

Oracle Direct Seminar



ORACLE®

Oracle Databaseによる効率的なXMLアプリ開発

日本オラクル株式会社

Oracle Direct



Agenda

1. はじめに
2. XML処理を実行するのはAP層？DB層？
3. 全体最適を考える(システム全体で必要となるハードウェア・リソースを最小化するには?)
4. 開発工数(本質ではない「開発」をいかに減らすか)
5. 運用・管理コスト
6. XML DB=Oracle Database であること、その価値

はじめに

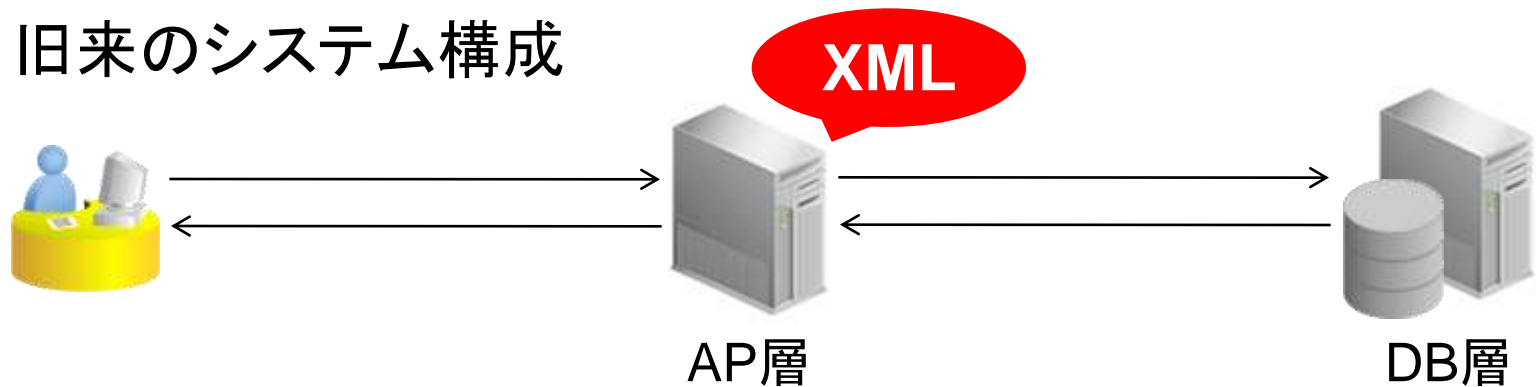
- 本資料の内容は、Oracle Database 11g Release 2 (11.2.0.2)の動作仕様をベースに記述されています
- 本資料では、アプリケーション層を「AP層」、データベース層を「DB層」と、それぞれ略記します

XML処理

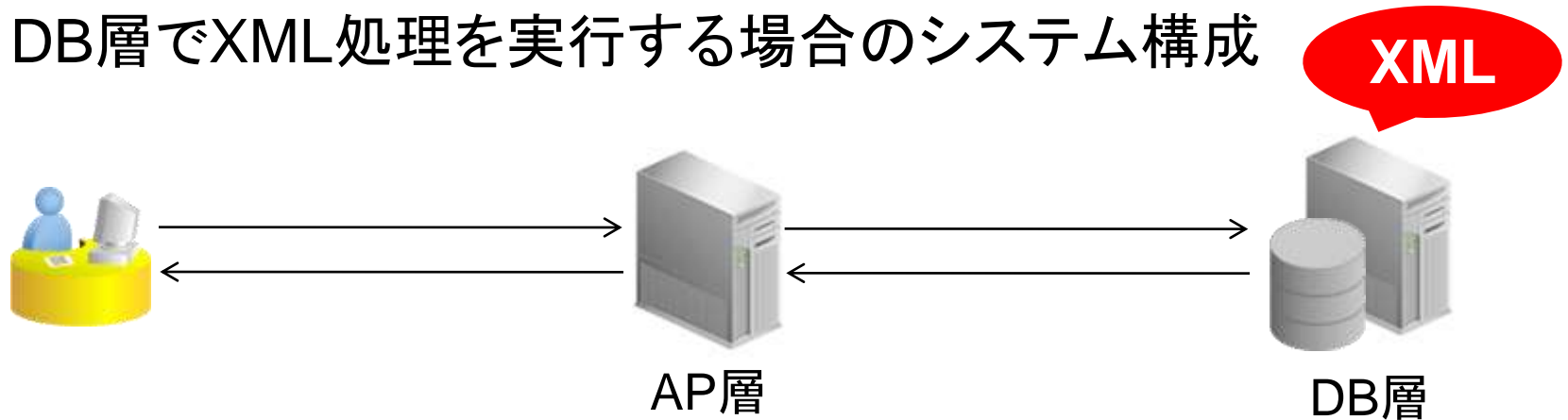
- 定義
 - この資料で言う「XML処理」とは、次の操作を含みます
 - XML文書のツリー構造を理解する
 - XPath の適用や XQuery の実行を行う
 - XML文書にXSLTスタイルシートを適用する
 - リレーショナル・データから、XML文書を生成する

XML処理を実行するのはAP層？ DB層？

- 旧来のシステム構成



- DB層でXML処理を実行する場合のシステム構成



次の観点から考察

- システム全体で必要となるハードウェア・リソース
 - CPU
 - メモリ
 - ハードディスク
 - ネットワーク
- 開発工数
- 運用・管理コスト
- 信頼性・汎用性・多機能性

DB層でのXML処理は非標準？

- DB層でのXML処理は、SQL標準 (ISO/IEC 9075-14:2003～) の中で定義された標準的な方法です
- SQL標準で定義されているXML処理は、以下を含みます
 - XMLデータを格納するためのデータ型
 - XML文書のツリー構造を理解する
 - XPath の適用や XQuery の実行を行う
 - リレーショナル・データから、XML文書を生成する

前提知識



Oracle XML DB とは？

- Oracle Database カーネルで実装された、XML文書の格納／問合せ機能
 - Oracle XML DB の機能のほとんどは、C言語を利用して実装されている
- Oracle9i Database Release 2(9.2) (2002年8月にリリース)以降で利用可能
- Oracle Database の全てのエディションで利用可能(※1)

※1 Oracle XML DB は、Oracle Database の全てのエディションで利用可能な機能です。Oracle XML DB を利用するために、追加のオプション・ライセンスは必要ありません。ただし、本セミナーで紹介する一部機能を利用するには、Enterprise Edition のライセンス、およびオプション・ライセンスが必要です。例えば、本資料で紹介する「SecureFile圧縮」を利用するには、Enterprise Edition のライセンスと、Advanced Compression オプションのライセンスの両方が必要です。

Oracle Database と XML 文書

- Oracle Database に XML 文書を格納するには？
 - VARCHAR2 型を用いる
 - CLOB 型を用いる
 - **XMLType** 型を用いる

XMLType 型の簡単な使い方

- XMLType コンストラクタ
 - VARCHAR2、CLOB、BLOB、BFILE 型からXMLType型への変換を行う(=XMLTypeインスタンスを作成する)
- 使用例

```
CREATE TABLE testxml (xml XMLTYPE);  
INSERT INTO testxml VALUES (XMLTYPE (' <a>dummy</a>' ));  
COMMIT;
```



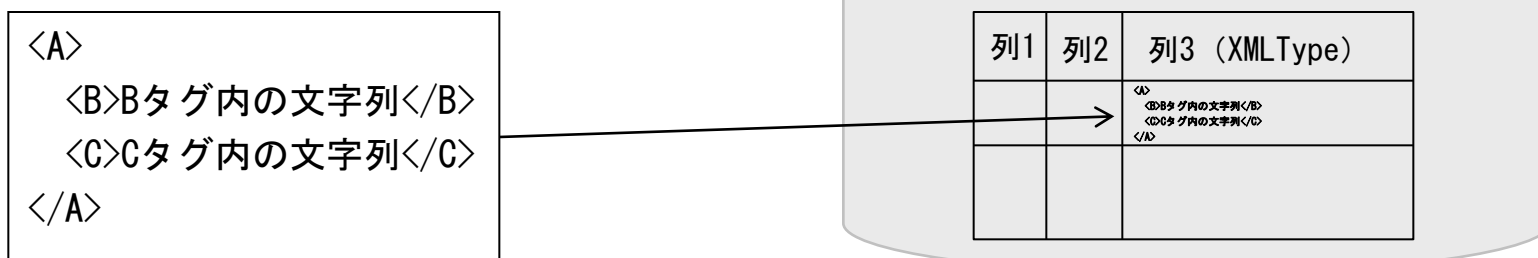
```
SQL> SELECT xml FROM testxml;  
XML  
-----  
<a>dummy</a>
```

