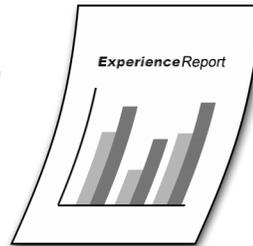


iSeries



Oracle Migration Toolkit

Experience Report



iSeries



Oracle Migration Toolkit

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： iSeries

Oracle Migration Toolkit

Experience Report

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2005.8

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 2005. All rights reserved.

© Copyright IBM Japan 2005

目次

第 1 章 Oracle の移行	1	データ転送スクリプトの生成	9
移行プロジェクトの管理	1	DB2 UDB for iSeries への配置	11
ソースの指定	2		
ソース・メタデータの変換	4	第 2 章 特記事項	13
メタデータ変換の詳細化	8		

第 1 章 Oracle の移行

この文書では、Oracle Migration Toolkit (MTK) を使用して Oracle^(TM) データベースを DB2^(R) Universal Database^(TM) for iSeries^(TM) に移行するプロセスについて説明します。このプロセスは 6 つのサブタスクに分かれており、それぞれについて詳細に説明しています。この報告を作成するために、ツールの「移行ウィザード」部分は使用しませんでした。ツールが提供する各サブタスクについてこの報告書で説明するため、すべてのサブタスクを通して実行しました。

移行プロジェクトの管理

ソースの指定

ソース・メタデータの変換

メタデータ変換の詳細化

データ転送スクリプトの生成

DB2 for iSeries への配置

詳細情報の検索

MTK ツールには、広範囲をカバーするオンライン・ヘルプが用意されており、各トピックがこの経験報告よりも詳細に説明されています。ただし、このツールはさまざまな DB2 プラットフォームに対して実行するよう作成されているため、汎用ヘルプの詳細の一部は iSeries への配置には該当しないか、または、ヘルプに記載されていません。この経験報告は、そのような理由のもとに作成されました。本書は特に iSeries での移行プロセスを対象としたものであるため、ヘルプ・テキストの補足として本書をご使用ください。

DB2 UDB for iSeries に関する包括的な情報は、iSeries Information Center (<http://www.ibm.com/eserver/iseries/infocenter>) を参照してください。

移行プロジェクトの管理

Oracle Migration Toolkit (MTK) を使用すると、作業をプロジェクトにまとめることができます。各移行プロジェクトは個別に保持されるため、それぞれの移行時に生成されるすべてのファイルを、簡単に管理することができます。一度プロジェクトを作成すれば、以前に作成したプロジェクトにいつでも戻って開くことができます。MTK を始動するときに、新規プロジェクトを作成するか、以前に作成したプロジェクトを開くかを選択できます。

MTK で新規プロジェクトを作成するには、以下のステップを実行します。

1. プロジェクト名を指定する。
2. プロジェクトのパスを指定する。このパスの下にすべてのファイルが保管されます。MTK はプロジェクト名をサブディレクトリーとして追加し、それを絶対パスの一部に組み込みます。このため、すべての移行プロジェクトにおいて、同じプロジェクト・パスを使用することをお勧めします。プロジェクト名が異なるため、各移行プロジェクトは個別のディレクトリーに保管されますが、すべてをグループ化しておくことができるというメリットがあります。

3. プロジェクトのテキスト記述を入力する。同じソース Oracle データベースを移行する予定の場合でも、複数のプロジェクトを作成して処理を整理することが必要な場合があります (例えば、複数のステップで作業を行う場合)。わかりやすい説明を入力しておく、各プロジェクトで行う作業を区別する際に役立ちます。
4. ソース・データベースとバージョンを指定する。(ただし、現時点でサポートされているソース・データベースは、Oracle バージョン 8 のみです。)
5. ターゲット・プラットフォームを指定する。DB2 がインストールされているシステムで、Oracle データベースの移行先とするシステムを指定します。iSeries での移行の場合は、使用する iSeries サーバーのバージョンとリリースを選択する必要があります。DB2^(R) Universal Database^(TM) for iSeries のリリースが違くと、サポートされている機能が異なるため、これは非常に重要です。現時点でサポートされているバージョンとリリースの組み合わせは、バージョン 5 リリース 2 とバージョン 5 リリース 3 です。使用する iSeries サーバーにこれらのリリースのどちらもインストールされていない場合は、V5R2 を選択します。一部の機能は、使用するシステムで稼働しない可能性があるため、正しく配置されません。ただし、V5R2 を選択することで、MTK が DB2 に V5R3 専用の機能を生成しないようにすることができます。
6. 「OK」をクリックする。

例

MTK を始動するとすぐに「プロジェクト管理 (Project Management)」ダイアログ・ボックスが表示されます。プロジェクト名として「payro11」と入力し、パス名はデフォルト値のままにします。作成する可能性のあるその他の移行プロジェクトとこのプロジェクトを区別できるように、「第 1 四半期の給与計算ファイルの移行」のような、分かりやすい説明を入力します。

次に、ターゲットとして DB2 UDB for iSeries を指定します。ドロップダウン・リストを使ってプラットフォームを「DB2 UDB iSeries V5R2」に変更して、バージョン 5 リリース 2 が稼働している iSeries システムにデータを移行することを指定します。これでプロジェクトの設定は完了です。「OK」をクリックします。

