

|  |  |
| --- | --- |
| * | とっておきの最新情報や MVP による技術解説など、お客様やパートナー様のお役に立つさまざまなコンテンツを満載した SQL Server 2005 情報発信ステーション。ぜひ、ご活用ください。 |
| ** ** | *【2006 年 2 月 27 日更新】**   |  | | --- | |  |   ***   |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | *READY プライムタイム　READY STATION 限定の特別情報 **   |  |  | | --- | --- | | *•* | [SSMA の無償ダウンロード開始　Oracle データベースからの移行を検討されている方は　ぜひ、このツールをご利用ください (2006/2/10 公開)](#t7) | | *•* | [スペシャル座談会　SQL Server 2005 対応ベンダ (ISV) に聞く　なぜ SQL Server を採用するのか (2006/1/13 公開)](#t6) | | *•* | [データベースの第一人者、ジム グレイが語る SQL Server 2005 (2005/12/14 公開)](#t5) | | *•* | [「SQL Server TechCenter」 オープン　IT プロフェッショナルに向けた SQL Server の技術情報は TechCenter へ (2005/11/22 公開)](#t4) | | *•* | [新世代 RDBMS に求められるすべてがある。吉田 育代が徹底解説、SQL Server 2005 のここがポイント! (2005/11/10 公開)](#t3) | | *•* | [Tech・Ed 2005 Yokohama 基調講演「Get Ready for 2005!」 (2005/10/14 公開)](#t2) |   *** | | *READY クローズアップ　MVP が語る SQL Server 2005 **   |  |  | | --- | --- | | *•* | [MVP スペシャル ： ついに発売! SQL Server 2005 はココがスゴイ! (2006/2/27 公開) *](#c8) | | *•* | [MVP インタビュー 第 5 回 ： MVP が見た SQL Server 2005 導入のポイント (2006/2/10 公開) *](#c7) | | *•* | [MVP スペシャル ： Visual Studio 2005 との統合で大きく変わる開発環境 (2006/1/13 公開)*](#c6) | | *•* | [MVP インタビュー 第 4 回 ： 新次元の開発生産性を持つ SQL Server 2005 (2006/1/13 公開)*](#c4) | | *•* | [MVP インタビュー 第 3 回 ： 高度なビジネス インテリジェンス機能を備えた SQL Server 2005 (2005/12/14 公開)](#c3) | | *•* | [MVP インタビュー 第 2 回 ： パフォーマンスと拡張性が向上した SQL Server 2005 (2005/11/10 公開)](#c2) | | *•* | [MVP インタビュー 第 1 回 ： 可用性とセキュリティがさらに向上している SQL Server 2005 (2005/10/14 公開)](#c1) |   *** | |

# **免責事項**

*2005年10月14日から2006 年2月27日まで、マイクロソフトの製品 Web ページにて掲載された連載をまとめたものです。本書に掲載されている情報は作成された時点での情報であり、その内容は予告なく変更される可能性があります。本書に記載された情報は、本書各項目に関する発行日現在の Microsoft の見解を表明するものです。Microsoftは絶えず変化する市場に対応しなければならないため、ここに記載した情報に対していかなる責務を負うものではなく、情報の信憑性については保証できません。本書に掲載されている情報は作成された時点での情報であり、その内容は予告なく変更される可能性があります。本書に記載された情報は、本書各項目に関する発行日現在の Microsoft の見解を表明するものです。Microsoftは絶えず変化する市場に対応しなければならないため、ここに記載した情報に対していかなる責務を負うものではなく、情報の信憑性については保証できません。*

*本書は情報提供のみを目的としています。 Microsoft は、明示的または暗示的を問わず、本書にいかなる保証も与えるものではありません。*

*すべての当該著作権法を遵守することはユーザーの責務です。Microsoftの書面による明示的な許可なく、本書の一部または全部について、転載や検索システムへの格納または挿入を行うことは、どのような形式または手段（電子的、機械的、複写、レコーディング、その他）、および目的を問わず、禁じられています。これらは著作権で保護された権利を制限するものではありません。*

*Microsoft は、本書の内容を保護する特許 (申請中のものも含む)、商標、著作権、またはその他の知的所有権を保有している場合があります。Microsoft から書面による明示的な使用許諾契約書が供給される場合を除き、本書の提供はこれらの特許、商標、著作権、またはその他の知的財産へのライセンスを与えるものではありません。*

*特に断りのない限り、本書に例示した会社、組織、製品、ドメイン名、電子メール アドレス、ロゴ、人物、場所、およびイベントは架空のもので、実在の会社、組織、製品、ドメイン名、電子メール アドレス、ロゴ、人物、場所、あるいはイベントとは一切無関係です。*

Copyright © 2011 Microsoft Corporation. All rights reserved.

Microsoft、SQL Server*、およびWindows は米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。*

*その他、記載されている会社名および製品名は、各社の商標または登録商標です。*

  
*

|  |  |  |
| --- | --- | --- |
| **READY STATION だけの特別情報をお届けする「READY プライムタイム」。** * 他では知ることのできない情報やインタビュー、座談会など、毎回、SQL Server 2005 に関するさまざまな情報を掘り下げてお届けするプレミアム企画ページが「READY プライムタイム」です。今回は、Oracle データベースから Microsoft SQL Server 2005 への移行を検討されている方に耳よりなニュースとして、無償マイグレーション ツール 「SQL Server Migration Assistant for Oracle 日本語版 (以下 SSMA)」をご紹介します。 | | |
| **SSMA の無償ダウンロード開始 Oracle データベースからの移行を検討されている方は ぜひ、このツールをご利用ください** | * | * | |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | * **Index** *   |  |  | | --- | --- | | * | [SQL Server は世界で最も利用されている RDBMS](#t7_1) * | | * | [SSMA は Oracle データベースから SQL Server への移行作業を効率化する画期的なツール](#t7_2) * | | * | [SSMA のデータベース アセスメントは、ぜひ 1 度 試していただきたい機能](#t7_3) * | | * | [PL/SQL で記述されたファンクションやプロシージャも自動的に移行することができます](#t7_4) * | | * | [SSMA のトレーニングやキャンペーンが目白押し。これからも SQL Server 2005 の動向にご注目ください!](#t7_5) * | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | * | * **斎藤 泰行** * マイクロソフト株式会社 サーバー プラットフォーム ビジネス本部 シニアプロダクトマネージャ * | * | * 斎藤泰行シニアプロダクトマネージャ * | * | |  |  | | --- | | * | | | * | |

*  
1 月 25 日より、Oracle データベースから Microsoft SQL Server 2005 へ移行するための作業支援ツール 「SQL Server Migration Assistant for Oracle 日本語版」のダウンロードの無償提供が始まりました。この SSMA は、従来は手作業であったものが自動的に行われるようになり、Oracle データベース システムから SQL Server 2005 への移行作業が効率化できるという画期的なツールです。そこで、マイクロソフト サーバー プラットフォーム ビジネス本部の斎藤泰行シニアプロダクトマネージャに現在の RDBMS 市場の状況と、SSMA の利便性について聞きました。   
*

|  |  |
| --- | --- |
| * | ***SQL Server は世界で最も利用されている RDBMS** |

*

|  |  |
| --- | --- |
| **――** | **SSMA の話の前に、現在の RDBMS 市場の状況から聞かせてください。** |
| **斎藤** | それでは、国内における SQL Server の実績から説明します。2004 年と 2005 年の日経BP社の顧客満足度調査によれば、SQL Server が RDB ソフトウェア部門で第 1 位という評価をいただきました。この評価は Windows プラットフォームだけではなく、メインフレームを除くオープン系すべての RDB ソフトウェア部門での結果が出ています。 また、ガートナー ジャパンによる 2004 年の国内 RDBMS 市場調査では、使用されているプラットフォームとして Windows Server が売上金額ベースでトップ シェアを誇ります。市場での成長率についても、他社が数字を落としている中で、Windows Server プラットフォームのみが、前年度より数字を上げています。そして、その OS 上で最も利用されている RDBMS が Microsoft SQL Server との結果が出ています。 また、海外のデータになりますが、IDC の調査では、RDBMS 市場における SQL Server のユニット ベースのシェアがすべてのプラットフォームにおいてトップであるという結果が出ています。したがって、世界で最も利用されている RDB ソフトウェアは SQL Server であると言っても言い過ぎではないでしょう。 次に、パフォーマンスや処理性能の面ですが、もっとミッション クリティカルなアプリケーションである SAP のベンチマーク結果を例にお話しします。従来の SQL Server 2000 でも、世界で最も大きな SAP ワークロードを持つ企業でも計りしれないほどの同時アクセス数を処理することが可能でした。そしてさらに SQL Server 2005 では、その 4 倍近い同時アクセス数を処理することができるようになり、パフォーマンス テストでも基幹業務システムを快適に運用できることが証明されています。 先日、全世界で行われた製品発表イベントでは、延べで 20 万人以上、また国内でも全国 7 都市で 7,000 人以上のお客様にご参加いただき、SQL Server 2005 に対して非常に高い期待を寄せられていることをあらためて実感することができました。 このように、発売を開始した SQL Server 2005 ですが、従来の SQL Server 7.0 や 2000 を利用されているお客様からも、SQL Server 2005 へ移行したいという声を数多くいただくのはもちろんですが、それ以上に UNIX / Oracle プラットフォームから、SQL Server 2005 へ移行を望まれているお客様が非常に多いことに驚かされています。そして、この Oracle からの移行を検討されているお客様へ、ぜひお勧めしたいのが、今回紹介させていただく SSMA なのです。 |

*

|  |  |
| --- | --- |
| * | ***SSMA は Oracle データベースから SQL Server への移行作業を効率化する画期的なツール** |

*

|  |  |
| --- | --- |
| **――** | **それでは SSMA がどのようなものであるか聞かせてください。** |
| **斎藤** | SSMA (SQL Server Migration Assistant for Oracle) は、Oracle 7.3 以上のデータベースから SQL Server 2000 および 2005 への移行作業の効率化、または自動化するためのマイグレーション ツールです。 本来は手作業で行わなければならなかったマイグレーション作業を自動化できる部分で、このツールを利用していただくことにより、作業を効率化して時間とコストの削減を行おうとするものです。従来では、このツールでは自動化できるようなデータ移行に対しても手作業で行っていたために非効率的だったのですが、これからはデータの移行に余計なリソースを割かなくて済むわけです。 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | **SSMA で効率化と自動化が可能な作業** |   * | | | • | **データベース移行の評価 *** | | • | **スキーマ変換 *** | | • | **データ移行 *** | | • | **データベース オブジェクト変換 (ストアド プロシージャ、関数、トリガ、プロシージャ、シーケンス) *** | | • | **移行後の評価 *** | | |  |  | | --- | --- | | ※ | これらの作業は SSMA が提供する GUI (グラフィカル ユーザー インターフェイス) ツールにより容易に行うことが可能 | | | | |

*

|  |  |
| --- | --- |
| * | ***SSMA のデータベース アセスメントは、ぜひ 1 度 試していただきたい機能** |

*

|  |  |
| --- | --- |
| **――** | **具体的にはどのようなことができるようになるのかを聞かせてください。** |
| **斎藤** | 代表的な機能として、データベースのアセスメント (解析) があげられます。もともとデータベースの移行は、データの解析が困難なために見積もりが難しかったという背景がありました。移行作業を委託されたシステム インテグレータが、データベースの解析を十分に行うことができなかったため、非常にリスクのあるプロジェクトとされてきたのです。しかし、この SSMA では、データベースの解析を瞬時に、かつ簡単に実行することができるため、システム インテグレータがより正確な見積もりを提案することが可能になりました。もちろん、お客様自身がデータの解析や見積もりをすることもできます。このデータベースのアセスメントだけでも利用したい企業も相当数出てきていますので、ぜひ 1 度試していただきたいと思います。 |

|  |  |
| --- | --- |
| http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/07pt_photo03_l.gif * | |
| アセスメントの画面。上のウィンドウで解析したいデータ ソースにチェックを付け、解析を実行する *→  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/07pt_photo04_l.gif  すると手動での作業時間や、自動で変換できる割合などが瞬時に表示されることになる | |
| |  |  | | --- | --- | | * | **PL/SQL で記述されたファンクションやプロシージャも自動的に移行することができます** | |

*

|  |  |
| --- | --- |
| **――** | **PL/SQL で記述されたオブジェクトは変更できるのでしょうか。** |
| **斎藤** | 自動化ができる移行プロセスとして、スキーマの移行、ビジネス ロジックの変換、Oracle と SQL で異なるデータ型の変換などがあります。ここでのビジネス ロジックとは、主に PL/SQL で開発されたストアド プロシージャやファンクションなどになります。PL/SQL の処理を SQL Server に移行することは難しいと思われるかもしれませんが、この SSMA では問題ありません。それは、Oracle 独自の関数などは、SSMA によって SQL Server 側にその関数と同様の処理を行う Transact-SQLで書かれたファンクションに自動的にマッピングされるからです。そしてデータ、スキーマおよびビジネス ロジックの移行後には、このツールのみで単体テストを行うことも可能です。たとえばファンクションであればパラメータを渡した後に、ラッシュ テストを実行してデータがきちんと移行できているかを確認することができます。 まとめますと、基本的にデータベースとして Oracle を利用されている場合は、SSMA で移行することが可能です。既に海外では、MOTOROLA 社を初めとして多くの企業での導入実績もあります。この MOTOROLA 社では、SSMA を活用して Unix/Oracle の環境を 1 日のうちに SQL Server へ移行することができたと聞いております。 しかし、アプリケーションのプラットフォームとして利用していて、データ アクセス コンポーネント数を介してビジネス ロジックやデザイン ロジックなどが入っているような場合には、SSMA での移行の対象外になりますので、その部分は手作業との併用を考慮していただく必要があります。 |

*

|  |  |
| --- | --- |
| * | ***キャンペーンや SSMA のトレーニングが目白押し。これからも SQL Server 2005 の動向にご注目ください!** |

*

|  |  |
| --- | --- |
| **――** | **最後に、SSMA は日本でどのように展開していくのかを聞かせてください。** |
| **斎藤** | Oracle などの他社製データベースから、SQL Server 2005 への移行を検討中のお客様を対象に、ディスカウント キャンペーンを実施します。期間は 2006 年の 2 月から 6 月末までで、ライセンス & ソフトウェア アシュアランス (L&SA) を購入されたお客様に対して、通常価格から最大 50% のディスカウントで提供します。対象データベースや購入製品についての詳細は、キャンペーンのページでご確認ください。 次に、SSMA の無償トレーニングを全国の主要な 7 都市 (札幌、仙台、東京、名古屋、大阪、広島、福岡) で実施します。このトレーニングは、SSMA を活用して移行作業を行う際に必要な技術知識を習得いただくためのトレーニングです。もともとこのツールの使い方自体は非常にシンプルなのですが、使用に際して不安を持たれている方は、このトレーニングをぜひご利用ください。 そして、少し先になりますが、Oracle Master の有資格者を対象としたオンライン トレーニングを、2006 年の 4 月から株式会社システム・テクノロジー・アイと共同で開催する予定です。 この SSMA は無償ですが、マイクロソフトが自信を持ってお勧めするツールです。他社製のデータベースから SQL Server への乗り換えを検討されているお客様には、ぜひ 1 度お試しいただき、その便利さを実感していただきたいと思います。 |

|  |  |  |
| --- | --- | --- |
| READY プライムタイム *   |  |  | | --- | --- | | **READY STATION だけの特別情報をお届けする「READY プライムタイム」。** * 他では知ることのできない情報やインタビュー、座談会など、毎回、SQL Server 2005 に関するさまざまな情報を掘り下げてお届けするプレミアム企画ページが「READY プライムタイム」です。今回は、ISV (独立系ソフトウェア ベンダ) の皆様をお招きし、ついに完成した SQL Server 2005 への期待や SQL Server を採用し続ける理由、Oracle や オープン ソース (PostgreSQL、MySQL) を選定する余地はないのかなどについて、本音で語っていただきました。 | * |   * |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| * | |  | | --- | | **スペシャル座談会 SQL Server 2005 対応ベンダ (ISV) に聞く なぜ SQL Server を採用するのか** | | | | | | |
|  | | | | | * | * | |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | * **Index** *   |  |  | | --- | --- | | * | [なぜ SQL Server を採用したのか](#t6_1) | | * | [SQL Server を採用し続ける理由](#t6_2) | | * | [SQL Server の GUI ツールは大きな魅力の 1 つ](#t6_3) | | * | [SQL Server を採用し続けるのは MSDE の存在も非常に大きい](#t6_4) | | * | [お客様が Oracle を希望した場合はどうしているのか](#t6_5) | | * | [Oracle の場合はサポートやコンサルティングの価格も考える必要がある](#t6_6) | | * | [オープン ソースのデータベースを選定する余地はあるか](#t6_7) | | * | [Access ユーザーへのアドバイス](#t6_8) | | * | [SQL Server 2005 に期待すること](#t6_9) |   座談会の様子   |  |  | | --- | --- | | **【座談会参加者】**(写真左から) | | | 黒川 明宣 氏 | ピーシーエー株式会社 Dream21事業部 部長 | | **玉木 栄三郎 氏** | トッパン・フォームズ株式会社 情報メディア統括本部 チーフアーキテクト | | **三部 雅法 氏** | 東芝テック株式会社 流通情報システムカンパニー 商品開発センター ソフトウェア開発技術部 アーキテクト | | **星野 晃一郎 氏** | 株式会社ダンクソフト 代表取締役 C.E.O. デジタルアーキテクト | |  |  | | --- | | * | | | | | | | * | |
| ** **マイクロソフトは日本語処理のノウハウを非常に持った会社 (星野氏) SQL Server は大変メンテナンスしやすい (玉木氏) ISV 向けサポートが非常に充実。聞きたいときに聞ける窓口がある (黒川氏) SQL Server 6.5 以降はようやく安定稼動するようになった (三部氏)** | | | |
| ***――** | | **まずは皆さんの自己紹介からお願いします。** |

|  |  |
| --- | --- |
| **星野氏** | 私はダンクソフトという会社で、「ロバート・システム」というパッケージ ソフトを販売しています。我々は利益管理と呼んでいますが、企業の中の基本的なお金の流れ、仕掛かりから受注、仕入れ、発注、請求、入金までの流れを 1 つのパッケージとして一元管理できるのが特徴です。もう 1 つのビジネスとしては、金融系のポータル サイトを関連会社が運営しており、5 年位が経ちます。そういった金融系の、特に証券会社系のビジネスもしています。 マイクロソフト製品は 1980 年代の後半から使い始めて、Windows NT は初期バージョンの 3.1 から、Access も初期のバージョンから採用しています。日本で最初に Access が発売されたときのマイクロソフトのダイレクト メールには、Access 上で動作する弊社製品の案内が一緒に入っており、そのころからデータベースに携わっています。 |

|  |  |
| --- | --- |
| **玉木氏** | 私はトッパン・フォームズという会社に勤めています。トッパン・フォームズは、凸版印刷の子会社になりまして、地味な会社ですが、おそらく皆さんの家にも 1 つは弊社の製品があるかと思います。たとえば大手宅配会社さんの送り状ですとか、携帯電話やクレジット カード、銀行の利用明細、あとは請求などによくあるバリっとはがす圧着ハガキなどが我々の製品です。そういうものの製造、印刷などをしている会社です。 私自身は、情報メディア統括本部という部署におりまして、IC カードや IC タグ、IC ラベルといった、ポスト バーコードと言われている RFID (Radio Frequency Identification : 無線 IC タグ) 技術を使った製品の取り扱いをしています。IC タグの方はあまり目にはしませんが、IC カードでは FeliCa や Suica、Edy、JAL といったカードを発行して、いくつかの企業に提供しています。 また、マイクロソフトさんと業務提携し、「RFID .NET ソリューション センター」というものを共同で運営しています。ここでは、RFID をうまくハンドリングしていくためのいろいろな技術を、.NET や Windows アーキテクチャの中にインプリメントしていくことを現在行っています。私自身は、そういった中での製品開発や啓蒙活動をしています。 |

|  |  |
| --- | --- |
| **黒川氏** | ピーシーエーの黒川です。当社では、パソコン用のパッケージ ソフトという形で、財務会計や PCA 会計というものを中心に、販売、仕入、在庫管理、その他周辺の税務まで、開発、販売しています。また、1999 年からは 21 世紀に向けて「PCA21」というプロジェクトを立ち上げ、「PCA Dream21」という ERP パッケージを開発して 2001 年より販売開始しました。 実は当時、私はシステム開発部におりまして、入社以来 16 年間、PCA 会計の設計、開発に携わってきました。Dream21 では開発部の立場として、特徴と優位性について社内外への普及活動を主に行っていました。「PCA Dream21」は、当時マイクロソフトが提唱していた Windows DNA というアーキテクチャを採用し、すべて COM+ で実装しており、いち早く Web サービス モジュールをパッケージとして発売しています。ただ、そういったメリットを正しく伝えられる営業がいなかったので、「作った本人が売ってこい」ということになりました。そこで 2003 年に Dream21 事業部という部署を作り、そこへ異動して現在に至っています。実は、今でもパートナー様向けの Web サービスのインターフェイスを実装したり、ちょっとした開発を行ったりすることもあります。 |

|  |  |
| --- | --- |
| **三部氏** | 東芝テックの三部です。流通システム研究所という部署に勤務し、ソフトウェアのアーキテクトをしています。今抱えている製品は、「CrossMission」という .NET の上にアプリケーションを構築するためのミドルウェアです。通常フレームワークというと何層にもなっているのですが、CrossMission は .NET Framework とアプリケーションの間に入るミドルウェアという位置付けです。オープン ソース系ではアプリケーション サーバーがそういった位置付けになるのですが、それらは Web アプリケーションを対象としています。我々の CrossMission は Web アプリケーションだけでなく、クライアント / サーバー型の Windows アプリケーションや XML Web サービスまでを対象とし、すべてをカバーできるような形で出荷しています。 |

|  |  |
| --- | --- |
| * | ***なぜ SQL Server を採用したのか － Access を共有利用するとパンクするのが目に見えていた － UNIX や Oracle は不便で面倒 － SQL Server は非常に強靭にできている － 数千以上の店舗へデータベースを導入するには価格も決め手だった** |

*

|  |  |
| --- | --- |
| **―――** | **次に、皆さんが SQL Server を採用した理由や、それにまつわるエピソードについてお話しいただけますか。** |

|  |  |
| --- | --- |
| * | * |

|  |  |
| --- | --- |
| **星野氏** | 私どもが SQL Server を最初に採用したのは 1995 年ごろになります。当時は Oracle などの大きめのデータベースは、どちらかというと汎用であるとか、ワークステーション クラスで止まっていて、パソコンのネットワーク上だと覚えている限り 1995 年までは、Btrieve (Novell NetWare) くらいしかまともに動作するものがありませんでした。それ以外の選択肢では Sybase の製品か Access をサーバー上において、みんなが共有するというレベルでした。星野氏 ところがそれだと、利用者が 10 人を超えるとパンクするというのが目に見えていたので、どうしようかと考えたときに、当時の SQL Server 6.0 と Windows NT 4.0 という組み合わせを採用することにしました。これが使ってみると非常に良かったので、それ以降はずっと SQL Server 6.5、7.0、2000 と使い続けています。 |

|  |  |
| --- | --- |
| **玉木氏** | 私個人は、UNIX、Sun、Oracle とずっと使ってきて、周りからは難しいことをやっているように見えて格好いいのですが、いかんせんコマンド ベースなので「不便だなあ」「面倒だなあ」と常々感じていました。その点、Windows は GUI ベースなので本当に便利ですね。会社では、最近は仕様を策定する立場にいますので、「面倒な部分は省いてビジネスに集中しましょう」ということで、積極的に Windows アーキテクチャを取り入れるようにしています。 私自身は SQL Server 6.5 からの個人あるいは法人ユーザーで、SQL Server というよりは Windows プラットフォーム全体を使っています。 |

|  |  |
| --- | --- |
| **黒川氏** | SQL Server は、会社としては SQL Server 4.2 から研究を重ねてきましたが、私自身は SQL Server 6.5 から本格的に利用するようになりました。弊社が最初にデータベースに携わったのは MS-DOS の時代で、そのころに会社として OS/2 用のプレゼンテーション マネージャに対応したパッケージを作ろうという話になりました。ただ、まともに使えるデータベースがなかったので、データベースを自社開発したというのが始まりです。しかし、自分たちで 1 から開発するというのは明らかに難しく、非常に面倒でした。 また、1997 年に PCA 会計をリニューアルするときに、Btrieve (NetWare が標準で提供しているデータベース機能) を利用するということで設計がほとんど完成していたのですが、そのころの ISAM 系や Btrieve では処理が多重化してくると、追いつかなくなりデータが壊れてしまうことがありました。また、当時の質の悪いネットワーク カードでは、瞬断などのネットワーク障害時に簡単にデータが飛んでしまったりと本当に苦労しました。 そういった理由から Btrieve では駄目だということになり、設計をすべてやり直して SQL Server を採用することにしました。SQL Server は、私自身は初めてでしたが、使っているうちに便利で非常に強靭にできているということを感じ、現在に至るまで SQL Server を使い続けています。 |

|  |  |
| --- | --- |
| **三部氏** | 私が SQL Server に携わったのは、最初のバージョンの SQL Server 4.2 からです。それまでは、Sybase や Oracle などを利用していました。 元々の流通系のシステムというのは、本部サーバーやマスタ サーバーにはオフコンを使ったり、ミニコンを使ったりと、どちらかというとファイル系が中心でした。そこから PC を使ったシステムに移行しようというときに、「店舗ごとにデータベースを入れたい」という話になりました。コンビニエンス ストアさんとかになりますと、各店舗のマネージャがデータを分析して、品揃えを決めたり、発注をしたりということがありましたので、それぞれの店舗にデータベースを置きたいということになりました。 そこで、数千以上にもなる店舗へ導入するデータベースは何にすべきかと考えたときに、選択肢としてはいろいろあったのですが、価格も決め手となって SQL Server を採用することになりました。実際使ってみると、正直 SQL Server 4.x の時代にはトラブルが多かったのですが、SQL Server 6.5 からはなんとか安定稼動するようになって、ダウンする確率が低くなりました。ここからスーパー マーケット系でも使えるようになりましたね。また、SQL Server 7.0 からは、本部系でもようやく使えるようになったという形です。 |

*

|  |  |
| --- | --- |
| * | ***SQL Server を採用し続ける理由 － 日本語処理のノウハウを非常に持っている会社であること － ISV 向けのサポートが非常に充実していること － メンテナンス フリーであり続けること** |

*

|  |  |
| --- | --- |
| **―――** | **皆さんは、SQL Server 4.x または 6.x 以降、SQL Server を使い続けておられますが、ズバリその理由は何でしょうか。** |

|  |  |
| --- | --- |
| **星野氏** | 私がデータベースを選択するうえで 1 番重要視しているのは、日本語処理の問題です。最近は少なくなってきましたが、昔はせっかくデータを入力したとしても、文字化けしてデータを取り出せなくなってしまうということがよくありました。 |

|  |  |
| --- | --- |
| **玉木氏** | ごくごく最近までそうでしたよね。データベース自体が対応していたとしても、ドライバが対応していなくてそういう問題が起こったり。 |

|  |  |
| --- | --- |
| **星野氏** | ええ、どこかがダメで結果が返ってこないということがありました。そういう意味では、SQL Server を含めた Windows プラットフォームは、日本語処理の部分で非常に安心感があります。あまり言われていないのですが、マイクロソフトは日本語処理のノウハウを非常に持っている会社の 1 つだと思っています。10 年以上も前から日本語をハンドリングしている会社ですから、たいへんな信頼感があります。 Linux 上で動作するデータベースは、その辺はどうなのかなと考えたときに、対応できる技術者は多分いないと思うんですよ。Linux を採用したとして、「日本語入力は本当にできるのか」とか、「文字化けせずにきちんと印刷できるのか」といったような点がものすごく気になります。そういったところがデータベースを選択する際の 1 番重要な要素の 1 つではないでしょうか。たとえ世界中で売れていても、日本でそれを採用するかどうかには、まったく違うポイントがあります。そういう問題はあまりメディアには出てこないので、企業としては心配になります。 |

|  |  |
| --- | --- |
| **黒川氏** | 私が SQL Server を採用し続けている理由は 2 つあります。1 つは、マイクロソフトの ISV 向けのサポートが非常に充実している点です。聞きたいときに聞ける窓口がある、これは何にも勝る安心感です。もう 1 つは、扱いが非常にイージーな点です。このサイトの MVP インタビュー (小川氏) の記事の中で「リッチなイージー」という表現がありましたが、まさにそのとおりだと思います。 |

|  |  |
| --- | --- |
| **玉木氏** | 私どもは、完成されたパッケージを販売するという形ではありませんので、SQL Server に限らずお客様の希望によっていろいろなデータベースをまんべんなく使ってきました。ただ、お客様がデータベースを細かく指定してくるというケースは少なく、「安定して動けばいい」という要望だったりします。そうなると、安定稼動かつメンテナンスがしやすいということで、3、4 年前からは Windows と SQL Server を積極的に提案してきました。 お客様の訪問先では、エンジニアが辞めてしまって、扱える人がいなくなって困っている UNIX 系のシステムがあったりします。バックアップさえもできずに個人的に相談されたりするのですが、そういった場面に遭遇するたびに Windows と SQL Server を勧める思いが強くなりました。 |

|  |  |
| --- | --- |
| * | * |

|  |  |
| --- | --- |
| **黒川氏** | サポートをビジネスにされている方にとっては、システムは複雑な方がいいかもしれませんが、パッケージにとってはデータベースはあくまでも裏に隠れていてほしいものです。我々の業務ソフトを利用されるお客様は、技術者だけではありません。PCA 会計を使う方が経理の事務社員であったり、また、PCA Dream21 (ERP) を使用されるのは企業の社長だったりします。 コンピュータを使い慣れた IT エンジニアだけが使うわけではありませんので、パッケージの裏で動くデータベースも、できる限り手間のかからないものの方が良いのです黒川氏。そういった意味では、SQL Server は今のところほとんど初期設定のままで入っていますし、導入後もほぼメンテナンス フリーです。仮に何かあったとしても、GUI ツールがしっかりしていますので、まったくわからないお客様でも、電話で指示を出してトレースしてもらうことができます。他のデータベースだとこう簡単にはいきません。 |

|  |  |
| --- | --- |
| **―――** | **確かに Oracle だと電話でトレースをとるための指示を出すのは困難ですね。** |

|  |  |
| --- | --- |
| **三部氏** | 私が SQL Server を使い続けている決定的な理由は、個人的にはもう使い慣れていて、酸いも甘いもかみ分けていると言いますか、どこをどうチューニングすればパフォーマンスを改善できるかというのがだいたいわかっているところにあります。初期のバージョンのころからのノウハウが蓄積されてきて、もちろんバージョンが変わるごとに多少の性格の違いはあるのですが、使い慣れたというのが 1 番ですね。 |

*

|  |  |
| --- | --- |
| * | ***SQL Server の GUI ツールは大きな魅力の 1 つ** |

|  |  |
| --- | --- |
| **―――** | **SQL Server の GUI ツールというのは、やはり大きな魅力の 1 つでしょうか。** |

|  |  |
| --- | --- |
| **星野氏** | 最初のとっかかりとしては重要ですよね。Windows のインターフェイスなので、かなり簡単に操作できて、クリックしていくだけで実際のデータまで見ることができますから。 |

|  |  |
| --- | --- |
| **三部氏** | SQL Server 6.5 からは、安定稼動すると共に、ユーザー インターフェイスが格段に良くなりましたね。当時、他のシステムを担当している同僚から「何故 Oracle を使わないんだ」と言われたことがありますが、「だって SQL Server はこんなに簡単なんだ」と見せると、「ああ本当だね」と容易に納得させることができました。 |

|  |  |
| --- | --- |
| **黒川氏** | 私は GUI ツールがなければ触っていなかったと思います。他社のデータベースも GUI ツールを出されていますが、どうも作り慣れていないというか、使いづらいんですよね。 |

|  |  |
| --- | --- |
| **星野氏** | 他社のデータベースは、そういうことを考えて作っていないのだと思います。データベースですからね。表に出てくるものではないので、そこは技術者が触ればよいだろうと考えている会社が多いと思います。 |

|  |  |
| --- | --- |
| **玉木氏** | Oracle も 8i からは Java ベースのインストーラと管理ツールが付きましたが、何と中途半端なものを付けたのだろうという印象でした。これが Windows ベースだったら少しは違ったのかもしれませんが、それでもう抵抗ができてしまって「それならコマンドでやる」と思ったものです。結局、Oracle を使うときには、サード パーティ製の GUI ツールを使われる方がほとんどですね。 |

|  |  |
| --- | --- |
| **黒川氏** | そう考えると、マイクロソフトは日本語の問題といい、GUI ツールといい、日本固有のニーズをよく吸い上げていると思います。 |