XMLType 型の物理的な格納方式

- 非構造化記憶域
 - 内部的にはCLOB型を利用
- バイナリXML記憶域 (Oracle Database 11g R1～)
 - 内部的にはBLOB型を利用
- 構造化記憶域 (Oracle9i Database R2～)
 - 内部的にはオブジェクト型を利用
 - XMLスキーマの登録により、Oracle Database が自動的に、XML文書のタグ／属性を、オブジェクト型の属性にマッピングする
 - このマッピングを、オブジェクト・リレーショナル・マッピング (以下、O/Rマッピング) と呼ぶ

非構造化記憶域

- XML文書を、文字データとして物理的に格納する
- 内部的にはCLOB型を利用
 - SecureFiles(データ圧縮、暗号化)を指定可能
- デフォルトの記憶域
- XMLスキーマの指定は不要
 - 任意の構造を持つXML文書を格納できる
 - XMLスキーマを指定することもできる



非構造化記憶域(使用例)

- 表作成

```
CREATE TABLE xmltab1 (xml XMLType);
```

- データ格納

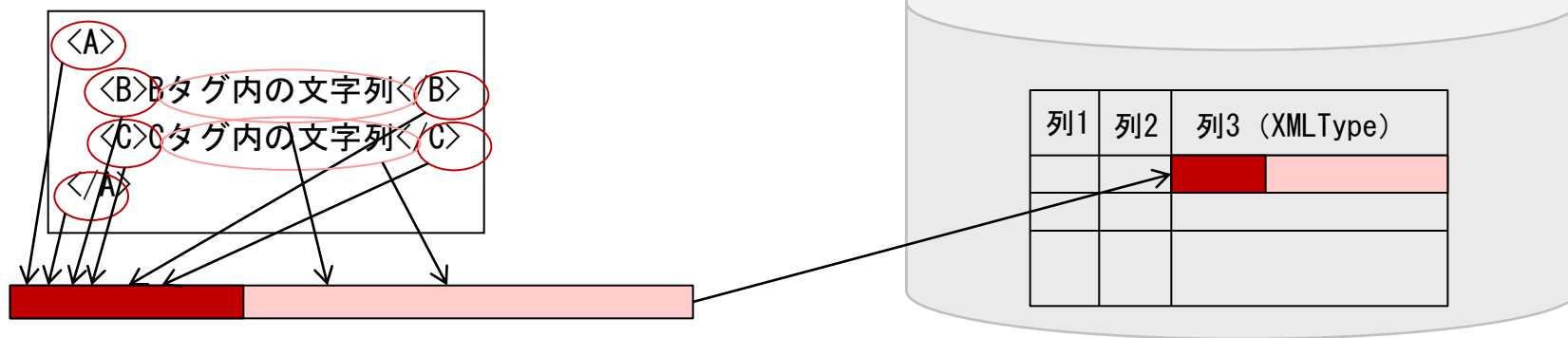
```
INSERT INTO xmltab1 VALUES (XMLType(' <A><B>Bタグ内の文字列  
</B><C>Cタグ内の文字列</C></A>' ));
```

【参考】SecureFiles を指定する場合(推奨)

```
CREATE TABLE xmltab1sf (xml XMLType)  
XMLTYPE COLUMN xml STORE AS SECUREFILE CLOB  
(COMPRESS LOW);
```

バイナリXML記憶域

- XML文書を、XMLの構造部分とコンテンツ部分に分け、独自のバイナリ形式で格納する
- 内部的にはBLOB型を利用
 - SecureFiles(データ圧縮、暗号化)を指定可能
- XMLスキーマの指定は不要
 - 任意の構造を持つXML文書を格納できる
 - XMLスキーマを指定することもできる



バイナリXML記憶域(使用例)

- 表作成

```
CREATE TABLE xmltab2 (xml XMLType)
XMLTYPE COLUMN xml STORE AS BINARY XML;
```

- データ格納

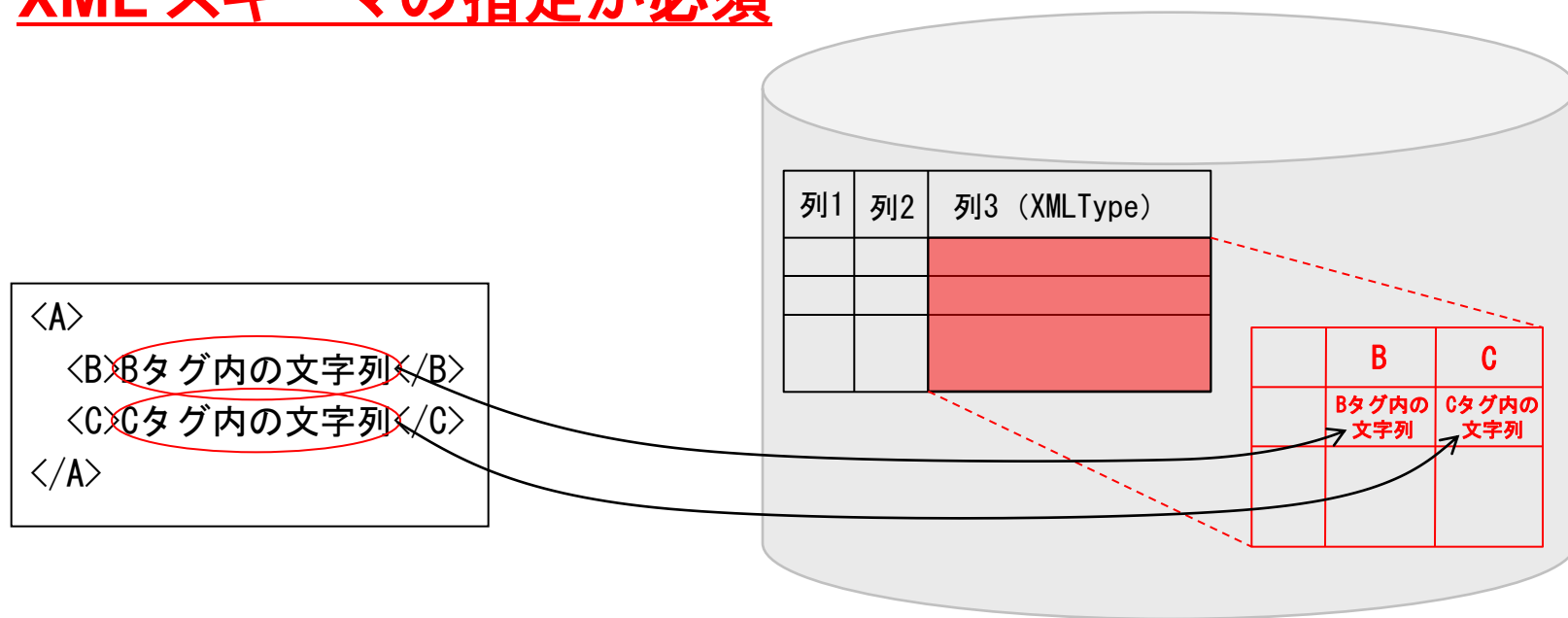
```
INSERT INTO xmltab2 VALUES (XMLType(' <A><B>Bタグ内の文字列
</B><C>Cタグ内の文字列</C></A>' ));
```

【参考】SecureFiles を指定する場合(推奨)

```
CREATE TABLE xmltab2sf (xml XMLType)
XMLTYPE COLUMN xml STORE AS SECUREFILE BINARY XML
(COMPRESS LOW);
```


