

dbrecover for oracleソフトウェアユーザーマニュアル

DBRECOVER for Oracle ソフトウェアユーザーマニュアル0.5

概要

DBRECOVER for Oracleは、Oracle 8iから21cのデータベースデータファイルから直接データテーブルのデータを抽出し、復元するエンタープライズレベルのOracle災害復旧ソフトウェアです。Oracleデータベースインスタンスを介してSQLを実行することなくデータを回復することができます。Javaで開発されたDBRECOVERは、追加のインストール不要で、ダウンロードして解凍するだけで直接使用できます。

DBRECOVERは直感的なGUIグラフィカルインターフェースを採用しており、操作が簡単です。ユーザーは特別なコマンドを学ぶ必要もなく、Oracleの下層データ構造の原理を理解する必要もなく、復旧ウィザードを通じてデータベース内のデータを容易に復旧することができます。

DBRecover for Oracle email: service@parnassusdata.com www.dbrecover.com Professional Oracle Database Disaster Recovery Version 2009

Start Options

Database

Database

- DB_20230529140659
 - Users
 - GSMROOTUSER
 - PD
 - Tables
 - EMP
 - SCOTT
 - Tables
 - BONUS
 - DEPT
 - EMP
 - SALGRADE
 - SYS
 - SYSTEM

TABLE: PD.EMP this view only shows some sample data

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-1987 00:00:00 AD	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981 00:00:00 AD	950		30
7902	FORD	ANALYST	7566	03-DEC-1981 00:00:00 AD	3000		20
7934	MILLER	CLERK	7782	23-JAN-1982 00:00:00 AD	1300		10
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-1987 00:00:00 AD	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981 00:00:00 AD	950		30
7902	FORD	ANALYST	7566	03-DEC-1981 00:00:00 AD	3000		20
7934	MILLER	CLERK	7782	23-JAN-1982 00:00:00 AD	1300		10
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-1987 00:00:00 AD	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981 00:00:00 AD	950		30

Version 2009 - Community Edition - 2020.09.25

Copyright © 2012 - 2020 ParnassusData Software, Inc.

<https://www.dbrecover.com>

なぜDBRECOVERを選ぶのでしょうか？

あなたは疑問に思うかもしれません。「従来のOracleリカバリマネージャーRMANを使ったバックアップと復元では不十分なのでしょうか？なぜDBRECOVERを選ぶ必要があるのでしょうか？」その疑問にお答えしましょう。

企業のITシステムが急速に成長する中で、データ容量は指数関数的に増加しています。Oracle DBAは、データの完全性を保証する際に、既存のディスクストレージシステムの容量が全量バックアップを保持するには不十分である、テープベースのデータバックアップのデータ復旧に必要な平均修復時間が予想を大幅に超えるといった問題に常に直面しています。

「データベースにとって、バックアップはすべてに優先する」というのは、すべてのDBAが心に留めている格言です。しかし、現実の環境は多様であり、企業のデータベース環境ではデータバックアップ用のスペースが不足していることがよくあります。また、購入したストレージデバイ

スが短時間で届かない、データ復旧の過程でバックアップが実際には使えないことが判明するなど、一般的な問題です。

このような現実世界で一般的に発生するデータ復旧の難題を解決するために、DBRECOVERソフトウェアは、Oracleデータベースの内部データ構造、コア起動プロセスなどの内部原理への深い理解を活かしています。これにより、バックアップが全くない状況でも、SYSTEM表領域の喪失、Oracleデータ辞書テーブルの誤操作、停電によるデータ辞書の不一致など、データベースがスムーズに開かない状況に対応できます。また、誤って切断 (Truncate) /削除 (Drop/Delete) されたビジネスデータテーブルなどの人為的な誤操作を取り戻し、容易にデータを復旧できます。

Oracleデータベースに数日間しか触れたことがない非DBAの方でも、DBRECOVERを簡単に使用できます。これは、DBRECOVERのシンプルなインストールプロセスとフルグラフィカルなユーザーインターフェースのおかげです。復旧を行う人は、専門的なデータベースの知識を必要とせず、どんなコマンドも学ぶ必要がなく、データベースの下層のストレージ構造を理解する必要もありません。マウスを数回クリックするだけで、容易にデータを復旧できます。

DBRECOVERは、データベース復旧作業を専門家だけに限定するという制約を打破し、データベース障害から完全なデータ復旧までの時間を大幅に短縮し、企業のデータ復旧の総コストを削減します。

DBRECOVERで復旧できるデータには、二つの形式があります。伝統的な抽出方法では、データをデータファイルから完全に抽出し、プレーンテキストファイルに書き込み、その後SQLLDRなどのツールを使用してデータベースに再度インポートします。この方法はシンプルで直感的ですが、現存するデータ容量の2倍のスペースが必要です：一つはプレーンテキストデータが占めるスペース、もう一つはテキストデータをデータベースにインポートするために必要なスペースです。さらに、時間的には、データファイルから原始データを抽出した後、新しいデータベースにインポートするまでの時間が通常2倍かかります。

私たちは、DBRECOVERの革新的なDataBridge方式を強く推奨します。この方法では、DBRECOVERを通じて抽出したデータを直接、新しいまたは他の利用可能なデータベースにロードし、データの地上での保管を避けます。伝統的な方法と比較して、データ復旧に必要なスペースと時間コストを効果的に節約します。

OracleのASM (Automatic Storage Management) 技術は、多くの企業で採用されています。伝統的なファイルシステムと比較して、ASMを使用したデータベースは高性能で、クラスターのサポートや管理の利便性があります。しかし、ASMの問題点は、一般ユーザーにとって、ASMの

ストレージ構造が複雑で理解しにくいことです。ASM内の特定のDisk Groupの内部データ構造が損傷し、MOUNTが成功しない場合、重要なデータがASMの「ブラックボックス」内に「ロックされる」ことがあります。このような場合、通常、Oracleの元の工場の資深エンジニアが現場に到着して、ASMの内部構造を手動で修復する必要があります。これは、一般ユーザーにとって高価で時間がかかることが多いです。

DBRECOVERの開発者はOracle ASMの内部データ構造に深い理解を持っているため、ASMに特化したデータ復旧機能がDBRECOVERに追加されました。

現在、DBRECOVERがサポートしているASMデータ復旧機能には以下が含まれます：

1. Disk Groupが正常にMOUNTできない場合でも、DBRECOVERを通じてASMディスク上の利用可能なメタデータ（metadata）を直接読み取り、これらのメタデータに基づいてDisk Group内のASMファイルをコピーすることができます。
2. Disk Groupが正常にMOUNTできない場合でも、DBRECOVERを通じてASMのデータファイルを直接読み取り、そこからデータを抽出することができます。これには、伝統的な抽出方法とDataBridge方法がサポートされています。

DBRECOVER For Oracleのソフトウェア紹介

DBRECOVER For OracleはJAVAベースで開発されており、AIX、Solaris、HPUXなどのUnixプラットフォーム、Redhat、Oracle Linux、SUSEなどのLinuxプラットフォーム、またWindowsプラットフォーム上でのクロスプラットフォーム実行が可能です。

DBRECOVERがサポートするオペレーティングシステムプラットフォーム：

プラットフォーム名	サポート状況
Windows	サポート
AIX	サポート
Solaris Sparc/X86	サポート
Linux x86/64	サポート
HPUX	サポート
MacOS	サポート

DBRECOVERは現在、8i~21Cのデータベースバージョンをサポートしています。

DBRECOVERは必要なJAVA環境を自身で備えているため、Windows/Linux上で別途JAVAソフトウェアをインストールする必要はありません。

Windowsでは、start_dbrecover_windows_local_java.batをダブルクリックして実行します。

Linuxでは、sh start_dbrecover_linux_local_java.shを実行します。

AIX/HPUX/SolarisなどのUNIXライク環境では、ユーザー自身でJAVA 8環境をインストールする必要があります。

DBRECOVERがサポートするデータベース文字セット：

言語	文字セット	エンコーディング
中国語 簡体/繁体	ZHS16GBK	GBK
中国語 簡体/繁体	ZHS16DBCS	CP935
中国語 簡体/繁体	ZHT16BIG5	BIG5
中国語 簡体/繁体	ZHT16DBCS	CP937
中国語 簡体/繁体	ZHT16HKSCS	CP950
中国語 簡体/繁体	ZHS16CGB231280	GB2312
中国語 簡体/繁体	ZHS32GB18030	GB18030
日本語	JA16SJIS	SJIS
日本語	JA16EUC	EUC_JP
日本語	JA16DBCS	CP939
韓国語	KO16MSWIN949	MS649
韓国語	KO16KSC5601	EUC_KR
韓国語	KO16DBCS	CP933
フランス語	WE8MSWIN1252	CP1252
フランス語	WE8ISO8859P15	ISO8859_15
フランス語	WE8PC850	CP850
フランス語	WE8EBCDIC1148	CP1148
フランス語	WE8ISO8859P1	ISO8859_1
フランス語	WE8PC863	CP863
フランス語	WE8EBCDIC1047	CP1047
フランス語	WE8EBCDIC1147	CP1147

ドイツ語	WE8MSWIN1252	CP1252
ドイツ語	WE8ISO8859P15	ISO8859_15
ドイツ語	WE8PC850	CP850
ドイツ語	WE8EBCDIC1141	CP1141
ドイツ語	WE8ISO8859P1	ISO8859_1
ドイツ語	WE8EBCDIC1148	CP1148
イタリア語	WE8MSWIN1252	CP1252
イタリア語	WE8ISO8859P15	ISO8859_15
イタリア語	WE8PC850	CP850
イタリア語	WE8EBCDIC1144	CP1144
タイ語	TH8TISASCII	CP874
タイ語	TH8TISEBCDIC	TIS620
アラビア語	AR8MSWIN1256	CP1256
アラビア語	AR8ISO8859P6	ISO8859_6
アラビア語	AR8ADOS720	CP864
スペイン語	WE8MSWIN1252	CP1252
スペイン語	WE8ISO8859P1	ISO8859_1
スペイン語	WE8PC850	CP850
スペイン語	WE8EBCDIC1047	CP1047
ポルトガル語	WE8MSWIN1252	CP1252
ポルトガル語	WE8ISO8859P1	ISO8859_1
ポルトガル語	WE8PC850	CP850
ポルトガル語	WE8EBCDIC1047	CP1047
ポルトガル語	WE8ISO8859P15	ISO8859_15
ポルトガル語	WE8PC860	CP860

対応している表ストレージタイプ：

表ストレージタイプ	対応状況
クラスタテーブル	YES
インデックス組織表、パーティション有り/無し	NO
普通ヒープテーブル、パーティション有り/無し	YES
普通ヒープテーブル 基本圧縮有効	NO

普通ヒープテーブル 高度圧縮有効	NO
普通ヒープテーブル 混合列圧縮有効	NO
普通ヒープテーブル 暗号化有効	NO
仮想フィールド付きテーブル	NO
チェーン行、移行行	YES

注意事項：

仮想カラムや11gで最適化されたデフォルトカラムに関しては、データ抽出は可能ですが、対応するフィールドは失われる可能性があります。これらは11g以降の新機能で、使用者は比較的少ないです。

DBRECOVERがサポートする列フィールドデータタイプ：

データタイプ	サポート状況
BFILE	No
Binary XML	No
BINARY_DOUBLE	Yes
BINARY_FLOAT	Yes
BLOB	Yes
CHAR	Yes
CLOB and NCLOB	Yes
Collections (VARRAYS、ネストテーブル含む)	No
Date	Yes
INTERVAL DAY TO SECOND	Yes
INTERVAL YEAR TO MONTH	Yes
SecureFilesとして保存されたLOB	Yes
LONG	Yes
LONG RAW	Yes
Multimediaデータタイプ (Spatial、Image、Oracle Text含む)	No
NCHAR	Yes
Number	Yes
NVARCHAR2	Yes

RAW	Yes
ROWID, UROWID	Yes
TIMESTAMP	Yes
TIMESTAMP WITH LOCAL TIMEZONE	Yes
TIMESTAMP WITH TIMEZONE	Yes
ユーザー定義型	No
VARCHAR2 と VARCHAR	Yes
CLOBとして保存されたXMLType	No
オブジェクトリレーショナルとして保存されたXMLType	No

DBRECOVERのASMサポート：

機能	サポート状況
ASMからデータを直接抽出し、ファイルシステムにコピーせずにサポート	YES
ASMからデータファイルをコピーしてサポート	YES

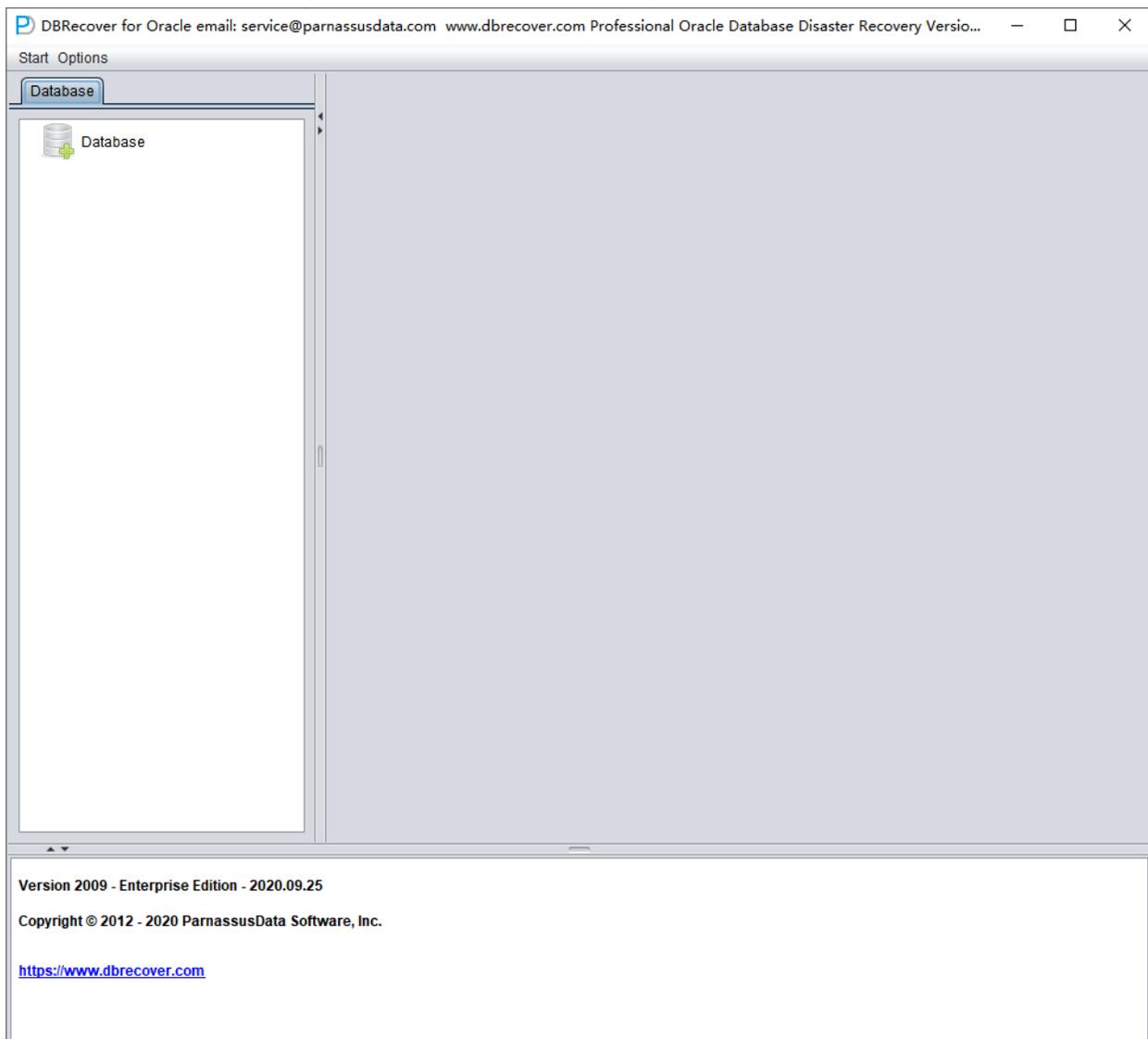
DBRECOVERのインストールと起動

DBRECOVERはJAVAベースの緑色ソフトウェアで、追加のインストールが不要です。ソフトウェアのZIPパッケージをダウンロードして解凍するだけで使用できます。

Windowsでは、start_dbrecover_windows_local_java.batをダブルクリックして実行します。

Linux環境では、ローカルのグラフィカルインターフェースやXmanager/VNCなどのリモートグラフィカルツールを使用できます。

1. xclockグラフィカルクロック小プログラムが開けることを確認します。
2. ソフトウェアの解凍ディレクトリで sh start_dbrecover_linux_local_java.sh を実行します。
- 3.



AIX/HPUX/Solaris環境でのDBRECOVERの使用

AIX/HPUX/Solaris環境では、ローカルのグラフィカルインターフェースやXmanager/VNCなどのリモートグラフィカルツールを使用できます。

1. 対応するプラットフォームのJAVA 8環境がインストールされていることを確認し、コマンド `java -version` で確認します。
2. xclockグラフィカルクロック小プログラムが開けることを確認します。

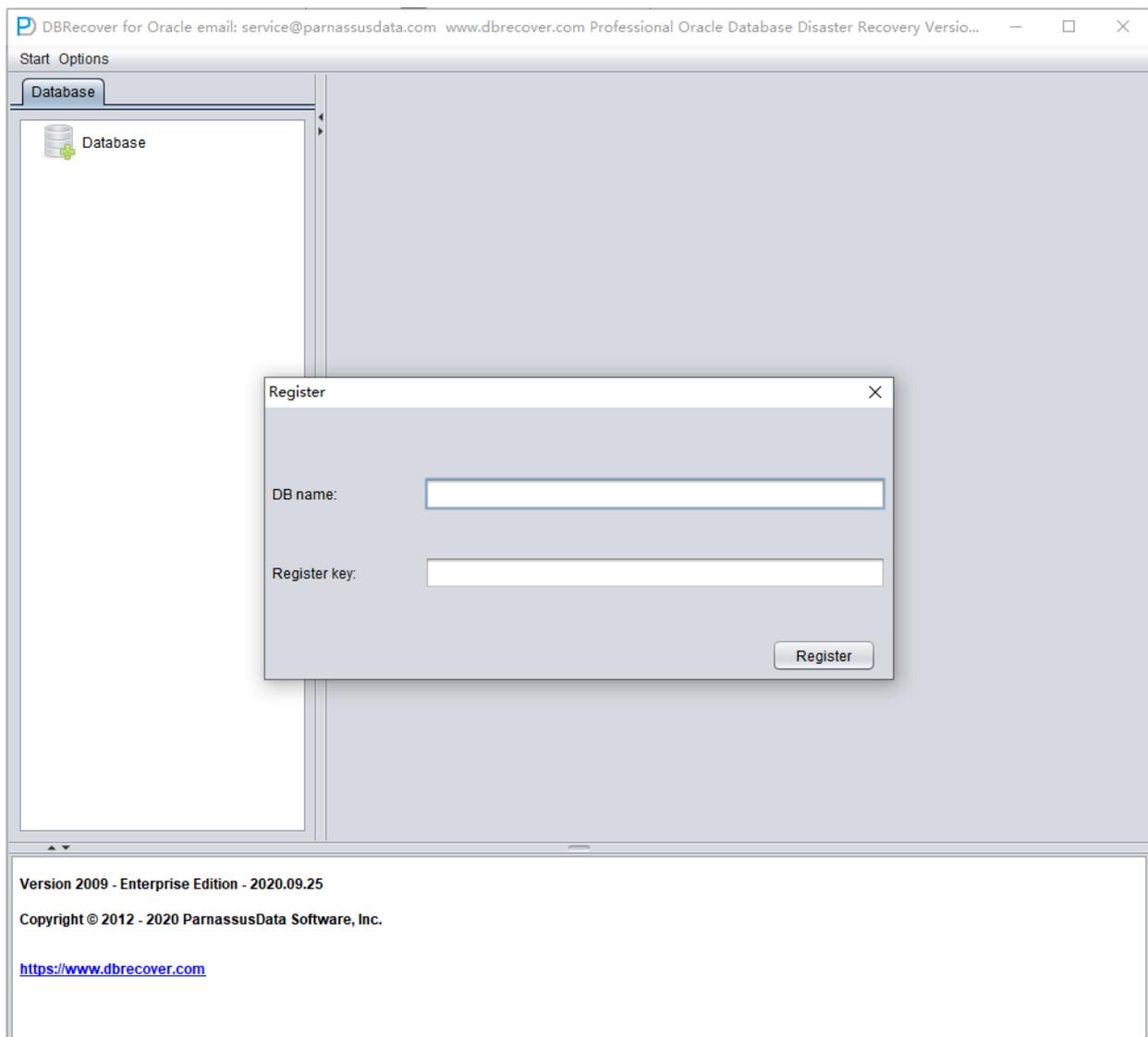
3. ソフトウェアの解凍ディレクトリで `sh start_dbrecover.sh` を実行します。

DBRECOVERのライセンス登録

DBRECOVER For Oracleは商業ソフトウェアです。DBRECOVERのコミュニティ版はテストや学習用として利用可能です。

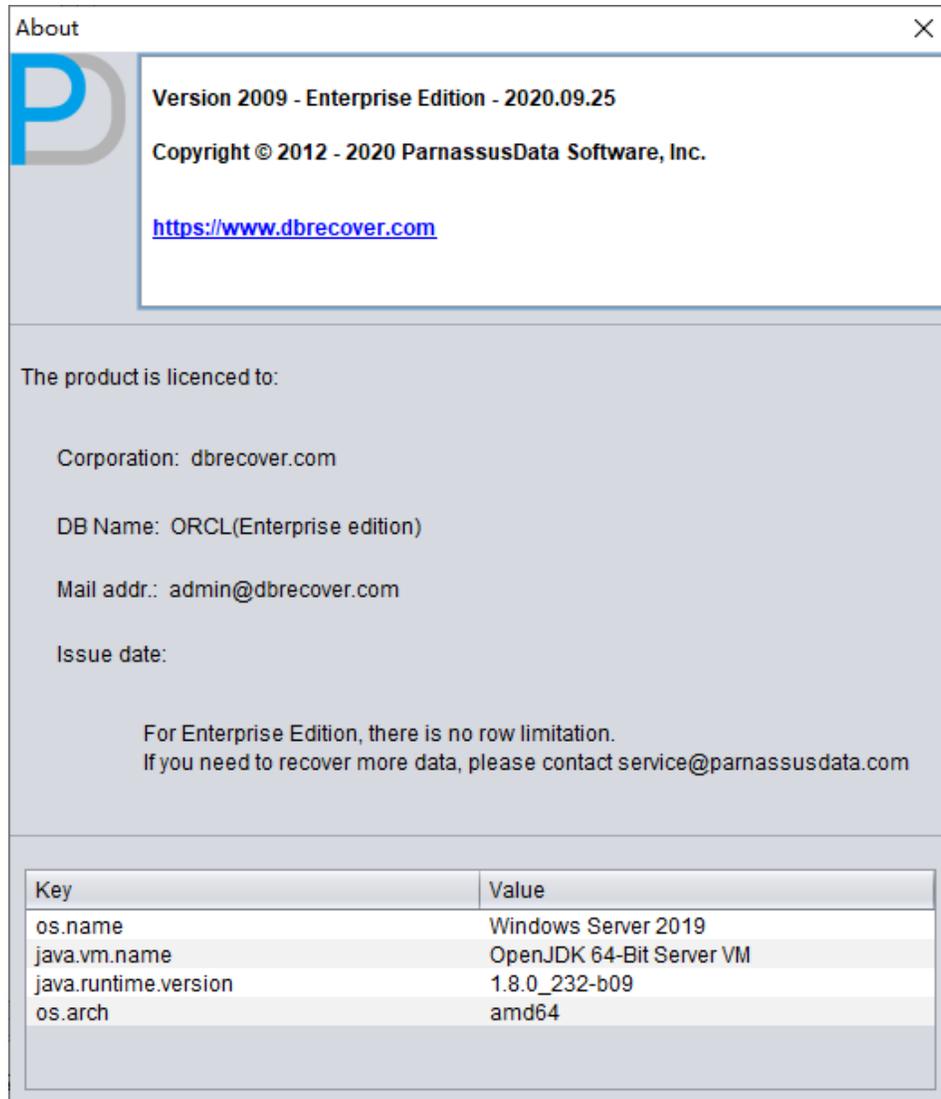
現在、企業版ライセンスのみを提供しています。購入情報は <https://www.dbrecover.com/> で入手できます。

ライセンスキーを取得した後、ソフトウェア内で自行に登録（Register）が可能です。具体的な使用方法は以下の通りです：



メニューバーの「Help => Register」から、購入時に送信された情報に従ってDB NAMEとキーを入力し、「Register」ボタンをクリックします。登録後、DBRECOVERを再起動するとライセンス登録情報が自動的にチェックされ、再登録の必要はありません。

成功した登録情報は「Help=>About」で確認できます。



DBRECOVERを使用した異なるOracleデータベースの復旧シナリオ

復旧シナリオ1：ORACLEデータファイルの損傷によりデータベースがOPENできない

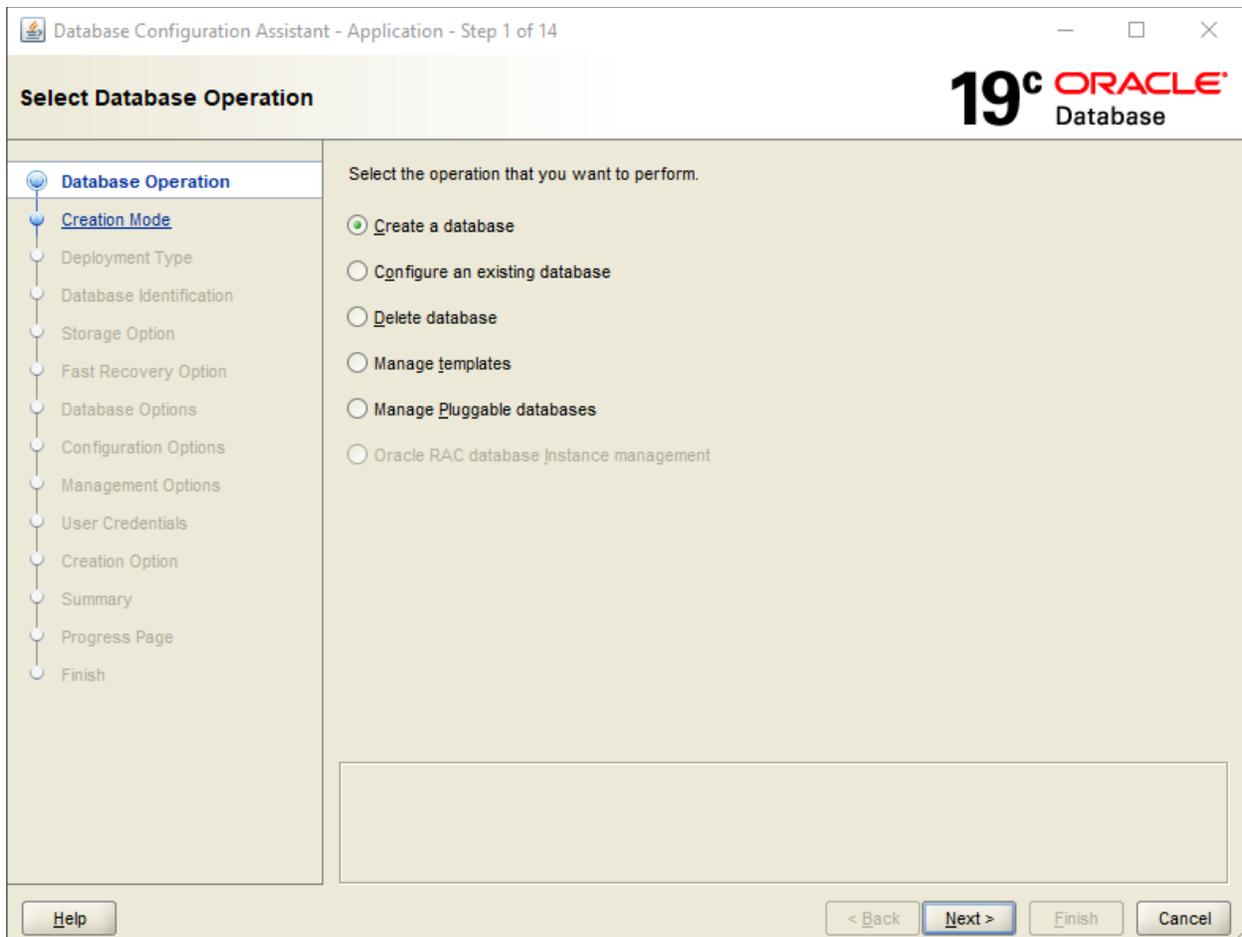
A社の生産データベースは非アーカイブモードで運用されており、たまにEXPの論理バックアップを行うが、物理バックアップは行わない。ある日、サーバーの停電後にデータベースが正常にOPENできなくなり、SYSTEM表領域の深刻な損傷が発見された。このような場合、

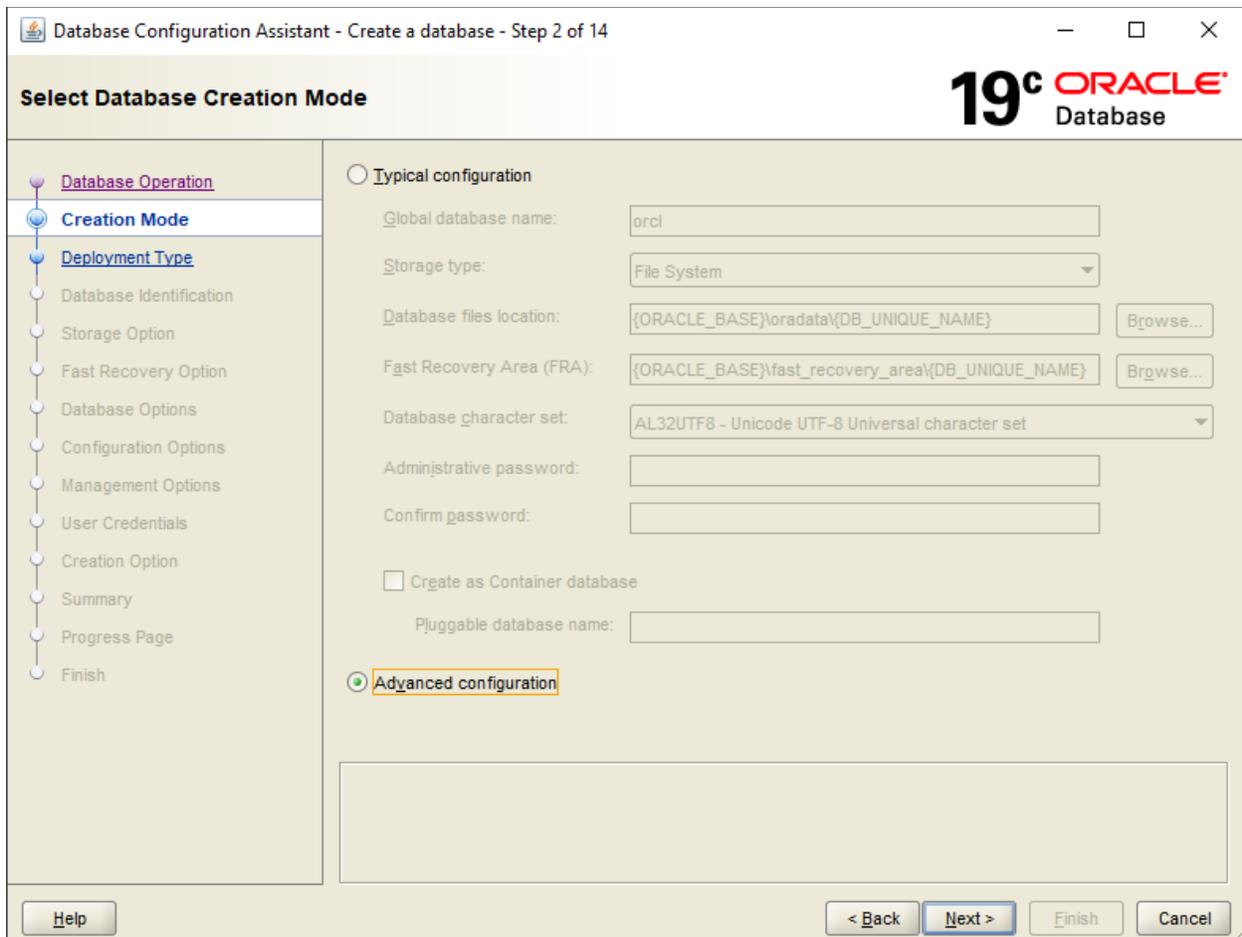
DBRECOVERを使用して、損傷したデータベースのデータを新しく作成したデータベースに迅速に転送し、ビジネスを迅速に復旧させることができる。