*

|  |  |
| --- | --- |
| * | ***SQL Server を採用し続けるのは MSDE の存在も非常に大きい** |

*

|  |  |
| --- | --- |
| **―――** | **SQL Server と互換性のある無償のデータベース エンジン「MSDE」\* についてはいかがでしょうか。** |

|  |  |
| --- | --- |
| **―――** | \* : MSDE は SQL Server 2005 からは Express Edition が後継となります。 |

|  |  |
| --- | --- |
| **黒川氏** | SQL Server を採用し続けるのは、MSDE の存在も非常に大きいですね。パッケージのスタンドアロン版には MSDE を使ってもらい、ネットワーク版にアップグレードしたときに SQL Server を使ってもらうことになるわけですが、ほとんど同じ操作性で使えるのが嬉しいですね。MSDE がなければ、Btrieve をそのまま使い続けて SQL Server を採用していなかったかもしれません。しかし、もし Btrieve を採用していたとしたら、今これだけのデータ量はさばけなかったと思います。 |

|  |  |
| --- | --- |
| **三部氏** | MSDE の存在は大きいですね。我々のシステムでは、POS レジに SQL Server または MSDE が入っています。各店舗のストア サーバーと本部のサーバーには SQL Server が入っています。本部にあるデータベースの構造 (テーブル レイアウト) と、店舗サーバーにあるデータベース構造、POS レジのデータベース構造を同一にして、店舗コードやレジ No. を含めた形でキーを構成することにより (本来は POS レジには店舗コードは不要なのですが) 上から下まで一気通貫で構築できる。そうなるとどこへでも簡単にデータの移動やシフトができる SQL Server は本当に助かりますね。SQL Server も MSDE もストアド プロシージャも含めてデータベースは完全互換がありますから。 |

*

|  |  |
| --- | --- |
| * | ***お客様が Oracle を希望した場合はどうしているのか** |

*

|  |  |
| --- | --- |
| **―――** | **お客様の会社が既に Oracle を導入済みで、「ぜひ Oracle でやってほしい」と希望されたケースはありましたか。** |

|  |  |
| --- | --- |
| **黒川氏** | ありましたね。パッケージの場合は、データベースはあくまでもエンジンとして動いていますので、滅多にあることではないのですが、大手メーカーさんの子会社で、親会社からデータベースの技術者の方がやってきて「うちは Oracle じゃないと技術者がいないから」と言われたことがあります。でも「エンジンに対して特別な技術者はいりません」と対応しました。 実は、PCA 会計は Oracle 版を出していたこともあるのですが、Oracle 版だと 1 件 1 件のサポートに手間がかかって仕方がありませんでした。データ量が増えてくると、チューニングが必要になってきますが、Oracle だとなかなか思ったように動いてくれません。その点 SQL Server であれば自動チューニングができるので、ほとんど何も考えずに使えます。もちろん 100 パーセント完璧とまでは言いませんが、自動チューニングがそこそこ効いて、メンテナンス フリーであり続ける SQL Server というのは非常にありがたい。今は、PCA 会計の Oracle 版が動いているところをどうやって SQL Server 版にリプレースしていこうかと考えています。そういう意味では、PCA という会社の中では Oracle の影は薄らいできています。 |

|  |  |
| --- | --- |
| **玉木氏** | Oracle を希望されるお客様の場合、必ずしも明確な理由があるわけではないケースも多いですね。Visual Basic や Access、Excel をクライアントとして使うのになぜ Oracle を使うんだろうと思ったこともあります。Oracle では、Excel を使うためにわざわざ KeySQL を購入したりするケースもあります。 |

|  |  |
| --- | --- |
| **―――** | **確かに Oracle 社の発した宣伝をそのまま受け取って、雰囲気だけで Oracle を選択されるという方もいらっしゃいますね。そういう方には、ぜひ 1 度 SQL Server を試してほしいと思います。** |

|  |  |
| --- | --- |
| **星野氏** | 当社では、お客様から Oracle の要望が出たことはないのですが、OS との親和性を考えるとやはり SQL Server をお勧めしますね。Oracle は、ミニコンでも、ワークステーションでも動いて、さらに PC のネットワークでも動くというマルチ プラットフォーム製品であるがゆえに「本当に速いの？」と疑いたくなります。OS が変わってもきちんとチューニングできているのか、という疑問が残り、それは今も引きずっていると思っています。 |

|  |  |
| --- | --- |
| **玉木氏** | なんだかんだで Oracle が 1 番稼動している OS は Windows が 1 番多いというのもあったりして、そういうところでも矛盾が起こっていますよね。 |

|  |  |
| --- | --- |
| **黒川氏** | 少し外れるかもしれませんが、マイクロソフトは OS のロードマップが明確にあるじゃないですか。それに対して SQL Server もきちんとロードマップに対応した製品が互換性を保ちながら出てくるという安心感があります。 Oracle の場合は、Windows をバージョン アップすると互換性の部分があまり良くなかったりします。我々のお客様は、100 人使っていれば 100 人が Windows を使っていますので、OS のバージョン アップにきちんと互換性がないと、提供する側の恐怖心があります。 |

*

|  |  |
| --- | --- |
| * | ***Oracle の場合はサポートやコンサルティングの価格も考える必要がある** |

*

|  |  |
| --- | --- |
| **三部氏** | 私どもは、お客様が Oracle を希望した場合は Oracle ですね。ただ、自分で選べる場合は、価格の面から SQL Server を選択します。これは単なるソフトウェアの価格だけではなくて、サポートやコンサルティングまでを含めた全体の価格になります。Oracle だとそういった諸々の費用が本当に高いですから。 |

|  |  |
| --- | --- |
| **―――** | **Oracle はソフトウェア (データベース) のライセンス料金だけでも本当に高いですね。特に Enterprise Edition で規模が大きくなると桁違いの金額差になりますし、SQL Server では標準搭載されるような機能がオプションとして別途費用がかかったりするのは大きな差ですね。** |

|  |  |
| --- | --- |
| **黒川氏** | 最近は、Oracle は高くないという宣伝を見ることがあります。Linux なら OS が無償だからということらしいのですが、実は Linux 上で Oracle というのは、それをサポートするために高額な対価を支払わないとサポートしてくれるところがなかったりします。そう考えると実は安くなくて、逆に高くつくのではと思ってしまいます。 |

|  |  |
| --- | --- |
| **三部氏** | Oracle は、データベースのチューニング ツールでも、Quest Software や VERITAS Software (現 Symantec) などが販売していますが、Oracle 版は SQL Server 版に比べて 1 桁高かったりします。その分機能も豊富ですが、Oracle だから許される特別な価格レンジというのがあるようです。 |

*

|  |  |
| --- | --- |
| * | ***オープン ソースのデータベースを選定する余地はあるか** |

*

|  |  |
| --- | --- |
| **―――** | **オープン ソースのデータベース (PostgreSQL や MySQL) の導入を検討されたことはありますか。** |

|  |  |
| --- | --- |
| * | 玉木氏 * |

|  |  |
| --- | --- |
| **玉木氏** | 低コストでどうしても何とかしなければならないという案件では、PostgreSQL や MySQL を使うこともあります。ただ、いくら初期コストが安く抑えられても、導入後の運用や拡張性までを含めた全体でコストが安いかと言うとそうではありません。急にエンタープライズ的な要素を求められて、たとえばクラスタを導入したいなどと言われると、サード パーティの製品を買わなければならず、結果として高く付いたりします。それなら最初から SQL Server を使おうということになるわけです。 また、最近の新聞記事の話になりますが、ある小学校で Linux ベースの PC を導入していたのを全部 Windows に入れ替えたそうです。読み進めていくと、結局メンテナンスやソフトのアップデートに時間がかかるのと、最後には「Linux に詳しい担当者が異動になったため」という経緯が書いてありました。だいたい Linux が入っているところは、Linux に詳しいエンジニアがいたりします。しかし、そのエンジニアがいなくなったときのことも考えて導入する必要がありますね。 |

|  |  |
| --- | --- |
| **黒川氏** | それはまさに当社の状況にも当てはまっています。社内に Linux 上で動作する、とあるシステムがあるのですが、それが PostgreSQL を使っています。当時、在籍していた技術者が無償だからということで作ったシステムなのですが、その技術者がいたときは良かったんですが、退職してしまった途端にそのシステムはバージョンが固定です。もうハードウェアが壊れないのを祈るばかりです。 |

|  |  |
| --- | --- |
| **―――** | **なるほど。オープン ソースの採用にあたっては、導入時だけでなく、その後の障害対応やバージョン アップといった運用管理までを含めて考慮する必要があるというわけですね。** |

|  |  |
| --- | --- |
| **玉木氏** | あと、オープン ソースで 1 番たいへんなのは、いろいろな場所からデータを集めてくるプログラムを書かなければいけない点ですね。SQL Server であれば、DTS (データ変換サービス) が標準で付いているので、わざわざそういったプログラムを書かなくてもデータが集められます。オープン ソースを使っていると「DTS を使わせてくれ」という場面がよくあるため、DTS は本当に便利だと感じます。 DTS は、SQL Server 2005 からは Integration Services に名称が変わって、さらに良くなっているのですが、これだけでも SQL Server を買う価値が十分にあります。 |

*

|  |  |
| --- | --- |
| * | ***Access ユーザーへのアドバイス － Visual Studio 2005 は Access と同じような感覚で SQL Server を利用できる** |

*

|  |  |
| --- | --- |
| **―――** | **日本では Access から SQL Server への移行を検討されているユーザーがたくさんいらっしゃると思いますが、何かアドバイスはありますか。** |

|  |  |
| --- | --- |
| **星野氏** | 日本では、MS-DOS から Windows に切り替わった時代のデータベース ソフトというと Access くらいしかありませんでした。1995 年ごろからの数年間で SQL Server に移行しないで Access をそのまま使い続けているというケースは本当に多いですね。しかし、ネットワークを介して Access を使用する場合、利用者が増えるとすぐにパンクしてしまいます。すべてのデータを持ってきてしまうので、これは厳しいですね。 また、今までは Access ユーザーにとって SQL Server は敷居が高く感じるところがあったかもしれませんが、Visual Studio 2005 からは大きく変わっています。Visual Studio 2005 を使えば、Access と同じような感覚で簡単に開発できます。SQL Server のデータベースをコピー & ペーストし、Access と同じようにローカル ファイルのように取り込んでテストすることができます。テスト環境のイメージは Access とほとんど同じですね。これがあれば、移行はそれほど難しくないと思います。 |

*

|  |  |
| --- | --- |
| * | ***SQL Server 2005 に期待すること** |

*

|  |  |
| --- | --- |
| **―――** | **ついに SQL Server 2005 が完成しましたが、どういったところに期待していますか。** |

|  |  |
| --- | --- |
| **黒川氏** | 業務ソフトとして心待ちにしているのは「データベース ミラーリング」機能です。これは本当にすごくいい。特にインターネットを介して物理的に離れた拠点でミラーリングを行いたいというニーズは非常に高いです。 |

|  |  |
| --- | --- |
| **玉木氏** | ディザスタ リカバリ (Disaster Recovery: 災害復旧) 用途ですね。 |

|  |  |
| --- | --- |
| **黒川氏** | 日本では地震の問題はやはり考慮しなければなりません。噂では、沖縄のデータ センターが人気があるらしいですね。そう考えると、データというのはすごく大切だと思われているということですね。 |

|  |  |
| --- | --- |
| **三部氏** | 今は回線が太くなってきて、大量のデータを流しても問題がなくなってきているのも大きいですね。 |

|  |  |
| --- | --- |
| **玉木氏** | 私が SQL Server 2005 で期待しているのは「64 ビット版」ですね。私は比較的大規模なデータをハンドリングすることが多いので、SQL Server 2000 でも途中から 64 ビット版は出ましたが、それまではずっと不自由をしていて、物理的にないということを言われて肩身の狭い思いをしてきました。非常に悔しい思いをしてきましたが、ようやく最近の大規模な案件では 64 ビット版を活用しています。 それから「Reporting Services」の存在も大きいですね。我々は、カードのポイントが何ポイント貯まったとか、そういった Web ベースのグラフを作成することが多くあります。今まではサード パーティ製のコンポーネントを買ったりする必要があったのが、Reporting Services があれば、ドラッグ & ドロップで簡単にレポートが作れたりします。システムに応じて臨機応変の対応が求められる会社にとっては、非常に便利な機能です。 |

|  |  |
| --- | --- |
| **―――** | **Reporting Services は、SQL Server のすべてのエディションに標準機能として同梱されているのもありがたいですね。** |

|  |  |
| --- | --- |
| * | 三部氏 * |

|  |  |
| --- | --- |
| **三部氏** | 私はやはり「SQLCLR (SQL Server 2005 に統合された、.NET 共通言語ランタイム)」ですね。私自身ここ 5 年ずっと .NET をやってきましたので、ようやく期待に沿うものが登場したという感じがします。パフォーマンスを考慮すると、ストアド プロシージャに頼ることが多いのですが、単にクエリを発行し、UPDATE して、INSERT するだけではなく、その中でかけ算や引き算、割り算をしたりとデータの操作が発生します。そうすると、ときどきパフォーマンスに問題が出たりするのですが、SQL Server 2000 では解決策がありませんでした。 それが SQL Server 2005 で SQLCLR が登場したことで、C# でストアド プロシージャが書けるようになります。もちろん、処理の内容によってパフォーマンスが速いものと遅いものがあるのですが、速いものであれば、50 倍から 70 倍ぐらいに速く演算できるという検証結果も出ています。今あるシステム全部を置き換えるとなるとたいへんですが、要所要所で困っているところを置き換えていける、そういった解決策が出てきたというのが非常にうれしいですね。たいへん助かります。 今まではストアド プロシージャで駄目ならもう駄目というところがあったのが、まだ改善できる余地があるというのは大きいですね。 |

|  |  |
| --- | --- |
| **黒川氏** | パッケージを全部 SQLCLR に置き換えるとなると、作り直しになってしまいますが、パッケージにアドオンするプログラムとか、それと連結するプログラムを書くときには、確かに SQLCLR は便利ですね。 あと、私個人としては「ADO.NET 2.0」が非常に気に入っています。ADO.NET は非接続型のアクセスがよく取り沙汰されるのですが、実は業務パッケージには非接続型というのは非常に使いづらいところがあります。今回お話させていただいたように、我々の業務ソフトを利用されるお客様は、社長や経理の事務社員といったコンピュータを使い慣れていないユーザーです。そのためセットアップ 1 つにしても、昔はお客様に「setup」という 5 文字を打たせるのではなく、「p」という 1 文字だけにして手間をなくすという努力をしてきました。 しかし非接続型では、登録ボタンを押してコミットしたときに、裏で誰かに書き換えられてますよというメッセージが表示されたりします。これでは、ユーザーにひどく怒られてしまいます。書き換えられているのなら、最初から入れた瞬間に通知しておけというのが理由です。入れられるから入れたのに、登録段階でこれは裏で更新されているというのは「冗談を言うな」と。そういう世界で私は仕事をしていますので、ある程度接続型のアクセスをしなければなりません。ところが従来の ADO.NET のデータ セットというのは、接続型で使うには非常に使いづらいところがありました。それが ADO.NET 2.0 になって、たいへん使いやすくなりました。こういった理由から、ADO.NET 2.0 は個人的にマイブームです。私と同じような業務アプリケーションを書かれる方には非常にお勧めの機能です。 |

|  |  |
| --- | --- |
| **三部氏** | あと SQL Server 2005 には、「リアルタイムなデータ分析」と「XML」機能に期待しています。最近の POS システムは、各店舗にサーバーを置かない集中センター サーバー系にだんだん変わってきています。POS システムでは、レシート 1 枚に対してトランザクションが発生しますが、トランザクションがリアルタイムでセンター サーバーに反映されるようになって、それをリアルタイムで分析できるようになることが求められてきています。また、リアルタイムに多事象分析をしつつ、その裏で動く発注系のシステムとリアルに連携できるだけのパフォーマンスが出ることも SQL Server に期待したいところになります。 もう 1 つは XML で、これはもしかしたら SQL Server 2005 ではなく、その次のバージョンに期待することなのかもしれません。今後 RFID がすべてのものに付くようになったときに、そのマスタ データは全部が同じテーブル構造かというとそうではないと思います。ただ、問い合わせてくるユーザーは、その RFID に入っているキーだけで問い合わせるので、その構造に対しても情報が必要になります。しかしさまざまなテーブルがあるので、それを全部 XML のデータとしてデータベースの中に格納するようにして、かつインデックスを付与してすばやく取り出せるようになるという需要が、今後あと 1、2 年で必ず高まると思っています。米国では、やはりそういうのをいろいろな研究者が既に研究していて、すごく速い XML 専用のデータベースが存在しているのですが、それらはコストが非常にいい値付けになっています。そこで、それらに匹敵するだけの XML のサーチ パフォーマンスを備えた SQL Server が登場し、かついつもの価格で提供されるということを期待しています。 |

|  |  |
| --- | --- |
| **黒川氏** | これは実は、ちょうど私どもの基幹系にもあります。最近 POS レジ系の案件が多いのですが、基幹業務の構造はわりと複雑でして、1 つの伝票から残高の問題や、在庫の問題とか、ERP 系では振替伝票の問題もあったり、かなり重い処理をすることになるのですが、そのときにリアルタイムでデータ分析をしたいということを必ず言われます。このとき、前裁きとしてどれだけ速いパフォーマンスが出るかというのがポイントになりますね。 また、異なった構造のテーブルを簡単に扱える製品というのを望まれる方も非常に多いですね。当然、基幹業務に流れてくるのは決まった POS レジの形式だけではなく、経費精算の振替伝票であったり、営業日報などに経費が埋め込まれたような形であったりもするわけです。そういったものを高速に前で裁いておいて、裏で重い処理 (ERP の基幹業務へのストア処理、裏でサマリされた結果) ができる、つまり請求書や決算書が作れます。お客様としてはそれらが取得できれば良いわけですから、そういったところが今後のポイントになってくると思います。 |

|  |  |
| --- | --- |
| **―――** | **皆様、本日はどうもありがとうございました。SQL Server を古くから導入され、現在も使い続けていらっしゃる皆様からたいへん貴重なお話を伺うことができました。これから SQL Server の導入を考えていらっしゃる方々にとって、大いに参考になったのではないかと思います。長い時間、本当にありがとうございました。** |



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **データベースの第一人者、ジム グレイが語る Microsoft SQL Server 2005**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  | | --- | | 「the Microsoft Conference 2005」のスペシャル キーノート 1 では、SQL Server の開発に大きな影響を与えてきたマイクロソフト コーポレーション リサーチ グループのジム グレイ (Jim Gray) が、SQL Server 2005 について、さまざまな視点から詳細な解説を行った。 | | |  |  |  | | --- | --- | --- | | |  |  | | --- | --- | | |  | | --- | | → [講演当日のビデオ (日本語字幕付き)](http://www.microsoft.com/japan/windowsserversystem/applicationplatform/launch2005/jimgray/500k.asx) (500KB) | | | | |  | | --- | |  | | | |  |  | | --- | --- | | |  | | --- | | → [講演当日のビデオ (日本語字幕付き)](http://www.microsoft.com/japan/windowsserversystem/applicationplatform/launch2005/jimgray/1mb.asx) (1MB) | | | | | * | |  | | --- | | PHOTO | | | |  | | |  |  |  | | --- | --- | | profile   |  | | --- | | **ジム グレイ (Jim Gray)**  マイクロソフト コーポレーション リサーチ グループ テクニカル フェロー  グレイ博士は、データベース、およびトランザクション処理の分野においての専門家であり、マイクロソフトにてスケーラブルコンピューティング (コモディティのソフトウェアとハードウェアからスーパーサーバーやワークグループシステムの構築) に注力。マイクロソフトへの入社前には、Digital、Tandem、IBM、AT&Tにて、RDB、ACMS、NonStop SQL、Pathway、System R、SQL/DS、DB2 および IMS-Fast Path などのデータベースやトランザクション処理システムに従事。 バークレイ校にて博士号、シュトゥットガルト、パリにて名誉博士号を取得。National Academy of Engineering、American、Academy of Arts and Science、the National Academy of Science の会員であり、ACM フェロー、データマネジメントに関する Morgan Kaufmann Series の編集者でもある。1998 年には、ACM 最高の栄誉であるチューリング賞を受賞。また、データベース、およびトランザクション処理システムの書籍、トランザクション処理/概念と技法などの著者。 | |  |  |  |  | | --- | --- | --- | | * | |  | | --- | | **SQL Server 2005 は第 4 世代、 もっともこだわった点がセキュリティ。** | | | * |   長年にわたってデータベースに関わっているジム グレイは、ここ 20年 におけるハードウェアとソフトウェアの進化から講演を始めた。「1985 年時点のゴールは 1,000 トランザクション/秒でしたが、実際は 100 トランザクション/秒しか達成することができませんでした。ところが、現在は 8,000 トランザクション/秒をらくらくと突破しています」と語る。このハードウェアとソフトウェアの驚異的な発展があってこそ、SQL Server 2005 の開発が可能となったのだ。しかも IT 革命は始まったばかりで、まだまだ継続するという。   SQL Server もまさにイノベーションの歴史であり、4 世代にわたった発展を続けてきたとジム グレイは説明する。SQL Server 6.0/6.5 の第 1 世代から SQL Server 7.0 の第 2 世代を経て、第 3 世代の SQL Server 2000 へ。そして、今回リリースされた SQL Server 2005 が第 4 世代というわけだ。   各世代において絶えず新たな機能を加えることで SQL Server は進化してきたが、今回の SQL Server 2005 が何よりもこだわったのはセキュリティだという。「SQL スラマー (コンピュータ ウィルス) からは本当に手痛い攻撃を受けました。しかし、私たちはそこから学ぶことも忘れませんでした」と語るジム グレイは、「SQL Server 2005 は、セキュア バイ デザイン、セキュア バイ デフォルト、セキュア バイ デプロイメントをモットーにしています」と、あらためて今回セキュアなデータベースの開発が大きなテーマであったことを強調した。   |  |  |  | | --- | --- | --- | | * | |  | | --- | | **SQL Server 2005 はここが "クール"。 「.NET Framework との統合」「Visual Studio との統合」。** | | | * |   ジム グレイは、SQL Server 2005 の "クール" な点として、「.NET Framework との統合」「Visual Studio との統合」「ビジネス インテリジェンス」を挙げる。そして、リレーショナル データベース、XML、テキスト、ETL ツール、レポーティング、自動チューニング…などが標準搭載されていること、Visual Studio を初め、Windows やBizTalk Server など他のマイクロソフト製品と統合できること、また、ソケット単位での価格設定のため、TCO (総所有コスト) を大幅に削減できることが、SQL Server 2005 の価値であると伝えた。   ここでモニタは、ぎっしりと細かな字で埋めつくされた画面に切り替わる。「これは、SQL Server 2005 の約 1,000 にも及ぶ機能の中から、100 個に絞り込んだものです」と、ジム グレイ。この機能のすべてが SQL Server 2005 の "クール" な点だとアピールした。   次いでプログラマに向け、「.NET Framework との統合」「Visual Studio との統合」について補足説明を行った。「プログラム言語とデータベースのミスマッチを解消するために .NET があるのです。共通言語ランタイム (CLR) の統合を使用すると、任意の .NET Framework 言語でストアド プロシージャを作成できます。このストアド プロシージャを使ってデバッギングやブレークポイントの設定なども可能になります。また、Visual Studio との統合によって、さらに充実した開発環境をプログラマに提供できるようになります」と解説。そして、「SQL Server と Visual Studio をいかに統合させるか。そのために、この SQL Server 2005 の開発に 5 年を要することになったのです」という内幕も語ってくれた。   |  |  |  | | --- | --- | --- | | * | |  | | --- | | **長期開発の副産物? SODA (サービス指向データ アーキテクチャ)。** | | |  |   SQL Server 2005 の開発期間中に Web サービスが本格化し、SOA (サービス指向アーキテクチャ) が注目を浴びるようになった。ジム グレイは「データを .NET Framework でカプセル化し、Web サービスでやりとりするのが SOA ですが、私たちはこのデータのカプセル化に特定の方法があることを発見し、それを SODA (Service Oriented Data Architecture: サービス指向データ アーキテクチャ) と名付けました」という。そしてデータを 4 種類 ― (1) エンタープライズのコアの情報であるリソース データ、(2) そのキャッシュ コピーである参照データ、(3) メッセージが行き交うサービス統合データ、(4) ジャーナルやログといったアクティビティ データ ― に分類し、SQL Server 2005 では、これら 4 種類のデータをフィーチャしながら、SODA を構築しやすい環境を提案していくと語った。   そのための機能として、データベース上でネイティブに実装、稼動できる Web サービスや疎結合型のアプリケーション構築の問題を解決できるサービス ブローカ、SQL Server 2000 から大幅に進歩したクエリ通知などがあるという。   そして「SQL Server 2005 は非常にすばらしい XML のサポートを持っている」という声が、既に多くのユーザーから寄せられていることを報告した。   |  |  |  | | --- | --- | --- | | * | |  | | --- | | **データの出し入れに新機能、 データソース統合とレポーティング。** | | | * |   SQL Server 2005 はデータをどのように抽出し変換するか。その答えが、データ変換サービス (DTS) と呼ばれていたものを刷新したSQL Server インテグレーション サービス (SSIS) だ。「ファイル、レガシー システム、Web サービス、さらにはオラクル、DB2 …、このインテグレーション サービスはどこからでも自由にデータを抽出できます」という。そしてそれをパイプラインで変換し、処理していくのだが、そのパフォーマンスが並列処理によって、従来のデータ変換サービスより約 10 倍速くなっているとも語った。   また、データをどう外に出すかについては、レポーティング サービスが挙げられる。Visual Studio との統合によって、インタラクティブなレポートが簡単に作成できることが SQL Server 2005 の特長だという。さらには KPI (主要業績評価指標) についても、プレゼンテーションがしやすいようにアイコン表示などを可能にしている。   |  |  |  | | --- | --- | --- | | * | |  | | --- | | **「データ→情報→知識→英知」 という流れの中で。** | | | * |   次にジム グレイはデータベースを高い視点から語ってみせる。「データベースから始まる流れがあります。まずデータがあって、それが情報に変わります。そしてその情報が知識へ、その知識が英知へと変わっていくのです。現時点のデータベースはデータをカバーしているに過ぎませんが、SQL Server 2005 はその境界をさらに上のレベルまで動かそうとしています」。   データから情報へ、情報から知識へと変えるために不可欠なのが BI (ビジネス インテリジェンス) やデータ マイニングであるが、中でも重要なのが「予測」だという。これはリレーショナル データベースにはできないことで、「サポートとコンフィデンス (信頼性)」という新たな考え方が必要だと説く。この概念は SQL Server 2005 の中に反映され、新たに追加されたアルゴリズムによって、データ マイニングはより一層の拡張性を見せている。「今までは、データ マイニングは統計学者やエキスパートの領域でした。SQL Server 2005 のリリースによって、ここにいるお客様にもアプローチしやすい環境ができたのです」と自信をのぞかせる。   さらなるデータ マイニングの革新や、ファイル システムとデータベースを融合させるための Win FS など、今後の展開を語りながらも、ジム グレイはあらためて今回の SQL Server 2005 に対する熱い想いを込めながら講演を締めくくった。   「SQL Server 2005 は、Microsoft Windows、Visual Studio、Office 12、SharePoint、BizTalk Server、そのすべてを束ねたものです。さまざまなプラットフォームでデータを管理します。SODA のプラットフォームとしても初めてのものです。XML や Web サービスにも対応します。そして、何よりも使いやすいのです。私自身も 5 年間、待ちわびました。皆さん、後はお使いいただくだけです」。 |

  
*

|  |
| --- |
| **READY STATION だけの特別情報をお届けする「READY プライムタイム」。** * 他では知ることのできない情報やインタビュー、座談会など、毎回、SQL Server 2005 に関するさまざまな情報を掘り下げてお届けするプレミアム企画ページが「READY プライムタイム」です。今回は、IT プロフェッショナル向けの SQL Server に関する運用情報や技術情報を集約した「SQL Server TechCenter」のオープンに合わせて、TechNet と SQL Server TechCenter の概要をご紹介します。 |

|  |  |  |
| --- | --- | --- |
| **「SQL Server TechCenter」 オープン IT プロフェッショナルに向けた SQL Server の技術情報は TechCenter へ** | * | * |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | * **Index** *   |  |  | | --- | --- | | * | [IT プロフェッショナルのための TechNet と、開発者のための MSDN*](#t4_1) | | * | [SQL Server の IT プロフェッショナル向け技術情報はすべて TechCenter に集約](#t4_2)* | | * | [英語版の情報が日本語でリアルタイムに](#t4_3)* | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | * | * **柄澤 文恵 氏** * マイクロソフト株式会社 IT Pro オーディエンス マーケティング グループ シニアプロダクトマネージャ * | * | * 柄澤文恵氏 * | * | |  |  | | --- | | * | | | * |

*  
11 月 17 日、SQL Server に関する IT プロフェッショナル向けの技術情報を一箇所に体系的に集約した情報サイト、「SQL Server TechCenter」が TechNet オンライン上で公開されました。この TechCenter のオープンによって SQL Server の評価、導入、運用に関する技術情報が、今まで以上に入手しやすくなります。このニュースは READY 編集局としても見逃すことはできないので、さっそくマイクロソフト IT Pro オーディエンス マーケティング グループの柄澤文恵シニアプロダクトマネージャに TechNet オンラインと SQL Server TechCenter について聞きました。   
*

|  |  |
| --- | --- |
| * | ***IT プロフェッショナルのための TechNet と、開発者のための MSDN** |
| **――** | **SQL Server TechCenter について伺う前に、TechNet をご存知でない方のために、その活動内容を紹介してください。** |
| **柄澤** | TechNet は「IT プロフェッショナル」とよばれる情報システムの評価、導入、運用を担当される方々に対して、技術情報を提供するマイクロソフトのプログラムです。 TechNet の具体的なサービスとしては、Web による情報提供である「TechNet オンライン」を中心にして、「TechNet Flash ニュースレター」による最新情報の配信、そして IT プロフェッショナルの方が TechNet オンラインの技術情報に加えて、マイクロソフト製品の評価用ソフトウェアやベータ版を CD-ROM で毎月入手できる「TechNet Plus サブスクリプション」を提供しています。 TechNet オンラインは、IT プロフェッショナルの方なら誰でも Web から簡単に、OS やソフトウェアの技術情報やセキュリティ情報などの最新情報を入手できるサービスです。マイクロソフトのサイトの中でも 5 本の指に入るほど頻繁に利用されているサイトです。そして、この TechNet オンラインの最新情報を隔週で無料配信しているサービスが TechNet Flash ニュースレターです。IT プロフェッショナルの方々に役立つイベント情報なども併せて配信しています。 TechNet Plus サブスクリプションは、TechNet オンラインの情報を CD-ROM に収録して有償で提供しているサービスです。TechNet オンラインでは、情報を自分で入手しにいかなければ更新された情報を見ることができませんが、TechNet Plus サブスクリプションを購読すれば、毎月 CD-ROM で提供されるので重要な情報を見落とすことなく、すぐに手元で必要な情報が得られるというメリットがあります。また、マイクロソフト製品のソフトウェアの評価版も試用期限無しで提供されますし、有償技術サポートのマイクロソフト プロフェッショナル サポートを 2 インシデント無償でご利用いただけますので、そのメリットは大きいと思います。 |

|  |  |
| --- | --- |
| Photo * | * |

|  |  |
| --- | --- |
| **――** | **技術者向けの情報提供というカテゴリでは、MSDN が既にありますが、TechNet と MSDN との違いはどこにあるのでしょうか。** |
| **柄澤** | MSDN は、情報システムの「開発」のフェーズで仕事をされている技術者の方々に対して情報を提供しています。一方、TechNet は、システムの評価、導入、運用といったフェーズで仕事をされている IT プロフェッショナルの方々に向けた情報の提供を行っています。 |

|  |  |
| --- | --- |
| **――** | **TechNet オンラインの中で一押しのコンテンツはありますか。** |
| **柄澤** | TechNet オンラインでは、タスク処理を自動化するためのスクリプトを「スクリプト センター」で公開しているのですが、そちらは多くの方々から好評をいただいています。 その他にも、IT プロフェッショナルの方々が Active Directory を初め Windows XP Professional や Office 2003 Editions を展開する際に必要となる情報やガイダンスを提供する「Deployment Center」というサイトがあります。10 月にリリースされたばかりなのですが、こちらもお勧めです。 TechNet オンラインでは、実に多くの情報が無償で提供されているので、もっと多くの IT プロフェッショナルの方々に、情報を活用していただければと思っています。ただ、情報が非常に膨大なため、それをうまくナビゲーションすることが必要です。その 1 つの試みが新しく開設した TechCenter なのです。 |

|  |  |
| --- | --- |
| * | ***SQL Server の IT プロフェッショナル向け技術情報はすべて TechCenter に集約** |
| **――** | **それでは、今回オープンする TechCenter について聞かせてください。** |
| **柄澤** | TechCenter は、これまで、TechNet オンラインや各製品サイト内に散在していた IT プロフェッショナル向けの技術情報を 1 か所にまとめ、わかりやすい分類やナビゲーションによって、お客様がより情報にアクセスしやすい環境を提供するものです。 たとえば、これまで SQL Server の技術情報は、SQL Server 製品サイトと TechNet オンラインの両方のサイトを探さなければなりませんでした。それが、これからは TechCenter の中で SQL Server の技術情報のすべてを探すことができるのです。 この TechCenter はワールド ワイドで展開を始めているサービスで、日本では最初に Exchange Server の TechCenter が 8 月にスタートし、SQL Server の TechCenter は、11 月 17 日、2 番目にオープンしました。 現在、Exchange Server TechCenter は、アメリカ、ラテン アメリカ、日本で公開されています。SQL Server については、これまでアメリカのみで TechCenter が提供されていましたが、今回日本でも世界に先駆けてサイトが公開されています。その理由は、IT プロフェッショナルの方々の人数が、日本はアメリカに次いで多いという背景があるからです。 |

|  |  |
| --- | --- |
| **――** | **今後の製品サイトは、いわゆるポータル サイトのような窓口的な形になるのでしょうか。** |
| **柄澤** | 今までと変わりありませんが、技術情報ではなく製品を選定する際に有用な情報が中心のサイトになります。したがって、開発フェーズでの情報を入手したい方は MSDN を、評価、導入、運用フェーズでの技術情報を入手したい方は TechNet をといったお客様の仕事内容に応じた情報提供を行っていくことになります。 |

|  |  |
| --- | --- |
| **――** | **技術情報がなくなってしまうのではなく、TechCenter のサイトに一部が移るということですか。** |
| **柄澤** | そうですね。各製品サイトの構成が変わるということはありません。入門レベルの技術情報に関しては、製品サイト側からも閲覧することができます。さらに深い技術情報に関しては、TechCenter の方に収録され、開発者向けの情報は MSDN に収録されるということになります。ですから、製品サイトで詳しい技術情報を探しているユーザーは、気が付かないうちに TechCenter へ誘導されているという形になります。 |

