

Oracle データベース

ポリテクセンター関西
(関西職業能力開発促進センター)

秋田 正秀*

1. はじめに

2年半前に電算機リースが関西職業能力開発促進センターに導入されました。そのリースの中で Oracle データベースが導入されました。導入された Oracle データベースは Oracle7.3.4 のバージョンです。私自身はそれまでに Oracle データベースの存在自体を知らず、扱った経験もありませんでした。その頃は Oracle データベース関連の市販されていた本も今ほど多くなく Oracle データベースのマニュアルを中心にセミナー開発を進めていました。そして、昨年度から Oracle データベースの能力開発セミナーとして RDB 利用技術 (Oracle 編) を行っています。来年度は計画として PL/SQL プログラミングと Oracle8i 導入技法を新たに考えています。もともと関西職業能力開発促進センターに赴任する前は短大機械系の講師をしていたので情報関係を専門に行うこと自体がここに来て初めてでした。

本内容では Oracle データベースのセミナー実践報告を行います。はじめに Oracle データベースの概要と構築を簡単に紹介した後、実際に能力開発セミナーで行っている RDB 利用技術 (Oracle 編) を紹介します。加えて、来年度計画の PL/SQL プログラミングの概要についても紹介します。

2. Oracle データベース

ここでは Oracle データベースの概要について紹

介します。主に Oracle データベースの製品、Oracle データベースの構成について基礎的な内容を紹介します。

2.1 データモデルと製品分類

データベースはデータベースが採用しているデータモデルから分類されています。よく紹介されるものとしては階層型とネットワーク型、リレーショナル型があります。Oracle データベースはデータモデルにリレーショナル型を用いたものです。データモデルにリレーショナル型を用いたデータベースはリレーショナルデータベース (Relational DataBase: RDB) と呼ばれています。ですから、Oracle データベースはこのリレーショナルデータベースに分類されます。最近では市販されているデータベースのほとんどがリレーショナルデータベースになっています。例えば、IBM 社の DB2 や Informix 社の Red Brick, Sybase 社の Adaptive Server, Microsoft 社の SQL Server などがあります。また、分類もさらに進み Oracle データベースはオブジェクト指向の内容を取り入れているのでオブジェクトリレーショナルデータベース (Object Relational DataBase: ORDB) とも呼ばれています。

Oracle データベースは 4 種類の商品に分かれています (図 1 参照)。現時点の Oracle8i では Enterprise Edition と Workgroup Server と Personal Edition と Lite に分類されています。簡易的にこれらは規模で分類されており、Enterprise Edition が規模的に最も大きくテラバイト 1000 ユーザを扱え、Workgroup

