Oracle
DBA & Developer Days 2011

日本オラクル、今年最大の技術トレーニングイベント

2011年11月9日(水)～11月11日(金) シェラトン都ホテル東京

ORACLE

SQL文でできる！
Oracle Databaseの全文検索機能「Oracle Text」の活用法

日本オラクル株式会社　テクノロジー製品事業統括本部
シニアエンジニア　金井 盛隆

以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント（確約）するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

Agenda

- Oracle Text とは？
- Oracle Textによるアプリケーション開発
- XML検索
- 多言語検索
- チューニング
- 従来のデータベース機能との親和性

Agenda

- Oracle Text とは？
  - 概要
  - 簡単な使用例
  - 索引作成のメカニズム
  - 検索のメカニズム
  - 索引メンテナンスのメカニズム
- Oracle Textによるアプリケーション開発
- XML検索
- 多言語検索
- チューニング
- 従来のデータベース機能との親和性

全文検索はなぜ必要か？

- 部分一致検索
  - ✓ パターン・マッチング全角・半角・大文字・小文字の区別あり
  - ✓ CHAR/VARCHAR2型のみ検索対象となる
- テキスト検索
  - ✓ 予め検索対象テキストから作成しておいたテキスト索引を使用し検索
  - ✓ 基本的に、全角・半角・大文字・小文字の区別なし
  - ✓ CHAR/VARCHAR2/CLOB/BLOB/BFILE 型などが検索対象となる

部分一致検索

```
SELECT * FROM doc
  WHERE text LIKE '%野田佳彦%';
```

テキスト検索

```
SELECT * FROM doc
 WHERE CONTAINS(text, '野田佳彦') > 0;
```

文書2のみヒット

文書1, 2共にヒット

スペース、「・」、「／」など記号の有無を無視した検索が可能

1: …野田　佳彦さんは選挙で、…
2: …挨拶で、野田佳彦さんは…

Oracle Textとは？

- Oracle Databaseカーネルで実装された全文検索エンジン
- SQL関数（CONTAINS）で全文検索を実行可能
  - 特に、SQLのみで全文検索アプリケーションを開発可能
- Oracle8i Database Release 8.1.6以降で利用可能

  ※ Oracle8i 当時の製品名は「interMedia Text」

  ※ Oracle9i Database Release 1（9.0.1）以降は「Oracle Text」
- オプション・ライセンスなしで利用できる
  - Enterprise Edition、Standard Edition、Standard Edition One、Expess Edition で利用可能
- XML検索に対応
- 多言語検索に対応

簡単な使用例(1)
ユーザー作成

- Oracle Textの索引を作成するデータベース・ユーザーを作成

```
CONNECT / AS SYSDBA
CREATE USER userctx IDENTIFIED BY userctx;
GRANT connect, resource, ctxapp TO userctx;
```

※ CTXAPPロールは、システム定義のロールで、Oracle Textの索引作成に必要な権限が含まれます。(Oracle Textが提供するPL/SQLパッケージの実行権限など)

簡単な使用例(2)
表作成

- 全文検索の対象となる表を作成(この表は、前ページで作成したデータベース・ユーザーが参照可能である必要がある。ここでは同一のuserctxユーザーで作成)

```
CONNECT userctx/userctx
CREATE TABLE texttab
  (id NUMBER PRIMARY KEY, text VARCHAR2(4000));
```

※ 全文検索の対象となる表の構造は任意です。また、主キーは必須ではありません。

- データをINSERT

```
INSERT INTO texttab VALUES (1, 'Oracle DBA & Developer Days 2011');
INSERT INTO texttab VALUES (2, '全文検索機能の活用法');
COMMIT;
```

簡単な使用例(3)
索引作成

- プリファレンス作成(プリファレンスとは、索引のメタ情報を指定するためのもの。詳細は後述。ここでは日本語レクサーのみを指定。)

```
BEGIN
  CTX_DDL.CREATE_PREFERENCE('jvl','JAPANESE_VGRAM_LEXER');
END;
/
```

- 索引作成(上記で作成したプリファレンスを指定して索引を作成)

```
CREATE INDEX textidx ON texttab (text)
  INDEXTYPE IS CTXSYS.CONTEXT
  PARAMETERS ('LEXER jvl');
```

※ 索引作成対象の列として指定可能なデータ型は、CHAR、VARCHAR、VARCHAR2、BLOB、CLOB、BFILE、XMLType、URIType のいずれかです。

簡単な使用例(4)
検索

- 検索を実行

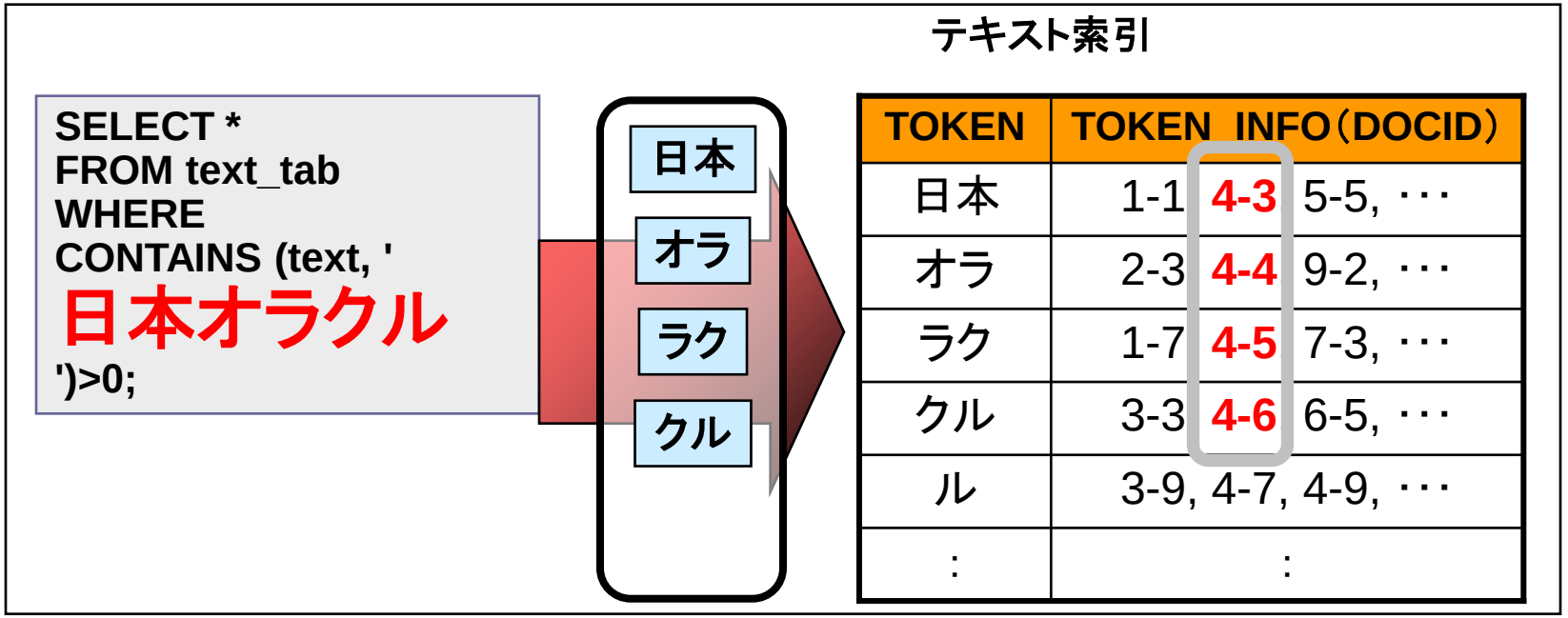
```
SQL> SELECT * FROM texttab  WHERE CONTAINS (text, '活用法') > 0;

        ID TEXT
---------- --------------------------------------------------
         2 全文検索機能の活用法
```