ソースの指定

Oracle Migration Toolkit (MTK) では、スクリプト・ファイルのセットを基にして、または Oracle データベースから直接、移行のソースを指定することができます。

データ自体はこの段階では処理されないという点に注意してください。ここでは単に、対象となる Oracle オブジェクトを指定し、その定義がフォームに変換されて iSeries システムの DB2 内に同じオブジェクトを作成できるようにします。これらのオブジェクトに含まれているデータの移動は、後続のステップの一部です。このステップでは、Oracle データベースと厳密に一致する空の データベースを DB2 上に作成します。

スクリプト・ファイルを使用したソースの指定

変換する SQL ステートメントを含むファイル (またはファイルのセット) を指定することができます。この方法は、Oracle データベースを作成するうえで必要なすべてのステートメントを含むスクリプトを、既に持っている場合に使用します。この方法を使うと、MTK はすべての入力ファイルから SQL 構文を取得して、それらを Oracle 互換 SQL から DB2 互換 SQL に変換します。

ファイルのセットに、相互依存性を持つすべてのファイルが含まれていることを確認してください。これは、変換プロセスによる DB2 互換 SQL の生成時に、関連するオブジェクトが参照される場合があるためです。例えば、スクリプトに CREATE INDEX ステートメントが含まれている場合は、基礎となるテーブル

ルの CREATE TABLE ステートメントが、ソースとして指定するファイルまたはファイルのセットに含まれている必要があります。基礎となるテーブルが既に移行済みの場合でも、その CREATE TABLE ステートメントを含める必要があります。

移行プロセスの次のフェーズである**変換**により、特定のソース・ファイルがコンテキスト提供専用として選択されているかどうかを指定します。これは、テーブルを再度移行したくない場合には便利ですが、索引やビューなどの依存オブジェクトを正しく変換するためには CREATE TABLE ステートメントが必要です。

移行するスクリプト・ファイルのセットを指定するには、以下の手順を実行します。

1. Oracle データベース・オブジェクトの作成に必要なすべての SQL ステートメントを含むスクリプトがあるかどうかを確認する。これには、テーブル、ビュー、索引、プロシージャ、トリガー、関数などが含まれます。
2. 「**インポート (Import)**」をクリックして、変換する SQL ステートメントを含むすべてのファイルを選択する。その後、必要に応じて、選択したファイルを表示、ソート、または除去することができます。
3. この時点でスクリプト・ファイルをソートして、オブジェクトの論理フロー、およびそれらの依存関係が存在することを確認する。最初にテーブルが作成され、次に、作成されたテーブル上にビュー、索引、およびトリガーが作成され、場合によってはその後に、オブジェクトの一部またはすべてに依存する SELECT ステートメントまたは CREATE PROCEDURE ステートメントが作成されます。**変換**ステップでも、ファイルをソートすることができます。
4. 「**変換 (Convert)**」タブをクリックして、移行プロセスを継続する。

Oracle データベースからのソースの直接指定

移行のソースを指定する 2 番目の方法として、記述を Oracle データベースから直接抽出する方法があります。この方法は、Oracle データベースを記述したスクリプト・ファイルがない場合、または、MTK 自体を使って移行対象のオブジェクトをサブセット化する場合に使用します。

この方法を使用するには、MTK の実行に使用するマシン上で Oracle データベースがアクセス可能である必要があります。この方法では、Oracle カタログおよびメタデータを使用して、DB2 へ移行できるすべてのオブジェクトのリストを生成します。

Oracle データベースを直接指定するには、以下の手順を実行します。

1. 「**抽出 (Extract)**」をクリックして、使用する Oracle データベースに接続するインターフェースをオープンする。移行する Oracle データベースの名前を、ユーザー ID およびパスワードと共に指定します。
2. 接続されると、Oracle データベース内の選択可能なすべてのオブジェクトがリストされる。移行するオブジェクトを選択します。
3. 選択したら、「**抽出 (Extract)**」をクリックして、SQL ステートメントを含むファイルを生成する。「**1 つのファイル / ストアド・プロシージャを作成 (Create one file / stored procedure)**」オプションおよび「**他の必要なオブジェクトを含める (Include other needed objects)**」オプションにチェックマークを付けることをお勧めします。1 つのファイルを作成することによって、将来の移行でこのファイルを使用することができます。これは、移行が既に完了しているためです。また、この処理には時間がかかる場合があります。他の必要なオブジェクトを含めるよう選択することで、見落としした可能性のある依存オブジェクトをツールに識別させることができます。
4. 「**変換 (Convert)**」タブをクリックして、移行プロセスを継続する。

詳細

通常、SQL スクリプト・ファイルには、ステートメント間に散在するコメントが含まれています。このようなコメントは以下のように表示されます。

```
-- The following statement creates a table

create table oracle_worker(
  Name varchar2(25),
  Age number check(Age between 18 AND 65),
  Lodging varchar2(15));
```