*現 京都職業能力開発促進センター

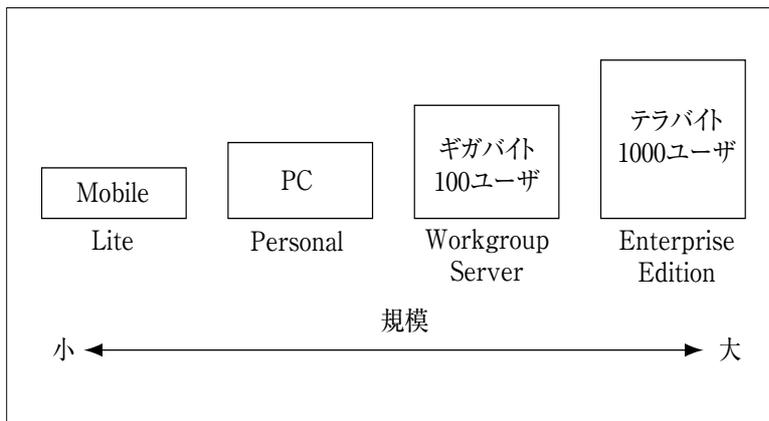


図1 Oracleデータベース製品

Serverはギガバイト100ユーザ、Personal Editionはパソコン版、Liteはモバイル版となっています。¹⁾

2.2 Oracleデータベースの構成

Oracleデータベースはデータベースとインスタンスから構成されています(図2参照)。データベースには論理構造と物理構造が存在します。物理構造は3種類の物理ファイルで構成されています。その3種類とはデータファイルとREDOログファイルと制御ファイルです。通常、それぞれのファイルは1つだけではなく複数存在しています。論理構造には表領域とデータベースのスキーマオブジェクトがあります。表領域は論理的な記憶領域でこの中に表などを保存します。スキーマオブジェクトには表やビュー、順序、ストアドプロシージャ、シノニム、索引などがあります。ちなみにスキーマとはオブジェクトの集合を意味しています。一方、インスタンスはシステムグローバル領域(System Global Area: SGA, 以降SGAと呼ぶ)と呼ばれるメモリ領域とバックグラウンドプロセスから構成されます。SGAはバッファとして使われ、共有プール、データベースバッファキャッシュ、REDOログバッファで構成されています。また、共有プールはライブラリキャッシュとデータディクショナリキャッシュから構成されています。バックグラウンドプロセスはシステムモニタ、データベースライター、プロセスモニタ、チェックポイント、ログライターなどから構成されています。



図2 Oracleデータベースの構成

2.3 データベースの論理構造

Oracleデータベースの論理構造として先ほど表領域を紹介しました。データベースはこの表領域単位でデータを分割します。表領域はOSで管理される物理的なデータファイルと対応します。表領域は1つ以上のデータファイルを持つことができます。これは大きいレベルでの記憶の単位でOracleデータベースでは細分化されておりデータブロック、エクステント、セグメントがあります。データブロックは記憶の最も小さい単位です。1つのデータブロックはOSの特定のバイト数に対応しています。このサイズはデータベースを作成するとき初期化パラメータファイルで指定します。このファイルでデータブロックを指定してデータベースインスタンスを作成した後は変更することはできません。初期化パラメータファイルはテキスト形式なので内容を変更することはできますが、データブロックのサイズを変更した初期化ファイルを使ってデータベースインスタンスを起動するとエラーになります。エクステントは連続したデータブロックの集合で記憶領域確保の単位になっています。表領域を作成するときのパラメータはエクステント単位で指定されます。セグメントは特定の論理構造に割り当てられるエクステントの集合です。セグメントにはデータセグメント、索引セグメント、ロールバックセグメント、一時セグメントがあります。

2.4 スキーマオブジェクト

スキーマオブジェクトはデータベースを参照するための論理構造です。スキーマオブジェクトには表、ビュー、順序、ストアードプロシージャ、シノニム、索引、データベースリンクなどがあります。表はデータを格納するオブジェクトです。Oracleデータベースのデータは表に格納されます。ビューは仮想表と言われ1つまたは複数の表のデータを加工した表現です。ビューはデータを持っておらず、あくまでもデータは表から取り出されます。順序はデータベースの連続の番号を作成する機能を持っているオブジェクトです。表の中の列で一意の連続した番号を入力したいときにこの順序を使います。ストアードプロシージャはサーバに格納される処理のブロックです。ネットワークの負荷を軽減できたりバッチを作成したりできます。同じようなものでストアードファンクションがあります。シノニムは別名のオブジェクトです。表やビューなどに対して作成することができます。例えば、他のユーザの表を使用する場合、表の指定はユーザ名：表名となり複雑です。これに対してシノニムを作成し別名をつけることで指定する名前が簡単になります。索引はその名のとおりで本などの索引と同じ目的で使われます。索引を作成するとデータ検索のパフォーマンスが向上します。データベースリンクはデータベースから別のデータベースへのパスを記述するオブジェクトです。データベースが分散しているときに使います。複数のデータベースのデータを使った処理をする場合、このデータベースリンクを作成しておけば、記述上、データベースがどこにあるかを考えなくてよいこととなります。

2.5 データベースの物理構造

データベースはデータファイル、REDOログファイル、制御ファイルから構成されます。データファイルにはデータベースのデータが格納されます。論理構造で表領域を紹介しましたが、この表領域がデータファイルに対応しています。ですから、表領域に格納されたデータはすべてデータファイルに格納されます。REDOログファイルはいわゆるログで変更