※ 検索にはCONTAINS関数を利用します。CONTAINS関数の返り値は0以上100以下の整数で、返り値が1以上のとき、全文検索の条件に一致することを示します。

※ CONTAINS関数は、必ずWHERE句の中で、「>0」の条件指定によって利用します。「>1」、「>75」など、0より大きい値を指定してはいけません。また、「=0」を指定してもいけません。

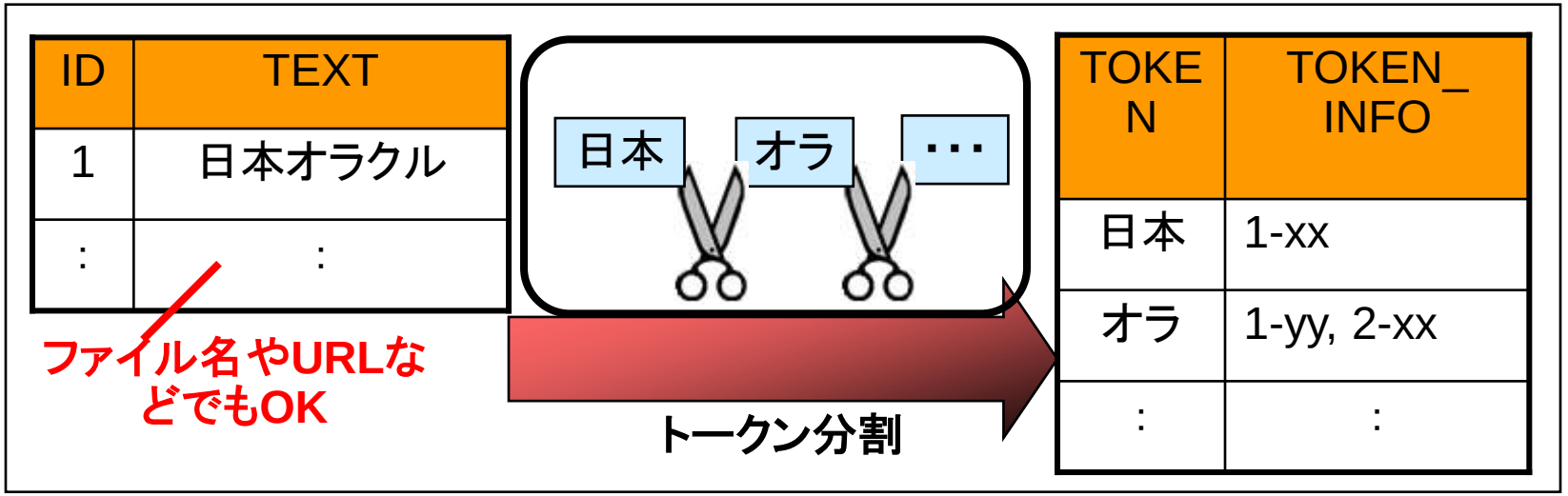
なぜOracle Textはそんなに速く結果を返すのか

- 全文検索専用のCONTEXT索引（以下、テキスト索引）を作成することで、検索対象ドキュメントをひとつひとつ検索するよりも高速に検索が行える