構造化記憶域

- XML文書を、XML スキーマに基づく O/R マッピングによって格納する
- 内部的にはオブジェクト型 (O/Rマッピング) を利用
- **XML スキーマの指定が必須**



XMLスキーマ

```
1 <schema
2   xmlns="http://www.w3.org/2001/XMLSchema"
3   xmlns:xdb="http://xmlns.oracle.com/xdb"
4   targetNamespace="http://www.oracle.co.jp/a"
5   xdb:storeVarrayAsTable="true"
6   elementFormDefault="unqualified">
7   <element name="A"> ← ルート要素
8     <complexType>
9       <sequence>
10        <element name="B" type="string"/>
11        <element name="C" type="string"/>
12      </sequence>
13    </complexType>
14  </element>
15 </schema>
```

サンプルXML文書

```
<A>
  <B>Bタグ内の文字列</B>
  <C>Cタグ内の文字列</C>
</A>
```

XMLスキーマのOracle Databaseへの登録

```
1 DECLARE
2   doc VARCHAR2(4000) :=
3   '
```

構造化記憶域(使用例)

- 表作成

```
CREATE TABLE xmltab3 (xml XMLType)
XMLTYPE COLUMN xml STORE AS OBJECT RELATIONAL
XMLSCHEMA "http://www.oracle.co.jp/a.xsd" ELEMENT "A";
```

- データ格納

```
INSERT INTO xmltab3 VALUES (XMLType(' <a:A
xmlns:a="http://www.oracle.co.jp/a"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation=
"http://www.oracle.co.jp/a http://www.oracle.co.jp/a.xsd">
<B>Bタグ内の文字列</B><C>Cタグ内の文字列</C></a:A' ));
```

XMLスキーマの登録で Oracle Database に何が起きるのか

```
<element name="A">
  <complexType>
    <sequence>
      <element name="B"
        type="string"/>
      <element name="C"
        type="string"/>
    </sequence>
  </complexType>
</element>
```

```
CREATE OR REPLACE TYPE
"SCOTT"."Entry183_T" AS OBJECT
(
  "SYS_XDBPD$"
  "XDB"."XDB$RAW_LIST_T",
  "B"    VARCHAR2(4000),
  "C"    VARCHAR2(4000)
) FINAL INSTANTIABLE
```

自動的な XML スキーマから
Oracle データ型 (オブジェクト型)
への構造のマッピング

データ型のマッピング

スライドには一部のマッピングを掲載しています。

XML スキーマ	Oracle データ型	
	デフォルト	指定可能
integer	NUMBER	—
float, double	NUMBER	FLOAT DOUBLE BINARY FLOAT
dateTime, time	TIMESTAMP	DATE TIMESTAMP WITH TIMEZONE
boolean	RAW(1)	VARCHAR2
string, ID, IDREF, anyURI, anyType, anySimpleType	VARCHAR2(4000)	CHAR CLOB

クエリー・リライト

- XPathやXQueryを利用した問合せが、オブジェクト型の属性に直接アクセスするドット表記の問合せにリライトされ実行される

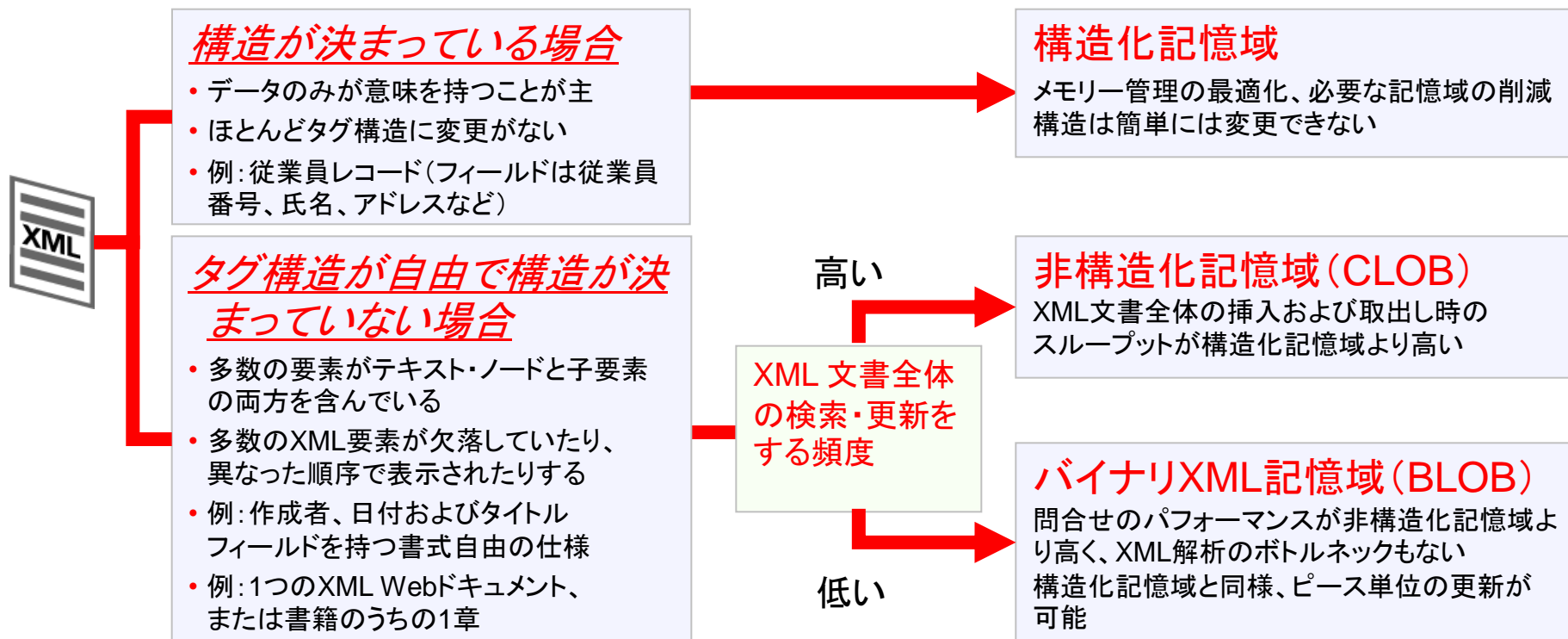
```
SELECT XMLCast(XMLQuery (' /a/b' PASSING xml  
RETURNING CONTENT) AS VARCHAR2(64)) FROM  
testxml;
```



```
SELECT x.xml.XMLData."B"  
FROM testxml x;
```

XMLType型で提供される記憶域

- XML 文書のデータ特性に合った適切な記憶域を選択可能



データ特性に適した豊富な格納方式

XMLType 型で提供される記憶域の種類と特徴

	構造化記憶域	非構造化記憶域	バイナリXML記憶域
格納効率 (SecureFiles未使用時)	◎	△	○
格納効率 (SecureFiles圧縮使用時)	◎	○	○
XPath ベースの 検索パフォーマンス	◎ ※XPath リライトされ検索	△ ※CLOB データから DOM ツリー構築後に検索	○ ※DOM ツリーは構築されないため、非構造化よりも高速に検索
更新処理パフォーマンス (XML文書全体)	○ ※これは、CPU時間での比較。ディスクI/Oとの相関で、一概に言えない部分もある	◎	○
更新処理 パフォーマンス (XML文書の一部)	◎ ※部分更新	△ ※全てのデータが更新	○ ※SecureFiles 使用時は部分更新
構造変更の柔軟性	○	◎	◎

まとめ

- Oracle Database には、XMLデータを格納するためのデータ型が存在する(XMLType型)
- XMLType型には、3つの物理的な格納方式が存在する
 - 非構造化記憶域
 - バイナリXML記憶域
 - 構造化記憶域
- 特に構造化記憶域では、XPath／XQuery問合せが高速
 - Oracle Database が自動的にO/Rマッピングを行い、XPath／XQuery問合せを、自動的にリレーショナル問合せに書き換えた上で実行するため、リレーショナル問合せと同等のパフォーマンス
- Oracle XML DB は、Oracle Database カーネル(特にC言語)で実装されているため、非常に高速