履歴などが記録されるファイルです。障害が発生した場合にこのファイルが使われて修復されます。REDOログファイルはグループを作成し、複数のファイルでログをとることに なっています。Oracleデータベースでは少なくとも2つ以上のREDOログファイルを持っています。制御ファイルはOracleデータベースの情報を記録しています。主にデータベース名、データファイルの位置と名前、REDOログファイルの位置と名前、データベースの作成日などを記録しています。これはOracleデータベース自身が管理するファイルでデータベース管理者は編集できません。このファイルが使えなくなるとOracleデータベース自体が使えなくなるので通常は多重化して複数のファイルが存在します。

2.6 Oracle インスタンス

Oracle インスタンスはSGAとバックグラウンドプロセスから構成されます。SGAは1つのOracleインスタンスのためのデータと制御情報を含む共有メモリ領域です。SGAはインスタンス起動時に割り当てられ、各データベースインスタンスは独自のSGAを持ちます。SGAにはデータベースバッファキャッシュとREDOログバッファ、共有プールがあります(図3参照)。データベースバッファキャッシュはデータファイルから読み込まれたデータブロックを格納している記憶領域です。データファイルからデータを毎回読み出しているとディスクI/Oが頻繁に発生し処理自体が遅くなるので使ったデータをここに格納しておきます。データベースバッファキャッシュにデータがあればディスクI/Oは発生しないので処理時間が短くなります。この領域のサイズは初期化パラメータのDB_BLOCK_SIZEと



図3 SGA

DB_BLOCK_BUFFERで指定します。領域が一杯になった場合は最低頻度 (LRU) アルゴリズムで処理が行われます。REDO ログバッファはデータベースに対して行われた変更内容を記録する記憶領域です。特徴としてはデータベースに加えられた変更を逐次記録し、記憶領域が一杯になれば循環して使われる循環バッファになっています。共有プールはライブラリキャッシュとデータディクショナリキャッシュから構成されています。ライブラリキャッシュは実行計画、解析済みコード、PL/SQL プロシージャなどの実行済みの内容が格納されています。このおかげで実行済みの内容は次回に同じ内容が実行される時に使用され処理が速くなります。ディクショナリキャッシュは使用されたデータディクショナリの情報が行単位で格納されます。これも処理の効率化を目指しており以前に使われたデータディクショナリはこのキャッシュから情報を読み出して使われることになります。

一方、バックグラウンドプロセスにはデータベースライター (DBWR)、ログライター (LGWR)、チェックポイント (CKPT)、システムモニタ (SMON)、プロセスモニタ (PMON)、アーカイバ (ARCH) などがあります。データベースライターは変更された内容が入っているデータベースバッファキャッシュからデータファイルに書き込みます。ログライターは REDO バッファに記録されているデータベースの変更履歴を REDO ログファイルに書き込みます。トランザクションがコミットしログバッファが一杯になると書き込みが発生します。チェックポイントはデータベースバッファキャッシュがデータベースライターによって書き込まれるためのイベントを発生します。チェックポイントがイベントを発生するとデータベースのすべてのデータファイルと制御ファイルが更新されます。システムモニタはインスタンスの起動時にインスタンスの回復を実行します。また、データベース内の空きエクステンツを結合して連続した空き領域を作成します。プロセスモニタはユーザプロセスが障害を起こすとプロセスの回復を実行します。また、データベースバッファキャッシュのクリーンアップとプロセスが使用していたリソースを解放します。また、ディスパッチャプロセスとサ

ーバプロセスをチェックして障害が発生している場合にはそれらを再起動します。アーカイバは Oracle データベースをアーカイブログ運転しているときにあるバックグラウンドプロセスです。オンライン REDO ログファイルが満杯になったときにそれらのログファイルをアーカイブファイルに書き出します。

2.7 その他

Oracle データベースの構成は紹介しましたが、まだ、紹介していない重要なメモリやファイルを紹介します。メモリとして紹介していないものにプログラムグローバルエリア (Program Global Area : PGA) があります。PGA はユーザプロセスがデータベースと接続し、セッションが確立されると割り当てられます。PGA は 1 つのサーバプロセスまたは 1 つのバックグラウンドプロセスごとに割り当てられる非共有のメモリ領域です。専用サーバ構成の場合、PGA にはスタック領域とセッション情報、カーソルの状態が格納されます。