*

|  |
| --- |
| SQL Server 関連サイト相関図 * SQL Server 関連サイト相関図 |

**

|  |  |
| --- | --- |
| * | ***英語版の情報が日本語でリアルタイムに** |
| **――** | **TechCenter を利用するメリットとして、技術情報が集約されるという以外に何がありますか。** |
| **柄澤** | 従来までは、製品情報や技術情報は英語で提供され、若干のタイム ラグがあった後に日本語版が提供されていました。しかし、TechCenter の一部の技術情報については、英語による情報提供と同じタイミングで日本語による情報提供を行いますので、情報をタイム ラグなしに日本語で参照することができます。 そして更新された情報は TechNet Flash で配信されます。製品ごとにタイミングが異なりますが、TechNet Flash を購読していただければ、常に最新の情報を得ることができます。他には TechNet オンラインのトップ ページにてブログを公開していますので、そちらでも最新の情報がチェックできます。 |

|  |  |
| --- | --- |
| **――** | **SQL Server の製品サイトや READY STATION と TechNet オンラインの関連性はどのようになっているのでしょうか。** |
| **柄澤** | READY STATION を含めた製品サイトは、今後さらに製品を選定する人向けのサイトに姿を変えていきます。したがって、READY STATION も製品サイトのコンテンツの 1 つとして位置付けられていき、これからもタイムリーな情報を集約していきます。製品サイトを今までご覧になられた方は感じられたと思うのですが、何が新しくて何が古い情報なのかが非常にわかりにくい構成になっていました。それをわかりやすく時系列的に網羅するという位置付けのものが READY STATION なのです。それ以外の競合製品との情報比較や、購入時の情報などについては、製品サイトの中に収録されます。 |
| **――** | **SQL Server ユーザーと READY STATION を読まれている方に対して、メッセージをどうぞ。** |
| **柄澤** | Photoまずは、TechNet オンラインと TechCenter に 1 度訪れていただきたいと思います。TechNet オンラインはお客様の声を受けて改善していくというプロセスを持っています。スクリプト センターが拡充されたことも、お客様からの要望が高かったからです。ユーザーの方のご意見、ご要望が TechNet オンラインを育てていくことになりますので、ぜひ、ご協力いただきたいと思います。 また、TechNet オンラインや TechCenter は、IT プロフェッショナルの皆様の仕事のうえで活用いただける情報をわかりやすく提供することが、私たちスタッフの最大のミッションだと思っています。 また、TechNet を活用していただく際には、無料登録で利用できる TechNet Flash ニュースレターの購読から始められることをお勧めします。TechCenter や、それ以外の TechNet オンラインの最新情報を定期的に入手することができるので、情報システムにかかわる仕事をされているのでしたら、ぜひ購読してください。皆様の仕事のお役に立つことができるものと確信しています。 |

  
*

|  |  |  |
| --- | --- | --- |
| **READY STATION だけの特別情報をお届けする「READY プライムタイム」。** * 他では知ることのできない情報やインタビュー、座談会など、毎回、SQL Server 2005 に関するさまざまな情報を掘り下げてお届けするプレミアム企画ページが「READY プライムタイム」です。今回は、いよいよリリースへの秒読みが始まった SQL Server 2005 の、時代を先取りする進化について、ポイントを絞り込んで解説します。 | | |
| **新世代 RDBMS に求められるすべてがある。 吉田 育代が徹底解説、SQL Server 2005 のここがポイント!** | * | * | |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | * **Index** *   |  |  | | --- | --- | | * | [進化を享受しよう](#t3_1) | | * | [大規模基幹システムの中核を担える最速 OLTP エンジン](#t3_2) | | * | [セキュリティこそ生命線](#t3_3) | | * | [これからの RDBMS は Web サービス レディでなければ](#t3_4) | | * | [全社で活用可能な BI 基盤](#t3_5) | | * | [軽快なフットワークで動くには、開発生産性、運用管理性の高さがカギ](#t3_6) | | * | [一番新しい RDBMS が、ビジネスの新しい地平を開いてくれる](#t3_7) | | |  |  |  | | --- | --- | --- | | * | * **吉田 育代 氏** * 大阪市出身。関西大学社会学部卒。阪急百貨店宣伝部に、コピーライターとして入社。 1 年半後に上京し、広告制作プロダクションを経て、IT 分野をカバーするフリー ライターに転身。 企業情報システムを初め IT 分野全般にわたって、幅広く執筆活動を行う。吉田育代氏 * | * | |  |  | | --- | | * | | | * | |

*

|  |  |
| --- | --- |
| * | ***進化を享受しよう** |

*ビジネスでも、プライベートでも、あらゆる場面で今やデジタル データなしには過ごすことが不可能になりつつあります。しかし、これでゴールというわけではありません。人は、「今よりも、より心豊かな、より安全な、より便利な、より快適な毎日を手に入れたい」とクオリティ オブ ライフを求め続けています。この生存のための競争に常にさらされている企業では、差別化を求めて、より速く、より安く、よりスマートにビジネスを進めていくことを考え続けなければなりません。ニーズは時々刻々と変わります。つまり、変化の時代こそ“進化”を迅速に享受することが不可欠です。IT フィールドで、今、最も進化を象徴するメジャーなプロダクトといえば、SQL Server 2005。いよいよ 11 月 17 日に発表される新世代 RDBMS が成し遂げたことはなんでしょう。

|  |  |
| --- | --- |
| * | ***大規模基幹システムの中核を担える最速 OLTP エンジン** |

データベースは、まちがいなく情報システムのエンジンということができます。ビジネスの基幹を担うオンライン トランザクション分野で利用する場合、データベースの処理スピードが、ビジネスの収益そのものに大きな影響を与えるといっても過言ではありません。RDBMS は 1 にも 2 にも、速くなくてはならないのです。  
  
SQL Server 2005 は、Oracle Database 10g との比較において、RDBMS の基本性能を測る TPC-C ベンチマークで 7%、TPC-H ベンチマークにおいては 38% と、そのいずれのテストでも Oracle のパフォーマンスを上回っていました。  
  
しかし、企業において IT 構築が聖域ではなくなった今日では、そのパフォーマンスもコストを度外視しては評価されません。企業では、すべてのプロジェクトで費用対効果が厳しく測られ、いくら高いパフォーマンスが期待できても、多額のコストがかかるようでは投資するわけにはいかないのです。  
  
その点、SQL Server 2005 はコスト パフォーマンスにも秀でており、前述のベンチマーク テストでも、Oracle と比較して TPC-C で 37%、TPC-H で 20% のコスト削減効果が得られました。  
  
さらに、SQL Server 2005 は、64 ビット サポートをネイティブで実現。これにより、ミッション クリティカルなオンライン トランザクションや巨大なデータ ウェアハウスなどが、高いスケーラビリティを確保できるようになり、シングル スレッド処理においては、最速の CPU パフォーマンスを発揮することが可能です。  
  
情報システムのエンジンはまた、ノンストップで走り続ける高い可用性が求められています。これに対応する機能として、SQL Server 2005 では、障害復旧機能が強化されています。具体的には、データベース ミラーリング、フェール オーバー クラスタリングの新機能で、たとえば、離れた拠点にバックアップ サーバーとしてミラーリング サーバーを置き、本番サーバーでデータがコミットされると、それと同じタイミングで、あるいはタイミングをずらしてバックアップ サーバーでもデータが更新されます。本番サーバーにトラブルが発生した場合、通常数秒以内にミラーリング サーバーが処理を自動的にフェール オーバーし、データ ロスは発生しません。本番サーバーが正常復帰したのちは、2 つのサーバーの間で、こちらも自動的に同期化が行われます。インデックス作成やデフラグ、高速リカバリなどの操作を、データベースをオフラインにすることなくオンラインのまま行えるようにもなっており、99.999%、いわゆるナイン ファイブの可用性、100％に限りなく近い可用性の実現も可能になっています。実際、周辺機器メーカーである Western Digital では、約 900 の表領域を持つ基幹システムをノンストップで、また、通信業の Verizon は 12,000 の同時ユーザー、5 テラバイトのデータ ボリュームを持つ請求管理システムを、99.999% の可用性で運用しています。このほかにもミッション クリティカルで大規模な基幹システムの中核を SQL Server が担っているケースは数多くあり、SQL Server がハードウェア サーバーと同時に出荷されるユニット シェア分野で、Windows プラットフォームにおける No.1 シェアを獲得しているという IDC の調査もうなずくことができます。

|  |  |
| --- | --- |
| * | ***セキュリティこそ生命線** |

*世界がネットワークで結ばれ、ビジネスがシステム上で行われるようになって、人はたくさんの恩恵を被るようになってきています。しかし、その副作用というべきか、1 つのやっかいな問題に悩まされるようにもなりました。それがセキュリティの確保です。中でも最も守らなければならないものがデータです。  
  
企業にとって重要な財産であるデータは、むやみに閲覧されたり、盗まれたり、改ざんされたりするわけにはいきません。RDBMS は最後の砦、これ自体が極めて堅牢なデータの格納庫でなければならないのです。  
  
SQL Server 2005 は新製品だけあって、最新のセキュリティ ニーズがきちんと反映されています。たとえば、一番大きなところでは、ネイティブでデータの暗号化/復号化機能を搭載したことです。しかも、その技術は、対称鍵と非対称鍵および認証、の 2 種類の方策を搭載しており、データの重要性に合わせて選択できるようになっています。これまで、その必要性は認識しながらも、コストをかけてまでデータの暗号化を行うべきか迷っていた企業にとって、これは朗報といえるでしょう。RDBMS の基本機能として、SQL Server 2005 にはそれが既に実装されているのだから、後はもう実行するだけでよいのです。  
  
また昨今では、RDBMS のユーザー管理についても留意しなければなりません。公開された事件を見ると、犯人は社外ではなく社内、あるいはその企業や関係会社にかかわる人間だった、というケースがかなり多いといえます。異動、転勤、退社など、人のリアルな動きに合わせて、迅速に権限管理やユーザー管理が行えないと、データの安全は確保できません。ここで効いてくるのが、SQL Server 2005 のユーザーとスキーマの分離、ユーザーとスキーマに対する新しいデータ定義用語 (DDL) の追加といった新しいルールです。  
  
従来では、ユーザー A に所有されていた表領域やファンクションやビュー、ストアド プロシージャなどのオブジェクトは、ユーザー A が削除されると、一緒に削除されて、ユーザー B がそれらを利用することはできませんでした。しかし、この製品ではこうしたオブジェクトはスキーマによって所有され、そのスキーマがユーザーに対して所有されるという形になるため、ユーザー A が削除されても、ユーザー B の利用にはまったく影響することはありません。ユーザーを削除する DROP USER という DDL を実行しても、それはアプリケーションの書き直しを意味しないという点では、開発者にとってうれしい利点といえるでしょう。

|  |  |
| --- | --- |
| * | ***これからの RDBMS は Web サービス レディでなければ** |

*冒頭でも述べたように、今日は、ビジネス活動においても、個人生活においても、ニーズは変化し続けています。このスピードにキャッチアップしていくためには、閉じたシステムで常にスクラッチから開発をスタートさせていたのでは到底間に合いません。最初から他のシステムとのつながりを重視したシステム構築を目指し、使えるプログラムがあれば、それが外部のもの、内部のものにかかわらずどんどん利用していく必要があります。そのために、システムの連携性が最重要課題なのです。  
  
これを実現する中核の技術として注目を集めているのが、Web サービスです。なぜ、Web サービスなのでしょうか。これは、いわば再利用可能なソフトウェアの「小片」で、ネットワーク越しに、お互いが XML、SOAP など業界標準の技術でメッセージを交換できるのです。相互運用性が高く、異なる OS 上で動き、異なるプログラミング言語のソフトウェアの間でのやりとりも行えます。メッセージ ベースで、ソフトウェアどうしの結びつきがゆるやかで、自動化が可能であるために、真にサービス指向のシステムを形成できるというわけなのです。このシステム構造は、SOA (サービス指向アーキテクチャ) と呼ばれ、米国を筆頭に今や世界的に適用が進んでいますが、こうした潮流を受け止めるため、SQL Server 2005 では、XML をネイティブでサポートしています。ただデータベースに XML を保存できるばかりではなく、インデックスも作成でき、XML データに対してクエリを発行することも可能です。直接 XML データを扱えるために管理負荷が軽減でき、セッションをデータベース コネクションにまで伸ばせるようになります。もはや Web アプリケーション サーバーを介す必要はなくなるのです。SQL Server 2005 で真に Web サービス レディになったことで、データベースへのアクセス方法が広がったといえます。

|  |  |
| --- | --- |
| * | ***全社で活用可能な BI 基盤** |

*ビジネス上の決断には 2 つの種類があります。賢明な決断とそうでない決断。これは企業のどの階層のユーザーにも起こりうることで、これが積み重なって、結局は企業の命運を左右していくことになります。常に賢明な決断を下すために求められているのがビジネス インテリジェンス (以下、BI) であり、これが企業に必須のインフラとして認められつつあることは、もはや異論のないところでしょう。  
  
しかし、全社で必要であるからといって、そのための専用環境を構築するのはコストもかかるし、社員の教育も大変です。BI にはいくつかプロセスがありますが、部分最適ソリューションで、プロセスごとに異なるベンダの製品を導入すると、何かトラブルが生じたときに、問題の切り分けやサポートがやっかいになります。何より全社利用ともなれば、誰にでもわかりやすく、使いやすい環境であることが大前提になります。  
  
マイクロソフトは、SQL Server 2005 でこの課題に正面から取り組みました。ビジョンは、「すべてのユーザーに対して、より良く、より速く、より確かな意思決定を引き出すためのビジネス洞察力を提供することによって、組織の改善に寄与すること」。このビジョンを実現するため、BI プラットフォームとしての SQL Server 2005、エンド ユーザー ツールとしての Office、パフォーマンス管理ソリューションとしての Office Business Scorecard Manager 2005 によって、すべての意思決定者がより高いビジネス パフォーマンスを発揮できる優れた環境を提供しています。  
  
BI プラットフォームとしての SQL Server 2005 の大きな進化ポイントは、ETL (抽出、変換、および読み込み) ツールの役割を果たす Integration Services が導入されたことです。外部の業務システムや他のデータベースからデータをロードして、SQL Server に統合することが可能になり、データ ウェアハウスの構築が極めて容易になりました。  
  
ひとたび統合が完了すれば、そこからがデータ分析の出発点となりますが、データ マイニングが可能になった Analysis Services や、プレゼンテーション能力がさらに向上した Report Services がそれを強力に支援します。この 3 つのサービスは、ユーザー インターフェイスも統合されており、管理者やユーザーが、高いコスト パフォーマンスと少しの管理と利用負荷で、BI 環境をまさにインフラとして利用できるようになります。

|  |  |
| --- | --- |
| * | ***軽快なフットワークで動くには、開発生産性、運用管理性の高さがカギ** |

*メインフレームが主役の時代、企業の情報システム開発には余裕がありました。1 年間かけて計画をじっくり立て、次の 1 年で開発をしカット オーバーするといったことが普通に行われていましたが、今はそんな悠長なことはしていられません。優れたソフトウェアを開発する力があっても、そんなに時間がかかるようでは競争に勝てません。猫の目のように変わるニーズを横目でにらみながら、ビジネス チャンスを逃さないためには、数か月単位でのマーケット インを考える必要があります。つまり、開発生産性が高いことも、現代の企業情報システムの必須条件なのです。  
  
SQL Server 2005 では、Visual Studio 2005 および .NET Framework との密接な連携により、最大 40% の開発工期短縮を実現できます。このゴール達成に貢献する新たな機能として、共通言語ランタイム (以下、CLR) の SQL Server 2005 への統合があります。これにより .NET や Visual Basic、C# などのプログラミング言語を利用してデータベース サーバー上の処理を記述できるようになるため、コーディング作業の効率を高めることができます。  
  
また、新しい SQL Server にはサービス ブローカも搭載されました。この機能が実現するのは、信頼性の高いメッセージ キューイングです。これは、ユーザーの多くが要望した機能だったのです。データのバックアップ、リストア、リカバリなどの処理もカバーしており、これにより、メッセージ キューイングおよびデータベース内のデータに対する、統合されたプログラミング モデルが提供されます。さらに、サービス ブローカによって、SQL Server における非同期処理が可能になるため、データベース アプリケーションが他の処理をあるタイミングでスタートさせるようなこともできます。このようなメッセージ キューイングによるリクエスト処理を利用すれば、サーバーのリソース消費を抑えることもできます。従来と比べ最大60％まで迅速な分散アプリケーションの構築が可能だとマイクロソフトは見ています。  
  
スピーディに開発してすばやくマーケット インできても、その後の運用管理に手間がかかるようでは何にもなりませんが、この点でも SQL Server 2005 には、統合された管理ツール スイートでありユーザー インターフェイスに優れた、SQL Server Management Studio、Web Administrator などが搭載されています。

|  |  |
| --- | --- |
| * | ***一番新しい RDBMS が、ビジネスの新しい地平を開いてくれる** |

今日の企業情報システム開発におけるニーズと、それに応える SQL Server 2005 の新機能を駆け足で見てきました。その 1 つひとつを眺めるにつけ、つくづく時代は動いているとの感を強くします。そうであるなら歩調を合わせて歩きたい。昨日と同じように今日を生きることはたやすいが、それではいつまでたっても明日を迎える準備ができません。変えたくない、変えられない理由は、いくつも思いつくことができますが、ここはあえて、どうしたら 1 歩前へ踏み出すことができるかを考えてみるのはいかがでしょうか。その先には必ず、誰も見たことがない新しい風景が広がっているはずであり、それを享受できるのは先に発見した人の特権でもあります。



|  |  |  |
| --- | --- | --- |
| READY プライムタイム *   |  |  | | --- | --- | | **READY STATION だけの特別情報をお届けする「READY プライムタイム」。** * 他では知ることのできない情報やインタビュー、座談会など、毎回、SQL Server 2005 に関する特別な情報を掘り下げてお届けするプレミアム企画が READY プライムタイムです。今回は、2005 年 8 月に開催された Tech・Ed Yokohama でのマイクロソフト デベロッパー部門のトップ、S. Somasegar コーポレート バイス プレジデントによる、SQL Server 2005 と Visual Studio 2005、BizTalk Server 2006 に関する基調講演のダイジェスト版です。 | * |   * |

|  |  |  |  |
| --- | --- | --- | --- |
| * | |  | | --- | | **Tech・Ed 2005 Yokohama 基調講演 「Get Ready for 2005!」** | | |
|  | | * | * |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | ***Index**   |  |  | | --- | --- | | * | [マイクロソフトの次世代アプリケーション プラットフォーム戦略](#t2_1) | | * | [高品質な Windows Server System の提供](#t2_2) | | * | [生産性の大幅な向上に貢献する Visual Studio 2005 と SQL Server 2005](#t2_3) | | * | [データ統合、データ分析、レポーティングを包含する SQL Server 2005](#t2_4) | | * | [最適な運用環境を実現する SQL Server 2005](#t2_5) | | * | [高度なセキュリティ](#t2_6) | | * | [卓越したスケーラビリティ](#t2_7) | | * | [今後のロードマップ](#t2_8) | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | * | * **S. Somasegar**  マイクロソフト コーポレーション デベロッパー部門 コーポレート バイス プレジデント * | * | * S. Somasegar * | * | |  |  | | --- | | * | | | | * |

*

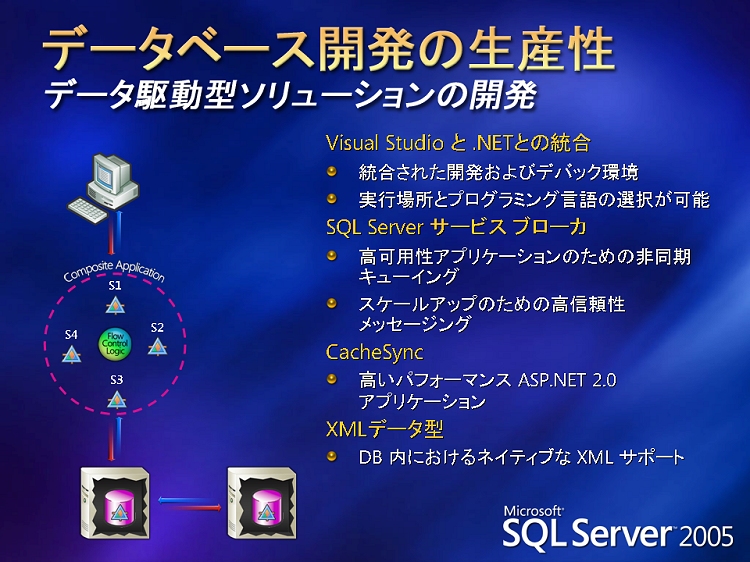
|  |  |
| --- | --- |
| * | ***マイクロソフトの次世代アプリケーション プラットフォーム戦略** |

*Tech・Ed 2005 Yokohama にお越しいただきありがとうございました。  
本日、私は、マイクロソフトが実現する「Connected System」をテーマに話をしたいと思います。  
現在は、サービス指向アーキテクチャ (SOA) の時代です。Visual Studio および .NET Framework を使用し、管理コードおよび Web サービスを使用して、業界標準システムと相互運用可能なサービス指向アーキテクチャ アプリケーションの構築が簡単にできます。  
5 年前の 2000 年、マイクロソフトは同時に、異なる 3 つの旅に出ました。1 番目は、インターネット向けの新たなビジョン、HTML を超えた、Web ブラウザを超えたインターネットのプログラミングです。2 番目は、インターネットでの分散型開発の新規モデルで、Web サービスと呼ばれている XML 方式のメッセージングを使用するものです。3 番目は、技術を 1 世代シフトしようという試み、アプリケーション プラットフォーム レベルと開発ツール レベル両方のシフトで .NET と呼んでいるものです。  
この 3 つの旅の方向が一緒になり、お客様における「Connected System」、およびアプリケーションの開発と展開をいかにして支援するか、というビジョンとなったのです。過去 5 年間の私たちの成果をみると、この 3 つの旅はすべてにおいてすばらしい進歩を遂げていることがわかります。  
**

|  |  |
| --- | --- |
| * | ***高品質な Windows Server System の提供** |

さて、プラットフォームという観点から見た「Connected System」をマイクロソフトが具現化したものが、Windows Server System です。Windows Server を中心に、マイクロソフトが提供し広く普及しているさまざまなサーバー アプリケーション製品を組み合わせて Windows Server System が構築されています。  
今、マイクロソフトが開発中の製品群、SQL Server 2005、Visual Studio 2005、BizTalk Server 2006、Windows Server 2003 R2 を皆様が組み合わせて導入されることにより、私たちが提供するビジョン、「Connected System」を約束できるものと信じています。つまり、これらの製品の導入により、皆様のデータを管理し、アプリケーションの展開を迅速に行い、最終的にミッション クリティカルなアプリケーションを実行するインフラストラクチャを確実に展開することができるということです。  
これから各製品の詳細に入りたいと思います。次世代製品がどのように、約束した「Connected System」を提供していくのかを話します。

|  |  |
| --- | --- |
| * | ***生産性の大幅な向上に貢献する Visual Studio 2005 と SQL Server 2005** |



まず、生産性に関してですが、Visual Studio 2005 および SQL Server 2005 では、開発者の生産性向上を目指してきました。私たちは、共通言語ランタイム (CLR) を SQL に統合しましたが、これにより .NET 言語、VB や C# などを使って SQL Server のサーバー サイドのコードを作成できるようになりました。  
次はサービス ブローカです。現在のインターネットでは同期処理だけでは不十分で、非同期処理の実行と非同期メッセージの送信も必要になります。SQL Server 2005 ではサービス ブローカ機能を導入し、メッセージの非同期送信を容易にしました。たとえば、皆様がインベントリ システムをアップデートしながら、在庫システムに非同期でメッセージを送りたいとしましょう。この場合、SQL Server 2005 のサービス ブローカ機能を使えば非常に簡単に実行できます。  
さらに、SQL Server では、XML データ型にネイティブで対応しているため、データベースに XML を保存すればよいばかりでなく、XML データに対して豊富なクエリを実行することもできます。

|  |  |
| --- | --- |
| * | ***データ統合、データ分析、レポーティングを包含する SQL Server 2005** |

*



ここからは、ビジネスへの深い洞察力ということで、ビジネスにおいて意思決定が正しく行われるようにすることを考えてみます。  
ビジネスへの深い洞察力に関しては、多くの方々が心を躍らせるようなことが 3 つあります。データについて言いますと、皆様は、データをどこに保存していようと、データを統合したいと考えています。データが SQL Server に入っていても、Oracle に入っていても、フラット ファイルとして保存されていても、すべてのデータを統合したいわけです。これは Integration Services ということになります。一度データを統合すれば、データの分析、つまり Analysis Services が必要になります。そして、リッチ クエリ ダンプを受け取ってデータを理解できれば、それをレポート化する、いわゆる Report Services が必要になるわけです。SQL Server 2005 では、Integration Services、Analysis Services、Report Services を共通 UI の下で統合し、データに対して深い洞察力を確保し、正しい意思決定ができるようになっています。  
**

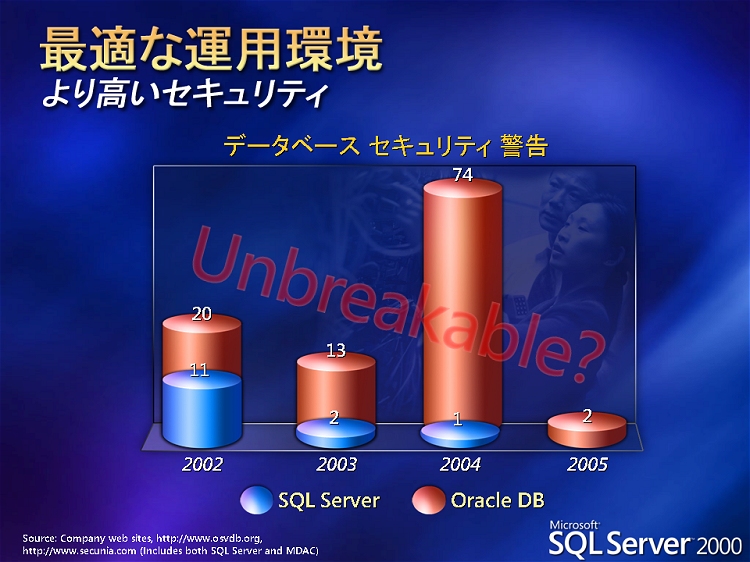
|  |  |
| --- | --- |
| * | ***最適な運用環境を実現する SQL Server 2005** |

*



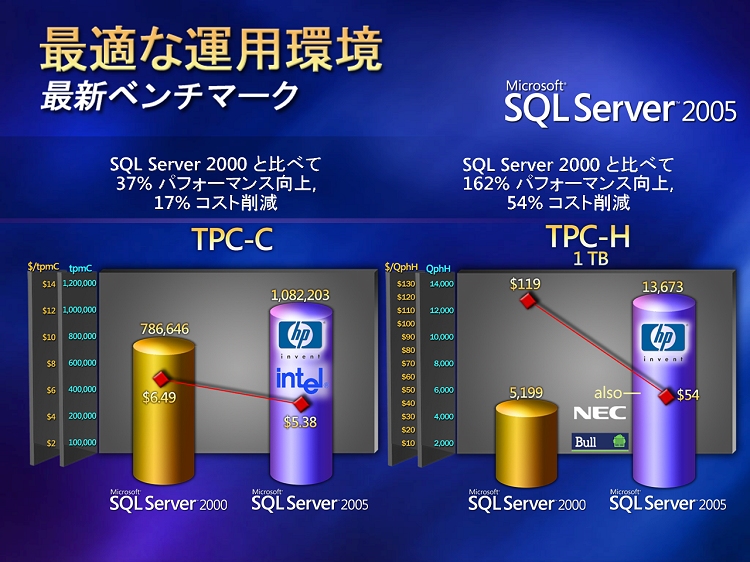
これまでは、開発者の生産性、そしてデータに対するより深いビジネス洞察力について話してまいりました。ここで、焦点を変えて、最適な運用環境について話をします。  
私たちは、長年にわたって可用性について話題にしてきており、フェールオーバー クラスタリングにも対応してきました。SQL Server 2005 では、さらにデータベース ミラーリングの新たな機能を追加し、障害復旧用に異なる地域に大きなバックアップ サーバーを設定することができるようになりました。そして、プライマリ データ サーバーにコミットするたびに、それをバックアップ サーバーに同期か、もしくは非同期でバックアップするかを選択することができます。この 2 つの機能により Windows プラットフォームで最高の可用性を確保することができます。  
SQL Server 2005 には、他にも多くの機能が追加されています。以前はデータベースをオフラインにしなければならなかった操作も、オンラインの状態でも実行できます。インデックスの作成、デフラグやインデックス作業、高速リカバリの操作が、オンラインのままでも実行できるようになっています。このように、可用性について、多くの改善を図っています。

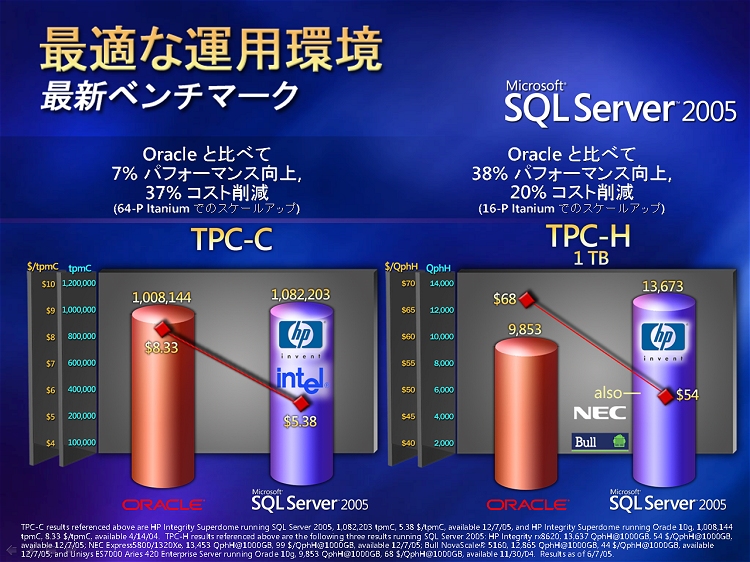
|  |  |
| --- | --- |
| * | ***高度なセキュリティ** |

*

最適な運用環境についての話は、セキュリティ抜きではできません。  
この数年間で、私たちはセキュリティを大幅に改善してきました。ツール、開発方法論、技術者のトレーニングなど、セキュリティに注目し、セキュアな製品を供給できるようになったのです。2003 年以降、SQL Server 製品には、重大なセキュリティの脆弱性は 1 つだけでした。  
セキュリティでは、開発プロセスやツールを変えるだけではなく、製品も変える必要が出てきます。私たちは、ネイティブ データ暗号化、認証管理、ポリシー機能など、たくさんのデータ暗号化機能を追加し、セキュアな SQL 構成の実行を容易にしてきました。  
また、攻撃を大幅に削減するような方法でデフォルト設定を変更し、デフォルトではサービスの多くをオフにしておき、SQL 管理者がログインして必要なサービスだけをオンにできるようにしてあります。  
*

|  |  |
| --- | --- |
| * | ***卓越したスケーラビリティ** |

*



最適な運用環境ということで、もう 1 つ話をしたいことは、スケーラビリティです。TPC-C ベンチマークと TPC-H ベンチマークを実施しましたが、パフォーマンス向上、コスト低減の両方の点で、SQL Server 2005 の方が SQL Server 2000 より優れていると言えます。競合他社と比較したらどうなのかと思われる方もいらっしゃるかも知れません。Oracle と比較したデータでは、TPC-C ベンチマークおよび TPC-H ベンチマークの双方とも、SQL Server 2005 の方が優れていることがおわかりいただけるかと思います。つまり、優れたパフォーマンスを、低コストで実現しているということが言えます。SQL Server 2005 でチームが達成したこの成果を、たいへん誇りに思っていますし、今年の末までにこれらのメリットすべてを皆様に提供したいと、ワクワクしております。

|  |  |
| --- | --- |
| * | ***今後のロードマップ** |

皆様のカレンダーの 11 月 17 日に印を付けておいてください。この日、SQL Server 2005、Visual Studio 2005、BizTalk Server 2006 が、東京で発表になります。私も、今からたいへん心が躍っています。なぜなら、皆様にこれらの製品を通じて、約束した「Connected System」というビジョンを提供できるからです。  
ぜひ、皆様がこの日を心待ちにしてくださることを願っています。  
長時間にわたってありがとうございました。