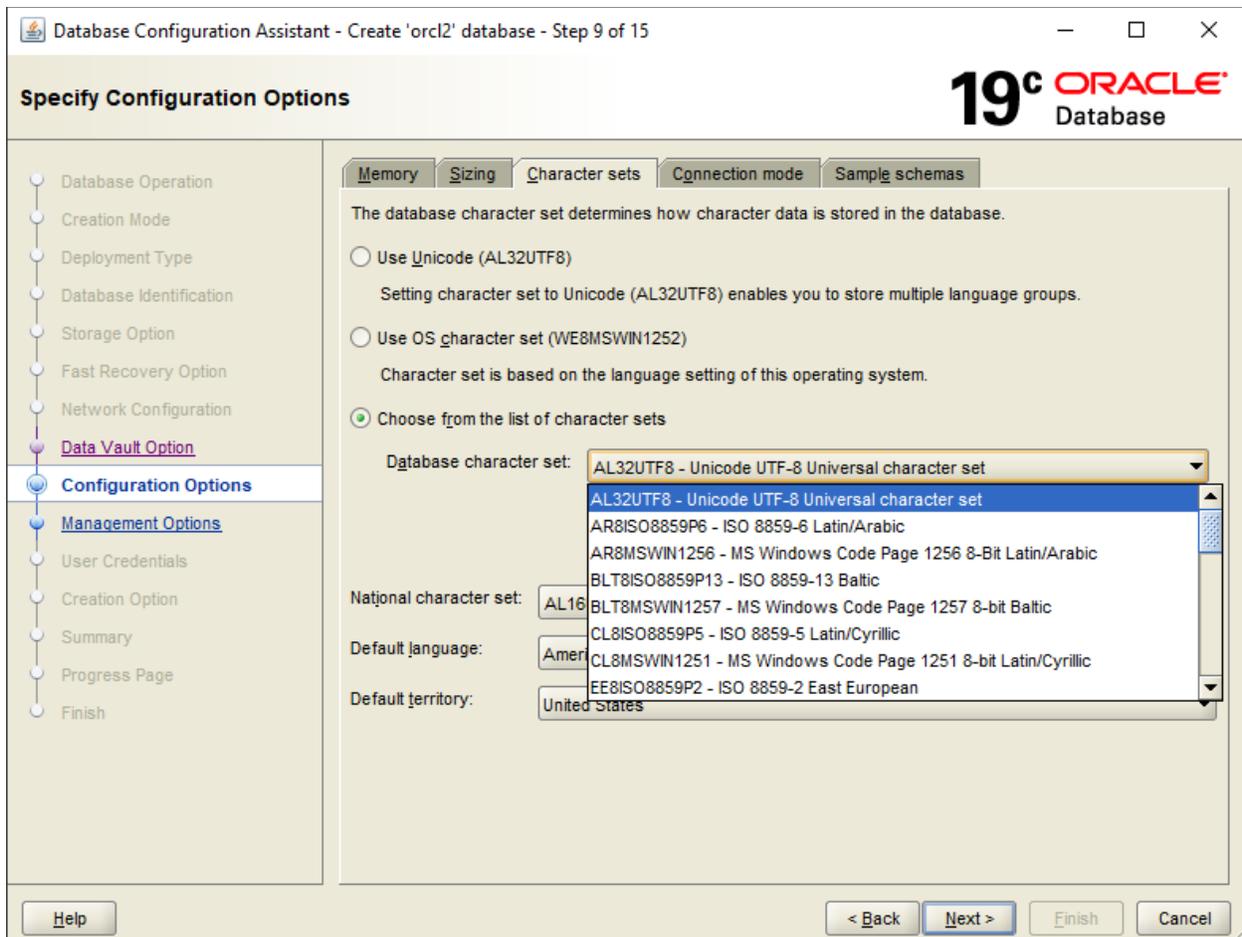
ORA-01194、ORA-01110、ORA-01033、ORA-01115、ORA-00368、ORA-00600
kcbzib_kcrsds_1、ORA-00333、ORA-01113、ORA-01122、ORA-27027などのエラーでデータベースが開けない場合も、この復旧シナリオの方法を試してみることができる。

以下は簡単な手順です：

1. dbcaを使用して新しいORACLEデータベースを作成し、文字セットを損傷したデータベースと一致させる。
2. 新しいデータベース内で対応するデータベースユーザーと表領域を作成し、暫定的にDBA権限を付与する。
3. リスナー（LISTENER）を起動し、データベースサービスがリスナーに登録されていることを確認する。
4. DBRECOVERを起動し、辞書モードで損傷したデータベースのすべてのデータファイルを読み込む。
5. DBRECOVER内で復旧するユーザー名を選択し、右クリックでデータブリッジを選択する。
6. データブリッジ画面でプラスアイコンをクリックし、新しいデータベースの接続情報（Connection）を追加する。
7. Data Bridgeをクリックして転送作業を開始し、SCHEMAのすべてのテーブルが目的のデータベースのSCHEMAに転送されるのを待つ。
8. 対象のSCHEMAを選択し、右クリックでEXPORTDDLを選択し、DDLを生成し、必要なオブジェクトタイプを選択してEXPORTする。
9. EXPORTDDLによって生成されたDDL SQLファイルをもとに、目的のデータベースの目的SCHEMAで手動で実行する。







```
C:\Users\testenv>lsnrctl status
```

```
LSNRCTL for 64-bit Windows: Version 11.2.0.1.0 - Production on 12-MAY-2023 10:01:48
```

```
Copyright (c) 1991, 2010, Oracle. All rights reserved.
```

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=DESKTOP-testenv)(PORT=1521)))
```

```
STATUS of the LISTENER
```

```
-----
```

```
Alias LISTENER
```

```
Version TNSLSNR for 64-bit Windows: Version 11.2.0.1.0 - Production
```

```
Start Date 12-MAY-2023 10:00:49
```

```
Uptime 0 days 0 hr. 0 min. 59 sec
```

```
Trace Level off
```

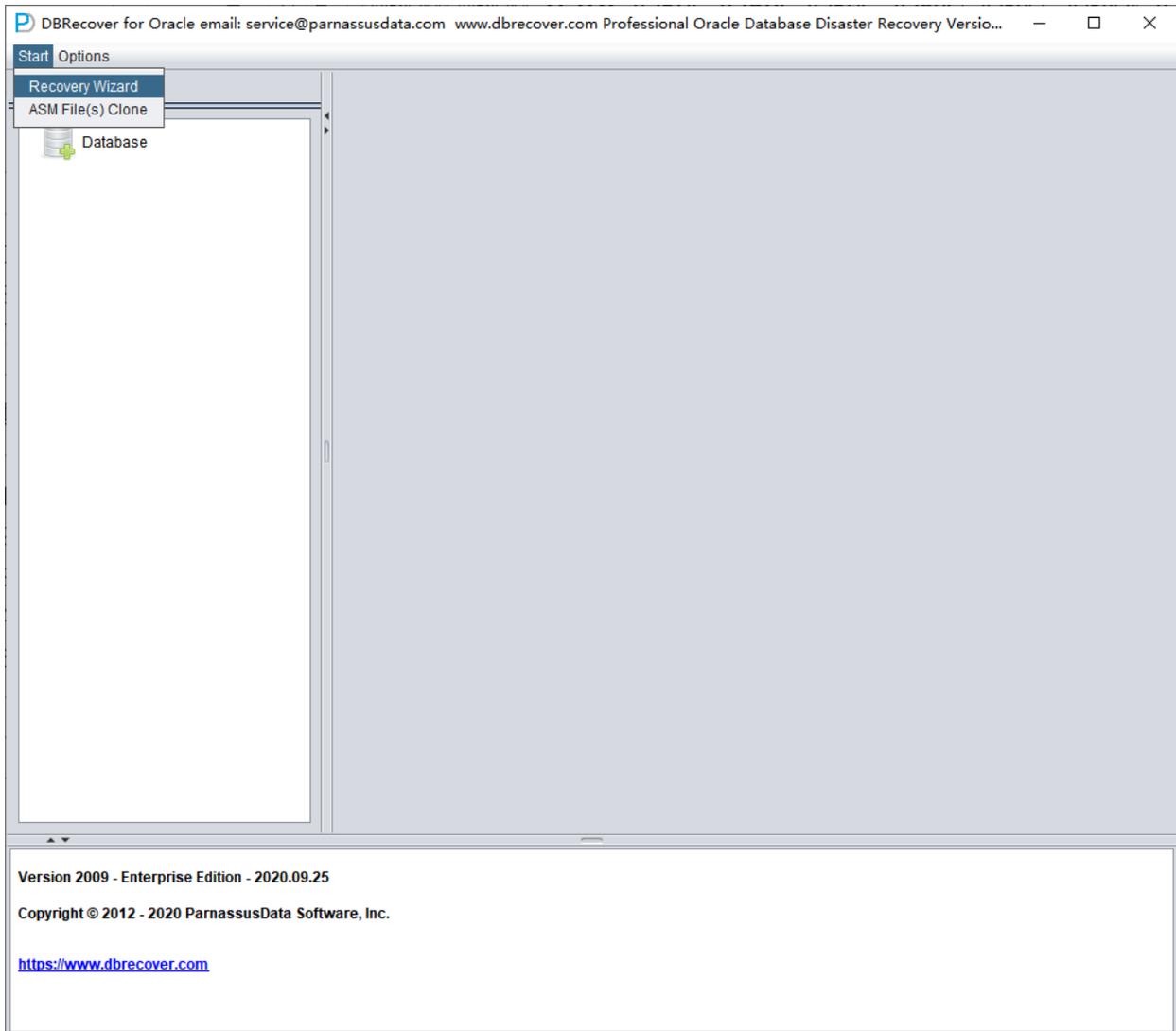
```
Security ON: Local OS Authentication
```

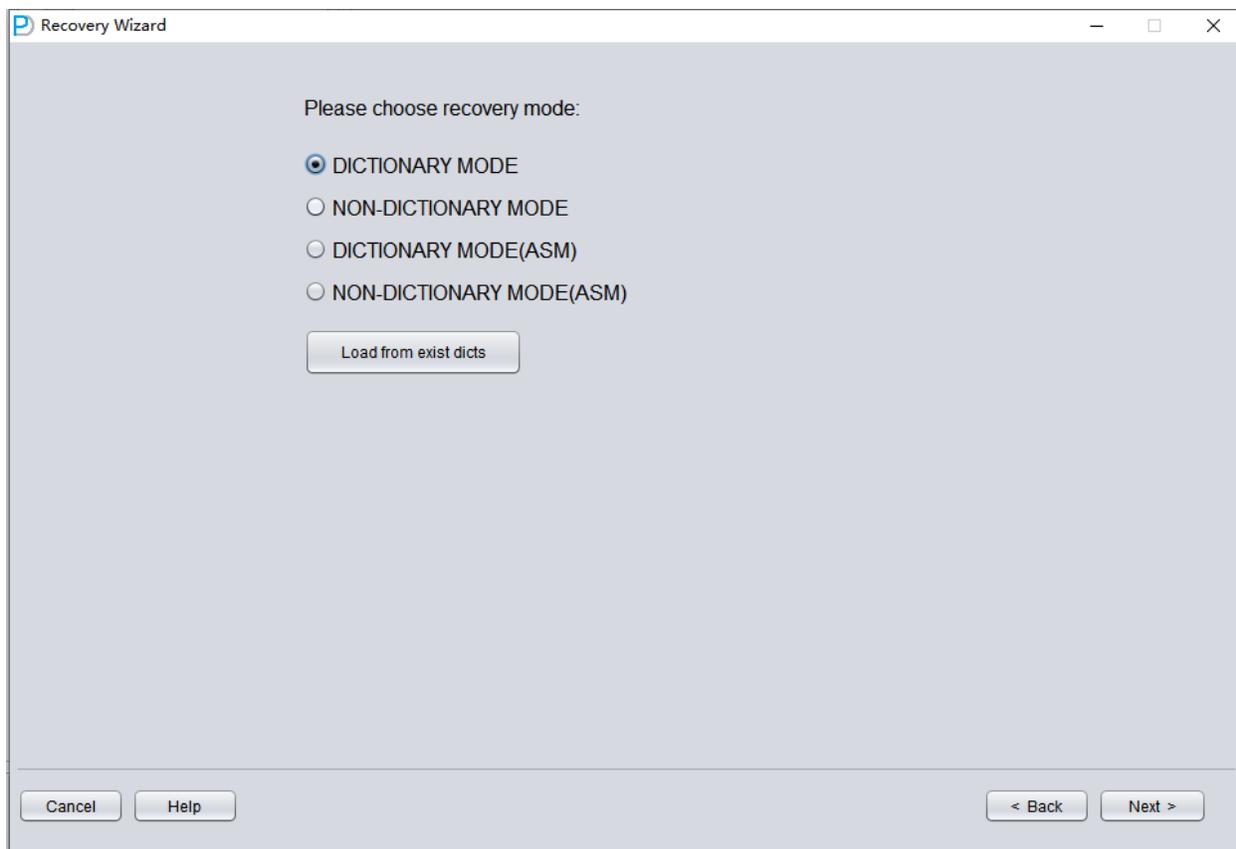
```
SNMP OFF
```

```
Listener Parameter File D:\app\testenv\product\11.2.0\dbhome_2\network\admin\listener.ora
Listener Log File d:\app\testenv\diag\tnslsnr\DESKTOP-testenv\listener\alert\log.xml
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=DESKTOP-testenv)(PORT=1521)))
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROC1521ipc)))
Services Summary...
Service "CLRExtProc" has 1 instance(s).
Instance "CLRExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "ORCL1XDB" has 1 instance(s).
Instance "orcl1", status READY, has 1 handler(s) for this service...
Service "ORCLXDB" has 1 instance(s).
Instance "orcl", status READY, has 1 handler(s) for this service...
Service "orcl" has 1 instance(s).
Instance "orcl", status READY, has 1 handler(s) for this service...
Service "orcl1" has 1 instance(s).
Instance "orcl1", status READY, has 1 handler(s) for this service...
The command completed successfully
```

```
set ORACLE_SID=ORCL1
sqlplus / as sysdba
SQL> create user pd identified by oracle;
User created.
SQL> grant dba to pd;
Grant succeeded.
SQL> create tablespace pdtbs datafile size 500M autoextend on next 100M;
Tablespace created.
SQL> alter user pd default tablespace pdtbs;
User altered.
```

DBRECOVERを起動し、「Tools => Recovery Wizard」を選択し、その後「Next」をクリックします。





次のステップでは、適切なENDIANバイト順序を選択します。WindowsやLinuxなどのよく使用されるプラットフォームはLittle Endianですので、デフォルト設定をそのまま使用します。対象のOracleデータベースファイルがLinux x86-64プラットフォーム上にある場合、EndianはLittleを選択します。

platform	endian
Solaris[tm] OE (32-bit)	Big
Solaris[tm] OE (64-bit)	Big
Microsoft Windows IA (32-bit)	Little
Linux IA (32-bit)	Little
AIX-Based Systems (64-bit)	Big
HP-UX (64-bit)	Big
HP Tru64 UNIX	Little

HP-UX IA (64-bit)	Big
Linux IA (64-bit)	Little
HP Open VMS	Little
Microsoft Windows IA (64-bit)	Little
IBM zSeries Based Linux	Big
Linux x86 64-bit	Little
Apple Mac OS	Big
Microsoft Windows x86 64-bit	Little
Solaris Operating System (x86)	Little
IBM Power Based Linux	Big
HP IA Open VMS	Little
Solaris Operating System (x86-64)	Little
Apple Mac OS (x86-64)	Little

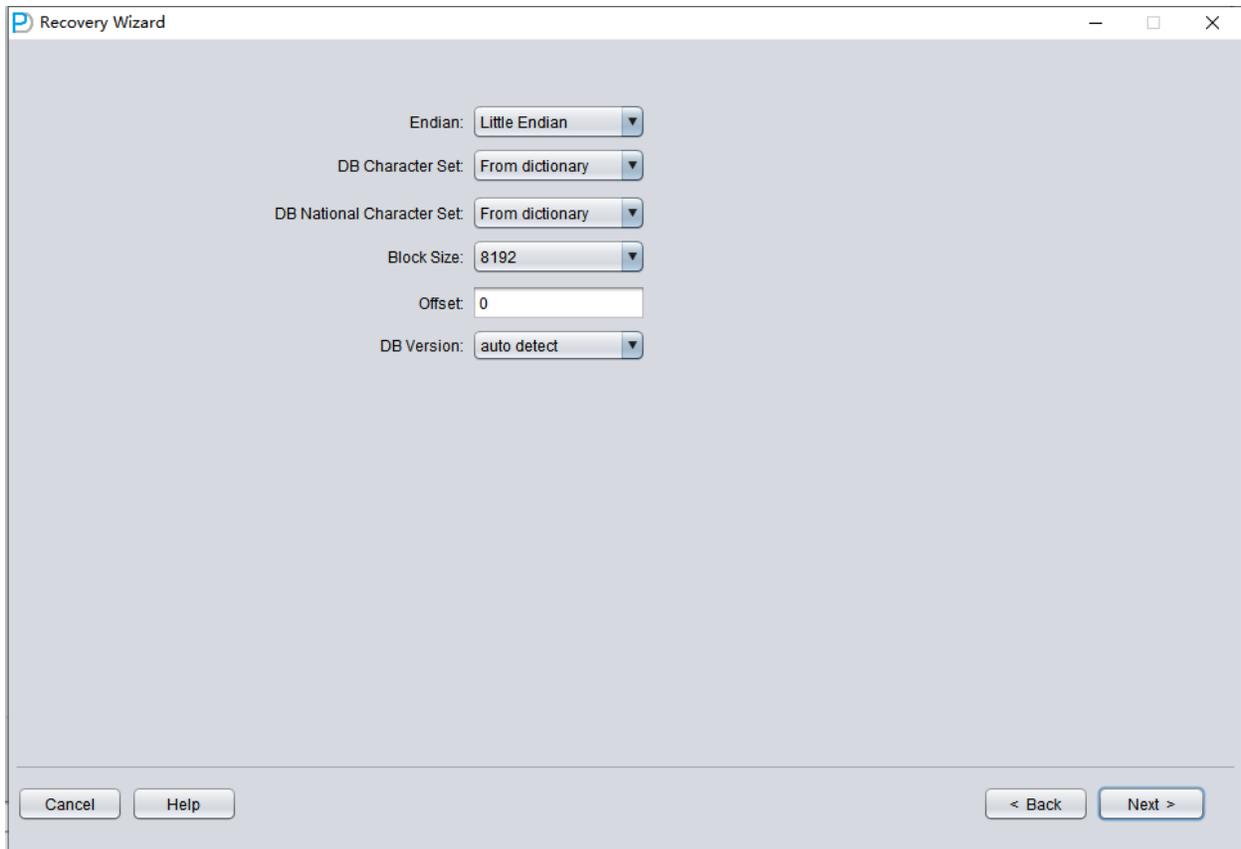
私たちが最もよく使用するWindowsとLinuxプラットフォームはLittle Endianであり、何も設定せずデフォルトのままです。

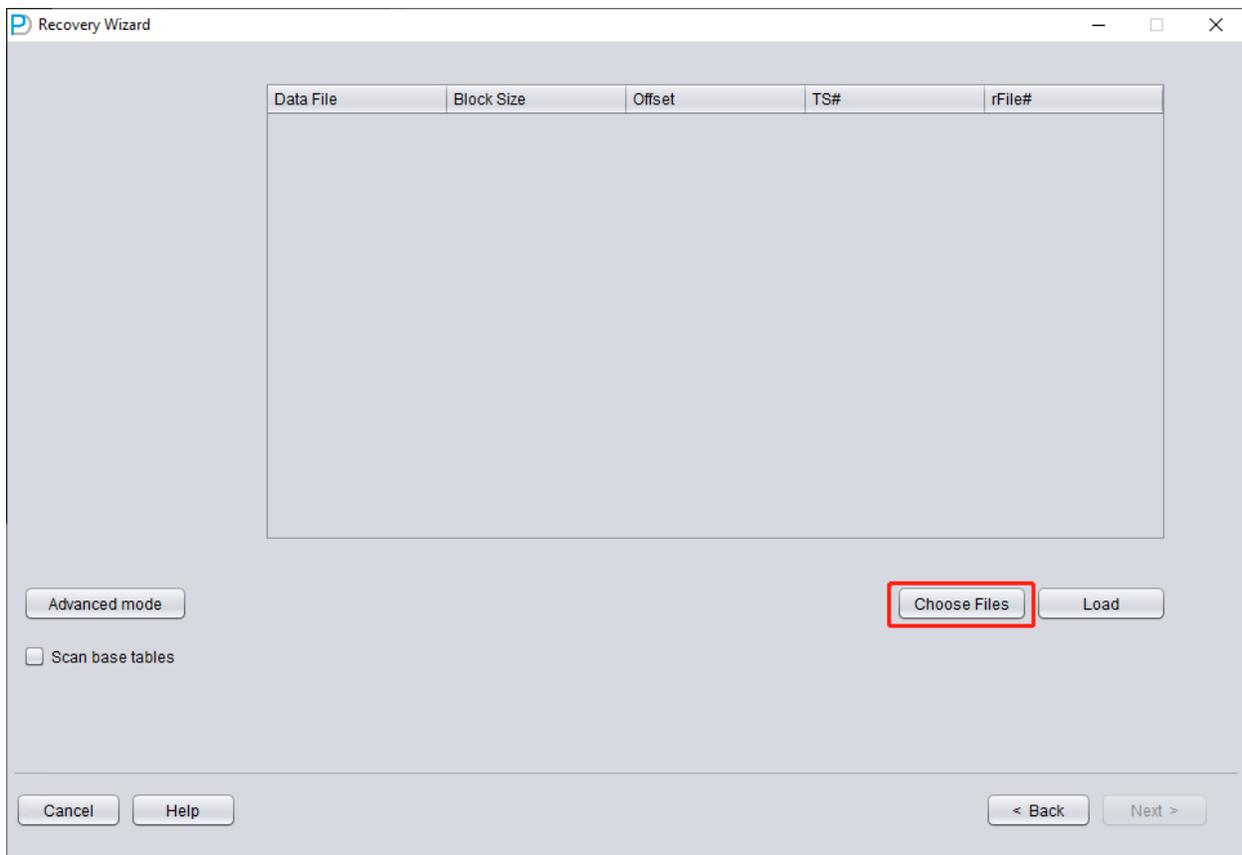
しかし、AIX-Based Systems (64-bit)やHP-UX (64-bit)などの小型機プラットフォームではBig Endianを使用していますので、ここではBig Endianを選択する必要があります。

注意点：データファイルがAIX（つまりBig Endian）で生成された場合、便宜上これらのファイルをWindowsサーバーにコピーしDBRECOVERでデータを復旧する場合でも、元のBig Endianフォーマットを選択する必要があります。

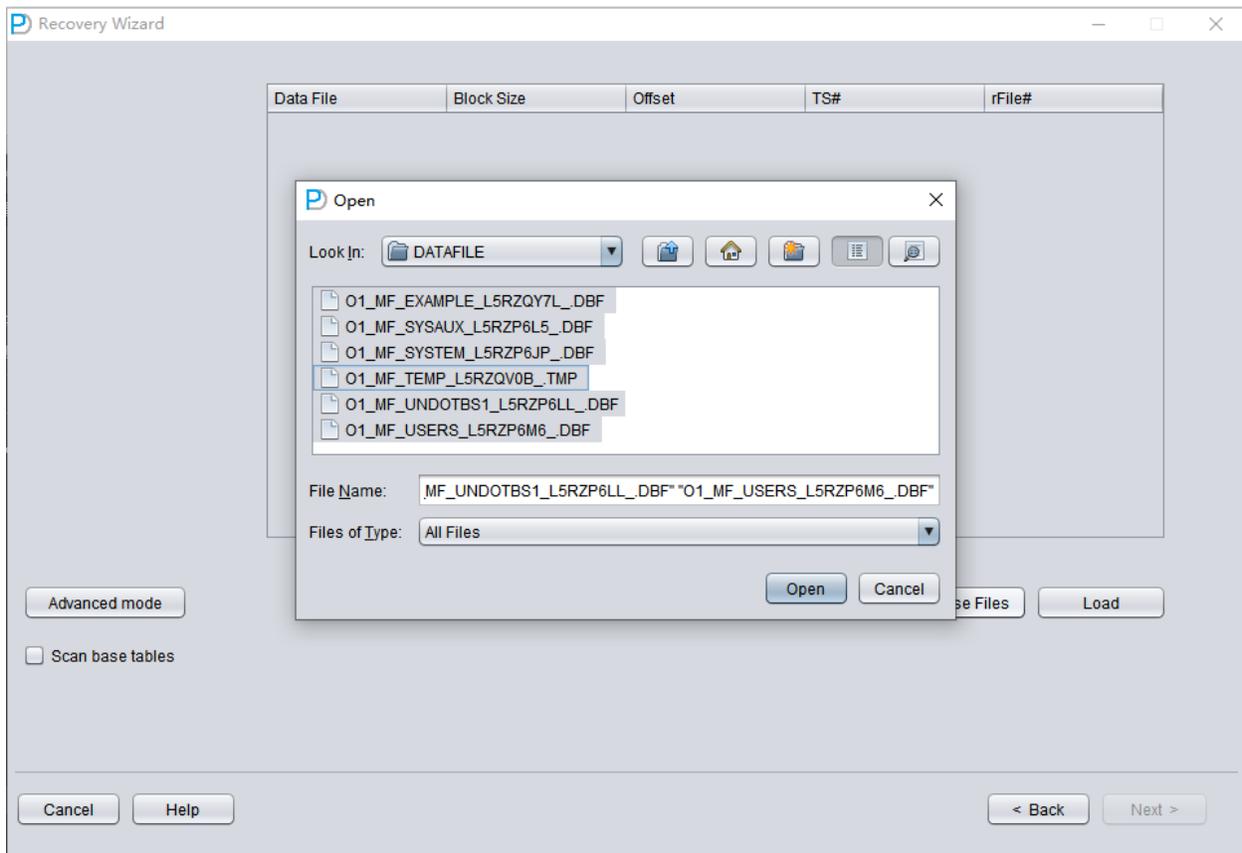
ここではLinux x86-64プラットフォーム上のOracleデータベースファイルを復旧するため、EndianはLittleを選択します。

「Next」をクリックします。

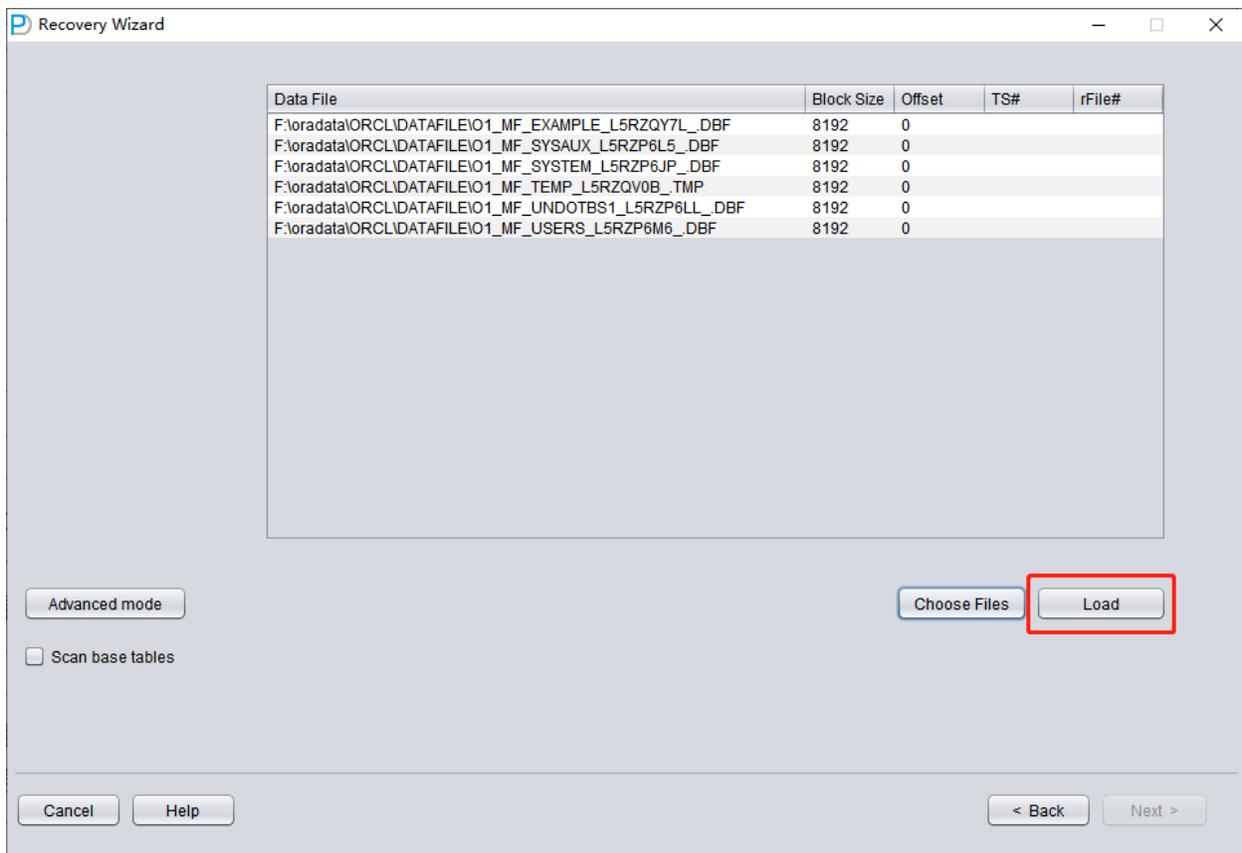




「Choose Files」をクリックし、データベースが大きくなければすべてのデータファイルを選択します。データベースが大きい場合や、データがどのデータファイルにあるか分かっている場合は、SYSTEM表領域（必須）とデータが存在するデータファイルのみを選択できます。



「Choose」画面ではCtrl + AやShiftなどのキーボード操作がサポートされています。



他のパラメーターについて不明な場合は、デフォルトのままにしておきます。

データファイルのBlock Size（ORACLEのデータブロックサイズ）を指定する必要があります。例えば、DB_BLOCK_SIZEが8Kだが一部の表領域で16Kのデータブロックサイズを指定している場合、8KでないデータファイルのみBLOCK_SIZEを変更します。

通常のファイルシステムを使用している場合、この段階でOFFSETを指定する必要はありません。OFFSETパラメータは、例えばAIX上の普通VGのLVをデータファイルとして使用している場合など、裸デバイスでデータファイルを保持しているシナリオに必要です。OFFSETが4Kの場合はここで指定します。

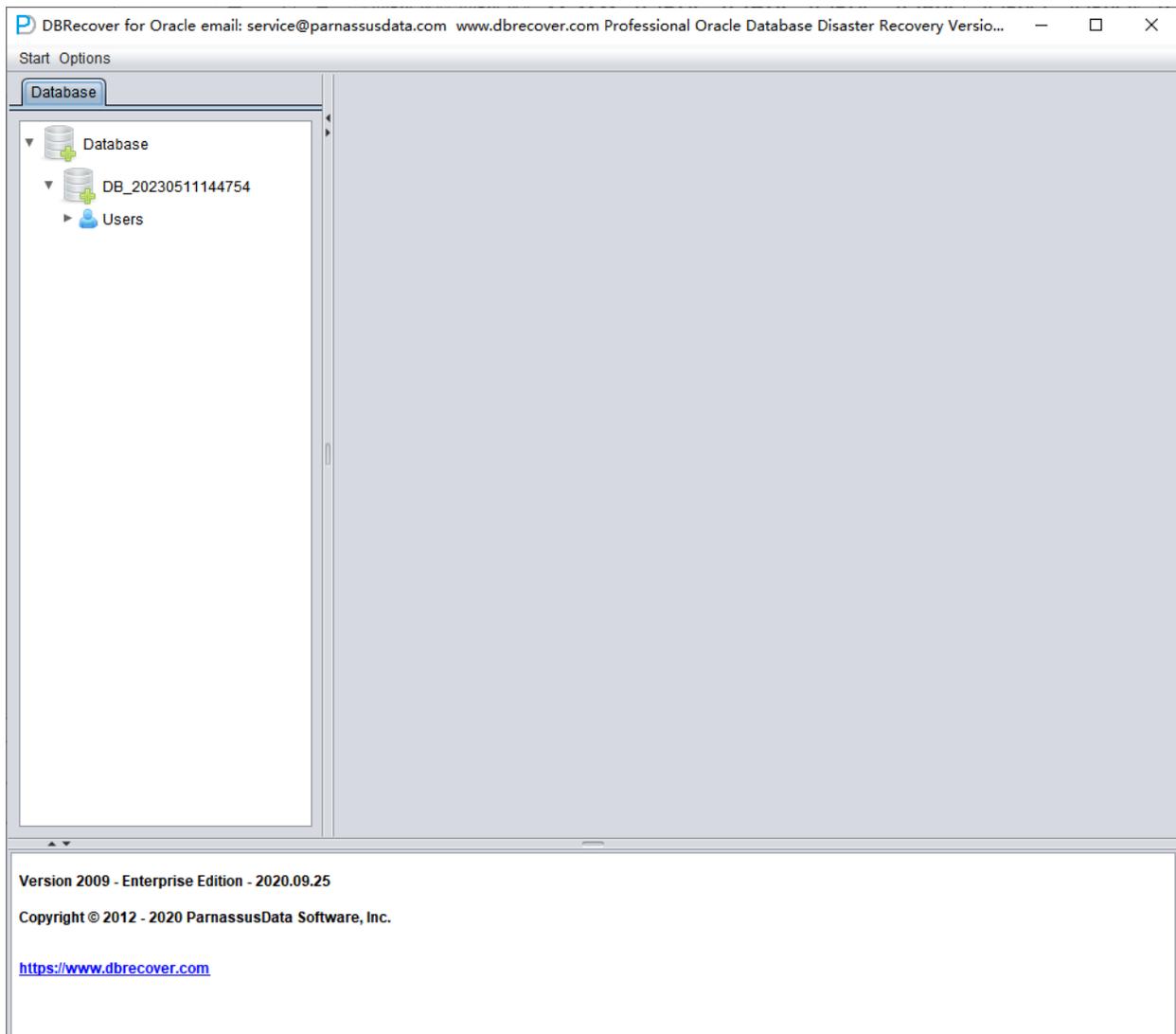
裸デバイスのデータファイルを使用していてOFFSETが不明な場合は、\$ORACLE_HOME/binに含まれるdbfsizeツールを使用して確認できます。

