SDEとオープン・クラウド
― Systems of Engagement を支えるクラウド技術 ―

2001年にITバブルがはじけ、IT産業は成熟産業の仲間入りをしました。それ以来IT産業全体の成長率は世界の平均GDP成長率を上回ったことはなく、先進国での2012年から2015年の成長率は毎年1%前後と予想されています。しかしそのような中、急成長を続けるSystems of Engagement（SoE）と呼ばれる領域があります。その主要技術である「クラウド」「アナリティクス」「モバイル」「ソーシャル」は桁違いに高い成長を続けており、とりわけクラウドは毎年27%の高成長が見込まれています。

本稿ではオープン・クラウド技術に着目し、ベンダー・ロックインのないOpen IaaS、Open PaaS、およびそれらが創り出すSoftware Defined Environment（SDE）、Cloud Operating Environment（Cloud OE）について解説します。

1. Systems of Engagement（SoE）

従来ITシステムは、定型業務を高速かつ信頼性高く行うことを得意としてきました。そのようなシステムはSystems of Record（以下SoR）と呼ばれ、銀行の勘定系システム、メーカーの生産管理システム、商社の受発注システム、パッケージ化されたERPなど、既存の多くのシステムはここに分類されます。エンド・ユーザーとシステムの間で、何を行うか、何が期待されているかについて、一定の合意があるところから始まるのがSoRです。

これに対し最近、堅牢性や信頼性よりも、柔軟性、俊敏性を重視するシステムが注目されており、Systems of Engagement（以下SoE）と呼ばれます（**図1**）。非定型業務を素早く開発したり、インターネットから波のように押し寄せる要求に瞬時に対応したり、初期投資を最小限に抑えて業務を立ち上げたり、「個客」セグメントに特化してサービスを提供したりするシステムです。SoEは市場とのさまざまなタッチ・ポイントからデータを収集し、分析し、短時間（数時間から数週間以内）でこれに対応したアプリケーションを開発してサービスとして提供することが可能です。新製品・サービス開拓や新市場開拓に直接役立つことから、マーケティング部門や事業部門が積極的に利用を進めています。

SoEで重要なIT技術の代表は、「クラウド」「アナリティクス」「モバイル」「ソーシャル」で、頭文字をとってCAMSと呼ばれています。上記の要件を満たすために必要なテクノロジーであり、逆に言えばそのようなテクノロジーの出現でSoEが可能になりました。

SoEとSoRは車の両輪の関係で、連携が重要です。SoEで獲得したビジネス（顧客や注文）を、SoRで確実かつ低コストで処理する必要があります。これをユーザー視点で考えれば、スマートフォンのユーザーがインターネット

