

# SAP Exodusの御紹介

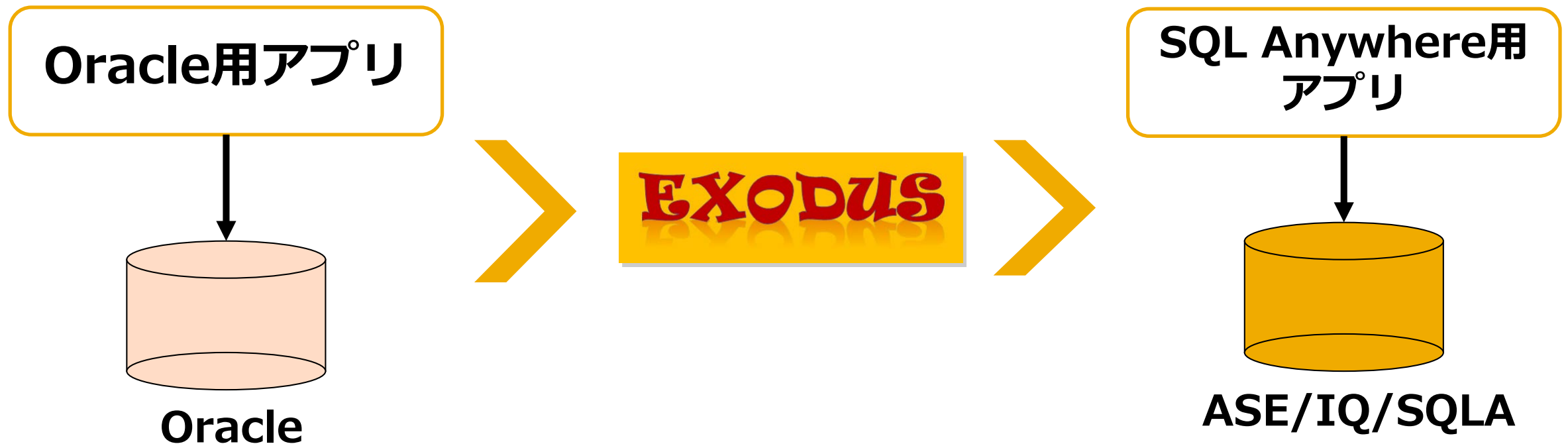
PartnerEdge パートナー様にのみ提供されるデータベースアプリケーション移行アシストツール



**EXODUS**

# What is Exodus ?

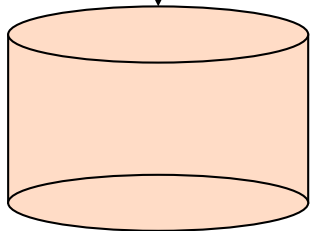
SAP ExodusはOracleやDB2、SQL Server用に開発されたデータベースアプリケーションをASE、IQ、SQL Anywhere用への変換をアシストするツールです。