  
*

|  |
| --- |
| **Microsoft MVP が語る SQL Server 2005** * 2006 年 2 月 1 日、エンタープライズ ユーザー待望の SQL Server 2005 が発売されました。約 5 年ぶりのバージョン アップとなる SQL Server 2005 では、SQL Server 2000 の優れた機能はそのままに、非常に多くの新機能が追加されています。その新機能の数は、実に 1,000 以上。その数だけでも SQL Server 2005 がいかに大きな進化を遂げたかが容易に想像できるのではないでしょうか。この特集では、SQL Server 2005 を導入することで、ユーザー企業では何がどのように変わるのか、そしてどのようなメリットがあるのかを解説していきます。 |

*[Microsoft MVP とは](#c8_7)



|  |
| --- |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| * | |  |  |  | | --- | --- | --- | | **MVP スペシャル**  **ついに発売! SQL Server 2005 はココがスゴイ!**  **（Ready編集局　松本美穂・松本崇博）** * | * |  | | |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | **Index**   |  |  | | --- | --- | | * | [既に多数の導入実績、品質の追求](#c8_1) | | * | [パフォーマンスとスケーラビリティの向上　他を圧倒するパフォーマンス、価格性能比](#c8_2) | | * | [パフォーマンスとスケーラビリティの向上に貢献する具体的な機能](#c8_3) | | * | [セキュリティの大幅な強化](#c8_4) | | * | [可用性を向上させる機能　データベース ミラーリング、オンライン操作、クラスタ機能の強化](#c8_5) | | * | [その他の機能　Oracle との機能比較、高度な BI 機能を標準搭載](#c8_7) | | * |  |  | | --- | | * | | | * |   *   |  | | --- | | **大規模かつミッション クリティカルなエンタープライズ システムにも対応 パフォーマンスとスケーラビリティの向上 セキュリティの強化、可用性と信頼性のさらなる向上** |   *SQL Server 2005 の 1 番の魅力は、何といってもエンタープライズ向け機能の充実にあります。たとえば、パフォーマンスとスケーラビリティの向上に貢献する機能には、「データベース エンジンの強化」「インデックスの進化」「スナップショット分離レベル」「64 ビット対応」「データ パーティション」など、ざっと挙げただけでもこれだけあります。そして、「データの暗号化」や「デフォルト セキュア」によるセキュリティの強化、「データベース ミラーリング」による高可用性の実現などにより、SQL Server 2005 は、大規模かつミッション クリティカルなエンタープライズ システムにも十分耐え得るプラットフォームへと大きく進化しています。   |  |  | | --- | --- | | * | **既に多数の導入実績 品質の追求** | | **金融や証券、大規模システムで多数の導入実績** | |   * SQL Server 2005 は、既に NASDAQ や Barclays Capital、名古屋銀行、Mediterranean Shipping Company、Barnes and Noble など、世界中の 100 社以上の企業によって製品出荷前の早い段階から導入されています。NASDAQ では 1 秒あたり 5,000 件のトランザクションを処理し、世界第 2 位のコンテナ船企業である Mediterranean Shipping Company では 1 日 5 億件のトランザクションを処理し、データベース サイズはなんと 5 TB、また、Barnes and Noble では 2 TB のデータ ウェアハウス システムの運用など、SQL Server 2005 は、金融や証券などのミッション クリティカルなシステムから、TB (テラバイト) クラスの大規模なエンタープライズ システムにまで幅広く活用されています。 一方、日本のユーザーでも、160 万以上の顧客口座データ管理に活用している名古屋銀行だけでなく、株式会社ジェイティービーや新日鐵住金ステンレス株式会社、メルシャン株式会社など、大規模なシステム環境で多くの導入実績があります。そのうちのいくつかは、導入事例記事として下の URL のサイトにまとめられており、導入にあたっては大いに参考になると思いますので、1 度ご覧ください。 もちろん、マイクロソフト社自身でも、既に社内の 100 以上のビジネス アプリケーションの SQL Server 2005 へのアップグレードを完了しています (そのうちのいくつはテラバイト級のデータベースです)。   |  | | --- | | **品質の追求** |   * 新しいバージョンだと、少しようすを見てから導入したいと考える企業もあるかと思います。しかし、SQL Server 2005 は、製品出荷前の品質テストに非常に多くの時間を割いているので、すぐにでも安心して導入できます。その SQL Server 2005 の品質テストでは、SQL Server 2000 よりも 40 倍以上もの時間を使い、また負荷ワークロードも 10 倍に増やして負荷テストを実施し、延べで 150,800 マシン時間を費やしています (SQL Server 2000 のときは 3,900 マシン時間)。 当初 SQL Server 2005 は、2003 年に出荷される予定でしたが、負荷テストの強化やセキュリティ機能の大幅な見直しなど、さらなる品質向上のために十分な時間を掛け、5 年以上を要して完成させています。納得のいくまでテストを行い、時間をかけて完成された製品は、発売開始時点から安心感が断然違います。   |  |  | | --- | --- | | * | ***パフォーマンスとスケーラビリティの向上 他を圧倒するパフォーマンス、価格性能比** |  |  | | --- | | **SQL Server 2000 を超えるパフォーマンス** |   SQL Server 2005 は、データベース エンジンの強化やインデックスの進化、スナップショット分離レベル、64 ビット対応などにより、パフォーマンスが大幅に向上しています。次の図は、SQL Server 2000 と SQL Server 2005 のパフォーマンスを比較したグラフです。  * SQL Server 2000 と SQL Server 2005 のパフォーマンス比較   |  | | --- | |  |   http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig01.jpg*このグラフにある TPC ベンチマークは、データベース システムのパフォーマンス (性能や価格性能比) を客観的に評価する指標として利用されているもので、結果は [TPC の Web サイト](http://www.tpc.org/) * から参照できます。TPC-C ベンチマークは OLTP (オンライン トランザクション処理) システムを、TPC-H はデータ ウェアハウス システムを評価するためのベンチマークです。この図から TPC-C ベンチマークでは SQL Server 2000 よりも SQL Server 2005 のほうが 37% のパフォーマンス向上、TPC-H ベンチマークでは 162% のパフォーマンス向上を実現していることが確認できます (2005 年 8 月時点)。 *   |  | | --- | | **他を圧倒するパフォーマンス** |   http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig02.jpg 次の図は、TPC-C および TPC-H のベンチマーク結果 (2005 年 12 月時点) を、もう少し細かくグラフ化したものです。 SQL Server 2005 の優れたパフォーマンス  TPC-C ベンチマークでは、1 CPU 構成で 1 秒あたり 38,000 件のトランザクションを処理し、16 CPU 構成で 492,000 件のトランザクションを処理しています。これはテストしたすべてのサーバー プラットフォームにおいて No.1 のパフォーマンスです。また、64 CPU 構成では 1 秒あたり 120 万件のトランザクションを達成しており、これは Windows プラットフォームにhttp://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig03.jpgおける No.1 のパフォーマンスです。SQL Server は、Windows プラットフォームのみに最適化されており、Windows OS との親和性の高さを確認できるのではないでしょうか。 また、TPC-H ベンチマークでは、100 GB のデータベース サイズにおいてすべてのプラットフォームで No.1 のパフォーマンス、1 TB のサイズでは価格性能比 (Price / Performance) で No.1、3 TB のサイズでは 32 CPU 構成の中で No.1 のパフォーマンスを叩き出しています。1 TB での価格性能比については、次の図でもう少し詳しく見ていきましょう。 *  TPC-H ベンチマーク 1 TB での Price / Performance のトップ 10  このランキングは、[TPC の Web ページ](http://www.tpc.org/tpch/results/tpch_price_perf_results.asp) * からの引用になりますが、1 位、2 位、4 位と SQL Server 2005 がランクされ、SQL Server 2005 が他のデータベースと比べて圧倒的な価格性能比であることが確認できます (2006 年 2 月時点)。また、第 1 位と 9 位の結果を見比べてみてください。これはどちらも Windows Server 2003、16 CPU、64 GB メモリとほとんど同じ構成で、データベース ソフトウェアが異なります。この結果からパフォーマンス (QphH: 1 時間あたりのクエリ実行数) および価格性能比 (Price / QphH) において、SQL Server 2005 が優れていることを確認できるのではないでしょうか。*   |  |  | | --- | --- | | * | ***パフォーマンスとスケーラビリティの向上に貢献する具体的な機能** |   * ここではパフォーマンスとスケーラビリティ (拡張性) の向上に貢献する新機能について、具体的に1つずつ解説していきます。 *   |  | | --- | | **データベース エンジンの強化** |   * SQL Server 2005 のデータベース エンジンでは、主に次のような強化が施されています。  ・メモリ管理と CPU スレッド処理の強化 ・NUMA アーキテクチャへの対応 ・先行読み取り (Read Ahead) の強化 ・クエリ オプティマイザの進化 ・再コンパイル時の統計の非同期更新 ・Plan Guide 機能の追加  SQL Server 2005 は、SQL Server 2000 よりも効率的にメモリ、CPU および I/O 処理を行えるようになり、クエリ オプティマイザはより最適な実行プランを予測できるようになりました。また、従来は実行プランの再コンパイル時に同時に統計が更新されていたので、再コンパイル時のクエリ パフォーマンスが低下するケースがあったのですが、SQL Server 2005 では非同期で統計が更新されるため、そのような現象は起こりません。 Plan Guide (プラン強制) 機能は、パッケージ製品などに組み込まれた SQL に対して、明示的に実行プランを与えることができます。従来は、パッケージ製品の SQL が非効率な実行プランで実行された場合にはチューニングの施しようがありませんでしたが、SQL Server 2005 では、この Plan Guide 機能によってチューニングしていくことができます。これは意外と役立つ機能です。 *   |  | | --- | | **インデックスの進化** |   SQL Server 2005 ではインデックス機能も強化されています。主なものとしては、次の 2 つがあります。  ・付加列インデックス ・オンライン インデックス再構築  付加列インデックスは、Include オプションとも呼ばれ、インデックスのリーフ レベルに指定した列を含める (Include する) 機能です。カバリング インデックス (複合インデックス) と似ていますが、違いは次の図のようになります。 *  カバリング インデックスと付加列インデックスの違い  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig04.jpg*カバリング インデックスは、インデックスのリーフ レベルだけでなく、ルート レベルと中間レベルにもカバリングに含めた列 (図では b 列) の値を格納しますが、付加列インデックスの場合はリーフ レベルにしか値を格納しません。これにより、インデックス サイズを小さくすることができます。インデックス サイズが小さくなれば、インデックスの Seek および Scan で読み取るページ数 (I/O 数) を少なくできるので、パフォーマンスが向上します。付加列インデックスは、カバリング インデックスの欠点を解消するために追加された機能なので、皆様にも、ぜひ活用していただきたいと思います。 「オンライン インデックス再構築」は、その名のとおり、インデックスの再構築をオンラインで行える機能です。従来のインデックス再構築は、オフラインでの再構築であったため、再構築中はユーザー アクセスがブロックされていました。これでは、24 時間フル稼動するシステムではなかなか再構築を実行することができません。これを解消するものとして、DBCC INDEXDEFRAG というオンラインでインデックスを再編成 (リーフ レベルの断片化を解消) する機能もありますが、再編成では物理的な断片化を解消することはできませんでした。しかし、SQL Server 2005 からはオンラインでのインデックス再構築が可能になるので、このような心配は不要となります。 このように SQL Server 2005 では、できる限り SQL Server を止めない運用を可能とする機能 (可用性を高める機能) が多く追加されています。 *   |  | | --- | | **スナップショット分離レベル** |   http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig05.jpg* スナップショット分離レベルは、Oracle の経験者であればおなじみの「読み取り一貫性」を提供する機能です。これにより、排他ロックのかかっている更新中のデータでも、排他ロックにブロックされることなくデータを読み取れるようになります。具体的な動作イメージは、次の図のようになります。 *スナップショット分離レベルの動作  更新中のデータを読み取る場合は、“更新前”のデータ (その時点での正しいデータ) を参照することで、排他ロックにブロックされない読み取りが可能になります。これは、ロック待ちが原因でパフォーマンスが低下している場合には、大きな威力を発揮する機能です。実際、NASDAQ や名古屋銀行では、この機能を活用することでパフォーマンスを向上させていると話されています。 もちろん、どんな状況でもスナップショット分離レベルを利用すべきというわけではありません。スナップショット分離レベルでは更新前データを tempdb データベース内へ行単位で格納するため、行サイズが大きく、かつ更新頻度の高いデータベースの場合には tempdb への I/O がボトルネックになる可能性があります。したがって、両者のトレード オフに注意しながらうまく活用していく必要があります。 *   |  | | --- | | **64 ビット対応** |  * SQL Server 2000 では、IA64 (Intel Itanium ベースの 64 ビット プロセッサ) に対応していますが、SQL Server 2005 からは IA64 対応はもちろんのこと、X64 (Intel EM64T および AMD64) にも対応しています。64 ビット対応による 1 番の恩恵は、SQL Server の自動メモリ チューニングが完全に活かされるという点です。従来の 32 ビット環境では、4 GB のメモリ空間の壁を超えるために AWE (Address Windowing Extensions) 機能を利用していましたが、これにはオーバーヘッドがあります。また、AWE ではバッファ キャッシュ領域を広げることはできても、その他の SQL Server が利用しているメモリ領域 (Workspace、Lock、Connection、Procedure Cache 用メモリなど) は、4 GB の壁を超えられませんでした。しかし、64 ビット環境であれば、4 GB 以上のメモリ空間を自由に利用することができます。 *   32 ビット AWE と 64 ビットの違い  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig06.jpg*このように、64 ビット環境では、SQL Server の自動メモリ チューニングが完全に活かされる形になるのでパフォーマンスが大きく向上します。特に Sort や Hash といった内部動作で利用される Workspace メモリ (一時的な作業領域) が制限されないのは大きなポイントです。これにより、大量データの並べ替えやグループ化、結合演算のパフォーマンスが大幅に向上します。 *   |  | | --- | | **データ パーティション** |   データ パーティションは、Oracle でいうところの「レンジ パーティション」に相当する機能です。これはパフォーマンスとスケーラビリティ、運用管理性の向上にたいへん役立つ機能です。データ パーティションでは、データの範囲によってデータの格納場所 (パーティション) を分けることができます。たとえば、次のように日付の範囲 (月単位や年単位) で分けることができます。  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig07.jpgデータ パーティションによるデータの格納  データ パーティション機能の 1 番のメリットは、不要になったパーティションを簡単に切り離せる点にあります。たとえば、売上データの過去 3 年分は保存し、3 年以上経過したデータは削除する場合でも、そのパーティションを切り離すだけで削除が完了します。このメリットは、データ パーティションを利用しなかった場合のデータ削除時の動作を考えると、理解しやすいと思います。次に、データ パーティションを利用しない場合のデータ削除の例を示します。 http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig08.jpg  データ パーティションを利用しない場合は、データの削除時にそれぞれのインデックス内のデータも (内部的に) 削除されます。削除するデータが 10 万行で、インデックスが 3 つある場合には、30 万行分のデータをインデックスから削除しなければなりません。これでは、トランザクション ログへ 40 万件分 (実際のデータ 10 万件分とインデックスから 30 万件分) のログ レコードが書き出されることになり、ログ サイズは膨れ上がり、トランザクションの完了に時間がかかることになります。また、削除時には対象データに排他ロックをかける必要があるので、40 万個分の排他ロックもかけなければなりません。しかも、40 万行分すべてのデータの削除が完了するまで、かけられたロックは解放さないのです。したがって、データ パーティションを利用しない場合の大量のデータ削除は、パフォーマンスが大きく低下することになります。一方、データ パーティションであれば、削除したいパーティションを切り離すだけで、削除は一瞬で完了します。 データ パーティションのメリットは、他にもあります。パーティション単位でインデックスを作成、再構築できる点や、パーティション単位でバックアップと復元が行える点などです。データ パーティションは、大規模環境では欠かせない便利な機能です。皆様も、ぜひ 1 度試してみていただければと思います。 *   |  |  |  | | --- | --- | --- | | * | ***セキュリティの大幅な強化** | | | ***コードの大幅な見直し** | |   * SQL Server 2005 は、セキュリティも大幅に強化されています。ここ数年マイクロソフトでは、Trustworthy (信頼できるコンピューティング) の考え方を根幹にしています。SQL Server は、2002 年にワームの問題が出ましたが、そのとき数百人体制で臨んでいた SQL Server 2005 の開発をいったんストップさせ、全員に 2 ～ 3 か月の間、セキュアなコードを書くトレーニングを課しています。そして、すべてのコードを見直し、より安全性を高めるために作り直しをしているのです。 また、早期導入企業からのフィードバックを得ながら製品をブラッシュアップし、その結果に対してフィードバックを挙げた顧客からサインを得るまでは正式版をリリースしていません。後述するデータベース ミラーリング機能を製品出荷から外して、サービスパックで提供することになったのはこの理由からです。不完全な状態で出荷するのではなく、完全に仕上げた状態 (フィードバックを挙げた顧客が納得した状態) で初めて正式版をリリースする、より品質の高い製品をリリースするという姿勢には、本当に安心感があります。 *   |  | | --- | | **セキュリティを向上させる機能** |   * セキュリティを向上させる機能には、主に次の 3 つがあります。  ・データの暗号化 ・パスワード ポリシー ・デフォルト セキュア * **◎データの暗号化** データの暗号化は、その名のとおり、データを暗号化する機能です。サポートしている暗号化のアルゴリズムは、対称鍵暗号化 (DES、3DES、DESX、RC シリーズ、AES シリーズ)、非対称鍵暗号化 (RSA 512 / 1024 / 2048 bit) です。次の画面は、暗号化しているときのようすです。 *  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig09.jpg* データの暗号化  もし、データを暗号化していない場合には、データベース ファイル (.mdf) やバックアップ ファイルを盗難、盗聴された際に機密データを容易に参照できてしまいます。したがって、顧客のパスワードやクレジット カード情報などの重要なデータは暗号化しておき、万一の盗難に備えておくことが重要です。 * **◎パスワード ポリシー** パスワード ポリシーは、SQL Server 認証用のログイン アカウントに対して、Windows OS のセキュリティ ポリシーを適用できる機能です。 *  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig10.jpg* Windows OS のセキュリティ ポリシー設定  従来、セキュリティ ポリシーは、Windows 認証用のログイン アカウントに対してのみ適用することができましたが、SQL Server 2005 からは、SQL Server 認証用のログイン アカウントに対しても適用できるようになります。これにより、sa アカウントに対してもパスワードの有効期間を設定したり、ロックアウト (パスワードを何回か間違ったアカウントを使用不能にする機能) を設定できるようになります。 * **◎デフォルト セキュア** SQL Server 2005 は、インストール直後はセキュリティが最も厳しい状態でセットアップされています。他のマイクロソフト製品と同様、デフォルト セキュア (Secure By Default) になっており、既定では安全性が高く設定され、多くの機能が無効になっています。必要に応じて、利用したい機能だけを有効にしていくことで、安全に運用することができます。 有効/無効の変更は、「セキュリティ構成」ツールを利用して GUI ベースで次のように行えます。 *  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig11.jpg* なお、デフォルトで無効になっている機能は、「リモート接続」「SQLCLR (CLR 統合)」「xp\_cmdshell」「サービス ブローカ」「HTTP エンドポイント」などです。リモート接続が無効の場合は、SQL Server 2005 へネットワークを介してアクセスすることはできませんので、注意してください。 *   |  |  |  | | --- | --- | --- | | * | **可用性を向上させる機能 データベース ミラーリング、オンライン操作、クラスタ機能の強化** | | | ***高可用性機能** | |   * SQL Server 2005 は、可用性を高める機能も充実しています。その主なものは、次のとおりです。  ・データベース ミラーリング ・各種のオンライン操作 ・クラスタ機能の強化  データベース ミラーリングは、前述したようにサービス パックでリリースされることになりますが、わずか数秒で自動フェールオーバーを実現できるデータベースの複製機能です。クラスタよりも、フェールオーバー時間を大幅に短縮できることが 1 番のメリットです。 *  http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig12.jpg*  オンライン操作は、前述のオンライン インデックス再構築のように、SQL Server をオフラインにすることなくオンラインで実行できる機能です。SQL Server 2005 では、多くの機能がオンライン操作に対応しています。CPU アフィニティや I/O アフィニティの設定もオンラインで動的に変更することができ、SQL Server 2005 の再起動は必要ありません。 SQL Server 2005 は、クラスタ機能も強化されています。8 ノード クラスタへの対応や、データベース エンジン以外の機能 (Analysis Services、FullText Search、Notification Services など) がクラスタ対応になりました。また、Standard Edition でもクラスタ構成をサポートするようになったことも、うれしいところです (ただし 2 ノードまで)。 *   |  |  |  | | --- | --- | --- | | * | **その他の機能 Oracle との機能比較、高度な BI 機能を標準搭載** | | | ***Oracle との機能比較** | |   http://www.microsoft.com/japan/sqlserver/2005/readystation/07/images/08_cusp_fig13.jpg* SQL Server 2005 には、まだまだたくさんの新機能があります。次の表は SQL Server 2005 の主な新機能と Oracle の機能を比較したものです。 *Oracle との主な機能の比較  SQL Server では標準で提供される機能が、Oracle だと Enterprise Edition のみで提供されたり、Enterprise Edition の有償オプションとして提供される機能が多くあります。特に SQL Server の場合は、高度なビジネス インテリジェンス機能 (OLAP、データ マイニング、ETL、Reporting Services) が標準搭載されているのは、うれしいところです。ビジネス インテリジェンス機能だけが使用目的でも、製品を購入する価値があるのではないでしょうか。 *   |  | | --- | | **Oracle からの移行を支援するツールも登場** |   * SQL Server 2005 には、Oracle からの移行 (マイグレーション) を支援する SSMA (SQL Server Migration Assistant for Oracle V2.0) という無償ツールも提供されています。SSMA を利用することによって、Oracle データベースを SQL Server 2005 へ移行することが容易にできます。テーブル データの移動はもちろん、データ型の自動変換、PL/SQL で書かれたストアド プロシージャの Transact-SQL への自動変換、Oracle 独自の関数も SQL Server 2005 用へ自動変換してくれます。これは本当に便利です。Oracle からの移行を考えている方にとっては、必須のツールといえるでしょう。 現在、SSMA の利用手順をステップ バイ ステップ形式で、画面ショット満載でわかりやすく解説した自習書もダウンロードできるようになっています。   |  | | --- | | **おわりに** |   * 今回は、SQL Server 2005 の機能の一部しか解説することができませんでしたが、今後も、リニューアルされたこのサイト上で、機能解説を続けていく予定です。ご期待ください。 *   |  |  | | --- | --- | | |  | | --- | | **Microsoft MVP とは** * Microsoft MVP (Most Valuable Professionals) プログラムは、マイクロソフトのさまざまなテクニカル コミュニティにおいて、技術力が高く、精力的でかつ顕著な活動をした方を表彰する世界的なプログラムです。このプログラムでは、オンライン、オフラインを問わず、マイクロソフト製品に関する貴重な知識や情報を共有することで、コミュニティの活性化に貢献する世界中のコミュニティ メンバを讃えるものです。 MVP の方々のバックグラウンドや専門分野はさまざまですが、掲示板やテクニカル ニュース グループへの積極的な回答やコミュニティの運営、ブログや Web サイトによる情報共有、著作、開発サポート、イベントや講習会での講演などを通じて活躍しています。 | |   * |

*

|  |
| --- |
| **Microsoft MVP が語る SQL Server 2005** * Microsoft MVP へのインタビューを通じて SQL Server 2005 の深層を探る READY クローズアップ。この連載では、SQL Server 2005 に対する期待や導入メリット、具体的な利用方法などについて、Microsoft MVP として表彰された方々へインタビューを行い、SQL Server 2005 に対する想いを語っていただきます。* *→*[Microsoft MVP とは](#c8_7) * |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| * | |  |  |  | | --- | --- | --- | | **MVP インタビュー 第 5 回**  **Microsoft SQL Server 2005 は、ココが違う MVP が見た SQL Server 2005 導入のポイント** | * | * | | |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | * | ***Index**   |  |  | | --- | --- | | * | [SQL Server は個人レベルで勉強しやすいのが 1 番の魅力　Developer Edition ならわずか 6,000 円程度で入手可能](#c7_1) | | * | [SQL Server 2005 のツールで SQL Server 2000 に対しても便利なツールがある　SQL Server 2005 に少しずつ慣れていくには最適](#c7_2) | | * | [SQL Server 2000 互換モードで動作させることも可能　プロファイラは、アップグレード時の動作確認ツールとしても利用できる](#c7_3) | | * | [SQL Server Migration Assistant (SSMA) で Oracle からの移行も簡単　PL/SQL を Transact-SQL へ自動変換。Oracle 独自の関数にも対応](#c7_4) | | * | [デフォルト セキュアでインストール直後はセキュリティが最も厳しい状態　リモート接続が無効なのは注意が必要](#c7_5) | | * | [SQL Server 2005 は、ほとんどの機能が XML をベースにしている点がうれしい](#c7_6) | | |  |  | | --- | --- | | * | * | | |  | **米田 慎一 氏** Microsoft MVP for SQL Server  米田慎一氏株式会社新来島どっく 技術設計本部  新来島どっくの設計部門にて、新日本製鐵の文書管理システム NSXPRES 300X を導入する際に、制限事項などを理解するために SQL Server を勉強する。 稼動後は、システム管理や障害対応といった運用面に従事。現在、PASSJ (SQL Server ユーザー グループ) のシステム管理分科会のボード リーダーとして活動中。 システム関連では、ドキュメント・マニュアルとの付き合い方が、1 番長持ちするスキルだと信じており、Books Online を重視する。 PASSJ Blog にて、SQL Server を中心に機能拡張や、競合システムの現状、比較について積極的に発言している。 | |  |  | | --- | | * | | | * |   *   |  | | --- | | **SQL Server 2005 のツールで SQL Server 2000 に対しても便利なツールがある SQL Server Migration Assistant (SSMA) で Oracle からの移行も簡単 デフォルト セキュアでのインストール後はセキュリティが最も厳しい状態** |   * MVP インタビューの第 5 回目は、DBA (Database Administrator: データベース管理者) 業務を得意としている株式会社新来島どっくの米田 慎一 (よねだ しんいち) 氏にお越しいただきました。米田さんは、SQL Server ユーザーのためのコミュニティ「PASSJ」でシステム管理分科会のボード リーダーとして活躍され、PASSJ のメーリング リストおよび掲示板上では、適切な解答を多数投稿し、SQL Server ユーザー同士の情報交換を積極的に行っています。   |  |  | | --- | --- | | * | ***SQL Server は個人レベルで勉強しやすいのが 1 番の魅力 Developer Edition ならわずか 6,000 円程度で入手可能** | | ***――** | **米田さんが SQL Server に携わることになったきっかけを教えていただけますか。** | | **米田** | 私は、新来島どっくという造船所の設計部門に勤めています。本業は設計関連のシステム業務になりますが、社内のファイル サーバーの管理を手伝ったり、SQL Server の運用管理のアドバイスをしたりと DBA (データベース管理者) 的な仕事もしています。SQL Server に携わることになったきっかけは、部内で文書管理システム NSXPRES 300X (新日本製鐵) を導入することになったときでした。当時の NSXPRES 300X は、エンジンに SQL Server 6.5 を利用していましたので、その制限事項などを理解するために SQL Server の勉強を始めました。 |  |  |  | | --- | --- | | **――** | **具体的にはどのような方法で SQL Server を勉強したのでしょうか。** | | **米田** | 当時 (SQL Server 6.5) は、SQL Server に関する書籍がほとんどなかったので、Books Online (SQL Server のヘルプ) を頼りに勉強しました。Books Online は当時から非常に充実していたので、本当に助かりました。 SQL Server 2000 が登場してからは、Office 2000 Developer に SQL Server 2000 の Personal Developer Edition が付属するようになったので、それを自分のパソコンに入れて検証しながら学習しました。その後、SQL Server 6.5 から SQL Server 2000 へバージョン アップするという作業を手伝うことになったのですが、作業の 1 年くらい前から自主的に勉強を始めていましたので、特にトラブルもなくスムーズに移行することができました。 |  |  |  | | --- | --- | | **――** | **十分な準備をしておくことでスムーズな移行ができたわけですね。米田さんは、古くから SQL Server に携わってますが、SQL Server の魅力はどこにあると考えられますか。** | | **米田** | SQL Server は、個人レベルで勉強しやすいのが 1 番の魅力ですね。昔のイメージでいうと、Oracle が入っているようなシステムはうかつには触れませんでしたし、Oracle はソフトウェアの値段が高価すぎて個人ではとても購入できませんでした。その点 SQL Server であれば、Enterprise Edition と同じ機能を備えた Developer Edition が 6,000 円程度で購入できます。エンタープライズ レベルの機能を備えたデータベース サーバーが、解説書 2 冊程度の値段で入手できるというのは、個人で勉強する人にとっては本当にうれしいですね。机上の勉強だけでなく、実際に試しながら勉強できるので、どんどん自分のスキルを磨くことができます。SQL Server なら実際の環境からテスト環境へデータベースをコピーするのも簡単にできますし、バックアップやリストアを気軽に試せるのもありがたいですね。 |  |  |  | | --- | --- | | **――** | **確かに、普段データベースの管理をされている方でも、バックアップはしていても、1 度もリストアをしたことがない方がたまにいらっしゃいます。事前にリストアを試しておかないと、実際にトラブルが起きたときに慌てることになりかねませんね。** | | **米田** | そうですね。自分のパソコンで常日ごろ練習しておくことは重要です。その際には、現実的ではないバックアップの設計を試しておくこともお勧めの練習方法の 1 つです。たとえば、1 回だけフル バックアップを取っておいて、あとはトランザクション ログのバックアップだけを取るような状況などです。このような状況でも Enterprise Manager であればバックアップの履歴機能が備わっているので簡単にリストアすることができますが、他のサーバーへリストアしようとする場合はこうはいきません。1 つ 1 つのバックアップ (何十個にもなるトランザクション ログのバックアップ) を手動でリストアしていく必要があるわけです。このような実際よりも厳しい状況をシミュレーションしておくと、いざというときに慌てなくて済みます。 |  |  |  | | --- | --- | | **――** | **最近はサーバーも安価で入手できるようになって、昔に比べて環境を構築するのが楽になったのも大きいですね。** | | **米田** | x64 (64-bit) のサーバーも本当に安価になりました。私が先日購入したのは 3 万円弱の x64 マシンなのですが、現在なら 10 万円以下で入手できる x64 マシンがたくさんあります。また、最近は Virtual Server のように仮想マシン環境を使って練習するという方法もあります。社内の稼動中のシステムだと気軽に触れないので、安価なマシンや Virtual Server を使ってテスト環境を構築し、疑似体験をしておくのは必ず役立つと思います。特に会社では、苦手なことのために時間をとれるとは限りませんので、自分でテスト環境を作って自身の弱点を克服することが能力を広げるチャンスになります。 | | [ページのトップへ](http://www.microsoft.com/japan/sqlserver/2005/readystation/07/closeup.mspx#top)* | ***SQL Server 2005 のツールで SQL Server 2000 に対しても便利なツールがある SQL Server 2005 に少しずつ慣れていくには最適** |  |  |  | | --- | --- | | ** | Photo * |  |  |  | | --- | --- | | **――** | **いよいよ SQL Server 2005 日本語版が発売されましたが、SQL Server 2005 を導入するにあたってのポイントは、ズバリ何でしょうか。** | | **米田** | SQL Server 2000 のユーザー、特にデータベース管理者にお勧めしたい導入のポイントは、会社が導入するかどうかにかかわらず、SQL Server 2005 をできるだけ早く入手してインストールしておくことです。新しいバージョンが出れば、それに対する評価というのは、必ず上司から聞かれると思います。早期に SQL Server 2005 を検証しておけば、そのときに的確な発言をしたり、必要な提案をしたりできるようになります。そのうえで実際の移行作業をスムーズにこなせれば、実力も認められメリットは非常に大きいのです。 |  |  |  | | --- | --- | | **――** | **SQL Server 2005 をいち早く試しておくことが、スムーズな導入につながるというわけですね。** | | **米田** | はい。そのときのテスト方法としてお勧めなのが、SQL Server 2005 のツールを SQL Server 2000 に対して使ってみることです。私のブログでもまとめ始めているのですが、SQL Server 2005 のツールは、SQL Server 2000 に対しても便利なものがたくさんあります。 たとえば、SQL Server 2000 のプロファイラでキャプチャしたトレース データ (.trc) は、SQL Server 2005 のプロファイラで開くこともできます。SQL Server 2000 では、グループ化の機能に物足りない部分があって、履歴を確認したり、チューニング作業をする場合に少し不便だったのですが、SQL Server 2005 のプロファイラでは、グループ化機能がよくできているので、作業効率が上がります。 |  |  |  | | --- | --- | | **――** | **確かにプロファイラのグループ化機能は便利なので、それを SQL Server 2000 に対して利用する発想はおもしろいですね。SQL Server 2005 に少しずつ慣れていくという意味では最適だと思います。** | | **米田** | 他にも SQL Server 2005 の bcp ユーティリティでは、フォーマット ファイルを生成する機能が追加されています。bcp は、二重引用符で囲まれた文字列を扱いたい場合や、複雑なコピーをする場合には、フォーマット ファイルが欠かせませんので、これはうれしい機能です。 それと、これは本来レプリケーションのテスト用のツールなのですが、TableDiff ユーティリティもお勧めです。これを利用すると、2 つのテーブルの内容を比較したり、テスト データの SQL を生成することができます。 このように SQL Server 2005 のツールで SQL Server 2000 に対しても便利なものがまだまだたくさんありますので、今後も私のブログや PASSJ を通して情報共有していきたいと思っています。 | | * | ***SQL Server 2000 互換モードで動作させることも可能 プロファイラは、アップグレード時の動作確認ツールとしても利用できる** | | ***――** | **SQL Server 2000 から SQL Server 2005 へ移行するにあたってのポイントはどのような点でしょうか。** | | **米田** | 移行にあたってのお勧めな方法は、まず SQL Server 2005 を SQL Server 2000 と同じように使ってみることです。データベース エンジンや Transact-SQL などの SQL Server 2005 の中核となる部分は、基本的には旧バージョンの拡張になりますので、SQL Server 2000 のユーザーにとっては敷居が低いです。データベース エンジンの改良は内部的なものでアプリケーションにはほとんど影響ありませんし、Transact-SQL に関しては、拡張機能を利用しなければ SQL Server 2000 と同じように利用できます。また、データベースの互換性レベルを 80 に設定すれば、SQL Server 2000 との互換モードで動作させることもできます。 |  |  |  | | --- | --- | | **――** | **互換性レベルを変更することで、そのままでも動作するのは大きいですね。** | | **米田** | SQL Server 2000 上でバックアップしたデータベースは、SQL Server 2005 へリストアすると、自動的に互換性レベルが 80 になるのもうれしいところです。これは昔からある機能で、たとえば SQL Server 6.5 上でバックアップしたデータベースを SQL Server 2000 へリストアすると、互換性レベルが 65 に設定されていました。 一般的にバージョン アップというと拡張機能に目が行きがちなのですが、同じ機能をそのまま利用して、製品サポートの保証期間が延びるメリットを享受するという考え方もあります。SQL Server 2005 から経験をつむ新人の判断ミスも考慮に入れると、拡張機能を徐々に使っていき、それに応じて互換性レベルを最新に整えていくというのが理想的です。 |  |  |  | | --- | --- | | **――** | **バージョン アップをするだけでも、データベース エンジンの改良によってパフォーマンスが向上する可能性もありますね。** | | **米田** | あと、互換性に関する情報は Books Online が非常に充実しています。SQL Server 2000 と SQL Server 2005 での変更点や、互換性レベルが 80 の場合の具体的な動作、アップグレードに関する情報、将来のバージョンでなくなる機能などが詳しく記載されています。SQL Server 2000 での継続運用を考えている場合でも、見ておいた方がいいと思います。 |  |  |  | | --- | --- | | **――** | **確かに Books Online は必読ですね。** | | **米田** | また、プロファイラは、アップグレード時の動作確認ツールとしても利用できます。アップグレード前のアプリケーションの動作をプロファイラでキャプチャしておき、それを (テスト環境の) SQL Server 2005 へアップグレードしたデータベースに対して再生 (再実行) させる使い方ができます。これで SQL Server 2005 へアップグレードした後にアプリケーションが正しく動作するかどうかを確認することができます。 また、プロファイラの Deprecation イベント クラスを利用すれば、将来のバージョンでなくなってしまう機能を使った場合にそれを喚起してくれます。これにより、SQL Server 2005 では互換性があって動作する機能でも、将来のバージョンではサポートされなくなる機能もわかります。これは便利ですね。 | | * | ***SQL Server Migration Assistant (SSMA) で Oracle からの移行も簡単 PL/SQL を Transact-SQL へ自動変換。Oracle 独自の関数にも対応** | | ***――** | **他社データベースからの移行の観点からはいかがでしょうか。** | | **米田** | SQL Server Migration Assistant (SSMA) for Oracle V2.0 は、Oracle データベースからの移行 (マイグレーション) のためのツールですが、これは便利ですね。Oracle の PL/SQL で書かれたストアド プロシージャを Transact-SQL へ自動変換してくれたり、TO\_CHAR や TO\_DATE などの Oracle 独自の関数を SQL Server 用に変換してくれます。無償でダウンロードできるので、これから活用したいと思っています。 | | * | ***デフォルト セキュアでインストール直後はセキュリティが最も厳しい状態 リモート接続が無効なのは注意が必要** |  |  |  | | --- | --- | | ** | Photo * |  |  |  | | --- | --- | | **――** | **SQL Server 2005 を運用管理するにあたってのポイントにはどういった点がありますか?** | | **米田** | SQL Server 2005 の管理ツールは、従来の Enterprise Manager とクエリ アナライザを合体させたようなツール「Management Studio」に代わりますが、SQL Server 2000 のユーザーにとっては慣れるまでに少し時間がかかるかもしれません。しかし、かゆいところに手が届いているというか、細かいところで本当によくなっています。たとえば、ほとんどの画面で「スクリプト」ボタンが付いていて、GUI ベースで設定したものが、スクリプト ボタンを押すだけでその場で Transact-SQL スクリプトを生成できます。 GUI ツールは、バージョンが変わると変更される可能性がありますが、Transact-SQL は変更される可能性が低いものです。今までも SQL Server 6.5 で利用してきた Transact-SQL は、SQL Server 2005 になってもそのまま利用できるものが多くあります。そういった意味で、将来のバージョンを見据えて、GUI ベースで作成したものに対してスクリプトを生成しておき、Transact-SQL の勉強をするという使い方もできます。 |  |  |  | | --- | --- | | **――** | **確かに Transact-SQL を覚えておくことは管理をするうえで重要ですね。** | | **米田** | これは MSDE 2000 Release A からの流れなのですが、SQL Server 2005 は、デフォルト セキュア (Secure by Default) ということで、インストールをした直後はセキュリティが最も厳しい状態で、多くの拡張機能が無効になっています。具体的には、.NET 統合 (SQLCLR) や sp\_OACreate などの OLE オートメーション、xp\_cmdshell、HTTP エンドポイント、サービス ブローカ、SQLMail、Database Mail などが無効になっています。 MSDE の場合には、無効になっている機能を有効にするのはたいへんだったのですが、SQL Server 2005 では「SQL Server セキュリティ構成」というグラフィカル ツールが付いているので、簡単に有効/無効の設定ができるようになっています。 |  |  |  | | --- | --- | | **――** | **デフォルト セキュアは重要ですよね。必要に応じて、使いたい機能だけを有効にしていけるというのはいいですね。** | | **米田** | はい。最初はセキュリティを厳しくしておいて、ステップ バイ ステップで必要な部分だけを広げていくことができます。ただ、注意点が 1 つだけあります。インストール直後のデフォルト状態では「リモート接続」が無効になっている点です。SQL Server 2005 をスタンドアロンで利用する分には問題ないのですが、ネットワークを介して SQL Server 2005 を利用する場合には、リモート接続を有効にしておかなければなりません。 |  |  |  | | --- | --- | | **――** | **リモート接続は盲点ですね。SQL Server 2005 にしたら動かなくなった、という FAQ (よくある質問) になりそうですね。** | | **米田** | その他に運用管理の点では、Books Online が MSDN ライブラリと統一された点が気に入っています。Visual Studio と SQL Server のヘルプを同時に参照できますし、タブ形式で複数のウィンドウを開けるツールなので、調べたい項目に関連する複数のページを同時に開いておくことができます。ちょっとした改善点ではあるのですが、実際に Books Online を参照しながら管理作業をするときに、関連する複数のページを開きながらというのはよくやります。以前のバージョンでは、1 つのページしか開けなかったので、ページ間を行ったり来たりしなければならず、作業効率がよくありませんでした。 | | [ページのトップへ](http://www.microsoft.com/japan/sqlserver/2005/readystation/07/closeup.mspx#top)* | ***SQL Server 2005 は、ほとんどの機能が XML をベースにしている点がうれしい** | | ***――** | **その他に SQL Server 2005 で気に入っている機能はありますか?** | | **米田** | SQL Server 2005 は、ほとんどの機能が XML をベースにしている点がうれしいですね。XML ネイティブ データ型のサポートや、それに対してインデックス付けができる点はもちろんですが、SQL Server の構成情報や監視ツール、ステートメントの実行結果など、いろいろな場面で XML 形式で結果を取得できる点が気に入っています。 たとえば、プロファイラで Blocked process report というイベント クラスをキャプチャすると、ロック待ちに関する情報を XML 形式で取得できます。どのプロセス ID によってロック待ちが発生していたのかということや、ロックをかけている側およびロック待ちをしている側の両方の実行中のステートメント、ロック待ちをしている時間、トランザクションの分離レベルなども確認できます。これはロック待ちのトラブル シューティングをするときに本当に便利です。 また、SET STATISTICS XML ステートメントを使えば、実行プランを XML 形式で取得することもできます。これには Missing Index という機能も備わっていて、そのステートメントを実行するときに「こういったインデックスがあるとパフォーマンスが向上しますよ」というアドバイスをしてくれる機能です。このようなちょっと気の利いた機能は、本当に役立ちます。 |  |  |  | | --- | --- | | **――** | **最後にこの記事を読んでいただいている皆さんにひと言お願いします。** | | **米田** | SQL Server 2005 には .NET 統合など、魅力的な機能がたくさん追加されています。しかし、社内のプログラマが開発したストアド プロシージャが思うようなパフォーマンスが出なかった場合などに、どのようにサーバーをチューニングしていくのかということや、サーバーのサイジング、特に SQL Server 2000 と比べて、ハードウェアへの負荷のバランスがどう変化し、それにどう対応していくべきかという情報はまだありません。たとえば、巨大な変数を野放しに使えてしまって tempdb データベースを大量に消費するなどの影響も調査していく必要があると感じています。そのようなまだ明確になっていない部分を、これからブログや PASSJ を通して情報共有していきたいと思っています。 | |