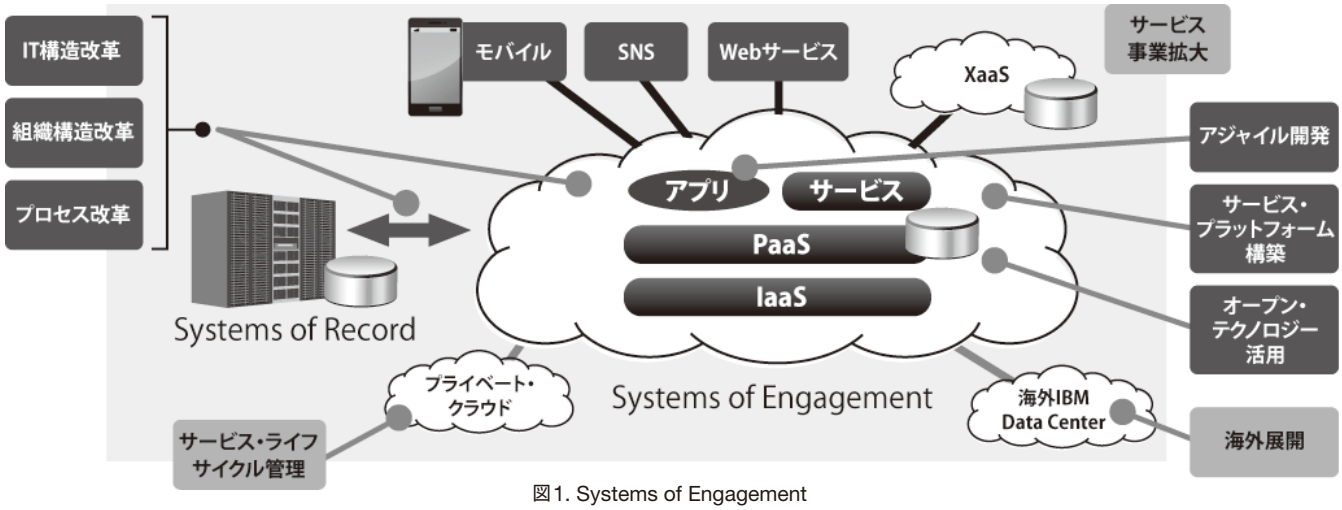

図1. Systems of Engagement

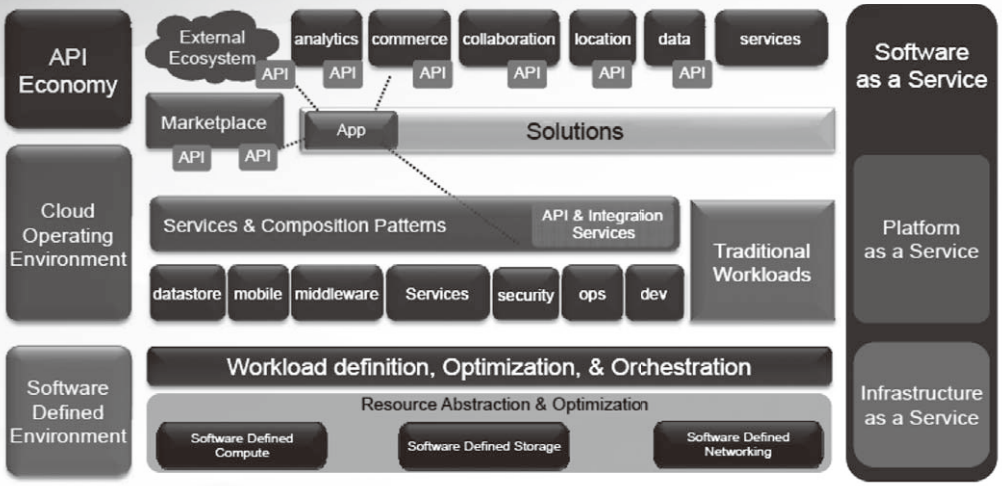

図2. IBM Open Cloud Architecture

に接続し、企業やコミュニティーからのさまざまな情報を得るとき最も重視するのは、情報の鮮度やそれが自分の今の興味にあっているか?ということでしょう。インターネットのサービスを通じてある商品を気に入り、いったん注文したとき、その商品が確かにデリバリーされ、料金が正しく決済されることが当然の期待となります。ここにSoEとSoRの分界点があり、同時に両者の連携の必要性も理解できるでしょう。クラウドの効果については、SoEではグローバルなマーケティングの広がりやビジネスの成長、SoRではシステム・コストの削減を中心に期待されています。

2. IBM オープン・クラウド・アーキテクチャー

SoEに見られるようにクラウド利用が高度化し、また一方でITシステム全般に普及が進む中、二つの課題が明らかになりました。クラウド技術の高品質、低コスト、迅速な開発と、ベンダー・ロックインのないオープン・スタンダードの必要性です。その鍵となるのが、「オープンソース・プロジェクト」と、クラウドの「オープン・アーキテクチャー」です。

オープンソース・プロジェクトは、技術者がインターネット上で協力してオープンソース・ソフトウェアを開発する、基本的にボトムアップな活動です。クラウドのような魅力的なテーマには世界中から優秀な技術者が多数参加するため、その潜在能力は大企業の研究開発能力をも凌駕すると言われています。ただしその力を最大限に引き出すには、プロジェクトのオープンな統治機構が必要です。オープンソースはソースコードが開示されるだけでは不十分で、その（実装ではなく）仕様のオープン化、設計へのプロジェクト・メンバー（コミュニティー）の参加、意思決定プロセスの公開、ライセンスに関する法的整備が必要です。さらに一般の企業が利用するには、コードのセキュリティー強化、多言語対応、保守サービスなどを有料で提供するサポート・ベンダーの存在も欠かせません。

IBMはOpenStack、Cloud Foundryなどの有望なオープンソース・プロジェクトに参加し、技術面のみならずオープン・ガバナンスの整備などを通じてプロジェクトの健全な発展に貢献しています。また成果物であるオープンソース・ソフトウェアを、自社の製品やサービスとして提供する「オープン・プラス」方式により、最先端のオープン・テクノロジーに簡単・安心を加えて企業ユーザーに提供しています。

オープン・アーキテクチャーとは、オープン・テクノロジーを統合して高い全体価値を実現するアーキテクチャーです。IBMオープン・クラウド・アーキテクチャー（**図2**）は、オープン・テクノロジーを統合して俊敏な企業活動、柔軟なクラウド・サービスを実現する、IaaS（Infrastructure as a Service）/PaaS（Platform as a Service）/SaaS（Software as a Service）を統合したクラウド全体のアーキテクチャーです。

IaaSは、物理リソースであるサーバー、ストレージ、ネットワークを上位層（PaaS）にサービスとして提供することで、ソフトウェアによる柔軟な制御と自動化が可能なSoftware Defined Environment（SDE）を提供します。OpenStack、OVF、Chef、Puppet、KVM、Open Daylight/OpenFlowなどのオープン・テクノロジーが利用されます。

PaaSは、コンポーザブルな環境、すなわち部品からの機能の組み立て、プログラムとサービスの柔軟な結合（バインド）、マルチ・プログラミング言語による開発環境（Polyglot programming）をサポートすることで、開発者の生産性を飛躍的に高めます。Cloud Foundry、OASIS TOSCA（Topology and Orchestration Specification for Cloud Applications）、OASIS OSLC（Open Services for Lifecycle Collaboration）などのオープン・テクノロジーが利用されます。TOSCAはIaaS、PaaSの境界領域のオープン・スタンダードで、

