

# ビルディング・ブロックで 対象領域を拡大

マルチベンダーのユーティリティ環境を実現する

ストレージ・ソフトウェアのベンダーとして展開してきたベリタスソフトウェア(ベリタス)は、今年5月に米国ラスベガスで開催された「VERITAS VISION 2003」でユーティリティ・コンピューティング分野への参入構想を発表した。企業買収によって基本技術を拡充し、可用性、パフォーマンス、自動化の3つの領域を対象に、ビルディング・ブロック方式でユーティリティ環境の実現を目指す。

ベリタスソフトウェア

## エンタープライズ環境の インフラへ

ユーティリティ・コンピューティング分野での展開を開始したベリタスだが、その主力製品は現在でもストレージ管理ソフトウェアである。この分野のソフトウェア専業ベンダーとしては、トップクラスの地位を確保している。

かつては、SunOS向けのクラスタ・ソフトウェアや高信頼性ファイルシステムの提供などでも有名だったが、現在では、ストレージ分野を中心としたエンタープライズ環境向けの“インフラ”ソフトウェア全般へと、その製品ラインを拡げており、ストレージ管理、データ保護、高信頼性実現、アプリケーション・パフォーマンス管理、ディザスター・リカバリ(災害復旧)など、さまざまなソリューションを提供している。

ストレージ分野を軸に展開していることから、主にコンピューティング・サーバを中心として提唱されることの多い“ユーティリティ・コンピューティング”市場への参入は、ベリタスにとっては一見畠違いにも見えるが、エンタープライズ環境におけるインフラの整備という観点で見れば、それほど違和感なく納得できるだろう。

この点について、米国ベリタスの会長兼社長兼最高経営責任者(CEO)であるゲイリー・ブルーム氏は、「現在のIT市場においては、ストレージ技術が重要な役割を担っている。そのなかでも、特にユーティリティ・コンピューティング分野では主要技術として位置づけられる」と語っており、ユーティリティ・コンピューティング分野は、同社の従来ビジネスの延長線上にあるとの考えを示している。

ベリタスのユーティリティ・コンピューティングへの取り組みは、技術買収から始まった。その対象となったのは、プリサイスとジャレバの2社である。昨年12月、米国ベリタスは、アプリケーション・パフォーマンス管理のソフトウェア・ベンダーであるイスラエルのプリサイス・ソフトウェア・ソリューションズ(Precise Software Solutions)を5億3,700万ドルで買収すると発表、同日にサーバ自動化／プロジェクトマネジメント・ソフトウェアのベンダーであるジャレバ・テクノロジーズ(Jareva Technologies)を6,200万ドルで買収することも発表した。

これによってベリタスは、ユーティリティ・コンピューティング市場への足がかりをつかんだ。そして、この2社

の技術の組み合わせによる純粋なソフトウェア・アプローチが、同社の大きな特徴になる。

一方、ストレージ管理の分野では、ベリタスはすでに高度な仮想化技術を製品化しており、豊富な経験も蓄積していた。「われわれは、企業内に導入されているストレージを有効活用することで、システム管理業務の省力化を図り、コスト削減を実現した。その最大の顧客が米国の通信業界だった。ベリタスが収益を拡大できたのは、この通信業界の寄与によるところが大きい」(ブルーム氏)。

ストレージ分野は、サーバ分野に比べて仮想化の実用化が進んでおり、最先端の環境では、物理的なストレージが何台接続されていて、各ストレージの利用率や空き容量がどの程度あるかといった情報が、管理画面で一元的に把握できるようになっており、それらの情報を管理者が個別に収集・管理するという状況は、過去のものになっている。

つまり、複数のストレージ装置をまとめて巨大な1つのストレージと考え、それを適宜必要なだけ利用するという利用形態が一般化している。そのようなシステムを利用しているユーザーであれば、サーバの演算処理能力についても同様に使いたいと考えるであろうことは容易に想像できる。実際、ベリタス自身も、こうした顧客からの要望がユーティリティ・コンピューティングに取り組む要因の1つにあったと語っている。