  
*

|  |
| --- |
| **Microsoft MVP が語る SQL Server 2005** * SQL Server 2005 と同時出荷される Visual Studio 2005。SQL Server 2005 は、Visual Studio 2005 と統合され、Visual Studio 2005 のリッチな統合開発環境 (IDE : Integrated Development Environment) の下で、SQL Server 2005 のデータベース オブジェクトの開発からコンパイル、デバッグ、テスト、配置までを一貫して行えるようになりました。他のデータベースを圧倒する開発生産性と、データベースと連携した実践的なアプリケーションを、いかにプログラム コードを記述することなく実装できるかにご注目ください。 * *→*[Microsoft MVP とは](#c8_7) |

|  |
| --- |
| * |
| **Index**   |  |  | | --- | --- | |  | | | * | [SQLCLR によるデータベース オブジェクトの開発、デバッグ、配置](#c6_1) | | * | [Visual Studio 2005 のサーバー エクスプローラ](#c6_2) | |  | | | * | [Visual Studio 2005 によるデータベース アプリケーション開発](#c6_3) | | * | [[DataGridView] の利用手順](#c6_4) | | * | [今度の Visual Studio は GUI だけではない](#c6_5) | |  | | | * | [Report コントロールによるレポート作成](#c6_6) | |
| * |

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | * | |

|  |  |  |
| --- | --- | --- |
| * | **MVP スペシャル**   |  | | --- | | **Visual Studio 2005 統合で大きく変わる開発環境** **（Ready編集局　松本美穂・松本崇博）** ** **SQLCLR による開発生産性とパフォーマンスの向上 プログラム コードをほとんど記述することなくリッチな DB アプリを構築 Report コントロールによるレポート作成。ドラッグ アンド ドロップで完成** |   * SQL Server 2005 は、Visual Studio 2005 との統合および連携機能が大幅に強化され、「SQLCLR のサポート」や「サーバー エクスプローラ」「データベース連携のためのコントロール」「レポート作成を行えるコントロール」など、現場で役立つ実践的な機能がたくさん追加されています。SQL Server 2005 と Visual Studio 2005 の高度な統合によって変わるデータベース アプリケーション開発の世界を、画面ショット満載で解説します。 |
|  | |  |  | | --- | --- | | * | **SQLCLR の最大のメリットは「開発生産性の向上」 使い慣れた Visual Basic や C# 言語で ストアド プロシージャやユーザー定義関数を作成可能** | |

|  |
| --- |
| **SQLCLR によるデータベース オブジェクトの開発、デバッグ、配置** |

*  
SQLCLR は、.NET Framework の CLR (Common Language Runtime : 共通言語ランタイム) を SQL Server 2005 に統合した機能です。これにより、ストアド プロシージャやユーザー定義関数、トリガ、ユーザー定義データ型といったデータベース オブジェクトを .NET Fremework 言語 (Visual Basic、C# など) で作成できるようになります。  
SQLCLR の最大のメリットは、使い慣れた Visual Basic や C# でデータベース オブジェクトを開発できる点です。従来は、ストアド プロシージャやユーザー定義関数は、Transact-SQL という SQL Server 独自の拡張言語を使って開発する必要がありましたが、Transact-SQL は一般的なプログラミング言語とは異なり、文法が独特なために習得が大変でした。また、プログラミング言語としては機能が少なかったため、複雑な処理を実装するのが大変であったり、実装できない処理もありました。そういった場合には“拡張ストアド プロシージャ”を作成するという回避策もありましたが、拡張ストアド プロシージャは、C++ 言語を使って Open Data Services API や Win32 API といった低水準の API を駆使しなければなりませんでした。これでは開発効率があまりにも悪く、生産性が低いものでした。  
一方 SQLCLR は、Visual Basic や C# で開発できるので、普段から Visual Basic や C# でアプリケーションを開発している方にとっては、特に新しい知識を覚えることなく、同じようにデータベース オブジェクトを開発できます。また、.NET Framework 2.0 で提供される豊富なクラス ライブラリも同様に利用できるので、暗号化や正規表現、配列、バイナリ操作、ファイル、ネットワーク、レジストリ アクセス、XML 操作、Web サービス利用、イベント ログ操作といった複雑な処理も簡単に実装できます。  
  
**◎SQLCLR はパフォーマンス向上もメリット**  
SQLCLR は、パフォーマンス上のメリットもあります。SQLCLR のデータベース オブジェクトは、SQL Server と同じプロセス内 (イン プロセス) で実行されるからです。また、複雑な処理を実行する場合は、SQLCLR のほうがパフォーマンスが良いという検証結果も挙がっています (私自身も検証しましたが、複雑な処理を何度も繰り返して呼び出すような場合は、SQLCLR のほうが断然パフォーマンスが良いという結果を得られました)。  
Transact-SQL では数十行にもわたって記述しなければならないような複雑な処理が .NET であればわずか数行で記述できるようになり、かつ .NET のほうが効率良く実行できるのです。また、アプリケーションとデータベース オブジェクトを同じ言語 (Visual Basic や C#) で一貫して開発できるというのは、開発生産性の面で大きなメリットとなります。  
もちろん、すべての処理において SQLCLR へ置き換えるべきというわけではなく、単純なデータ アクセスや演算処理であれば、SQLCLR よりも Transact-SQL のほうがパフォーマンスが良いので、そういった場面では従来どおり Transact-SQL を利用することをお勧めします。あくまでも Transact-SQL では実装が難しい処理や、複雑な処理を行う場合、特に何度も複雑な処理を呼び出すような場合には、SQLCLR を積極的に利用すると良いでしょう。  
*  
**◎Visual Studio 2005 による SQLCLR 開発の流れ**  
ここからは、Visual Studio 2005 による SQLCLR 開発がいかに簡単かを、ユーザー定義関数を作成するときの手順を例に説明していきます。  
*

|  |  |
| --- | --- |
| **1.** | ユーザー定義関数を作成するには、まず Visual Studio 2005 を起動して、[ファイル] メニューの [新規作成] から [プロジェクト] を選択し、新しいプロジェクトを作成します。 |
| * http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr01_l.gif   |  | | --- | | 新しいプロジェクトの作成 | | |

*

|  |  |
| --- | --- |
| **2.** | [プロジェクトの種類] から利用したい言語 (Visual Basic や Visual C#) を選択して [データベース] をクリックし、[テンプレート] の一覧から [SQL Server プロジェクト] を選択します。これにより、SQLCLR のデータベース オブジェクトを作成するためのプロジェクトを作成できます。 |

*

|  |  |  |
| --- | --- | --- |
| **3.** | ソリューション エクスプローラで [プロジェクト名] を右クリックして [追加] を選択します。すると、SQLCLR で作成可能なデータベース オブジェクトが一覧表示されます。 *   |  | | --- | | http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr02_l.gif* SQLCLR のデータベース オブジェクトの選択。ここでは [ユーザー定義関数] を選択します。 | |

*

|  |  |  |
| --- | --- | --- |
| **4.** | ユーザー定義関数を作成するための雛形 (テンプレート) が自動生成され、任意のコードを記述できるようになります。 *   |  | | --- | | http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr03_l.gif * ユーザー定義関数の雛形。ファンクション名 (画面では Function1) がユーザー定義関数の名前になり、Return で戻り値を指定します。コードは、通常のアプリケーションを開発するときと同様、.NET Framework の基本クラス ライブラリを利用してさまざまな処理を記述できます。 | |

*

|  |  |  |
| --- | --- | --- |
| **5.** | 最後に [ビルド] メニューの [ <プロジェクト名> の配置] をクリックすると、コードがコンパイルされ、ユーザー定義関数が SQL Server 上のデータベース内へ配置されます。 *   |  | | --- | | http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr04_l.gif * コードのコンパイルと SQL Server への配置。配置されたユーザー定義関数を確認するには、[サーバー エクスプローラ] を表示して [関数] を展開します。 | |

*  
以上でユーザー定義関数の作成は完了です。Visual Studio 2005 と統合されたことにより、雛形が提供され、コンパイルや SQL Server 上への配置も GUI ベースで簡単に行えます。  
*  
*  
**◎SQLCLR はデバッグも簡単**  
Visual Studio 2005 との統合で何といってもうれしいのはデバッグ機能です。SQLCLR のデータベース オブジェクトは、通常のアプリケーションを開発するときと同じようにデバッグすることができます。開発時は、コードを記述する時間よりもデバッグをしている時間の方が何倍もの時間を費やすものです。そんなとき、Visual Studio 2005 のようなリッチなデバッグ機能があれば、バグをすばやく発見でき、デバッグ時間を大幅に短縮することができます。これは本当にうれしい機能です。  
SQLCLR をデバッグするには、次のように操作します。まず、[ソリューション エクスプローラ] から [Test Scripts] フォルダにある [Test.sql] という名前のテスト スクリプトを修正します。  
*

|  |
| --- |
| http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr05_l.gif *テスト スクリプト「Test.sql」の修正。デバッグしたいデータベース オブジェクトの名前に変更し、必要に応じてパラメータを与えます。 |

*  
テスト スクリプトの修正後、次のように [デバッグ] メニューの [デバッグ開始] をクリックすると、デバッグが開始されます。  
*

|  |
| --- |
| http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr06_l.gif Visual Studio 2005 で SQLCLR のデータベース オブジェクトをデバッグしているところです。通常のアプリケーションのデバッグ時と同様、ブレーク ポイントの設定や 1 行ずつ実行するステップ イン、ステップ オーバー、マウス オーバーによる変数表示、ローカル ウィンドウでの変数確認、呼び出し履歴などを確認できます。 |

*  
このように SQLCLR は、Visual Studio 2005 を利用することで、開発、デバッグおよび配置が簡単に行えます。また、使い慣れた Visual Basic や C# で開発でき、かつデバッグ機能も提供されることによる「開発生産性の高さ」が SQLCLR の最大のメリットです。

|  |  |  |
| --- | --- | --- |
| * | **Visual Studio 2005 のサーバー エクスプローラで SQL Server 2005 上にテーブルや DB ダイアグラムを作成、クエリの実行** | |
| **Visual Studio 2005 のサーバー エクスプローラ** | |

*  
Visual Studio 2005 には、サーバー エクスプローラと呼ばれる SQL Server 2005 との連携機能があります。これは、SQL Server 2005 上のテーブルやデータベース ダイアグラム、制約、インデックス、ビュー、ストアド プロシージャといったさまざまなデータベース オブジェクトをグラフィカルに作成、編集できる機能です。  
次の画面は、サーバー エクスプローラでデータベース ダイアグラムを作成および編集しているところです。  
*

|  |
| --- |
| http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr07_l.gif * データベース ダイアグラムの編集。テーブル間のリレーションシップやテーブルの列名、データ型をグラフィカルに表示したり、列の追加と削除、インデックスや制約の追加と削除をしたりすることもできます。 |

*  
サーバー エクスプローラは、クエリ デザイナによってグラフィカルにクエリ (SQL ステートメント) を実行したり、ビューを作成したりすることもできます。次の画面は、[商品] テーブルと [商品区分] テーブルを結合した結果を返すビューを作成しているところです。  
*

|  |
| --- |
| http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr08_l.gif * クエリ デザイナでグラフィカルにビューを作成している画面 |

*  
また、サーバー エクスプローラは、Transact-SQL ベースのストアド プロシージャやユーザー定義関数を作成する機能も備わっています。しかも、デバッグ機能も備わっているので、ブレーク ポイントを設定してステップ実行したり、実行中の変数を確認したりすることもできます。次の画面は、ストアド プロシージャをデバッグしているところです。  
*

|  |
| --- |
| http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr09_l.gif * Transact-SQL で記述されたストアド プロシージャをデバッグしている画面 |

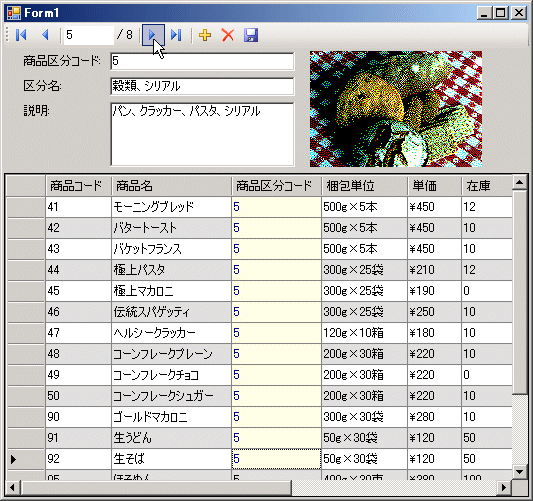
*  
このように Visual Studio 2005 に搭載されたサーバー エクスプローラを利用することで、SQL Server 2005 上のさまざまなオブジェクトをグラフィカルに編集することができます。特にデータベース ダイアグラムや Transact-SQL ストアド プロシージャのデバッグ機能は大変便利なので、ぜひ活用していただければと思います。  
なお、サーバー エクスプローラは、従来の Visual Studio から提供されている機能ですが、1 つ前のバージョンである Visual Studio .NET 2003 では、SQL Server 2005 には対応していないため、一部の機能が利用できなかったり、SQL Server 2005 からの新機能は利用できませんのでご注意ください。

|  |  |
| --- | --- |
| * | **データベース連携のための便利なコントロールが多数提供され プログラム コードをほとんど記述することなくリッチな DB アプリが開発可能** |

|  |
| --- |
| **Visual Studio 2005 によるデータベース アプリケーション開発** |

Visual Studio 2005 には、データベース連携のための便利なコントロールが数多く追加されています。特に Windows アプリケーション用に提供される [BindingNavigator] コントロールと [DataGridView] コントロールは、非常に高機能でたいへん便利なコントロールです。この 2 つのコントロールを利用すると、プログラム コードをほとんど記述することなくリッチなデータベース アプリケーションを開発できます。

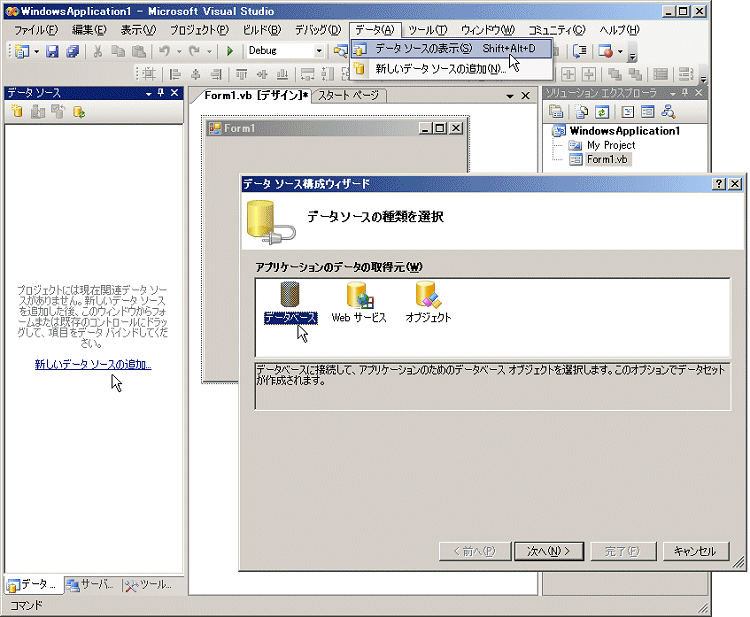


**◎[BindingNavigator] コントロールの利用手順**  
ここからは、実際に [BindingNavigator] コントロールを利用するときの手順を画面ショットを交えて説明します。ここでは Access のサンプル データベースとしてお馴染みの Northwind を SQL Server 上へ移行した NorthwindJ という名前のデータベースを利用して、次のようなアプリケーションを作成する手順を例に説明します。

これから作成するアプリケーション。上部に商品区分 (商品分類名や説明、画像) を 1 件ずつ表示し、その商品区分内に含まれる商品の一覧を表形式で表示します。このような [マスタ] － [詳細] 形式もプログラムを 1 行も記述することなく実現できます。

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **1.** | まず最初の作業はデータ セット (DataSet) の作成です。データ セットを作成するには、次のように [データ] メニューの [データ ソースの表示] をクリックします。すると [データ ソース] ウィンドウが表示されるので [新しいデータ ソースの追加] をクリックします。 |



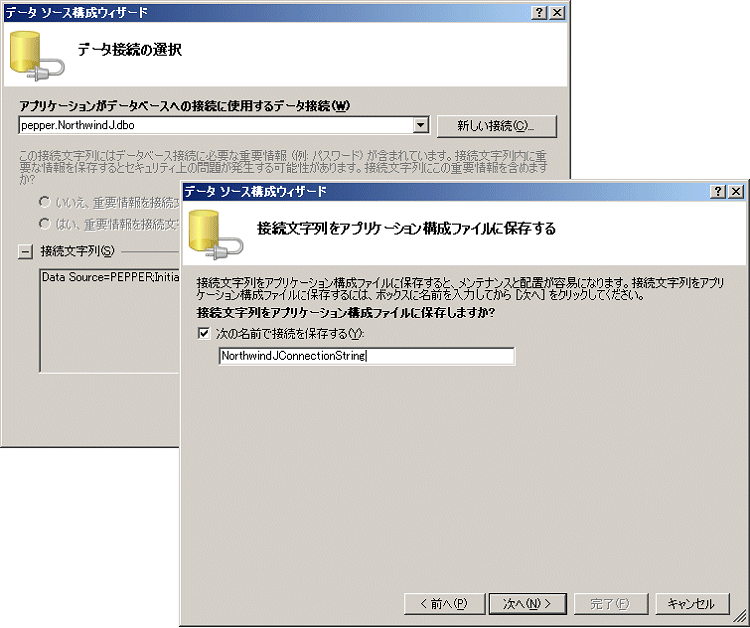
データ ソース構成ウィザードの起動。[アプリケーションのデータの取得元] では [データベース] を選択します。これにより、SQL Server などのデータベース データを元にデータ セットを作れるようになります。

|  |
| --- |
|  |

*

|  |  |
| --- | --- |
| **2.** | [データ接続の選択] ページでは、SQL Server 上のデータベースへの接続先情報を選択または新しく作成します。 |

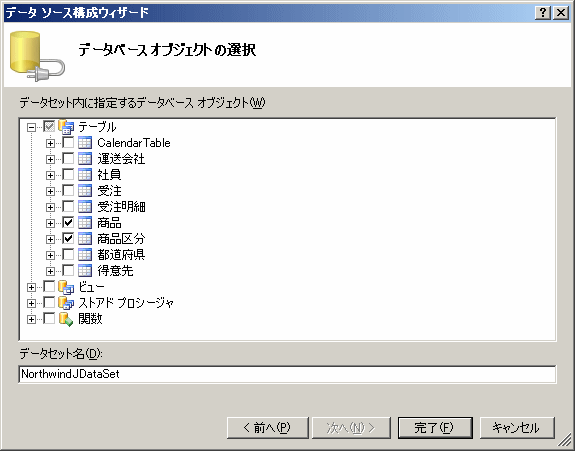
SQL Server データベースへの接続先情報。ここでは、SQL Server 上の NorthwindJ データベースを選択しています。



|  |  |
| --- | --- |
| |  | | --- | |  | |

*

|  |  |
| --- | --- |
| **3.** | [データ ソース オブジェクトの選択] ページでは、データ セットに含めたいテーブルを選択します。 |

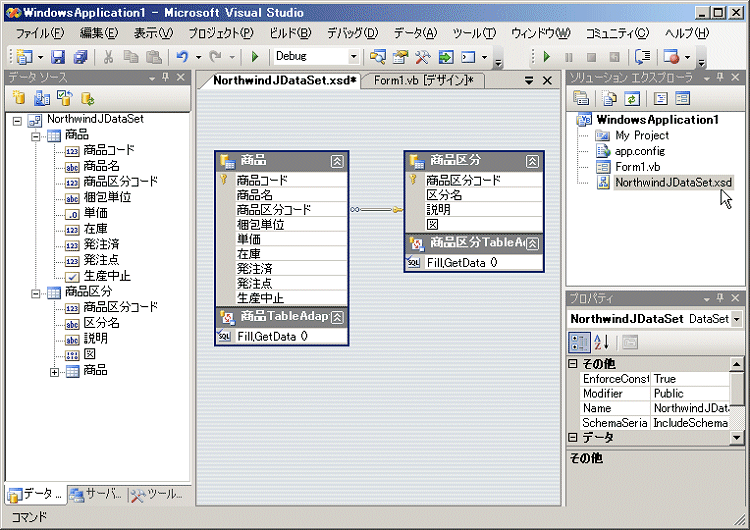


データ セットに含めたいテーブルの選択。ここでは、商品と商品区分テーブルを選択しています。[完了] ボタンをクリックするとデータ セットの作成が完了します。

|  |  |
| --- | --- |
| |  | | --- | |  | |

*

|  |  |
| --- | --- |
| **4.** | ソリューション エクスプローラで「<データ セット名>.xsd」というファイルをダブル クリックして、作成されたデータ セットを確認します。 |

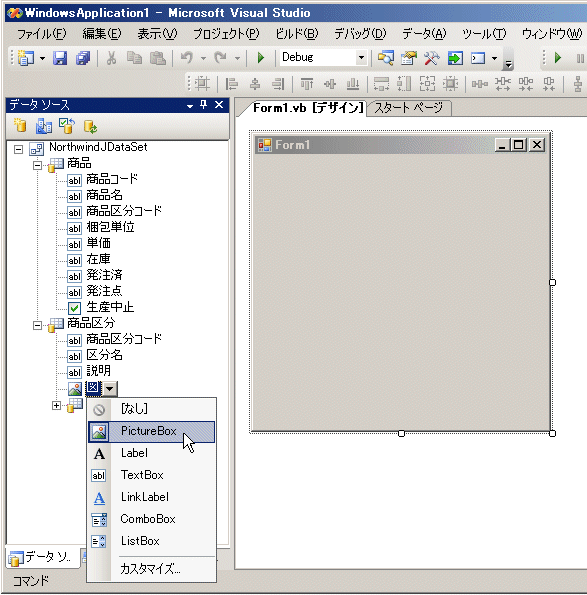
作成されたデータ セット (.xsd ファイル) の確認

|  |  |
| --- | --- |
| |  | | --- | |  | |

*

|  |  |
| --- | --- |
| **5.** | Windows フォームのデザイン画面へ戻ります。[データ ソース] ウィンドウで [商品区分] テーブルを展開し、[図] 列をクリックして [PictureBox] を選択します。 |

|  |
| --- |
|  |

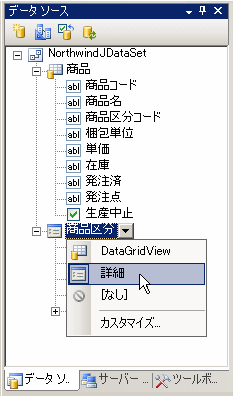


[商品区分] テーブルの [図] 列を [PictureBox] へ変更。[図] 列には、商品区分の画像データ (bmp 形式のバイナリ データ) が格納されているので、それを [PictureBox] コントロールで表示しようとしています。

*

|  |  |
| --- | --- |
| **6.** | [商品区分] テーブルをクリックして [詳細] をクリックします。 |

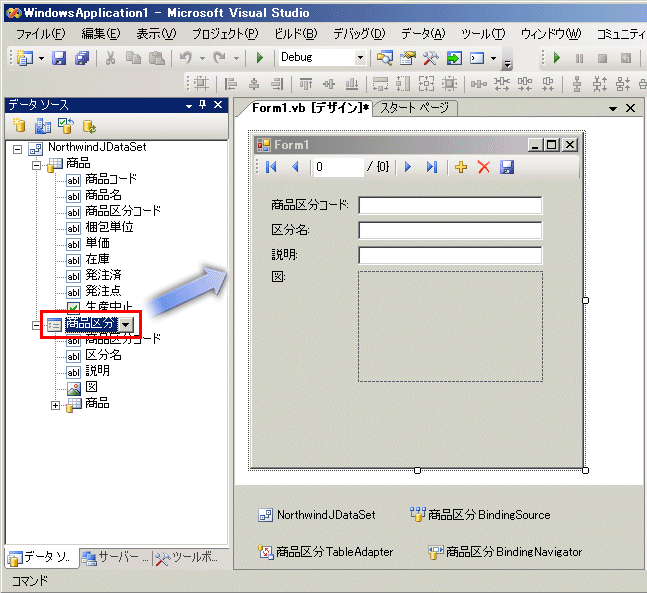
|  |
| --- |
|  |



[商品区分] テーブルを [詳細] へ変更。デフォルトは [DataGridView] で表形式での表示になりますが、[詳細] を選択するとテキスト ボックスやラベル コントロールでの 1 件ずつのデータ表示が可能になります。