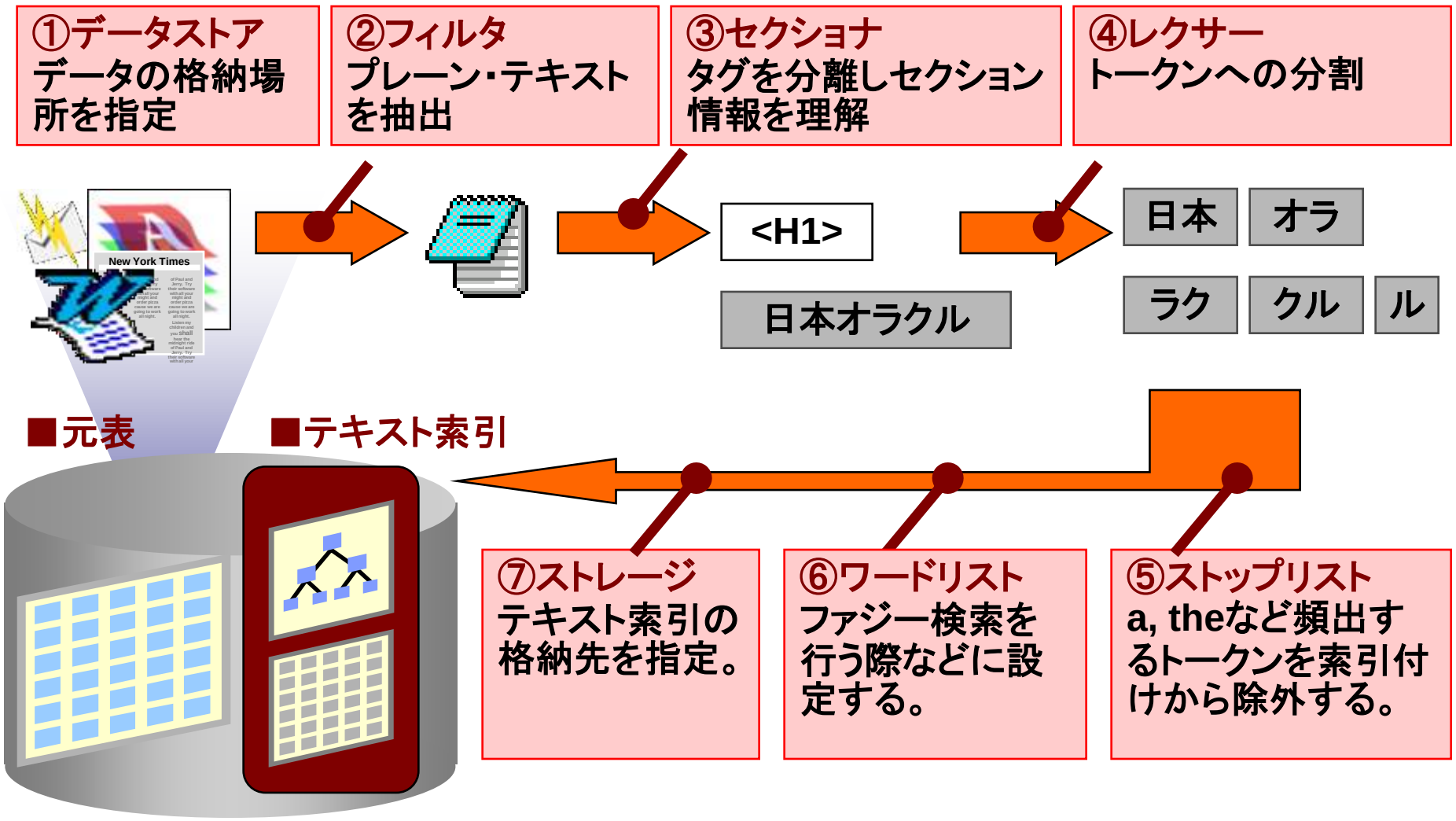


索引作成のメカニズム

- 文字列を「トークン」という細かい単位に区切り、出現位置情報を付けて「テキスト索引」に格納する
- WordやPDFをはじめ200種類以上のファイル形式、HTMLページなどにも索引付けできる

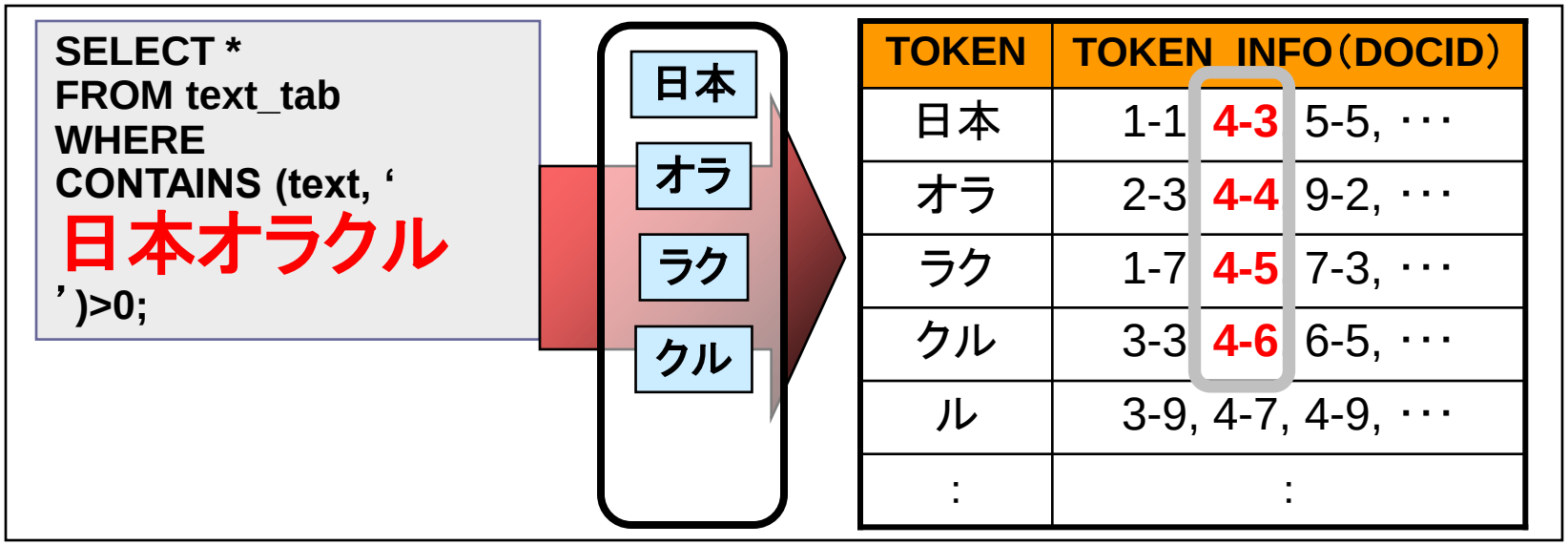


索引作成のメカニズム



検索のメカニズム

1. 索引付け時と同様のアルゴリズムで、検索文字列をトークン分割する
2. テキスト索引を問い合わせ、分割した各トークンの出現位置情報を取得する
3. 取得した出現位置情報から、「トークン分割順に連続する部分」を割り出す