物理ファイルで紹介していないものに初期化パラメータファイルとパスワードファイル、アーカイブログファイルがあります。初期化パラメータファイルはインスタンス構築時およびインスタンス実行時に必要になります。このファイルはテキスト形式で Oracle データベース運用時に設定するパラメータの値が入っています。その初期化パラメータは多くあり最低限必要なパラメータを覚えておくといえると思います。なかにはデータベースの名前やインスタンスの名前、ブロックのサイズなどがあります。パスワードファイルはリモート管理するときに必要になります。リモートで OS 認証しない場合にこのパスワードファイルで SYSDBA の権限が認証されます。SYSDBA はデータベースを構築するときに必須の権限です。アーカイブログファイルは Oracle データベースをアーカイブログモードで運転しているときに使われるファイルです。REDO ログファイルは一杯になると REDO ログファイルの切り替えが起こり、すべての REDO ログファイルが一杯になると最初の REDO ログファイルに戻り上書きします。この上書きする前に REDO ログファイルの

内容をアーカイブログファイルに書き出します。これでREDO ログの内容は残されることになり障害発生してもアーカイブログファイルから回復することができるようになります。

3. Oracle データベース構築

ここでは簡単に Oracle データベースの構築を紹介します。私自身は WindowsNT 上でしか構築したことがありませんので Oracle データベース構築する対象は WindowsNT です。そして、Oracle データベースのバージョンは Oracle 8i Release 8.1.5 を使用します。また、構築方法は Database Configuration Assistant で簡単に作成することができますが、ここでは手動で作成する方法を紹介します。

3.1 サービスの作成

はじめに Oracle インスタンス用のサービスを作成します。サービスを作成するにはコマンドとして ORADIM を使います。書式は以下のようになります。

```
ORADIM -NEW -SID SID名 -INTPWD パスワード  
-STARTMODE AUTO -PFILE 初期化パラメータファイルのパス
```

ORADIM コマンドは ORACLE_HOME の BIN フォルダに入っています。SID 名はシステム識別子、またはインスタンス名と言われ Oracle データベースでは必須になっています。特に Net8 でクライアントから Oracle データベースに接続するときには必ずこの名前が必要になります。パスワードは SYSDBA 権限のパスワードです。このインスタンスでデータベースを作成できるのは SYSDBA の権

限を持っているユーザだけです。STARTMODE は NT サービスのモードの設定で起動時に自動で起動させるようにするので AUTO となっています。最後に初期化パラメータファイルですが、データベースを起動するときにさまざまな条件を設定するファイルです。データベース運用に関する重要なパラメータがたくさん入っています。インスタンス名や制御ファイルのパス、ブロックのサイズなどいろいろで、あらかじめ調べておく必要があります。次に ORADIM コマンド実行の一例を紹介します。

```
ORADIM -NEW -SID kansai -intpwd oracle -  
startmode auto -pfile  
e:\oracle\admin\kansai\pfile\initkansai.ora
```

実行すると図 4 に示すように NT のサービスにインスタンス用のサービス名が表示されます。この例はインスタンス名が kansai で SYSDBA のパスワードが oracle、インスタンスのスタートモードが自動となっています。初期化パラメータファイルの名前は決め方が決まっております

```
init インスタンス名.ora
```

となっています。ちなみに新たに ORADIM コマンドを使って Oracle インスタンスを作成するときは環境変数の設定が必要になります。Database Configuration Assistant を実行したときはそのツールがやってくれますが、手動で行う場合は ORACLE_SID という環境変数を設定します。レジストリを変更するか MS-DOS プロンプトを開いて環境変数 ORACLE_SID を設定します。これはこれから構築する Oracle データベースのインスタンス名になります。逆に Oracle のインスタンスを削除したい場合は同じコマンド ORADIM を使います。次に書式を示します。

```
ORADIM-DELETE-SID SID名
```