システム全体で必要となる ハードウェア・リソース



システム全体で必要となるハードウェア・リソース

- CPU使用量 (CPU時間)
- メモリ使用量
- ディスク使用量
 - 1処理あたりのI/O量
 - 1データを格納するために必要となるディスク容量
- ネットワーク使用量

CPU使用量(CPU時間)

※ 本セクションにおける「DB層でのXML処理」は、すべて、Oracle Databaseの利用を前提としています。

- AP層でのXML処理
 - Java
 - DOM構築が必要
 - I/O waitが多い(XML文書全体を取得)
- DB層でのXML処理
 - C
 - DOM構築が不要
 - ※ バイナリXML記憶域、構造化記憶域利用時
 - I/O waitが少ない(XML文書の一部を取得)
 - ※ バイナリXML記憶域、構造化記憶域利用時

メモリ使用量

※ 本セクションにおける「DB層でのXML処理」は、すべて、Oracle Databaseの利用を前提としています。

- AP層でのXML処理
 - Java
 - DOM構築が必要
 - DB層でのXML処理
 - C
 - DOM構築が不要
- ※ バイナリXML記憶域、構造化記憶域利用時

ディスク使用量 (1処理あたりのI/O量)

※ 本セクションにおける「DB層でのXML処理」は、すべて、Oracle Databaseの利用を前提としています。

- AP層でのXML処理
 - DOM構築が必要 ⇒ XML文書全体を参照する必要がある
 - データ圧縮をAP層で行うには、圧縮用のAPIを個別に呼び出す必要がある
- DB層でのXML処理
 - DOM構築が不要 ⇒ XML文書全体を参照する必要がない
 - ※ バイナリXML記憶域、構造化記憶域利用時
 - データ圧縮を行える
 - ※ 非構造化記憶域、バイナリXML記憶域利用時には、SecureFile圧縮を行える
 - ※ バイナリXML記憶域利用時は、独自のバイナリ形式を採用しており、(SecureFile圧縮の前の状態で)元のXML文書のサイズよりも少ないデータ量の書込／読込が行われる
 - ※ 構造化記憶域利用時は、XMLデータに含まれるタグ値・属性値などがO/Rマッピングされて格納されるため、元のXML文書のサイズよりも圧倒的に少ないデータ量の書込／読込が行われる

ディスク使用量

(1データを格納するために必要となるディスク容量)

※ 本セクションにおける「DB層でのXML処理」は、すべて、Oracle Databaseの利用を前提としています。

- AP層でのXML処理
 - データ圧縮をAP層で行うには、圧縮用のパッケージを個別に呼び出す必要がある
- DB層でのXML処理
 - データ圧縮を行える
 - ※ 非構造化記憶域、バイナリXML記憶域利用時には、SecureFile圧縮を行える
 - ※ バイナリXML記憶域利用時は、独自のバイナリ形式を採用しており、(SecureFileの圧縮を行わなくても)元のXML文書のサイズよりも少ないデータ量の書込／読込が行われる
 - ※ 構造化記憶域利用時は、XMLデータに含まれるタグ値・属性値などがO/Rマッピングされて格納されるため、元のXML文書のサイズよりも圧倒的に少ないデータ量の書込／読込が行われる

ネットワーク・トラフィック

※ 本セクションにおける「DB層でのXML処理」は、すべて、Oracle Databaseの利用を前提としています。

- AP層でのXML処理

- 処理の対象となるXMLインスタンス全体をDB層から取得する必要がある

※ ただし、APサーバーのローカルファイルシステムから取得する場合は該当しない

- DB層でのXML処理

- XML処理はDB層で完結するため、XML処理を実行するためのネットワーク・トラフィックは発生しない

まとめ(全体最適の観点から)

※ 本セクションにおける「DB層でのXML処理」は、すべて、Oracle Databaseの利用を前提としています。

- DB層でのXML処理は、システム全体で必要となるCPU、メモリ、ディスク、ディスク、ネットワーク・リソースを最小化する
 - 特に、同一ハードウェア構成での比較では、DB層でのXML処理の方が、AP層でのXML処理よりも高速
- Oracle XML DB は、Oracle Database カーネルで実装されている
 - Oracle Database が従来持つ、高速なデータ処理、同時リクエスト処理などが、XML DB においてもそのまま継承される

開発工数

本質ではない「開発」をいかに減らすか



AP層／DB層におけるXMLアプリケーション開発

- DOMプログラム 要(※1)／不要
- 開発柔軟性 低(※1)／高
- 開発コスト 高／低
- メモリの効率的な利用 不可(※2)／可

※1 Webサービスなどでは、DOMプログラムが隠蔽されたシンプルなパッケージが提供されている場合があるが、ここではあくまで、XML文書を直接読み書きするアプリケーションの開発を想定している。

※2 Oracle Coherence を利用するなど、AP層でメモリを効率的に利用する工夫をすることも可能。ただし、これにはAP層でのコーディングの段階で、メモリを効率的に利用するための検討を十分にすることが必要である。

AP層でXMLの処理を実行

- 複雑で冗長なDOM プログラム
- 開発コストの増加につながる

```
public void processDOM( Reader reader,   Writer writer)
    throws Exception
{
    org.apache.xerces.parsers.DOMParser parser;
    org.w3c.dom.Document doc;
    org.w3c.dom.NodeList nodeList;
    parser = new org.apache.xerces.parsers.DOMParser();
    parser.parse(new InputSource(reader));
    doc = parser.getDocument();
    nodeList =
        doc.getElementsByTagName("SomeNode");

    for (i = 0; i < nodeList.getLength())
    {
        System.out.print(nodeList.item(i).getNodeValue());
    }
}
```



DB層でXMLの処理を実行 (Oracle)

- XML 用のシンプルな API
 - コード量の削減
 - 習得が容易
 - 必要に応じ低レベルのDOM APIも使用可能

```
select
extractValue(object_value, '/PurchaseOrder/Refer
ence')
from PURCHASEORDER p
/
```

Oracle XML DB を利用した XMLアプリケーション開発

- XMLデータをXML形式あるいはリレーショナル形式で参照する
 - XQuery／XPathの適用
→ 適用結果をXML形式、あるいはリレーショナル形式で参照
 - XSLTスタイルシートの適用
- リレーショナルデータをXML形式で参照する
 - リレーショナル表の各列をタグ値、属性値にマッピング



これらの処理を、全て
DB層 (SQL標準) のみで実行可能

XMLデータを参照するために利用可能なSQL関数(抜粋)

- **XMLCast** ... XMLインスタンスあるいはフラグメントからテキストノードを取り出し、指定されたSQLデータ型へ変換する
- **XMLQuery** ... 与えられたXMLインスタンスに対してXPathあるいはXQueryを適用する
- **XMLTable** ... 与えられたXMLインスタンスに対してXPathあるいはXQueryを適用する(結果をデータベース表として取得する)
- **XMLExists** ... 与えられたXMLインスタンスに対してXPathあるいはXQueryを適用し、結果が空でなければTRUEを、結果が空であればFALSEを返す
- **XMLTransform** ... 与えられたXMLインスタンスに対してXSLTスタイルシートを適用する

準備

- XMLType型の列を1つだけ持つ表を作成する

```
CREATE TABLE testxml (xml XMLTYPE)
XMLTYPE COLUMN xml STORE AS SECUREFILE BINARY XML
(COMPRESS LOW);
```

- この表に3件のデータを格納する

```
INSERT INTO testxml VALUES
(XMLTYPE(' <a><b>dummy_b1</b><c>dummy_c1</c></a>' ));
INSERT INTO testxml VALUES
(XMLTYPE(' <a><b>dummy_b2</b><c>dummy_c2</c></a>' ));
INSERT INTO testxml VALUES
(XMLTYPE(' <a><b>dummy_b3</b><c>dummy_c3</c></a>' ));
COMMIT;
```