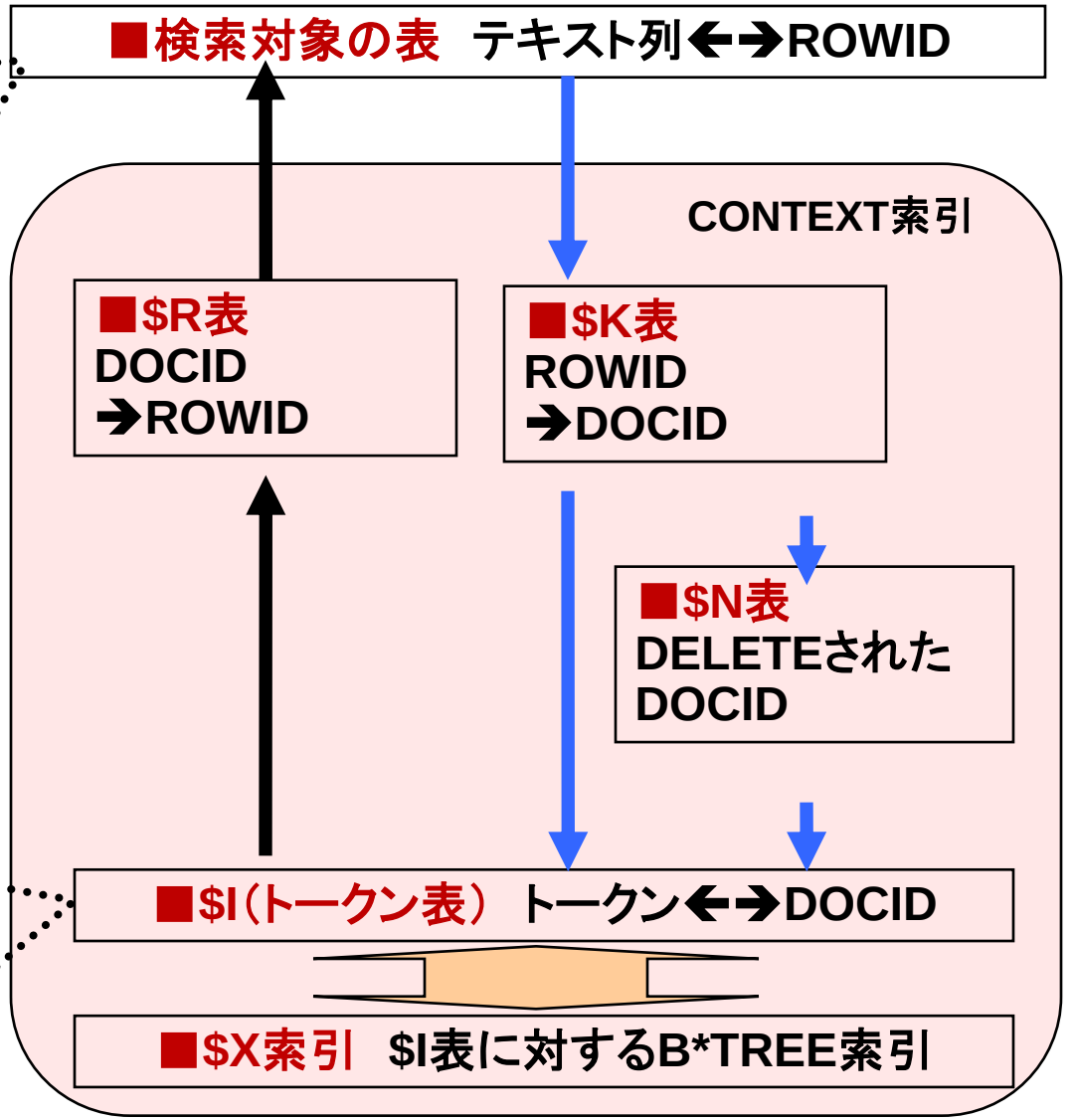


CONTEXT索引の構成要素

ID	TEXT
1	ORACLE
2	TEXT
3	ORACLE

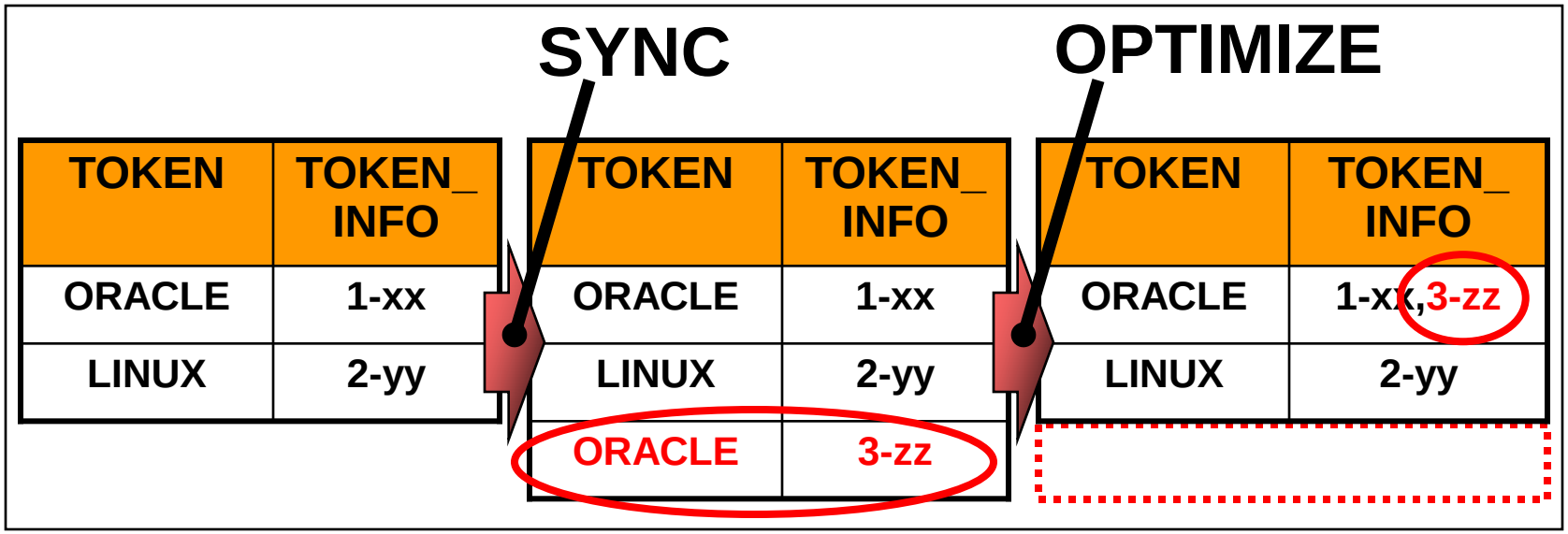
※ CONTEXT索引の実体は、右図のように4つの表と1つのB*TREE索引から構成される。元表の各行には、Oracle Textによって自動的にDOCIDが割り振られ、$I表にはDOCIDに基づくトークン情報が格納される。DOCIDとROWIDの対応付けは、$R表および$K表によって行われる。図の矢印は、検索時の処理の流れを示している。

トークン	出現位置
ORACLE	1-1 3-1
TEXT	2-1



索引メンテナンスのメカニズム

- 「SYNC」（同期化）を行うと、新たにINSERTされた行のトークンが索引の末尾に追加される
- 「OPTIMIZE」（最適化）を行うと、複数の同一トークン・エントリがマージされる



Agenda

- Oracle Text とは？
- Oracle Textによるアプリケーション開発
  - 全文検索のための演算子
  - ハイライト、スニペット
  - 複数列を対象にした検索
- XML検索
- 多言語検索
- チューニング
- 従来のデータベース機能との親和性