SID 名は作成のときに指定した SID 名になります。次に一例を紹介します。先ほど作成したインス



図 4 NT サービス

タンスを削除する場合があります。

ORADIM-DELETE-SID kansai

これでインスタンス名kansaiが削除されサービスの欄からも名前がなくなります。

3.2 データベース作成

Oracle インスタンスが作成できたら続いてはデータベースの作成になります。はじめに作成したインスタンスにSYSDBA 権限を持って接続します。SYSDBA 権限はインスタンスを作ったときのパスワードを使います。以前はINTERNAL という特別な名前がありましたが、現在では互換性のためだけに残しているようで将来的にはなくなりそうです。また、Oracle インスタンスに接続するときに使われていたツールでSVRMGR コマンドがありますが、Oracle8i からはSQL*PLUS に統合されています。ですから、SVRMGR で行っていたことはSQL*PLUS で行うことができます。SVRMGR コマンドは8.1.5 では、まだ残っています。ここではSQL*PLUS を使ってこれから先を行います。

はじめにSQL*PLUS を実行します。MS-DOS プロンプトを開いて次のように入力します。

SQLPLUS /NOLOG

するとプロンプトが「SQL>」になります。続いて次のコマンドを実行しインスタンスに接続します。次のコマンドを入力します。

CONNECT SYS/パスワード AS SYSDBA

パスワードはSYSDBA 権限を持っているパスワードです。また、SYS はユーザ名であらかじめSYS とSYSTEM というユーザ名があります。

インスタンスに接続できたら次にはインスタンスを起動します。次のコマンドを実行します。

STARTUP NOMOUNT PFILE=初期化パラメータファイルのパス

STARTUP コマンドはインスタンスを起動する

コマンドです。起動にもいくつか種類がありますが、ここではデータベースが作成できていないのでNOMOUNT オプションでインスタンスを起動します。初期化パラメータファイルのパスはインスタンス作成時に使われたものと同じものです。次に一例を紹介します。

startup nomount pfile=

e:\oracle\admin\kansai\pfile\initkansai.ora

続いてデータベースの作成になります。次にCREATE DATABASE コマンドの書式を紹介します。

CREATE DATABASE データベース名

[CONTROLFILE REUSE]

[LOGFILE [GROUP 番号] ファイルパス,

[GROUP 番号] ファイルパス]

[MAXLOGFILE 数値]

[MAXLOGMEMBERS 数値]

[MAXLOGHISTORY 数値]

[MAXDATAFILES 数値]

[MAXINSTANCES 数値]

[ARCHIVELOG|NOARCHIVELOG]

[CHARACTER SET 文字セット]

[NATIONAL CHARACTER SET 文字セット]

[DATAFILE ファイルパス size 数値[K|M]]

[REUSE] AUTOEXTEND {OFF|ON} [NEXT 数値

[K|M]] [MAXSIZE {UNLIMITED|数値[K|M]}]

このコマンドの各内容は詳しくはマニュアルを参照してください。ここでは簡単に次のように指定して実行します。

CREATE DATABASE kansai

maxlogfiles 32

maxlogmembers 5

maxdatafiles 30

maxloghistory 100

logfile group 1

('e:\oracle\oradata\kansai\log1\redo0101.log', 'e:

\oracle\oradata\kansai\log2\redo0102.log') size

```

2M, group 2
('e:\oracle\oradata\kansai\redo0201.log', 'e:
\oracle\oradata\kansai\redo0202.log') size
2M
datafile
'e:\oracle\oradata\kansai\database\system01.db
f' size 50M
character set JA16SJIS

```

これを実行するとデータベースが作成されます。図5にCREATE DATABASEの実行結果を示します。逆にOracleデータベースではデータベースを削除するコマンドはありません。他のデータベースであればDROP DATABASEコマンドがありますが、Oracleデータベースはありません。それで、データベースを削除する場合はインスタンスを削除してデータファイルを削除することになります。



図5 データベース作成

3.3 データディクショナリの作成

データベースが構築できた次はデータディクショナリを作成します。データディクショナリは対応付けられているデータベースに関する情報を提供する読み取り専用の一連の表とビューです。例えば、ユーザがテーブルを作るとこのデータディクショナリに登録されます。