*

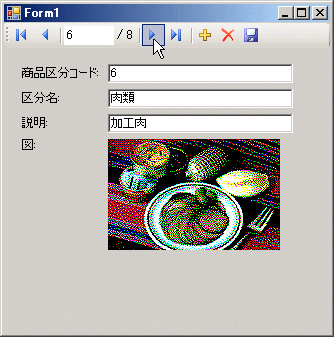
|  |  |
| --- | --- |
| **7.** | [商品区分] テーブルを Windows フォーム上へドラッグ アンド ドロップします。 |

[商品区分] テーブルを Windows フォーム上へドラッグ アンド ドロップします。これにより、[BindingNavigator] コントロールとテキスト ボックス、ラベル、PictureBox などが自動生成され、商品区分テーブルのデータ (商品区分コード、区分名、説明、図) を 1 件ずつ表示できるようになります。

|  |
| --- |
|  |

*

|  |  |
| --- | --- |
| **8.** | [デバッグ] メニューの [デバッグの開始] をクリックすると、次のように実行結果を確認できます。 |

実行結果。[BindingNavigator] コントロールで次のデータへ移動したり、前のデータへ戻ったりできます。また、[+] ボタンでデータを追加したり、[×] ボタンでデータの削除、 [フロッピー] ボタンでデータを更新したりすることも可能です。

|  |
| --- |
| **[DataGridView] の利用手順** |

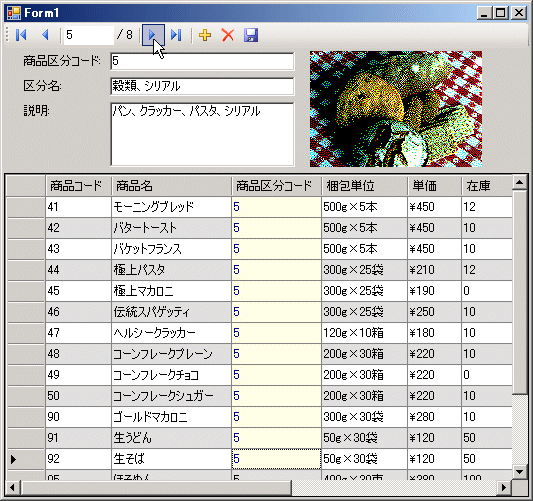
次に [DataGridView] コントロールを使って、同じ商品区分に属する商品の一覧を表形式で表示していきます。とはいっても、追加の作業はわずか 1 つだけです。次のように [データ ソース] ウィンドウで [商品区分] テーブルの階層下に表示される [商品] テーブルを Windows フォーム上へドラッグ アンド ドロップするだけです。



|  |
| --- |
| [商品区分] テーブルの階層下の [商品] テーブルを Windows フォーム上へドラッグ アンド ドロップ。商品テーブルのように階層表示されるのは、親子関係のリレーションシップがあるテーブルのみです。http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr19_l.gif |

[デバッグ] メニューの [デバッグの開始] をクリックすると、次のように実行結果を確認できます。

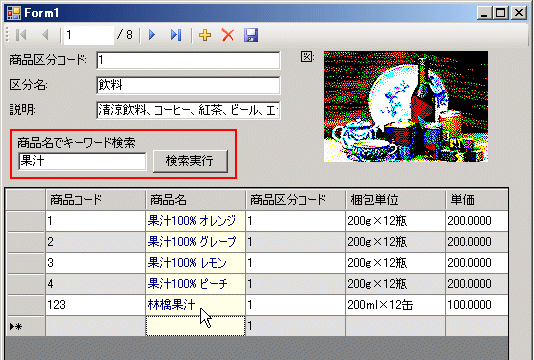


  
実行結果。[BindingNavigator] コントロールで商品区分を切り替えると、その該当区分に属する商品のみが [DataGridView] コントロールへ表形式で表示されます。



このように [BindingNavigator] と [DataGridView] コントロールを利用すれば、[マスタ] － [詳細] 形式のアプリケーションもプログラムを 1 行も記述することなく実現できます。[DataGridView] は、セル単位で背景色を変更したり、1 行ごとに背景色を変更したりといった書式変更も [プロパティ] ウィンドウから操作するだけで簡単に行えます。  
また、今回作成したアプリケーションは、[BindingNavigator] や [DataGridView] へ連結する列データは、テキスト ボックスと [PictureBox] を利用しましたが、その他にもチェック ボックスやコンボ ボックス、リスト ボックス、ボタン、ハイパーリンク (LinkButton)、DateTimePicker (カレンダーから日付の選択)、カスタム コントロールなど、いろいろな種類へ変更することができます。これらの機能は多くの開発者が待ち望んでいたものではないでしょうか (従来の Visual Studio では、DataGrid という表形式でのデータ表示が行えるコントロールがありましたが、画像データやコンボ ボックスといったテキスト ボックス以外のコントロールを表示したり、セル単位で書式変更したりするのは非常にたいへんでした)。

|  |
| --- |
| **今度の Visual Studio は GUI だけではない** |

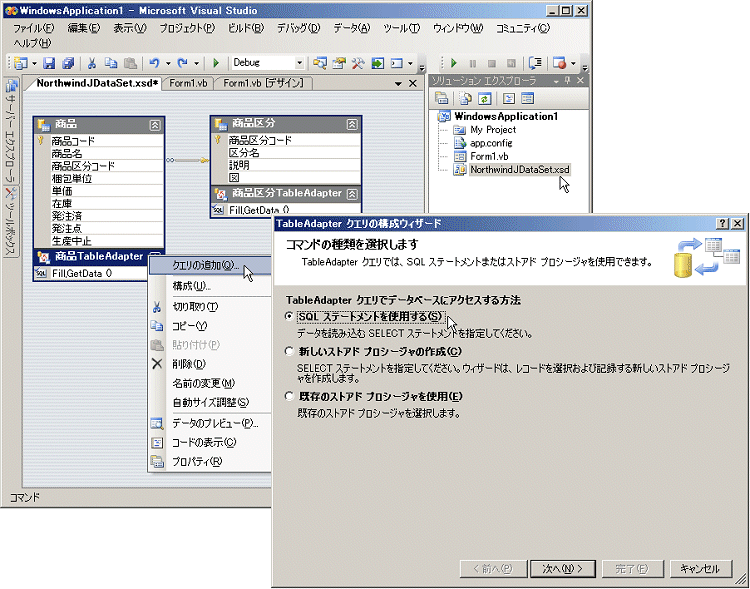
  
従来の Visual Studio では、GUI ベース (ウィザードやドラッグ アンド ドロップ) で作成したアプリケーションは、コード レベルでカスタマイズしようとするとたいへんでした。これでは、“型にはまった”アプリケーションは作成できても、利用者のニーズに合わせた柔軟なアプリケーションは作成できません。しかし、今度の Visual Studio 2005 は違います。Visual Studio 2005 の [BindingNavigator] と [DataGridView] コントロールは、GUI ベースで作成したデータ セットだけでなく、他のクラスによって生成されたデータ セットや、Web サービス経由で取得したデータ セットにも対応しています。これにより、データ セットを取得する部分は別クラスにして、UI (プレゼンテーション) とビジネス ロジックを分離するといったことも簡単に行えます。  
また、GUI ベースのデータ セットを利用する場合にも、柔軟にコードを追加してカスタマイズできるようになっています。たとえば、次のような商品名でキーワード検索を行うアプリケーションであっても、わずか 1 行コードを追加するだけで簡単に完成します。



商品名でキーワード検索を行うアプリケーション

*  
このアプリケーションを作成する手順は、次のとおりです。  
*

|  |  |
| --- | --- |
| **1.** | まず、ソリューション エクスプローラでデータ セット (.xsd ファイル) を表示します。[商品] テーブルの [商品 TableAdapter] を選択した後、右クリックして [クエリの追加] をクリックします。 |
|  | |



[商品 TableAdapter] を右クリックして [クエリの追加] をクリック。これにより、[TableAdapter クエリの構成] ウィザードが起動します。今回は、SELECT ステートメントの LIKE 演算子を使って商品をキーワード検索するクエリを作成していくので、最初の [コマンドの種類を選択します] ページでは [SQL ステートメントを使用する] を選択します。

*

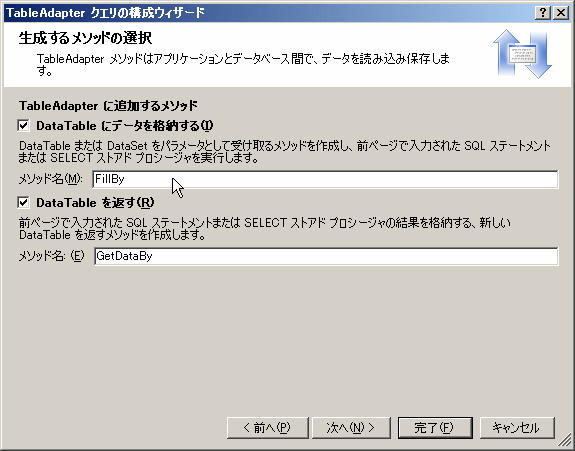
|  |  |
| --- | --- |
| **2.** | [クエリの種類の選択] ページでは、[複数行を返す SELECT] を選択します。 |

*

|  |  |
| --- | --- |
| **3.** | [SQL SELECT ステートメントの指定] ページでは、[クエリ ビルダ] ボタンをクリックして、クエリ ビルダを起動します。起動後、次のように [商品名] 列の [フィルタ] セクションで「LIKE @p1」と入力します。  [商品名] 列の [フィルタ] セクションで「LIKE @p1」と入力。これにより、SQL ペインの WHERE 句が「WHERE (商品名 LIKE @p1)」と自動生成され、クエリをパラメータ化することができます。[[商品名] 列の [フィルタ] セクションで「LIKE @p1」と入力。これにより、SQL ペインの WHERE 句が「WHERE (商品名 LIKE @p1)」と自動生成され、クエリをパラメータ化することができます。](http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr23_l.gif) |
|  | |

*

|  |  |
| --- | --- |
| **4.** | [生成するメソッドの選択] ページでは、[DataTable にデータを格納する] セクションのメソッド名が「FillBy」になっていることを確認します。このメソッドは、後ほどコード エディタから利用します。 |

ウィザードで FillBy というメソッドを自動生成

|  |
| --- |
|  |

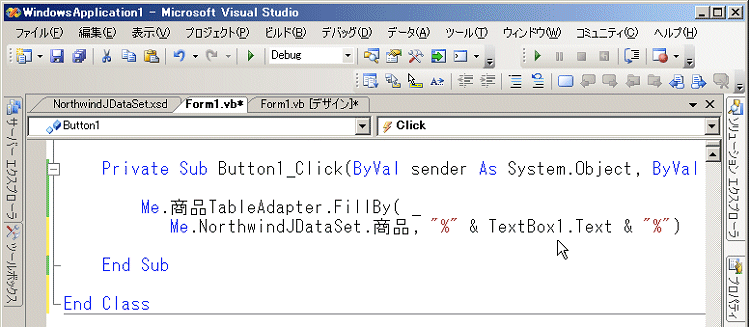
*

|  |  |
| --- | --- |
| **5.** | Windows フォームのデザイン画面へ戻ります。ツール ボックスからボタン (Button1) とテキスト ボックスを [DataGridView] コントロールの上部へ 1 つずつ配置します。 |

|  |  |
| --- | --- |
| |  | | --- | | http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr25_l.gifボタン (Button1) とテキスト ボックス (TextBox1) を 1 つずつ配置 | |

*

|  |  |
| --- | --- |
| **6.** | 配置した Button1 をダブル クリックして、Button1 の Click イベント ハンドラを生成します。イベント ハンドラへ次の画面のようにコードを記述します。 |
|  | |



FillBy メソッドを利用する記述。第 2 引数へ「"%" & TextBox1.Text & "%"」と与えることで、クエリが「WHERE 商品名 LIKE '%xxx%'」と組み立てられて (xxx はテキスト ボックスへ入力された値) 実行され、商品名によるキーワード検索が行えるようになります。

*

|  |  |
| --- | --- |
| **7.** | http://www.microsoft.com/japan/sqlserver/2005/readystation/06/images/06cusp_scr27_l.gif[デバッグ] メニューの [デバッグの開始] をクリックすると、次のように実行結果を確認できます。 * |

実行結果。商品名でキーワード検索が行えます。

|  |
| --- |
|  |

このように Visual Studio 2005 では、GUI ベースで作成したアプリケーションでも柔軟にコードを追加し、簡単にカスタマイズしていけるようになりました。

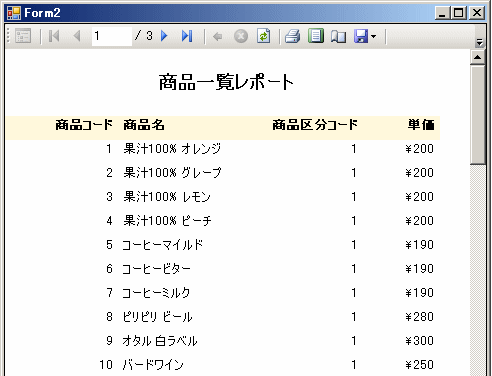


|  |  |
| --- | --- |
| * | ***Report コントロールでレポート作成 ドラッグ アンド ドロップで完成** |

|  |
| --- |
| **Report コントロールによるレポート作成** |

Visual Studio 2005 には、SQL Server 2005 の Reporting Services と同様のレポートをローカルに作成できる Report コントロールという機能 (正確には ReportViewer コントロール) が提供されています。レポートのデータ ソースにはデータ セットを指定することができ、レポートの作成方法は Reporting Services の場合とまったく同じで、ドラッグ アンド ドロップでグラフィカルに作成できます。ここでは、次のような商品一覧を表示するレポートを作成する手順を例に ReportViewer コントロールの利用手順を説明します。



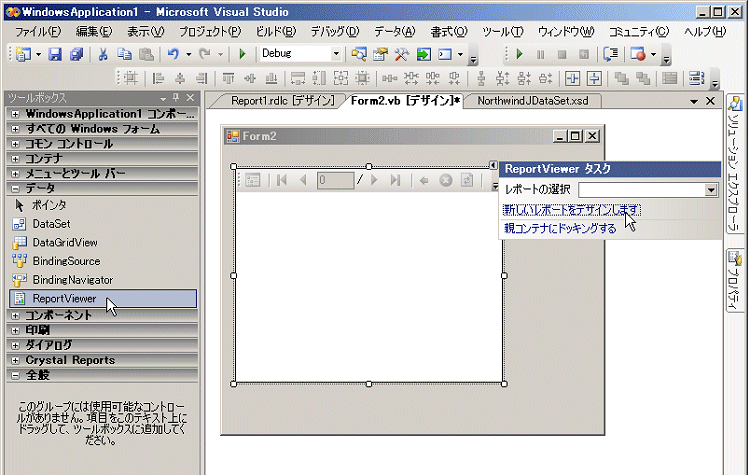


これから作成するレポート アプリケーション

|  |  |
| --- | --- |
| **1.** | まず、[[BindingNavigator] コントロールを利用したアプリケーションのとき](#c6_3)の手順 1 ～ 4 と同じようにデータ セットを作成します。 |

*

|  |  |
| --- | --- |
| **2.** | Windows フォームのデザイン画面へ戻り、ツール ボックスの [データ] セクションから [ReportViewer] コントロールを Windows フォーム上へドラッグ アンド ドロップします。 |

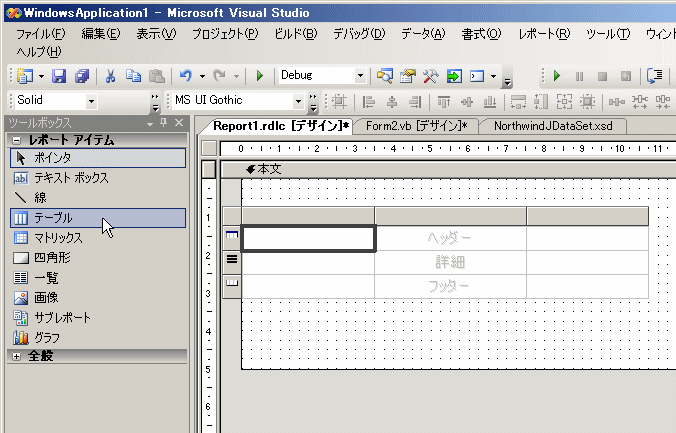


ReportViewer コントロールを Windows フォーム上へドラッグ アンド ドロップ。貼り付け後、[ReportViewer] タスクで [新しいレポートをデザインします] をクリックします。すると、Reporting Services と同じレポート デザイナが起動します。

|  |  |
| --- | --- |
| |  | | --- | |  | |

*

|  |  |
| --- | --- |
| **3.** | ツールボックスの [レポート アイテム] セクションから [テーブル] を選択し、レポート デザイナ上へドラッグ アンド ドロップします。 |

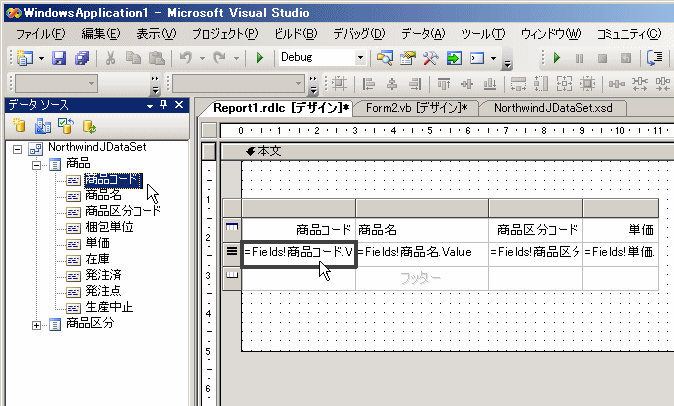


[テーブル] をレポート デザイナ上へドラッグ アンド ドロップ。これにより、表形式のレポートを作成できるようになります。

|  |  |
| --- | --- |
| |  | | --- | |  | |

*

|  |  |
| --- | --- |
| **4.** | [データ ソース] ウィンドウを開き、[商品] テーブルを展開します。「商品コード」と「商品名」「商品区分コード」「単価」をテーブルの詳細行の各列へドラッグ アンド ドロップします。 |

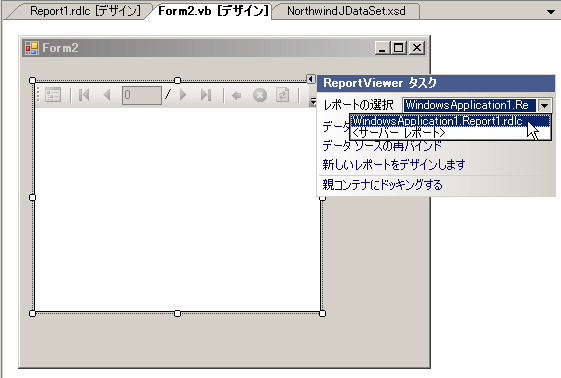


商品コード、商品名、商品区分コード、単価をテーブルの詳細行の各列へドラッグ アンド ドロップ。テーブルに列を追加するには、テーブルを右クリックして [右に列を挿入] をクリックします。

|  |  |
| --- | --- |
| |  | | --- | |  | |

*

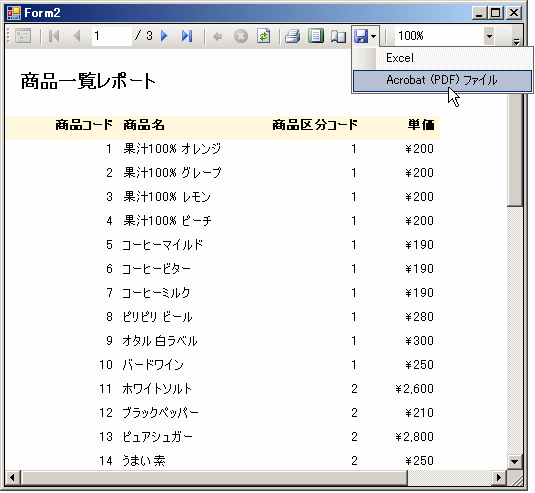
|  |  |
| --- | --- |
| **5.** | Windows フォームのデザイン画面へ戻り、[ReportViewer] タスクの [レポートの選択] で今作成したレポート (.rdlc ファイル) を選択します。 |



|  |  |
| --- | --- |
| |  | | --- | |  | |

*

|  |  |
| --- | --- |
| **6.** | [デバッグ] メニューの [デバッグの開始] をクリックすると、次のように実行結果を確認できます。 |



実行結果。商品コード、商品名、商品区分コード、単価を表形式で表示するレポートを確認できます。ツール バーのフロッピー ボタンをクリックすると、レポート内容を Acrobat PDF ファイルや Excel ファイル形式へエクスポートすることもできます。また、ツール バーには [印刷レイアウト] [ページ設定] [印刷] ボタンもあり、レポートの印刷も可能です。

このように Report コントロールを利用すると、ドラッグ アンド ドロップだけで簡単にレポートを作成することができます。従来の Visual Studio では、レポート作成に Crystal Report やサード パーティ製のレポート ツールを利用されていたという方が多かったと思いますが、Report コントロールの登場によって、標準でパワフルなレポートを作成できるようになりました。これは私もたいへん気に入っている機能の 1 つです。  
以上、いろいろな Visual Studio 2005 と SQL Server 2005 との統合および連携機能を紹介してきましたが、従来の Visual Studio とは比較にならないほどパワー アップしているのがおわかりいただけたのではないでしょうか。皆様もぜひ 1 度試してみていただければと思います。

Web ストリーミング配信として、SQL Server 2005 の SQLCLR 機能を、簡潔なデモ付きでわかりやすく解説しているムービーもありますので、ぜひこちらもご覧ください。

* [SQLCLR によるデータベース オブジェクト開発](http://www.microsoft.com/japan/seminar/SQLServer2005/databaseobject/play.aspx)
* [Visual Studio 2005 の注目の新機能を短期間で学習したい方は、「300 秒でズバリ !!」「10 行でズバリ !!」シリーズがお勧めコンテンツです](http://msdn.microsoft.com/ja-jp/events/dd283146.aspx)
* [ASP.NET 2.0 の注目の新機能を短期間で学習したい方は、「ステップ 7」シリーズがお勧めコンテンツです](http://msdn.microsoft.com/ja-jp/vstudio/dd266977.aspx)



|  |
| --- |
| **Microsoft MVP が語る SQL Server 2005** * Microsoft MVP へのインタビューから SQL Server 2005 を探る READY クローズアップ。この連載では、SQL Server 2005 に対する期待や導入メリット、具体的な利用方法などについて、Microsoft MVP として表彰された方々へインタビューを行い、SQL Server 2005 に対する想いを語っていただきます。 * *→*[Microsoft MVP とは](#c8_7) |
| |  |  | | --- | --- | | **MVP インタビュー 第 4 回**  **Microsoft SQL Server 2005 は、ココが違う 新次元の開発生産性を持つ SQL Server 2005** | * | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * | |  | | --- | |  |   * | | |
|  | |  |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | **Index***   |  |  | | --- | --- | | * | [SQL Server のエンジンは、すべてのエディションで共通のソース コード。セキュリティと開発生産性もほぼ共通の実装。上位エディションへシームレスに移行できる柔軟性は本当にありがたい](#c4_1) | | * | [SQL Server 2005 で 1 番気に入っているのは SQLCLR。開発生産性だけでなく、パフォーマンスとセキュリティ向上のメリットもある](#c4_2) | | * | [SOA (サービス指向アーキテクチャ) で再利用性の向上と変化に耐えうる柔軟性を持った IT システムの実現](#c4_3) | | * | [HTTP エンドポイントで IIS を使用しないで XML Web サービスを公開可能](#c4_4) | | * | [SQL Server 2005 は非同期フレームワークを標準提供。サービス ブローカによってスケーラビリティが向上](#c4_5) | | * | [Visual Studio 2005 との統合により開発、デバッグ、テスト、展開といった各サイクルがいずれも強化されている](#c4_6) | | * | [SQL Server 2005 はデータベース エンジンのソース コードとセキュリティや開発生産性に関する機能がすべてのエディションでほぼ共通の実装。ホビー ユースから大規模な基幹系システムまでシームレスに利用できる](#c4_7) | | |  | **沖 要知 氏** Microsoft MVP for SQL Server  NRIラーニングネットワーク株式会社  マイクロソフト認定データベース アドミニストレータ (MCDBA) 資格を有し、主にデータベース関連のトレーニング開発と実施を担当。  日経BPソフトプレス刊行の「Inside Microsoft SQL Server 7.0」および「Microsoft SQL Server 2000 リソース キット」などの日本語化で技術監修も担当し、昨年マイクロソフトより SQL Server カテゴリで MVP アワードを受ける。  沖要知氏最近は、SQL Server 2005 CTP 版をベースにしたセミナーやハンズオン トレーニングを展開している。 |  |  | | --- | | * | | | |

*

|  |
| --- |
| **SQL Server 2005 の SQLCLR は、開発生産性が向上するだけでなく、 パフォーマンスおよびセキュリティ向上のメリットもある。 サービス指向アーキテクチャ (SOA) で柔軟性を備えたシステム構築も可能** |

* MVP インタビューの第 4 回目は、NRIラーニングネットワーク株式会社の沖 要知 (おき としかず) さんにお越しいただきました。沖さんは、古くから SQL Server 関連のトレーニングに携わっておられ、マイクロソフトのパートナー向け特別トレーニング (mstep) でも講師としてご活躍されています。SQL Server 2005 のトレーニング コンテンツは、日本で 1 番早く実施され、その作成されたコンテンツは、SQLCLR**\*** や SOA を含めたデベロッパ向けから、BI (ビジネス インテリジェンス)、DBA (データベース管理者) 向け、パフォーマンス チューニングまで、幅広く手がけています。  
  \* : SQL Server 2005 に統合された、.NET 共通言語ランタイム

|  |  |
| --- | --- |
| * | **SQL Server のエンジンは、すべてのエディションで共通のソース コード。 セキュリティと開発生産性もほぼ共通の実装。 上位エディションへシームレスに移行できる柔軟性は本当にありがたい** |
| **――** | **まずは、沖さんの現在の業務と SQL Server との出会いについて教えていただけますか。** |
| **沖** | 現在は、SQL Server を中心としたトレーニング コンテンツを作成して提供するのが主な業務です。1986 年にこの業界に入りましたが、トレーナーを始めたのは Windows NT 3.1 の日本語版が出荷された時期 (1994 年初頭) になります。それ以前はメインフレーム環境でシステム管理や開発をしていました。当時は、IMS と呼ばれる階層型データベースや VSAM (IBM の仮想記憶アクセス方式) を利用するのが主流でしたが、メインフレーム環境だと、マニュアルを読んでも課題が解決しない場合、ベンダー SE にお伺いを立てて追加情報を入手しないと前に進めませんでした。また、莫大なコストがかかっているので当然なのですが、メインフレームが何をするにしても情報システム部門の管理下で行わなければならなかったことは、今の Windows 環境と比べると大きな差がありますね。 トレーナーを始めたきっかけは、当時所属していた会社の上司から「Microsoft Corporation で Windows NT 3.1 というネイティブな 32 bit OS を開発している。出向させてやるからローカライズの作業を手伝ってこないか」と言われました。これはチャンスだと思い、米国レドモンドの Microsoft 本社へ行かせてもらいました。おかげでたいへん貴重な経験をすることができ、この経験を活かすためにも Windows NT の講師をやってみようと思ったのが始まりでした。その後、元々のシステム開発経験も活かしたいと思って SQL Server や Oracle などデータベース関連のトレーニングに携わるようになりました。 |

|  |  |
| --- | --- |
| **――** | **沖さんは、古くからいろいろな製品に携わっていらっしゃいますが、SQL Server の魅力はどこにあるとお考えですか。** |
| **沖** | SQL Server は、1999 年に出荷された SQL Server 7.0 から第 2 世代と呼ばれ、今後 20 年間に通用するリレーショナル データベース エンジンとしてアーキテクチャが刷新されました。自動管理や自己最適化といった機能が提供され、使いやすく洗練された管理ツールが提供されたのもこのバージョンからです。この年に、コンピュータ業界のノーベル賞とも呼ばれるチューリング賞 (Turing Award) を受賞されたジム グレイ博士 (Dr. Jim Gray) が SQL Server 7.0 の製品発表会のために来日し、「データベース エンジンの全面的な設計の見直しを行ったこと」と「SQL Server のすべてのエディションのデータベース エンジンはすべて共通のソース コードを使用していること」を発表されたのが印象的でした。SQL Server は、それ以前のバージョンから気に入っていましたが、SQL Server 7.0 でさらに好きになりました。 SQL Server の魅力は、データベース エンジンと同様、セキュリティや開発生産性についても、すべてのエディションでほぼ共通に実装されている点です。これは、無償提供される Express Edition でアプリケーションを作成し、セキュリティを構築したとしても、上位のエディションへシームレスに移行できるということを意味しています。したがってデータの規模が増えたり、利用ユーザー数が増えたりしても、Express Edition から Workgroup Edition へ、あるいは Standard Edition、Enterprise Edition へと簡単に移行できるわけです。アプリケーションを修正することなく、上位エディションへ透過的に移行できる柔軟性は本当にありがたいですね。また開発者用には、最上位の Enterprise Edition とまったく同じ機能を備え、かつ非常に安価な Developer Edition も提供されているのもうれしいですね。 |

*

|  |  |
| --- | --- |
| * | **SQL Server 2005 で 1 番気に入っているのは SQLCLR。 開発生産性だけでなく、パフォーマンスとセキュリティ向上のメリットもある** |

*

|  |  |
| --- | --- |
| **――** | **いよいよ SQL Server 2005 日本語版が登場しましたが、SQL Server 2005 の機能の中で 1 番のお気に入りは何ですか。** |
| **沖** | SQL Server 2005 は、本当に多くの機能が追加されているので、気に入っている機能はたくさんあるのですが、中でも 1 番のお気に入りは「SQLCLR」です。SQLCLR は、.NET ランタイムを SQL Server のプロセスの中で実行できる機能です。これにより、データ層においても .NET の基本クラス ライブラリとプログラミング言語 (Visual Basic や C#) を使用してプロシージャ コードを書けるようになります。従来は、Transact-SQL では書きづらかったコード、もしくは書けなかったコードは、C や C++ 言語で Open Data Services API や Win32 API を使用して苦労しながら書いて、拡張ストアド プロシージャとして実装しなければなりませんでした。これが .NET で書けるようになったのは本当にありがたいですね。開発生産性が比較にならないほど向上しています。 また、データ層ではロジックを実装せずに、ビジネス層に持ち込んでコードを書いているような場合、データ アクセスの多いコードでは、データ層とビジネス層との間で非効率なラウンド トリップが発生している場合があります。そのような処理をデータ層に戻すことで、データ層とビジネス層とのラウンド トリップを減らし、パフォーマンスを向上させることができます。 SQL Server 2005 は、SQLCLR によってデータ層で複雑なアプリケーション ロジックが書けるようになり、アプリケーション プラットフォームとしての側面も大きく進化しました。 |

|  |  |
| --- | --- |
| **――** | **なるほど。SQLCLR で開発生産性が上がるだけでなく、パフォーマンス上のメリットもあるというわけですね。** |
| **沖** | はい。それに加えてセキュリティ上のメリットもあります。SQLCLR は .NET のコード アクセス セキュリティに準じた制御を行えるので、ビルドしたアセンブリを SQL Server のデータベースにロードする際に、アクセス許可セットを付与できます。既定では「SAFE」という最もセキュリティの高いアクセス許可セットに設定され、外部リソース (ファイルやネットワーク、環境変数、レジストリ) へのアクセスは禁止されるので、悪意のある不正なリソース アクセスを防ぐことができます。外部リソースへのアクセスを許可したい場合は「EXTERNAL\_ACCESS」、アンマネージドなコードの呼び出しも許可したい場合は「UNSAFE」というアクセス許可セットへ変更することもできます。 また、SAFE 以外のアクセス許可セットを使用する場合でも、事前に「コード署名」が要求されるので、セキュリティ面でも安心です。コード署名は、非対称キーか証明書を使用して行うのですが、その非対称キーや証明書をマップしたログインを作成できるので、管理者は特定の非対称キーや証明書に対してのみ EXTERNAL\_ACCESS 権限を許可できます。つまり、安全を確認できたコードに対してのみコード署名してデータベースにロードできるわけです。署名の検証が正常に行われないとアセンブリのロードに失敗しますので、セキュリティ上のリスクを伴うコードを安全に配置できますし、コードの改ざん防止にもなります。 |

|  |  |
| --- | --- |
| **――** | **それは魅力的ですね。コード署名があれば、本当に安全になりますね。** |
| **沖** | さらに、このコード署名は .NET アセンブリだけでなく、Transact-SQL で書かれたストアド プロシージャに対しても行えるようになっています。SQL Server 2005 はセキュリティが向上したと言われていますが、本当に安心して利用できるようになりました。 もちろん、生産性などの理由からコード署名の手続きを省略したい場合もあると思います。そういった場合でもデータベースに対して TRUSTWORTHY データベース プロパティを ON にすることで、コード署名されていないアセンブリでもロードできるという柔軟性もあります。安易に TRUSTWORTHY を ON にすべきではありませんが、柔軟性があるという点は SQL Server らしいですね。 |