Oracle Text を利用した 全文検索アプリケーションの開発

- 基本的に全てSQLで開発
- 全文検索の条件は全てCONTAINS関数の引数で指定

CONTAINS問合せ演算子（抜粋）

論理演算子 AND OR ACCUM NOT MINUS MNOT

ワイルドカード
% _

近傍
NEAR

スコア関連 * >

シソーラス
SYN BT NT

等価
EQUIV

セクション
WITHIN INPATH HASPATH
MDATA SDATA NDATA

エスケープ文字 ¥ { }

ハイライト、スニペット

- 検索条件に一致した箇所を強調表示するには・・・？
  → CTX_DOCパッケージのプロシージャ、およびファンクションを利用する（これにより、全文検索の条件にヒットした箇所を特定のテキスト文字列（たとえば「<B>」と「</B>」）で囲むことができる）
- **テキスト索引がある場合**には次を利用
  ※ 通常は、問合せの後で処理するドキュメントを特定してから使用する
  - HIGHLIGHT
  - MARKUP
  - SNIPPET、SNIPPET_CLOB_QUERY
- **テキスト索引がない場合**には次を利用
  ※ 通常は、問合せの後で処理するドキュメントを特定してから使用する
  - POLICY_HIGHLIGHT
  - POLICY_MARKUP
  - POLICY_SNIPPET、POLICY_SNIPPET_CLOB_QUERY

CTX_DOC.SNIPPET関数の使用例

- 準備（表作成、索引作成）

```
1  CREATE TABLE testtab (id NUMBER PRIMARY KEY, text VARCHAR2(4000));
2  INSERT INTO testtab VALUES (1,'日本オラクル株式会社');
3  COMMIT;
4  BEGIN
5    CTX_DDL.CREATE_PREFERENCE('jvl','JAPANESE_VGRAM_LEXER');
6  END;
7  /
8  CREATE INDEX testidx ON testtab (text)
9    INDEXTYPE IS CTXSYS.CONTEXT PARAMETERS ('LEXER jvl');
```

- CTX_DOC.SNIPPET 関数の実行

```
1  SQL> SELECT CTX_DOC.SNIPPET('testidx',1,'オラ|株式') result
2    2    FROM testtab WHERE CONTAINS (text, 'オラ|株式') > 0
3
4  RESULT
5  --------------------------------------------------------------
6  日本<b>オラ</b>クル<b>株式</b>会社
```

CTX_DOC.POLICY_SNIPPET関数の使用例

- 準備（ポリシー作成）

```
1  BEGIN
2    CTX_DDL.CREATE_PREFERENCE('jvl','JAPANESE_VGRAM_LEXER');
3    CTX_DDL.CREATE_POLICY(
4      policy_name => 'my_policy',
5      lexer       => 'jvl'
6    );
7  END;
8  /
```

- CTX_DOC.POLICY_SNIPPET 関数の実行

```
1  SQL> SELECT CTX_DOC.POLICY_SNIPPET('my_policy','日本オラクル株式会
2  社','オラ|株式') result FROM DUAL;
3
4  RESULT
5  --------------------------------------------------------------
6  日本<b>オラ</b>クル<b>株式</b>会社
```

複数列を対象にした検索
MULTI_COLUMN_DATASTORE

- 表の複数列に対して、1つのテキスト索引を作成
  - COLUMNS属性に、検索対象の列名をコンマ区切りで列挙する
  - 列を指定した検索と、列を指定しない検索の両方を、1つのテキスト索引で実行するには、MULTI_COLUMN_DATASTOREとBASIC_SECTION_GROUPを組み合わせる
- 「CREATE INDEX <索引名> ON <表名>(<列名>)」で指定される列は、表の中の1列を指定するが、実際には COLUMNS属性で指定された列が検索対象となる
- 注意点
  - Oracle Text は、CREATE INDEX文で指定された列の更新のみを監視している
  - このため、CREATE INDEX文で指定された列以外で更新（UPDATE）が発生し、かつ、その更新を索引に反映したい場合には、CREATE INDEX文で指定された列についても同時にUPDATEする必要がある
  - この実装は、トリガーを使えばよい。このとき、CREATE INDEX文で指定された列に対する更新処理は、更新前と更新後のデータ内容が同一であっても問題ない
  - 行単位のDML処理（INSERT、DELETE等）については、特に考慮する必要なし

MULTI_COLUMN_DATASTOREの使用例（1）

- 表作成

```
1  CREATE TABLE multicoltest (
2    id     NUMBER  PRIMARY KEY,
3    col1 VARCHAR2(4000),  col2 VARCHAR2(4000),  col3 VARCHAR2(4000),
4    dummy  CHAR(1));
5  INSERT INTO multicoltest VALUES (1,'あい','うえ','お','0');
6  INSERT INTO multicoltest VALUES (2,'か','き','くけこ','0');
7  COMMIT;
```

- プリファレンス作成

```
1  BEGIN
2    CTX_DDL.CREATE_PREFERENCE('mcd','MULTI_COLUMN_DATASTORE');
3    CTX_DDL.SET_ATTRIBUTE('mcd', 'COLUMNS','col1,col2,col3');
4    CTX_DDL.CREATE_PREFERENCE('jvl','JAPANESE_VGRAM_LEXER');
5  END;
6  /
```

- 索引作成

```
1  CREATE INDEX multicolidx ON multicoltest (dummy)
2    INDEXTYPE IS CTXSYS.CONTEXT  PARAMETERS ('DATASTORE mcd LEXER jvl');
```

MULTI_COLUMN_DATASTOREの使用例(2)

- 検索

```
SQL> SELECT * FROM multicoltest WHERE CONTAINS (dummy, 'あい') > 0;

        ID COL1       COL2       COL3       D
---------- ---------- ---------- ---------- -
         1 あい       うえ       お         0
```

```
SQL> SELECT * FROM multicoltest WHERE CONTAINS (dummy, 'あいう') > 0;

レコードが選択されませんでした。
```

```
SQL> SELECT * FROM multicoltest WHERE CONTAINS (dummy, 'あい AND う') >
0;

        ID COL1       COL2       COL3       D
---------- ---------- ---------- ---------- -
         1 あい       うえ       お         0
```

MULTI_COLUMN_DATASTOREの使用例(3)
BASIC_SECTION_GROUPとの組み合わせ

- BASIC_SECTION_GROUP 用のセクション・グループを作成

```
1  BEGIN
2    CTX_DDL.CREATE_SECTION_GROUP('bsg', 'BASIC_SECTION_GROUP');
3    CTX_DDL.ADD_ZONE_SECTION('bsg', 'col1', 'col1');
4    CTX_DDL.ADD_ZONE_SECTION('bsg', 'col2', 'col2');
5    CTX_DDL.ADD_ZONE_SECTION('bsg', 'col3', 'col3');
6  END;
7  /
```