すると「ユーザが作成したテーブルは」とOracleデータベースに問い合わせることができることになります。データディクショナリの構築は非常に簡単です。あらかじめ、CATALOG.SQLというコマン

ドスクリプトファイルがあるのでそれを実行するだけです。CATALOG.SQLはORACLE_HOMEフォルダのRDBMSフォルダのAdminフォルダにあります。ですから、SQL*PLUS上で以下のように実行します。

```

SQL>@%ORACLE_HOME%\rdbms\admin\cata
log

```

するとスクリプトが実行されてデータディクショナリが作成されます。時間はかなりかかります。途中、エラー表示されますが、これはスクリプトの中で存在しないスキーマオブジェクトを削除しようとするからです。スクリプトの作り方としてこれから作成するスキーマオブジェクトがあることを前提にしているからです。問題はありませんので気にしないでください。

3.4 PL/SQLの作成

Oracleデータベースのプログラミング言語として有名なPL/SQLがあります。これもオプションでOracleデータベースソフトウェアのインストールやCREATE DATABASEコマンドを実行しただけでは存在しません。データベースを作成した後にスクリプトを実行してPL/SQLを作成します。PL/SQL作成のスクリプトはORACLE_HOMEのRDBMSフォルダの中のAdminフォルダの中のCATPROC.SQLになります。これも実行するとかなりの時間がかかります。実行して画面を見ているとエラーが表示されますが、これはデータディクショナリの作成で紹介した内容と同じです。SQL*Plusで実行するには以下のようにします。

```

SQL>@%ORACLE_HOME%\rdbms\admin\catp
roc.sql

```

これでPL/SQLが使えるようになります。

3.5 表領域の作成

ここまでで最低限のデータベースができ上がりま

した。ユーザを作ったり表を作ったりできます。CREATE DATABASEを実行するときDATA FILEを指定してデータベース作成と同時にSYSTEM表領域を作成しています。ですから、データディクショナリやPL/SQLをデータベースの中に作成できました。通常、使うときは用途別に表領域を作成します。表領域はOracleデータベースの論理構造でこの表領域ごとにデータベースの中を切り分けていくとはじめの方でも紹介しました。Oracleのツールとしてデータベース構築を助けるDatabase Configuration Assistantがあります。これを使うと必要な表領域としてSYSTEM, USER, ROLLBACK, INDEX, TEMPORARYという5つの表領域を作成します。これらの表領域はそれぞれ役割があります。SYSTEM表領域にはデータディクショナリやPL/SQLなど全体の管理情報が保存されます。USER表領域にはこのデータベースを使うユーザのスキーマオブジェクトが保存されます。ROLLBACK表領域はトランザクションで使われるロールバックセグメントを保存する領域です。ロールバックセグメントはトランザクションが実行されてからコミットまたはロールバックするまでのビフォアイメージを保存しています。このおかげでトランザクション実行中に障害が発生した場合にトランザクション開始前の状態に戻すことができます。INDEX表領域には索引が保存されます。表を作成するときに検索の効率を上げるために索引を作りますが、その索引を保存する場所です。TEMPORARY表領域はデータベースが処理を行う場合に一時的にデータを格納するための作業領域です。表の結合や索引作成、Order By句、Group By句、Analyzeコマンドなどのソートで使われます。

それでは次に表領域を作成するコマンドCREATE TABLESPACEの書式を紹介します。

```
CREATE TABLESPACE 表領域名
DATAFILE 'ファイルパス' SIZE 数値
DEFAULT STORAGE
(INITIAL エクステントサイズ
NEXT エクステントサイズ
```

```
MINEXTENTS エクステント数
MAXEXTENTS エクステント数
PCTINCREASE 増加率)
```