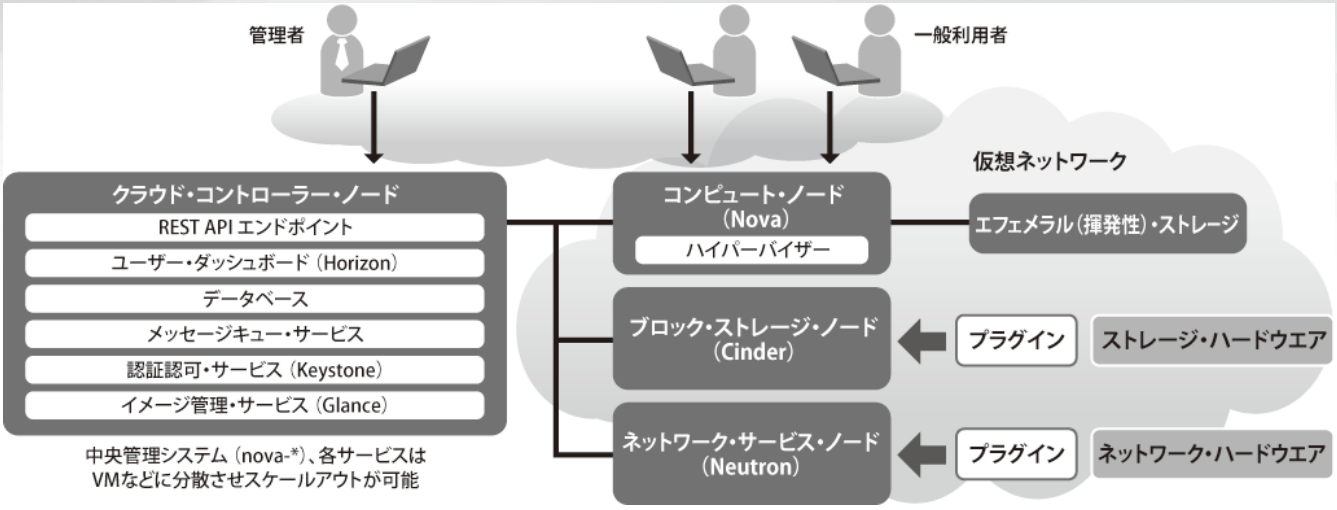

図3. OpenStack – Open IaaS

その実装はOpenStackの中でも始められています。

SaaSはソフトウェア機能をRESTful APIとして提供し、それらはPaaSからサービスとして利用（Subscribe）できるようになります。ソフトウェアは、所有しDVDからインストールする「モノ」ではなく利用するサービスとなり、ユーザーがAPIを利用するための権利を時間で買うモデルが可能になると予想されます。APIが価値の源泉になり、その利用に価格がついて市場で取り引きされることから、API Economyとも呼ばれます。OAuth 2.0、OASIS SAML、JSON/BSON、HTML5、Apache Cordova、OpenSocialなど多様なオープン・テクノロジーが利用されます。

IBMオープン・クラウド・アーキテクチャーは、開発者がビジネス価値を生み出す中心的役割を担うという発想から、プログラムの開発、実行の場であるPaaSを中心とし、それをインフラとして支えるIaaSと多様なサービスを供給するSaaSという構造になっています。オープン化によりクラウド自体の進化が促進されるだけでなく、多数のハードウェア・ベンダーの製品がOpen IaaSを通じて利用可能になり、また多数のソフトウェアおよびサービス・ベンダーの製品・サービスがAPIとして利用可能になることで、オープンな自由市場が形成され、利用者は規模の経済（Economies of Scale）、範囲の経済（Economies of Scope）の恩恵により、低コストかつ豊富な選択肢を得られるようになります。

3. OpenStackとSoftware Defined Environment（SDE）

OpenStackは、IaaSおよび関連機能を提供するオープンソース・プロジェクトです。モジュラー構造を持ち、中核機能としてNova（サーバー）、Cinder（ブロック・ストレージ）、Swift（オブジェクト・ストレージ）、Neutron（ネットワーク）、Keystone（認証）、Glance（イメージ管理）などがあります。**図3**にOpenStackによるIaaSの構成例を示します。全体の司令塔であるクラウド・コントローラー・ノードはAPIによるリクエストを受信し、内容を解釈して適当な被管理ノードにメッセージとして指示を出します。例えばLinux VM（Virtual Machine）のプロビジョニングがリクエストされたら、コンピュート・ノードにメッセージを送ります。コンピュート・ノードは管理下のハイパーバイザーを通じてVMを作成し、コントローラーはVMアクセスのためのIPアドレスなどをユーザー（管理者）に返します。この流れはIaaSとしては一般的で、ノード同士がメッセージキューを用いてコミュニケーションすることでそれぞれの独立性、分散化を図っているのが特徴です。

オープン・クラウドとしての優位性は、APIとプラグインのオープン化にあります。前者はOpenStack上でゲスト・システムを構築、運用するためのスクリプト、例えばLinuxのシェル・スクリプトやChefレシピー（Ruby DSL – Domain Specific Language）を書いた場合、OpenStackに基づくシステム間ではそれらの可搬性が高まることを意味します。企業はプライベート・クラウド、パブリック・クラウドの両方にOpenStackを採用することで統合管理が容易になるでしょう。後者では、ハードウェア・ベンダーがプラグインと呼ばれるデバイス・ドライバーのセットを自社製品用に開発することで、OpenStackで製品が使用できるようになります。この仕組みがあるため、OpenStackの支援企業にはデバイス・メーカーも多く、ユーザーは多様な製品を選択できるようになります。

OpenStackによるIaaSはさまざまな構成が可能で、**図3**はその一例です。ネットワーク・サービスにNeutronを使っていることに注目してください。ネットワーク・サービ