*

|  |  |
| --- | --- |
| * | ***SOA (サービス指向アーキテクチャ) で 再利用性の向上と変化に耐えうる柔軟性を持った IT システムの実現** |

*

|  |  |
| --- | --- |
| * | Photo * |

|  |  |
| --- | --- |
| **――** | **その他に SQL Server 2005 で注目している機能はありますか。** |
| **沖** | やはり、サービス指向アーキテクチャ (SOA : Service-Oriented Architecture) への対応ですね。これは 1996 年にガートナーが提唱したインフォメーション テクノロジを組織化するための考え方です。今、非常に脚光を浴びているのですが、その理由には国内市場の閉塞感や市場競争の激化とグローバル化、消費者のライフ スタイルや価値観の多様化など、ビジネスを取り巻く変化に対応していくため、ビジネス スピードを向上させ、適切なビジネス モデルの変革、時には事業や企業の合併や再編、統廃合などを行っていくことが要求されています。そのためには、再利用性の向上や、変化に耐えうる柔軟性が IT システムにも求められています。 サービス指向アーキテクチャに基づくシステム構築は、それを実現するための技術として、インターネット基盤、Web サービス、XML といった標準技術の成熟も大きな要因となります。SQL Server 2005 は、データ層からサービス指向アーキテクチャへの対応を図っています。SQL Server の General Manager である David Campbell は「SODA」(Service-Oriented Database Architecture : サービス指向データベース アーキテクチャ) と呼んでいるのですが、「HTTP エンドポイント」と「ネイティブな XML データ型」「型指定のための XML スキーマ ライブラリ」「サービス ブローカによる非同期フレームワーク」が中核となる要素になります。 |

*

|  |  |
| --- | --- |
| * | **HTTP エンドポイントで IIS を使用しないで XML Web サービスを公開可能** |

*

|  |  |
| --- | --- |
| **――** | **SODA の構成要素を 1 つずつ説明していただけますか。** |
| **沖** | はい。まず HTTP エンドポイントですが、これはデータベースに置かれたストアド プロシージャやユーザー定義関数を直接 Web メソッド (XML Web サービス) として公開できる機能です。これは、CREATE ENDPOINT というステートメントを記述して実行するだけで実現でき、IIS は必要ありません。また、このステートメントによって WSDL も自動生成されるので、あとは Web サービス クライアントから利用するだけです。もちろん、ここで公開する Web メソッドの中身は .NET 言語で書けますし、暗号化や認証のメカニズムはフレームワークが提供するものを利用できます。 |

|  |  |
| --- | --- |
| **――** | **なるほど、IIS を必要としない Web サービス公開ができるわけですね。XML データ型についてはいかがですか。** |
| **沖** | これまでは、フロントエンドに Office System を配置して型指定された XML データを生成し、バックエンドのプロセスと連携させるため Web サービスを利用しようとすると、受け取った XML データをデータベースに格納するには、どこかで XML からクラスへ、そしてクラスからリレーショナル データへの変換が必要になりました。しかし SQL Server 2005 では、標準データとしてネイティブな XML 型が提供されるので、受け取った XML データをそのままテーブルへ格納することができます。 格納した XML データは「XQuery」で検索や加工を行えます。XQuery は W3C Query Working Group で標準化が進められている XML クエリ言語です。XPath の仕様も含んでいるのですが、特徴的なのは検索だけでなく、ノード セットの構造を変更した結果セットを返したり、指定された要素や属性の値を更新することもできることと、リレーショナル データとの相互運用をしやすい実装になっていることです。また、XML スキーマを使用して変数やテーブルの列に格納した XML データに対する型指定もできます。さらに、テーブルの XML 列に対してインデックスを作成できるので、クエリを最適化できるうえにパフォーマンスが向上します。 このようなフレームワークでは、使用する XML スキーマが明確であれば、それを InfoPath や VSTO (Visual Studio Tools for the Microsoft Office System) に組み込みます。そこから UI を作成し、受け取った XML データを SQL Server 2005 で公開している HTTP エンドポイントを経由して、そのまま同じスキーマで型指定された XML 列を持つテーブルに格納することができます。さらには、格納された XML データで必要な情報だけを XQuery で検索して加工し、レポーティングのソースとして利用するといった処理までをとても簡単に実装できます。 |

*

|  |  |
| --- | --- |
| * | **SQL Server 2005 は非同期フレームワークを標準提供。 サービス ブローカによってスケーラビリティが向上** |
| ***――** | **次は、SODA の構成要素の非同期フレームワークについてお願いします。** |
| **沖** | SODA は、非同期メッセージングのフレームワークも取り込んでいます。アプリケーションをインターネット上のサービスという形で公開した場合、通常 24 時間 365 日、絶えず機能を提供し続けることが期待されます。しかし、公開されたサービスに対するアクセスは、想像以上に負荷のかかるピーク タイムが存在することがあります。このような高負荷時に、通常時のワークロードに基づいて設計していると、処理能力を超えてしまってパンク状態になり、サービスを提供できない事態に陥ったりします。 このような場合、アプリケーションのスケーラビリティを高める方法の 1 つとして「非同期処理」の採用があります。非同期動作では、時間のかかる処理要求をキューに置き、独立したプロセスとして非同期に実行します。たとえば、商取引サイトにおいて会計処理でクレジット カード照会を行う場合、利用者の多いサイトでは照会サービスがボトルネックになる可能性があります。そこで、クレジット カード照会を非同期処理にすることで、スケーラビリティを高めることができます。 従来は、このような非同期なメッセージ処理を組み込もうとすると MSMQ や MQ シリーズのようなフレームワークを使用する必要がありました。 |

|  |  |
| --- | --- |
| **――** | **今までは SQL Server だけでは非同期処理が実現できなかったというわけですね。** |
| **沖** | はい。それが SQL Server 2005 では、「サービス ブローカ」と呼ばれるフレームワークが提供され、SQL Server だけで非同期処理を実現できるようになりました。サービス ブローカは、ストアド プロシージャの呼び出しを非同期で行える機能です。新しく、メッセージ型、コントラクト、サービス、キューといったデータベース オブジェクトが追加され、事前に定義しておいたメッセージがキューイングされると、関連付けられたストアド プロシージャが起動して、メッセージを読んで対応する処理を行うというしくみです。 これでビジネスのピーク時間帯が訪れても、追いつかない処理要求はキューイングしておくことができます。このサービス ブローカによるメッセージ ルーティングも、内部的にはエンドポイント機能が使用され、リモートのインスタンス間でも安全で信頼性の高いメッセージ交換を行えます。 サービス ブローカ自体は SQL Server 2005 に特化した実装ですが、メッセージ交換するサービス ブローカ プログラムは、ストアド プロシージャとして実装されるので、相互運用性が要求されれば、そのままサービス ブローカ プログラムを HTTP エンドポイントによって XML Web サービスとして公開することもできるわけです。 また、サービス ブローカは、SQL Server 2005 の通知サービス (Notification Services) で提供される「イベント通知」と「クエリ通知」という機能でも内部的に使用されています。[前回の長崎さんのインタビュー](http://www.microsoft.com/japan/sqlserver/2005/readystation/05/closeup.mspx)で話題となったプロアクティブ キャッシュ機能もそうですが、SQL Server 2005 ではプロアクティブな管理というのが 1 つの特徴となっています。このプロアクティブな動作を支えるフレームワークとして通知サービスが提供され、その構成要素としてサービス ブローカが使用されています。 |



講師として活躍中の沖氏

*

|  |  |
| --- | --- |
| * | **Visual Studio 2005 との統合により 開発、デバッグ、テスト、展開といった各サイクルがいずれも強化されている** |

*

|  |  |
| --- | --- |
| **――** | **Visual Studio 2005 との統合機能についてはいかがですか。** |
| **沖** | Visual Studio 2005 は、SQL Server 2005 と同時に出荷されると共に、.NET Framework 2.0 も同時に提供されます。SQL Server 2005 のツールは、Visual Studio 2005 のシェルに統合されることで、開発、デバッグ、テスト、展開といった各サイクルでいずれも強化されています。SQLCLR の開発、デバッグ、テストについては、Visual Studio 2005 で簡単に行えますし、展開 (データベース内への配置) も容易です。 また、Integration Services や Analysis Services、Reporting Services といったビジネス インテリジェンス系のプロジェクトでは、1 つのソリューションという形で束ねることができるので、客先で展開する場合などにたいへん便利です。これまでのように DTS パッケージは構造化ストレージ ファイルとして保存しないといけなかったり、OLAP キューブはバックアップして復元したり、レポート プロジェクトは Visual Studio プロジェクトで作成したりというような、バラバラな開発環境や展開手順であったのが統一されました。これはうれしいですね。 |

|  |  |
| --- | --- |
| **――** | **ADO.NET 2.0 についてはいかがですか。** |
| **沖** | ADO.NET 2.0 では「SqlDependency」と「SqlNotification」クラスが気に入っています。SqlDependency クラスは、通知のフレームワークを高度に抽象化していて、アプリケーション コードに非同期通知を組み込みたい場合にとても便利です。また、非同期通知を低レベルから操作したい場合は、SqlNotification クラスも提供されています。SQL Server の SODA フレームワークは .NET コードからも利用しやすいものになっています。 また、分散トランザクション コーディネータ (DTC) を隠蔽する .NET Framework 2.0 のトランザクション マネージャが利用できる「System.Transactions」名前空間のクラスも便利です。これまでビジネス層から分散トランザクション処理のコードを実行する場合、DTC に登録した COM+ コンポーネントを呼び出すために Enterprise Services による宣言型トランザクション管理のコードを記述する必要がありましたが、System.Transactions クラスを使用することで、分散トランザクション処理をより簡単に記述できるようになりました。 その他、データ プロバイダが強化されて、サーバーとクライアント間の送受信バイト数や実行時間、ラウンド トリップ回数などの統計情報を取得できるようになったことで、アプリケーションの最適化がしやすくなりました。これも本当に便利ですね。 |

*

|  |  |
| --- | --- |
| * | ***SQL Server 2005 はデータベース エンジンのソース コードと セキュリティや開発生産性に関する機能がすべてのエディションでほぼ共通の実装。 ホビー ユースから大規模な基幹系システムまでシームレスに利用できる** |

*

|  |  |
| --- | --- |
| **――** | **SQL Server 2005 はどういったユーザーにお勧めでしょうか。** |
| **沖** | 最初の話に戻ってしまいますが、SQL Server 2005 は、リレーショナル データベース エンジンのソース コード、セキュリティや開発生産性に関する機能が、すべてのエディションでほぼ共通の実装です。したがって、ホビー ユースから小規模、中規模な部門システム、大規模なエンタープライズ レベルの基幹系システムまで、共通したフレームワーク、コード、UI をシームレスに利用できます。SQL Server 2005 は非常に広いレンジで使用できる可能性があると思っています。 |
|  | * |

*Web ストリーミング配信として、SQL Server 2005 の SQLCLR 機能を、簡潔なデモ付きでわかりやすく解説しているムービーもありますので、ぜひこちらもご覧ください。

* [SQLCLR によるデータベース オブジェクト開発](http://www.microsoft.com/japan/seminar/SQLServer2005/databaseobject/play.aspx)

|  |
| --- |
| READY クローズアップ ***Microsoft MVP が語る SQL Server 2005**  *Microsoft MVP へのインタビューから SQL Server 2005 を探る READY クローズアップ。この連載では、SQL Server 2005 に対する期待や導入メリット、具体的な利用方法などについて、Microsoft MVP として表彰された方々へインタビューを行い、SQL Server 2005 に対する想いを語っていただきます。 **→*[Microsoft MVP とは](#c8_7) |
| |  |  | | --- | --- | | **MVP インタビュー 第 3 回 Microsoft SQL Server 2005 は、ココが違う**  **高度なビジネス インテリジェンス機能を備えた SQL Server 2005** | * | |

|  |
| --- |
| * |
|  | | * |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Index**   |  |  | | --- | --- | | * | [SQL Server 2005 の BI 機能はすばらしい。過去のプロジェクトで使用した他社製品と比べても、SQL Server 2005 の方が断然機能が豊富](#c3_1) | | * | [Integration Services が 1 番のお気に入り。単体製品として販売されていても、おかしくないくらい強力](#c3_2) | | * | [Integration Services は、運用管理ツールとしての側面もあります](#c3_3) | | * | [BI は「敷居が高い」「コストがかかる」というイメージがありますが、Analysis Services の UDM は、それを払拭するほどの機能を備えています。UDM を考えた人は本当にすごい](#c3_4) | | * | [プロアクティブ キャッシュによりリアルタイムに近いデータ分析が可能。従来までは、昨日までの集計結果しか参照できなかったのが、5 分前の集計結果を参照できるようになります](#c3_5) | | * | [Office Business Scorecard Manager でスコアカードを簡単構築。KPI で売り上げ目標の達成度などをグラフィカルに表示](#c3_6) | | * | [Reporting Services は、SQL Server 2000 版が出てすぐなのに多くの機能が追加されている](#c3_7) | | * | [小規模な環境でも SQL Server 2005 の BI 機能を、どんどん活用してほしいと思います](#c3_8) | | **長崎 友嘉 氏**    Microsoft MVP for SQL Server * 株式会社アイエイエフコンサルティング 前職のシステム開発会社で、主に SQL Server と ASP/ASP.NET のプロジェクトを担当。 SQL Server 2000 Analysis Services の案件を経験したことで情報系、分析系システムに興味を抱き、ビジネス インテリジェンス専業のコンサルティング企業であるアイエイエフコンサルティングに転職。 現在は、主に Microsoft 製品を使用した BI システムのコンサルティングとシステム構築を行っている。長崎友嘉氏 | | |  |  | | --- | | * | | | |

*

|  |
| --- |
| **BI は「敷居が高い」「コストがかかる」というイメージがありますが、 SQL Server 2005 の BI 機能は、それを払拭するほどの機能を備えています。 SQL Server 2005 を購入すれば、BI に必要な機能はすべて付いてきます** |

*  
MVP インタビューの第 3 回目は、BI (Business Intelligence : ビジネス インテリジェンス) システムおよびデータ ウェアハウスの設計と構築、チューニングを得意とする長崎 友嘉 (ながさき ともよし) さんにお越しいただきました。長崎さんは、SQL Server ユーザーのためのコミュニティ「PASSJ」で BI 分科会のボード リーダーを勤め、マイクロソフトおよび PASSJ の Web サイト上では SQL Server 2000 Analysis Services の連載記事を執筆されています。また、PASSJ のセミナーでは講師としてもご活躍されています。今回のインタビューでは、BI のプロフェッショナルの目から見た SQL Server 2005 の BI 機能に注目しました。

*

|  |  |
| --- | --- |
| * | **SQL Server 2005 の BI 機能はすばらしい。 過去のプロジェクトで使用した他社製品と比べても、 SQL Server 2005 の方が断然機能が豊富** |

*

|  |  |
| --- | --- |
| **――** | **まずは、長崎さんの現在の業務と SQL Server とのかかわりについて教えていただけますか。** |
| **長崎** | 現在の業務は、BI (ビジネス インテリジェンス) システムを含めたデータ ウェアハウスの設計と開発が中心になります。SQL Server だけでなく、Oracle や他社のデータ ウェアハウス製品、他社の BI ツールを利用した案件にも携わり、いろいろなシステムを幅広く手がけています。 数年前までは Oracle による開発と運用管理を中心に行っていたのですが、最近は SQL Server の案件に携わることが多くなりました。SQL Server に本格的に携わりだしたのは SQL Server 2000 からで、当時在籍していた会社で SQL Server と ASP (Active Server Pages) による Web サイトの構築を手がけたのが始まりです。その後、Analysis Services を利用した BI システムのチューニングを行ったり、ASP.NET による ERP パッケージ開発のプロジェクトでは、BI に関する部分をごく少人数で構築したりしました。そういった背景からデータ活用およびデータ分析のシステムに興味を抱き、ビジネス インテリジェンス専業のコンサルティング企業へ転職しました。 |

|  |  |
| --- | --- |
| **――** | **長崎さんは、Oracle を初めとして、いろいろな製品に携わっておられますが、SQL Server の魅力はどこにあるとお考えですか。** |
| **長崎** | SQL Server は、「安くていいのがあるらしい」という噂を聞いたのが最初でした。実際に使ってみると「Oracle に比べてやることが少なく、とっても使いやすい」という印象を持ちました。たとえば Oracle では、最初の導入の段階でものすごく慎重かつ細かい設計をしてガチガチに固めないと駄目なのですが、SQL Server の場合は柔軟性が高く、設計はほどほどでもよいので、すぐに導入、利用できます。それに、自動チューニング機能のおかげで、日々の運用管理もとても楽です。ほとんど何もしなくても運用できるというのは本当に驚きでした。 SQL Server 2005 の 1 番の魅力は BI 機能ではないかと思います。正直 SQL Server 2000 のときは、他社の製品と比べると弱い部分もあったのですが、SQL Server 2005 では驚くほど機能が強化されていて、ものすごくよくなっています。 また、今まで過去のプロジェクトで使用した他社の製品と比べても、断然機能が豊富ですね。Analysis Services の UDM (Unified Dimensional Model : 統合ディメンショナル モデル) とプロアクティブ キャッシュのような機能は他社製品にはありませんし、Integration Services (従来の DTS) を含めた BI の総合力は、圧倒的に他社製品に勝っています。しかも、SQL Server 2005 の BI 機能は、SQL Server 2005 の Standard Edition 以上であれば同梱されているので、コスト的なメリットも大きいですね。今になって振り返ると、過去のプロジェクトで SQL Server 2005 が利用できればどんなに楽だっただろうという場面がたくさん思い浮かびます。 |

|  |  |
| --- | --- |
| * | ***Integration Services が 1 番のお気に入り。 単体製品として販売されていても、おかしくないくらい強力** |

*

|  |  |
| --- | --- |
| * | Photo * |

|  |  |
| --- | --- |
| **――** | **もう間もなく SQL Server 2005 の日本語版が登場しますが、SQL Server 2005 の機能の中で 1 番のお気に入りは何ですか。** |
| **長崎** | Analysis Services 機能も気に入っていますが、何といっても Integration Services 機能ですね。Integration Services は、従来は DTS (データ変換サービス) と呼ばれていた機能で、DTS のときから気に入っているのですが、これを利用すると異機種間でのデータ転送を簡単に行えます。SQL Server どうしでのデータ転送はもちろん、Oracle から SQL Server へデータ転送したり、メインフレーム上のファイルを SQL Server へ取り込んだりすることができます。また、単なるデータ転送だけでなく、データ変換も定義できるので、コード化されたデータを分析しやすい形へ変換したり、複数の列を連結したり (姓と名を連結して氏名を作ったり) することもできます。データ ウェアハウスを構築する際に、データの Extract (抽出)、Transform (変換)、Load (ロード) が行える、いわゆる「ETL ツール」として利用できるのが Integration Services です。 私自身、他社製品を利用したプロジェクトのときに、ETL 処理の部分を自分たちで 1 から開発し、実装したことがあるのですが、Integration Services があれば、わざわざそういった開発は必要ありません。SQL Server 2005 に標準で搭載されているというのは本当にうれしいですね。 |

|  |  |
| --- | --- |
| **――** | **私たちも Integration Services が大好きです。Oracle だと ETL ツールは別プロダクトなので別途費用がかかりますし、さらに追加機能が別ライセンスというのが非常に多いですね。SQL Server の Analysis Services に相当する OLAP 機能も、Oracle では Enterprise Edition のオプション機能として提供されています。Enterprise Edition のオプションとなると、オプションの値段に加えて、高い Enterprise Edition のライセンスも購入しなければなりません。その点、SQL Server であれば、Integration Services や Analysis Services は Standard Edition 以上に同梱されているので、本当にお得だと思います。** |
| **長崎** | そうですね。Oracle はともかく、ETL 専用製品は、百万単位でコストがかかります。従来の DTS ですと、そういった製品と比べて機能的に劣る部分もあったのですが、Integration Services ではまったく遜色ありません。単体製品として販売されていても全然おかしくないくらい強力ですね。 他社のデータベース製品を利用していると、コストの制約上、ETL ツールを購入できないという場合もあります。そうなると自力で開発しなければなりません。実際、そういった開発をしているというケースは本当に多いと思います。 |

|  |  |
| --- | --- |
| **――** | **Integration Services でこれはイチ押しという機能はありますか。** |
| **長崎** | イチ押しは、複数のデータ ソースを簡単に結合したり、分岐したりできるようになった点ですね。通常、データ ウェアハウスを構築する際には、「スター スキーマ」という構成をとるのですが、これにはデータ ソースの結合や複数テーブルの結合が欠かせません。従来の DTS では、ルックアップという機能を使うことで、複数のテーブルを結合することはできたのですが、ユーザー インターフェイスがわかりづらく、使い勝手は良くありませんでした。また、ルックアップ機能では、データ ソースが CSV ファイル (カンマ区切りのテキスト ファイル) の場合は、利用できないという制約もありました。これでは、複数のデータ ソースをいったん SQL Server 上に取り込んだ後に、別途結合を行うためのタスクを追加する必要があり、大変でした。 しかし、SQL Server 2005 の Integration Services であれば、テキスト ファイルでもどんなデータ ソースでも簡単に結合できます。ツールも直感的で使いやすくなったので、本当に楽になりました。また、余分な処理をしなくて済む分、パフォーマンスも向上します。 Integration Services は、データ転送時のログが取れる点やエラー処理 (イベント ハンドラ) を記述できるようになった点、複数のデータ転送をパラレル処理できる点なども気に入っています。 |

*

|  |  |
| --- | --- |
| * | **Integration Services は、運用管理ツールとしての側面もあります** |

*

|  |  |
| --- | --- |
| **――** | **Integration Services は、DBA (データベース管理者) 向けの機能も充実していますね。** |
| **長崎** | はい。データベースのフル バックアップやログ バックアップ、インデックスの再構築、整合性のチェックといった従来の保守計画ウィザードで提供されていた機能が Integration Services のタスクとして提供されるようになったので、1 つ 1 つ実行できて、ワークフローも書けるようになりました。新しい機能としては、差分バックアップやインデックスのデフラグ (再編成)、履歴のクリーンアップ、Visual Basic .NET で記述可能なスクリプト タスク、SMTP でのメール送信が行えるタスクも追加されているので、データベース管理者にとってはうれしいでしょう。Integration Services は、運用管理ツールとしての側面もありますね。 |

|  |  |
| --- | --- |
| * | ***BI は「敷居が高い」「コストがかかる」というイメージがありますが、 Analysis Services の UDM は、それを払拭するほどの機能を備えています。 UDM を考えた人は本当にすごい** |

*

|  |  |
| --- | --- |
| **――** | **Analysis Services についてはいかがでしょうか。** |
| **長崎** | Analysis Services は、BI システムの中核となる「キューブ」を作成する機能です。キューブは、多次元データベースとも呼ばれ、いろいろな角度からの集計結果を格納し、データ分析を高速化するためのものです。たとえば、売り上げ分析であれば、商品や店舗、地域、顧客、年度ごとに売り上げ額やコスト、利益などを集計したものがキューブです。 Analysis Services で注目すべきは、何といっても UDM (統合ディメンショナル モデル) です。UDM のおかげで、スキーマを柔軟に構成できるので、キューブを簡単に構築できるようになりました。BI やデータ ウェアハウスは、「敷居が高い」「コストがかかる」というイメージを持たれる方が多いのですが、Analysis Services の UDM は、それを払拭するほどの機能を備えています。 たとえば、SQL Server 2000 では、Analysis Services 専用のスキーマをうまく作らないと、データ分析に役立つ実践的なキューブは作れなかったり、複数のデータ ソースを組み合わせてキューブを作るためには仮想キューブを構成する必要がありました。これでは、キューブを作るための事前設計に高いスキルが必要になり、一部のユーザーにしか使いこなせません。しかし、SQL Server 2005 の UDM では、複数データ ソースへの対応や属性ベースの分析、マルチ ファクト対応など、スキーマを柔軟に構成できるようになりました。 |

|  |  |
| --- | --- |
| **――** | **確かに SQL Server 2000 のときは、キューブを作成する前の段階で大変でした。事前にいかにデータ ウェアハウスを構築しておくかが鍵でしたね。** |
| **長崎** | はい。特にデータ ウェアハウスの構築には、スター スキーマ構成が欠かせませんので、ここに大きな壁がありました。そういったところで BI への敷居の高さを感じている方が多いようです。しかし UDM では、スキーマを柔軟に構成できるので、スター スキーマを構成していなくても、簡単にキューブを構築できます。現在のデータ構造を基に、ディメンション (分析軸) とメジャー (集計対象) の関係 (一対多、多対多、親子、参照など) を、ウィザードが自動認識してくれるので、何も考える必要がないわけです。極端な話、業務システムのデータ モデルそのままでも、ウィザードを実行するだけでキューブを構築できます。UDM を考えた人は本当にすごいと思います。 もちろん、パフォーマンスなどを考慮するとスター スキーマ (とデータ ウェアハウス) を構成することが基本であることに変わりはありませんが、スター スキーマを構成しなくてもキューブを作れるようになったというのは大きいですね。本当に敷居が低くなりました。小規模な環境であれば、わざわざデータ ウェアハウスを構築しなくてもパフォーマンス的に問題ない場合も多いので、そのような企業にはぜひ試してほしいと思います。 |

|  |  |
| --- | --- |
| **――** | **スター スキーマを構成するとしても Integration Services がありますね。** |
| **長崎** | そうですね。先ほどの話のように、Integration Services はスター スキーマ構成を作るのに本当に役立ちます。また、Integration Services には、SCD (Slowly Changing Dimension : 緩やかに変化するディメンション) タスクと呼ばれる便利な機能も提供されています。これは、商品や顧客の属性といったディメンションの変更管理をするためのタスクで、たとえば、ある顧客のランクが B から A へ変わったときに、ランクが B のときの売り上げと、ランクが A に変わってからの売り上げを同一視するのか、それとも変更の履歴を残すのか (変更前後の売り上げを比較したいのか) などを実現できます。SCD は、スター スキーマの提唱者である Ralph Kimball 氏の著書「The Data Warehouse Toolkit」で触れられている考え方なのですが、これが Integration Services に実装されているわけですから本当に便利なツールですね。 |

|  |  |
| --- | --- |
| * | ***プロアクティブ キャッシュによりリアルタイムに近いデータ分析が可能。 従来までは、昨日までの集計結果しか参照できなかったのが、 5 分前の集計結果を参照できるようになります** |

*

|  |  |
| --- | --- |
| Photo * | * |

|  |  |
| --- | --- |
| **――** | **そのほかに Analysis Services で注目されている機能はありますか。** |
| **長崎** | 他社製品にはない Analysis Services ならではの特徴は、集計のレベルをカスタマイズしたり、リアルタイムに近い集計結果を参照できる点でしょう。集計レベルをカスタマイズできれば、アクセス頻度の高い集計結果だけを保持しておけばよいので、キューブが肥大化するのを防ぎ、キューブ生成時間も短くて済みます。 また、リアルタイムに近いデータ分析は、「プロアクティブ キャッシュ」という機能によって実現可能になります。これにより、従来までは昨日までの集計結果しか参照できなかったのが、たとえば 5 分前の集計結果を参照できるようになります。できる限り最新のデータで分析したいというお客さんは多いのですが、Analysis Services ならそれに応えることができます。 プロアクティブ キャッシュの内部的な動作は、「SQL プロファイラ」ツールを使って監視することもできます。従来の Analysis Services は、裏で何をやっているのかがわかりづらかったのですが、SQL Server 2005 では SQL プロファイラですべて確認できるのです。しくみがわかると安心して利用できるので、これはうれしいですね。 |

|  |  |
| --- | --- |
| **――** | **Analysis Services はサーバーとしての機能が充実しているというわけですね。** |
| **長崎** | はい。ほかにも、Time Intelligence や Account Intelligence、キューブのスクリプト生成、データ マイニング機能ではアルゴリズムが多数追加されているなど、たくさんのサーバー機能が提供されています。これは Analysis Services の強いところですね。 Time Intelligence と Account Intelligence は、従来は MDX (Multi Dimensional Expressions) と呼ばれる特殊なステートメントを使って書かなければならなかった処理を自動生成してくれる機能です。たとえば、売り上げ額や利益、コストなどの前月比成長率や前年比成長率、年度累計、移動平均などを求めるには、従来は MDX を使って手で書く必要があったのですが、SQL Server 2005 では Time Intelligence 機能が提供されるので、ウィザードを実行するだけで MDX を自動生成してくれるのです。 キューブのスクリプト生成は、キューブの定義を XML 形式のファイルとして出力してくれる機能です。これは、本番環境からテスト環境へキューブを移動したい場合や、作成したキューブをテンプレートとして使いたい場合などで役立ちます。今までは、キューブの定義を移動するのに、リソース キットのツールを使ってキューブ定義を抽出するか、バックアップと復元の機能を使うしかありませんでした。スクリプト生成は、標準形式である XML for Analysis 形式で定義を抽出できますし、定義の反映も簡単に行うことができます。これもたいへん便利な機能です。 |

*

|  |  |
| --- | --- |
| * | ***Office Business Scorecard Manager でスコアカードを簡単構築。 KPI で売り上げ目標の達成度などをグラフィカルに表示** |

*

|  |  |
| --- | --- |
| **――** | **Analysis Services のクライアントという視点からはいかがですか。** |
| **長崎** | 何といっても Office Business Scorecard Manager 2005 ですね。これは、BSC (バランス スコアカード) という経営手法で利用されるスコアカードを簡単に構築できるツールです。スコアカードは、売り上げ目標に対する達成度や、利益、コスト、株価といった戦略目標に対する達成の度合いを一覧表示するためのものです。 戦略目標の実現を測る基準は KPI (Key Performance Indicator : 主要業績評価指標) と呼ばれますが、Office Business Scorecard Manager では、KPI の作成や数値目標の設定、キューブとの関連付けなどをグラフィカルに設定できます。また、スコアカードを参照するツールは、SharePoint の Web パーツとして提供されるので、Web ブラウザだけあれば大丈夫です。これは本当に便利ですね。 |

|  |  |
| --- | --- |
| * | ***Reporting Services は、SQL Server 2000 版が出てすぐなのに 多くの機能が追加されている** |

*

|  |  |
| --- | --- |
| **――** | **Reporting Services についてはいかがですか。** |
| **長崎** | Reporting Services は、レポートの作成と管理、配布が行える Web ベースのレポーティング ツールですが、もともと SQL Server 2005 で提供される予定だったものを SQL Server 2000 版で先に提供されたものでした。SQL Server 2000 版が出てすぐにもかかわらず、多くの機能が追加されているのには、驚いています。 たとえば、Analysis Services のクライアントとしての機能では、従来は MDX ステートメントを手で書く必要があったのが、SQL Server 2005 では GUI ベースでグラフィカルにクエリを書けるようになったり、MDX のパラメータにも対応しました。また、複数のパラメータへの対応や印刷機能の強化、SharePoint Web パーツの提供、レポート ビルダによるエンド ユーザー レベルでのレポート作成、Report コントロールなど、実に多くの機能が追加されています。 |

|  |  |
| --- | --- |
| * | ***小規模な環境でも SQL Server 2005 の BI 機能を、 どんどん活用してほしいと思います** |

*

|  |  |
| --- | --- |
| **――** | **最後にこの記事を読んでいただいている皆様にひと言お願いします。** |
| **長崎** | 繰り返しになってしまいますが、BI やデータ ウェアハウスは、「敷居が高い」「コストがかかる」というイメージを持たれている方が多いのですが、SQL Server 2005 の BI 機能はそれを払拭するほどの機能を備えています。SQL Server を購入すれば、BI に必要な機能はすべてついてきます。Integration Services と Analysis Services、Reporting Services は、1 つのライセンスがあれば利用できます。今までシステムにそれほどコストをかけられなかった企業でも、ぜひ試してほしいと思います。インストールもすごく簡単ですし、クライアントは Excel があれば分析できるので、気軽に設計してまずは試してみていただければと思います。 また、大規模なシステムでも、プロアクティブ キャッシュによって、リアルタイムに近いタイミングでの分析結果を、高いクエリ パフォーマンスを上手に保持しつつ提供し続けるしくみが用意されています。Integration Services もたいへん強力なツールです。今まで手作りで作成していたものが、GUI ベースで、ノンプログラミングで実現できます。ぜひ試してみていただければと思います。 |

現在、Web ストリーミング配信として、SQL Server 2005 の BI 機能を短く簡潔に、デモ付きでわかりやすく解説しているムービーもありますので、ぜひこちらもご覧ください。

* [Integration Services によるデータ統合の実現](http://www.microsoft.com/japan/seminar/SQLServer2005/integrationservices/play.aspx)
* [Analysis Services によるリアルタイム データ分析の実現](http://www.microsoft.com/japan/seminar/SQLServer2005/analysisservices/play.aspx)
* [Reporting Services によるデータ活用](http://www.microsoft.com/japan/seminar/SQLServer2005/ReportingServices/play.aspx)

|  |
| --- |
| READY クローズアップ ***Microsoft MVP が語る SQL Server 2005** * Microsoft MVP へのインタビューから SQL Server 2005 を探る READY クローズアップ。この連載では、SQL Server 2005 に対する期待や導入メリット、具体的な利用方法などについて、Microsoft MVP として表彰された方々へインタビューを行い、SQL Server 2005 に対する想いを語っていただきます。 * *→*[Microsoft MVP とは](#c8_7) * |
|  |