# Why Exodus ?

WindowsPCで稼働するアプリを想定

Oracle向けに記述したアプリ



Oracle DB

対応・変換・改造

## アプリ

ロジック部分は変更不要  
DB周りの変更が必要。

## 接続API

汎用規格 (ODBC/JDBC/.NET  
等) であれば変更量少

## SQL

ANSI規格に従っていれば共通だ  
が独自構文の場合もある。

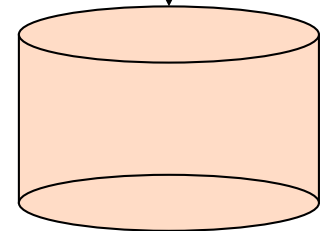
## DB構造

何らかの互換性のある形で移行できる  
場合が殆ど。ツールも豊富。

## DB内ロジック (プロシジャ等)

独自言語、長文

SQL Anywhere向けに記述したアプリ



SQL Anywhere DB

出来なくはないが、面倒

# 移行するデータベースアプリケーションの種類

## SAP アプリケーション (例 : SAP Business Suite)

- SAP Rapid Deployment Servicesを使用することで移行できます。

## ORMapperのような中間層のフレームワークを使用した非SAPのカスタムアプリケーション (例 : Hibernate)

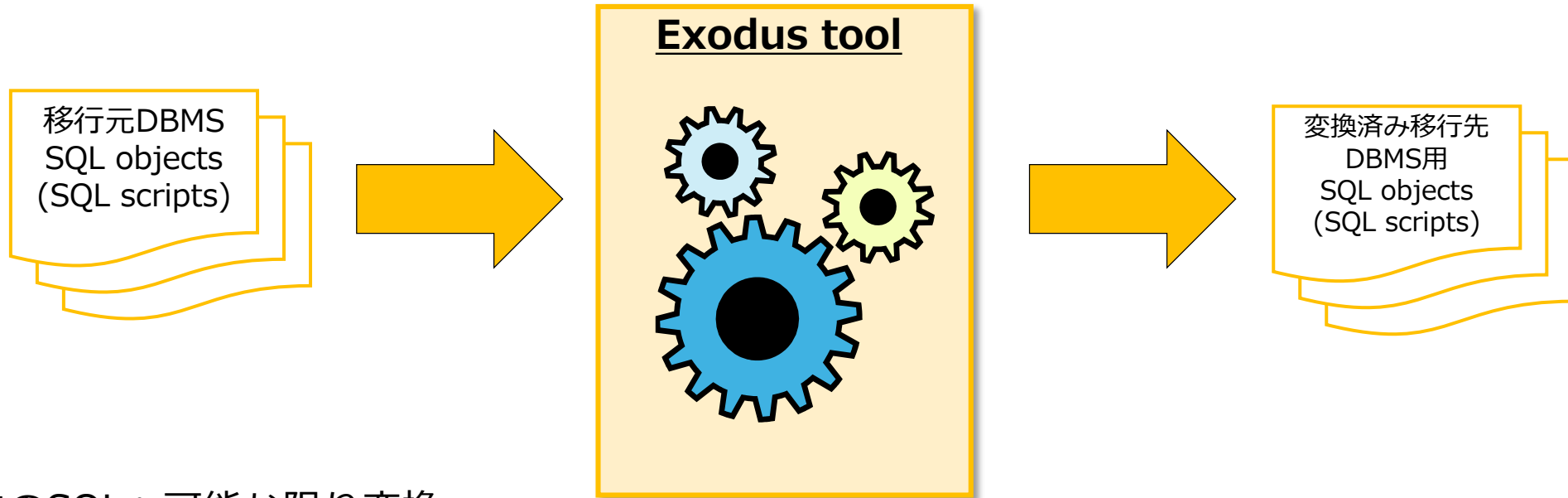
- そのフレームワークが移行先のDBをサポートしていれば簡単。
- この種のアプリはストアドプロシジャやユーザー定義関数等を使用していることは少ない。

## CやJava、.NET等で開発されたカスタムアプリケーション

- アーキテクチャやSQLコード、API、DBMS毎の独自機能の使用など多岐に渡る。
- この種のアプリはストアドプロシジャやユーザー定義関数等を使用していることが多い
- サードベンダーライブラリを使用している場合は注意

Exodusが対象とするもの

# Exodusによる変換



移行先のSQLへ可能な限り変換:

-- example: Oracle  
SELECT SYSDATE from DUAL;

いくつかの関数では代用関数を提供:

-- example: Oracle  
SELECT INITCAP(msg) from MyTable;

-- example: SAP SQL Anywhere  
SELECT CURRENT TIMESTAMP

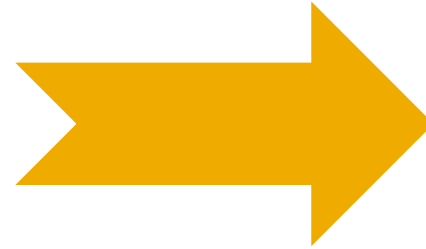
-- example: SAP SQL Anywhere  
SELECT  
dbo.sp\_f\_dbmtk\_capitalize\_word(msg)  
from MyTable

※OracleのINITCAP関数に相当するものはSQL Anywhereに存在しないので、代用となる関数をExodusが提供し、それで置換する

## オリジナル : Oracle PL/SQL

```
create procedure ml_set_listening(
  p_name in varchar2,
  p_device in varchar2,
  p_listening in varchar2,
  p_ignore_tracking in varchar2,
  p_source in varchar2 )
as
  d varchar2( 255 );
begin
  begin
  select device_name into d from ml_listening where name = p_name;
  exception
    when NO_DATA_FOUND then
      d := NULL;
  end;
  if d is null then
insert into ml_listening( name, device_name, listening, ignore_tracking, source )
  values( p_name, p_device, p_listening, p_ignore_tracking, p_source );
  else
    if p_source = 'tracking' then
      update ml_listening
      set device_name = p_device,
        listening = p_listening,
        ignore_tracking = p_ignore_tracking,
        source = p_source
      where name = p_name and ignore_tracking = 'n';
    else
      update ml_listening
      set device_name = p_device,
        listening = p_listening,
        ignore_tracking = p_ignore_tracking,
        source = p_source
      where name = p_name;
    end if;
  end if;
end;
```