DATAFILEでは作成するファイルのパスを記述します。表領域1つに対して1つ以上の物理ファイルが対応します。

SIZEはファイルの容量を指定します。INITIALでは初期のエクステント数を指定します。NEXTでは表領域に割り当てられる次のエクステントサイズを指定します。MINEXTENTSでは表に対して割り当てられる最小のエクステント数を指定します。MAXEXTENTSでは表に対して割り当てられる最大のエクステント数を指定します。PCTINCREASEでは3回目以降に割り当てられるエクステントの増加率を指定します。次に表領域作成の簡単な例を紹介します。

```
create tablespace kansai
datafile 'e:\oracle\oradata\kansai\database\
kansai.dbf' size 3M
default storage
(initial 50K
next 50K
minextents 1
maxextents 121
pctincrease 0)
```

実際に実行すると次のようになります。



4. セミナー実施報告

ここでは関西職業能力開発促進センターで行っているRDB利用技術（Oracle編）のセミナー実施報告と来年度予定のPL/SQLプログラミングの紹介を行います。

4.1 RDB利用技術（Oracle編）

RDB利用技術（Oracle編）は2年前から行っています。使用環境ですが、Oracleデータベースは、はじめの方でも紹介しましたがOracle 7 Server Enterprise Edition Release7.3.4です。Windows NT Server4.0 SP3上に構築しています。クライアントからサーバにアクセスする形をとっておりクライアントではSQL*Plusを使っています。セミナーの内容は市販の本と自作テキストを使っています。市販の本で一通りすすめて、足りない部分を補う形で自作テキストを使っています。RDB利用技術（ORACLE編）のメインはOracleデータベースを利用したSQLプログラミングです。市販の本の内容では主に以下の項目を取り上げています。

- ・リレーショナルデータベースの基礎
- ・SQLの基礎
- ・演算子
- ・関数
- ・SELECT文の句
- ・表の結合
- ・サブクエリー
- ・データの操作
- ・表の作成と保守
- ・ビューとインデックス
- ・トランザクション
- ・セキュリティ

これらでリレーショナルデータベースの基礎的な内容とSQLの文法、Oracleデータベースでのスキーマ、トランザクションを紹介します。自作テキストではSQL*Plusの使い方、Oracle特有の関数、自己結合、置換、テーブル作成時の制約、トランザクションを紹介しています。時間的に余裕があれば、SQL*Loaderも紹介しています。表に平成11年度、平成12年度の実績を示します。

表よりいずれのコースも定員以上の応募がありま

(単位 人)

年度	定員	応募	修了者数
11年	18	26	17
11年	18	24	17
12年	18	31	17
12年	18	24	18
12年	18	28	18

す。全体で定員90人に対して応募人数は133人で定員に対する応募率が約147%になっています。簡単に考えれば年にもう2コース設定できることになります。

4.2 PL/SQLプログラミング

今年度まではRDB利用技術（Oracle編）だけを行ってきましたが、来年度計画としてPL/SQLを考えています。PL/SQLはOracleデータベース専用言語です。Oracleデータベース自体はリレーショナルデータベースでSQL言語を使うことができます。SQL言語は非手続き型言語で単文のプログラムしか作成できません。それで、手続き的なプログラムも作成できるようにと作られたのがPL/SQLです。SQLを実行する形式としてはPL/SQLプログラムの中にSQLプログラムを埋め込み実行します。Oracleデータベースを扱っている方にはよく知られた言語でOracleデータベースを運用管理していくうえでは必須に近いものになっています。また、Oracleアプリケーション開発ソフト（OracleFormなど）では開発用の言語としても使われています。実際に計画している内容の詳細は以下の項目になります。

- ・PL/SQLの概要
- ・無名ブロック
- ・PL/SQL言語の基礎
- ・フロー制御
- ・カーソル
- ・プロシージャ
- ・ファンクション
- ・ストアードプロシージャ
- ・ストアードファンクション
- ・パッケージ
- ・PL/SQLのエラー処理
- ・トリガー
- ・組み込みパッケージ

これらで一通りのPL/SQLに関する内容を習得できます。セミナーを実施する側からするとある程度プログラミング経験が必要だと思います。どのセミナーでもそうだと思いますが、PL/SQLは特にそう思います。それはデバッグ環境がないこと、エラーメッセージがわかりづらいことがあげられます。実際にセミナーでは使用ソフトとしてSQL*Plusを考えています。SQL*PlusではPL/SQLのプログラムを作成および実行できますが、デバッグ環境はありません。