セミコロンが SQL ステートメントの終了区切り文字として使われていることにも注意してください。

ソース・メタデータの変換

これまでのステップで、移行プロジェクトを作成し、DB2^(R) Universal Database^(TM) for iSeries に移行される Oracle データベース内のオブジェクトのセットを定義しました。このステップでは、これらの定義を使用して、SQL を DB2 互換 SQL ステートメントに変換します。

ソース・メタデータを DB2 互換のスクリプトに変換するには、以下のステップを実行します。

1. デフォルト値を指定変更する場合は、生成されたファイルに接頭部を指定する。デフォルトでは、このツールは強調表示されたソース・ファイルの名前を使用しますが、それを、よりわかりやすい名前に変更することをお勧めします。この名前は、『変換』ステップで作成される .db2 ファイルおよび .rpt ファイルに対する接頭部であり、MTK によって生成されるファイル名にのみ影響します。データベース・オブジェクト名自体には何の影響もなく、出力スクリプトの内容にも影響しません。
2. 日付リテラルを持つ SQL ステートメントがある場合は、必ず、選択可能な項目から正しい「**ソースの日付形式 (Source date format)**」を選択する。日付リテラルがない場合、または不明な場合には、デフォルトの形式のままにしてください。
3. 他のオブジェクトにコンテキストを提供するためにのみリストに含まれているソース・ファイルがあるかどうかを確認する。該当するファイルがある場合は、「**コンテキストの設定 (Set Context)**」をクリックし、スクリプト内のその他のオブジェクトによって参照されるオブジェクトへの、定義の提供にのみ使用されるファイルをすべて選択します。

コンテキスト・ファイルとして指定したファイル内の SQL ステートメントは、出力スクリプトに変換されません。したがって、すべてのファイルをコンテキスト・ファイルに指定できるわけではありません。最低 1 ファイル、可能であれば複数のファイルを、ソース・ファイルのままにする必要があります。

4. 『**ソースの指定**』ステップで述べたようなファイルの順序を設定していない場合は、ファイルのリストの下にある「**ファイルを上へ移動 (Move File Up)**」または「**ファイルを下へ移動 (Move File Down)**」ボタンを使用して、ソース・ファイルの順序を変更する。
5. 必要な場合は、「**拡張オプション (Advanced Options)**」および「**グローバル・タイプ・マッピング (Global Type Mapping)**」を表示できますが、何らかの変更を加えることはお勧めできません。「**グローバル・タイプ・マッピング (Global Type Mapping)**」は、SQL ステートメントの変換時に使用される Oracle の型および対応する DB2 の型を参照したい場合に、使用することができます。
6. 「**変換 (Convert)**」をクリックする。これにより、変換プロセスが開始します。処理されるステートメントの数および複雑さによっては、終了までに数分を要する場合があります。
7. 変換プロセスが完了すると、「**詳細化 (Refine)**」タブが開き、変換プログラム情報が表示される。ただし、この時点で「**変換 (Convert)**」タブに戻ることをお勧めします。このタブでは、変換プロセスで生成された出力ファイルを確認することができます。特に、以下の 2 つのファイルに注目してください。

.db2 ファイルは、実際の出力スクリプトです。これには、DB2 互換 SQL に変換された、入力スクリプトの SQL ステートメントがすべて含まれています。一部のケースでは、ステートメントの修正、または省略が行われます。 .db2 ファイルには組み込みのコメントが含まれており、そのコメントにより、エラーが発生した場所、意図的にステートメントが省略された場所、およびステートメントが修正された場所が示されます。この .db2 ファイルは、入力スクリプトのオリジナルの各 SQL ステートメントが保存され、コメント化される、という優れた特長を備えています。それらの各コメントの後に、同等の DB2 互換 SQL ステートメントが続きます。これにより、変換プロセスで各ステートメントがどのように修正されたかが分かります。

.rpt ファイルは、すべてのメッセージをそのままリストしたものです。このファイルには、SQL ステートメントは 1 つも含まれていません。変換プロセスが生成したすべてのメッセージを単にレポートしたものです。このファイルを使用すると、変換プロセスがどの程度円滑に実行されたかを知ることができます。

詳細

名前の修飾

入力スクリプトでオブジェクトが修飾されている方法に応じて、出力されたスクリプトでは、テーブル、ビュー、索引、およびその他の SQL オブジェクトの名前が修飾される場合と、修飾されない場合とがあります。例えば、入力 Oracle スクリプト内のすべてのオブジェクトが修飾されていない (テーブル・スペースが指定されていない) 場合、出力スクリプト内のすべてのオブジェクトも修飾されません。入力スクリプト内のすべてのオブジェクトが同じテーブル・スペースを使用して修飾されている場合も、出力スクリプトは修飾されません。これらのケースでは、オブジェクトは、iSeries サーバーに配置される際に、ユーザーが DB2 への接続時に使用するプロファイルによって暗黙的に修飾されます。入力スクリプトで異なる複数の修飾子が使用されている場合は、最初に出現するものがデフォルトとなり、このデフォルトのテーブル・スペースを使用しているオブジェクトはいずれも、出力スクリプトでは修飾されていないように見えます。このデフォルト以外の修飾子を使用しているオブジェクトはすべて、出力スクリプトでは同じスキーマ名によって修飾されます。オブジェクトに対する最初の参照は修飾されているものの、入力スクリプト内のそれ以降の参照が修飾されていない場合は、特別なケースとなります。MTK では、このようなケースを、最初スキーマをデフォルトとし (したがって、出力スクリプトでこれらは修飾されない)、スキーマ名 DBA を使用して、後続の修飾されていない参照をすべて修飾する、という方法で処理します。