**注 : 変換後のスクリプトには変換前の行がコメントアウトした状態で含まれます。右記は変換前のコメントアウトした行は削除してあります。**



## Exodus変換後 : SQL Anywhere WATCOM SQL

```
CREATE OR REPLACE PROCEDURE mluser1.ml_set_listening
(
  IN p_name VARCHAR(75) ,
  IN p_device VARCHAR(75) ,
  IN p_listening VARCHAR(75) ,
  IN p_ignore_tracking VARCHAR(75) ,
  IN p_source VARCHAR(75)
)
SQL SECURITY DEFINER
BEGIN
  DECLARE DBMTK_CALLER_ID INT; SET DBMTK_CALLER_ID = @@PROCID;

  DECLARE d VARCHAR(255);
  BEGIN
    SELECT
      device_name
    INTO
      d
    FROM
      ml_listening
    WHERE name = p_name;

    EXCEPTION /* RESOLVE: EXCEPTION handler Details: Manual adjustments required to convert to SA
    EXCEPTION syntax; see Exodus User Guide. */
    WHEN NO_DATA_FOUND THEN /* RESOLVE: Manually add DECLARE...EXCEPTION (SA syntax)
    for ORACLE 'NO_DATA_FOUND' condition (see SA documentation for corresponding SQLSTATE value) */
      SET d = NULL;
    END;

    IF d IS NULL
    THEN
      INSERT INTO ml_listening ( name, device_name, listening, ignore_tracking, source )
      VALUES( p_name, p_device, p_listening, p_ignore_tracking, p_source );
    ELSE

      IF p_source = 'tracking'
      THEN
        UPDATE ml_listening
        SET device_name = p_device,
          listening = p_listening,
          ignore_tracking = p_ignore_tracking,
          source = p_source
        WHERE name = p_name
        AND ignore_tracking = 'n';
      ELSE

        UPDATE ml_listening
        SET device_name = p_device,
          listening = p_listening,
          ignore_tracking = p_ignore_tracking,
          source = p_source
        WHERE name = p_name;
      ENDIF;
    ENDIF;
  END;
```

# 移行コスト見積もり

移行元DBMSに接続し、コスト見積もり機能を使用することで、移行対象のオブジェクト数、Exodusで変換可能なオブジェクト数、変換できないオブジェクト数等を見積もり、移行にかかるコストの超概算見積もりが可能です。

----- Schema-based Migration Cost Estimate -----

```
Estimated migration cost for schema:   49 days   Cost breakdown(1 day=8 hrs)
Schema Identifiers as Reserved Words : 235           1 day      (15m/unit)
Regular Tables : 1428                               36 days    (4h/unit)
Nested Tables : 1                                   4 hours    (4h/unit)
Object Tables : 2                                  1 day      (4h/unit)
Index Types : 5                                    25 minutes (5m/unit)
Views : 54                                          4 hours    (10m/unit)
Columns with Oracle BFILE datatype : 1              1 day      (1d/unit)
Columns with Oracle INTERVAL/TIME ZONE datatype : 12 4 days     (3h/unit)
Composite User-Defined Datatypes : 21              2 days     (1h/unit)
Collection User-Defined Datatypes : 7              2 days     (2h/unit)
Local Synonyms : 48                                2 hours    (5m/unit)
Remote Synonyms : 2                                30 minutes (15m/unit)
Other Object Types : 3                             6 hours    (2h/unit)
```

「何にどのくらいかかるか」（例：1つのテーブルの移行にどのくらいの工数を要するか）はExodus上で事前定義が必要です。

----- SQL-based Migration Cost Estimate -----

Total estimated migration cost for all 755 files: 51 days (1 day=8 hrs)

Conversion statistics for all files: Cost breakdown(1 day=8 hrs)

Datatype length exceeds target DBMS limit : 4	20 minutes (5m/unit)
Syntax construct not supported in SQLA : 1	15 minutes (15m/unit)
Feature not suppt'd in SQLA SQL Functions : 1276	9 days (1h/unit)
Delimited identifiers in SQLA : 469	5 hours (5m/unit)
Performance feature not supported : 29	5 hours (15m/unit)
Review semantics in source/target DBMS : 2856	3 days (15m/unit)
Oracle Standard Package call not suppt'd : 36	3 days (1h/unit)
Mismatch formal/actual parameters : 481	1 day (10m/unit)
Mismatch nr of variables/expressions : 24	1 hour (5m/unit)
Owner/Scope of identifier unresolved : 568	4 days (30m/unit)
Identifier not found : 956	4 days (30m/unit)
%TYPE declaration : 4949	0 Fully converted
Autonomous transaction : 2	4 hours (2h/unit)
ANSI outer join syntax recommended : 257	1 day (10m/unit)
Dynamically generated SQL : 178	3 hours (5m/unit)
Database link : 15	3 hours (15m/unit)

[...]