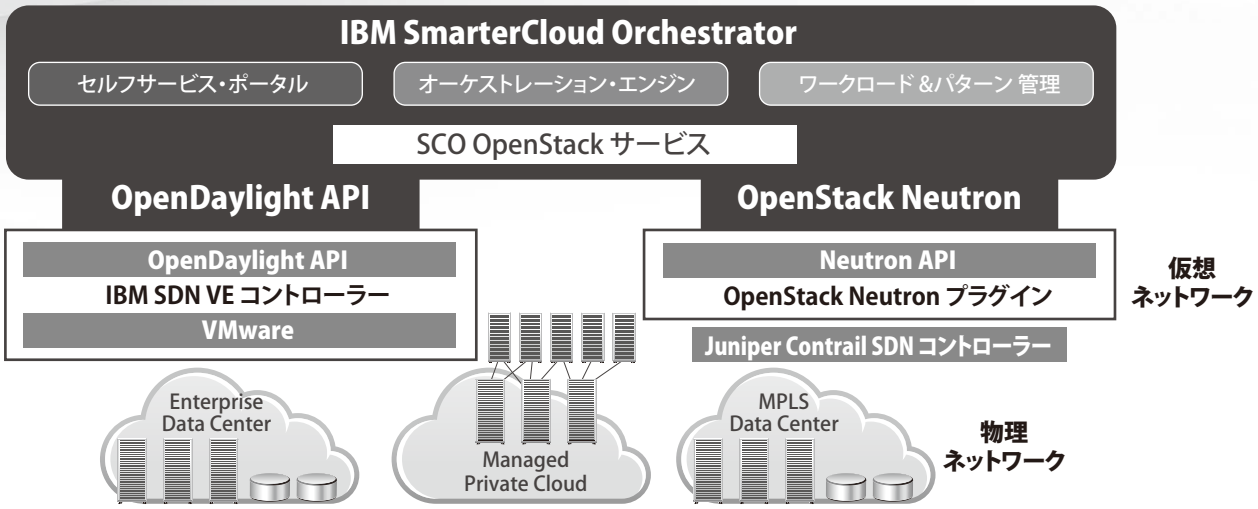

図4. Software Defined Network

スとして歴史と実績があるのはNova-network で、コンピュート・ノードのそれぞれに配置するマルチホストHA（High Availability）方式でVMのネットワーク接続を単一障害点なしにサポートしてきました。一方Neutronは L2仮想ネットワークを自由に構築でき、それとVMやファイアウォール、ロードバランサーなどのネットワーク機器を仮想ポートで接続することで、複雑な仮想ネットワークを構築可能です。クラウド上のテナントごとに多層ネットワークが容易に構築できるなど、今後の活用が期待されています。

図4は、OpenStackを実装したIBM SmarterCloud Orchestratorが、Nova管理下のVMware上のIBM SDN VE（Software Defined Network for Virtual Environment）、Neutronとそのプラグイン管理下のJuniper Contrailを通じて、Software Defined Network（SDN）をサポートしているところです。なお本稿執筆時点で最新版のSCO 2.3はNeutronをサポートしていませんので実装時期にはご注意ください。

図5はOpenStack Foundationのオープン・ガバナンスの仕組みを示しています。オープンソース・ソフトウェアとして、OpenStackのコア、ツール、プラグインなどがオープンソース・コミュニティーで開発され、年2回リリースされます。OpenStackのメジャー・リリースは、Folsom、Grizzly、Havana、Icehouseという具合にアルファベット順の頭文字の名前がつけられ、2013年秋にHavanaがリリースされています。メジャー・リリースと合わせてOpenStack Summitというカンファレンスが行われます。Havanaリリースの際は香港で開催され、世界中から3,000人以上が参加しました。OpenStackでは、各開発プロジェクトの将来リリースや新たなプロジェクトの立ち上げについてインターネットやサミットでオープンな議論を行い意思決定を行っています。これら全体のオープン・ガバナンスを支え、コミュニティー・メンバーの活動を支援しているのがOpenStack Foundationで、2012年にIBM、Rackspaceなどのリーダー企業により発足し、今日に至っています。

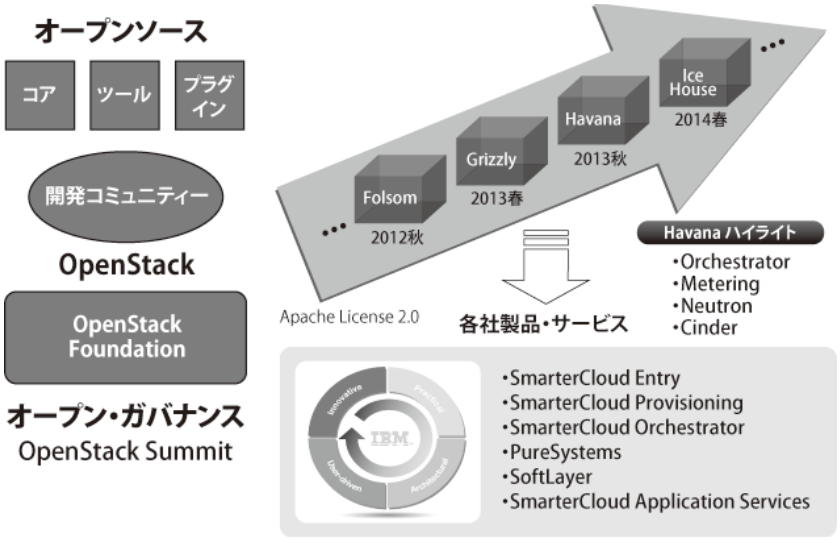

図5. OpenStack Foundation

4. OASIS TOSCAと仮想パターン

単体の仮想サーバーを構成するだけでは複雑に構成されたアプリケーションを全体で最適化することはできません。これまで基盤設計と言われていたサーバー間の接続などの処理系全体（トポロジー）と、可用性やパフォーマンス、セキュリティー管理などの非機能要件もソフトウェアで自動化される必要があります。