特定のタグ値を取得する

- 特定タグ値の取得

```
SELECT XMLCast(XMLQuery (' /a/b' PASSING xml RETURNING  
CONTENT) AS VARCHAR2(64)) b FROM testxml;
```

- 実行結果

B

dummy_b1

dummy_b2

dummy_b3

ビューの使用例

- ビューの作成

```
CREATE VIEW relview AS SELECT
  XMLCast(XMLQuery ('/a/b' PASSING xml RETURNING CONTENT) AS
  VARCHAR2(64)) b,
  XMLCast(XMLQuery ('/a/c' PASSING xml RETURNING CONTENT) AS
  VARCHAR2(64)) c FROM testxml;
```

- ビューの参照

```
SQL> SELECT * FROM relview;
```

B	C
-----	-----
dummy_b1	dummy_c1
dummy_b2	dummy_c2
dummy_b3	dummy_c3

特定のタグ値で絞り込み検索を行う(1)

- XMLCast と XMLQuery を利用するパターン

```
SELECT XMLCast(XMLQuery (' /a/b' PASSING xml RETURNING  
CONTENT) AS VARCHAR2(64)) b FROM testxml  
WHERE XMLCast(XMLQuery (' /a/c' PASSING xml RETURNING  
CONTENT) AS VARCHAR2(64))=' dummy_c3' ;
```

- 実行結果

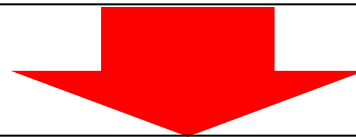
B

dummy_b3

特定のタグ値で絞り込み検索を行う(2)

- XMLExists を利用するパターン

```
SELECT XMLCast(XMLQuery (' /a/b' PASSING xml RETURNING  
CONTENT) AS VARCHAR2(64)) b FROM testxml  
WHERE XMLExists (' /a[c="dummy_c3"]' PASSING xml);
```



- 実行結果

B


dummy_b3

XQuery の実行結果をリレーショナル形式で参照

- XMLTable を利用

```
SELECT x.b, x.c FROM
  testxml t,
  XMLTable('/a' PASSING t.xml
    COLUMNS
      b VARCHAR2(4000) PATH 'b',
      c VARCHAR2(4000) PATH 'c'
  ) x;
```

- 実行結果



B	C
dummy_b1	dummy_c1
dummy_b2	dummy_c2
dummy_b3	dummy_c3

XSLTスタイルシートを適用する

```
1 SELECT XMLTransform (xml, '<?xml version="1.0"?>
2 <xsl:stylesheet version="1.0"
3 xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4   <xsl:template match="/">
5     <transformed>
6       <xsl:apply-templates select="//b"/>
7     </transformed>
8   </xsl:template>
9   <xsl:template match="b">
10    <xsl:value-of select="../c"/>
11  </xsl:template>
12 </xsl:stylesheet>') result FROM testxml;
```

- 実行結果

RESULT

```
<transformed>dummy_c1</transformed>
<transformed>dummy_c2</transformed>
<transformed>dummy_c3</transformed>
```

リレーショナルデータをXML形式で参照するために利用可能なSQL関数(抜粋)

- **XMLElement** ... リレーショナルデータから単一のXML要素を生成する
- **XMLForest** ... リレーショナルデータから複数のXML要素を生成する
- **XMLAttributes** ... リレーショナルデータからXML要素の属性を生成する

準備

- リレーショナル表を作成する

```
CREATE TABLE testrel  
(  
  b VARCHAR2(4000),  
  c VARCHAR2(4000)  
);
```

- この表に3件のデータを格納する

```
INSERT INTO testrel VALUES ('dummy_b1', 'dummy_c1');  
INSERT INTO testrel VALUES ('dummy_b2', 'dummy_c2');  
INSERT INTO testrel VALUES ('dummy_b3', 'dummy_c3');  
COMMIT;
```

「リレーショナル→XML」変換例(1)

- XMLElement を利用

```
SELECT
  XMLElement ("a",
    XMLElement ("b", b),
    XMLElement ("c", c)
  ) result
FROM testrel;
```

- 実行結果

RESULT

```
<a><b>dummy_b1</b><c>dummy_c1</c></a>
<a><b>dummy_b2</b><c>dummy_c2</c></a>
<a><b>dummy_b3</b><c>dummy_c3</c></a>
```


「リレーショナル→XML」変換例(2)

- XMLElement とXMLForestを利用

```
SELECT
  XMLElement ("a",
    XMLForest (b AS "b", c AS "c")
  ) result
FROM testrel;
```

- 実行結果

RESULT

```
<a><b>dummy_b1</b><c>dummy_c1</c></a>
<a><b>dummy_b2</b><c>dummy_c2</c></a>
<a><b>dummy_b3</b><c>dummy_c3</c></a>
```

「リレーショナル→XML」変換例(3)

- XMLElement と XMLAttributes を利用
(Aタグの属性として列Bを指定)

```
SELECT XMLElement ("a",  
    XMLAttributes (b AS "b"),  
    XMLElement ("c", c)  
    ) result FROM testrel;
```

- 実行結果

RESULT

```
<a b="dummy_b1"><c>dummy_c1</c></a>  
<a b="dummy_b2"><c>dummy_c2</c></a>  
<a b="dummy_b3"><c>dummy_c3</c></a>
```

ビューの使用例

- ビューの作成

```
CREATE VIEW xmlview AS SELECT
  XMLElement("a",
    XMLElement("b", b), XMLElement("c", c)
  ) xml FROM testrel;
```

- ビューの参照

```
SQL> SELECT * FROM xmlview;
```

XML

```
<a><b>dummy_b1</b><c>dummy_c1</c></a>
<a><b>dummy_b2</b><c>dummy_c2</c></a>
<a><b>dummy_b3</b><c>dummy_c3</c></a>
```

まとめ

- Oracle Database によるXMLアプリケーション開発では、SQL関数の引数にXPathやXQueryを渡すだけで処理が実行できる
 - 実行結果は、XML形式、リレーショナル形式のいずれでも取得可能
- リレーショナルデータをXMLデータとして参照することも可能
- DB層でのXMLアプリケーション開発は、AP層でのXMLアプリケーション開発に比べてシンプル

運用・管理コスト



XMLアプリケーションの運用・管理

- 運用・管理
 - 定期的なバッチ処理
 - チューニング
 - バックアップ／リカバリ
etc.
- XMLアプリケーションがAP層で実装されている場合
 - XMLアプリケーション(AP層)とデータベース(DB層)を個別に管理
- XMLアプリケーションがDB層で実装されている場合
 - XMLアプリケーション＝データベースであるため、データベースのみが管理対象

Oracle Enterprise Manager

※ 画面は DB 10.1 付属のEM Database Control のスクリーンショットです。

結果
統計
プラン

サマリー

一般

モジュール **SQL Worksheet Connection**
 アクション
 解析スキーマ **XTEST**
 PL/SQLソース(行番号) **DBSNMP.BSLN_INTERNAL**
 (1951)
 SQLプロファイル **N/A**
 SQL計画ベースライン **N/A**

待機ごとのアクティビティ

■ 残りの待機(18.2%)
■ CPU(81.8%)

時間ごとのアクティビティ

経過時間(秒) **0.74**
 CPU時間(秒) **0.61**
 待機時間(秒) **0.14**

経過時間ブレイクダウン

SQL時間(秒) **0.74**
 PL/SQL時間(秒) **<0.01**
 Java時間(秒) **0.00**

共有カーソル統計

合計解析 **2**
 ハード解析 **1**
 子カーソル **1**
 ロード済計画 **1**
 無効化 **0**
 最大カーソル・サイズ(KB) **128.94**
 すべてのカーソル・サイズ(KB) **128.94**
 初回ロード時間 **2007/09/13 11:22:06 (UTC+09:00)**
 最新ロード時間 **2007/09/13 11:22:06 (UTC+09:00)**