- 索引作成

```
1  --DROP INDEX multicolidx;
2
3  CREATE INDEX multicolidx ON multicoltest (dummy)
4    INDEXTYPE IS CTXSYS.CONTEXT
5    PARAMETERS ('DATASTORE mcd LEXER jvl SECTION GROUP bsg');
```

MULTI_COLUMN_DATASTOREの使用例(4)
BASIC_SECTION_GROUPとの組み合わせ

- 列名を指定した検索

```
SQL> SELECT * FROM multicoltest WHERE CONTAINS (dummy, 'あい WITHIN col1') > 0;

        ID COL1       COL2       COL3       D
---------- ---------- ---------- ---------- -
         1 あい       うえ       お         0
```

```
SQL> SELECT * FROM multicoltest WHERE CONTAINS (dummy, 'あい WITHIN col2') > 0;

レコードが選択されませんでした。
```

- 列名を指定しない検索(＝全ての列を対象とする検索)

```
SQL> SELECT * FROM multicoltest WHERE CONTAINS (dummy, 'くけ') > 0;

        ID COL1       COL2       COL3       D
---------- ---------- ---------- ---------- -
         2 か         き         くけこ     0
```

Agenda

- Oracle Text とは？
- Oracle Textによるアプリケーション開発
- XML検索
  - 概要
  - 使用例
  - INPATH、HASPATH演算子の動作
- 多言語検索
- チューニング
- 従来のデータベース機能との親和性

Oracle TextによるXML全文検索（1）

- 検索対象の列データ型は、XMLTypeである必要はない
  - CLOB、VARCHAR2、BFILE等であってもOK
- XMLTypeの物理的な格納方式のいずれも検索対象にできる
  - 非構造化記憶域（内部的にCLOB型）
  - バイナリXML記憶域（内部的にBLOB型）
  - 構造化記憶域（内部的にタグ値をTYPE属性にO/Rマッピング）
- PATH_SECTION_GROUPを利用して索引を作成
  - INPATHおよびHASPATH演算子を使用して検索を実行（XPathに準じたシンタックスによる検索）

Oracle TextによるXML全文検索（2）

- INPATHおよびHASPATH演算子を使用して検索を行う
- XPathに準じたシンタックスでの検索

表作成

```
1  CONNECT userctx/userctx
2  CREATE TABLE xmltab (id NUMBER PRIMARY KEY, xml XMLTYPE);
3  INSERT INTO xmltab VALUES
4    (1, XMLTYPE('<AAA><BBB>dummy</BBB><CCC>dummy</CCC></AAA>'));
5  COMMIT;
```

索引作成

```
1  BEGIN
2    CTX_DDL.CREATE_SECTION_GROUP('psg','PATH_SECTION_GROUP');
3  END;
4  /
5  CREATE INDEX xmlidx ON xmltab(xml)
6    INDEXTYPE IS CTXSYS.CONTEXT
7    PARAMETERS ('SECTION GROUP psg');
```

Oracle TextによるXML全文検索（3）

- INPATHの後に指定したXPath式の対象にキーワードが含まれている文書を返す

```
SELECT * FROM xmltab
  WHERE CONTAINS (xml, 'dummy INPATH(/AAA/BBB)') > 0;
```

- HASPATHの後に指定したXPath式の対象が存在する文書を返す

```
SELECT * FROM xmltab
  WHERE CONTAINS (xml, 'HASPATH(/AAA/BBB)') > 0;
```

INPATH演算子によるXMLタグ検索（1）

トップレベルのタグ検索

dog INPATH(A)	<A>dog</A>	○
	<B><A>dog</A></B>	×

任意レベルのタグ検索

dog INPATH(//A)	<A>dog</A>	○
	<B><A>dog</A></B>	○

直接の親子関係の検索

dog INPATH(A/B)	<A><B>My dog is friendly</B></A>	○
	<C><B>My dog is friendly</B></C>	×

INPATH演算子によるXMLタグ検索（2）

単一レベルのワイルド・カード検索

dog INPATH (A/*/B)	<A><D><B>dog</B></D></A>	○

マルチレベルのワイルド・カード検索

dog INPATH (A/*/*/B)	<A><C><D><B>dog</B></D></C></A>	○

任意レベルの子検索

dog INPATH (A//B)	<A><C><D><B>dog</B></D></C></A>	○

属性の検索

dog INPATH(//A/@B)	<C><A B="snoop dog"></A></C>	○

タグ値のテスト

dog INPATH(A[B="dog"])	<A><B>dog</B></A>	○
	<A><B>My dog is friendly</B></A>	✕

HASPATH演算子による検索

- 指定したセクション・パスを含むすべてのXMLドキュメントを検索する
- セクションの等価性をテスト

<A><B><C>という構造のタグが存在するかどうかを確かめたい時・・・

HASPATH(A/B/C)	<A><B><C>dog</C></B></A>	○

<A>dog</A>というタグが存在するかどうかを確かめたい時・・・

HASPATH(A="dog")	<A>dog</A>	○
	<A>dog park</A>	✕

Agenda

- Oracle Text とは？
- Oracle Textによるアプリケーション開発
- XML検索
- 多言語検索
  - Oracle Textの多言語対応
  - MULTI_LEXER
  - WORLD_LEXER
- チューニング
- 従来のデータベース機能との親和性

Oracle Textの多言語対応

- 言語が影響する要素
  - レクサーの動作（レクサーとは、検索対象のテキスト文字列から、索引情報を生成するための仕組み）
- 「多言語」のパターン
  - 1つの行データの中に、複数の言語が混在
    → WORLD_LEXERを利用
  - 1つの行データの中には、1つの言語しか存在しない
    → MULTI_LEXERを利用
  - 1つの表には、1つの言語しか存在しない
    → 多言語対応を意識しない、通常のレクサーを利用

※ データベースキャラクタセットによって、利用可能なレクサーは異なります。たとえば日本語レクサーであるJAPANESE_VGRAM_LEXERは、データベースキャラクタセットがAL32UTF8、UTF8、JA16SJIS、JA16EUC、JA16EUCTILDE、JA16EUCYEN、JA16SJISTILDE、JA16SJISYENの場合のみ動作します。

※ データベースキャラクタセットがAL32UTF8であれば、全てのレクサーが動作します。

MULTI_LEXER

- 多言語レクサーの1つ
- 言語とレクサー（SUB_LEXER）の対応付けを管理者が行う
- 各レコード（行）と言語の対応付けを明示的に行う（言語列が必要）
  - 1つのレコードに複数の言語が含まれていている場合、言語列で指定された言語に相当する部分のみが索引に格納される
- 日本語を扱える
  - 日本語用レクサーとしては、以下を指定可能
    - JAPANESE_VGRAM_LEXER（WORLD_LEXER）
      ※ 日本語文書に対する SUB_LEXER として、
      JAPANESE_VGRAM_LEXER を指定しても、WORLD_LEXER を指定しても、動作としては同じ
    - JAPANESE_LEXER（DB 9.0.1～）
    - USER_LEXER（DB 9.2～）