## マトリックス・モデルで実現

ユーティリティ・コンピューティングという概念は、ベンダー／ユーザーを問わずおおむね共通の理解に達しているようだが、一方で「どのように実

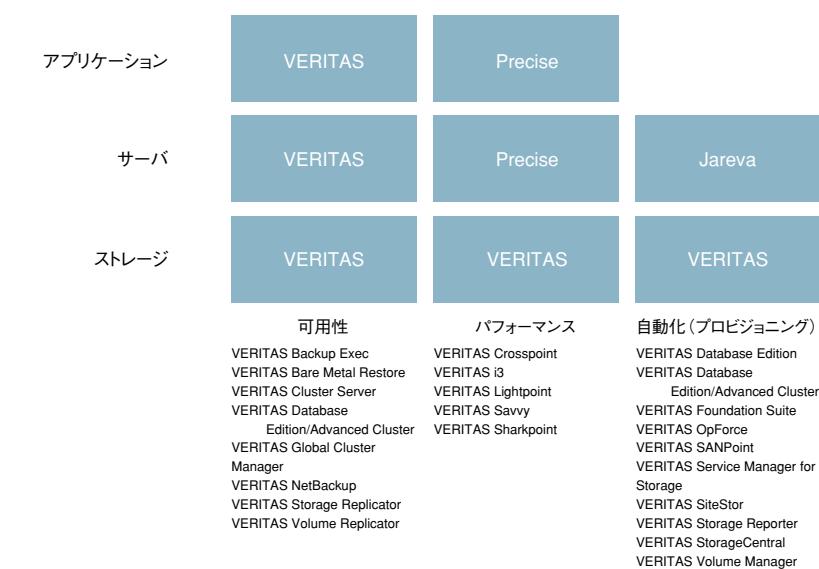


図1:ベリタスのコンピューティング・モデル

現するか」という部分は各社各様である。当然、そのコンセプトの実装方法によって、システム環境の利用方法や使い勝手も異なってくる。

ベリタスのユーティリティ・コンピューティング構想は、“アプリケーション・パフォーマンスの最適化”を基本とし、そこにサーバのプロビジョニング機能を加えるというのが基本的な考え方だ。

そして、その実現に際しては、領域をマトリックスで切り分け、ビルディング・ブロック方式によるソリューションとして提供される(図1参照)。

まず、システム環境をストレージ、サーバ、アプリケーションの3層に切り分け、それぞれに対して要求される機能性として、アベイラビリティ(可用性)、パフォーマンス(性能／キャパシティ)、自動化(プロビジョニング)という3つの観点で対応していく。

このマトリックスのそれぞれの領域に対応する製品を提供すれば、ベリタスの提唱するユーティリティ・コンピューティングが具現化することになる。ベリタスが、ストレージ管理ソフトウェ

アやクラスタ・ソフトウェアを提供することによって、従来より実現してきたのは、ストレージ層における可用性、パフォーマンス、自動化と、サーバ層とアプリケーション層における可用性の部分である。

そして、サーバ層とアプリケーション層のパフォーマンス、自動化の部分が欠けていたが、これを技術買収によって補完した。サーバ層とアプリケーション層のパフォーマンス領域に位置づけられるのがプリサイス社の技術、サーバ層のプロビジョニング領域に位置づけられるのが、ジャレバ社の技術である。このマトリックス・モデルに沿って考えると、“アプリケーション層の自動化”という領域も用意されることになるが、この領域の機能は、現実にはサーバ層の自動化／プロビジョニングにほぼ含まれるため、現時点ではこの領域の概念も対象にした製品計画もない。

なお、プリサイス社の製品はすでに統合を完了しており、現在その製品ラインは「VERITAS i3」ファミリーとして提供されている。VERITAS i3

という製品名は、Insight、Indepth、Informの3語の頭文字にちなんで付けられている。Insightとは、アプリケーションのパフォーマンスをエンドユーザーの応答時間に基づいて測定し、問題のあるコンポーネントを切り分けることを意味している。

また、Indepthは各コンポーネントについてより詳細に解析することを、さらにInformは得られた結果をわかりやすく報告することを、それぞれ意味している。

現在のWebアプリケーションでは、1つのアプリケーションが複数のコンポーネントの集合で構成されている。このため、アプリケーション性能が低下した場合でも、アプリケーションそのもののコーディングから、アプリケーション・サーバ(J2EE)の問題、アプリケーション・サーバ・ハードウェアの処理性能、データベースの応答速度など、性能に影響を及ぼす可能性のある要素は数多く、原因を追究するのは容易なことではない。