実行統計

	合計	1実行当たり	1行当たり
実行	2	1	1.00
経過時間(秒)	0.74	0.37	0.37
CPU時間(秒)	0.61	0.30	0.30
バッファ読取り	3,608	1,804.00	1,804.00
ディスク読取り	0	0.00	0.00
ダイレクト書き込み	0	0.00	0.00
行	2	1.00	1
フェッチ数	2	1.00	1.00

その他統計

すべての行を取り出す実行(%) **100.00**
 平均永続メモリー(KB) **112.38**
 平均実行時メモリー(KB) **92.28**
 シリアライズ可能な異常終了 **0**
 リモート **いいえ**
 不変 **いいえ**
 子ラッチ番号 **0**

結果
統計
プラン

SQL修復アドバイザー
SQLの詳細
SQLチューニング・アドバイザーのスケジュール

Oracle Enterprise Manager

SQLワークシート: u0728

実行するSQL文を入力します。複数の文がある場合は、カーソルの場所またはハイライト表示された文により、実行する文を判別します。文は空白行で区切る必要があります。

SQLコマンド

```
SELECT XMLCast(XMLQuery('ばちオーダ/ばちオーダ企業'  
PASSING po.ときゅ RETURNING CONTENT) AS VARCHAR2(30))  
FROM しーからむなし po  
WHERE XMLElementExists('/ばちオーダ[企業コード="CC500"]'  
PASSING po.ときゅ);
```

- 実行にはバインド変数を使用
- 自動コミット
- SELECT文のみを許可

実行

最終実行SQL

```
SELECT XMLCast(XMLQuery('ばちオーダ/ばちオーダ企業' PASSING po.ときゅ RETURNING CONTENT)  
AS VARCHAR2(30))  
FROM しーからむなし po  
WHERE XMLElementExists('/ばちオーダ[企業コード="CC500"]' PASSING po.ときゅ)
```

最終実行の詳細

SQL修復アドバイザー

SQLの詳細

SQLチューニング・アドバイザーのスケジュール

結果

統計

プラン

実行時間(秒) 0.769

```
XMLCAST(XMLQUERY('ばちオーダ/ばちオーダ企業'PASSINGPO.ときゅRETURNINGCONTENT)ASVARCHAR2(30))
```

日本オラクル株式会社

ORACLE

Oracle Enterprise Manager を使用した XML DBの管理(10g~)

※ 画面は DB 11.2 付属のEM Database Control のスクリーンショットです。

The screenshot shows the Oracle Enterprise Manager Database Control interface. The main navigation menu includes: ホーム, パフォーマンス, 可用性, サーバー, スキーマ, データ移動, ソフトウェアとサポート. The main content area is divided into several sections: データベース・オブジェクト, プログラム, マテリアライズド・ビュー, 変更の管理, データ・マスキング, ユーザー定義タイプ, Workspace Manager, and Text Manager. A red box highlights the 'XML データベース' (XML Database) section under 'データベース・オブジェクト', which includes links for: 構成, リソース, アクセス制御リスト, XMLスキーマ, XMLType表, XMLTypeビュー, XML索引, and XMLリポジトリ・イベント. The footer contains copyright information for Oracle Corporation, 1996-2009.

- [XMLデータベース](#)
- [構成](#)
- [リソース](#)
- [アクセス制御リスト](#)
- [XMLスキーマ](#)
- [XMLType表](#)
- [XMLTypeビュー](#)
- [XML索引](#)
- [XMLリポジトリ・イベント](#)

表の作成(1)

- VARCHAR2
- NUMBER
- DATE
- CHAR
- FLOAT
- INTEGER
- NCHAR
- NVARCHAR2
- LONG
- LONG RAW
- RAW
- ROWID
- UROWID
- BLOB
- CLOB
- NCLOB
- BFILE
- TIMESTAMP
- INTERVAL YEAR
- INTERVAL DAY
- BINARY_DOUBLE
- BINARY_FLOAT
- XMLTYPE**

Oracle Enterprise Manager 11g Database Control

データベース: TESTXMLとしてログイン

データベース: TESTXMLとしてログイン

表の作成

SQL表示 取消 OK

名前:

スキーマ: TESTXML

表領域: <デフォルト>

編成: 標準(テーブル構成)

次を使用して定義: 列仕様

デフォルトのLOB属性の設定 暗号化オプション

選択	名前	データ型	サイズ	スケール	NOT NULL	デフォルト値	暗号化
<input checked="" type="radio"/>	<input type="text"/>	VARCHAR2	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	VARCHAR2	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	NUMBER	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	DATE	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	CHAR	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	FLOAT	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	INTEGER	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	NCHAR	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	NVARCHAR2	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	LONG	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	LONG RAW	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	RAW	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	ROWID	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	UROWID	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	BLOB	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	CLOB	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	NCLOB	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	BFILE	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	TIMESTAMP	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	INTERVAL YEAR	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	INTERVAL DAY	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	BINARY_DOUBLE	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	BINARY_FLOAT	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>
<input type="radio"/>	<input type="text"/>	XMLTYPE	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>

5列追加

主キー列の指定

一意キー列の指定

データベース | ヘルプ | ログアウト

Copyright (c) 1996, 2009, Oracle. All rights reserved.

Oracle, ID Edwards, PeoplesoftおよびRetailはOracle Corporationおよびその関連企業

Oracle Enterprise Manager 11gの機能情報

この名前は、それぞれの所有者の商標または登録商標です。

SQL表示 取消 OK

表の作成(2)

ORACLE Enterprise Manager 11g Database Control

データベース・インスタンス: [al112.jp.oracle.com](#) > 表 > 表の作成 > XMLTYPE列の拡張属性: XML

ヘルプ ログアウト データベース TESTXMLとしてログイン

取消 続行

XML型

XMLスキーマの詳細

XMLスキーマ型 **XMLスキーマに基づく／基づかない**

記憶域の詳細を指定

記憶域のタイプ **BINARY**

LOB属性

セグメント名 <割り当てられたシステム>

LOB記憶域タイプ **BasicFile**

SecureFile

処理速度と拡張性を高めるとともに、圧縮、暗号化、重複除外などの高度な機能を使用できます

1度にアクセスするデータのサイズ (CHUNK) バイト

旧バージョンの保持

システムのデフォルトを使用

保存を使用

旧バージョンを時間ベースのアプローチで保存します

PCTVERSION(%)を指定

古いバージョンの保持に使用するLOBセグメントをパーセンテージで指定します

キャッシュ **NOCACHE**

LOBデータを行に格納(ENABLE STORAGE IN ROW)

LOB記憶域属性

ヒントほとんどの場合、記憶域パラメータのデフォルト値を許容できます。これらのオプションは、オブジェクトのパフォーマンスを向上させるために変更することができます。

表領域

名前

表領域属性のリフレッシュ

エクステント管理 **ローカル**

セグメント管理 **手動**

割当てタイプ **USER**

ロギング **<デフォルト>**

エクステント 空きリスト

表の作成(3)

**LOB記憶域タイプで
SecureFileを選択すると
さらに詳細なオプションが
表示される**

記憶域の詳細を指定

記憶域のタイプ **BINARY**

LOB属性

セグメント名 <割り当てられたシステム>

LOB記憶域タイプ

- BasicFile
- SecureFile**
処理速度と拡張性を高めるとともに、圧縮、暗号化、重複除外などの高度な機能を使用できます

キャッシュ **NOCACHE**

LOBデータを行に格納(ENABLE STORAGE IN ROW)

保存

- 自動
古いバージョンの保存をシステムが自動的に管理します
- 最大 サイズ バイト
最大サイズ制限に達するまで旧バージョンを保存します
- 最小 秒
指定された期間、旧バージョンを保存します
- なし
旧バージョンを保存しません

圧縮

- なし
LOBを無圧縮で格納します
- 低
最も高速で、最も圧縮率の低い圧縮です
- 中
高速で圧縮率の低い圧縮です
- 高
低速で圧縮率の高い圧縮です

重複除外

- 無効
同一のLOB間でデータを共有しません
- 有効
同一のLOBを1回しか格納しないようにして、記憶域を節約します

LOB記憶域属性

☑ ヒント ほとんどの場合、記憶域パラメータのデフォルト値を許容できます。これらのオプションは、オブジェクトのパフォーマンスを向上させるために変更することができます。