WORLD_LEXER
（Oracle Database 10g Release 1（10.1）以降）

- 多言語レクサーの1つ
- テキスト文字列を、そのコードポイントにより言語を自動判別してトークン分割を行う
  - 1つのレコードに複数の言語が含まれていても、すべての言語を正しく索引付け可能
  - MULTI_LEXER とは異なり、言語列を必要とせず、また、SUB_LEXER の設定も必要としない
  - Unicode 5.0 標準で定義されるほとんどの言語で動作（次スライドに対応言語一覧）
- 日本語を扱える
  - 日本語部分は JAPANESE_VGRAM_LEXER と同一の方法でトークンが生成される

WORLD_LEXER の対応言語一覧

言語グループ	含まれる言語
アラビア語	アラビア語、ファルシ語、クルド語、パシュトー語、シンド語、ウルドゥー語
アルメニア語	アルメニア語
ベンガル語	アッサム語、ベンガル語
Bopomofo	客家（ハッカ）語、ビンナン語
キリル語	ベラルーシ語、ブルガリア語、マケドニア語、モルダビア語、ロシア語、セルビア語、セルビア・クロアチア語、ウクライナ語を含む50以上の言語
デーヴァナーガリー文字	ボジュプリー語、ビハール語、ヒンディー語、カシミール語、マラーティー語、ネパール語、パーリ語、サンスクリット語
エチオピア語	アムハラ語、ゲーズ語、ティグリニャ語、ティグレ語
グルジア語	グルジア語
ギリシャ語	ギリシャ語
グジャラート語	グジャラート語、カッチ語
グルムキー語	パンジャブ語
ヘブライ語	ヘブライ語、ラディノ語、イディッシュ語
カガンガ文字	レジャン語
カンナダ語	カナラ語、カンナダ語
韓国語	韓国語、ハンジャ・ハングル語
ラテン語	アフリカーンス語、アルバニア語、バスク語、ブルトン語、カタロニア語、クロアチア語、チェコ語、デンマーク語、オランダ語、英語、エスペラント 語、エストニア語、フェロー語、フィジー語、フィンランド語、フラマン語、フランス語、フリジア語、ドイツ語、ハワイ語、ハンガリー語、アイスランド語、 インドネシア語、アイルランド語、イタリア語、ラップ語、古典ラテン語、ラトビア語、リトアニア語、マレー語、マルタ語、中国標準語（ピンイン表記）、マ オリ語、ノルウェー語、ポーランド語、ポルトガル語、プロヴァンス語、ルーマニア語、サモア語、ゲール語（スコットランド）、スロバキア語、スロベニア 語、ソルビア語、スペイン語、スワヒリ語、スウェーデン語、タガログ語、トルコ語、ベトナム語、ウェールズ語
マラヤーラム語	マラヤーラム語
モンゴル語	モンゴル語
オリヤー語	オリヤー語
シンハラ語	パーリ語、シンハラ語
シリア語	アラム語、シリア語
タミル語	タミル語
テルグ語	テルグ語
ターナ文字	ディベヒ語、モルディブ語
中国語	広東語、中国標準語、ピンイン表音文字
日本語	日本語（ひらがな、漢字、カタカナ）
クメール語	カンボジア語、クメール語
ラオ語	ラオ語
ミャンマー語	ビルマ語
タイ語	タイ語
チベット語	ゾンカ語、チベット語

http://download.oracle.com/docs/cd/E16338_01/text.112/b61357/amultlng.htm#CEGJBDEJ

Agenda

- Oracle Text とは？
- Oracle Textによるアプリケーション開発
- XML検索
- 多言語検索
- チューニング
  - 概要
  - 索引作成が遅い場合
  - 検索が遅い場合
  - 索引同期化が遅い場合
  - 索引最適化が遅い場合
- 従来のデータベース機能との親和性

Oracle Textのチューニング

- 基本的には、Oracle Databaseのチューニングと同じ
- PGAとSGA
  - 索引作成時、および索引メンテナンス時には、PGAを多く必要とする
  - 検索時には、SGAを多く必要とする
- 次ページ以降のスライドでは、Oracle Text に特化したチューニング手法を紹介

索引作成が遅い場合

- パラレルでの索引作成を検討する
  - パラレル索引作成は、非パーティション索引、パーティション索引のいずれに対しても有効
- Index Memoryを、物理メモリが許す限り大きく設定する
  - Index MemoryはPGA内に取られるため、PGAについても十分に大きく設定しておく必要がある
  - Index Memory は、各スレーブ・プロセス毎に取られる
    - 例:「MEMORY 256M」と「PARALLEL 4」を同時に指定した場合、Index Memory は合計で 256 MB × 4 = 1,024 MB となる
- 特定のドキュメントのフィルタ処理に時間がかかっている場合には、フィルタのタイムアウト値を設定する

検索が遅い場合（全般）

- 1つのSELECT文に含まれるCONTAINS句を1つにする
  - 全文検索が複数列にまたがる場合には、MULTI_COLUMN_DATASTOREを利用する
- SELECT文に、CONTAINS句以外の条件式や、ORDER BY句が含まれている場合、コンポジット・ドメイン索引の利用を検討する
- 検索結果を全件フェッチするようなアプリケーションになっている場合には、FIRST_ROWS(n) ヒントを指定した上で、結果画面表示に必要な最低限の行数をフェッチする動作に変更する
- SELECT文で、CONTAINS句の条件と同時に、他表との結合処理が含まれる場合には、マテリアライズド・ビューの利用を検討する
  - 他表と結合した状態のマテリアライズド・ビューを作成し、このマテリアライズド・ビューに対してテキスト索引を作成する
  - これによって、SELECT文から結合処理を除外する

検索が遅い場合（索引断片化）

- 索引作成時、索引同期化時のIndex Memoryを、物理メモリが許す限り大きく設定する（これにより、索引の断片化を防ぐ）
- 索引同期化の頻度を出来る限り小さくする（これにより、索引の断片化を防ぐ）
  - 索引の同期化を繰り返すことによって、索引が断片化し、検索パフォーマンスが劣化する
    → 索引の最適化を実行することで、検索パフォーマンスを回復できる
- 索引が断片化している場合には、FULLモード、あるいはREBUILDモードでの索引最適化を実行する

