

# Oracle+.NET

# 3つの誓い



株式会社サンブリッジ テクノロジーズ  
プロジェクトマネージャ  
一志 達也 ICHISHI, Tatsuya  
<http://www.sunbridg-tech.com/>

## 第3回 オブジェクトモデルを理解しよう

### Technology Tools

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

### Level



### Samples

・この記事で取り上げたソースコードおよびサンプルプログラムは、付録CD-ROMの¥DOTNET¥ORACLEディレクトリに収録しています。

¥WINDOWSAPPLICATION1  
接続プーリング確認用プログラム

### プログラム作成に必要な知識

今回は、Oracle Data Provider for .NET (以下ODP.NET) を使った開発環境の準備をし、Oracle9i DBに接続する簡単なプログラムを作成するまでについて紹介しました。今回は、接続後に行なうデータ処理を知るために必要な「ODP.NETのオブジェクトモデル」について紹介しようと思います。そのうえで、前回のプログラムの意味を理解し、さらに応用的なプログラムの作成方法も理解してもらえればと思います。

### オブジェクトモデルとは?

オラクル (Oracleデータベース)に限らず、データベースと連携したプログラムを作成するためには、そのインターフェイスに合わせたオブ

ジェクトモデルの理解が不可欠です。なぜなら、古くはDAOにはじまり、RDO、ADO、ADO.NETそしてOO4O (Oracle Objects for OLE) も、すべてデータベースへのアクセスを抽象化するモデルを持っているからです (図1~5)。それぞれモデルは、構成も使い方も異なりますが、プログラムが煩雑になるのを防ぎ、より簡単にデータを操作するために役立ちます。

これらのモデルに含まれる各オブジェクトは、それぞれデータベースとの接続やデータベース内の表、それに含まれる列などを表わしています。プログラムを作成する際には、それらのオブジェクトを変数に割り当てるなどしたうえで、プロパティ (属性) やメソッド (命令) を使ってデータ操作を進めてゆくのです。そのため、ADO.NETなどを使ったデータアクセスプログラムについて理解を深めるには、オブジェクトの役割や関連性だけでなく、

図1：DAOのオブジェクトモデル

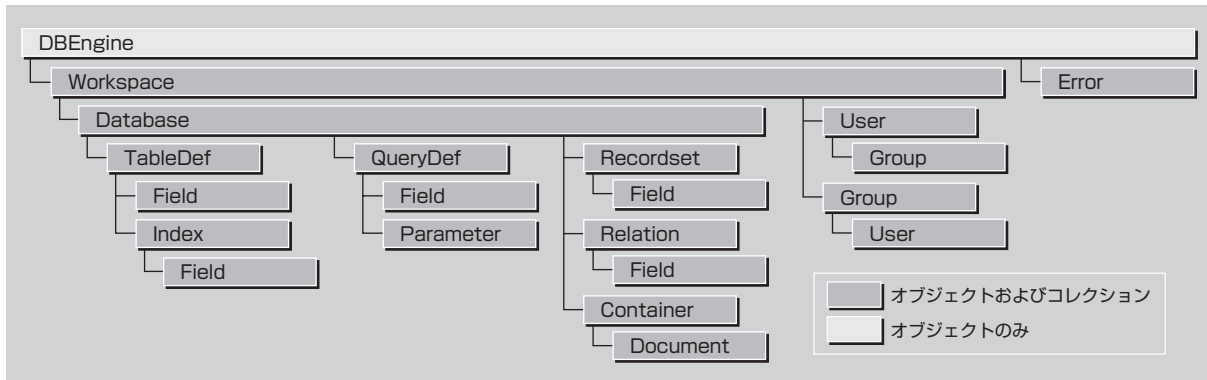


図2：RDOのオブジェクトモデル

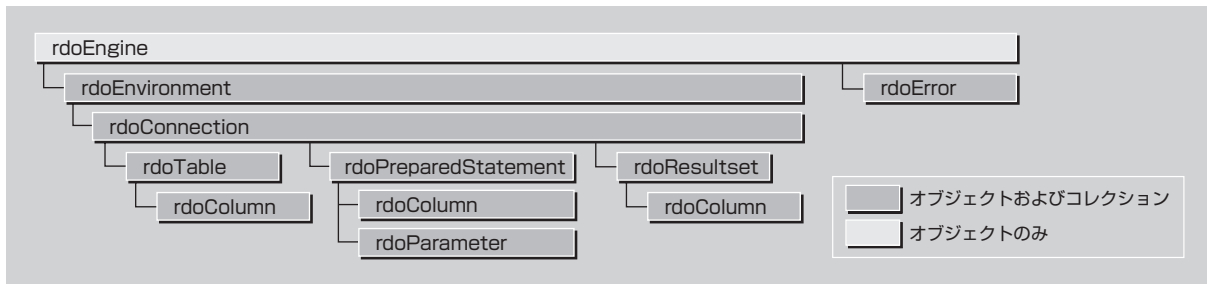
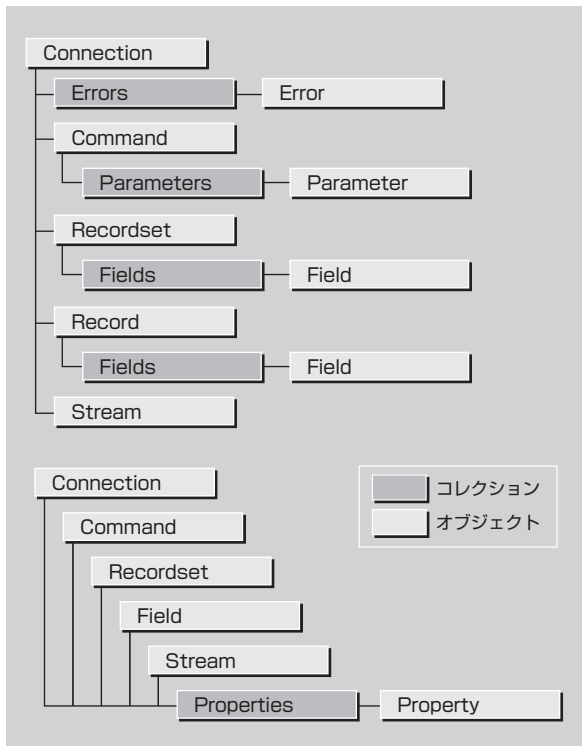


図3：ADOのオブジェクトモデル



各オブジェクトのプロパティやメソッドについても理解してはなりません。

それはさておき、今回セレクトしたODP.NETは、ADO.NETに準拠したモデルを持っています(図6)。そのため、ADO.NETの使い方を理解している方には、比較的わかりやすいのではないかと思います。そうでない方にとっても、ODP.NETを理解することが特殊なものではなく、“ADO.NETへの応用が利く”というのはメリットになるでしょう。

とにもかくにも、ODP.NETによるデータベース操作を理解するためには、図6のオブジェクトモデルについてまずは理解する必要があります。図6に含まれているオブジェクトの役割については、表1に簡単にまとめておきます。



## オラクルへの接続



これから、これらのオブジェクトを使ったデータ操作