OASIS TOSCAは複数のノードから成るトポロジーお

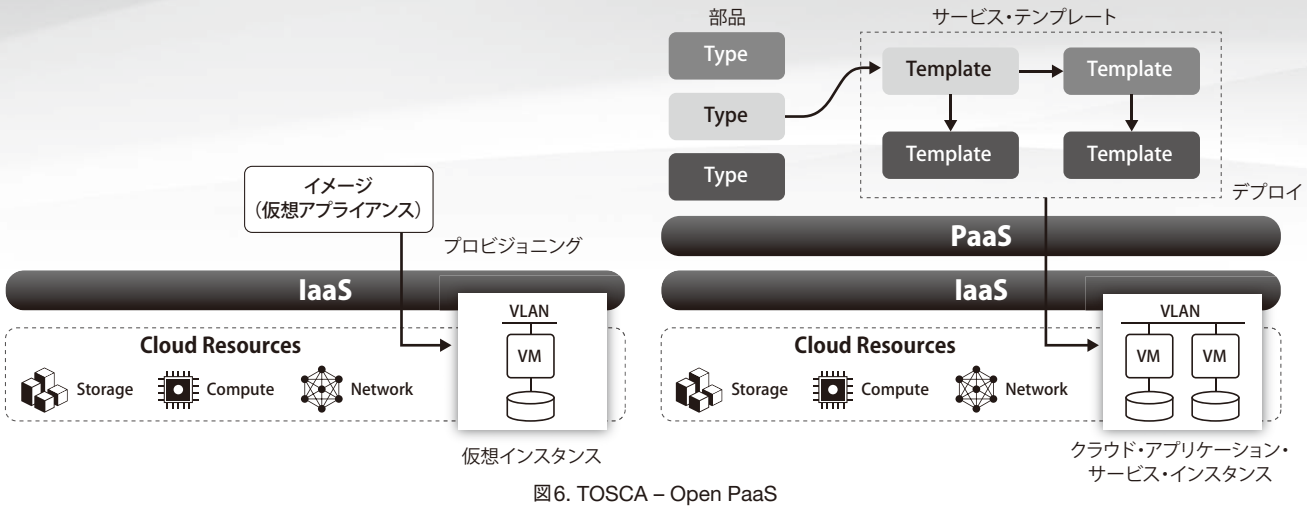

図6. TOSCA – Open PaaS

よびそのトポロジーをデプロイするための手順を、XMLで記述するための標準です。トポロジーとは、例えばWebサーバー、アプリケーション・サーバー、データベース・サーバー(これらがノードです)から成るシステム構成図のようなものです。このXMLファイルをGUIエディターで簡単に作成できれば、それを解釈、実行するシステム(それがPaaSです)にインプットすることで複雑なシステムが一気にデプロイできます。IBMがOASIS TOSCAの技術委員会(Technical Committee)の一員として、仮想アプリケーション・パターンの技術を提供しTOSCAの仕様が開発されました。

図6はIaaSとPaaSの比較をしていますが、IaaSがイメージから一つのVMをプロビジョニングするのに対し、PaaSはサービス・テンプレートを解釈してIaaSのAPIを複数回呼ぶことで複数のVMから成るクラウド・アプリケーション・サービス・インスタンスをデプロイしています。

図7はサービス・テンプレートにおいて、トポロジー・テンプレート、すなわち複数のノード・テンプレート(ノードのひな型)をリレーションシップ・テンプレート(リレーションシップのひな型)で結んだものが、ノードタイプ(型の型)、リレーションシップタイプ(型の型)から組み立てられる様子を示しています。ノードはVMであったり、あるいはVM内のOS、アプリケーションのそれぞれであったり、いろいろな使い方が可能です。リレーションシップは二つのノード間の関係で、例えばアプリケーション・サーバーがデータベース・サーバーに「コネクトする」といったものです。部品から複雑な機能を組み立てる能力をコンポーザビリティーと言い、それこそがTOSCAの本質です。

さらにサービス・テンプレートではBPMN(Business Process Modeling Notation)などで記述されたプランをオプションで指定することが可能です。これによりサービス・インスタンスのデプロイや管理のワークフローを明示することができ、例えばノードのデプロイの順序を指定したり、人が確認したりすることが可能になります。プランを指定しなければ、手順は自動的に決められ実行されます。

TOSCAのもう一つの特徴は、モデルとそれをデプロイするために必要な実装が分かれていることです。トポロジー・テンプレート(XML)はノード・テンプレート(XML)をポイントし、ノード・テンプレートはノードタイプ(XML)をポイントし、それら全体で一つのクラウド・サービスをモデル化しています。しかしモデルをデプロイするためには、実際のソフトウェア(バイナリーコード)、スクリプト、OVFファイルなどが必要になります。これらはモデルとは別にアーティファクトとして存在し、アーティファクト・テンプレート(XML)からポイントされます。そしてモデルとアーティファクトをつなぐのは、ノードタイプ・インプリメンテーション(XML)、リレーションシップタイプ・インプリメンテーション(XML)です。例えばノードタイプ・インプリメンテーションはアーティファクト・テンプレートを指し、それと同時にノードタイプを指します。こ