簡単な例を以下に示します。

1st reference:	tab1	output:	tab1
2nd reference:	tab2	output:	tab2
1st reference:	tab1 table space x	output:	tab1
2nd reference:	tab2 table space x	output:	tab2
1st reference:	tab1 table space x	output:	tab1
2nd reference:	tab2 table space y	output:	y.tab2
1st reference:	tab1 table space x	output:	tab1
2nd reference:	tab2	output:	dba.tab2

セット演算

DB2 UDB for iSeries では、V5R2 において、INTERSECT や EXCEPT などの一部のセット演算をサポートしていません。Oracle スクリプトにこれらの演算を使用する SQL ステートメントが含まれている場合は、UNION によって置き換えられます。これは、出力 SQL を確実に有効なものとするために行われますが、所期のものとは異なる結果が生成される可能性が高くなります。こ

これらの警告が発生した箇所を検討して、記述しなおす必要があるかどうか、あるいは UNION での置き換えが妥当であるかどうかを判断することをお勧めします。

SQL トリガー

他の DB2 プラットフォームでは、トリガー・プログラムを使用して、識別子 OLD を持つ古い行イメージや、識別子 NEW を持つ新しい行イメージを暗黙的に参照することができます。DB2 UDB for iSeries では、SQL ステートメント CREATE TRIGGER に、明示的な行 REFERENCING OLD AS OLD NEW AS NEW が含まれている必要があります。この文節は、CREATE TRIGGER ステートメントの変換プロセスで追加されます。

ラージ・オブジェクト (LOB) フィールド

DB2 UDB for iSeries の LOB フィールドのインプリメントでは、NOT LOGGED 属性をサポートしていません。ラージ・オブジェクトの型 (CLOB、BLOB、および DBCLOB) ではないずれも、DB2 UDB for iSeries へ変換される際に、NOT LOGGED 文節は除去されます。

CREATE INDEX

Oracle 内では使用されているが、DB2 UDB for iSeries には存在しないインデックス属性があります。CLUSTER や ALLOW REVERSE SCANS などの属性は、Oracle ではサポートされていますが、DB2 UDB for iSeries ではサポートされていません。これらの文節が入力スクリプト内に含まれている場合、DB2 出力スクリプトでは省略されます。

NOT NULL 文節

他の DB2 プラットフォームでは、UNIQUE および PRIMARY KEY 制約に含まれる列が、必ず NOT NULL として定義されている必要があります。DB2 UDB for iSeries では、そのような要件はありません。このため、移行のターゲットが DB2 UDB for iSeries である場合には、この文節は追加されません。

制約列の不一致

DB2 UDB for iSeries では、参照保全制約に含まれる列のデータ型が一致していなければなりません。主キー列のデータ型は、外部キーのデータ型と完全に同一である必要があります。よくある不一致の一例としては、INTEGER と SMALLINT の相違があります。この関係は、他の DB2 プラットフォームや Oracle では許可されていますが、DB2 UDB for iSeries では許可されていません。このエラーが発生する可能性を減らすために、変換では、Oracle の NUMBER データ型を INTEGER に変更し、INTEGER と SMALLINT との区別を試行しません。ただし、CHARACTER と VARCHAR のように、制約列が一致しないケースが残る可能性があり、一部の箇所です列定義の修正が引き続き必要となる場合があります。

NULL; ステートメント

Oracle のプロシージャ型言語である PL-SQL では、NULL; ステートメントを使用して、以下の簡単な例のように、空の論理分岐を示すことができます。

```
CREATE TRIGGER TRIG1 AFTER INSERT ON T
FOR EACH ROW
BEGIN
    NULL;
END;
```

DB2 UDB for iSeries では、こうした NULL; ステートメントの使用は許可されません。この PL-SQL ロジックの動作を保持するには、以下の例のように、NULL; ステートメントを、ダミー変数を 0 に設定する有効なステートメントで置き換えます。

```
CREATE TRIGGER TRIG1 AFTER INSERT ON T
REFERENCING NEW AS NEW FOR EACH ROW MODE DB2SQL
BEGIN ATOMIC

    DECLARE NULLSTMT INTEGER;
```

```
--| NULL;
      SET NULLSTMT = 0;
END;
```

