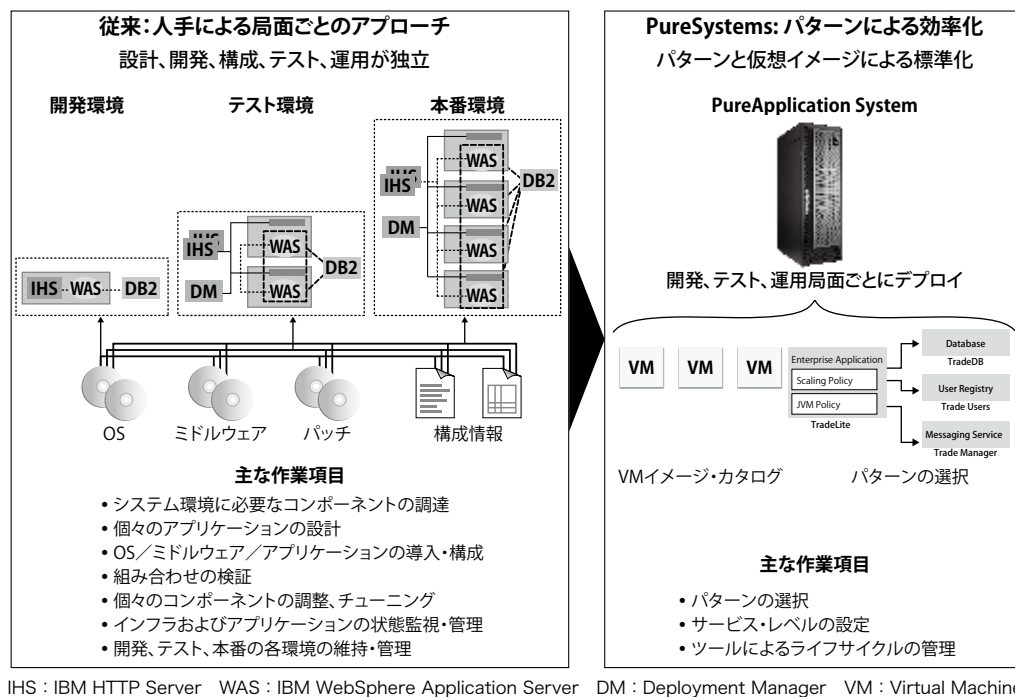


図5. PureSystemsによる開発・運用アプローチの変化

とアプリケーション基盤の管理・運用を自動的に行う機能が備わっています。特にアプリケーション環境の構成（トポロジー）もパターンとして持ち、そのパターンを基に環境を自動的に構築できるため、アプリケーション展開作業の各局面に合わせて必要な環境を迅速に提供することが可能になります。具体的には、サービスの開始時において、使用する仮想アプリケーション（および仮想システム）のパターンを指定し、仮想アプリケーションをデプロイします。運用・管理段階においては、デプロイしたシステムを監視し、規定したサービス・レベルに合わせて自動的にスケーリングを行います。さらに、アプリケーションの自動的な更新作業も可能です。そして、サービス終了とともに、使用していたITリソースを解放しリソース・プールに戻し、再利用可能にします。PureSystemsでは、ハードウェアとソフトウェアを統合して、これらの一連のステップを、整合性を保ちながら実行できるため、ユーザーはインフラストラクチャーの詳細を気にせず、目的のアプリケーションに注力することができます。このように人手で行っていた作業がPureSystemsによってどのように変わるのかを図5に示しました。

英国のInnovation Group社の試算によると、同社のソフトウェアのデプロイに要する時間が従来は13時間であったのに対して、PureApplication Systemの機能を用いることにより30分程度に収まり、96%もの短縮を実現しています[8]。このような短縮化は複数のアプリケーションを運用している場合には、より大きな効果をもたらす、TCO削減につながります。

## 5 おわりに

本稿では、「エキスパート・インテグレートド・システム」と呼ばれるまったく新しいカテゴリーの製品であるPureSystemsがどのような背景で開発され、どのような機能と特長を持ち、専門家の知見を実装したシステムがどのような課題に対して有効なのかを述べました。

PureSystemsは、汎用システムの柔軟性とクラウドの弾力性、そしてアプライアンスの単純さと迅速さを兼ね備えたシステムです。その設計段階より専門家の知見を十分に活用する機能を搭載することで、ITインフラの構築・運用の効率化を図るとともに、アプリケーションの開発、デプロイメント、管理、さらにはライフサイクル全体に大きな変革をもたらします。

PureSystemsを利用することで、システムとアプリケーション・ワークロードを統合し、ITインフラストラクチャーのTCOの削減が可能になります。それは、開発環境、テスト環境、および本番環境を共通のインフラストラクチャー上で実行し、共通の方法で構成できるためです。そのため、デプロイメン

トの自動化が図れ、アプリケーション自体が、本番環境においても開発環境やテスト環境とまったく同じように動作するようになります。PureSystemsが提供する、ワークロードをインテリジェントに管理・制御できる環境は、動的なスケーラビリティを持ち、サービス・レベルが満たされるように機能します。その結果、クラウド環境への移行も容易になります。

この結果として、PureSystemsは、新しいインフラストラクチャーの実装に伴うコストとリスクを減らし、さまざまな種類のプラットフォーム全体でサービス・ライフサイクルの最適化を可能にします。

### [参考文献]

- [1] IBM Corp.: PureSystems, <http://www.ibm.com/ibm/jp/puresystems/>
- [2] IDC: Three Data Centers - One Vision?, Doc# DR2010\_T2\_MB (2010).
- [3] Gary Barnett: The road to Smarter Computing, The Bathwick Group Whitepaper, Doc# BG-TL-W-00078UKEN-00 (2011).
- [4] 榎原彰: 開発と運用の融合 - DevOpsの波一, ProVIOSION, No.75, pp.26-31 (2012).
- [5] David Watts, et al.: IBM PureFlex System and IBM Flex System Products and Technology, IBM Redbook SG24-7984-00 (2012).
- [6] Distributed Management Task Force: オープン仮想化フォーマット, <http://dmf.org/jp/technology/ovf>
- [7] Jarek Miszczyk, et al.: Create Smart Virtual Appliances with the IBM Image Construction and Composition Tool, IBM Redbook SG24-8042-00 (2012).
- [8] IBM Software Group: Innovation Group boosts competitive edge with an IBM PureSystems solution, IBM Case Study, WAC12346-USEN (2012).



日本アイ・ビー・エム株式会社  
研究開発 / IBM Research & Development - Japan  
IBM ディスティンクイッシュト・エンジニア  
技術理事

佐貫 俊幸 Toshiyuki Sanuki Ph.D.

### [プロフィール]

1983年日本IBMジャパン・サイエンス・インスティテュート（現IBM東京基礎研究所）入社。プログラミング言語の研究、大規模コンテンツ配信などのデジタル・メディア・システムの構築、サーバー製品の開発などに従事。1997年米国IBMに出向。ソフトウェアならびにソリューションの技術企画を担当。2000年帰国後、Cellプロセッサと関連システムの開発およびビジネス開発を推進。2008年より次世代プラットフォームのシステムおよびテクノロジーの開発・企画に従事。現在、研究開発部門のテクニカル・ストラテジー担当。情報処理学会、IEEE Computer Societyの各会員。