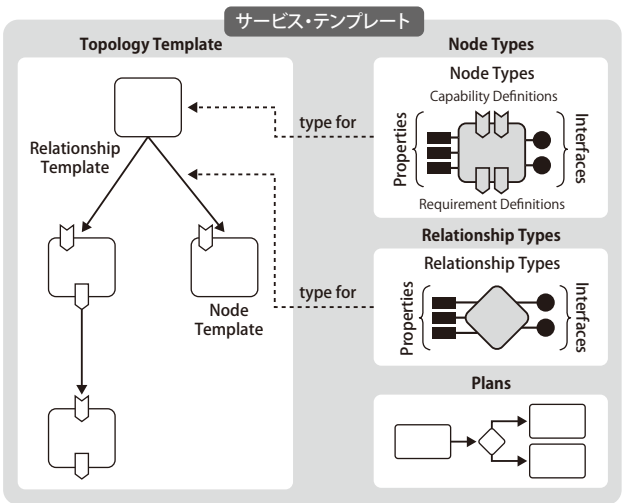

図7. OASIS TOSCA サービス・テンプレート

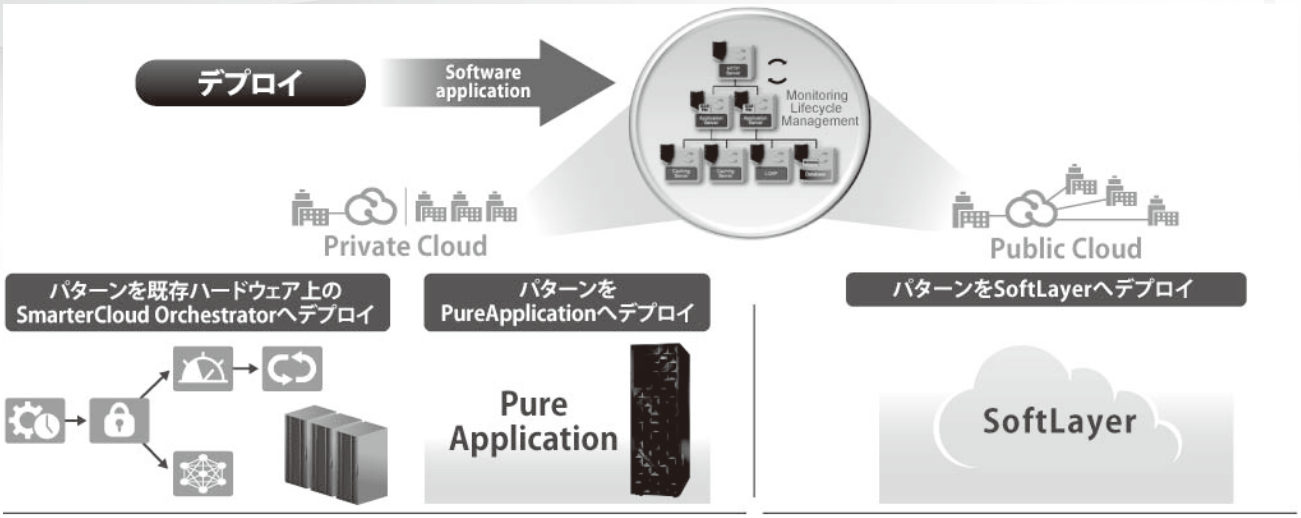

図8. 仮想パターンのインターオペラビリティー

れでモデルとアーティファクトがつながるのですが、大切なのはノードタイプ・インプリメンテーションからノードタイプをポイントするという向きです。これは、ノードタイプはノードタイプ・インプリメンテーションに依存しておらず、ノードタイプ・インプリメンテーション、すなわちアーティファクトを差し替えられることを意味しています。つまりノードタイプ・インプリメンテーションとアーティファクトはPaaSの処理系ごとに都合のよいものを用意し、サービス・テンプレートをデプロイするときにポイントすればそれがバインドされてデプロイされます。

このメカニズムは少々難解ですが、これにより異なるクラウド間でのサービス・テンプレートのインターオペラビリティーを実現しています。**図8**は、TOSCAサービス・テンプレートあるいは仮想アプリケーション・パターンが、IBMのプライベート・クラウド、パブリック・クラウド間で共通に利用できることを示しています。

5. Cloud FoundryとIBM BlueMix

Cloud FoundryはCloudfoundry.orgを活動拠点とするオープンソース・プロジェクトで、オープンソース・ソフトウェアのCloud Foundryを開発しています。ライセンスはOpenStackと同じApache License 2.0で、第三者によるソースコードの再利用や製品化が容易になっています。オープン・ガバナンスを支え、コミュニティー・メンバーの活動を支援するため2013年にCommunity Advisory Board（CAB）が設置され、2013年12月現在IBM、Pivotal、Intel、Canonicalなど9社の代表がボードメンバーを務めています。インターネット上の活動に加え、今後年2回「Platform」というカンファレンスを行う予定です。第1回は2013年9月に米国で開催され、世界中から500名以上が参加しました。

Cloud Foundryはプログラムの開発実行環境を提供するPaaSです。プログラマーがプログラム言語とアプリケーション・フレームワークを選ぶと数十秒でプログラムの実行環境がセットアップされ、Webアプリケーションなどのサンプル・コード（Starter code）が動き始めます。このサンプル・コードをローカルPCにダウンロードすれば、直ちにプログラム開発が始められます。

図9にCloud Foundryの動作概要を示します。アプリケーション開発者はポータルまたはCLI（Command Line Interface）でCloud Foundryにアクセスします。そして上記のようにプログラム環境を選ぶと、DEA（Droplet Execution Agent）と呼ばれるVMにコンテナがセットアップされ、そこにサンプル・コードがロードされ実行を開始します。サンプル・コードをPCにダウンロードし、プログラムを上書きし、CLIのcf pushコマンドでCloud Foundryに戻せば、自分が書いたプログラムが直ちに動き始めます。プログラムを複数のDEA上で稼働させることも可能で、それにより簡単にスケールアウト