変換されないステートメント

一部のステートメントは、入力スクリプトに含まれていても、出力スクリプトには組み込まれません。DROP TABLE などの一部のステートメントは、それらのステートメントを含めることに意味がありません。これは、移行が最初から開始されるので、テーブルの除去は必要ないためです。他にも、DB2 内では無効となる、特定の形式をもつステートメントがあります。この例としては、制約を使用可能または使用不可にする ALTER TABLE があります。ALTER TABLE ステートメントは DB2 でサポートされていますが、このステートメントを使用して制約を使用可能にしたり、使用不可にすることはできません。最後に、ステートメント自体を iSeries サーバー上で実行できないというケースがあります。この例としては、CREATE SEQUENCE ステートメントがあります。この機能は V5R2 ではまだ使用できないため、ステートメント全体が省略されます。これらすべてのケースにおいて、ステートメント・テキストはコメント化されますが、そのテキスト自体は出力スクリプト内に引き続き表示されます。また、コメントにメッセージが追加され、そのステートメントが変換されなかったこと、および変換されなかった理由が示されます。これらのコメントは、.rpt ファイルと .db2 ファイルの両方に表示されるため、除外されたすべてのステートメントを容易にチェックできます。

Oracle の型

DB2 では、以下の例のような、複数の要素を含む型をサポートしていません。

```
create type name_ty as object
  (firstName varchar2(20),
   lastName varchar2(20),
   title varchar2(10));
```

変換されないステートメントのもう 1 つの例を以下に示します。こうしたステートメントが含まれている場合は、この型を使用するテーブルを検索して、その型への参照を、明示的な列定義に置き換える必要があります。ユーザーの意図する内容が必ずしも明確ではないため、このツールでは CREATE TABLE ステートメントに対してそのような修正を行いません。ただし、次のような簡単なケースは除きます。

```
create table persons
  (deptNo int primary key,
   deptName char(20) not null,
   managerID number(5),
   managerName name_ty not null);
```

以下のように、型オブジェクトの列を組み込み、それらの名前を、そのテーブルに適したものに変更します。

```
create table persons
  (deptNo int primary key,
   deptName char(20) not null,
   managerID int,
   mgrfirstName varchar(20),
   mgrlastName varchar(20),
   mgrtitle varchar(10));
```

SEQUENCE オブジェクト

Oracle の SEQUENCE オブジェクトは、DB2 UDB for iSeries ではサポートされていません。ただし、以下のステップに従い、IDENTITY 列を使用して同じ振る舞いをさせることができました。例えば、Oracle にあるシーケンスが、次のステートメントによって作成されたとします。

```
CREATE SEQUENCE IDENTITY_C2
  INCREMENT BY 1
  START WITH 1000;
```

iSeries サーバー上で、1 つの IDENTITY 列でテーブルを作成します。

```
CREATE TABLE IDENTITY_C2 (IDENTITY_C2 DECIMAL(31,0) GENERATED ALWAYS AS IDENTITY
  (START WITH 1000, INCREMENT BY 1));
```

次に、シーケンスの前の値と次の値を与える関数を作成します。Oracle スクリプトを調べて、PREVVAL FOR IDENTITY_C2 を PREV_IDENTITY_C2() で置き換え、NEXTVAL FOR IDENTITY_C2 を NEXT_IDENTITY_C2() で置き換えます。以下の同じ例で、関数を作成するステートメントは次のとおりです。

```
CREATE FUNCTION PREV_IDENTITY_C2 () RETURNS DECIMAL(31,0) LANGUAGE SQL
MODIFIES SQL DATA
BEGIN
  RETURN (SELECT identity_val_local() FROM QSYS2.QSQPTABL);
END;
```

```
CREATE FUNCTION NEXT_IDENTITY_C2 () RETURNS DECIMAL(31,0) LANGUAGE SQL
MODIFIES SQL DATA
BEGIN
  INSERT INTO IDENTITY_C2 VALUES(DEFAULT) ;
  DELETE FROM IDENTITY_C2;
  RETURN (SELECT identity_val_local() FROM QSYS2.QSQPTABL);
END
```

SEQUENCE オブジェクトの完全なサポートが DB2 UDB for iSeries に追加されると、MTK で、Oracle スクリプトを同等の DB2 スクリプトに正しく変換できるようになります。

例

入力スクリプトで「25.12.2002」という日付リテラルが表示されるので、「ソースの日付形式 (Source date format)」に DD.MM.YYYY を選択します。

制約を使用不可にする ALTER TABLE ステートメントがあります。同様のことを実行するためには、WRKPFCSST コマンドを直接 iSeries システム上で使用します。これは、ALTER TABLE ステートメントを使用して制約を使用不可にすることができないためです。WRKPFCSST コマンドを使用すると、制約を使用可能および使用不可にすることができます。

メタデータ変換の詳細化

変換ステップでは、ソース SQL ステートメントを DB2 互換の SQL ステートメントに変換することを目的としていました。詳細化ステップでは、変換の結果を表示して変更を加えることができます。例えば、ソースのストアード・プロシージャと一部の DB2 オブジェクトの名前を変更することができます。ただし、これらの変更を適用するには、変換ステップに戻る必要があります。

以下のいずれかのタブを使用して、変換の結果を表示することができます。

- 「Oracle」ビューにはソース・データベースが表示されます。
- 「DB2」ビューにはターゲット・データベースが表示されます。
- 「レポート (Report)」ビューには、データベース・オブジェクト順にソートされたエラー・メッセージのレポートが表示されます。
- 「メッセージ (Messages)」ビューには、メッセージ番号順にソートされたメッセージが表示されます。