さらに、大規模な環境であれば、そもそもアプリケーション性能が低下しているという事実ですら、適切に把握するのは困難な場合が多い。深刻な影響や問題が発生してから問題点を探し始めるのでは、サービス・レベルの低下やビジネスの継続性に大きな支障をきたす。特に大規模な環境では、問題の検出と事前対策のために適切なツールを利用することが不可欠だと言える。

さらに、ジャレバ社の製品によって、サーバの自動インストールが可能になる。つまり、あらかじめ作成しておいたサーバのディスク・イメージを、自動的にインストールする。これによって、たとえば、Windowsが動作しているサーバを、Linux+アプリケーション・

サーバの環境に完全に入れ替えて再起動するといった作業が自動的に行なえる。

この機能を単独で見ると、“サーバ環境の便利なコピーツール”というイメージで捉えられるが、ベリタスの提供する他のソフトウェアと組み合わせることで、サーバ運用の自動化が実現する。たとえば、アプリケーションの性能低下を検出したら、アプリケーション・サーバを新規に起動してクラスタ環境に組み込み、稼働を開始するといった一連の処理が、自動実行できるようになる。

つまり、ベリタスのユーティリティ・コンピューティング環境は、アプリケーション性能をトリガーにして、必要なサーバを自動的に追加する機能であると言い換えることもできる。この方法であれば、特定のハードウェア・アーキテクチャに依存することなく、ソフトウェアのみで実現できる。現時点ではまだ製品化されていないコンポーネントも残っているが、2005年までにはひと通りの製品が出揃う予定になっている。

処理能力に余裕のあるサーバや待機中のサーバをプールしておき、パフォーマンスに問題が生じた処理があれば、それを実行するサーバとして追加するという考え方は、単純ではあるが、システム管理者の負担軽減という意味では非常に効果的だ。ベリタスは、このように現実的な解決策を目指し、現在すぐにでも利用できる具体的なソリューションとして提示することに成功していると言えるだろう。

## システム利用の現場から

ベリタスのユーティリティ・コンピューティング構想では、ビジネスユーザーの需要を意識して、システム全体を

ポリシーベースで管理できるように考えられている。この点も、エンタープライズ・ソフトウェアの分野で豊富な経験を持つベリタスの特徴的な部分と言えるだろう。

将来的にはサービス提供ポータルを用意し、そこでサービス・レベルを一元管理して必要なリソースの割り当てを自動的に行なったり、ビジネス・レベルのレポートингや、ユーザーに対するコスト配分などが実現する計画だ。現時点では、“ユーティリティ”を名乗るために必要な技術要素の整備が先行しており、コスト管理を意識した具体的な取り組みを行なっているベンダーは多くない。

しかし、ユーティリティ・コンピューティングが、電気や水道のような公益事業的なサービスを意味するのであれば、当然そこには「使った分だけ支払う」「コストを明確にする」といった要求も含まれてくる。

ベリタスでは、使用したリソースを正確に把握し、あらかじめ設定された価格に従って集計することで、サービスごとのリソース使用料やITコストを明確化する計画だ。これが実現すれば、あるサービスを実現するためのコストが正確に把握できることになり、ビジネス・プランの立案や遂行の際の重要な資料となることに加え、コストセンターとなっているIT部門を、社内の他部門に対してサービスを実行するプロフィットセンターへと、その位置づけを変えることも可能になる。

さらに、従来は曖昧なまま隠されていたIT関連のコストを明確化することで、さらなるコスト削減が実現したり、本当に追加投資が必要な領域が明確に見えてくる。ベリタスはソフトウェア・ベンダーとして、異種環境への対応、マルチベンダー対応を標榜してきた。現

## ポリシーベースでサービスを管理する 「VERITAS Service Manager」

米国ベリタスは今年5月、米国ラスベガスで開催されたVERITAS VISION 2003で、ITシステム全体のサービス・レベルを定義・管理し、コスト換算する「VERITAS Service Manager」の概要を明らかにした。Service Managerは、開発コード名で「Global Operations Manager」と呼ばれていたもので、システム全体のサービス・レベルを定義し、ポリシーベースでサービス全体を管理す