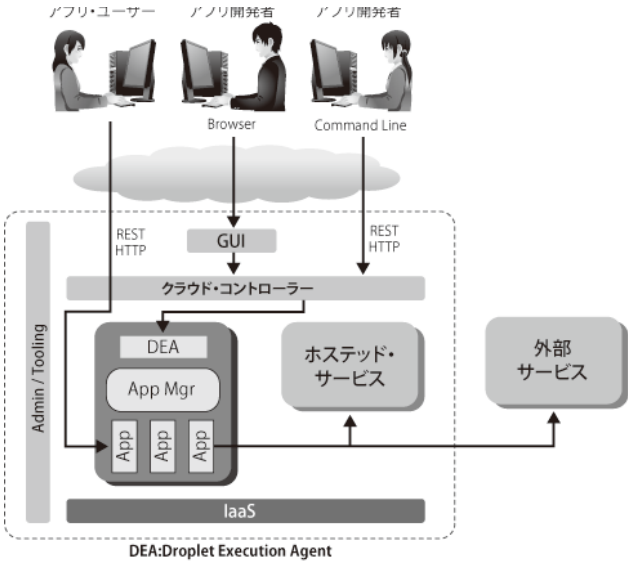

図9. Cloud Foundry – Open PaaS

図10. IBM BlueMix GUI

させることができます。またプログラムにCloud Foundryが提供するサービス、例えばRDBサービスを追加するのも数十秒の操作で行えます。サービスにアクセスするためのクレデンシャルなどの情報は環境変数 VCAP_SERVICESに自動的にセットされるので、プログラムはそれを使ってサービスにアクセスできます。

図10はCloud FoundryによるIBM Public PaaSであるBlueMixのポータル画面です。BlueMixはIBMのパブリック・クラウドであるSoftLayer上にOpenStack、その上にCloud Foundryを載せて実装されています。左側の画面で「ADD AN APPLICATION」をクリックすると右上のプログラム言語やアプリケーション・フレームワークの選択画面が表示され、「ADD A SERVICE」をクリックするとサービスの選択画面が表示されます。

BlueMixを使えばモバイル・アプリケーションも素早く開発できます。モバイル・アプリケーションには、iOSやAndroid端末側のアプリケーションと連動して動くサーバー側のバックエンド・サービスBaaS（Backend as a Service）が必要ですが、これをBlueMix上のアプリケーションとして開発することができます。**図11**に示すようにモバイル・アプリケーション管理、プッシュサービス、Node.jsアプリケーション、モバイル・データ・サービス（サーバー側のデータストア）が統合された形で提供され、すぐに必要な開発ができます。これはIBM Worklightテクノロジーをベースに提供されています。

ここまでの説明でTOSCAとCloud Foundryの違いは明らかですが、TOSCAはサービスの実行環境をモデル化して構築するホワイトボックス型PaaSであり、Cloud Foundryはアプリケーション・フレームワークやサービスのインターフェースより下を隠蔽する（クラウドに任せる）ブラックボックス型PaaSと言えます。インフラの明示的な構成と管理が必要な従来型アプリケーションにはTOSCAが適しており、新たなアプリケーションをクラウド環境専用に素早く作るにはCloud Foundryが適しています。それぞれのPaaS上で適材適所にアプリケーションを構築し、連携させるとよいでしょう。開発者にとってはPaaSこそがプログラムの実行環境かつ意識すべき境界面であることから、Cloud Operating Environment（Cloud OE）と呼ばれます。

6. まとめ

ITシステムには堅牢性・信頼性を重視するSoRと柔軟性・俊敏性を重視するSoEがあり、ビジネス・ニーズに合わせてアプリケーションを最適配置することでコスト削減と事業成長をバランスよく達成することが求められます。そしてSoEの柔軟性は、インフラのサービス化、部品から機能を組み立てるコンポーザビリティー、アプリケーション実行環境の迅速な立ち上げ、多様なサービスを提供するAPI Economyという形で実現されることを解説してきました。

このSoEのEngagementという言葉には二つのものを噛み合わせるという意味があり、自社だけで完結せず外部のデータやサービスを取り込む（噛み合わせる）ことで顧客や社会の変化に迅速に対応するシステムというニュアン