DB2 列に編集アイコン (鉛筆) が表示されている場合は、そのオブジェクトの名前を変更することができます。例えば、列、テーブル、ビュー、索引、およびプロシージャをの名前を変更できます。トリガー名および外部キー名は変更できません。このテーブルで加えた変更は全体に影響を及ぼし、ソース・メタデータを再変換した場合にのみ有効になります。オブジェクト名を変更するには、以下のステップを実行します。

1. 「詳細化 (Refine)」ページの「DB2」ビュー、またはソースでオブジェクトを選択する。
2. DB2 列の編集アイコン (鉛筆) をクリックする。「オブジェクト名編集 (Edit Object Name)」ウィンドウがオープンします。
3. 「名前 (Name)」フィールドに新しい名前を入力する。
4. 「適用 (Apply)」をクリックする。新しい名前がテーブルに表示されます。ただし、これらは次に変換するときまで有効にはなりません。

また、以下の手順を実行して SQL プロシージャまたはトリガーのロジックを変更することもできます。

1. 「詳細化 (Refine)」ページのソースまたはターゲット・ビューからプロシージャまたはトリガーを選択する。
2. 「ソース編集 (Edit Source)」を選択する。プロシージャ (トリガー) の本体がデフォルトのエディターでオープンします。
3. 変更内容を入力し、保管して、エディターを閉じる。
4. 「変更内容を最新表示 (Refresh Changes)」をクリックして、プロシージャ (トリガー) 本体に加えた変更を表示する。ここで新しい変更内容を表示できますが、次に変換するまで、この変更内容は有効にはなりません。

詳細

他のステップとは異なり、詳細化ステップはオプションであるということに注意してください。多くの場合、移行を行う際には、詳細化プロセスを適宜スキップすることができます。変換プロセスで生成されたメッセージをチェックし、オブジェクト名を変更する必要がない場合は、このステップ全体をスキップすることができます。

変換を詳細化する際には、個々のステートメントの変換をテストすることが役立ちます。このテストは、SQL Translator を使って実行することができます。最初に、テストするステートメントが参照しているオブジェクトを変換する必要があります。関連するオブジェクトの変換が完了した後、あるいは、テストするステートメントが完全な自己完結型である場合は、MTK メニューから「ツール (Tools)」>「SQL Translator」を選択します。「SQL Translator」ウィンドウがオープンします。このウィンドウに SQL ステートメントを入力するか、クリップボードからペーストすることができます。次に、「変換 (Convert)」ボタンをクリックします。

データ転送スクリプトの生成

この時点で、Oracle テーブルから iSeries 上の DB2 テーブルにデータを移動するスクリプトを生成する準備ができています。先ほどの変換ステップでは、データベース・オブジェクトの作成にのみ関係するスクリプトを生成しました。このステップでは、Oracle からデータをインポートして、DB2 ファイルにデータを取り込むスクリプトを作成します。オブジェクトを作成するスクリプトの配置のみに関心があり、Oracle データベースから DB2 にデータを移動しない場合は、このステップをスキップすることができます。

データ転送スクリプトを生成するには、以下の手順を実行します。

1. 画面の左側のパネルで、データの転送に使用するスクリプトを強調表示する。ここで選択するスクリプトは、データが転送される Oracle テーブルを決定するために使われます。Oracle オブジェクトは、「ソースの指定 (Specify Source)」画面で既に選択されているため、もう一度戻って選択する必要はありません。
2. スクリプトを保管する場所を決定する。スクリプト・ファイルは、MTK を実行しているローカルのマシン上に生成することも、リモートの iSeries サーバー上に生成することも、あるいは両方へ同時に生成することもできます。このパネルで選択する場所は、「DB2 への配置 (Deploy to DB2)」タブのデータ抽出ステップで作成される、バイナリー・データ・ファイルの保管先としても使用されます。

両方の場所へ同時にスクリプトを保管し、スクリプトの両方のセットに対してデフォルトの場所を指定することをお勧めします。ただし、大量のデータがあり、複数の iSeries サーバーに配置する予定の場合は、スクリプトをローカルにのみ生成することをお勧めします。このオプションは非常に高速で実行されます。ユーザーはその後でファイルを手動で各 iSeries サーバーに送信して、表示されるディレクトリー内にそれらを保管することができます。ローカルでのスクリプト生成を選択するもう 1 つのケースとしては、iSeries サーバーに接続していない状態で作業を行っており、この段階ではスクリプトの生成のみを行い、後でスクリプトを手動で複製したい、という場合があります。このシナリオでは、このウィンドウに表示される iSeries 宛先ディレクトリーにファイルを後で移動することもできます。

スクリプトを iSeries サーバー上だけに保管することを選択する唯一の理由は、使用するローカル・マシンの記憶域を節約するためです。この場合、*.qsh ファイルはローカル・マシン上にはないため、生成後にこのファイルを表示することはできません。ただし、スクリプト生成中にエラーが発生しなかった場合は、このファイルが iSeries 上に存在すると確信でき、すべて問題はありません。