```
$ dbfsize /dev/lv_control_01
Database file: /dev/lv_control_01
Database file type: raw device without 4K starting offset
Database file size: 334 16384 byte blocks
```

このシナリオでは、すべてのデータファイルが8KのBLOCK SIZEであり、ファイルシステムに基づいているためOFFSETはありません。「Load」をクリックします。

Loadの段階でDBRECOVERはSYSTEM表領域からOracleのデータ辞書情報を読み取り、内蔵のDerbyに独自のデータ辞書を作成します。これにより、Oracleデータベース内の様々なデータを分析する能力がDBRECOVERに与えられます。

Loadが完了すると、DBRECOVERのインターフェースの左側に、データベースユーザーによってグループ化されたツリー構造が表示されます。



復旧したいテーブルを選択し、データを確認するためにダブルクリックします。

DBRecover for Oracle email: service@parnassusdata.com www.dbrecover.com Professional Oracle Database Disaster Recovery Versio... Start Options

Database

DB_20230511144754

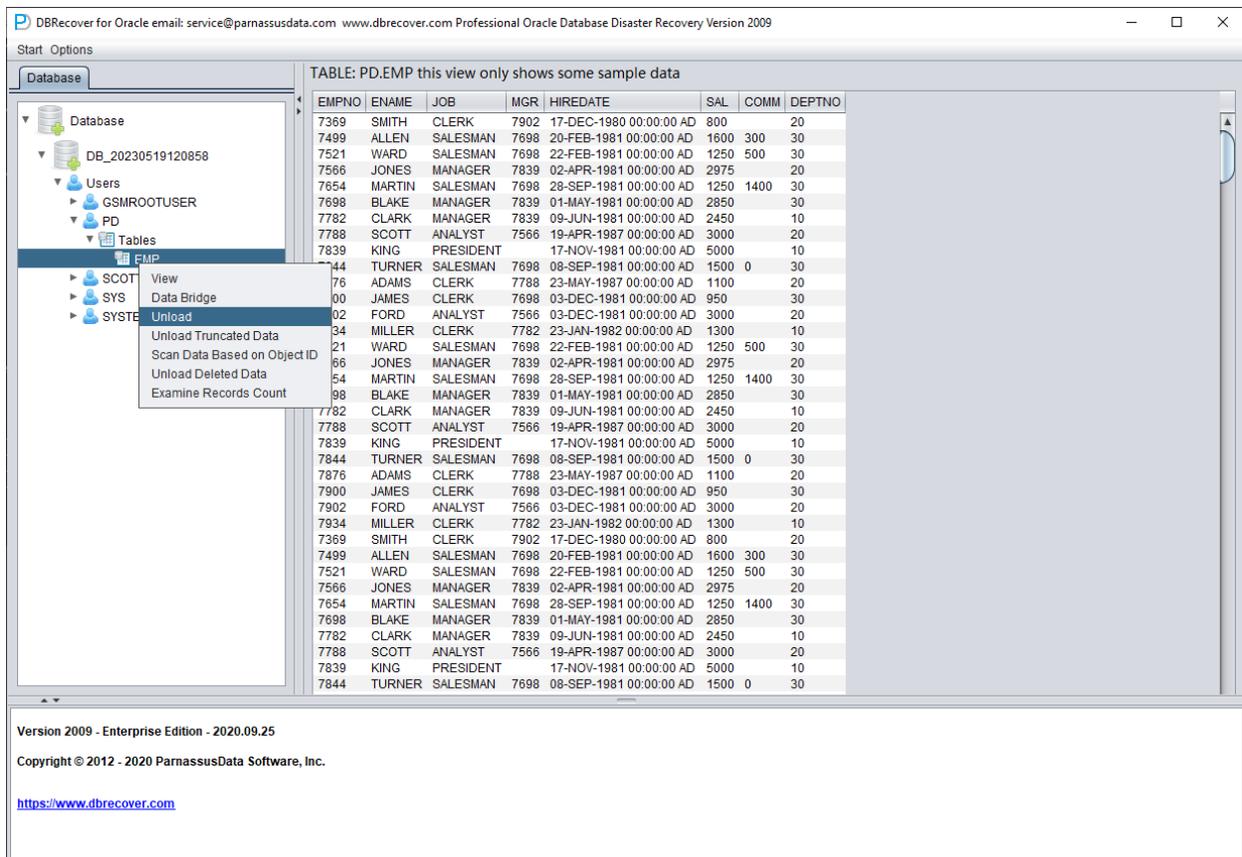
- Users
 - BI
 - HR
 - IX
 - OE
 - PD
 - Tables
 - EMP
 - PM
 - SCOTT
 - SH
 - SYS
 - SYSTEM

TABLE: PD.EMP this view only shows some sample data

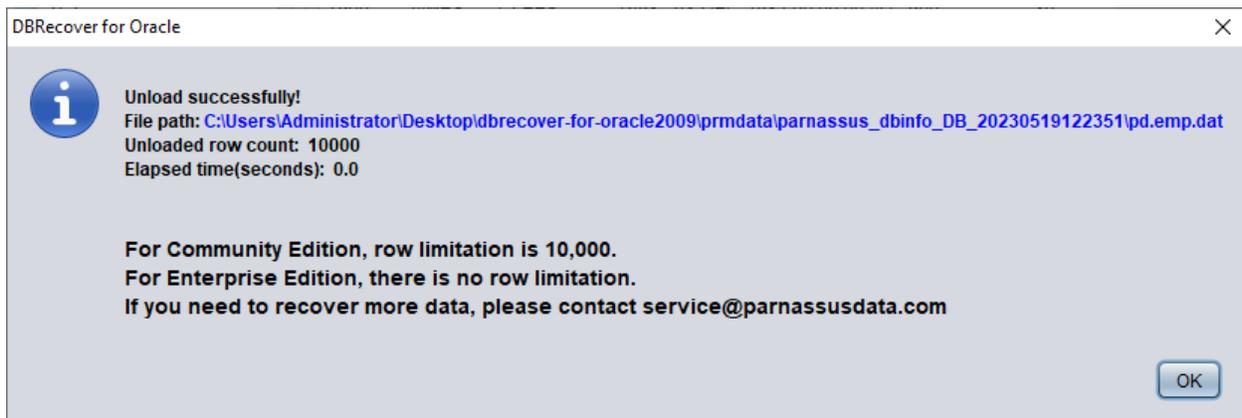
EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-1987 00:00:00 AD	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981 00:00:00 AD	950		30
7902	FORD	ANALYST	7566	03-DEC-1981 00:00:00 AD	3000		20
7934	MILLER	CLERK	7782	23-JAN-1982 00:00:00 AD	1300		10

Version 2009 - Enterprise Edition - 2020.09.25
 Copyright © 2012 - 2020 ParnassusData Software, Inc.
<https://www.dbrecover.com>

ソフトウェアのライセンスを購入する前に、データテーブルの確認、少なくとも1万行のデータ抽出、復旧可能な行数の検証を通じて、DBRECOVERが十分な量のデータを復旧できるかどうかを評価できます。



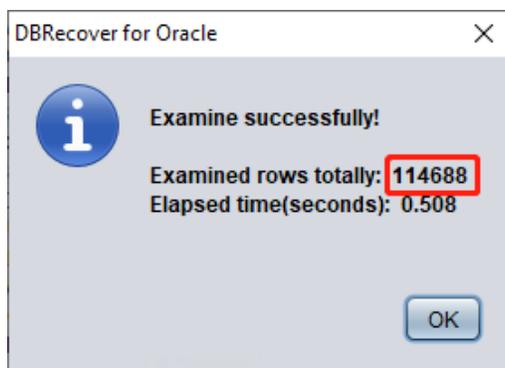
テーブルを選択した後、右クリックして「UNLOAD」を選択すると、テーブルデータがテキスト形式でエクスポートされます。



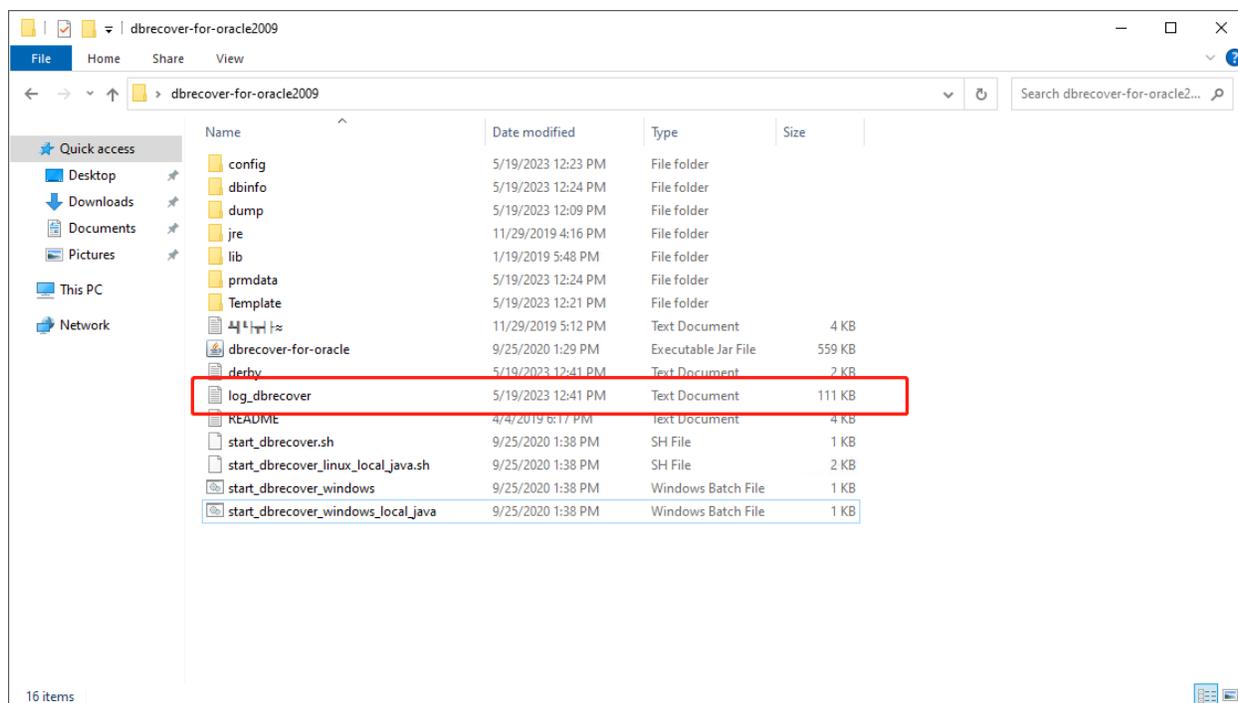
ソフトウェアライセンスを未登録の場合、単一のテーブルから最大1万行のデータを抽出できます。1万行を超えるデータを含むテーブルの場合は、回復可能な行数を検証するために「EXAMINE RECORDS COUNT」を選択して確認します。

The screenshot shows the DBRecover for Oracle Professional interface. The main window displays a table titled 'TABLE: PD.EMP this view only shows some sample data'. The table has columns: EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, and DEPTNO. A context menu is open over the 'EMP' table in the left-hand tree view, with 'Examine Records Count' selected. The bottom of the window shows the version information: 'Version 2009 - Community Edition - 2020.09.25' and 'Copyright © 2012 - 2020 ParnassusData Software, Inc.' with a link to <https://www.dbrecover.com>.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7839	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30
7844	ADAMS	CLERK	7788	23-MAY-1987 00:00:00 AD	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981 00:00:00 AD	950		30
7902	FORD	ANALYST	7566	03-DEC-1981 00:00:00 AD	3000		20
7934	MILLER	CLERK	7782	23-JAN-1982 00:00:00 AD	1300		10
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30



Oracle 10g以降では自動的に統計情報を収集する機能が導入されており、これを利用してテーブルの履歴統計情報、特に行数を確認することができます。字典モードでテーブルの確認、抽出、検査などの操作を行うと、その情報はソフトウェアのログ（log_dbrecover.txt）に記録されます。このログファイルはソフトウェアのディレクトリにあります。



```

log_dbrecover - Notepad
File Edit Format View Help
TABLE SYS.TYPE$ 5889 rows unloaded
TABLE SYS.COLLECTION$ 1385 rows unloaded
TABLE SYS.ATTRIBUTE$ 15376 rows unloaded
TABLE SYS.LOBFRAGS 25 rows unloaded
TABLE SYS.LOBCOMPPART$ 0 rows unloaded
TABLE SYS.TS$ 6 rows unloaded
Warning can be ignored: insert prm_tables_collection rows number is 2206
Warning can be ignored: delete SYS_NC000$ & SYS_C00 & Virtual Column for col$ rows number is 1178
Warning can be ignored: delete SYS_STU SYS_ST5 Column for col$ rows number is 0
Warning can be ignored: delete BIN$ recyclebin object for obj$ rows number is 0
created view pd_tab_col
the manual path for tabpart$ is ./manual/sys.tabpart$.dat
the manual load tabpart$.dat not exists, using default :./prmdata/parnassus_dbinfo_DB_20230519125028/./sys.tabpart$.dat
the manual path for tabsubpart$ is ./manual/sys.tabsubpart$.dat
the manual load tabsubpart$.dat not exists, using default :./prmdata/parnassus_dbinfo_DB_20230519125028/./sys.tabsubpart$.dat
the manual path for lob$ is ./manual/sys.lob$.dat
the manual load lob$.dat not exists, using default :./prmdata/parnassus_dbinfo_DB_20230519125028/./sys.lob$.dat
the manual path for ind$ is ./manual/sys.ind$.dat
the manual load ind$.dat not exists, using default :./prmdata/parnassus_dbinfo_DB_20230519125028/./sys.ind$.dat
the manual path for lobfrag$ is ./manual/sys.lobfrag$.dat
the manual load lobfrag$.dat not exists, using default :./prmdata/parnassus_dbinfo_DB_20230519125028/./sys.lobfrag$.dat
Use default path to load sys.indpart$.dat
Use default path to load sys.indsubpart$.dat
Database character set is AL32UTF8
Database national character set is AL16UTF16
Current character set for decoding is UTF8
Current national character set for decoding is UTF16

For Community Edition, row limitation is 10,000.
If you need to recover more data, please contact service@parnassusdata.com

object information user#:106 object_name: EMP object_id:74042 data_object_id:74042 object_type:2
table information object_id:74042 data_object_id:74042 ts#:4 rfile#:7 block#:386 rowcnt:114688 blkcnt:751 analyzetime:2023-05-19 12:41:29.0
TABLE PD.EMP 666 rows unloaded

```

日志情報には、DBRECOVERで行われた操作や、テーブルの行数、抽出プロセスの詳細などの情報が含まれています。