**MVP インタビュー 第 2 回  
Microsoft SQL Server 2005 は、ココが違う  
パフォーマンスと拡張性が向上した SQL Server 2005**

|  |  |
| --- | --- |
|  | * |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Index**   |  |  | | --- | --- | | *[SQL Server の魅力は何といってもイージー アンド リッチ](#c2_1) |  | | *[同じ SQL 文を実行しても SQL Server 2005 の方が断然速い](#c2_2) |  | | *[デベロッパーにお勧めの機能はデータベース エンジン チューニング アドバイザ](#c2_3) |  | | *[動的管理ビューで詳細な監視 SQL Server 2005 はブラック ボックスではない](#c2_4) |  | | *[パーティション テーブルは今すぐにでも使いたい機能　エンタープライズ案件にはなければ困ります](#c2_5) |  | | *[SQL Server 2005 は、金融や証券のシステムなど大規模なエンタープライズ環境でも十分に通用します](#c2_6) |  | | **小川 貢 氏** Microsoft MVP for SQL Server * 株式会社 CRC ソリューションズ　ビジネス ソリューション第 1 事業部　ソリューション技術部 テクノロジー ソリューション課。 前職にて ASP (Active Server Pages) によるサイト構築を担当した後、現在の会社においても Web 関連の開発に従事。 仕事上、SQL Server を利用する機会が多いことから、PASSJ の立ち上げに関与。現在 PASSJ の理事および Web テクノロジー分科会ボード リーダーとして活動。 PASSJ での活動が Microsoft に認められ、2001 年に MVP for SQL Server として認定され、現在に至る。 Microsoft のプロダクトの中で何をおいても SQL Server が大好き。小川貢氏 | | |  |  | | --- | | * | | |

*

|  |
| --- |
| **SQL Server 2005 の向上したパフォーマンスは、今までなかなか踏み込めなかった金融や証券などの大規模なエンタープライズ環境でも十分に通用します。** |

「Microsoft MVP が語る」シリーズの第 2 回目となる今回は、Web アプリケーションの設計と開発、パフォーマンス チューニングを得意とする株式会社 CRC ソリューションズの小川 貢 (おがわ みつぎ) 氏をお招きし、SQL Server 2005 に対する熱い想いを語っていただきました。小川さんは、SQL Server ユーザーのためのコミュニティ「PASSJ」の立ち上げ時のメンバであり、現在は、PASSJ の理事および Web テクノロジー分科会ボード リーダーとして活躍されています。小川さんは、日本における Microsoft MVP 制度の立ち上げ時の初期メンバ (7 人) のうちの 1 人でもあり、古くからコミュニティ活動を積極的に行っています。



|  |  |
| --- | --- |
| * | ***SQL Server の魅力は 何といってもイージー アンド リッチ** |

*

|  |  |
| --- | --- |
| **――** | **現在の小川さんの業務と SQL Server の関係について教えていただけますか。** |
| **小川** | 私の業務は、Web アプリケーションの設計と開発が中心になります。.NET や Javaによる開発、そしてデータベースでは SQL Server だけでなく Oracle も利用し、小規模なシステムから大規模なエンタープライズ案件まで、いろいろなシステムを幅広く手がけています。それとは別に、いわゆる「火消し」役として、他の人の開発を手伝ったり、技術支援を行ったりすることも多いです。 |

|  |  |
| --- | --- |
| **――** | **さまざまなテクノロジに携わっておられる中で、小川さんは常日ごろ「SQL Server が大好き」とおっしゃられていますが、SQL Server の魅力はどこにあるとお考えですか。** |
| **小川** | SQL Server の魅力は、何といってもイージーな点でしょう。インストールを 1 つとっても SQL Server なら簡単です。もし UNIX 環境で Oracle となると、OS のインストールが大変、Oracle のインストールも大変、インストール前後の各種パラメータ設定も大変、開発をするにも別途 Oracle Net Client (Net8 Client) のインストールが必要になる、といった具合に多くの知識が必要となります。しかし、SQL Server であれば、開発に必要なコンポーネントは OS に標準で入っているので、誰にでも簡単にインストールできます。 イージーというのは、決してチープという意味ではなく、リッチなイージーという意味です。たとえば、SQL Server の自動チューニング機能は本当にリッチです。普通は、自動チューニングというと「本当に大丈夫なの？」と疑いたくなりますが、SQL Server の自動チューニングは本当に賢い。必要に応じてメモリの使用量が動的に変化したり、物理メモリ以上のメモリを消費しないように動作したりと非常によくできていて、安心して使えます。 |

|  |  |
| --- | --- |
| **――** | **確かに安心して利用できますね。最近の Oracle にも動的チューニングの機能はありますが、SQL Server における自動チューニングとは違って、ある程度手動でパラメータの調整をする必要があり、そのためには高いスキルが必要です。これではデータベースに詳しくないユーザーが安心して利用するには、専門のコンサルタントを手配したり、サポートに問い合わせなければなりません。** |
| **小川** | そうですね。それに SQL Server の場合には、何か問題があったとしても、監視をするためのツールが充実されています。贅沢なくらいに Enterprise Manager を始めとする GUI ツールが提供されていて、クエリ アナライザの「実行プランのグラフィカル表示」機能を 1 つとっても実に良くできています。その中でも私が特に気に入っているツールは「プロファイラ」ですが、これを使えば、実行中の SQL 文をトレース (記録保存) できるので、パフォーマンスに問題があった場合でも簡単にボトルネックが発見できるようになります。 Oracle だとこうはいかなくて、サード パーティ製の高価なツールを利用したり、複雑な管理用の SQL 文をたたかないとボトルネックを発見できないわけです。今でこそ Oracle も GUI ツールを備えるようになってきましたが、SQL Server が 7、8 年前から実装している機能がほとんどです。それに最近の Oracle の GUI ツールは Web ベースなので、操作性では SQL Server のツールの方がはるかに馴染みやすいと言えるでしょう。 このように SQL Server は、イージーで敷居が低いにも関わらず、多機能で動作の信頼性もあり、専門のデータベース管理者でなくても気軽に利用できるというのが大きな魅力です。 |

|  |  |
| --- | --- |
| * | ***同じ SQL 文を実行しても SQL Server 2005 の方が断然速い** |

*

|  |  |
| --- | --- |
| Photo * | * |

|  |  |
| --- | --- |
| **――** | **SQL Server 2005 の製品発表が近づいてきましたが、SQL Server 2005 のどこに期待されていますか。** |
| **小川** | 何をおいてもパフォーマンスの向上でしょう。インデックスでは、インデックスの Include オプション (付加列インデックス) やオンラインでのインデックス再構築など、多くの機能が追加されています。 パフォーマンスを検証するため、私の環境では、同じマシンに SQL Server 2000 と SQL Server 2005 の両方をインストールしており、実際の検証時にはどちらかの SQL Server だけを起動させています。また、検証で使用するデータは、お客様の実際のデータとだいたい同じものを用意し、テストで使用する SQL 文もできる限り実際と同じものにしています。試しに数千万件のデータを使って、SQL Server 2000 と SQL Server 2005 で同じ SQL 文を実行してみると、SQL Server 2005 の方が断然速いのです。これはエンジンが強化されたことも 1 つの要因でしょうが、インデックスの Include オプションの恩恵でもあるでしょう。しかもこれはベータ版の段階でのテストなので、出荷される製品版ではさらにパフォーマンスが向上していると期待ができます。 |

|  |  |
| --- | --- |
| **――** | **そのほか「これは速い」と体感された機能はありますか。** |
| **小川** | Transact-SQL の拡張機能の「共通テーブル式」ですね。共通テーブル式 (CTE : Common Table Expressions) は、再帰クエリを実現でき、階層構造を持ったデータを簡単に取り出せるようになります。今までは、階層構造を取り出すのに tempdb データベース内に一時テーブルを作ったり、自己結合をしたりしなければならなかったのが、共通テーブル式であれば非常にシンプルな SQL 文で書けます。かつ、パフォーマンスも良いのです。これは、掲示板でよく利用されるスレッド表示機能など、再帰的にループで回すような処理に本当に便利です。 また、新しく追加された RANK 関数と ROW\_NUMBER 関数は、売り上げなどのランキングを表示するときに役立つのですが、こういったプログラミング的な要素が拡張されたことも、パフォーマンス向上の一助になっている。よく SQL Server 2005 は速くなったと聞くけれど、「なるほど、こういうことか」というのが体感できたのです。 |

|  |  |
| --- | --- |
| * | ***デベロッパーにお勧めの機能は データベース エンジン チューニング アドバイザ** |

*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **――** | **パフォーマンス チューニングの観点から SQL Server 2005 で気に入っておられる機能はありますか。** | | **小川** | データベース エンジン チューニング アドバイザは、デベロッパにお勧めの機能ですね。デベロッパにとっては、クエリ アナライザで表示された実行プランを基に最適なインデックスを見つけ出すという作業は、たいへんに労力がいります。そもそも一般的なエンジニアにとって実行プランは難解であり、実行プランの解析には高いスキルが必要です。しかし、データベース エンジン チューニング アドバイザであれば、チューニングしたい SQL 文を読み込ませるだけで、最適なインデックスを提示してくれるのです。必要であればパーティションやインデックス付きビューも提示してくれます。 データベース エンジン チューニング アドバイザは、インデックス チューニングという職人芸の世界を一般的なエンジニアでもできるようにした本当に便利な機能ですね。それでも駄目な場合は、職人を呼びましょう。 |  |  |  | | --- | --- | | * | ***動的管理ビューで詳細な監視 SQL Server 2005 はブラック ボックスではない** |   *   |  |  | | --- | --- | | **――** | **小川さんは、その職人という立場になると思いますが、データベースの専門家としてSQL Server 2005でうれしい機能はありますか。** | | **小川** | より詳細なチューニングには「動的管理ビュー」が欠かせないでしょう。動的管理ビュー (DMV : Dynamic Management View) は、従来の DBCC コマンドやシステム ストアド プロシージャに置き換わるものなのですが、SQL Server 2000 と比べて調べられる項目が圧倒的に増えています。たとえば、tempdb の利用状況を詳しく調べたり、メモリやインデックスの利用状況を詳しく調べたりできるようになっています。未使用のインデックスも調べられるので、不要なインデックスを探すのも簡単になりました。よく「SQL Server 2000 はブラック ボックスだ」と言われていたのですが、SQL Server 2005 ではそうではないのです。 |  |  |  | | --- | --- | | **――** | **ブラック ボックスと言えば、SQL Server 2000 にはヘルプ (Books Online) に記載されていない隠しコマンドもありますね。** | | **小川** | そうですね。SQL Server 2000 では一部の Top Gun と呼ばれる人たちしか知らないアンドキュメンテッドなコマンドやトレース フラグが結構あり、これらを使ってチューニングすることもできました。しかし、こういった隠しコマンドは正式にはサポートされていないので、たとえばコマンドを打ってデータベースが壊れたとしても保証がないわけです。これでは、いくらチューニングに役立つとはいっても一般ユーザーには怖くて利用できません。 一方、動的管理ビューであればヘルプに記載されているので、安心して利用できます。また、動的管理ビューは、出力結果がテーブル形式なので、結果を 2 次利用するのも簡単です。たとえば、ある監視項目に対してしきい値を超えた場合にメールでアラートを送信するといったこともできますし、インデックスの断片化の状態を調べて、「断片化が 30% 以上ならインデックスの再構築を実行する」といったように、結果を基に別のアクションを起こすことも簡単にできます。従来の DBCC コマンドやシステム ストアド プロシージャでは、監視結果を 2 次利用するには一時テーブルをわざわざ作らなければならなかったり、そもそも 2 次利用できないコマンドもあったりしました。この辺の違いは、データベース管理者 (DBA) にとっては大きな差となるでしょう。 |  |  |  | | --- | --- | | **――** | **そのほかパフォーマンス関連で注目している機能はありますか？** | | **小川** | スナップショット分離レベル (Snapshot Isolation) ですね。これは Oracle でいうところの「読み取り一貫性」と同じように、排他ロック中のデータを参照できる機能です。会計システムなどでは、バッチ処理をすると、結構な数のロックがかかり、場合によってはテーブル単位のロックがかかることもあります。その間にリアルタイムでデータを参照したいというニーズは結構あるわけです。そういった場合でもスナップショット分離レベルを利用すれば、ロック中のデータでも、待たされることなくすぐに参照できるようになります。これは絶対お勧めですね。 ただし、スナップショット分離レベルを利用する場合は、tempdb データベース内にスナップショット データを作成することになるので、tempdb の監視や物理配置もきちんと考えておかなければなりません。tempdb データベースは、できる限りハード ディスクの数をたくさん用意した RAID 0+1 上に配置し、I/O のスピードを上げておくことが重要です。 |   *   |  |  | | --- | --- | | * | ***パーティション テーブルは今すぐにでも使いたい機能 エンタープライズ案件にはなければ困ります** |   *   |  |  | | --- | --- | | * | Photo * |  |  |  | | --- | --- | | **――** | **拡張性 (スケーラビリティ) という観点からは、どういった機能に注目されていますか。** | | **小川** | パーティション テーブルですね。これは Oracle でいうところの「レンジ パーティション」機能で、データの範囲 (レンジ) によってデータの格納場所 (パーティション) を分ける機能です。これは、SQL Server 2000 の分割ビュー機能でも似たようなことができたのですが、運用管理面では大きな違いがあります。 パーティション テーブルは、データの消しこみ作業やデータのアーカイブを作成する場合にたいへん役立ちます。もし、パーティション機能を利用せずに、1 つのテーブルで運用していた場合のデータのアーカイブ (保存) を作成する作業を考えてみてください。この場合は、データをアーカイブへコピーした後、基のテーブルからデータを削除する作業が発生します。削除するデータが大量になると、トランザクションが長くなりますし、トランザクション ログは膨れ上がります。また、ロックも大量にかかることになり、テーブル単位のロックにエスカレーションすることもあり得ます。このような事態を回避する方法は、削除するデータを少しずつ選択し、Wait をかけながら少しずつ消していくというものです。これにより、トランザクションを短く、ロックをかける範囲を小さくできます。しかし、これでは削除に時間がかかってしまいますし、運用がたいへんです。このような状況でパーティション テーブルを利用すれば、データのアーカイブは一瞬で終わります。パーティションを切り離すだけでいいのです。これは、今すぐにでも使いたい機能ですね。 |  |  |  | | --- | --- | | **――** | **運用管理面のメリットが大きいというわけですね。** | | **小川** | はい。先日の案件でも SQL Server 2005 が製品として発売されていれば、迷わずこの機能を利用していました。利用できればどんなに運用管理が楽だったかと思います。パーティション テーブルは、エンタープライズ案件にはなければ困りますね。SAP R/3 環境などでも欠かせない機能といえるでしょう。 また、パーティション機能では、インデックスもパーティション化できるのですが、これはパーティションごとにインデックスの再構築が行えるということを意味しています。もし、インデックスがテーブル単位だとすると、インデックスの再構築に時間がかかりますし、オフラインで再構築している場合は、その間ユーザー アクセスは一切できなくなるわけです。たとえば、1,000 万件のデータのインデックスを再構築するのに 1 時間かかっていたとすると、パーティションを 10 個に分割していて、インデックスもパーティション化されていれば、単純計算で 1 つのパーティションを再構築するには 6 分で済むことになります。 このように、テーブル パーティションは、データの切り離しや追加が簡単に行えるだけでなく、パーティションごとにインデックスの再構築が行えるのも運用管理面およびパフォーマンス面での大きなメリットです。 |  |  |  | | --- | --- | | **――** | **64-bit 機能についてはいかがですか。** | | **小川** | 64-bit は大規模なデータベース環境では欠かせないでしょう。32-bit のシステムでは、4 GB というメモリ空間の壁を超えるために AWE (Address Windowing Extensions) という機能を利用しているのですが、AWE ではオーバーヘッドがあります。それに、SQL Server の自動チューニング機能が完全に活かされている形ではありません。というのも AWE では、バッファ キャッシュに関しては、メモリ領域を有効利用できるのですが、Sort や Hash 処理で利用される一時領域用の Workspace メモリや、実行プランのキャッシュを格納しておくためのプロシージャ キャッシュ、ロック用のメモリ領域などは、AWE の機能を享受できず、4 GB の壁を超えられないのです。 しかし 64-bit 環境であれば、ネイティブにメモリ空間を利用できるようになります。これは SQL Server の自動チューニング機能が完全に動作し、Workspace メモリやロック用のメモリ領域も大きく取れるようになるということです。巨大なデータベースを扱っている人にとって、待望の機能といえるでしょう。 | | * | ***SQL Server 2005 は、金融や証券のシステムなど 大規模なエンタープライズ環境でも十分に通用します** |   *   |  |  | | --- | --- | | **――** | **SQL Server 2005 はどういったユーザーにお勧めでしょうか。** | | **小川** | 間違いなくエンタープライズ系の企業ですね。金融や証券のシステムなど、今まで SQL Server 2000 ではなかなか踏み込むことのできなかった世界でも SQL Server 2005 は十分に通用します。そういったシステムを手がけるエンジニアの方々にぜひとも評価していただきたいと思います。直ちに評価する価値のある製品だと思っています。 |  |  |  | | --- | --- | | **――** | **最後にこの記事を読んでくださっている皆様にひと言お願いします。** | | **小川** | 新しい製品だと、どうしても少し様子をみてから使おうという方が多いかと思います。しかし、今回の SQL Server 2005 はものすごくいい。マイクロソフトがパフォーマンスを最大限に上げるように努力をした結果だと思いますので、ぜひ評価環境を作って試してみてほしいです。パーティション テーブルやスナップショット分離レベル、豊富な GUI ツールによって、今までに運用上で困っていたことが大分緩和されるでしょう。 また、SQL Server 2005 には、PASSJ からの要望が反映された機能もあります。Express Edition の「AttachDbFileName」というデータベース ファイル (.mdf) をそのままデータベースとして利用できる機能なのですが、これはマイクロソフトの SQL Server 開発チームと PASSJ がミーティングをしたときに河端さんが発したひと言から始まり、それが実装されました。現状の SQL Server に物足りなさを感じていたり、不満を抱えている方は、ぜひとも PASSJ に参加していただき、その声を聞かせてください。恥ずかしがらずに何でもいいので声を挙げていただき、私たちに相談してほしいのです。そうすれば、私たちで解決できることもあるでしょうし、マイクロソフトの製品開発担当者へフィードバックを挙げるというパスもあります。 この一例のように、あなたのひと言が製品を変えられる可能性があります。コミュニティにはそれだけの力があるのです。また、皆様が声を発信することで、PASSJ はより高い技術集団になれると思っていますし、技術習得だけではなく、いろいろな問題や情報を共有していくことで、全員が Win-Win の関係になれると信じています。 | |  |

|  |
| --- |
| READY クローズアップ ***Microsoft MVP が語る SQL Server 2005**  *Microsoft MVP へのインタビューから SQL Server 2005 を探る READY クローズアップ。この連載では、SQL Server 2005 に対する期待や導入メリット、具体的な利用方法などについて、Microsoft MVP として表彰された方々へインタビューを行い、SQL Server 2005 に対する想いを語っていただきます。 * *→*[Microsoft MVP とは](#c8_7) |

**MVP インタビュー 第 1 回**

**Microsoft SQL Server 2005 は、ココが違う**

**可用性とセキュリティがさらに向上している SQL Server 2005**

|  |  |
| --- | --- |
|  | * |
| |  | | --- | | * |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Index**   |  |  | | --- | --- | | *[SQL Server 2005 の最大の魅力は「データベース ミラーリング」と「セキュリティ機能の強化」](#c1_1) |  | | *[出会いはフロッピー ディスク 4 枚の時代から](#c1_2) |  | | *[データベース ミラーリングはすべてのユーザーにお勧めしたい機能です](#c1_3) |  | | *[セキュリティ強化にも注目　アカウントのマッピング機能、データの暗号化 etc.](#c1_4) |  | | *[HTTP エンドポイントとサービス ブローカにも注目](#c1_5) |  | | *[ぜひフィードバックを!](#c1_6) |  | | **河端 善博 氏** Microsoft MVP for SQL Server * 有限会社コザック代表取締役社長。 1984 年 ダイナウェアにて建築用 3 次元パース作成システム、建築用 2 次元 CAD システムの開発に従事。 1994 年 クボタシステム開発へ転職。Windows システムの開発、構築に従事。また、Windows 関連の雑誌記事の執筆、ASP、ADO、ADSI 関連の翻訳監修を行う。 2000 年 SQL Server ユーザーのためのコミュニティである PASSJ を立ち上げ、理事を務める。 2004 年 12 月より、Windows ホスティング サービス「cervi.jp」を提供。 河端善博氏 | | |  |  | | --- | | * | | |

*

|  |  |
| --- | --- |
| * | ***SQL Server 2005 の最大の魅力は 「データベース ミラーリング」と「セキュリティ機能の強化」** |

*  
いよいよ SQL Server 2005 の出荷が近づいてきました。この連載では、Microsoft SQL Server MVP でもある READY 編集長の松本美穂さん・松本崇博さんから Microsoft MVP として表彰された方々にインタビューし、SQL Server 2005 に関する導入メリットや具体的な利用方法、さらには期待などについて語っていただきます。  
第 1 回となる今回は、日本における Microsoft MVP の第 1 号である河端善博さんをお迎えしました。河端さんは、2000 年に SQL Server ユーザーのためのコミュニティ「PASSJ」を立ち上げ、理事／セキュリティ ボード リーダーとして活動されると共に、現在は .NET 開発者のためのコミュニティやサイト運営者を支援する組織「INETA」のリードも務め、マイクロソフトの日本におけるコミュニティ活動の牽引役を果たしています。

|  |  |
| --- | --- |
| * | ***出会いはフロッピー ディスク 4 枚の時代から** |

*

|  |  |
| --- | --- |
| **――** | **まずは SQL Server との出会いから教えていただけますか。** |
| **河端** | 私が SQL Server と出会ったのは、日本における最初のバージョンである SQL Server 4.21a のベータ版が登場したころになります。11 年も前のことで、SQL Server がフロッピー ディスクわずか 4 枚で提供されていたことを覚えています。当時は、OS にしても、SQL Server にしても情報がほとんどなく、使いこなすのには本当に苦労しました。 |

|  |  |
| --- | --- |
| **――** | **当時と比べて現在の SQL Server はかなり進化していますが、どのような点を気に入っておられますか。** |
| **河端** | 本当に進化しましたね。関連情報も増えました。特に SQL Server 7.0 からはユーザー インターフェイスががらりと変わり、便利なウィザードもたくさん追加されて、これからデータベースを運用しなければならない人にとっては使いやすいインターフェイスになりました。当時は、Oracle が GUI に力を入れていなかった時代ですから、SQL Server の GUI は本当にありがたかったですね。 また、メモリ管理の自動化や、DTS (SQL Server 2005 から Integration Services に名称変更)、OLAP Services (SQL Server 2000 から Analysis Services に名称変更) などが提供されたのもこのバージョンからです。これらの機能を備えた製品を購入しようとすると、それだけで数十万～数百万円もすることになります。しかし、SQL Server には標準でついてきます。「これらをトータルに使いこなしてお客様に提案すれば、かなり強力な武器になる」と感じました。SQL Server 2000 ではさらに Notification Services や Reporting Services が加わり、ますます魅力的な製品になりました。SQL Server 2005 では、HTTP エンドポイントによる Web サービス機能やサービス ブローカによる非同期処理なども加わり、もはや単なる RDBMS 製品ではなくなっていますね。 |

|  |  |
| --- | --- |
| **――** | **現在、河端さんは Windows ベースのホスティング サービスを営まれていますよね。** |
| **河端** | はい。昨年の 12 月に「cervi.jp」をスタートしました。そこでは、SQL Server のホスティング サービスを提供しています。最初にホスティング事業を始めようと海外を調査したときに、非常に安く提供されていることがわかったので、自分でも海外と同じように安く提供したいと考えました。そうした背景から、現在は SQL Server に加えて ASP (Active Server Pages)、ASP.NET、FTP、SSL、各種メール機能などを利用できるサービスを、月額 1,000 円から用意しています。 先ほどの話のように、SQL Server は、非常に使いやすい製品なのですが、データベースをまったく知らないシステム管理者にとっては、導入から日々の運用、監視、障害対応までをしなければならないとなると大変です。そこで、ホスティングという形で SQL Server をこちらですべて管理し、お客様には管理業務の煩わしさで悩まされることなく、SQL Server のパワーを活用していただければと思っています。 |

|  |  |
| --- | --- |
| * | ***データベース ミラーリングは すべてのユーザーにお勧めしたい機能です** |

*

|  |  |
| --- | --- |
| * | Photo * |

|  |  |
| --- | --- |
| **――** | **SQL Server 2005 の中で一番うれしい機能は何ですか。** |
| **河端** | デベロッパとしては「SQLCLR」ですが、ホスティング事業をするうえで一番うれしいのは、「データベース ミラーリング」ですね。データベースを二重化できれば、ホスティング サービスの価値を数倍に引き上げることができます。また、データベース ミラーリングは、設定や運用が容易である点もたいへん気に入っています。たとえば、これまでにも可用性を向上させるテクノロジには、MSCS (Microsoft Cluster Service) やログ配布、レプリケーションなど、いろいろとありましたが、いずれも設定が大変で、運用が難しかったのです。特に、障害時の復旧作業は、マニュアルどおりに操作しようとしても、操作手順があまりにも複雑でした。また、ちょっとした操作をミスするだけで、うまく復旧できなくなったりします。これでは、一部のユーザーにしか使いこなすことはできません。 |

|  |  |
| --- | --- |
| **――** | **データベース ミラーリングは、コスト面でのメリットも大きいのではないでしょうか。** |
| **河端** | はい。MSCS と比べて 10 分の 1 程度のコストに抑えられると思っています。たとえば、MSCS では複数のサーバーでストレージを共有しますので、まずここを強化しなければなりません。これには高価なストレージが必要になり、多大なハードウェア コストが必要となります。また、障害復旧までを考えた場合の運用管理コストで見ると、ここでも大きな差が出てきます。 データベース ミラーリングであれば、安価なサーバーを 2 台に、ライセンスは Standard Edition を 1 つ、ウィットネス (ミラーリングを監視するためのマシン) には無償の Express Edition を利用すれば、非常に低価格でデータベースの二重化を実現できます。なによりも、設定と運用、復旧作業が簡単で、運用管理コストを大幅に抑えられるのが大きな魅力です。データベース ミラーリングは、SQL Server のすべてのユーザーにお勧めしたい機能ですね。 |

|  |  |
| --- | --- |
| **――** | **データベース ミラーリングのその他の魅力はいかがでしょうか。** |
| **河端** | これは Enterprise Edition でのみ提供され、ライセンスも追加で必要になるのですが、データベース ミラーリングでは、待機系のサーバーに対して「データベース スナップショット」を作成することができます。これにより、待機系のサーバーへ二重化したデータを参照できるようになり、開発時やテスト環境を作成したいときに簡単にデータを切り離せるようになります。待機系のサーバーを別の目的で利用したいというニーズは高いので、このような使い方ができるとたいへん便利です。 |

|  |  |
| --- | --- |
| * | ***セキュリティ強化にも注目 アカウントのマッピング機能、データの暗号化 etc.** |

*

|  |  |
| --- | --- |
| **――** | **SQL Server 2005 で注目の機能をもう 1 つ挙げるとしたら何になりますか。** |
| **河端** | セキュリティが強化された点ですね。ホスティング事業者の目から見て、今までの SQL Server では、複数の企業で 1 つのサーバーを共有して利用しようとする場合に、お互いに迷惑をかけないようにするため、セキュリティの設定が、SQL Server の機能だけでは難しい部分がありました。 しかし、SQL Server 2005 では違います。たとえば、SQL Server 2005 では作成した SQL Server のログイン アカウントに対して Windows のユーザー アカウントをマッピングできる機能が追加されています。これにより、ファイル システムに対してどういった Windows ユーザーの権限でアクセスできるのかを指定したり、どのユーザー権限でプログラムを実行するのかなどを制御できるようになります。そうなれば、たとえ SQL Server の管理者権限を与えたとしても、ユーザーは指定されたフォルダに対してしか書き込みができないなどの設定ができるようになります。また、これに加えて Windows Server 2003 R2 で提供予定のフォルダ単位のクォータ機能と組み合わせれば、フォルダに対して指定されたサイズ以上の書き込みができないようにするといった設定も可能になります。 |

|  |  |
| --- | --- |
| **――** | **データの暗号化機能も搭載されましたが、いかがですか。** |
| **河端** | そうですね。今までは ASP.NET など上位層のアプリケーションで暗号化しなければならなかったのが、SQL Server のレベルで暗号化できるようになったのはうれしいですね。ホスティングにおいては、昨今の個人情報保護法の問題もありますし、お客様にデータを安心して預けていただけるようになります。企業において特定の人にしか見せたくないデータがある場合に、暗号化機能をスマート カードなどと組み合わせて利用し、ユーザーの所へ持っていった段階で解読できるシナリオにすれば、さらにセキュリティが高められます。 |

|  |  |
| --- | --- |
| **――** | **セキュリティは細かい部分でも強化されていますが、その点はいかがですか。** |
| **河端** | イベント ビューアへのログが詳細に出力されるようになった点や、トレース機能でのトレースできるイベントが増えた点は、詳細なログ (操作履歴) を残すという目的を達成できるので、不正アクセスの検知に非常に役立ちます。もし、ログを残せない場合は、不正アクセスの被害にあったということにさえ気が付かないケースも出てきてしまいます。 また、「DDL トリガ」機能も管理系の操作に対して厳しく制限を設けられるようになるので便利です。たとえば、悪意のあるユーザーが不正にアカウントを作成しようとしても、DDL トリガでアカウントを作成できないように制限を設けておけば、防げるわけです。 このようなセキュリティ機能の充実は、SQL Server 2005 を導入するうえでの大きなメリットになりますね。 |

|  |  |
| --- | --- |
| * | ***HTTP エンドポイントとサービス ブローカにも注目** |

*

|  |  |
| --- | --- |
| **――** | **SQL Server 2005 でその他に注目している機能はありますか。** |
| **河端** | 「HTTP エンドポイント」による Web サービス機能と「サービス ブローカ」を組み合わせた非同期処理は、ニーズが高いでしょう。私もすぐにでも利用したいと思っています。ユーザー登録が行える Web サイトを不特定多数のユーザーに公開している場合を考えてみてください。このようなサイトでは、ピーク時には通常の 100 倍以上ものアクセスがあったり、予想をはるかに超えるアクセスが集中する時間帯があったりします。これを同期型のアプリケーションで処理したとすると、処理が追いつかなかったり、デッドロックが多発して、一部のユーザーしか登録ができずに、ほとんどのユーザーへエラーが返るといったことが起こりえます。 しかし、ユーザー登録を行う処理を Web サービス化してサービス ブローカと利用すれば、非同期で順々に処理できるようになります。こうすれば、瞬間的なサーバーの処理能力を向上させることができ、デッドロックの発生も抑えられ、ピーク時にも耐えられるようになります。 |

|  |  |
| --- | --- |
| * | ***ぜひフィードバックを!** |

|  |  |
| --- | --- |
| Photo * | * |

|  |  |
| --- | --- |
| **――** | **私たちが MVP になってからはマイクロソフトの製品開発者の方に対するイメージが大きく変わったのですが、開発担当の方々はフィードバックをたくさん欲しがっていて、ユーザーの意見をどんどん取り入れていこうという姿勢を強く感じています。** |
| **河端** | そうですね。昔はフィードバックをする手段がほとんどありませんでしたが、日本では PASSJ を作り始めたころから、フィードバックを挙げやすい環境になりました。私も PASSJ や MVP プログラムを通じて、実際に製品開発を担当されている方々と話をすると、日本だとクレームと思われるような内容でも「どう改善したらよいのか、もっと詳しく聞かせて欲しい」とよく言われます。実際に作っている人たちは、反応を知りたがっていて、ユーザーが喜ぶ顔が見たいという気持ちが伝わってきます。ユーザーの声に真摯に耳を傾けてくれる人たちだということがわかったのです。 |

|  |  |
| --- | --- |
| **――** | **これからもフィードバックを挙げ続けていくわけですね。** |
| **河端** | はい。特にホスティングに関するフィードバックをどんどん挙げ、ホスティング環境でも Windows プラットフォームのパワーが十分に発揮できるようになってほしいと思っています。そうなれば、Windows や SQL Server の価値がもっともっと上がると思います。ビジネスや業務の内容が日々進化する以上、完璧な製品というのはありえません。今後もどんどんフィードバックを挙げて、私たちのビジネスの成長と共に SQL Server が進化し、常に私たちを支え続けてくれることを願っています。 |

|  |  |
| --- | --- |
| **――** | **最後にこの記事を読んでいただいている方々にひと言お願いします。** |
| **河端** | マイクロソフトの製品開発担当の方々は、皆さんの声を聞きたがっていますので、どんどんフィードバックを挙げていただくと良いと思います。PASSJ に要望を投げていただいてもいいですし、[MSDN Product Feedback Center](http://lab.msdn.microsoft.com/productfeedback/) では、日本語でフィードバックを投稿することもできます。私たち MVP やコミュニティ メンバは皆様の要望が通るように常に援護しています。もし、今回のバージョンに間に合わなくても、次のバージョンがあります。それがより良い製品作りにつながり、製品開発者にとっても、私たち利用する側にとっても互いにハッピーになれるものと信じています。 |