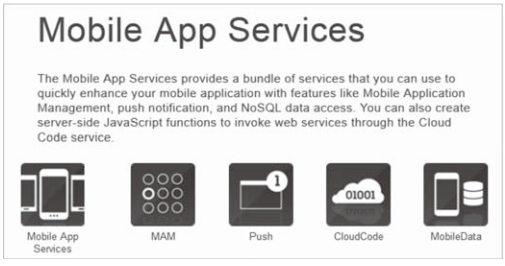

図11. モバイル・バックエンド・サービス（BaaS）を作成

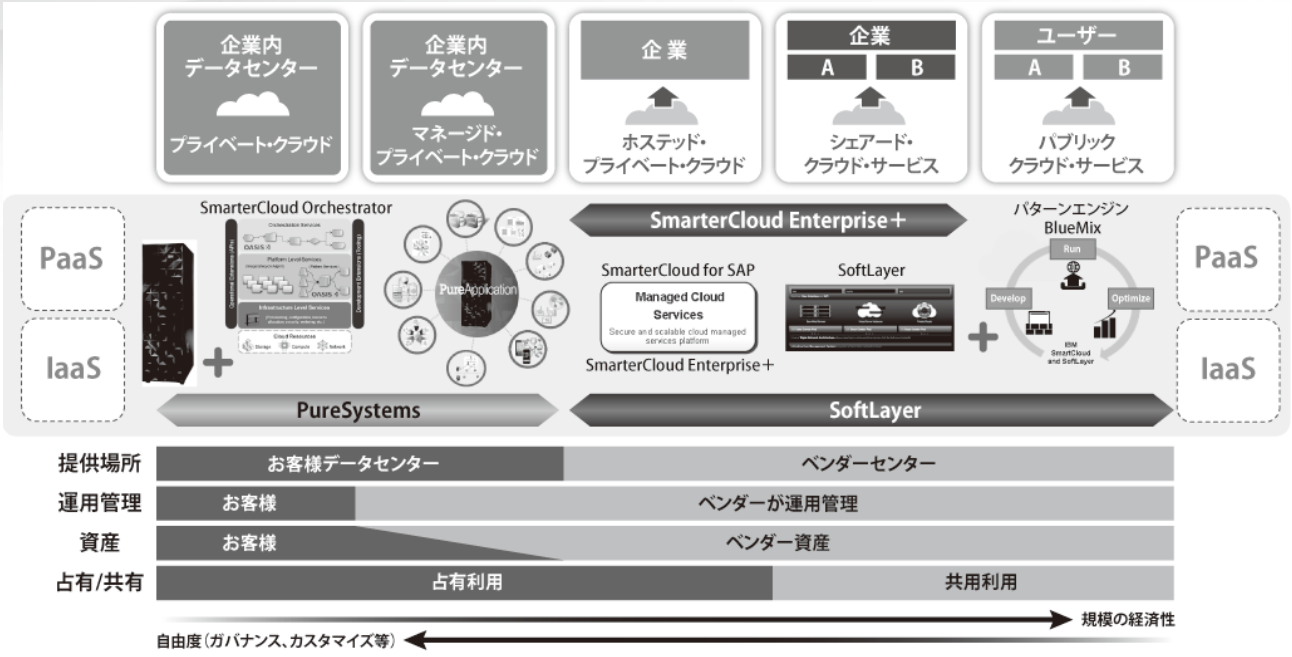

図12. IBMの主要なクラウド・オファリング

スが込められています。それを実現するためには部品やサービスの標準化が不可欠であり、クラウドの適用範囲の拡大と社会インフラ的役割の進展を考え合わせれば、ベンダー・ロックインのないオープン・クラウドが求められるのは時代の必然と言えます。

IBMはオープン・スタンダード、オープンソース・プロジェクトを、技術とオープン・ガバナンスの両面から支援しており、とりわけOpenStack、OASIS TOSCA、Cloud Foundry、OSLCには多くのリソースを投入しています。そして成果物であるオープンソース・ソフトウェアに簡単・安心を加えたIBM製品やサービスをお客様にお届けしています。

図12はIBMの主要なクラウド・オファリングを示しています。PureSystemsのPureFlex（ハードウェア）は仮想基盤およびプライベートIaaSであり、SmarterCloud Orchestrator（ソフトウェア）を導入すればプライベートIaaS兼PaaSになります。また、PureApplicationはハードウェア、ソフトウェアを一つにパッケージしたプライベートIaaS兼PaaSです。

SmarterCloud Enterprise+（SCE+）はホステッド・プライベートIaaS（サービス）で、SmarterCloud for SAP（SC4SAP）はホステッド・プライベートPaaS（サービス）です。これらはゲストVMのOS、ミドルウェア層に対するITIL準拠の運用を含むのが特徴です。SoftLayerはパブリックIaaS（サービス）であり、パターン・エンジンおよびBlueMixの追加でパブリックPaaS（サービス）になります。

IBMは幅広いポートフォリオとオープン・アーキテクチャーによるクラウド間の高いインターオペラビリティーにより、SoR、SoEおよびそれらの統合から成るオープン・エンタープライズ・クラウドを提供することで、お客様のイノベーションをご支援しています。

［参考文献］
［1］A Sea Change in Enterprise IT by Geoffrey Moore, AIIM（2011）.
［2］IBM's open cloud architecture by Angel Diaz, Chris Ferris, developerWorks（2013）.
［3］Supercharging the Cloud for an Agile Enterprise by Mac Devine, Walter Falk, Chris Ferris, Rob Sauerwalt, IBM Redbook（2013）.
［4］OpenStack Operations Guide by Tom Fifield, Diane Fleming, et.al. , OpenStack Foundation（2013）.
［5］Topology and Orchestration Specification for Cloud Applications V1.0, OASYS（2013）.
［6］Topology and Orchestration Specification for Cloud Application（TOSCA）Primer V1.0, OASYS（2013）.
［7］IBM SmartCloud Orchestrator Version 2.3 User's Guide, IBM（2013）.

日本アイ・ビー・エム株式会社
グローバル・テクノロジー・サービス事業
スマーター・クラウド事業統括
理事　クラウド・マイスター

紫関 昭光
Akimitsu Shiseki

［プロフィール］
1981年日本IBM入社。藤沢研究所、大和研究所にて通信機器開発、中型システム開発、ソフトウェア開発に従事。IBM Asia Pacific クロスインダストリーにてビジネス・インテリジェンス・マーケティング。開発製造にてビジネス・ディベロップメント理事。2007年よりGTS事業ITSデリバリー、クラウド事業を経て現職に至る。