```

object information user#:106 object_name: EMP object_id:74042 data_object_id:74042 object_type:2
table information object_id:74042 data_object_id:74042 ts#:4 rfile#:7 block#:386 rowcnt:114688
blkcnt:751 analyzetime:2023-05-19 12:41:29.0
TABLE PD.EMP 666 rows unloaded

```

object_idオブジェクト番号	74042
data_object_idデータオブジェクト番号	74042
ts#表領域番号	4
rfile#表頭部の相対ファイル番号	7
block#表頭部のデータブロック番号	386
rowcnt統計情報で記録された行数（統計情報は推定値）	114688
blkcntその表の総ブロック数	751
analyzetime統計情報の収集時間	2023-05-19 12:41:29.0

一般的に統計情報の誤差は10%を超えないため、ここでのrowcntとEXAMINEの結果を比較して行数の正確性を確認できます。rowcntが114688（100万行未満の表では統計情報の誤差は非常に

小さい) で、EXAMINEの結果も114688行であれば、その結果の正確性が検証できます。

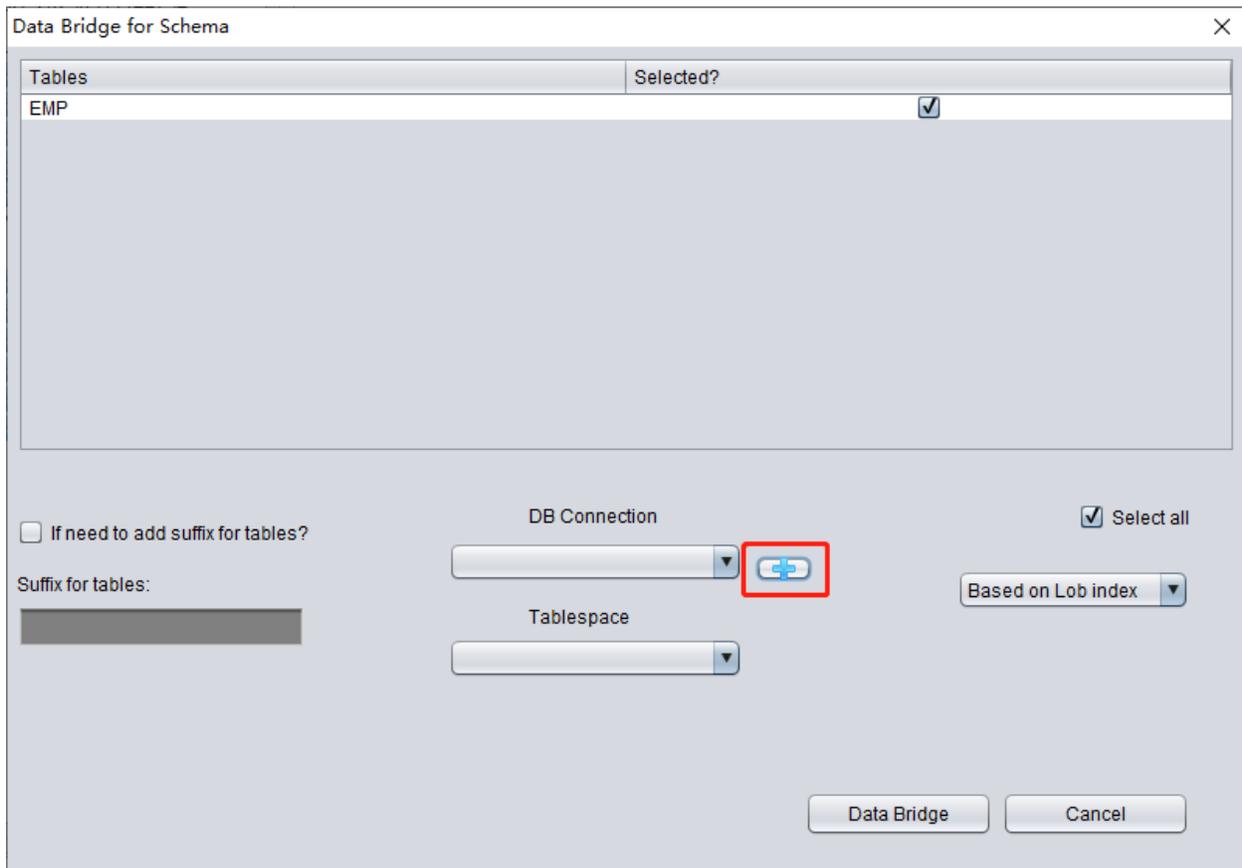
ユーザーは自身のニーズに基づいて、重要な各データテーブルに対して上記の検証を行うことができます。ソフトウェアライセンスを購入する前に、回復可能なデータの完全性を十分に検証することをお勧めします。

検証が完了したら、SCHEMAユーザーレベルでのデータブリッジ転送を開始します。復旧したいユーザー名を選択し、右クリックで「Data Bridge」を選択します。

The screenshot shows the DBRecover for Oracle Professional interface. The title bar indicates the version is 2009. The main window is divided into a left sidebar and a main content area. The sidebar shows a tree view of the database structure, with 'PD' selected under the 'Users' folder. The main content area displays a table titled 'TABLE: PD.EMP this view only shows some sample data'. The table has columns for EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, and DEPTNO. The data is sorted by EMPNO and shows a list of employees with their details. At the bottom of the window, there is a footer with version information and a copyright notice.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-1987 00:00:00 AD	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981 00:00:00 AD	950		30
7902	FORD	ANALYST	7566	03-DEC-1981 00:00:00 AD	3000		20
7934	MILLER	CLERK	7782	23-JAN-1982 00:00:00 AD	1300		10
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981 00:00:00 AD	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981 00:00:00 AD	2450		10
7788	SCOTT	ANALYST	7566	19-APR-1987 00:00:00 AD	3000		20
7839	KING	PRESIDENT		17-NOV-1981 00:00:00 AD	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-1981 00:00:00 AD	1500	0	30

SCHEMAレベルのデータブリッジ画面で「+」ボタンをクリックし、ターゲットデータベースの接続情報を追加します。



ここではPDユーザーの接続情報を入力します。

DBRECOVERは、指定されたデータベース接続情報のユーザーにデータを転送します。ターゲットデータベースに同じユーザーと表領域を作成し、必要な権限（DBAロール）を付与します。複数のデータベースユーザーを復旧する場合は、それぞれに対応するアカウントをターゲットデータベースに作成し、DBRECOVERで複数のデータベース接続情報を設定します。

「TEST」ボタンでターゲットデータベース接続の可用性をテストします。

The screenshot shows the 'New Database Connection' dialog box. On the left, there is a table with two columns: 'Connection Name' and 'Connecti'. The main area contains the following fields and options:

- Connection Name: orcl1-conn
- Username: pd
- Password: *****
- Hostname: localhost
- Port: 1521
- Radio buttons: SID, Service name
- Service name: ORCL1

At the bottom, there are three buttons: 'Save', 'Test' (highlighted with a red rectangle), and 'Cancel'.

This screenshot shows the same 'New Database Connection' dialog box as above, but with a success message overlay. The message box is titled 'DBRecover for Oracle' and contains the text 'Connect to db server successfully!' with an 'OK' button.

The background dialog box shows the same fields as in the previous image, but the 'Test' button is no longer highlighted.

接続が成功したら、「SAVE」をクリックして保存します。

New Database Connection

Connection Name	Connecti
Connection Name	orcl1-conn
Username	pd
Password	*****
Hostname	localhost
Port	1521
<input type="radio"/> SID	
<input checked="" type="radio"/> Service name	ORCL1

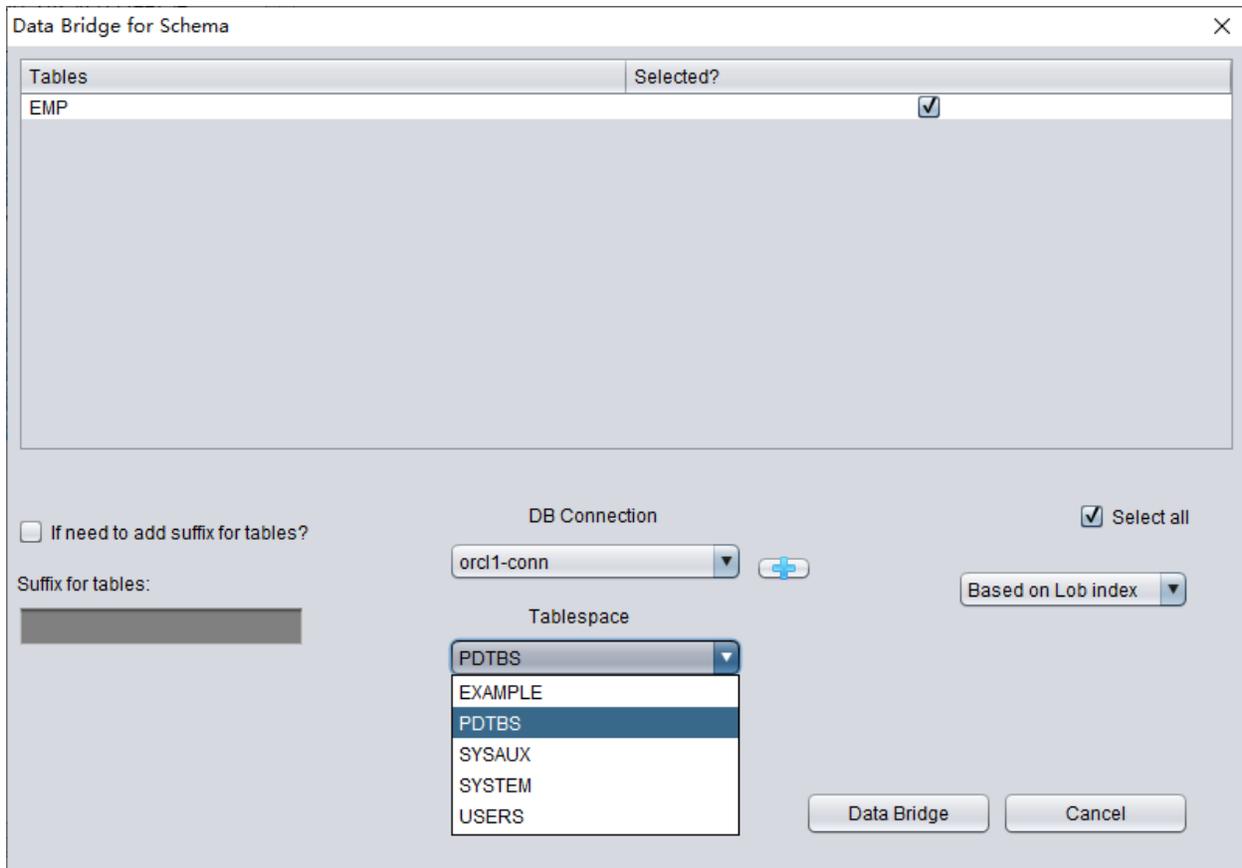
Data Bridge for Schema

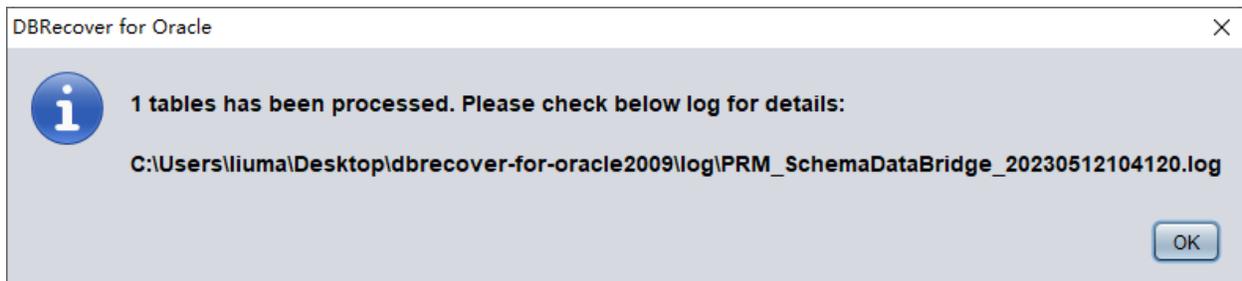
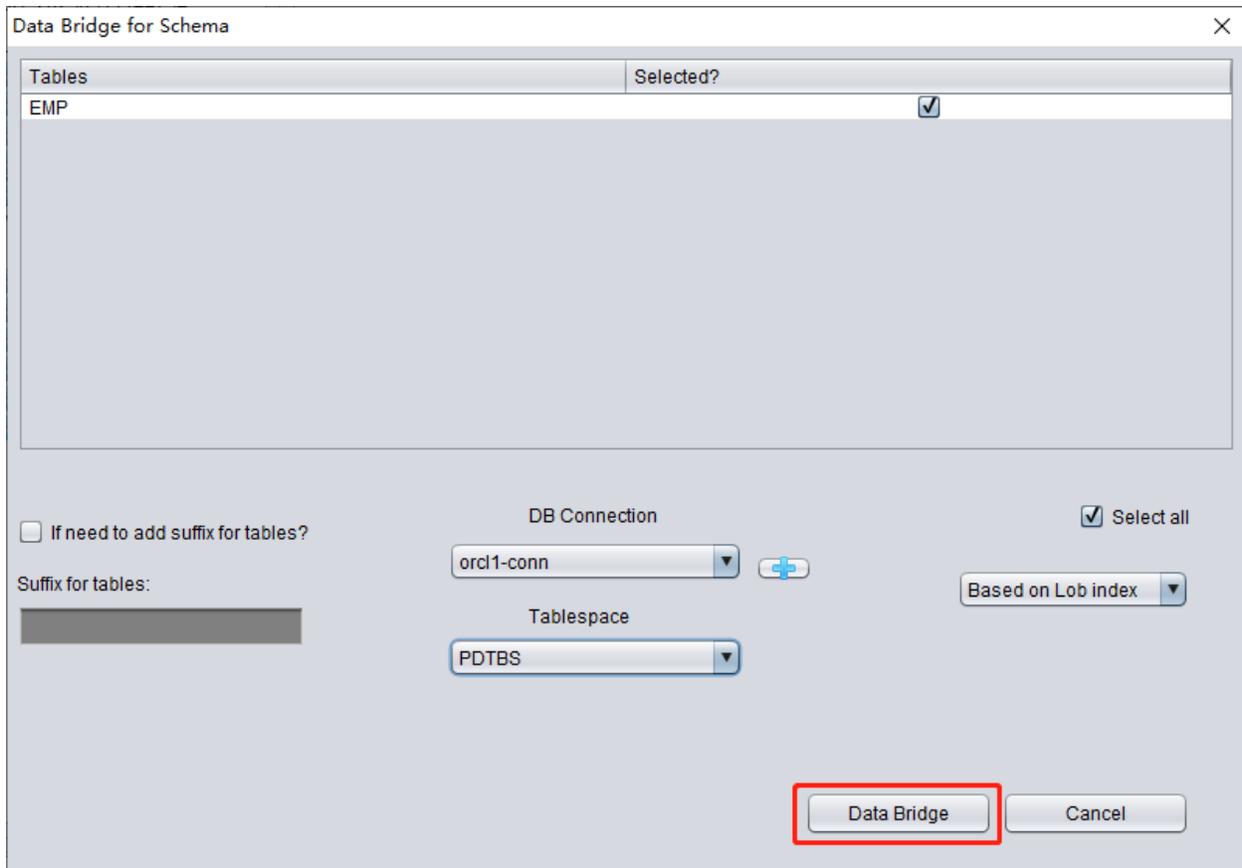
Tables	Selected?
EMP	<input checked="" type="checkbox"/>

If need to add suffix for tables?
 Suffix for tables:

DB Connection:

Select all



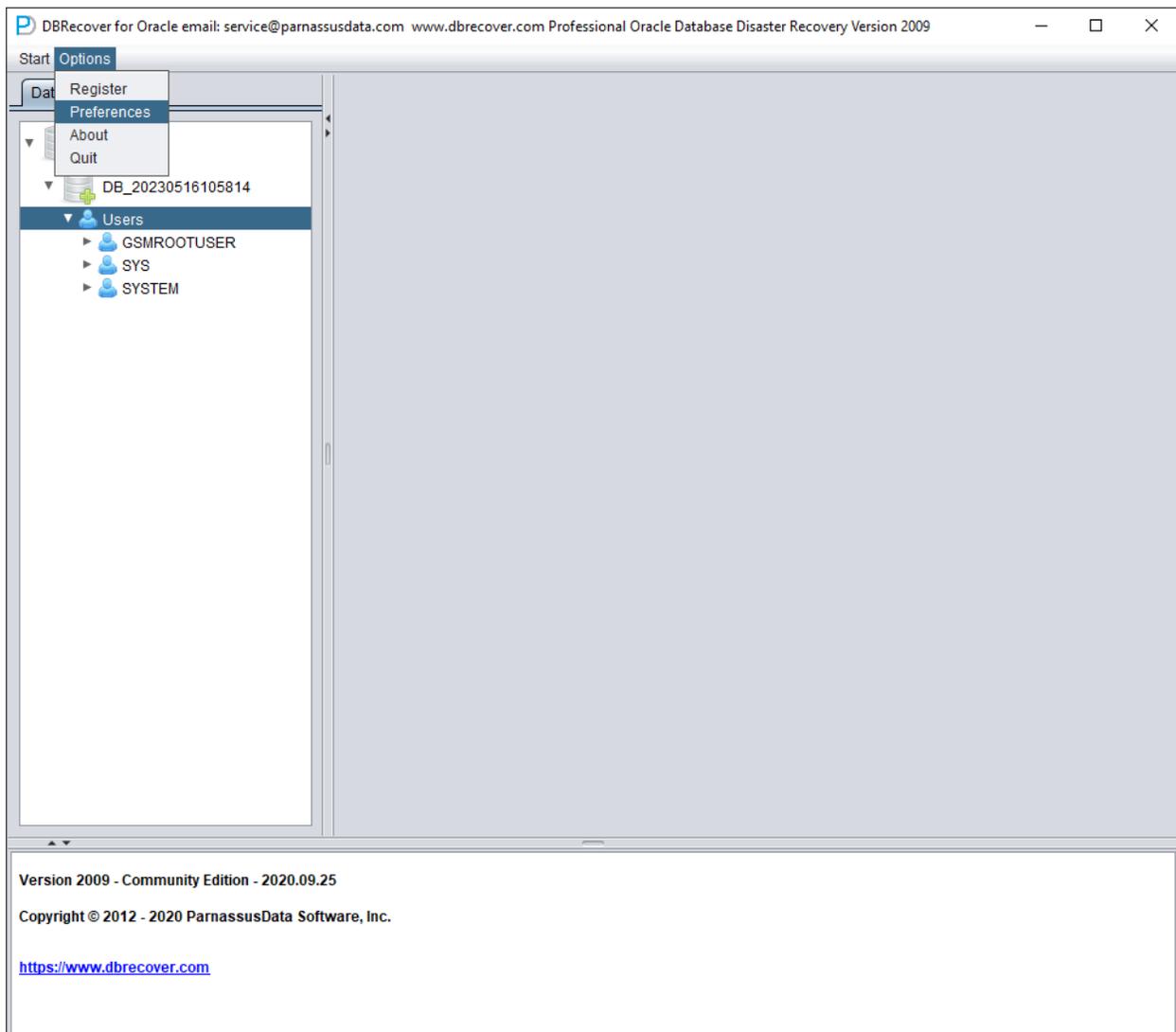


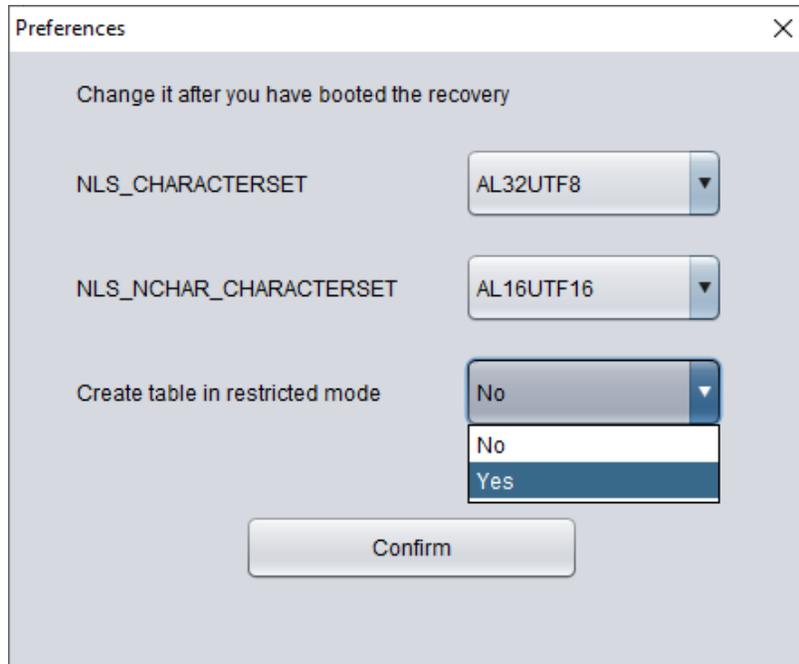
```
SQL> show parameter db_name
NAME TYPE VALUE
-----
db_name string ORCL1
SQL> select count(*) from pd.emp;
COUNT(*)
-----
```

WIDE TABLE 宽表模式介绍

デフォルトでは宽表模式 (wide table mode) を使用し、CHAR、NCHAR、VARCHAR、NVARCHARのフィールドタイプを最大長 (2000または4000) に変換します。これは、フィールドが短すぎて復旧文字列を挿入できない可能性を避けるためです。

宽表模式を使用しない場合は、「Options => Preferences」をクリックして設定を変更します。



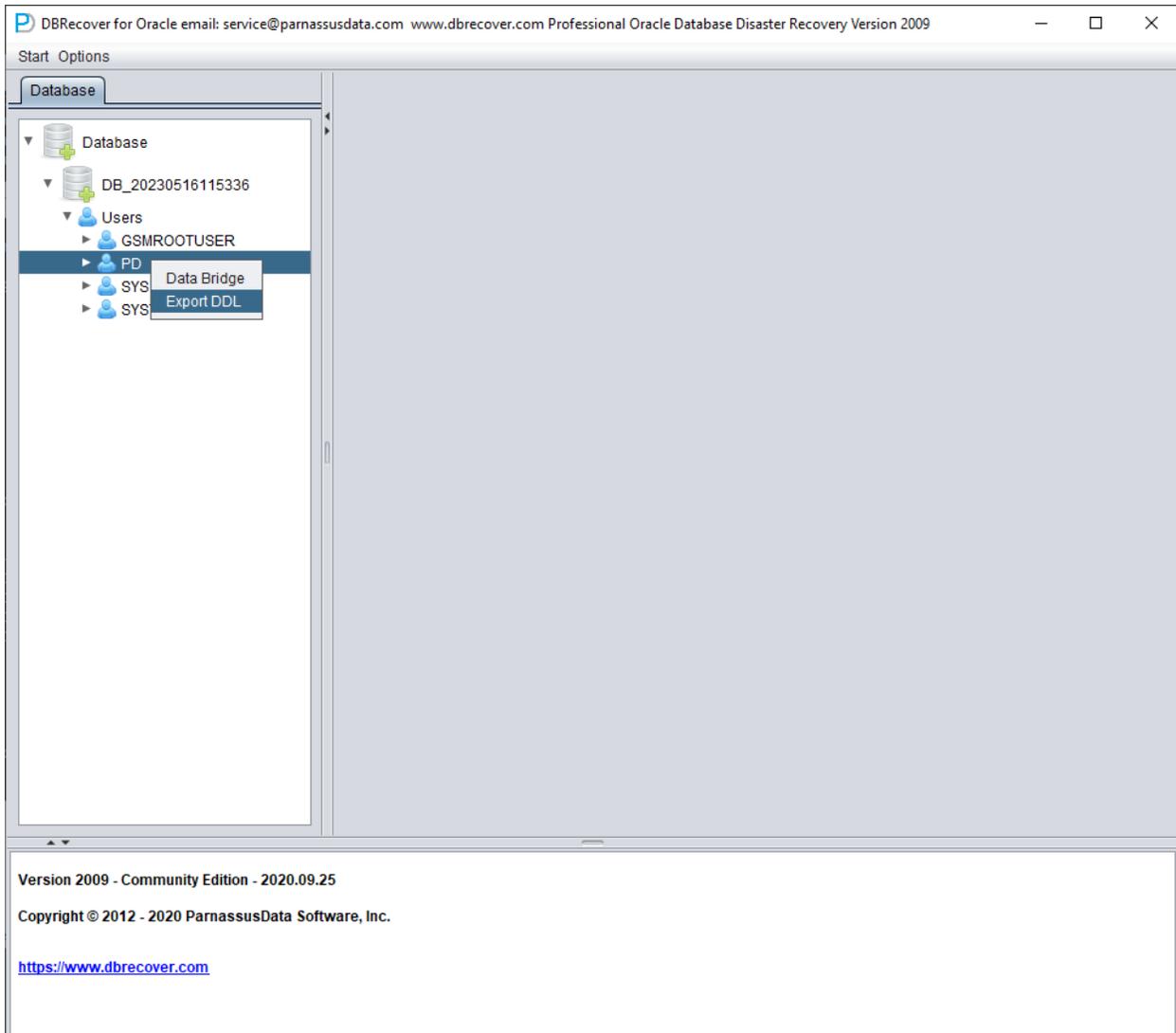


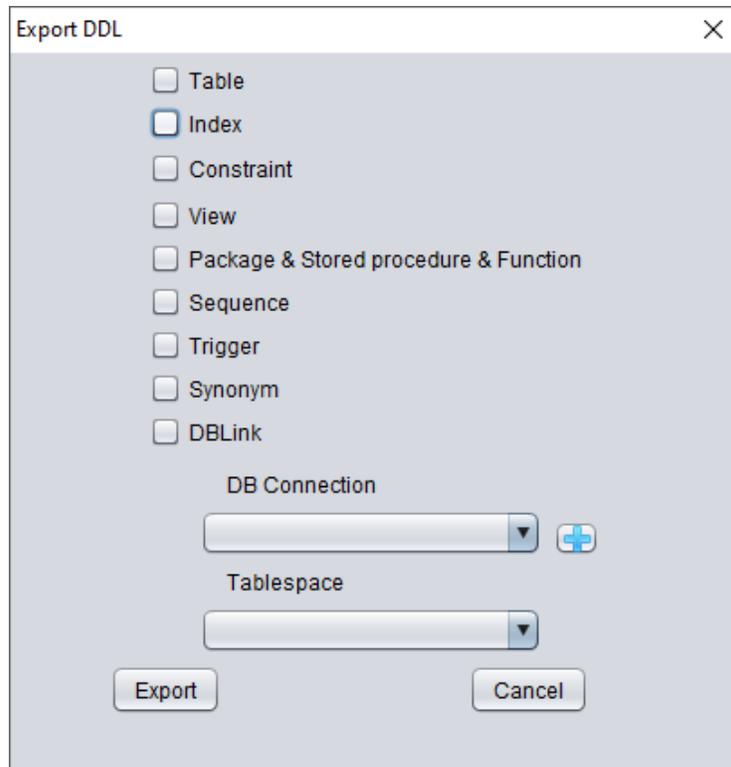
「Create table in restricted mode」のドロップダウンメニューで「Yes」を選択すると、寛表模式を使用せずにデータテーブルを作成します。

EXPORT DDL機能の紹介

単一のSCHEMAのデータテーブルの復旧を行い、対応するデータテーブルを作成し、復旧可能なデータを挿入しました。インデックス、制約、ビュー、トリガーなどのオブジェクトの復旧には、EXPORT DDL機能を使用します。

復旧したいSCHEMAを選択し、右クリックでEXPORT DDL機能を選択します。

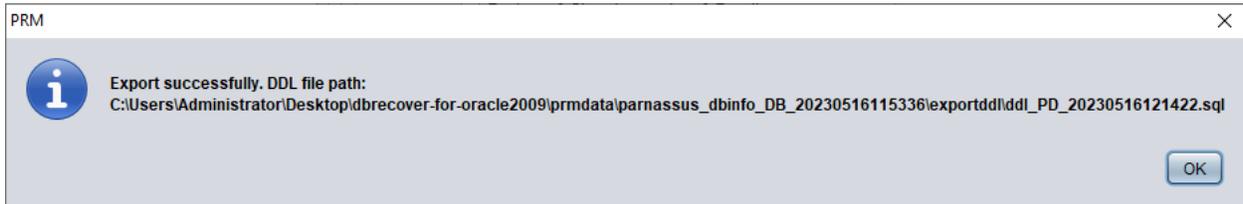
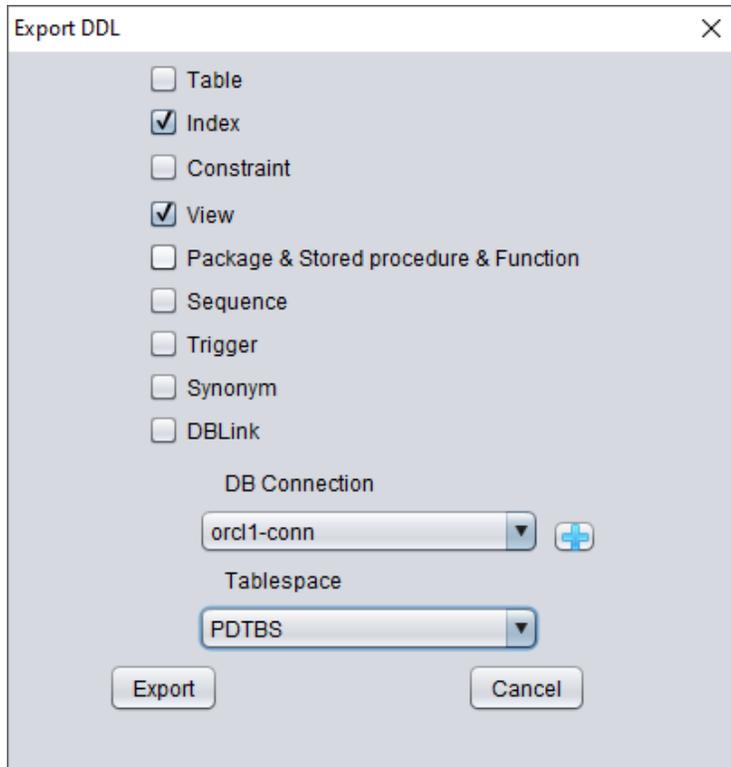




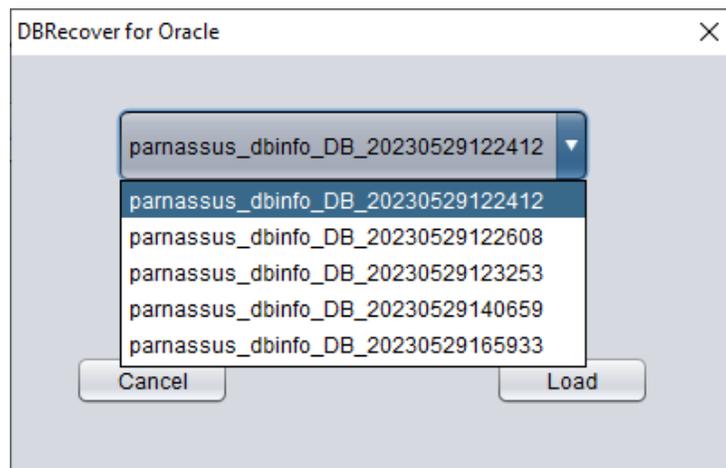
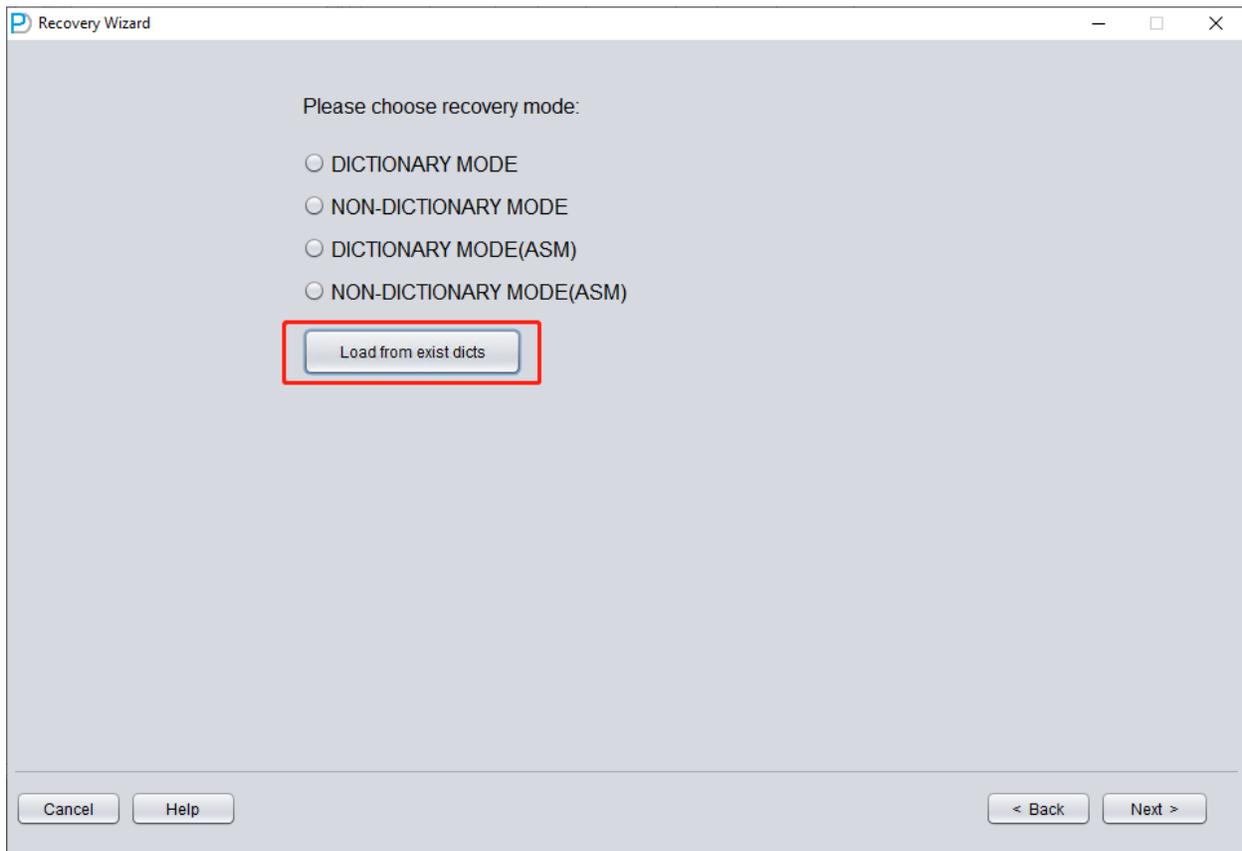
復旧可能なオブジェクトの種類には以下が含まれます：

- 建表语句（分区情報は含まない）
- 索引（分区情報は含まない）
- 约束
- 视图
- パッケージ、ストアドプロシージャ、関数
- シーケンス
- トリガー
- 同義語
- データベースリンク

ここでも以前入力したデータベース接続情報を選択し、DDL情報の一時処理に使用します。



弹出窗口提示了DDL SQL文件的路径，查看该文件：



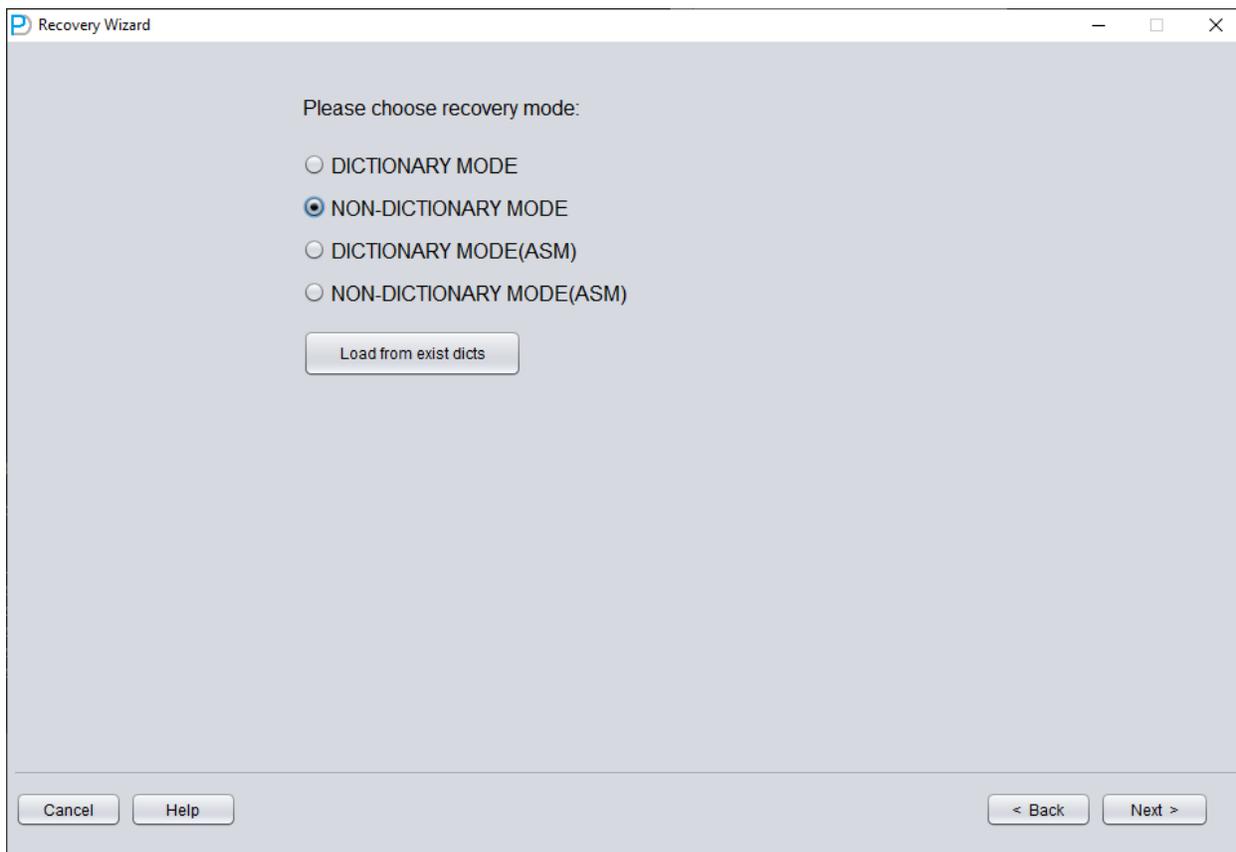
復旧状態は時間順に並んでおり、適切なものを選んで「LOAD」ボタンをクリックすると読み込みます。DBRECOVERの「字典モード (DICTIONARY-MODE)」と「非字典モード (NON-

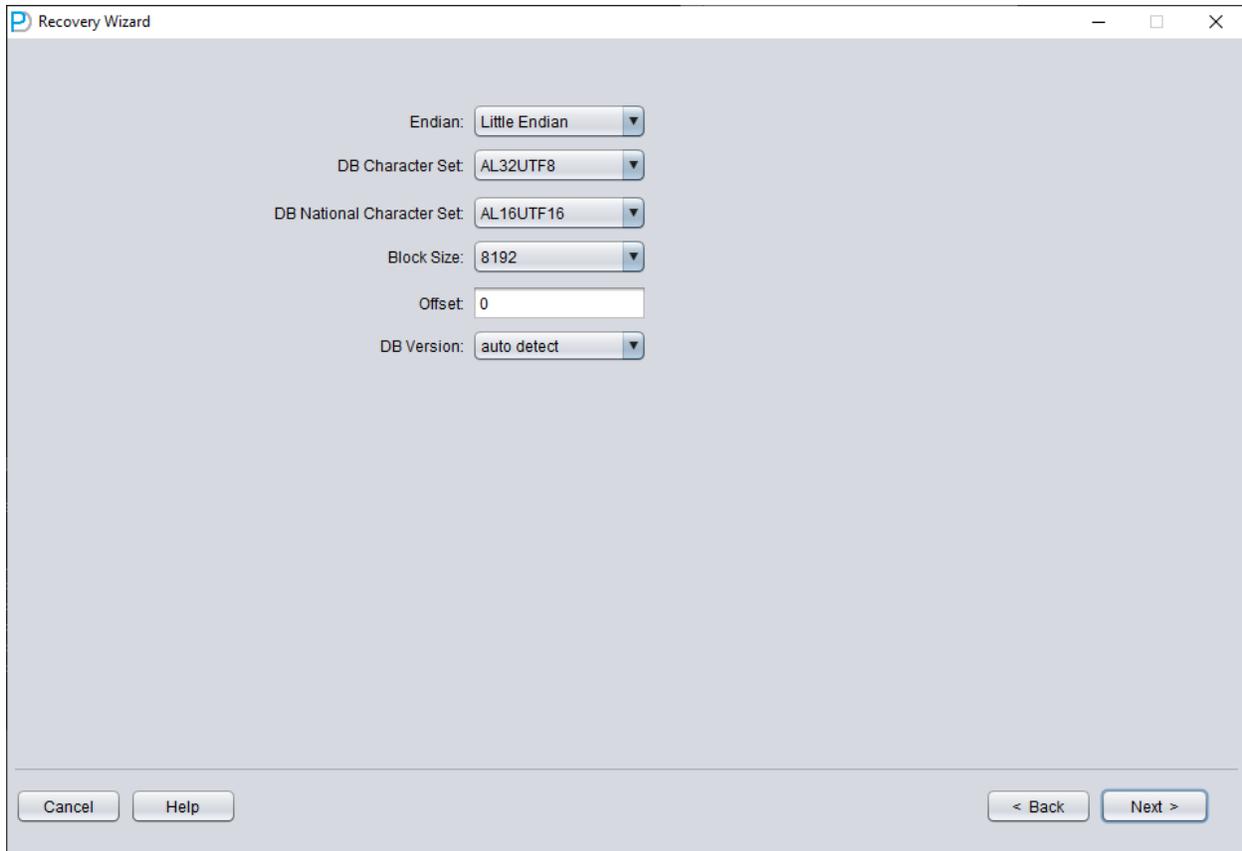
DICTIONARY MODE) 」の両方で、この迅速な読み込み機能を利用して、重複操作を避けることができます。

恢复场景2：SYSTEM表領域の誤削除または完全な紛失

D社のSAシステム管理者が誤ってあるデータベースのSYSTEM表領域のデータファイルを削除し、データベースが全く開かなくなりました。バックアップがない場合、DBRECOVERを使用してデータを掘り出すことができます。

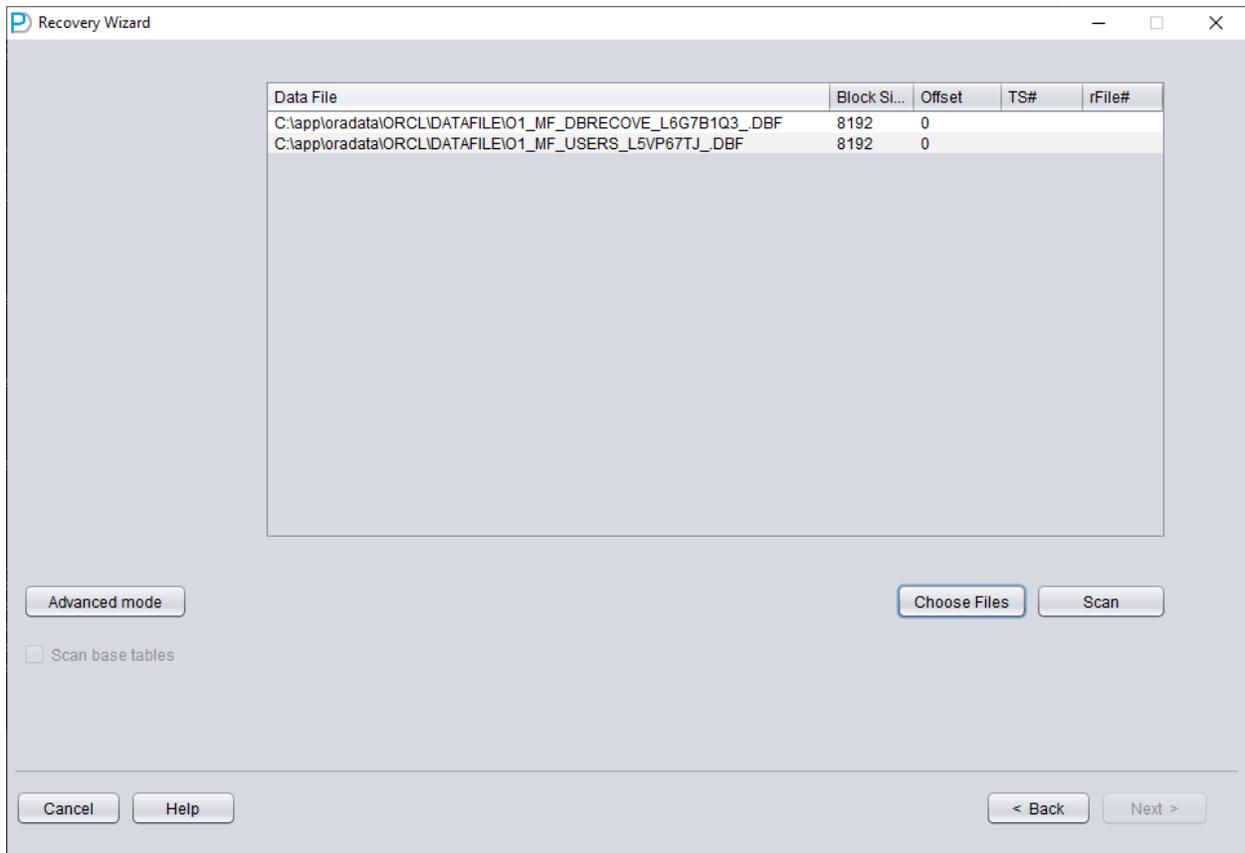
このシナリオでは、DBRECOVERを起動し、Recovery Wizardに入った後、「Non-Dictionary mode」非字典モードを選択します。



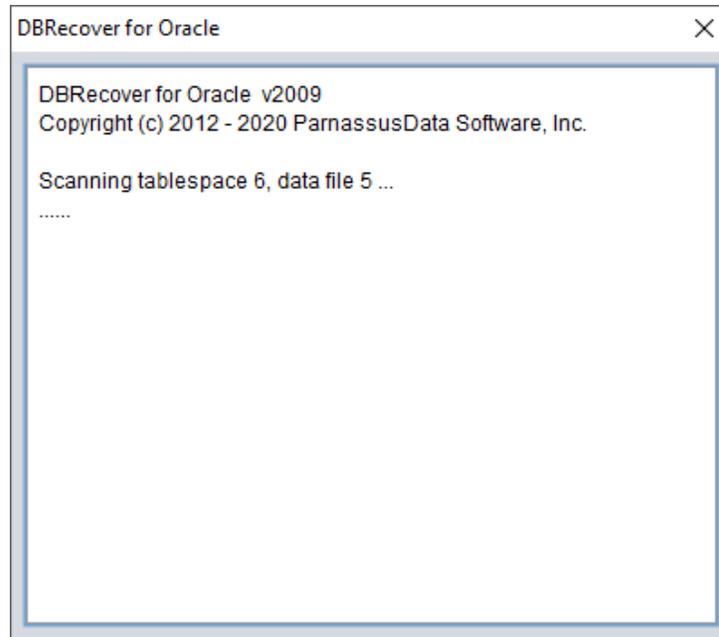


NoN-dictionaryモードでは、正しい文字セットと国家文字セットを指定する必要があります。これはSYSTEM表領域が失われたため、データベースの文字セット情報が正常に取得できないためです。適切な文字セットと必要な言語パックのインストールが、多言語データの正常な抽出を保証します。

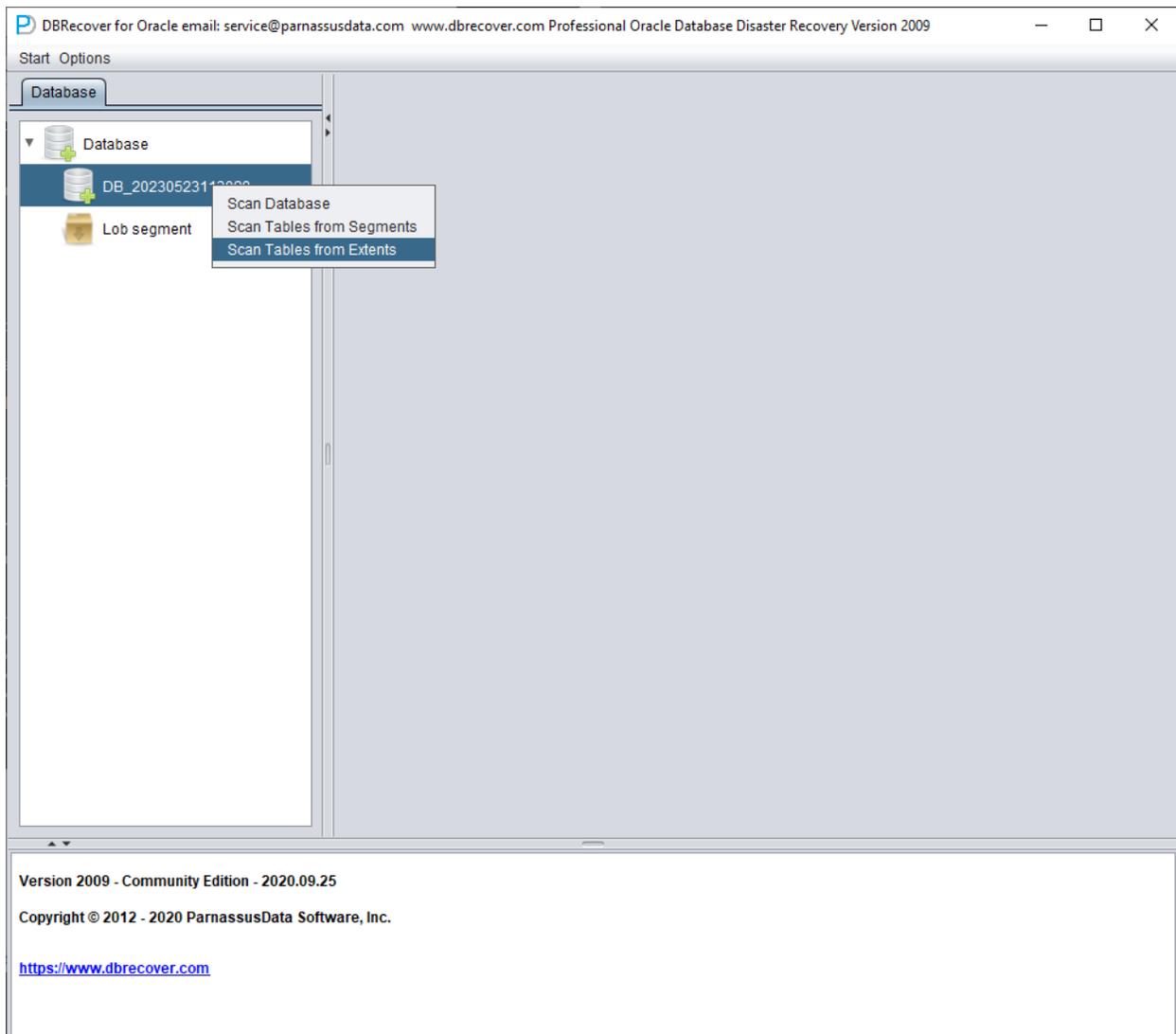
シナリオ1と同様に、ユーザーが利用可能なすべてのデータファイル（一時ファイルは除く）を入力し、正しいBlock SizeとOFFSETを設定します。



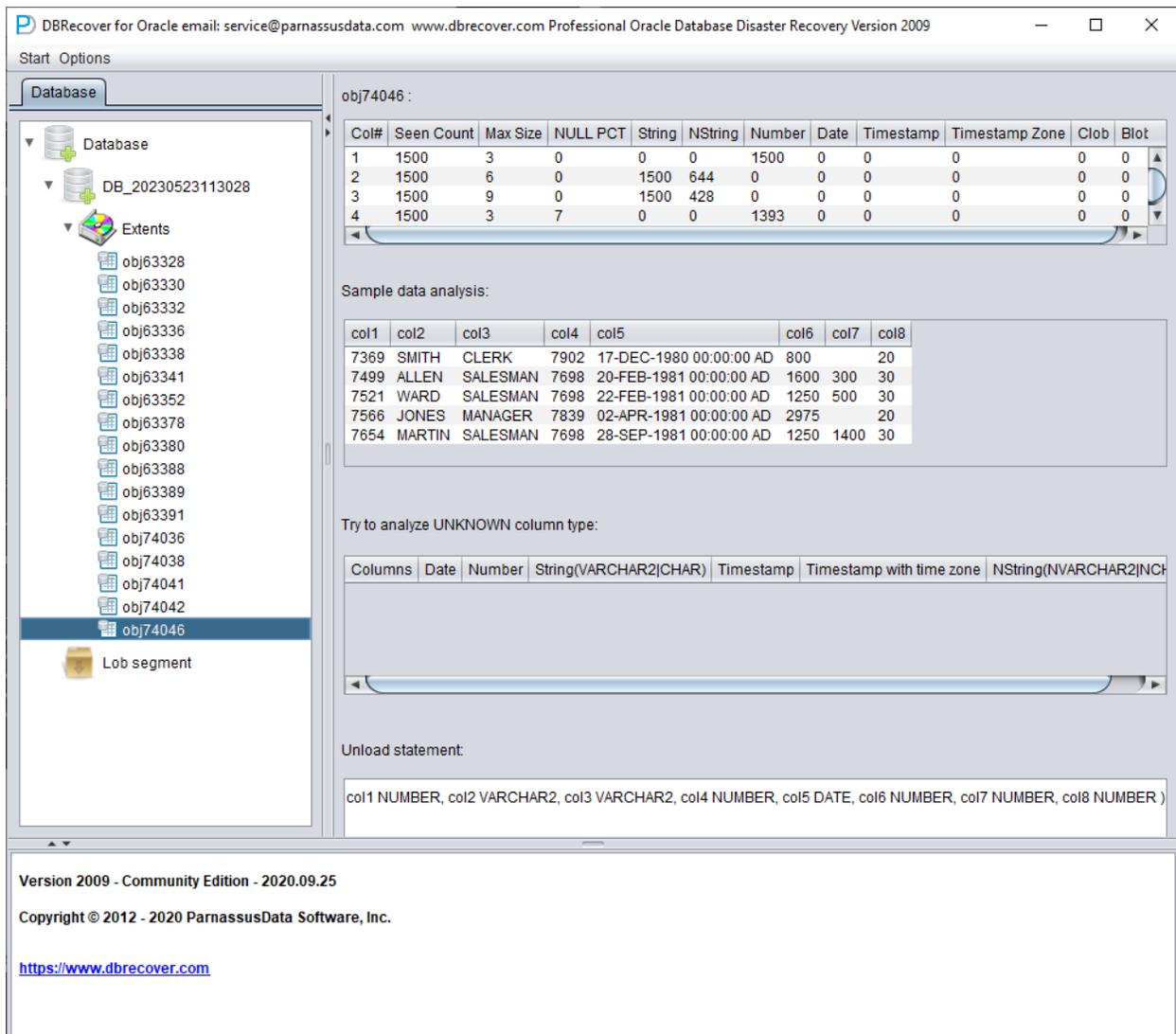
之后点击SCAN，SCAN的作用是扫描所有数据文件上数据信息。



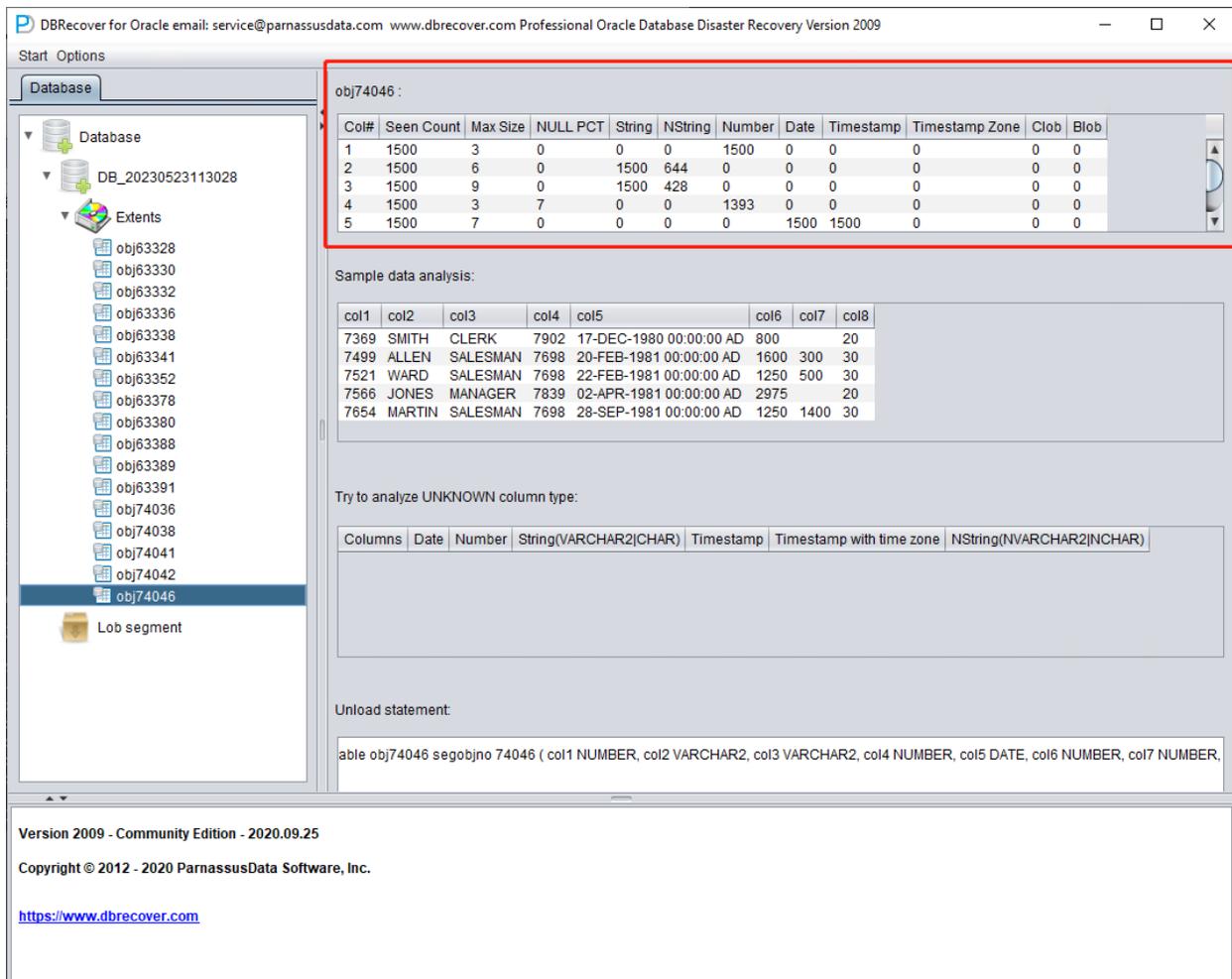
SCANをクリックして、すべてのデータファイル上のデータ情報をスキャンします。その後、左側のツリーのデータベースノードを右クリックしてSCAN EXTENTを実行します。すべてのデータファイル（SYSTEM01.DBFを除く）が利用可能な場合にのみ、「SCAN TABLE FROM SEGMENTS」モードを使用します。このモードの利点は速度がわずかに速いことですが、データファイルが不完全であるか損傷している場合、SCAN EXTENTモードに比べて復旧程度が低下します。



「Scan Tables From Extents」が完了すると、メイン画面の左側にあるツリーを確認できます。



ツリー上の各ノードは普通のヒープテーブルや分割データセグメントを表しており、名前は「obj + データオブジェクトID」です。



Non-Dictionaryモードでは、SYSTEM表領域の喪失により、テーブルの構造情報（フィールド名やタイプ）が欠如しています。Oracleではこれらの情報はディクショナリ情報としてのみ保存され、データテーブル上には存在しません。そのため、ユーザーはデータセグメント上のROW行データを基にして、各フィールドのタイプを推測する必要があります。以下のような主要なデータタイプが解析されます：

- 文字列（String）：char、varcharなど
- 国家言語文字列（NString）：nchar、nvarcharなど
- 数値（Number）
- 日付（Date）

- タイムスタンプ (TimeStamp)
- タイムゾーン付きタイムスタンプ (TimeStamp Zone)
- CLOB
- BLOB

サンプルデータ分析 (Sample Data Analysis) を行います。

The screenshot shows the DBRecover for Oracle Professional interface. The left pane displays a tree view of the database structure, with the object 'obj74046' selected. The main pane shows the following table:

Col#	Seen	Count	Max Size	NULL PCT	String	NString	Number	Date	Timestamp	Timestamp Zone	Clob	Blob
1	1500	3	0	0	0	0	1500	0	0	0	0	0
2	1500	6	0	1500	644	0	0	0	0	0	0	0
3	1500	9	0	1500	428	0	0	0	0	0	0	0
4	1500	3	7	0	0	0	1393	0	0	0	0	0
5	1500	7	0	0	0	0	0	1500	1500	0	0	0

Below this table, the 'Sample data analysis:' section is highlighted with a red box, showing the following data:

col1	col2	col3	col4	col5	col6	col7	col8
7369	SMITH	CLERK	7902	17-DEC-1980 00:00:00 AD	800	20	
7499	ALLEN	SALESMAN	7698	20-FEB-1981 00:00:00 AD	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981 00:00:00 AD	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981 00:00:00 AD	2975	20	
7654	MARTIN	SALESMAN	7698	28-SEP-1981 00:00:00 AD	1250	1400	30

The interface also includes a 'Try to analyze UNKNOWN column type:' section with a table of column types and an 'Unload statement:' section with a generated SQL statement:

```
able obj74046 segobjno 74046 ( col1 NUMBER, col2 VARCHAR2, col3 VARCHAR2, col4 NUMBER, col5 DATE, col6 NUMBER, col7 NUMBER,
```

At the bottom, the version information is displayed: 'Version 2009 - Community Edition - 2020.09.25' and 'Copyright © 2012 - 2020 ParnassusData Software, Inc.' with a link to <https://www.dbrecover.com>.

このセクションでは、フィールドタイプの解析結果に基づいて10件のデータを解析し、結果を表示します。このサンプルデータは、ユーザーがそのデータセグメントに格納されている実際のデ

ータを理解するのに役立ちます。データセグメントに10件未満のレコードがある場合、すべてのレコードが表示されます。

The screenshot shows the DBRecover for Oracle Professional Oracle Database Disaster Recovery Version 2009 interface. The left pane shows a tree view of the database structure, with 'obj63389' selected under 'Extents'. The main pane displays the following information:

obj63389 :

Col#	Seen Count	Max Size	NULL PCT	String	NString	Number	Date	Timestamp	Timestamp Zone	Clob	Blob
1	7	16	0	0	0	0	0	0	0	0	0
2	7	14	0	7	6	0	0	0	0	0	0

Sample data analysis:

col1	col2
3BA5B92CFC6B40B69F6A7312E9F591B0	MISSING_ATTR
3BA5B92CFC6B40B69F6A7312E9F591B0	INVALID_LENGTH
3BA5B92CFC6B40B69F6A7312E9F591B0	MISSING_MAGIC
3BA5B92CFC6B40B69F6A7312E9F591B0	MISSING_HEADER
3BA5B92CFC6B40B69F6A7312E9F591B0	INVALID_VR

Try to analyze UNKNOWN column type:

Columns	Date	Number	String(VARCHAR2 CHAR)	Timestamp	Timestamp with time zone	NString(NVARCHAR2 NCHAR)
1	3BA5B92CFC	3BA5B92CFC	;00,0k@00js 0000	3BA5B92CFC	3BA5B92CFC	楡戮0踰踰踰0踰
1	3BA5B92CFC	3BA5B92CFC	;00,0k@00js 0000	3BA5B92CFC	3BA5B92CFC	楡戮0踰踰踰0踰
1	3BA5B92CFC	3BA5B92CFC	;00,0k@00js 0000	3BA5B92CFC	3BA5B92CFC	楡戮0踰踰踰0踰
1	3BA5B92CFC	3BA5B92CFC	;00,0k@00js 0000	3BA5B92CFC	3BA5B92CFC	楡戮0踰踰踰0踰
1	3BA5B92CFC	3BA5B92CFC	;00,0k@00js 0000	3BA5B92CFC	3BA5B92CFC	楡戮0踰踰踰0踰

Unload statement:

```
unload table obj63389 segobjno 63389 ( col1 UNKNOWN, col2 VARCHAR2 )
```

Version 2009 - Community Edition - 2020.09.25
 Copyright © 2012 - 2020 ParnassusData Software, Inc.
<https://www.dbrecover.com>

「TRY TO ANALYZE UNKNOWN column type」 :

この部分は、フィールド解析機能がタイプを完全に特定できないフィールドに対し、様々なフィールドタイプを試して解析し、ユーザーが判断できるように提示します。

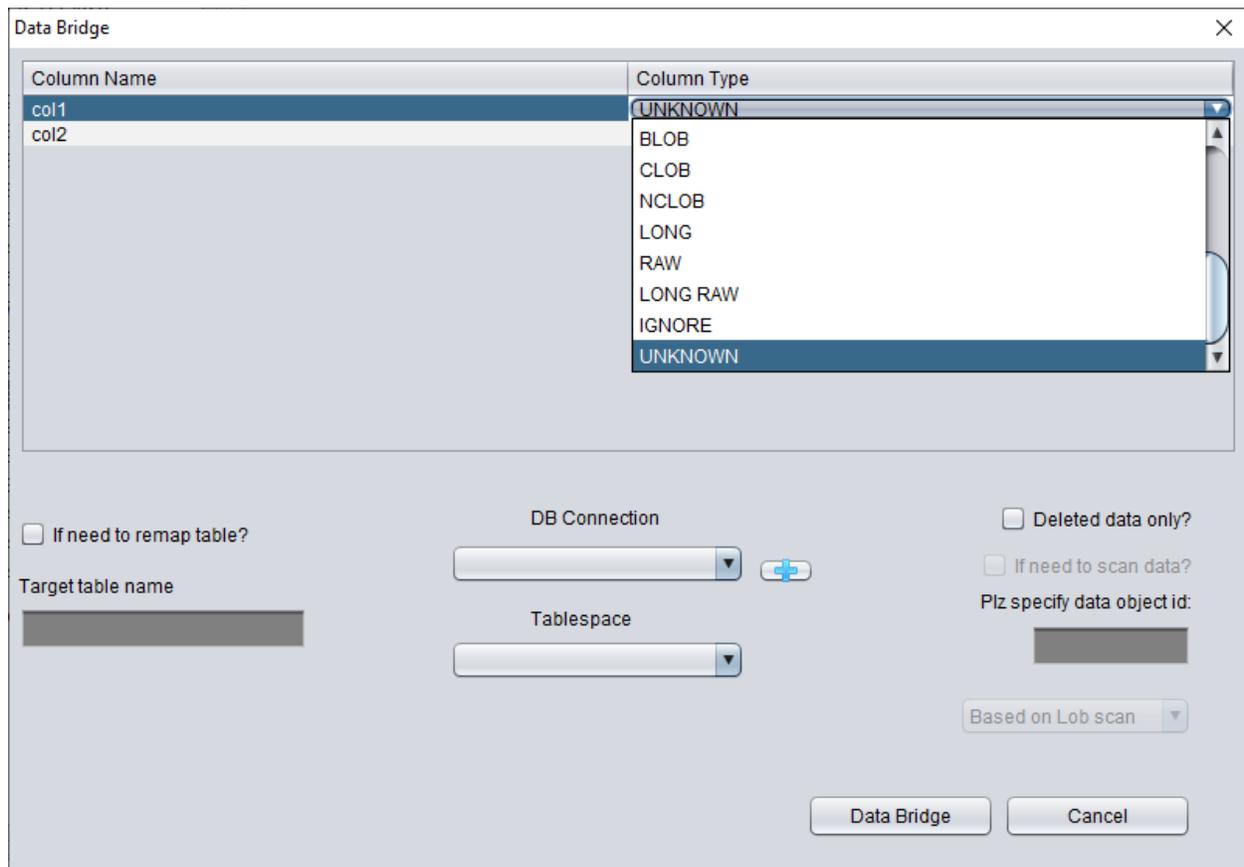
タイプが特定できないフィールドには以下のような状況があります :

1. RAWまたはLONG RAW

2. サポートされていないデータタイプ：XDB.XDB\$RAW_LIST_T、XMLTYPE、ユーザー定義タイプなど
3. データブロック自体が深刻に損傷している

「Non-Dictionary Mode」非字典モードでは、通常モードとデータブリッジモードを使用できません。字典モードとの主な違いは、非字典モードでのデータブリッジ時にユーザーがフィールドタイプを自由に決定できることです。下の画像に示されているように、いくつかのフィールドタイプは「UNKNOWN」、つまり未知となっています。

ユーザーがそのテーブルの設計構造を知っている場合（アプリケーション開発者のドキュメントからも得られる場合もあります）、正しい「Column Type」タイプを自分で選択して、そのテーブルのデータをターゲットデータベースにブリッジすることができます。



復旧シナリオ3: 勒索ウイルスソフトウェアによるデータファイルの暗号化と損傷

勒索ウイルス (ransomware malware) は、Oracleデータファイルの一部または全体を暗号化して破壊します。Oracleのデータファイルが一般に大きいため、全体を暗号化するのに時間がかかる可能性があります。そのため、一部の勒索ウイルスはOracleデータファイルのヘッダー部分のみを連続的またはランダムに暗号化することを選択します。

このような部分的な暗号化の損傷に対して、DBRECOVERを使用してデータを回復することができます。データファイルヘッダーが損傷しているため、SYSTEM01.DBFの内容を調査して、各データファイルの表領域番号(TS#)や相対ファイル番号(RFILE#)などの情報を特定する必要があります。

以下はデータファイルのリストです：

```
Administrator: Command Prompt
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

C:\Users\Administrator>cd C:\Users\Administrator\Desktop\DATAFILE

C:\Users\Administrator\Desktop\DATAFILE>dir
Volume in drive C is System Drive
Volume Serial Number is 5EB5-5EB4

Directory of C:\Users\Administrator\Desktop\DATAFILE

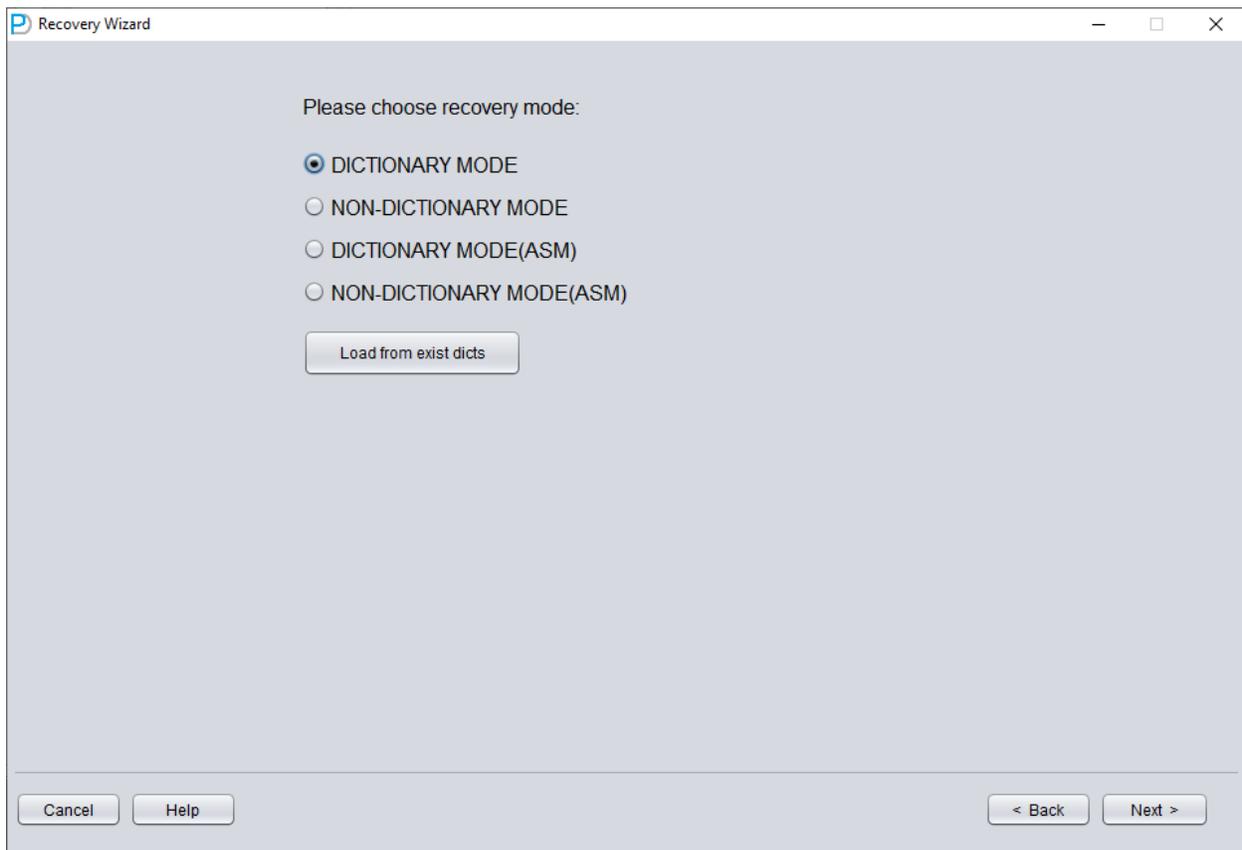
05/29/2023 11:35 AM <DIR> .
05/29/2023 11:22 AM <DIR> ..
05/29/2023 11:22 AM          524,296,192 01_MF_APP01_L782YY4Y_.DBF.eking
05/29/2023 11:22 AM          104,865,792 01_MF_APP01_L782ZBM3_.DBF.eking
05/29/2023 11:22 AM          104,865,792 01_MF_APP01_L782ZCP1_.DBF.eking
05/29/2023 11:22 AM          524,296,192 01_MF_APP02_L782Z07W_.DBF.eking
05/29/2023 11:22 AM          104,865,792 01_MF_APP02_L78300TG_.DBF.eking
05/29/2023 11:22 AM          104,865,792 01_MF_APP02_L7830FJ6_.DBF.eking
05/29/2023 11:22 AM          524,296,192 01_MF_DBRECOVE_L6G7BIQ3_.DBF.eking
05/29/2023 11:22 AM          1,069,555,712 01_MF_SYSAUX_L5VP5QJ8_.DBF.eking
05/29/2023 11:22 AM           964,698,112 01_MF_SYSTEM_L5VP4N7Y_.DBF.eking
05/29/2023 07:03 AM          135,274,496 01_MF_TEMP_L5VPCQGO_.TMP.eking
05/29/2023 11:22 AM           68,165,632 01_MF_UNDOTBS1_L5VP66PM_.DBF.eking
05/29/2023 11:22 AM           10,493,952 01_MF_USERS_L5VP67TJ_.DBF.eking
          12 File(s) 4,240,539,648 bytes
           2 Dir(s) 6,546,952,192 bytes free