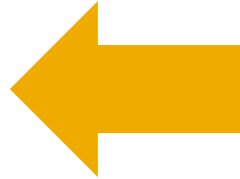
Package declaration : 194	0	Fully converted
DBMS_OUTPUT.PUT_LINE : 5178	0	Fully converted
Execute Immediate : 178	0	Fully converted
Standard Package call : 16	0	Fully converted
Procedure call : 3527	0	Fully converted
NULL statement : 209	0	Fully converted
Oracle hint : 407	0	Fully converted
Oracle pragma : 5	1 hour	(15m/unit)
Additional manual conversion required		
SELECT-INTO : 2027	0	Fully converted
Built-in function : 6446	0	Fully converted
Procedure declaration : 822	0	Fully converted
Function declaration : 1821	0	Fully converted
Variable declaration : 7804	0	Fully converted
Variable assignment : 36123	0	Fully converted
RETURN : 7092	0	Fully converted
END of block : 10924	0	Fully converted
IF-THEN : 5584	0	Fully converted
ELSE (IF-THEN) : 1150	0	Fully converted
ELSIF (IF-THEN) : 310	0	Fully converted
[...]		

SYSDATE,SYSTIMESTAMP,etc. : 5342	0	Fully converted
SQLCODE,SQLERRM : 2856	0	Fully converted
RAISE_APPLICATION_ERROR : 805	0	Fully converted
Boolean constant : 595	0	Fully converted
Boolean condition : 276	0	Fully converted
SQL Function call : 3776	0	Fully converted
DUAL table : 389	0	Fully converted
Outer join : 255	1 day	(10m/unit)
Manual conversion recommended		
WHILE loop : 58	0	Fully converted
Loop control : 379	0	Fully converted
EXCEPTION handler : 6230	23 days	(2h/unit)
Manual conversion to ASE required		
Sequence CURRVAL,NEXTVAL : 304	0	(0/unit)
Transaction COMMIT : 1366	0	Fully converted
Transaction ROLLBACK : 2176	0	Fully converted
Transaction SAVEPOINT : 10	0	Fully converted
MINUS operator : 13	6 hours	(30m/unit)
Manual conversion required		
INSERT statement : 1112	0	Fully converted
[...]		

```
UPDATE statement : 953          0          Fully converted
DELETE statement : 565          0          Fully converted
SELECT statement : 2858         0          Fully converted
```

----- Total Migration Cost Estimate -----

Based on SQL conversion : 51 days



**移行工数は51人日と見積もり**

「何にどのくらいかかるか」（例：1つのテーブルの移行にどのくらいの工数を要するか）はExodus上で事前定義が必要です。

----- Conversion totals over all 755 files: -----

Totals: 212627 lines processed (in 755 files) in 4040 seconds (52 lines/sec)

Conversion coverage: 84%

Total issues found, still to be resolved: 10794 (in 18 categories)



**(移行対象) DBオブジェクトのうち84%がExodusで変換可能**

# Exodusのサポート状況

- **Exodusは現在以下をサポートしています。**
  - 動作環境：Windows 7~10
    - Cドライブに5GB以上の空きスペース
    - 上記条件で動作するが、開発としては快適な動作のために以下のスペック以上のHWを推奨
      - マルチコアCPU（2コア以上）
      - 8GB以上のRAM
    - Cygwin / JRE 環境が必要です。
  - 移行元DBMS：Oracle, Microsoft SQL Server and IBM DB2 UDB
  - 移行先DBMS：SAP Adaptive Server Enterprise, SAP IQ, SAP SQL Anywhere (Sybase ASE/Sybase IQ/SQL Anywhere)
  - Oracle/DB2からの変換精度は80%~90%、SQL Serverからは90~95%
  - 移行先DBMSとしてのSAP HANA、及び移行元としての他DBMSのサポートは現在計画中/開発中です。

# Exodusの注意点

## SAP Exodus DBMS migration toolは

- SAPの“商品”ではありません。
- SAPパートナー契約 (PartnerEdge)が必要です。  
(申請の際にPartnerEdge IDが必要です。  
規定のWebcastなどの受講と追加契約が必要です。)
- 無料です。
- 全てのSQL/SQLオブジェクトの移行を保証するものではありません。
- 移行の全てを行えるものではありません。
  - スキーマ移行には対応していません。(SAP PowerDesignerを御使用下さい。)
  - データ移行には対応していません。(ETLツールをご利用下さい。)
  - クライアントアプリケーションや管理アプリケーションの移行はサポートしていません。
- アプリケーション開発ライフサイクル管理ツール、その機能はありません。



} **SQL Anywhereは  
SQL Centralで可能**



# ご清聴ありがとうございました。

問い合わせ先:

境 直人

SAPジャパン株式会社  
ソリューション統括本部  
デジタル・ビジネス・プラットフォーム部  
シニア・ソリューション・スペシャリスト

[naoto.sakai@sap.com](mailto:naoto.sakai@sap.com)