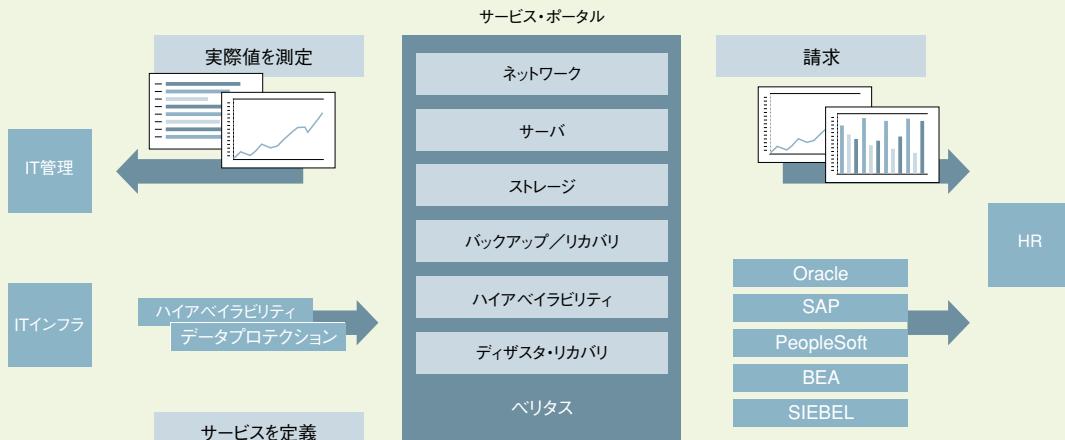
るためのコンソール・ダッシュボードとして位置づけられる。

ITをユーティリティ・サービスとして提供するためには、ユーザーに対するサービス・レベルを設定することが不可欠で、課金や使用料についても明細が必要になるが、Service Managerによって管理者やシステム担当者は、ビジネス・アプリケーションに必要なITリソース(ストレージ、サーバ、バックアップ、高可用

性など)を特定できるようになり、実際にサービスとして提供されたITリソースを追跡し、あらかじめ設定した価格体系に応じて費用を算出できる。

Service Managerは現在はベータ版の段階で、ベリタスによると年末または来年初頭には、第1段階の製品を米国内で提供開始する予定。

Service Managerの概念図



在の企業や組織の環境がマルチベンダー／マルチプラットフォームであることを考えると、ユーザー側でも既存のシステムへの影響を最小限に抑え、無理なく導入できる点がメリットとなる。

### 組織内の ユーティリティ化の実現

ベリタスでは、ユーティリティ・コンピューティングは最終的に、パフォーマンス、自動化されたシステム、共有化されたインフラの可用性という3つの要素で決定づけられるとしている。そして、米国ベリタスのゲイリー・ブルーム氏は、「公益事業と同様に、コンピューティング環境も同じ形で提供されるべ

きだ」と言う一方で、ベリタスがコンピューティング環境をユーティリティ化するという意味は、外部をリソースとして公益事業的な使い方をするのではなく、企業や組織の中において、可用性とパフォーマンスが高く、自動化された共有インフラの環境を構築し、ユーザー専用の公益事業として利用することだと強調している。

この位置づけは、ユーティリティ・コンピューティングを提唱しているベンダーのなかでも特徴的であり、アイ・ビー・エム(IBM)やヒューレット・パッカード(HP)などの大手システムベンダーが、ユーザーのシステム環境全体の運用・管理を一手に引き受けるアウトソ

ーサーとしてもビジネス展開し、ITリソースをサービスとして提供することこそが本来のユーティリティ・コンピューティングであるとする考えと、大きく異なる。これはハードウェアを持たないソフトウェア・ベンダーならではの戦略だと見えるだろう。そして、複雑化したシステム環境の上にソフトウェアを被せ、ハードウェアの複雑性を覆い隠すことでアプリケーション層を構築し、その層の上で可用性とパフォーマンスを管理し、リソースの共有化を図る。ベリタスは、「より少ないコストで多くのことを達成する」という極めてシンプルなコンセプトをもとに、ユーティリティ環境を実現しようとしている。