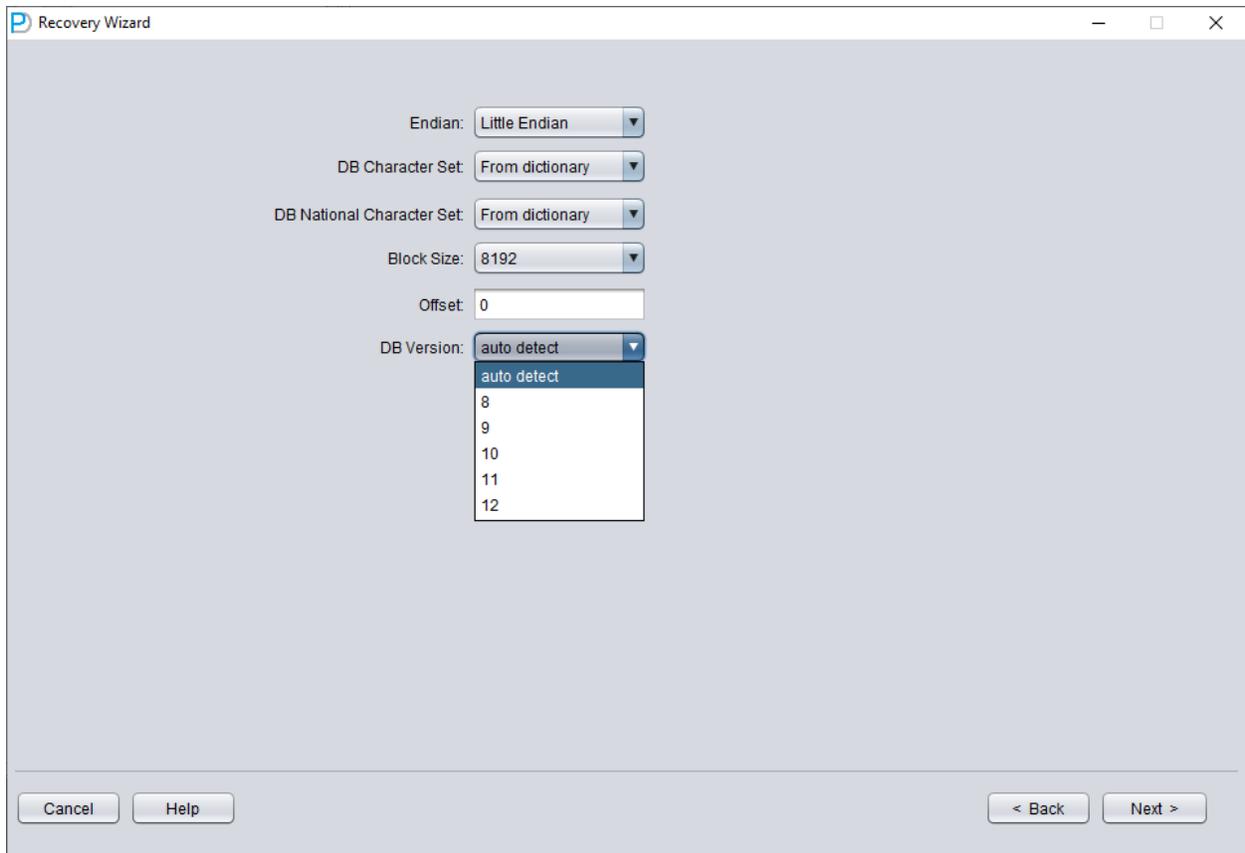
C:\Users\Administrator\Desktop\DATAFILE>
```

01_MF_APP01_L782YY4Y_.DBF.eking

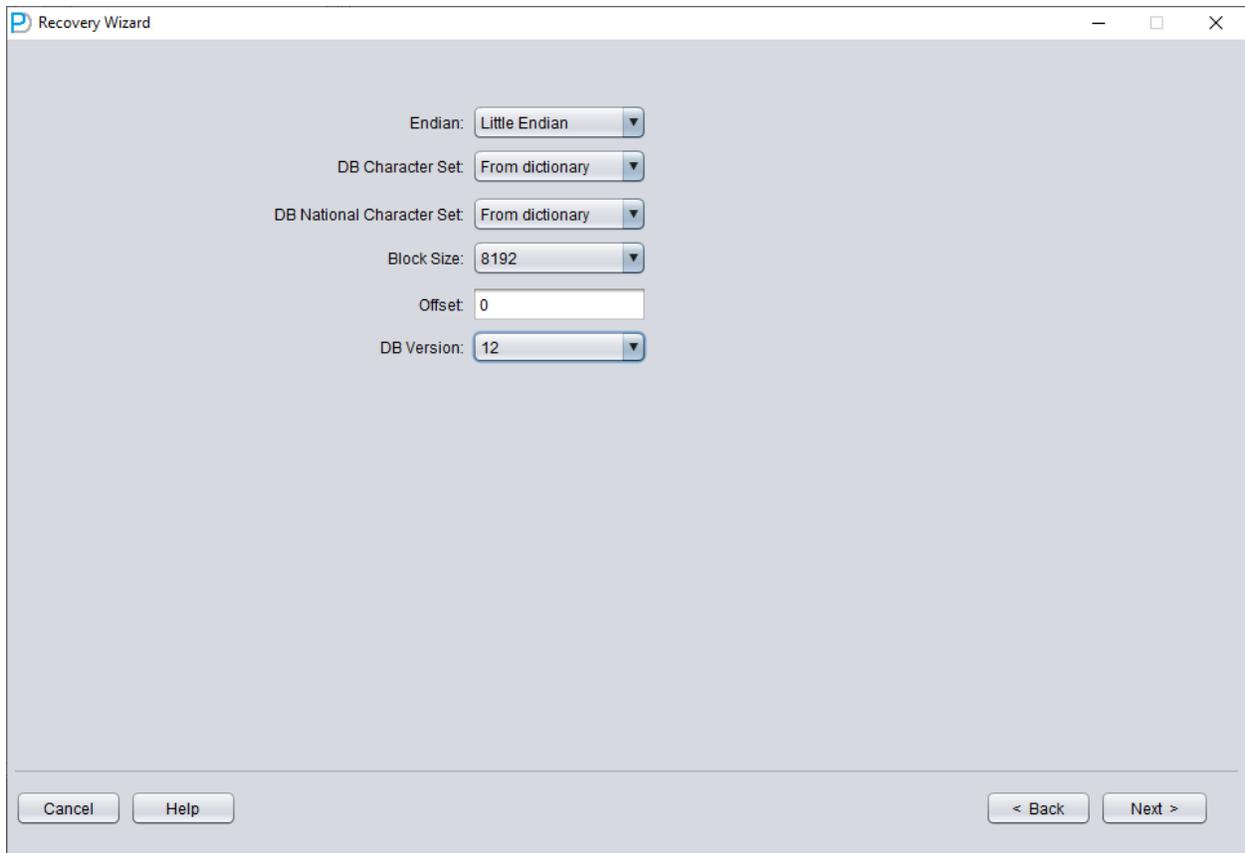
01_MF_APP01_L782ZBM3_.DBF.eking
01_MF_APP01_L782ZCP1_.DBF.eking
01_MF_APP02_L782Z07W_.DBF.eking
01_MF_APP02_L7830DTG_.DBF.eking
01_MF_APP02_L7830FJ6_.DBF.eking
01_MF_DBRECOVE_L6G7B1Q3_.DBF.eking
01_MF_SYSAUX_L5VP5QJ8_.DBF.eking
01_MF_SYSTEM_L5VP4N7Y_.DBF.eking
01_MF_TEMP_L5VPCQG0_.TMP.eking
01_MF_UNDOTBS1_L5VP66PM_.DBF.eking
01_MF_USERS_L5VP67TJ_.DBF.eking

勒索ウイルス（例: 拡張子「eking」）により暗号化されたファイルを扱います。TEMP、UNDOTBS1、SYSAUXは無視し、DBRECOVERを「DICT-MODE」で起動します。

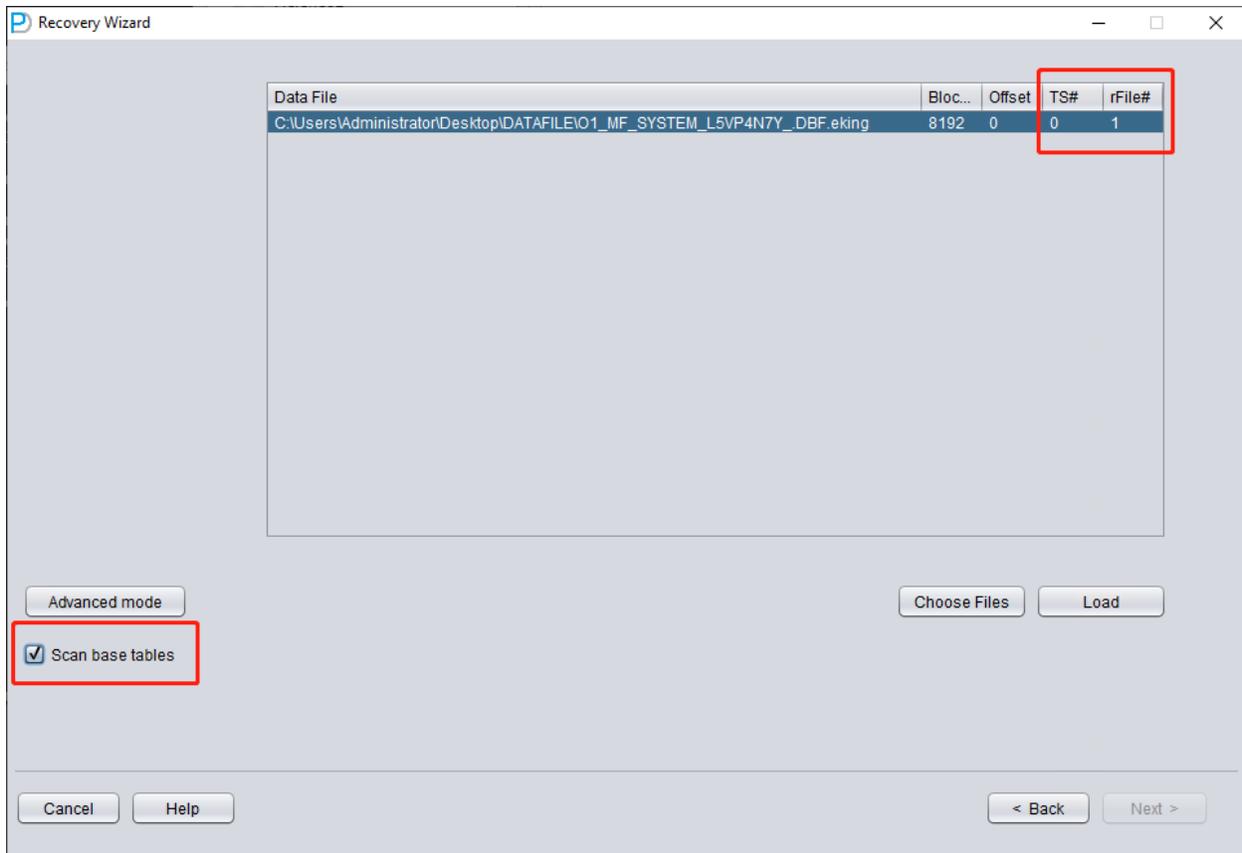




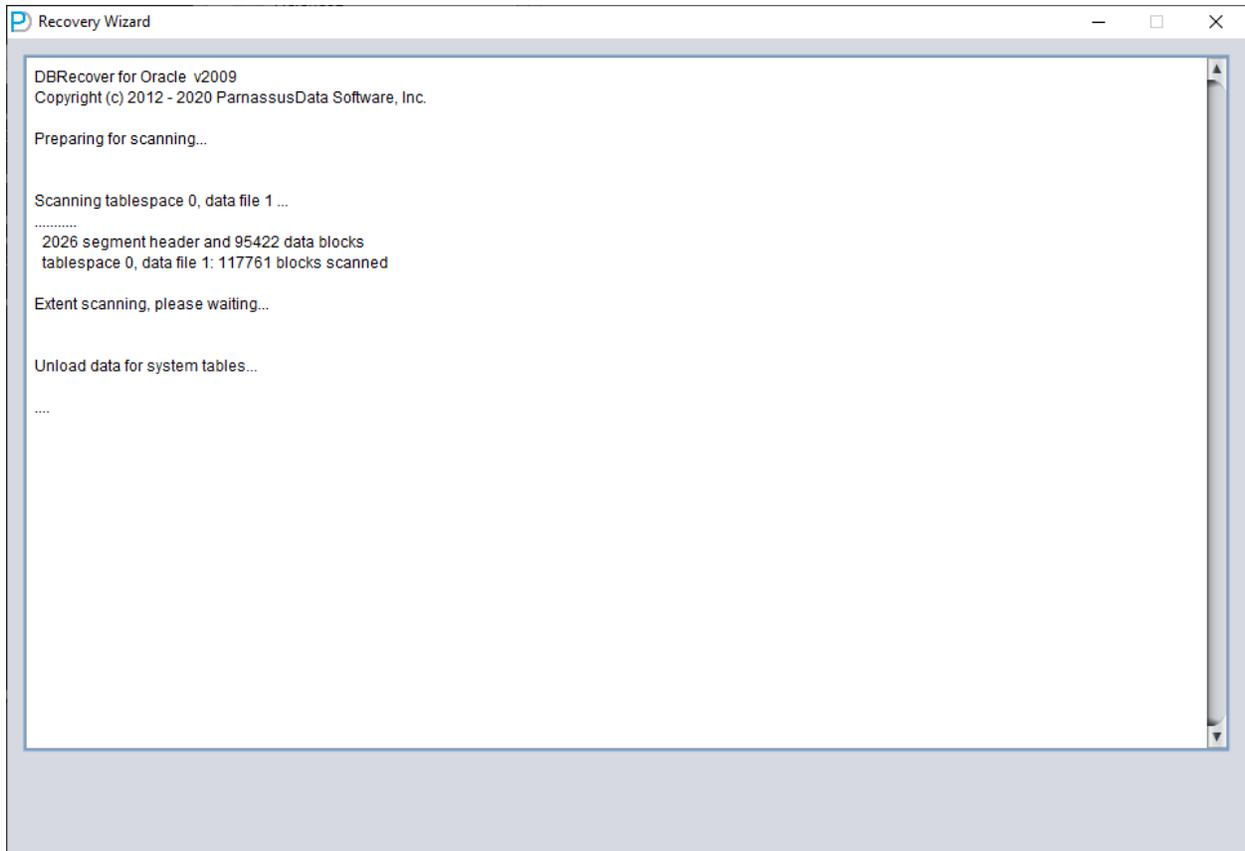
DBバージョンは実際の状況に応じて選択し、12c以上（例: 18c、19c）は12を選択。



SYSTEM01.DBFのみを追加し、TS#=0、RFILE#=1を指定。



SCAN BASE TABLESオプションでより強力に対応。LOAD後、SYSTEM01.DBFをスキャンしてデータ辞書基本テーブルデータを検出。



SYSユーザーノードでTS\$とFILE\$基本テーブルを探索。

DBRecover for Oracle email: service@pamassusdata.com www.dbrecover.com Professional Oracle Database Disaster Recovery Version 2009

Start Options

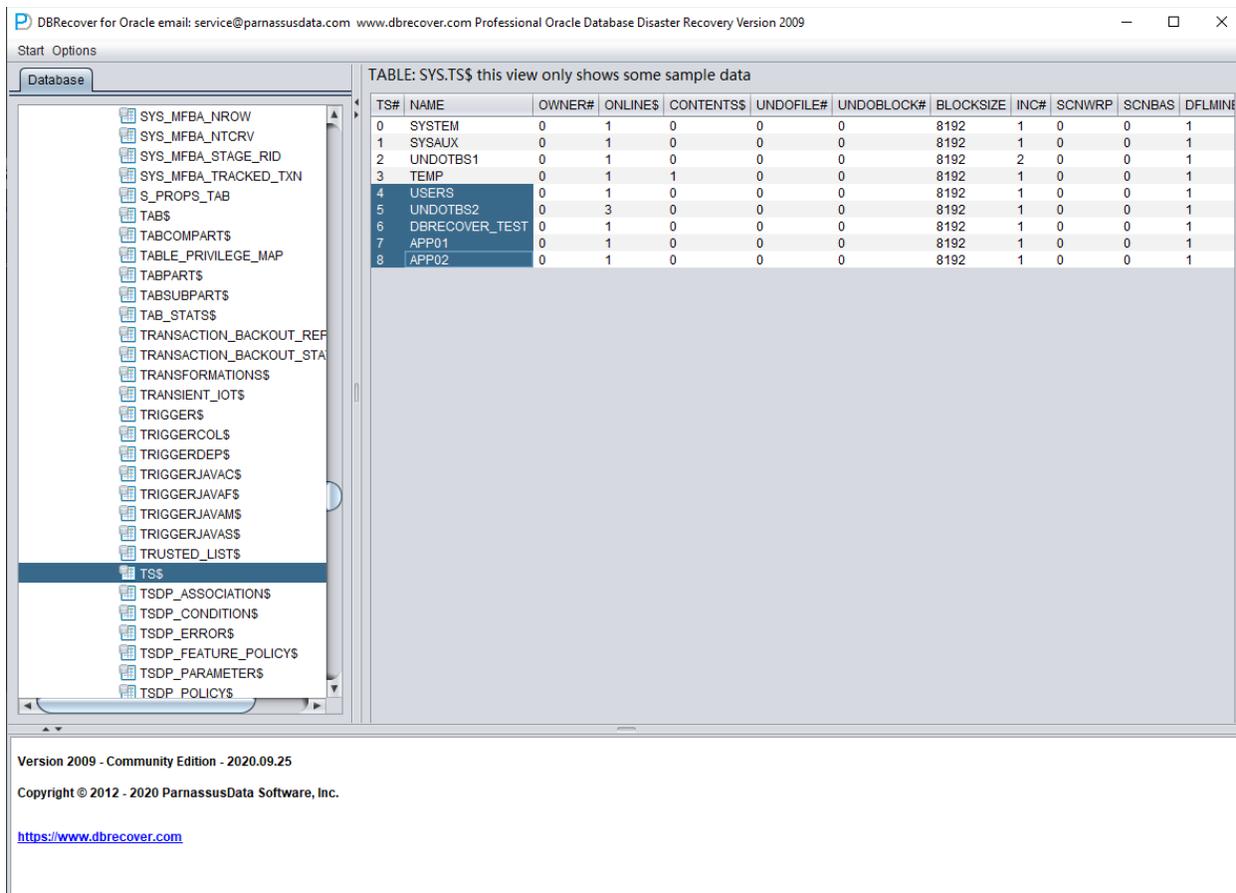
Database

- Database
 - DB_20230529123253
 - Users
 - GSMROOTUSER
 - PD
 - SCOTT
 - SYS
 - Tables
 - ACCESS\$
 - ACLMV\$
 - ACLMV\$REFLOG
 - ACLMVREFSTAT\$
 - ACLMVSUBTBL\$
 - ADMINAUTH\$
 - ADO_IMPPARAM\$
 - ADO_IMSEGSTAT\$
 - ADO_IMSEGTASKDETAILS\$
 - ADO_IMSTAT\$
 - ADO_IMTASK\$
 - ALERT_QT
 - ALL_UNIFIED_AUDIT_ACTIONS
 - APPLY\$_AUTO_CDR_COLUMN
 - APPLY\$_BATCH_SQL_STATS
 - APPLY\$_CDR_INFO
 - APPLY\$CHANGE_HANDLERS
 - APPLY\$CONF_HDLR_COLUMN
 - APPLY\$CONSTRAINT_COLUMN

TABLE: SYS.TS\$ this view only shows some sample data

TS#	NAME	OWNER#	ONLINE\$	CONTENT\$	UNDOFILE#	UNDOBLOCK#	BLOCKSIZE	INC#	SCNWRP	SCNBAS	DFLMIN\$
0	SYSTEM	0	1	0	0	0	8192	1	0	0	1
1	SYS_AUX	0	1	0	0	0	8192	1	0	0	1
2	UNDOTBS1	0	1	0	0	0	8192	2	0	0	1
3	TEMP	0	1	1	0	0	8192	1	0	0	1
4	USERS	0	1	0	0	0	8192	1	0	0	1
5	UNDOTBS2	0	3	0	0	0	8192	1	0	0	1
6	DBRECOVER_TEST	0	1	0	0	0	8192	1	0	0	1
7	APP01	0	1	0	0	0	8192	1	0	0	1
8	APP02	0	1	0	0	0	8192	1	0	0	1

Version 2009 - Community Edition - 2020.09.25
 Copyright © 2012 - 2020 PamassusData Software, Inc.
<https://www.dbrecover.com>



TS\$テーブルは表領域情報を格納し、TS#列で表領域番号を特定します。

TS#	NAME
0	SYSTEM
1	SYSAUX
2	UNDOTBS1
3	TEMP
4	USERS
5	UNDOTBS2
6	DBRECOVER_TEST
7	APP01
8	APP02

APP01表領域のTS#は7、APP02表領域のTS#は8です。FILE\$テーブルにはデータファイル情報があり、TS#とRELFIL#の2列が必要です。

TABLE: SYS.FILE\$ this view only shows some sample data

FILE#	STATUS\$	BLOCKS	TS#	RELFIL#	MAXEXTEND	INC	CRSCNWRP	CRSCNBAS	OWNERINSTANCE	SPARE1	SPARE2	SPARE
1	2	64000	0	1	4194302	1280	0	8	4194306	0		
3	2	51200	1	3	4194302	1280	0	6336	12582914	0		
5	2	64000	6	5	0	0	0	2447818	20971522	0		
7	2	640	4	7	4194302	160	0	37099	29360130	0		
2	2	64000	7	2	0	0	0	3207824	8388610	0		
4	2	3200	2	4	4194302	640	0	1928280	16777218	0		
8	2	12800	7	8	0	0	0	3207853	33554434	0		
9	2	12800	7	9	0	0	0	3207872	37748738	0		
10	2	64000	8	10	0	0	0	3207900	41943042	0		
11	2	12800	8	11	0	0	0	3207935	46137346	0		
12	2	12800	8	12	0	0	0	3207955	50331650	0		

Version 2009 - Community Edition - 2020.09.25
 Copyright © 2012 - 2020 ParnassusData Software, Inc.
<https://www.dbrecover.com>

TS#	RELFIL#
0	1
1	3
6	5

4	7
7	2
2	4
7	8
7	9
8	10
8	11
8	12

これらのテーブルのデータをマッピングして結合することで、不要なSYSAUX、UNDOTBS1、既知のSYSTEM表領域を除いて、残りのデータファイル名とその対応関係を特定できます。

TS#	RELFIL#	Tablespace Name
0	1	SYSTEM
1	3	SYSAUX
6	5	DBRECOVER_TEST
4	7	USERS
7	2	APP01
2	4	UNDOTBS1
7	8	APP01
7	9	APP01
8	10	APP02
8	11	APP02
8	12	APP02

TS#	RELFIL#	Tablespace Name
6	5	DBRECOVER_TEST
4	7	USERS

7	2	APP01
7	8	APP01
7	9	APP01
8	10	APP02
8	11	APP02
8	12	APP02

01_MF_APP01_L782YY4Y_.DBF.eking

01_MF_APP01_L782ZBM3_.DBF.eking

01_MF_APP01_L782ZCP1_.DBF.eking

01_MF_APP02_L782ZO7W_.DBF.eking

01_MF_APP02_L7830DTG_.DBF.eking

01_MF_APP02_L7830FJ6_.DBF.eking

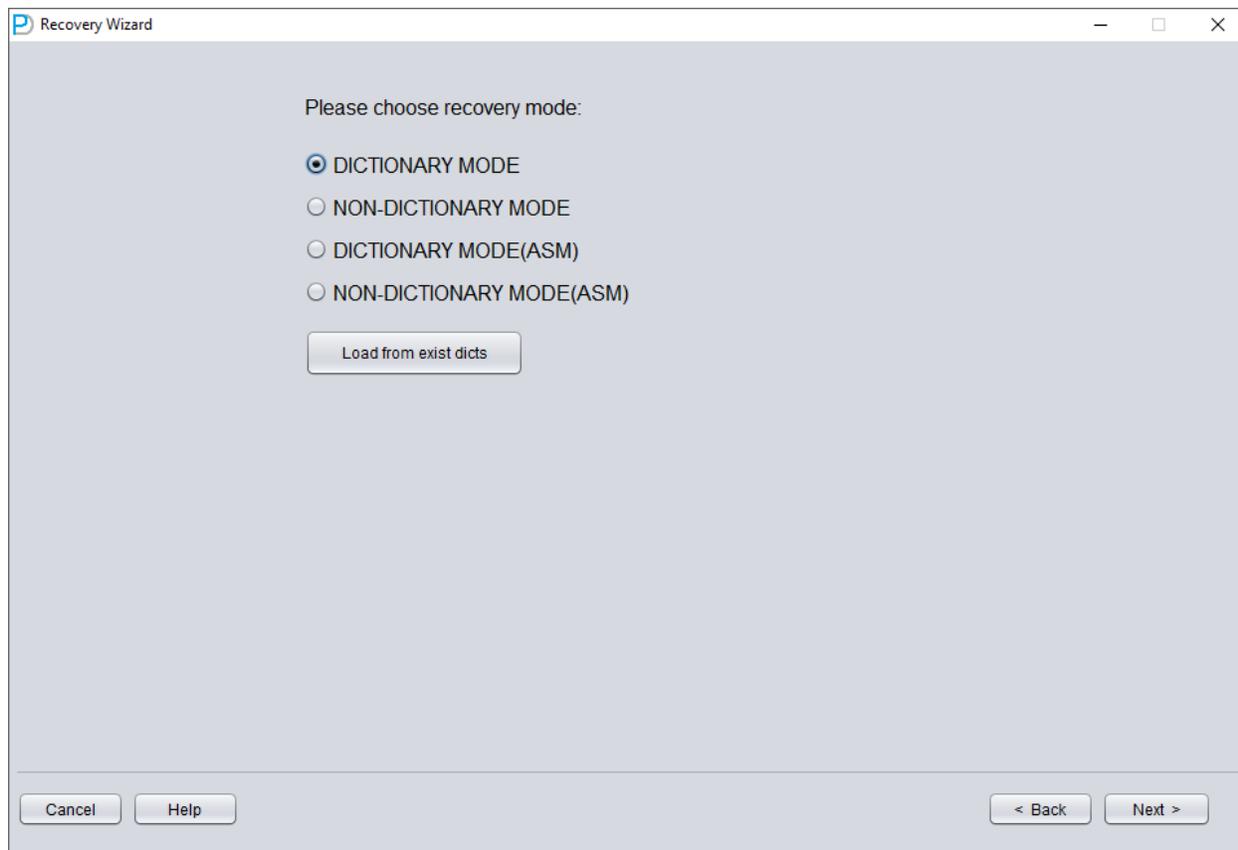
01_MF_DBRECOVE_L6G7B1Q3_.DBF.eking

01_MF_USERS_L5VP67TJ_.DBF.eking

2つのテーブルを比較すると、それぞれの対応関係が明らかになります。OMFで管理されたファイルやユーザー管理のファイル名（例：APP0101、APP0102）からも関係を導き出せます。これにより、完全な情報表が推測できます。

TS#	RFILE#	Tablespace Name	FILE NAME
6	5	DBRECOVER_TEST	O1_MF_DBRECOVE_L6G7B1Q3_.DBF.eking
4	7	USERS	O1_MF_USERS_L5VP67TJ_.DBF.eking
7	2	APP01	O1_MF_APP01_L782YY4Y_.DBF.eking
7	8	APP01	O1_MF_APP01_L782ZBM3_.DBF.eking
7	9	APP01	O1_MF_APP01_L782ZCP1_.DBF.eking
8	10	APP02	O1_MF_APP02_L782ZO7W_.DBF.eking
8	11	APP02	O1_MF_APP02_L7830DTG_.DBF.eking
8	12	APP02	O1_MF_APP02_L7830FJ6_.DBF.eking

DBRECOVERを再度開いて、字典モードで処理を続けます。



データベースバージョンを選択し。

Recovery Wizard

Endian: Little Endian

DB Character Set: From dictionary

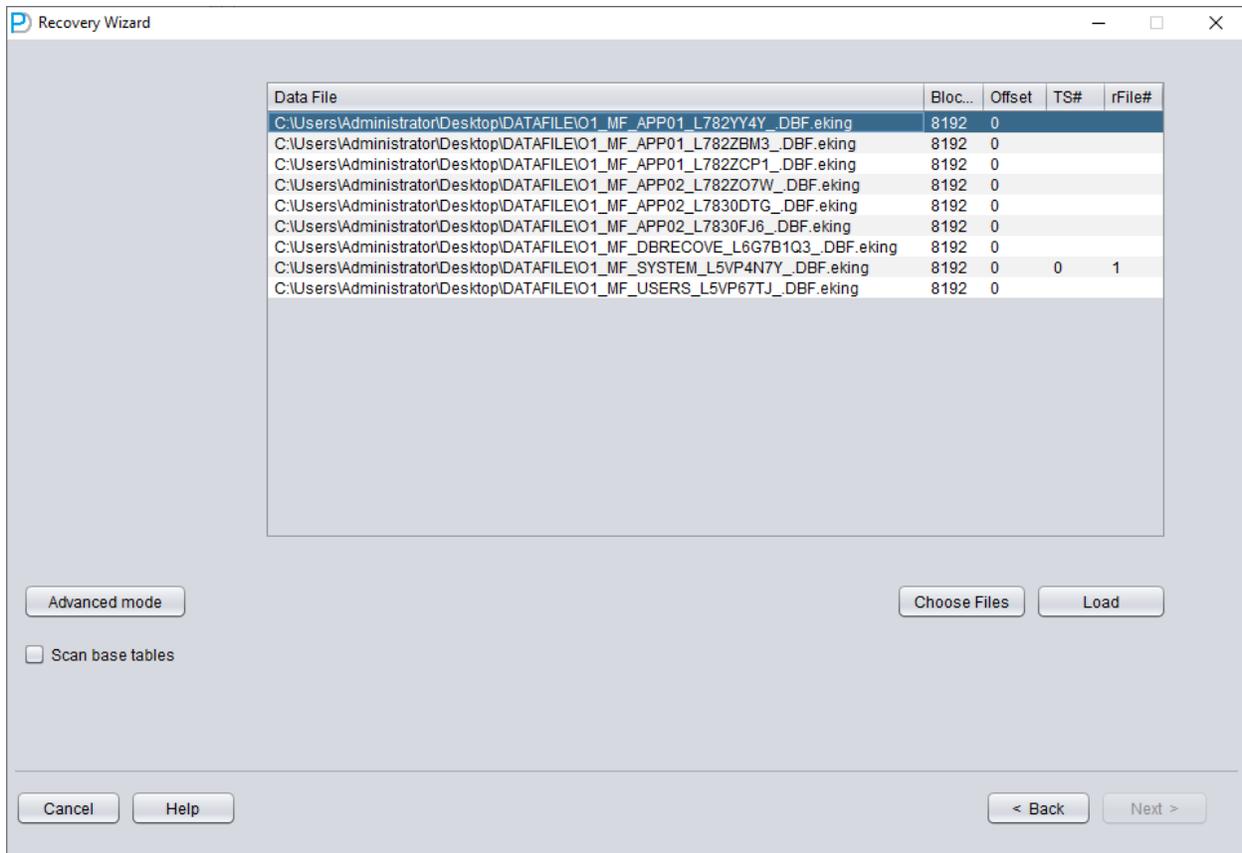
DB National Character Set: From dictionary

Block Size: 8192

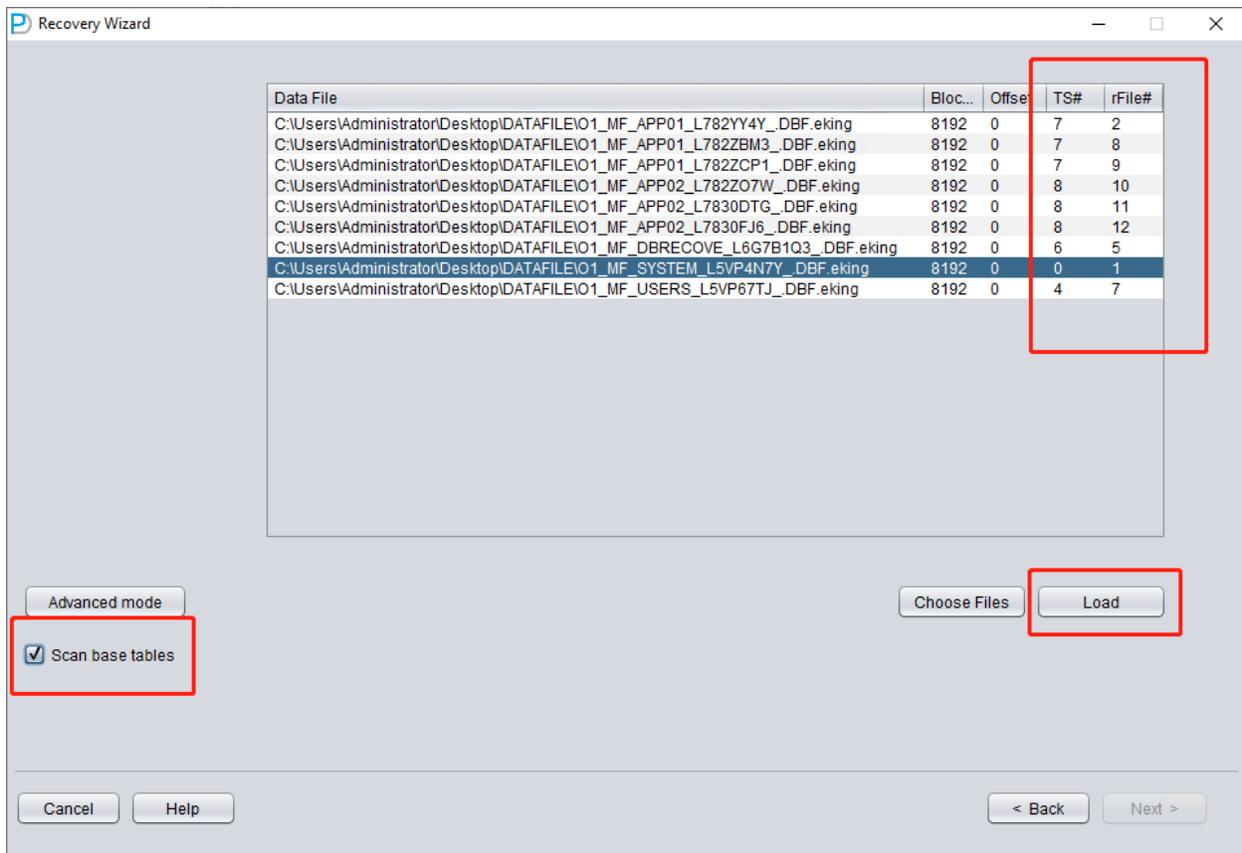
Offset: 0

DB Version: 12

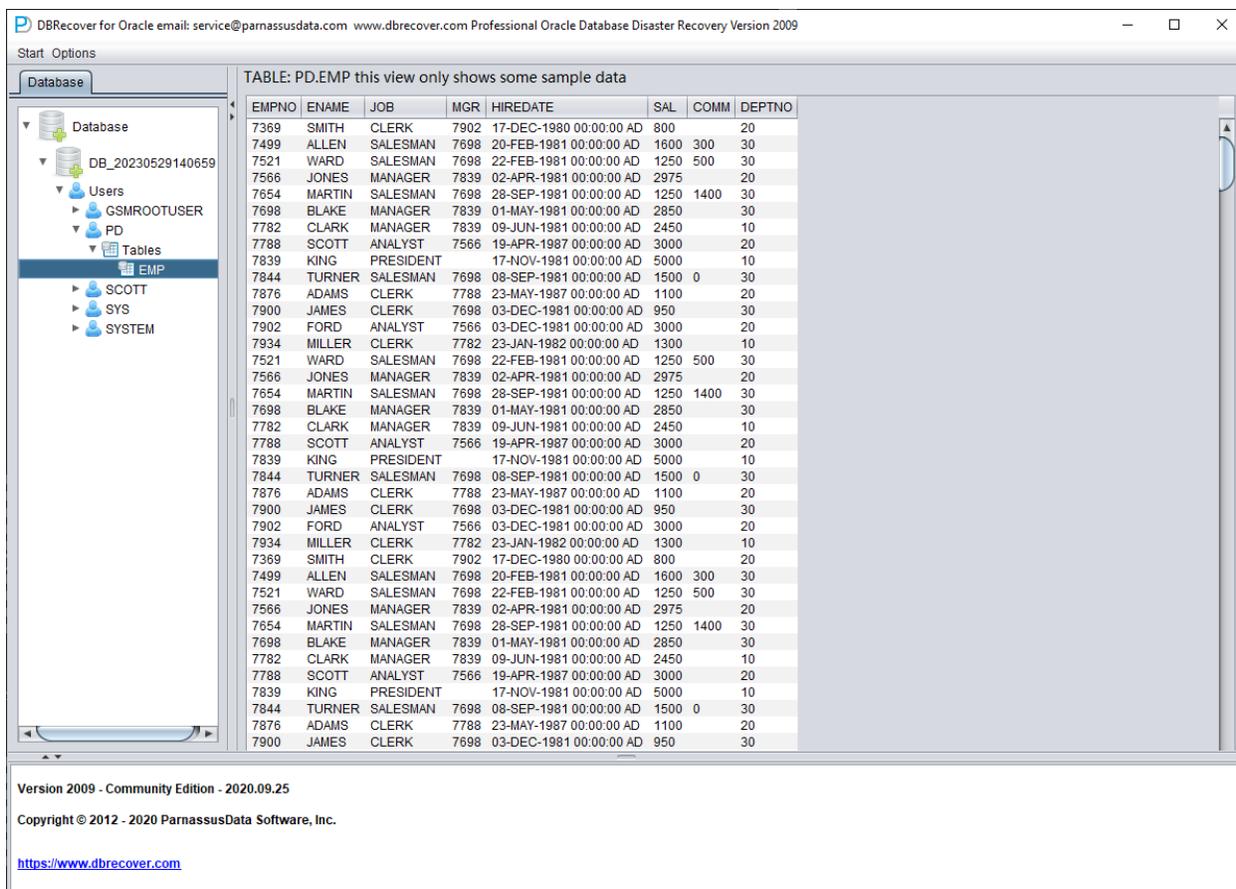
Cancel Help < Back Next >



必要なデータファイル（ユーザーデータを含む可能性のある全ファイル、但しUNDOTBS1、TEMP、SYSAUXを除く）を追加し、TS#とRFILE#情報を入力します。



適切に情報を入力し、暗号化の損傷が軽度であれば、データを直接読み取ることができます。



勒索ウイルスの特性は異なるため、実際の操作ではより多くの問題に直面する可能性があります。問題があれば、service@pamassusdata.comまでメールでお問い合わせください。

復旧シナリオ4: 誤ってDELETE FROM TABLEでデータ行を削除した場合の回復

D社の開発者が誤ってテスト環境の削除スクリプトを本番環境に接続して実行し、あるテーブルのデータをすべてDELETEしました。DBRECOVERを使用して削除されたデータを回復できますが、データの上書きを最大限に防ぐために以下の操作が必要です：

1. 対象表領域を読み取り専用を設定：`ALTER TABLESPACE {TABLESPACE_NAME} READ ONLY`

2. データベースインスタンスを停止：**SHUTDOWN IMMEDIATE**

これらのいずれかを選択してください。

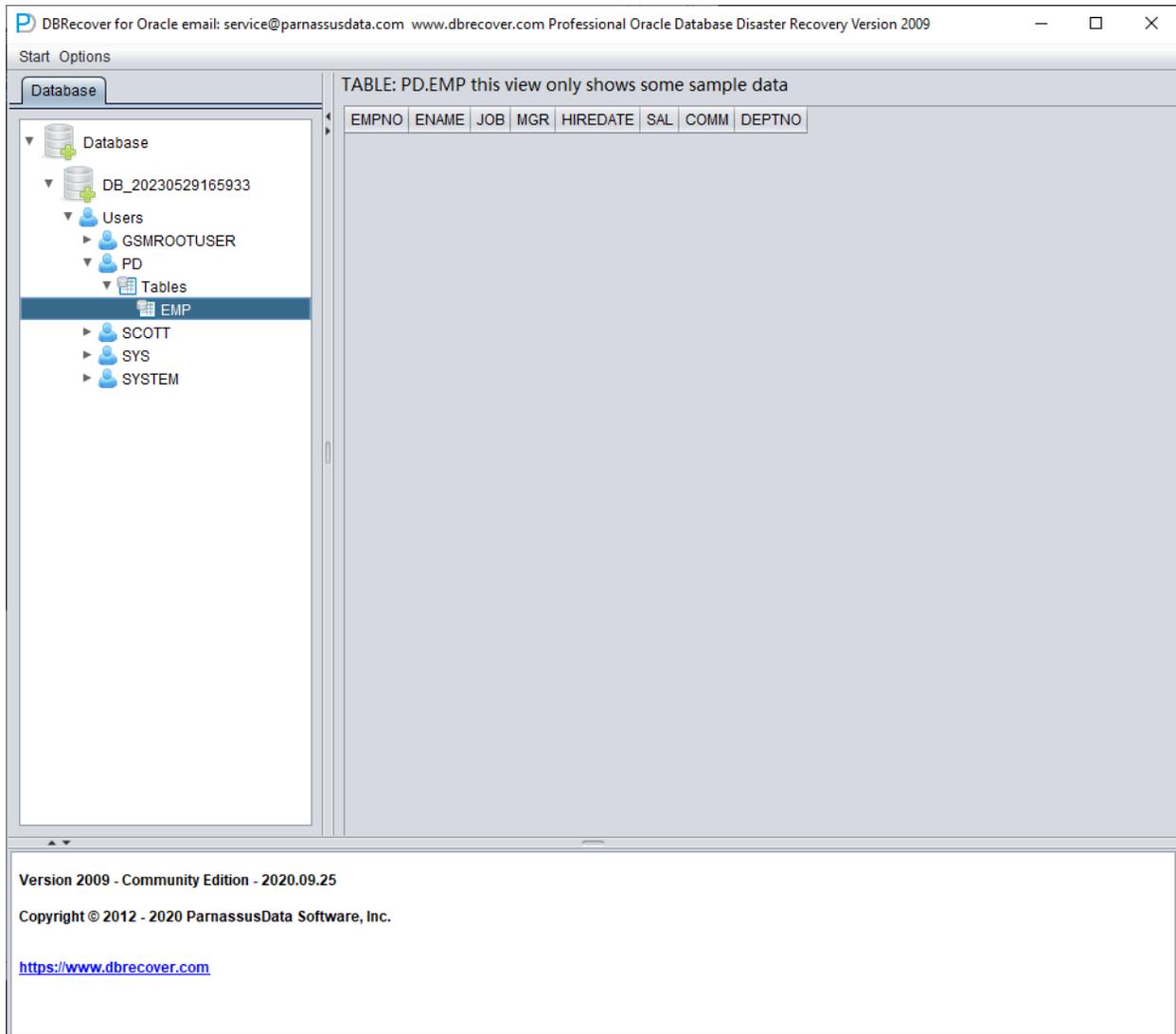
```
SQL> select count(*) from pd.emp;
COUNT(*)
-----
114688
SQL>
```

```
SQL> delete from pd.emp;
114688 rows deleted.
SQL> commit;
Commit complete.
SQL> alter system checkpoint;
System altered.
SQL> select count(*) from pd.emp;
COUNT(*)
-----
0
```

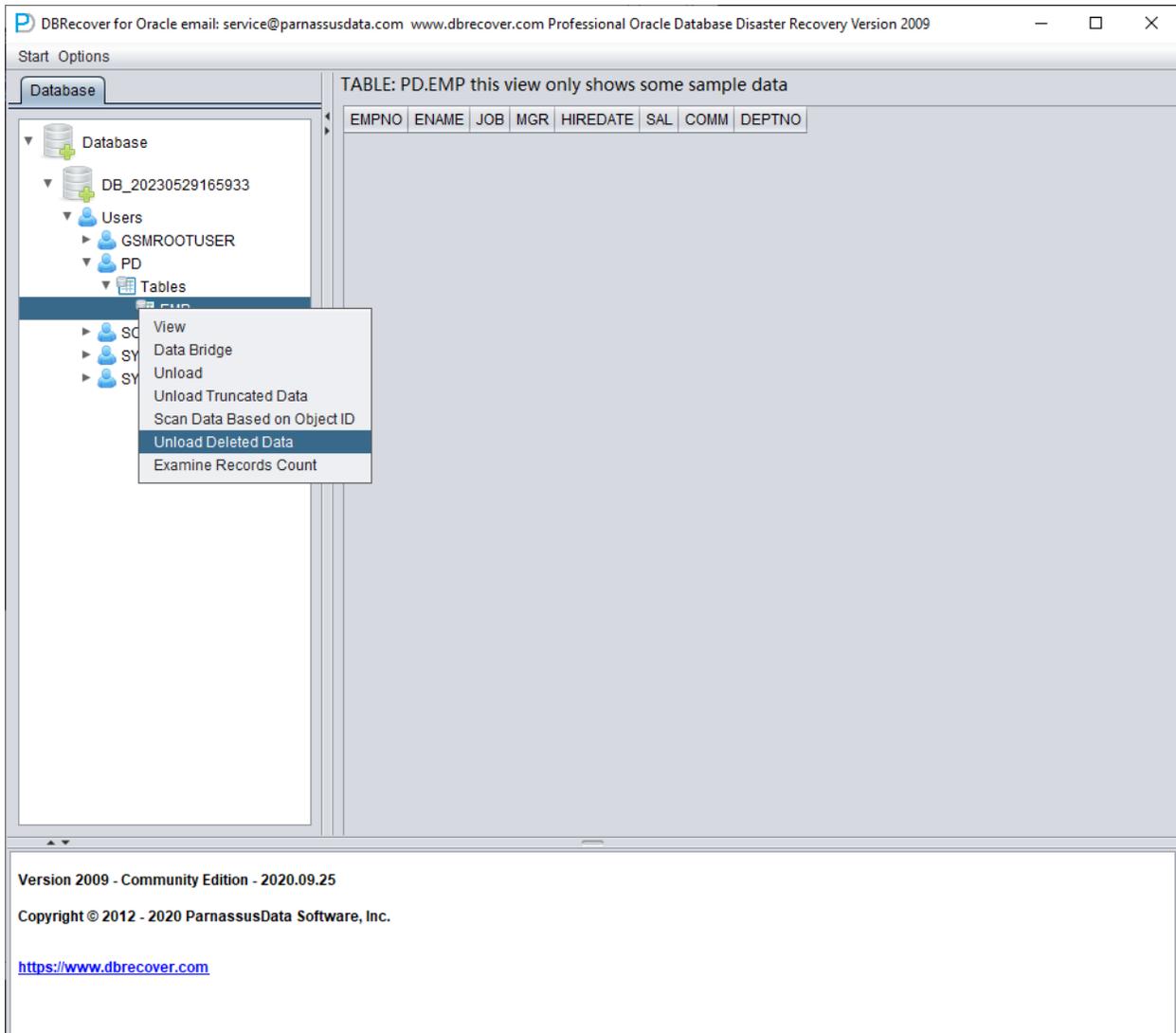
復旧を開始する前に、表領域を読み取り専用を設定して環境を保護します。

```
SQL> select tablespace_name from dba_segments where owner='PD' and segment_name='EMP';
TABLESPACE_NAME
-----
DBRECOVER_TEST
SQL> alter tablespace DBRECOVER_TEST read only;
Tablespace altered.
```

DBRECOVERを起動し、字典モードで利用可能なすべてのデータファイルを追加します。



選択した表から「Unload Deleted Data」を使用してデータを抽出し、企業版ライセンスがない場合は各テーブルから100行のデータのみが抽出されます。



抽出されたデータは表示されたパスに保存され、ユーザーはSQLLDRやSQLDEVELOPERなどのツールを使用してデータベースにデータを挿入する必要があります。