3. 「スクリプトを生成 (Create Scripts)」をクリックする。これにより、データ転送に必要なスクリプトが作成され、指定したマシン上に保管されます。データの移動はまだ行われません。実際のデータの移動は、移行プロセスの最後の『DB2 UDB for iSeries への配置』ステップで行われます。

詳細

データを複数の iSeries サーバー上に複製する最も簡単な方法は、単に『データ転送スクリプトの生成』プロセスを実行してから、各 iSeries ごとに『DB2 for iSeries への配置』プロセスを実行することです。何らかの理由により Oracle データベースを使用できない場合や、単に Oracle データベースが大きすぎて再度抽出できない場合は、プロジェクト・ディレクトリー全体をターゲット・システムに手動で送信することができます。

- /QIBM/UserData/MTK/projects/*projname*/DataOutScripts データ・ディレクトリー全体を、他の iSeries サーバー上の適切なディレクトリーにバイナリー・コピーします。*.qsh スクリプト・ファイル内のコメントに、このファイルを保管すべき場所が正確に示されています。
- 新しいシステムに対して、MTK からデータ配置ステップを実行します (抽出ステップはスキップします)。あるいは、/QIBM/UserData/MTK/projects/*projname*/DataOutScripts/*name*.qsh を実行することで、STRQSH を使用して新しいシステム上で qsh スクリプトを手動で実行することもできます。

MTK ツールは、iSeries サーバー上の /QIBM/UserData/MTK/projects/*projname*/DataOutScripts ディレクトリーに、以下のような複数のファイルを生成します。

- *.qsh スクリプトは、CPYFRMIMPF を使って、実際にデータのインポートを行います。
- *.fdf ファイルは CPYFRMIMPF のフィールドを定義します。

MTK ログまたは出力メッセージからは診断できないような問題が発生した場合は、MTK を実行する際に -debug オプションを使用します。このオプションは、MTK ツール、およびデータ転送スクリプトの生成と実行から、詳細出力を生成します。この出力は、より詳細な問題判別を行ううえで活用で

き、また、分析のために IBM に送信することもできます。さらに、*.qsh スクリプトの実行によってメッセージに表示される CL コマンドを手動で実行して、問題をより詳細に診断することもできます。例えば、CHGPF CST、DLTF、CRTSRCPF、および CPYFRMIMPF の各コマンドを 1 つずつ実行して、問題の原因特定を試みることができます。

これらを MTK ツールから実行するのではなく iSeries サーバー上で手動で実行したい場合は、スクリプト自体にその実行方法を示すコメントが含まれています。

DB2 UDB for iSeries への配置

このステップが、移行プロセスの最終ステップです。これまでに、DB2 オブジェクトの作成、Oracle テーブルからのデータ抽出、および DB2 テーブルへのデータのインポートに必要な、すべてのスクリプトを作成しました。

iSeries の DB2 UDB へデータを配置するには、以下のステップを実行します。

1. 配置のターゲットとなる iSeries サーバーの名前を入力する。入力する名前は、JDBC ドライバーがリモート・システムにアクセスするために使用する、有効なシステム名である必要があります。
2. 適切なスキーマが iSeries サーバーに存在していることを確認する。最も一般的なケースにおいて、1 つの Oracle スキーマから Oracle オブジェクトをすべて抽出した場合、生成される DB2 スクリプトは、スキーマ名を含みません。つまり、デフォルトのスキーマ名が使用されます。この名前は、ユーザーが接続に使用するユーザー・プロファイルの名前です。ほとんどのユーザーは、既にこの名前の付いたスキーマを持っていますが、ない場合には、iSeries 上サーバーで手動で作成する必要があります。これは、オブジェクトが作成される場所です。iSeries ナビゲーターを使用してこのスキーマを作成できます。iSeries サーバーに直接サインオンし、STRSQL を使用して対話式 SQL 環境に入り、CREATE SCHEMA *myprofile* を実行することもできます。ここで、*myprofile* は、ユーザーが接続に使用するユーザー・プロファイルです。

自分のユーザー・プロファイルを使用したくない場合には、DB2 スクリプトを修正して先頭に SET SCHEMA ステートメントを組み込み、デフォルト・スキーマを自分のユーザー・プロファイル以外のものに設定することができます。ただし、プロセス全体で、デフォルト・スキーマを使用することをお勧めします。このオブジェクトは、必要に応じて、移行後にコピーまたは移動することができます。

また、既にスキーマがあり、必ずそこにオブジェクトが作成されるようにする必要がある場合は、その名前でも CRTUSRPRF を実行し、そのプロファイルを使用して接続する、という別の方法があります。

3. iSeries への接続用のユーザー ID とパスワードを入力する。最初の画面でこれらを入力しない場合、ウィンドウが表示されて、プロファイルおよびパスワードの入力を求めるプロンプトが出されます。現行のパスワードと共に、iSeries サーバー上にある有効なユーザー・プロファイルを指定する値を入力する必要があります。プロファイルが定義されていない場合、アクティブではない場合、またはパスワードが期限切れか誤りである場合には、接続は失敗し、配置プロセスは停止します。
4. .db2 スクリプトの配置のみを行うのか、またはデータの抽出とロードも行うのかを指示する。完全な移行を行う場合には、3 つのボックスすべてにチェックマークを付けます。これら 3 つのオプションは、完全な配置における以下の 3 つのステップに対応しています。
 - a. DB2 内にデータベース・オブジェクトを作成する。
 - b. Oracle からデータを抽出し、中間形式で iSeries サーバーに保管する。
 - c. 抽出されたデータを DB2 テーブルにインポートする。