表領域

名前

エクステント管理 **ローカル**

セグメント管理 **自動**

割当てタイプ **SYSTEM**

ログギング

まとめ

- XMLアプリケーションがDB層で実装されている場合、運用・管理対象は Oracle Database のみ
- 従来のデータベースの管理タスクに加え、XMLに関連した操作についても、Enterprise Manager(Database Control)から行える

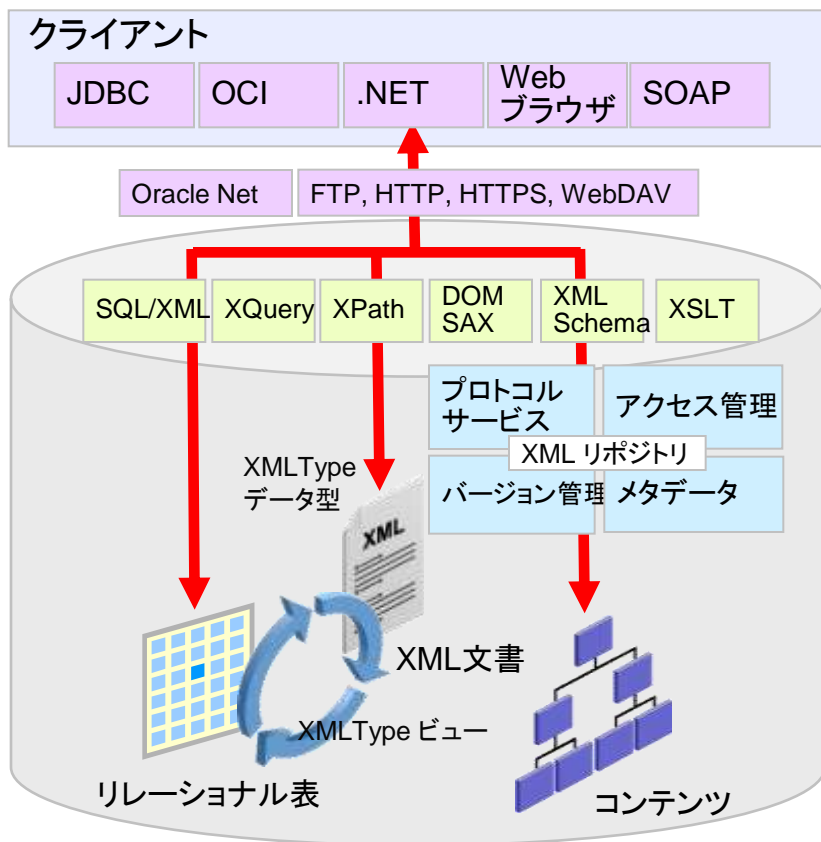
信頼性・汎用性・多機能性



XML DB=Oracle Database であること、 その価値

- Oracle XML DB は、信頼性の高い**Oracle Database** **カーネル**で実装された XML DB エンジン
- **Oracle Database** の**既存機能との親和性**が非常に高い
 - RAC (Oracle Real Application Clusters)
 - Partitioning
 - Advanced Compression
 - Oracle Text (特に、XPathに基づく全文検索に対応)
 - パラレル・クエリ
 - Data Guard
etc.
- **Oracle Exadata** 上で利用可能

Oracle XML DB の全体像

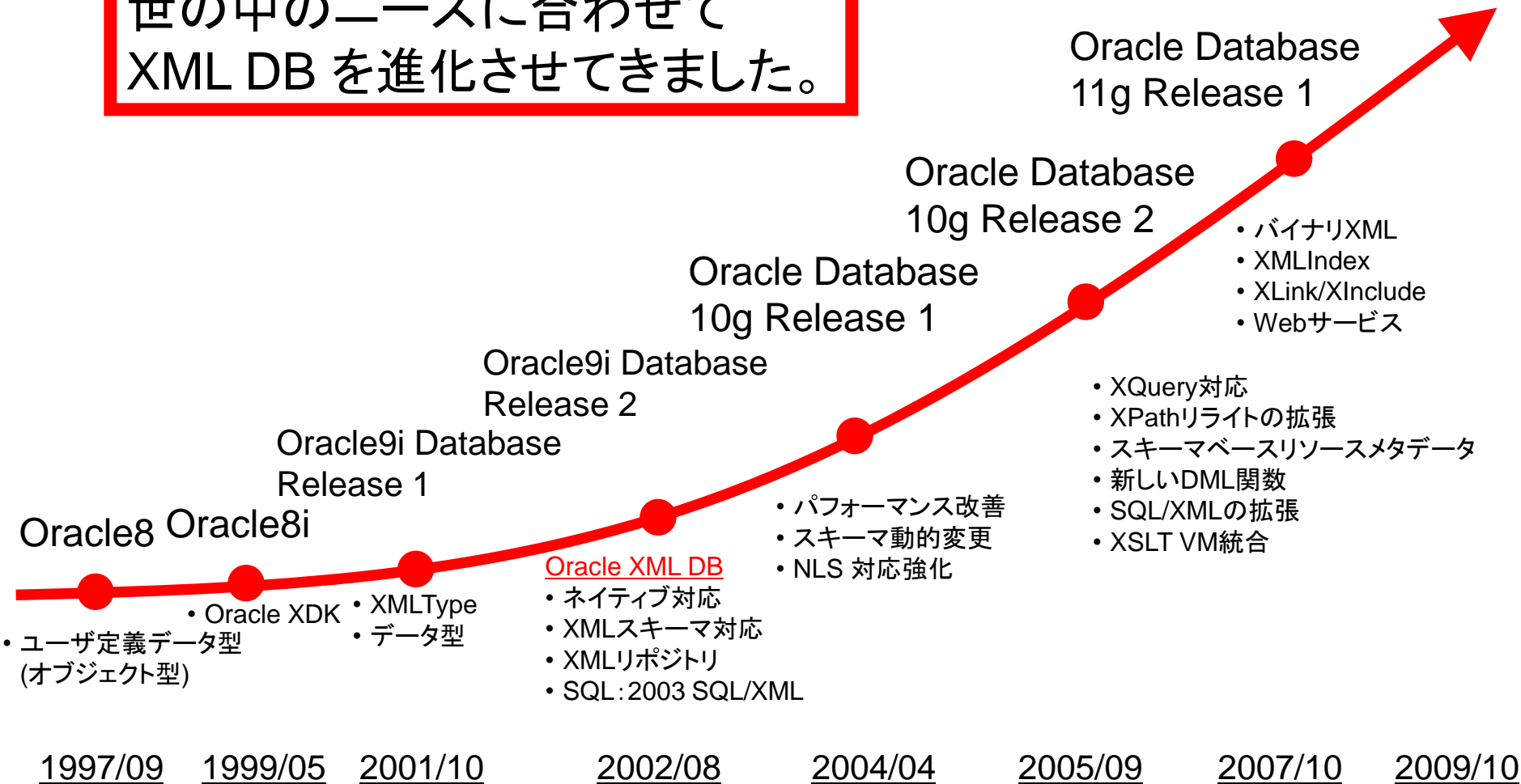


- XML 文書をそのままOracle Database に格納できる
- XML関連の処理は全てデータベース・プロセスによって実行される
- XML 文書の特性に合わせて、効率的な格納方式を選択できる
- 大規模システムへの対応
- 開発生産性
 - W3C 標準規格への準拠
 - XQuery1.0, SQL:2008 (SQL/XML) など技術者の得意な言語で開発ができる
 - 容易な RDBMS データとの連携

Oracle XML DB ロードマップ

Oracle Database
11g Release 2

世の中のニーズに合わせて
XML DB を進化させてきました。



ORACLE

全体のまとめ

- DB層でのXML処理は、全体最適（システム全体で必要となるCPU、メモリ、ハードディスク、ネットワーク・リソースの削減）の観点から有効（特に「高速」）
- 構造化記憶域を利用した場合、XPathの適用、XQueryの実行は、非常に高速（自動的にリレーショナル問合せに書き換えられて実行されるため）
- Oracle XML DB を利用することで、XMLアプリケーション開発（の本質的な部分）は、XMLスキーマの設計と、（XPath または XQueryを含む）SQL文の記述だけで完結する
- XMLデータとリレーショナルデータの相互変換ができる
- Oracle XML DB は、Oracle Database カーネルで実装されており、既存のデータベース機能との親和性が高く、信頼性が高い

参考

- 製品マニュアル

- 『XML DB開発者ガイド』

http://download.oracle.com/docs/cd/E16338_01/nav/portal_5.htm#xml

- OTN

- Oracle XML DB

<http://www.oracle.com/technetwork/jp/database/index-091758-ja.html>

- フォーラム(会議室)

<http://forums.oracle.com/forums/main.jspa?categoryID=484>

- 「データベース」カテゴリ→「データベース一般」

<http://forums.oracle.com/forums/forum.jspa?forumID=1328>

- 「テクノロジー」カテゴリ→「XML」

<http://forums.oracle.com/forums/forum.jspa?forumID=1356>

OTN×ダイセミ でスキルアップ!!