```
C:\dbrecover-for-oracle2009\prmdata\pamassus_dbinfo_DB_20230529165933\deleted\pd.emp.dat - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
pd.emp.dat
1 "7369" "SMITH" "CLERK" "7902" "17-DEC-1980 00:00:00 AD" "800" "" "20"
2 "7499" "ALLEN" "SALESMAN" "7698" "20-FEB-1981 00:00:00 AD" "1600" "300" "30"
3 "7521" "WARD" "SALESMAN" "7698" "22-FEB-1981 00:00:00 AD" "1250" "500" "30"
4 "7566" "JONES" "MANAGER" "7839" "02-APR-1981 00:00:00 AD" "2975" "" "20"
5 "7654" "MARTIN" "SALESMAN" "7698" "28-SEP-1981 00:00:00 AD" "1250" "1400" "30"
6 "7698" "BLAKE" "MANAGER" "7839" "01-MAY-1981 00:00:00 AD" "2850" "" "30"
7 "7782" "CLARK" "MANAGER" "7839" "09-JUN-1981 00:00:00 AD" "2450" "" "10"
8 "7788" "SCOTT" "ANALYST" "7566" "19-APR-1987 00:00:00 AD" "3000" "" "20"
9 "7839" "KING" "PRESIDENT" "" "17-NOV-1981 00:00:00 AD" "5000" "" "10"
10 "7844" "TURNER" "SALESMAN" "7698" "08-SEP-1981 00:00:00 AD" "1500" "0" "30"
11 "7876" "ADAMS" "CLERK" "7788" "23-MAY-1987 00:00:00 AD" "1100" "" "20"
12 "7900" "JAMES" "CLERK" "7698" "03-DEC-1981 00:00:00 AD" "950" "" "30"
13 "7902" "FORD" "ANALYST" "7566" "03-DEC-1981 00:00:00 AD" "3000" "" "20"
14 "7934" "MILLER" "CLERK" "7782" "23-JAN-1982 00:00:00 AD" "1300" "" "10"
15 "7521" "WARD" "SALESMAN" "7698" "22-FEB-1981 00:00:00 AD" "1250" "500" "30"
16 "7566" "JONES" "MANAGER" "7839" "02-APR-1981 00:00:00 AD" "2975" "" "20"
17 "7654" "MARTIN" "SALESMAN" "7698" "28-SEP-1981 00:00:00 AD" "1250" "1400" "30"
18 "7698" "BLAKE" "MANAGER" "7839" "01-MAY-1981 00:00:00 AD" "2850" "" "30"
19 "7782" "CLARK" "MANAGER" "7839" "09-JUN-1981 00:00:00 AD" "2450" "" "10"
20 "7788" "SCOTT" "ANALYST" "7566" "19-APR-1987 00:00:00 AD" "3000" "" "20"
21 "7839" "KING" "PRESIDENT" "" "17-NOV-1981 00:00:00 AD" "5000" "" "10"
22 "7844" "TURNER" "SALESMAN" "7698" "08-SEP-1981 00:00:00 AD" "1500" "0" "30"
23 "7876" "ADAMS" "CLERK" "7788" "23-MAY-1987 00:00:00 AD" "1100" "" "20"
24 "7900" "JAMES" "CLERK" "7698" "03-DEC-1981 00:00:00 AD" "950" "" "30"
25 "7902" "FORD" "ANALYST" "7566" "03-DEC-1981 00:00:00 AD" "3000" "" "20"
26 "7934" "MILLER" "CLERK" "7782" "23-JAN-1982 00:00:00 AD" "1300" "" "10"
27 "7369" "SMITH" "CLERK" "7902" "17-DEC-1980 00:00:00 AD" "800" "" "20"
28 "7499" "ALLEN" "SALESMAN" "7698" "20-FEB-1981 00:00:00 AD" "1600" "300" "30"
29 "7521" "WARD" "SALESMAN" "7698" "22-FEB-1981 00:00:00 AD" "1250" "500" "30"
30 "7566" "JONES" "MANAGER" "7839" "02-APR-1981 00:00:00 AD" "2975" "" "20"
31 "7654" "MARTIN" "SALESMAN" "7698" "28-SEP-1981 00:00:00 AD" "1250" "1400" "30"
32 "7698" "BLAKE" "MANAGER" "7839" "01-MAY-1981 00:00:00 AD" "2850" "" "30"
33 "7782" "CLARK" "MANAGER" "7839" "09-JUN-1981 00:00:00 AD" "2450" "" "10"
34 "7788" "SCOTT" "ANALYST" "7566" "19-APR-1987 00:00:00 AD" "3000" "" "20"
35 "7839" "KING" "PRESIDENT" "" "17-NOV-1981 00:00:00 AD" "5000" "" "10"
36 "7844" "TURNER" "SALESMAN" "7698" "08-SEP-1981 00:00:00 AD" "1500" "0" "30"
37 "7876" "ADAMS" "CLERK" "7788" "23-MAY-1987 00:00:00 AD" "1100" "" "20"
38 "7900" "JAMES" "CLERK" "7698" "03-DEC-1981 00:00:00 AD" "950" "" "30"
39 "7902" "FORD" "ANALYST" "7566" "03-DEC-1981 00:00:00 AD" "3000" "" "20"
40 "7934" "MILLER" "CLERK" "7782" "23-JAN-1982 00:00:00 AD" "1300" "" "10"
Normal text file length: 7,529 lines: 101 Ln: 1 Col: 1 Pos: 1 Windows (CR LF) UTF-8 INS
```

復旧シナリオ5 誤操作によるTruncateテーブルの復元

D社のビジネスメンテナンススタッフが製品データベースをテスト環境データベースと誤認し、テーブル上の全データを誤ってTRUNCATEしたため、DBAが回復を試みたが、最近のバックアップが使用不可で、そのデータテーブル上のレコードをバックアップから回復することができない

かった。そこでDBAは、DBRECOVERを使用してTRUNCATEされたデータを回復することにした。

この環境では、すべてのデータベースファイルが利用可能で健全であるため、ユーザーは辞書モードでSYSTEMテーブルスペースのデータファイルと、TRUNCATEDされたテーブルのデータファイルをロードするだけでよい。例えば：

```
SQL> select count(*) From pd.salgrade;
```

```
COUNT(*)
```

```
-----
```

```
655360
```

```
SQL> select tablespace_name from dba_segments where owner='PD' and segment_name='SALGRADE';
```

```
TABLESPACE_NAME
```

```
-----
```

```
APP01
```

```
SQL> truncate table pd.salgrade;
```

```
Table truncated.
```

```
SQL>
```

```
SQL> alter system checkpoint;
```

```
System altered.
```

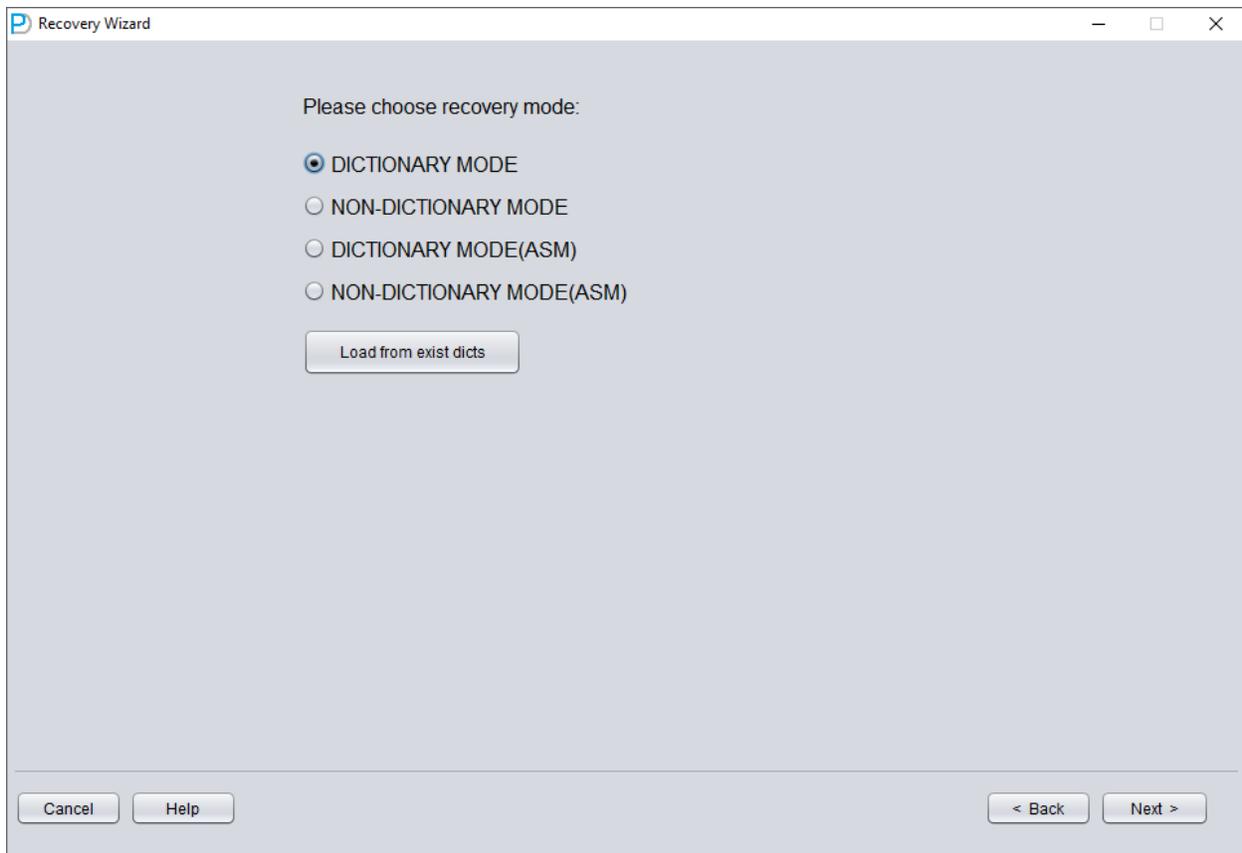
```
SQL> select count(*) from pd.salgrade;
```

```
COUNT(*)
```

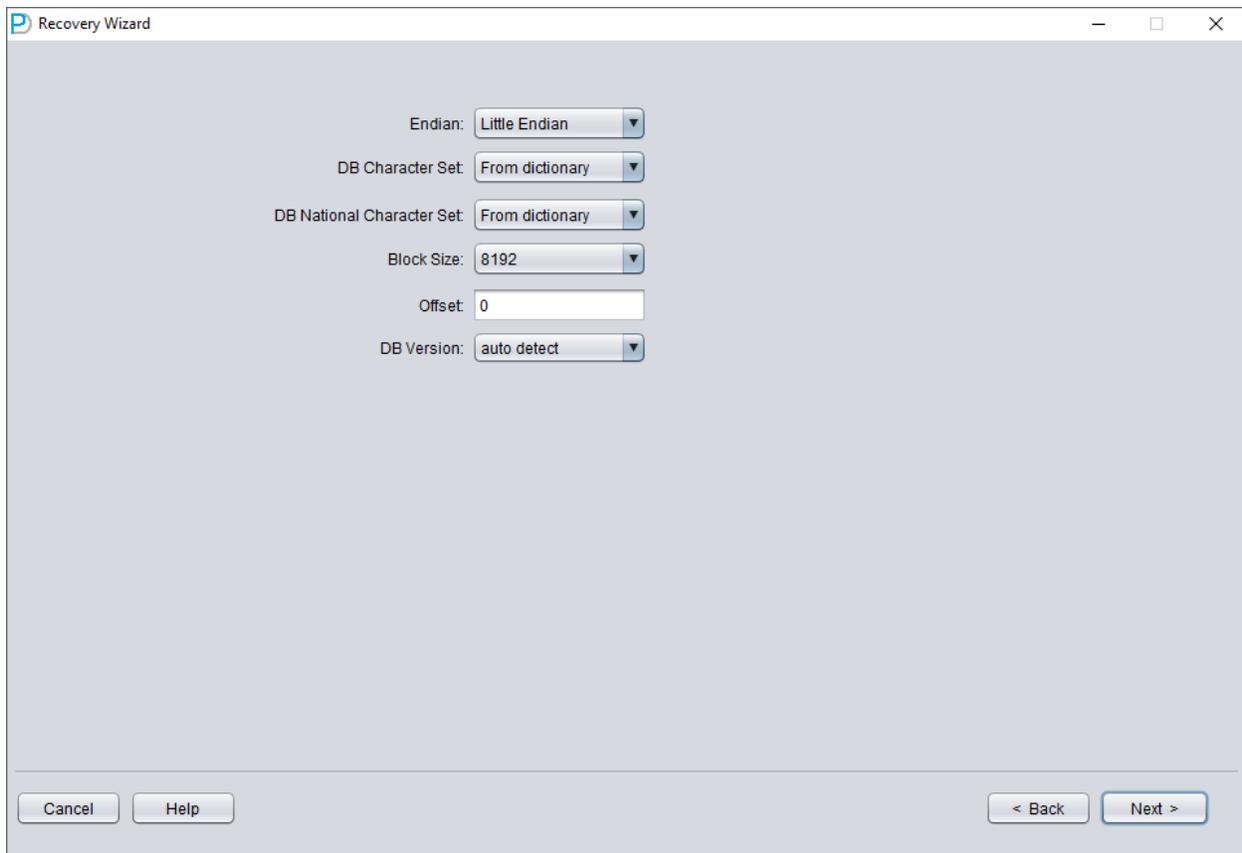
```
-----
```

```
0
```

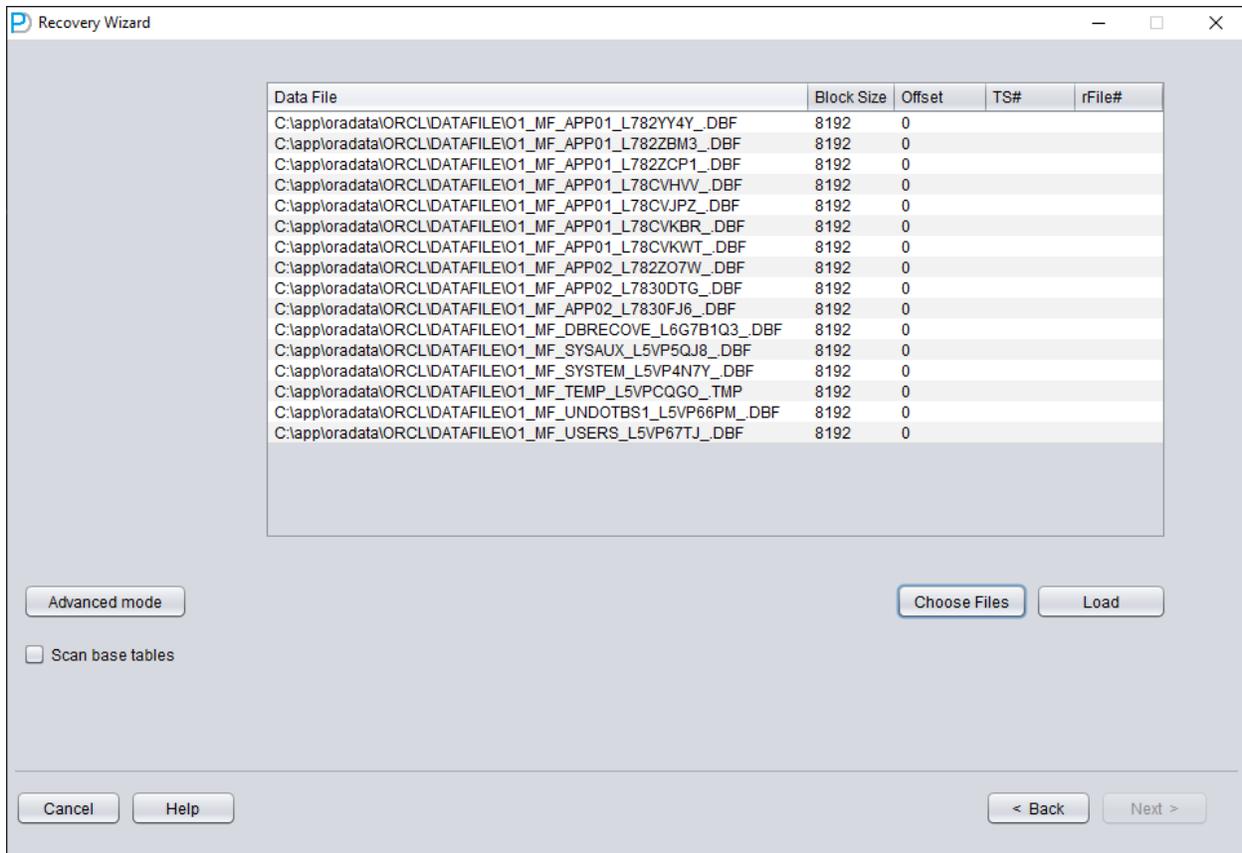
このTRUNCATEシナリオではASMストレージを使用していないため、《Dictionary Mode》辞書モードの選択のみが必要です：



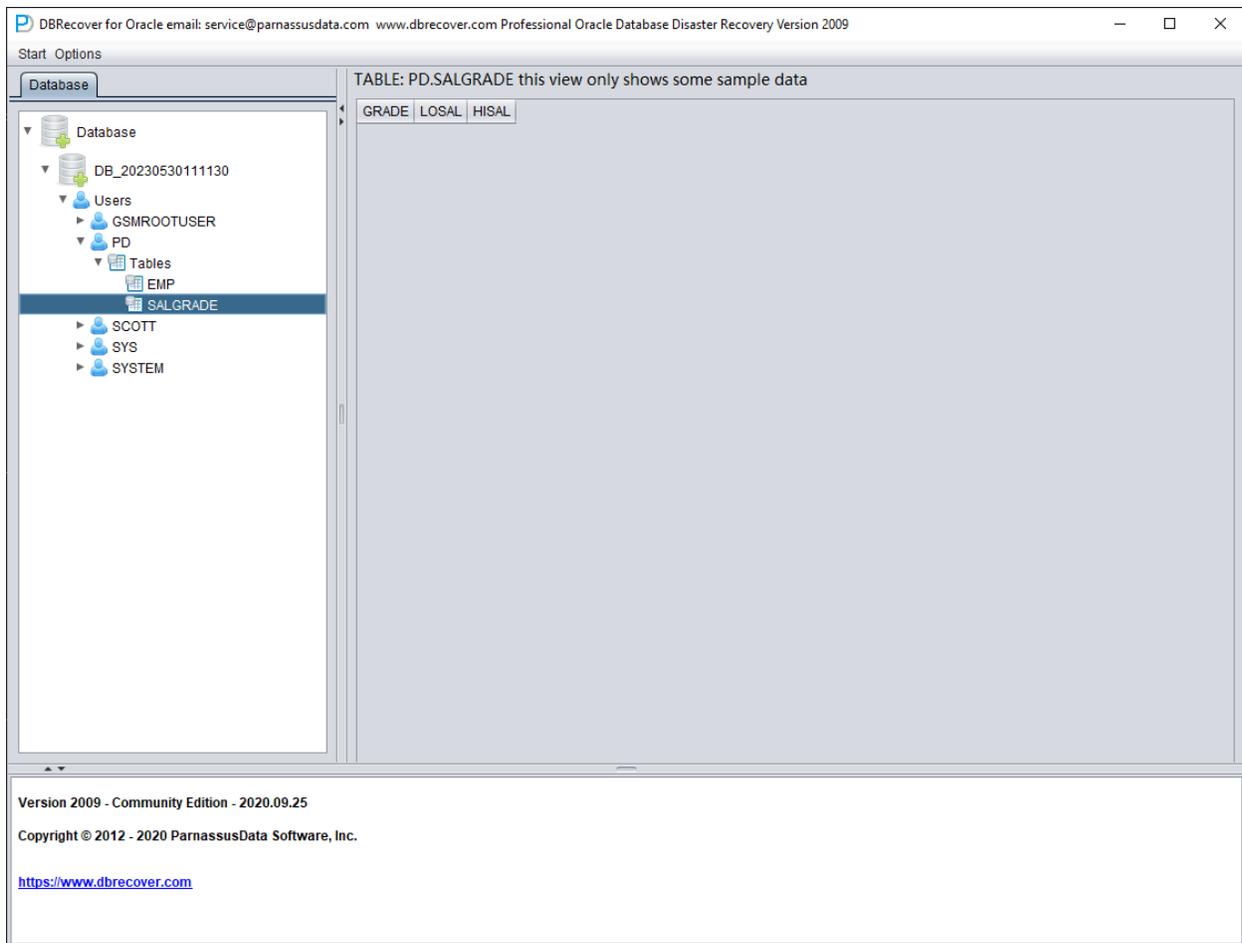
ほとんどの場合、パラメータを変更する必要はありません：



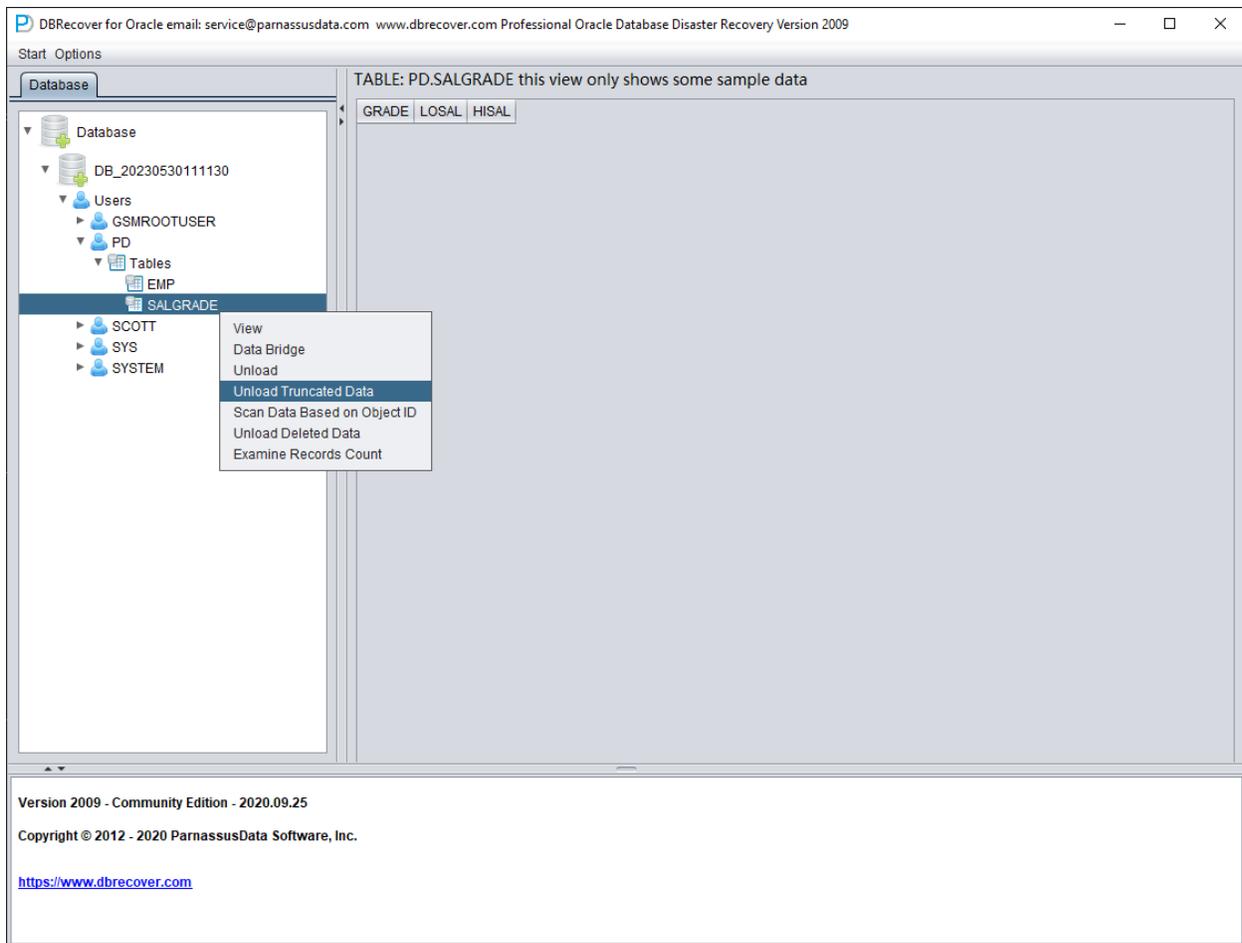
使用可能なすべてのデータファイルを追加してください：



USERSをクリックすると、複数のユーザー名が表示されます。たとえば、PD SCHEMAの下のテーブルを復元する必要がある場合、PDを開き、テーブル名をダブルクリックします：



このテーブルは以前にTRUNCATEDされていたため、ダブルクリックしてもデータは表示されません。この時点で、テーブル上で右クリックし、Unload truncated dataを選択します：



DBRECOVERは、そのテーブルが存在するテーブルスペースをスキャンし、TRUNCATEDされたデータを抽出します：上記のように、TRUNCATEされたテーブルから655360の完全なレコー

ドを抽出し、指定されたパスに保存します。

ユーザーはこのDATファイルをチェックして復元結果を確認できます。

TRUNCATEされたデータを復元する際の重要な点の一つは、TRUNCATE前のテーブルのDATA_OBJECT_IDを確認することです。この例では：

```
SQL> select object_id ,data_object_id from dba_objects where owner='PD' and object_name='SALGRADE';
```

OBJECT_ID	DATA_OBJECT_ID
76112	76113

TRUNCATEが発生する前、このテーブルのOBJECT_IDとDATA_OBJECT_IDは両方とも76612でした。TRUNCATEの後、DATA_OBJECT_IDが変更されました。

したがって、ここでの元のDATA_OBJECT_IDは76612です。しかし、テーブルが多数の回数TRUNCATEされ、最後のTRUNCATE前のデータを復元する必要がある場合、OBJECT_IDを使用して元のDATA_OBJECT_IDを推測することはできません。

FLASHBACKクエリ、辞書検索、ログマイニングなどの技術を使用してDATA_OBJECT_IDを特定することができます。ここではFLASHBACKクエリの例を紹介します：

```
SQL> select user# from sys.user$ where name='PD';
```

USER#
106

```
SQL> select obj#,dataobj# from sys.obj$ as of timestamp systimestamp -1/24 where name='SALGRADE' and owner#=106;
```

OBJ#	DATAOBJ#
76112	76112

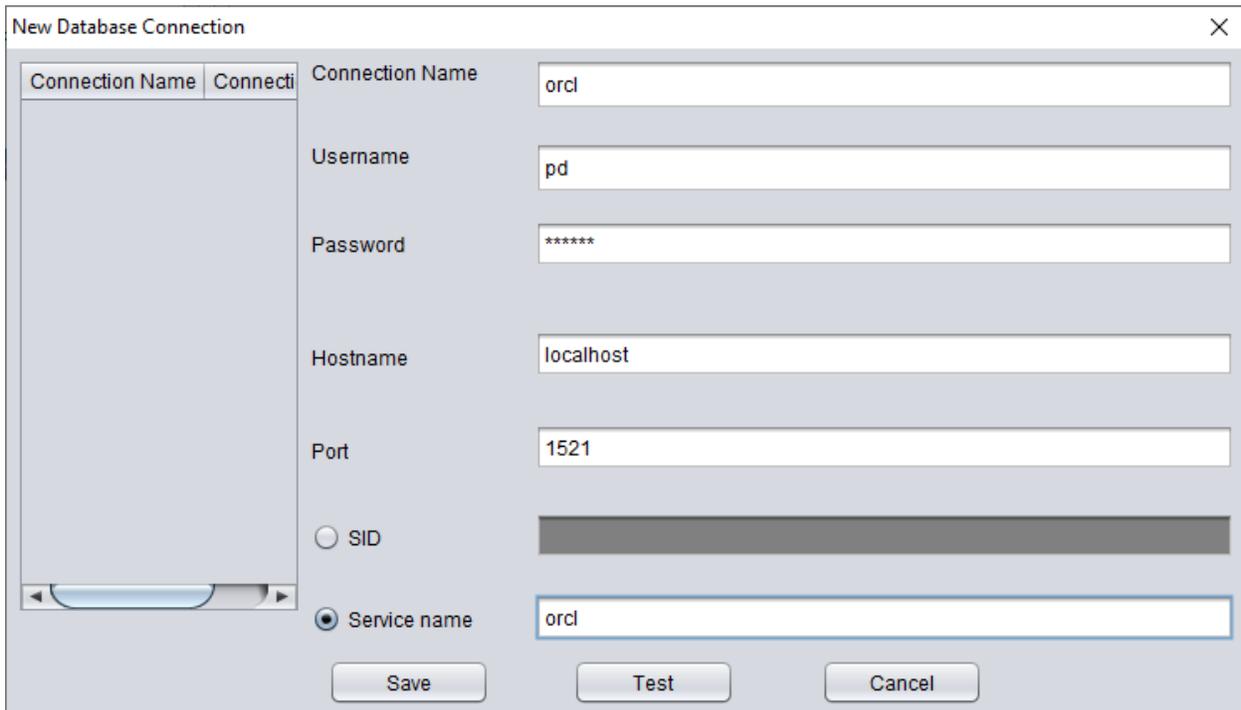
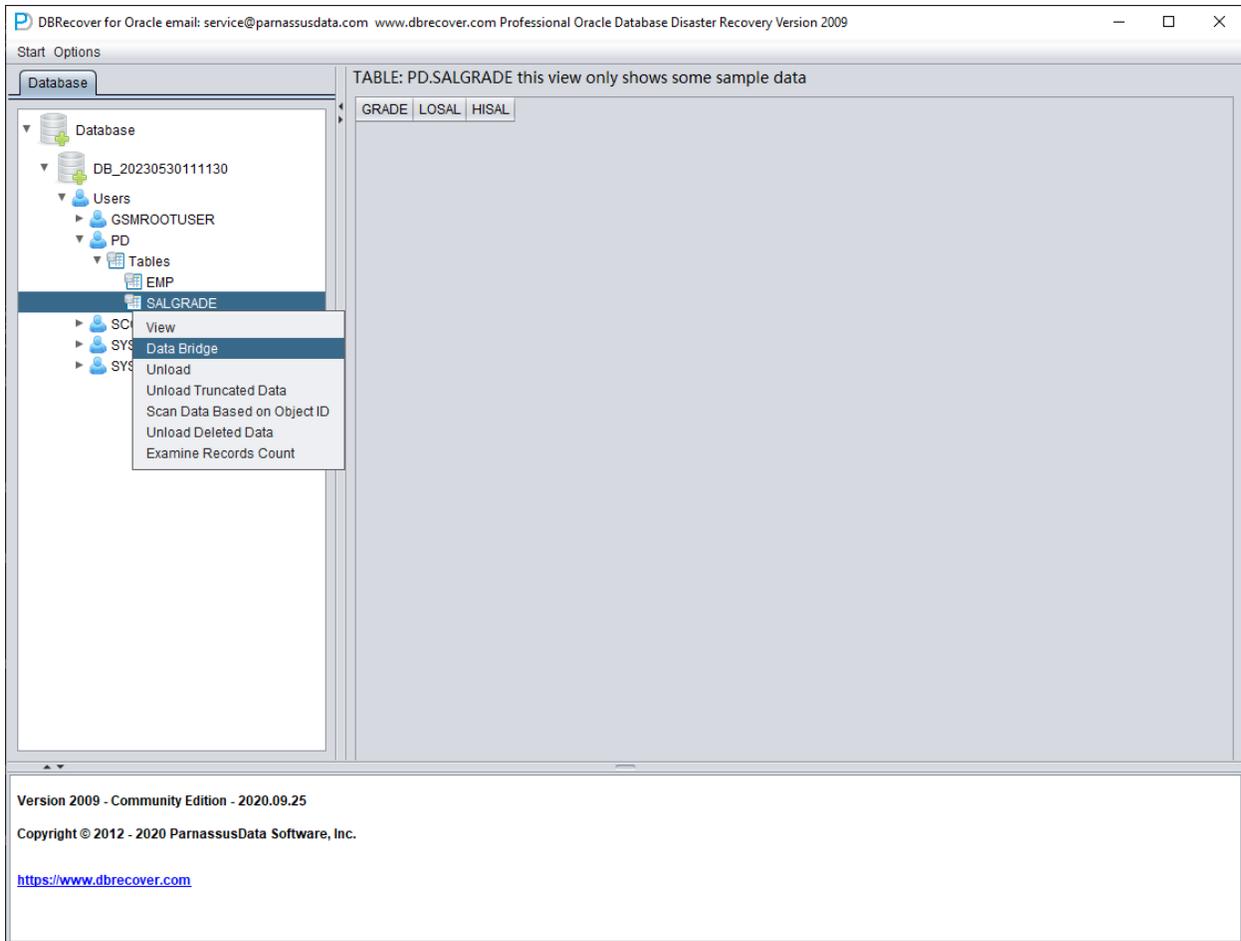
上記の方法でFLASHBACKクエリを使用して、元のDATAOBJ#、すなわちDATA_OBJECT_IDを取得しました。

その後、Data Bridgeの特性を利用して、復元する必要のあるデータをターゲットデータベースに挿入する必要があります。Data Bridgeを使用してtruncateデータを復元する際の注意点：元のデータベースからtruncateデータを復元し、Data Bridgeオプションを使用してデータを元のデータベースに戻す場合（データを元のデータベース以外に戻す場合はこの問題はありません）、Data Bridgeが新しく作成したテーブルに挿入する位置は、元のデータベース内でtruncateされたデータのテーブルスペースではなく、元のテーブルに挿入しないように注意してください。そうしないと、truncateデータの復元を行っている間に、必要とされるデータが新しいデータに上書きされ、この回復シナリオでのデータが完全に失われる可能性があります。したがって、Data Bridgeを使用して元のデータベースにデータを復元する場合、Data Bridgeでテーブルスペースを指定する際には、復元する必要のあるデータが存在するテーブルスペースを使用しないように注意してください！！！！！！

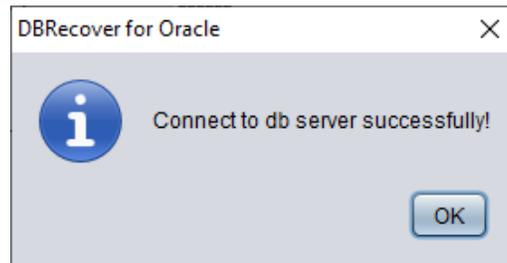
そこで、ここではまず新しいテーブルスペースを作成して、復元されたデータテーブルを格納します：

