

Oracle+.NET

3つの誓い



EMCジャパン株式会社
ソリューション・コンサルティング部
一志 達也 ICHISHI, Tatsuya
<http://www.emc2.co.jp/>

特別編

最終回

システムのパフォーマンスを考える

Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:

Level



Samples

最後の難関

特別編と称して続けてきた連載も、いよいよ最終回を迎えることになりました。今回は、Oracle+.NETシステムにおける、パフォーマンスについて考えてみます。

パフォーマンスである理由は言うまでもないと思いますが、処理が高速に終わるほうがユーザーに喜ばれ、システムが消費するリソース（CPUやメモリ）も少なく済むからです。逆にパフォーマンスが悪ければ、ユーザーはイライラしてしまい、クレームを受けるハメになります。もしこれが、顧客向けのシステム（ECサイトなど）であれば、ビジネスチャンスを逸してしまうかもしれません。

そうならないために、開発者はプログラムコードをチューニングし、少しでもパフォーマンスが向上するように努力しています。です

が、プログラムコードをチューニングするだけがすべてではない、ということも理解されていることでしょう。

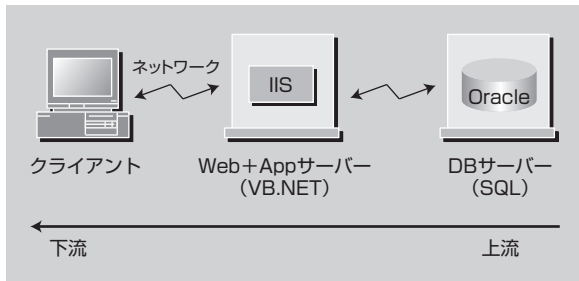
測定と切り分け

この連載で紹介してきたように、VB.NETで作成されたアプリケーションとOracleデータベースが連携し、しかもWebサービスを用いるようなシステムの場合、パフォーマンスを変化させる要因はさまざまです。図1にも示したように、少なくとも3か所にプログラムを実行するコンポーネントがあり、さまざまなソフトウェアやインフラが関連します。

それだけに、どうにも複雑に思えてしまうでしょうが、パフォーマンスを測定するには個々のコンポーネントに切り出すしかありません。そうすれば、どこに問題が起

Oracle+.NET 3つの誓い

図1：システムを構成するコンポーネント



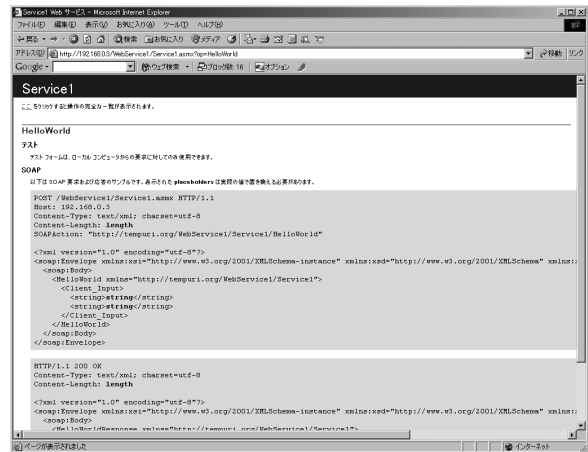
きているのかを判断しやすくなりますし、考えられる対処も限られてくるからです。パフォーマンスに限ったことではなく、何かしらの問題を解決する場合には、原因となる箇所の切り分けと絞り込みが重要になります。

切り分けを行なう場合には、システムの上流から下流へと切り分けを進めるのが基本です。図1の場合であれば、DBサーバーから始めて、Appサーバー、クライアントというように進めていきます。

DBサーバーであれば、DBに発行しているSQLを調べ、それをSQL*Plusなどで実際に発行してみます。その処理時間をみれば、時間がかかっているかどうかわかるはずです。

Appサーバーは、以前に紹介したように、Webサービスのテストページを呼び出してみるといいでしょう(図2)。パラメータをひとつひとつ入力しなくてはならず、結構面倒ですが他にいい方法がありません。サーバーで直接テストページを呼び出して実行し、戻ってくるまで

図2：Webサービスのテスト画面



の時間からDBサーバーで測定した時間を引けば、Appサーバー単体での処理時間を推定できます。

最後にクライアントで処理を始めてから終わるまでの時間から、Appサーバーで測定した時間を引けば、クライアント単体での処理時間+ネットワークの通信にかかる時間を推定できます。ネットワークの通信にかかる時間を除きたければ、サーバーそのものでクライアント/アプリケーションを実行するか、ネットワークのロスを見無視できるほど近いところに設置すればいいでしょう(図3)。

こうしてひとつひとつ調べてみて、おおよそのボトルネックが目星がついたら、そこで行なわれている処理(プログラム)を分割して調べていきます。たとえば、ひ

図3：パフォーマンスの測定ポイント