3 つのステップすべてを一度に実行することをお勧めしますが、このルールには次のような例外があります。データではなく、オブジェクト定義のみを移行する場合は、1 番目のオプションのみにチェック

マークを付けます。既にデータベース・オブジェクトが作成済みであり、データの移動のみを行う場合には、2 番目と 3 番目のオプションにチェックマークを付けます。iSeries サーバー間で、すべてのスクリプト・ファイルとデータ・ファイルのコピーが完了している場合は、3 番目のオプションのみにチェックマークを付けます。

5. 「**配置 (Deploy)**」をクリックする。このステップは、処理に非常に長時間を要することがあります。MTK ツールでは、処理時間が増える可能性のある、いくつかのステップを実行します。すべてのデータは、リモートで抽出されて保管されるため、パフォーマンスは直接 Oracle テーブルのデータ量に比例します。データが抽出され、iSeries サーバーに保管されると、CPYFRMIMPF コマンドが各テーブルごとに実行されて、データが DB2 にインポートされます。テーブル内にラージ・オブジェクト (LOB) 列がある場合、各 LOB 値にはそれぞれ、iSeries 統合ファイル・システムに保管されている固有のファイルがあり、それらのファイルについてもインポートする必要があります。データの抽出を選択していない場合でも、DB2 内で作成しなければならないオブジェクトが多数存在する可能性があり、この処理にも数分を要します。
6. 結果の出力画面で、配置の結果を検討する。MTK ツールでは、移行に関する有用な情報を含む、非常に詳細な HTML ファイルを自動的に生成します。配置が完了すると、ブラウザのウィンドウが表示されます。このウィンドウには、移行されたすべてのオブジェクトが表示され、また、データの移動を選択した場合には、Oracle ソース・データベースと DB2 ターゲット・データベースの両方にあるすべての列の数が表示されます。このツールは、配置プロセスで発生したエラーまたは警告をすべて記載したテキストを生成します。これらのログおよびレポートはすべて、すべてのスクリプトおよびファイルと共にプロジェクト・ディレクトリーに保管されます。このため、ユーザーの作業は保管され、移行が完了した後も、再利用したり検討したりすることができます。

詳細

配置が失敗した場合、MTK ツールは、配置できなかったオブジェクトを示したレポートを生成します。そのオブジェクト名をクリックすると、試行されたそのオブジェクトの作成に関連するエラー・メッセージが表示されます。また、配置プロセス全体で生成されたすべてのメッセージを記録した配置ログもあります。このファイルは `filename_deploy.log` という名前で、すべてのスクリプトと共にプロジェクト・ディレクトリー内に含まれています。HTML 形式の配置レポートから、このログへの直接リンクもあります。このログには、メタデータの配置と、データ自体の配置の両方で生成されたメッセージが記録されています。スクリプトの手動での更新、または iSeries での何らかの作成または削除によって修正することのできる問題がある場合は、スクリプト全体を再実行できます。再実行の前に、スキーマを「除去」および「作成」して、各オブジェクトで既存の エラーを発生させないようにする必要があります。全スクリプトの再実行は行いたくない場合には、別の選択肢として、失敗したステートメント自体を使用して新しいスクリプトを作成するか、対話式 SQL または iSeries ナビゲーターを使用して、手動でその SQL ステートメントを実行する、という方法があります。

iSeries サーバーへの最初の配置時に、MTK ツールは、さまざまな Oracle 組み込み関数に代わる、ユーザー定義関数 (UDF) を生成します。これらの UDF は、ORA8 スキーマにビルドされます。このプロセスはすべて自動化されており、ORA8 スキーマが削除されるか、ORA8 スキーマに含まれる 1 つ以上の UDF が削除されない限り、1 回だけ実行されます。

重要: 配置を実行する前に、PTF SI06748 および SI06675 が iSeries システムに適用されているかどうかを検査して、必要なすべてのコードがサーバー上にあることを確認してください。これらの PTF を適用せずに配置プロセスを実行しても、正しく完了しません。

第 2 章 特記事項

本書の情報は特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。他社製品への言及および参照は、単に情報提供目的で記載されたものであり、IBM がそれらの製品を推奨するものではありません。

本書に含まれるパフォーマンス・データは、管理環境下で標準の IBM ベンチマークを使用し得られた測定結果と予測に基づくものです。ユーザーが実際に得られるスループットまたはパフォーマンスは、ユーザーのジョブ・ストリームにおけるマルチプログラミングの量、I/O 構成、記憶域構成、および処理されるワークロードなどの考慮事項によって異なります。したがって、個々のユーザーがここで述べる比率と同等のスループットまたはパフォーマンスの向上を得られるという保証はありません。



Printed in Japan