```
SQL> create tablespace pd_recover_data datafile size 600M;
```

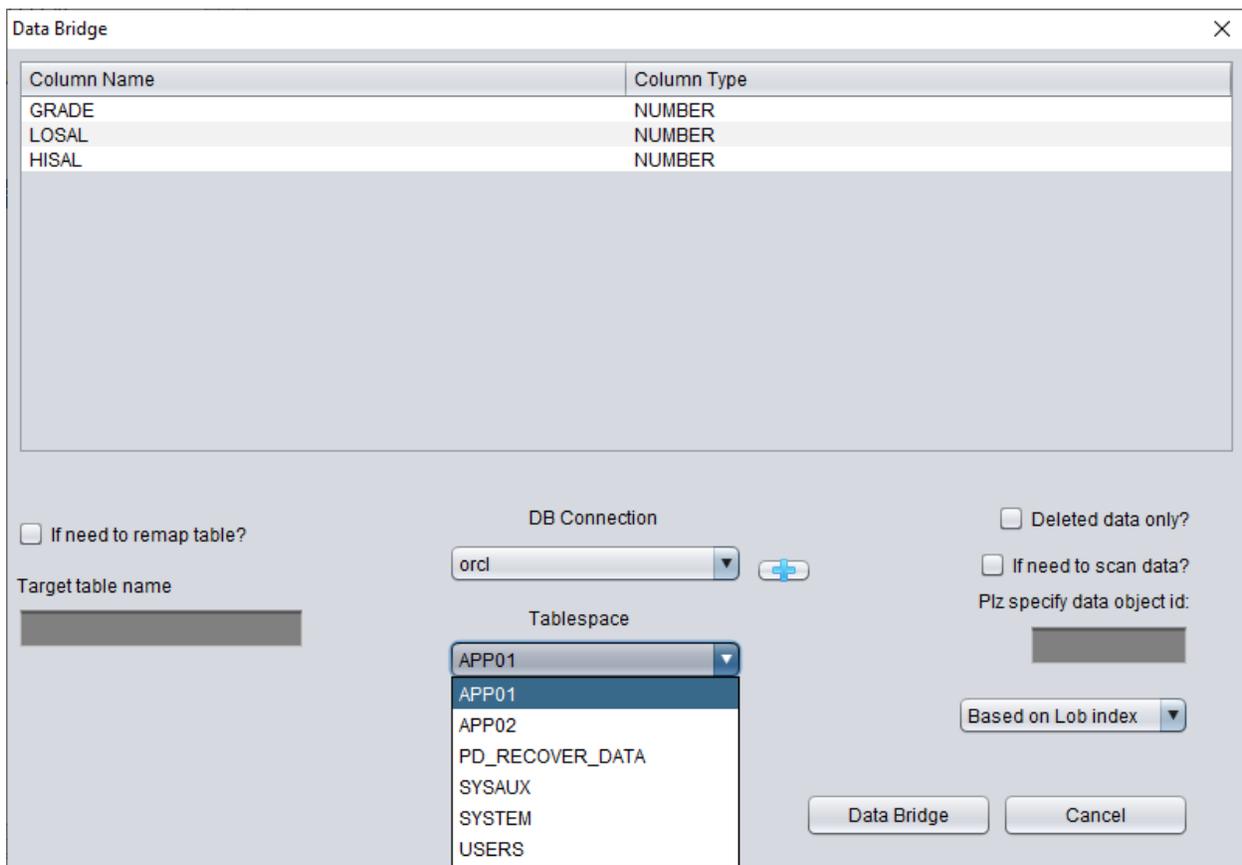
```
Tablespace created.
```



必要なログイン情報を作成します。注意：データベースユーザーには必要な権限が必要です（DBAロールを付与することを推奨）。



TESTに成功したらSAVEをクリックして保存します。



以上、TRUNCATEされたデータテーブルを格納するために選択されたテーブルスペースです。

Column Name	Column Type
GRADE	NUMBER
LOSAL	NUMBER
HISAL	NUMBER

If need to remap table?
Target table name:

DB Connection:
Tablespace:

Deleted data only?
 If need to scan data?
Plz specify data object id:

Based on Lob scan

ここでは、「if need to scan data」をチェックし、以前に取得した元のDATA_OBJECT_IDを入力する必要があります。そうすることで、DBRECOVERはそのIDに対応するデータを明確にスキャンします。

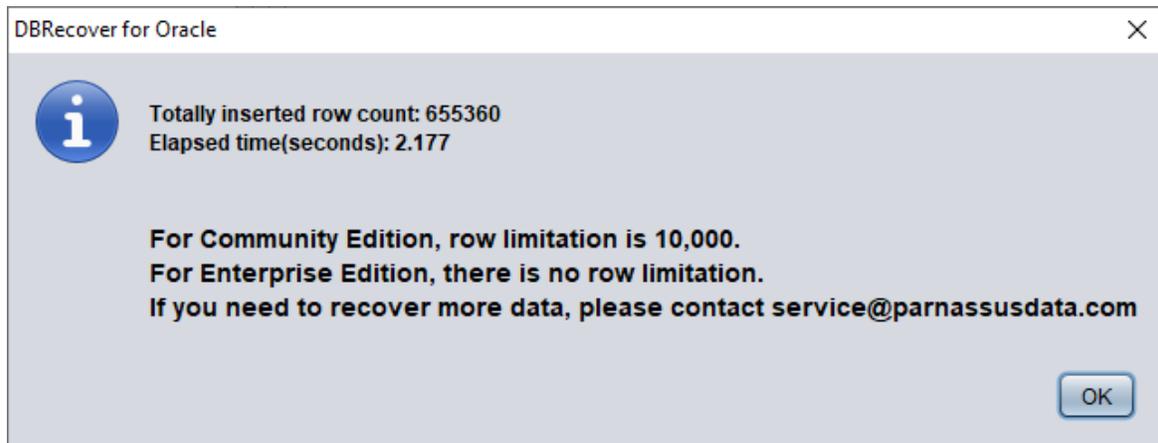
また、「if need to remap table」をチェックし、新しいテーブル名を入力します。これにより、データが新しいテーブル（新しいテーブルスペース下）に挿入され、データの上書きの可能性を排除します。

注意：

1) 目的のデータベースに既に対応するテーブル名が存在する場合、DBRECOVERは新しいテーブルを作成せず、既存のテーブル上に必要なデータを挿入します。テーブルがすでに作成されているため、指定されたテーブルスペースは無効になります。

2) 目的のデータベース-SCHEMAにまだ対応するテーブル名が存在しない場合、DBRECOVERは指定されたテーブルスペース上にテーブルを作成し、復元データを挿入しようとします。

上記のステップを完了した後、Data Bridgeボタンをクリックします。



復元された行数を確認します：

```
SQL> select count(*) from pd.salgrade_recover;
```

```
COUNT(*)
```

```
-----
```

```
655360
```

Truncateデータの大まかな原理は、Truncateが発生するとORACLEはデータディクショナリとSegment Header内でテーブルのData Object IDを更新しますが、実際のデータ部分のブロックは変更されません。データディクショナリとセグメントヘッダーのDATA_OBJECT_IDと、その後のデータブロック内のIDが一致しないため、ORACLEサービスプロセスは全表データを読み込む際に、TRUNCATEされたが実際にはまだ上書きされていないデータを読み取りません。した

がって、DBRECOVERは変更されていないデータ領域(Data Extent)を通じてデータを復元することができます。

回復シーン6 誤操作によるDropテーブルの回復

D社のアプリケーション開発者がバックアップなしでシステム内のコアアプリケーションテーブルをDROPしました。この場合、DBRECOVERをすぐに使用することで、DROPされたデータテーブルの大部分を回復することができます。10g以降ではrecyclebin回収ステーションの特性が提供されており、まずDBA_RECYCLEBINSビューを照会して、DROPされたテーブルが回収ステーションにあるかどうかを確認できます。もしそこにあれば、優先的に回収ステーションのflashback to before dropを使用します。回収ステーションにもない場合は、すぐにDBRECOVERを使用して回復します。

TRUNCATEの回復と同様に、DROPテーブルの回復には元のDATA_OBJECT_IDを特定する必要があります。

回復の簡単な流れは以下の通りです：

1. DROPされたデータテーブルがある表空間を読み取り専用モードに設定します（ALTER TABLESPACE {TABLESPACE_NAME} READ ONLY; または、最初にテーブルスペースのすべてのデータファイルをコピーします）。
2. データディクショナリまたはLOGMINERを照会して、DROPされたデータテーブルのDATA_OBJECT_IDを見つけます。
3. NON-DICT非辞書モードでDBRECOVERを起動し、DROPされたデータテーブルがある表空間のすべてのデータファイルを追加した後、SCAN DATABASE+SCAN TABLE from Extent MAPを実行します。
4. DATA_OBJECT_IDを使用して展開されたオブジェクトツリーの中で対応するデータテーブルを特定し、Data Bridgeモードを使用して元のデータベースに戻します。

LOGMINERを使用するとDATA_OBJECT_IDの概要を取得できます。LOGMINERを使用する場合の大まかなスクリプトは以下の通りです：

```
EXECUTE DBMS_LOGMNR.ADD_LOGFILE( LOGFILENAME => '/oracle/logs/log1.f', OPTIONS => DBMS_LOGMNR.NEW);  
EXECUTE DBMS_LOGMNR.ADD_LOGFILE( LOGFILENAME => '/oracle/logs/log2.f', OPTIONS =>  
DBMS_LOGMNR.ADDFILE);  
  
Execute  
DBMS_LOGMNR.START_LOGMNR(DBMS_LOGMNR.DICT_FROM_ONLINE_CATALOG+DBMS_LOGMNR.COMMITTED_DATA_ONLY);
```

```
SELECT * FROM V$LOGMNR_CONTENTS ;  
EXECUTE DBMS_LOGMNR.END_LOGMNR;
```

AWRデータを掘り下げてDATA_OBJECT_IDを取得することも試みるすることができます：

```
Select * from  
(select object_name,object# from DBA_HIST_SQL_PLAN  
UNION select object_name,object# from GV$SQL_PLAN) V1 where V1.OBJECT# IS  
NOT NULL minus select name,obj# from sys.obj$;
```

```
select obj#,dataobj#, object_name from WRH$_SEG_STAT_OBJ where object_name  
not in (select name from sys.obj$) order by object_name desc;
```

```
SELECT tab1.SQL_ID,  
current_obj#,  
tab2.sql_text  
FROM DBA_HIST_ACTIVE_SESS_HISTORY tab1,  
dba_hist_sqltext tab2  
WHERE tab1.current_obj# NOT IN  
(SELECT obj# FROM sys.obj$  
)  
AND current_obj#!=-1  
AND tab1.sql_id =tab2.sql_id(+);
```

上記の3つのクエリは、AWRデータとOBJ\$ディクショナリ基本テーブルを比較して、DROPされたテーブルを見つけます。

実際にデモンストレーションしてみましょう：

```
SQL> create table dropit as select * from dba_objects;  
Table created.
```

```
SQL> select count(*) from pd.dropit;
COUNT(*)
-----
73095

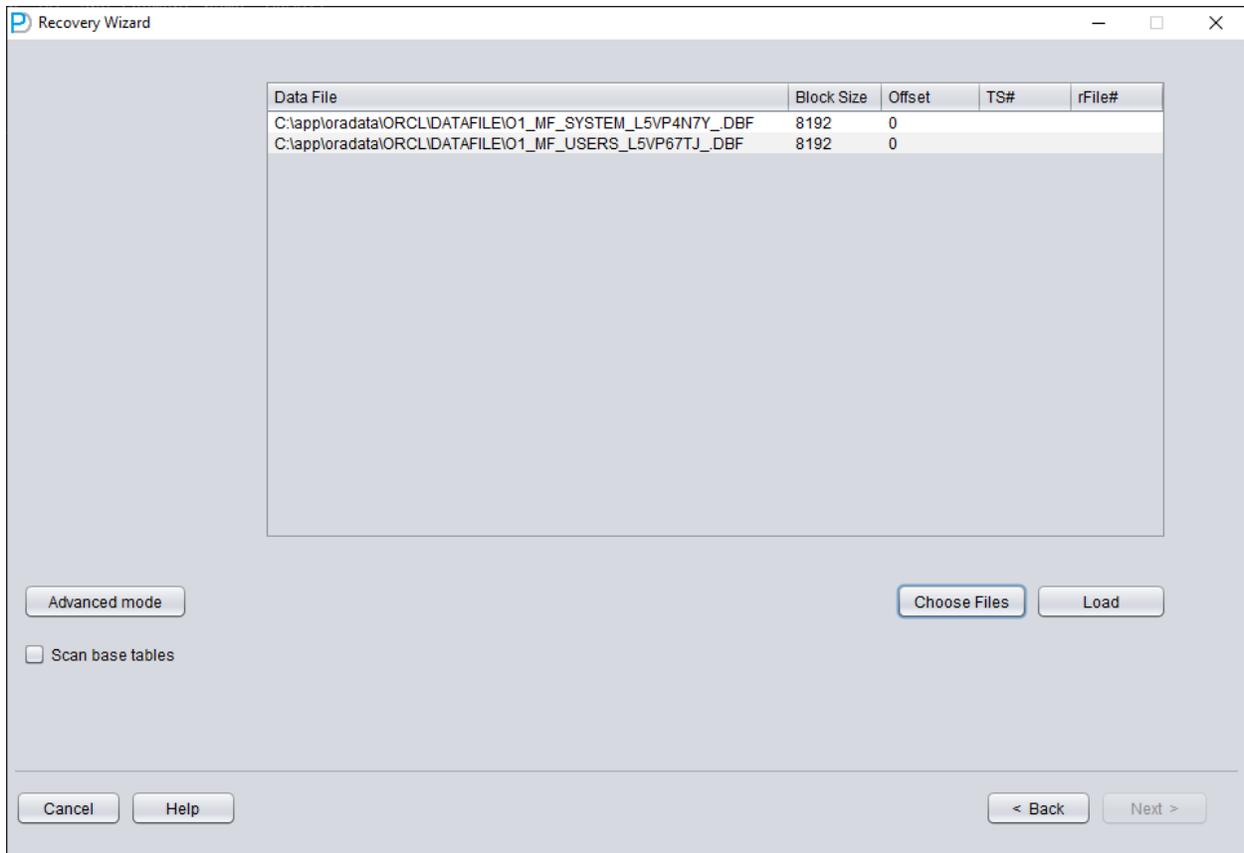
SQL> select tablespace_name from dba_segments where owner='PD' and segment_name='DROPIT';
TABLESPACE_NAME
-----
USERS

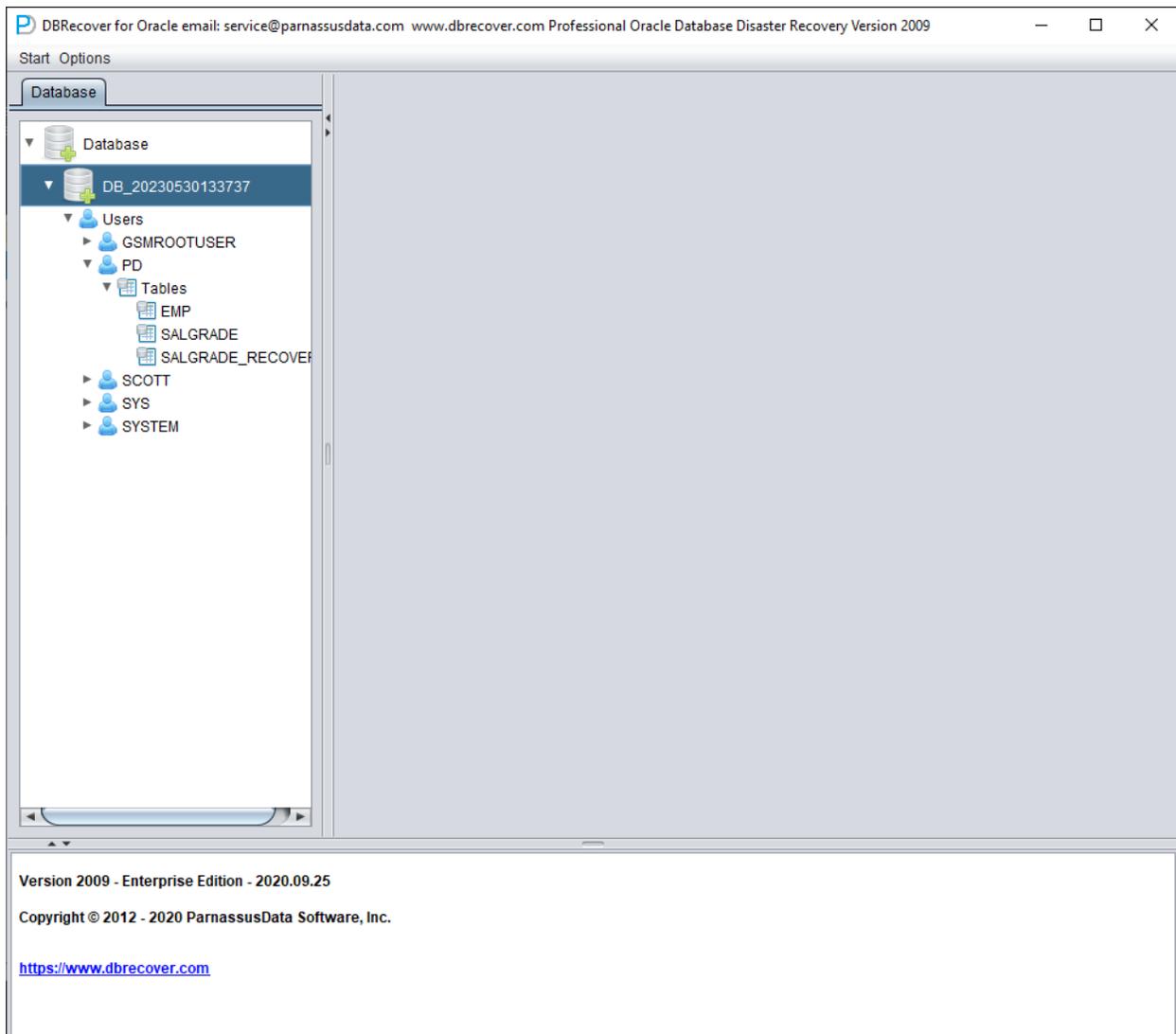
SQL> select object_id ,data_object_id from dba_objects where owner='PD' and object_name='DROPIT';
OBJECT_ID DATA_OBJECT_ID
-----
76116 76116

SQL> drop table dropit;
Table dropped.

SQL> alter system checkpoint;
System altered.
```

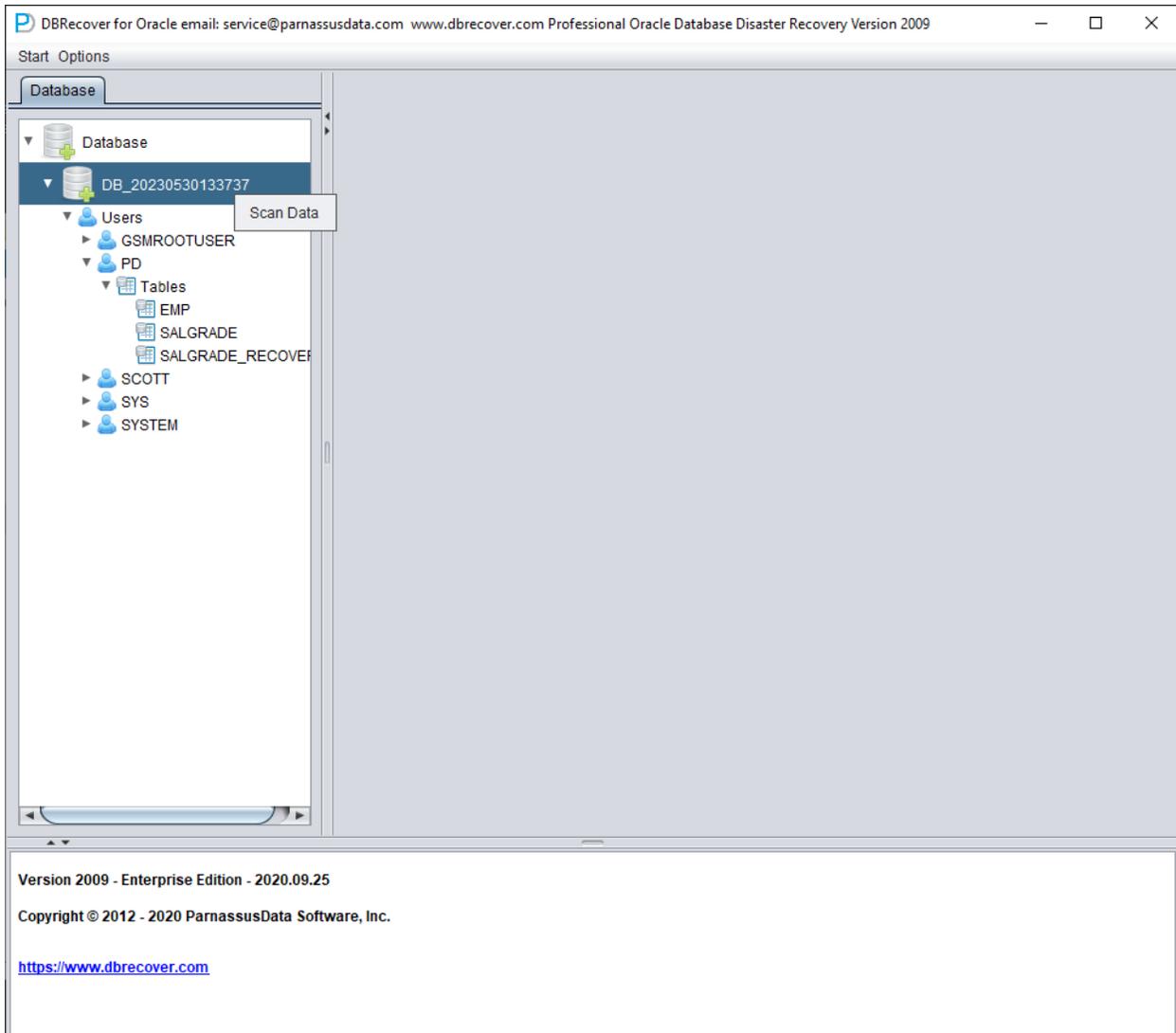
DBRECOVERを辞書モード(DICTIONARY-MODE)で起動し、SYSTEM01.DBFとテーブルがあるUSERSテーブルスペースのみを追加します：

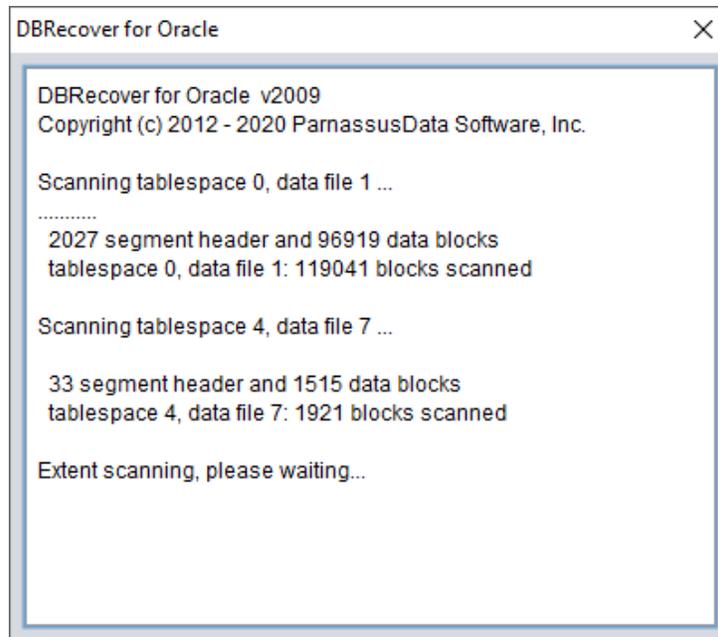


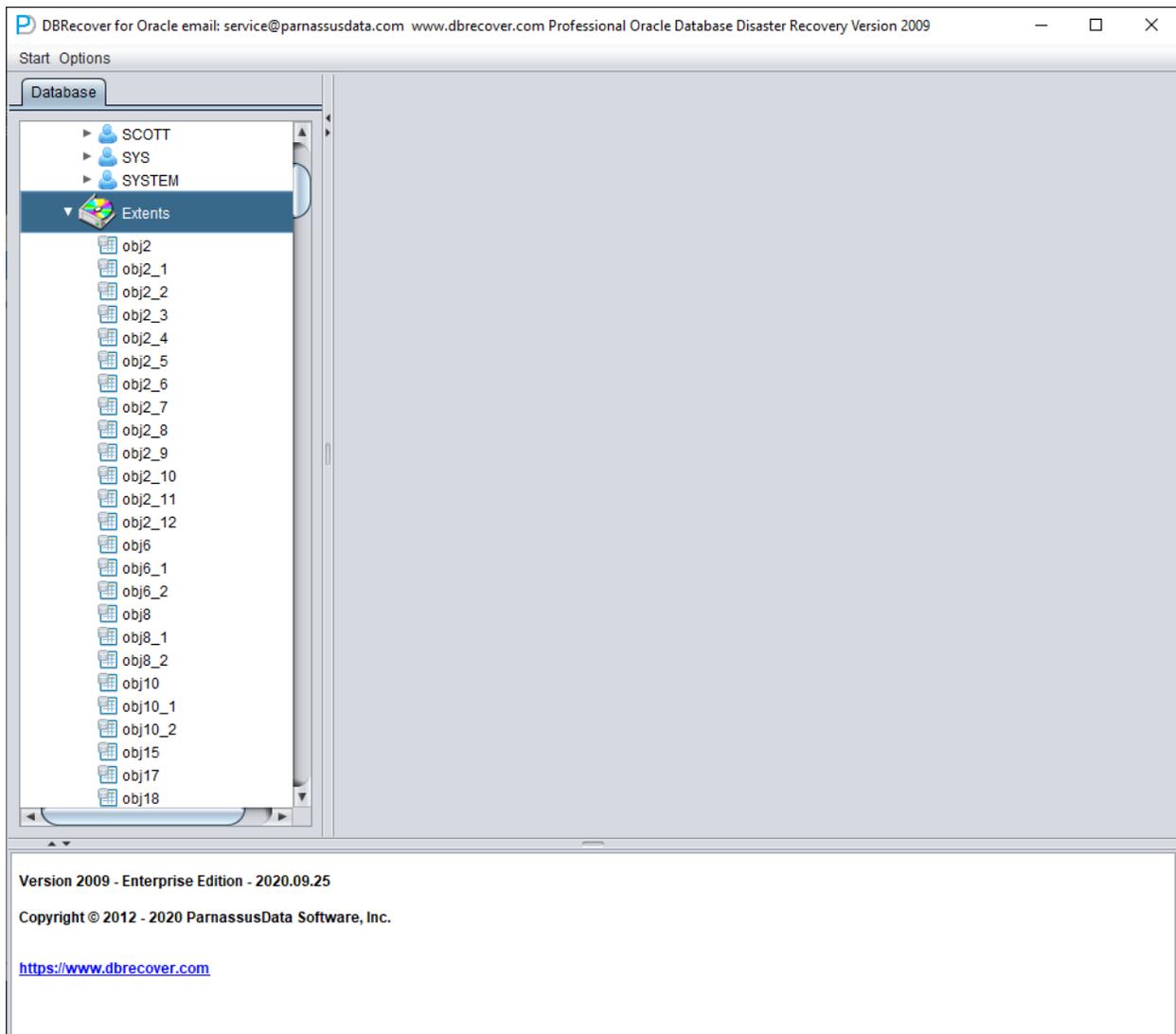


ロードが完了すると、復元する必要があるテーブルはPD SCHEMAには存在しないことが分かります。これは正常です。

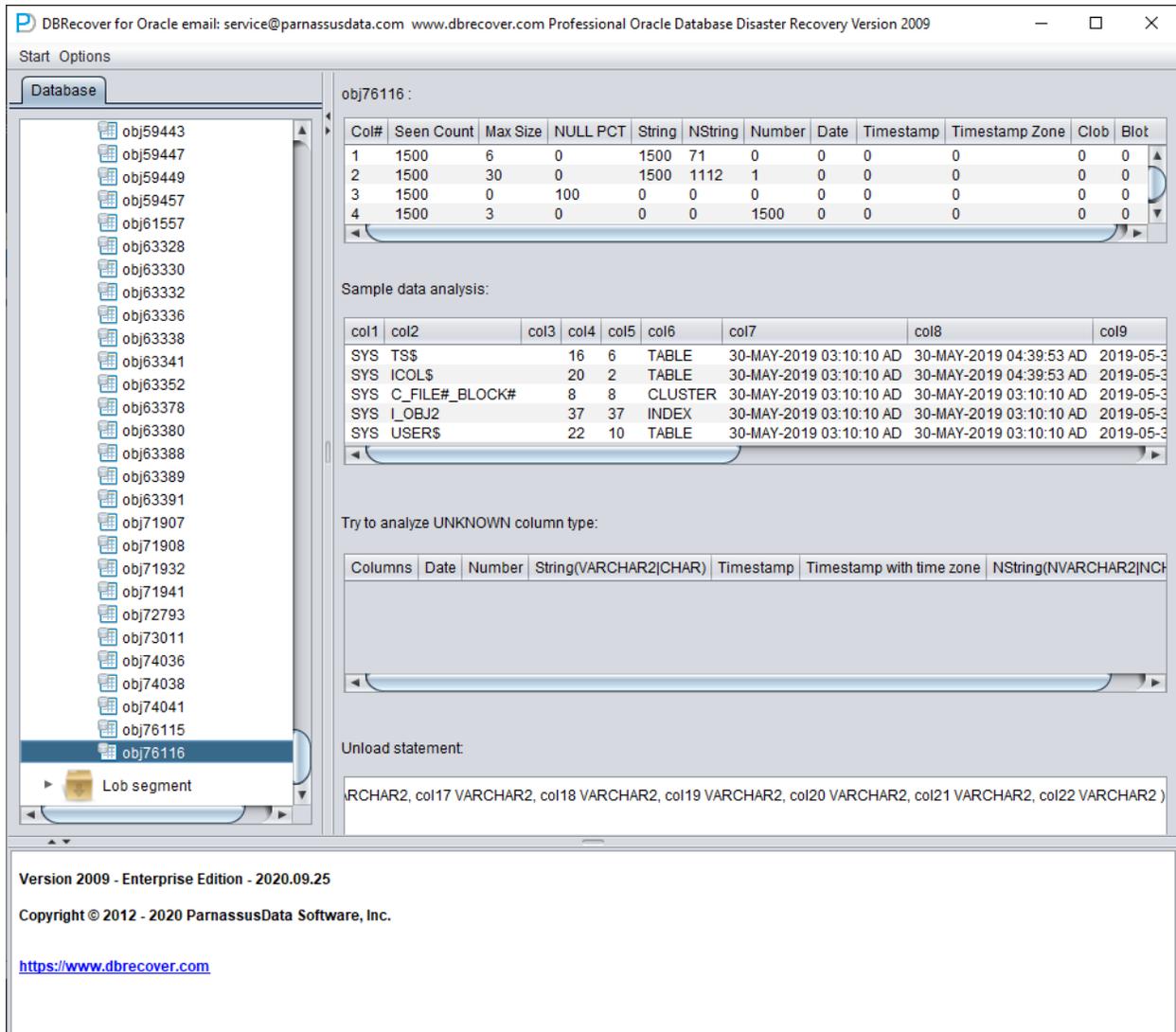
データベースノードを選択し、右クリックしてSCAN Dataを実行します。







その後、EXTENTSノードが表示され、OBJ76116ノードを探します：



その後、Data Bridgeの特性を利用して元のデータベースに挿入します。

回復シナリオ7 誤操作によるDROP TABLESPACEの回復

D社の従業員が不要な表空間を削除するためにDROP TABLESPACE INCLUDING CONTENTS操作を実行しましたが、操作後に開発部門から、DROPされたTABLESPACEに重要なSCHEMAの

データがあったと報告されました。現在、表空間はDROPされており、バックアップもありません。

この場合、DBRECOVERの非辞書モード(NON-DICTIONARY)を使用して、DROPされたTABLESPACEのすべてのデータファイルからデータを抽出することができます。この方法で大部分のデータを回復することができますが、非辞書モードのため、回復されたテーブルとアプリケーションデータテーブルを一つずつ対応付ける必要があります。この際、通常はアプリケーション開発メンテナンススタッフが介入し、手動でどのデータがどのテーブルに属するかを識別します。DROP TABLESPACE操作はデータディクショナリを変更し、OBJ\$から対応する表空間のオブジェクトを削除するため、OBJ\$からDATA_OBJECT_IDとOBJECT_NAMEの対応関係を得ることはできません。この場合、DROP TABLEシナリオで紹介された方法を使用して、DATA_OBJECT_IDとOBJECT_NAMEの対応関係をできるだけ多く取得します。

その大まかな流れは以下の通りです：

DROP TABLESPACE時にデータファイルも物理的に削除された場合は、まずデータファイルを回復する必要があります。ファイルシステムレベルの回復ソフトウェアを試したり、PRMSCANソフトウェアを使用してORACLEデータブロックレベルでスキャンし、データファイルを再構築することができます。

PRMSCANはORACLEデータブロック断片スキャンおよび結合ツールで、以下のシナリオに適用されます：

1. 誤ってファイルシステム（任意のファイルシステム NTFS、FAT、EXT、UFS、JFSなど）またはASM上のデータファイルを手動で削除した。
2. ファイルシステムが損傷し、データファイルのサイズが0バイトになり、データファイルがクリアされた。
3. ファイルシステムが損傷し、ファイルシステムをMOUNTできなくなった。
4. ASMストレージメタデータが損傷し、diskgroupをmountできなくなった。
5. ファイルシステムまたはASMのLVまたはPVが物理的に破損したり失われたりした。
6. 上記のシナリオでは、prmscanを使用してファイルシステムまたはASMのPV、LVに存在する未覆盖のoracle blockを直接スキャンし、これらのoracleデータブロックを結合再構築することでデータ回復を実現できます。

PRMSCANはJAVA言語で開発されており、JDK 1.6以降をサポートするすべてのオペレーティングシステム、Windows、Linux、Solaris、AIX、HP-UXで使用できます。

現在、この製品は小売りされておらず、回復サービスの提供については私たちに連絡することができます。

例えば、以下の例では/dev/sdb1がext4ファイルシステムのパーティションで、ext4ファイルシステムの損傷によりSDB1はMOUNTできませんが、そのファイルシステムには一連のoracleデータベースのデータファイルが保存されています。ファイルシステムをMOUNTできない場合、oracleデータベースも使用できません。

ここでは、prmscanのoracleデータファイルブロックのスキャンと結合機能を使用して、損傷したファイルシステムから直接データファイルを再構築します。

ディスク全体をスキャン：

```
[oracle@dbdao01 ~]$ java -jar PRMScan.jar -scan /dev/sdb1 -guess 8k
```

-scan オプションは/dev/sdb1デバイスをスキャンし、Oracle blocksizeを8kとして指定します。

```
[oracle@dbdao01 ~]$ java -jar PRMScan.jar -outputsh ./8kfull.txt
```

-outputsh はスキャンされた情報を結合するSHELLファイルを出力することを意味し、ここでは8kfull.txtです。

```
[oracle@dbdao01 ~]$ sh 8kfull.txt
```

8kfull.txtを実行すると、必要なすべてのデータファイルを現在のディレクトリに生成します。

如下

```
[oracle@dbdao01 ~]$ ls -ll PD*
```

```
rw-r--r-- 1 oracle oinstall 295428096 Jul 28 00:37 PD_DBF1.dbf
```

```
rw-r--r-- 1 oracle oinstall 83427328 Jul 28 00:37 PD_DBF2.dbf
```

```
rw-r--r-- 1 oracle oinstall 220266496 Jul 28 00:37 PD_DBF3.dbf
```

```
rw-r--r-- 1 oracle oinstall 1324482560 Jul 28 00:38 PD_DBF4.dbf
```

データファイルが物理的に削除されていない場合は、非辞書モード(NON-DICTIONARY MODE)でDBRECOVERに追加し、その中のデータをスキャンできます。

後続のステップは上記のDROP TABLEの操作を参照できますが、DROP TABLESPACEの回復対象は多くのテーブルになります。