5. おわりに

RDB利用技術（Oracle編）は今年度で2年目になります。今年はコース数を増やしましたが昨年と同じような応募率でした。コースを1つ増やしたので応募人数から考えれば増えています。ですから、数的にはもっとコース数を増やすこともできます。

そして、来年度の計画ではRDB利用技術（Oracle編）を導入としてPL/SQLプログラミング、Oracle8i導入技法を体系的に実施することを考えています。PL/SQLはOracleデータベース専用のプログラミング言語でこれを習得できればOracleデータベース運用管理はもとよりアプリケーション開発にも幅ができます。ある意味、Oracleデータベースにとっては必須の言語と言えます。PL/SQLはOracle 7で導入されOracle8およびOracle8iで拡張されています。PL/SQL自体はもともとプログラミング言語のAdaをベースに開発されました。Ada自体はオブジェクト指向の言語なのでPL/SQLでもそのオブジェクト指向の内容が入っています。

Oracle8i導入技法では本編でも紹介しましたが、ベースはOracleデータベースの構築が主になります。Oracleデータベースは構成が複雑になっています。データベースを構築するところなど少しでも間違えると作り直しになったり、はじめて行ったときなど間違えるとなぜ間違ったのかなど分からなかったりします。特にエラーはきちんと表示されないのでおさらわかりません。そして、Oracleデータベースはかなりのパラメータがあります。初期化パラメータからはじまり、データベースを作成するためのパラメータ、表領域作成のためのパラメータなど

紹介しました。はじめてOracleデータベースを構築する方にとってはかなりの苦勞になると思います。一方、今現在行っているOracleデータベースを段階的に展開していくことを考えるとOracleデータベース運用・管理、チューニング、バックアップ・リカバリ、アプリケーション開発、データベース設計などがあります。今後はこういった内容を見据えて体系的なセミナー開発を考えています。

<参考文献>

- 1) URL : www.oracle.co.jp
- 2) MbCD訳 : Oracle7 SQL入門, アスキー, 1998年8月11日.
- 3) MbCD訳 : Oracle7 データベース開発, アスキー, 1997年3月4日.
- 4) 木村博文監修 : SQLプログラミング, ソフトバンク, 1998年12月14日.
- 5) 木村博文, 他著 : 入門SQL, ソフトバンク, 2000年1月24日.
- 6) 梅田弘之, 他著 : Oracle8入門, エーアイ出版, 2000年4月21日.
- 7) OPEN DESIGN 1999年6月号 : Oracle&Java データベース構築技法, CQ出版.
- 8) 日本オラクル株式会社 : Oracle 7 マニュアル 「Oracle7 Server 概要」.
- 9) 日本オラクル株式会社 : Oracle7 マニュアル 「Oracle7 Server 管理者ガイド」.
- 10) 宮原徹監訳 : Oracle システム設計, オーム社, 1997年12月.
- 11) SE編集部訳 : Oracle8 アーキテクチャ, 翔泳社, 1998年3月11日.
- 12) SE編集部訳 : Oracle8 ビギナーズガイド, 翔泳社, 1997年12月16日.
- 13) SE編集部訳 : Oracle8 DBA ハンドブック, 翔泳社, 1998年5月30日.
- 14) SE編集部訳 : Oracle8 PL/SQL プログラミング, 翔泳社, 1999年5月30日.
- 15) 原陽一監訳 : オラクルPL/SQLプログラミング 基礎編, オーム社, 1996年10月.
- 16) 原陽一監訳 : オラクル PL/SQLプログラミング 応用編, オーム社, 1997年1月.
- 17) 石井史子訳 : Oracle PL/SQLプログラミング, ピアソン, 1998年11月.
- 18) 望月康司著 : ORACLE マスターハンドブック Silver編, リックテレコム, 2000年1月15日.
- 19) 望月康司著 : ORACLE マスターハンドブック Gold編, リックテレコム, 2000年4月16日.