- ・一般的な技術問題解決方法などを知りたい!
- ・ 세미나資料など技術コンテンツがほしい!

Oracle Technology Network(OTN)を御活用下さい。

<http://forums.oracle.com/forums/main.jspa?categoryID=484>

一般的技術問題解決にはOTN揭示版の
「データベース一般」をご活用ください

※OTN揭示版は、基本的にOracleユーザー有志からの回答となるため100%回答があるとは限りません。
ただ、過去の履歴を見ると、質問の大多数に関してなんらかの回答が書き込まれております。

<http://www.oracle.com/technetwork/jp/content/index-086873-ja.html>

過去のセミナー資料、動画コンテンツはOTNの
「OTNセミナー オンデマンドコンテンツ」へ

※ダイセミ事務局にダイセミ資料を請求頂いても、お受けできない可能性がございますので予めご了承ください。
ダイセミ資料はOTNコンテンツ オン デマンドか、セミナー実施時間内にダウンロード頂くようお願い致します。

ORACLE

OTNセミナー オンデマンド コンテンツ

ダイセミで実施された技術コンテンツを動画で配信中!!

ダイセミのライブ感はそのままに、好きな時間で受講頂けます。

最新のコンテンツ

 <p>エンジニアのための ITIL実践術 再生時間: 60分</p>	 <p>ここからはじめよう Oracle PL/SQL入門 再生時間: 60分</p>	 <p>実践!!高可用システム構築 -RAC基本 再生時間: 60分</p>	 <p>お悩み解決! Oracle のサイジング 再生時間: 60分</p>
--	--	---	--

Database

 <p>今さら聞けない!?バックアップ・リカバリ 再生時間: 60分</p>	 <p>意外と簡単!? Oracle Database 11g -セ 再生時間: 60分</p>	 <p>実践!!バックアップ・リカバリ 再生時間: 60分</p>	 <p>意外と簡単!? Oracle Database 11g -デ 再生時間: 60分</p>
---	---	--	--

>> もっと見る

twitter

最新情報つぶやき中

oracletechnetjp

- ・人気コンテンツは?
- ・お勧め情報
- ・公開予告 など

OTN トップページ <http://www.oracle.com/technetwork/jp/index.html>
ページ左「基本リンク」>「OTN セミナー オンデマンド」

※掲載のコンテンツ内容は予告なく変更になる可能性があります。

期間限定での配信コンテンツも含まれております。お早めにダウンロード頂くことをお勧めいたします。

ORACLE

Oracle エンジニアのための技術情報サイト オラクルエンジニア通信

<http://blogs.oracle.com/oracle4engineer/>

twitter

最新情報つぶやき中
oracletechnetjp

技術資料

- ダイセミの過去資料や製品ホワイトペーパー、スキルアップ資料などを多様な方法で検索できます
- キーワード検索、レベル別、カテゴリ別、製品・機能別

コラム

- オラクル製品に関する技術コラムを毎週お届けします
- 決してニッチではなく、誰もが明日から使える技術の「あ、そうだったんだ！」をお届けします



こんな資料が人気です

- ✓ 6か月ぶりに資料ダウンロードランキングの首位が交代！
新王者はOracle Database構築資料でした。
- ✓ データベースの性能管理手法について、Statspack派もEnterprise Manager派も目からウロコの技術特集公開中

オラクルエンジニア通信



ORACLE

Oracle Databaseの価格ご存知ですか？

問題：

Oracle Databaseの最小構成はいくらでしょうか？

ヒント：

Oracle Standard Edition Oneを
5Named User Plus(指名ユーザ) というのが最小構成です。

問題：

Real Applications Clusters(RAC) Optionはいくらでしょうか？

ヒント：

RACはOracle Database Enterprise EditionのOptionです。

答えはこちら↓ ログイン不要の簡単見積もり

[ライセンス見積もりヘルプ](#)

検索

見積もり
Start!

ORACLE

ITプロジェクト全般に渡る無償支援サービス

Oracle Direct Conciergeサービス

■ パフォーマンス診断サービス

- Webシステム ボトルネック診断サービス **NEW**
- データベースパフォーマンス 診断サービス

■ 移行支援サービス

- SQL Serverからの移行支援サービス
- DB2からの移行支援サービス
- Sybaseからの移行支援サービス
- MySQLからの移行支援サービス
- Postgre SQLからの移行支援サービス
- Accessからの移行支援サービス
- Oracle Application ServerからWeblogicへ移行支援サービス **NEW**

■ システム構成診断サービス

- Oracle Database構成相談サービス
- サーバー統合支援サービス
- 仮想化アセスメントサービス
- メインフレーム資産活用相談サービス
- BI EEアセスメントサービス
- 簡易業務診断サービス

■ バージョンアップ支援サービス

- Oracle Databaseバージョンアップ支援サービス
- Weblogic Serverバージョンアップ支援サービス **NEW**
- Oracle Developer/2000(Froms/Reports) Webアップグレード相談サービス

オラクル社のエンジニアが 直接ご支援します
お気軽にご活用ください!

オラクル 無償支援

検索

ORACLE



1日5組限定！

製品無償評価サービス

提供シナリオ一例

- ・データベースチューニング
- ・無停止アップグレード
- ・アプリケーション性能・負荷検証
- ・Webシステム障害解析

インストールすることなく、すぐに体験いただけます

- サービスご提供までの流れ
 1. お問い合わせフォームより「製品評価サービス希望」と必要事項を明記し送信下さい
 2. 弊社より接続方法手順書およびハンズオン手順書を送付致します
 3. 当日は、弊社サーバー環境でインターネット越しに製品を体感頂けます
- ※サービスご提供には事前予約が必要です

Web問い合わせフォーム

「ダイデモ」をキーワードに検索することで申し込みホームページにアクセスできます

<http://www.oracle.com/jp/direct/services/didemo-195748-ja.html>

ORACLE

あなたにいちばん近いオラクル



Oracle Direct

まずはお問合せください

検索

システムの検討・構築から運用まで、ITプロジェクト全般の相談窓口としてご支援いたします。

システム構成やライセンス/購入方法などお気軽にお問い合わせ下さい。

Web問い合わせフォーム

専用お問い合わせフォームにてご相談内容を承ります。

<http://www.oracle.com/jp/direct/inquiry-form-182185-ja.html>

※こちらから詳細確認のお電話を差し上げる場合がありますので、ご登録されている連絡先が最新のものになっているか、ご確認下さい。

フリーダイヤル

0120-155-096

※月曜～金曜 9:00～12:00、13:00～18:00

(祝日および年末年始除く)

ORACLE

Appendix

一般的なOracleのPPT作成時のガイドライン(ダイセミに限らず)。マーケティング本部が管理されている内容です。

最新版はこちらをご確認ください。

<http://my.oracle.com/portal/page/myo/global/Marketing%20Japan/Branding:Logo>

「プレゼンテーション資料」