検索が遅い場合（日本語）

※ 内部的に一文字検索が発生するパターンの詳細に関しては、参考資料『Oracle Text 詳細解説』p.128～130を参照

- JAPANESE_VGRAM_LEXERで、特定のキーワードでの検索が遅い場合
  - レクサーを変更する
    - JAPANESE_LEXER＋レキシコンのカスタマイズ
      ※ Oracle Database 10g Release 1（10.1）以降で利用可能
- 内部的に一文字検索が発生している場合（※）
  - BASIC_WORDLISTのPREFIX_INDEX属性をTRUEに設定する
    ※ ただし、この方法ではそれほど改善効果が見込めない場合がある
  - データ蓄積時に文字列置換を行い、一文字検索を回避する
    ※ 検索時にも同様の文字列置換を検索アプリケーション側で行った上で検索を実行する
    - 例1：アラビア数字を漢数字に置き換え
    - 例2：アルファベット文字を特定の漢字に置き換え
- 日本語文書に含まれる英数字のワイルドカード検索が遅い場合
  - BASIC_WORDLISTのSUBSTRING_INDEX属性をTRUEに設定する

索引同期化が遅い場合

- パラレルでの索引同期化を検討する
  - パラレルでの索引同期化は、非パーティション索引、パーティション索引のいずれに対しても有効
- Index Memoryを、物理メモリが許す限り大きく設定する
  - Index MemoryはPGA内に取られるため、PGAについても十分に大きく設定しておく必要がある
  - Index Memory は、各スレーブ・プロセス毎に取られる
    - 例:「MEMORY 256M」と「PARALLEL 4」を同時に指定した場合、Index Memory は合計で 256 MB × 4 = 1,024 MB となる
- 特定のドキュメントのフィルタ処理に時間がかかっている場合には、フィルタのタイムアウト値を設定する

索引最適化が遅い場合

- パラレルでの索引最適化を検討する
- REBUILDモードでの索引最適化を検討する
  ※ REBUILDモードでの索引最適化はOracle Database 10g Release 1（10.1）以降で利用可能。
  - 索引が断片化が進むと、FULLモードでの最適化に、極端に長い時間がかかる場合がある
  - この場合、REBUILDモードでの最適化を利用すると、新規に索引を作成するのと同程度の時間で最適化を完了できる
  - ただし、REBUILDモードでの最適化では、一時的に新旧2つの索引が同時に保持されるため、索引用のディスク領域が一時的に2倍必要となる

Agenda

- Oracle Text とは？
- Oracle Textによるアプリケーション開発
- XML検索
- 多言語検索
- チューニング
- 従来のデータベース機能との親和性
  - 全文検索エンジン＝Oracle Databaseであることと、その価値
  - システムの全体最適 / 開発工数 / 信頼性

全文検索エンジン＝Oracle Databaseであることと、その価値

- Oracle Textは、Oracle Database カーネルで実装された全文検索エンジン
- Oracle Database の既存機能との親和性が非常に高い
  - RAC（Oracle Real Application Clusters）
  - Partitioning（テキスト索引を、ローカルパーティション索引として作成できる）
  - Advanced Compression（SecureFile圧縮したデータを検索対象にする）
  - Advanced Security（暗号化したデータを検索対象にする）
  - XML DB（XMLType型を検索対象にする）
  - パラレル・クエリ、Data Guard、etc.
- Oracle Exadata 上で利用可能
  ※ 2件の導入実績あり（2011年11月現在）

システムの全体最適 / 開発工数 / 信頼性

- DB層での全文検索は、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」）
- Oracle Textを利用することで、全文検索アプリケーション開発（の本質的な部分）は、SQL文の記述だけで完結する
- 全文検索と定型検索の条件を、SELECT文のWHERE句で簡単に組み合わせられる
- Oracle Text 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

まとめ

- 簡単な実装
  - SQLだけで全文検索を実行できる
  - 強調表示（スニペット）についてもDB層で処理可能
  - XML検索、多言語検索などの複雑な処理も簡単に実装できる
  - 全文検索と定型検索の組み合わせが容易

  → システム全体で必要となる開発工数の削減が見込める
- ハードウェア資産の有効活用
  - DB層での全文検索処理は、ネットワーク使用やプロセス間通信が発生せず、システム全体で必要となるハードウェア・リソースの削減が見込める
- ソフトウェア資産の有効活用
  - 追加のオプション・ライセンスなしで使える（特に、Express Edition、Standard Edition One、Standard Editionでも利用可能）
- 信頼性、可用性、スケーラビリティ、管理の容易さ
  - Oracle Database の特長をそのまま継承して、堅牢な全文検索システムを構築できる

参考(1)

- 製品マニュアル
  - Oracle Text リファレンス
    http://download.oracle.com/docs/cd/E16338_01/text.112/b61357/toc.htm
  - Oracle Text アプリケーション開発者ガイド
    http://download.oracle.com/docs/cd/E16338_01/text.112/b61358/toc.htm
- OTN(Oracle Technology Network)
  - Oracle Text
    http://www.oracle.com/technetwork/jp/database/enterprise-edition/index-086432-ja.html
  - Oracle Text パフォーマンスFAQ
    http://otndnld.oracle.co.jp/products/iserver/oracle9i/htdocs/o9i_920_otpf_10_1928.html
  - OTNセミナー オンデマンド コンテンツ
    http://www.oracle.com/technetwork/jp/ondemand/db-technique/index.html#Content02
    - Oracle Text 概要(動画あり)
    - Oracle Text 詳細解説(動画なし)

参考(2)

- oracletech.jp ～Oracle Databaseでの全文検索の仕組みと動き
  http://oracletech.jp/products/pickup/000257.html
  ※前ページのスライドとこのスライドで紹介されている資料へのリンクは、上記のURLにまとまっています。
- [書籍]日下部明、他 著『これは使えるOracle新機能活用術』(翔泳社、2009年6月16日、ISBN: 9784798119915)
  → 第10章「Oracleカーネルの高速な全文検索機能」(p.180～199)

OTNセミナーオンデマンド

コンテンツに対する
ご意見・ご感想を是非お寄せください。

OTNオンデマンド　感想

http://blogs.oracle.com/oracle4engineer/entry/otn_ondemand_questionnaire

上記に簡単なアンケート入力フォームをご用意しております。

セミナー講師/資料作成者にフィードバックし、
コンテンツのより一層の改善に役立てさせていただきます。

是非ご協力をよろしくお願いいたします。

OTNセミナーオンデマンド

日本オラクルのエンジニアが作成したセミナー資料・動画ダウンロードサイト

ORACLE TECHNOLOGY NETWORK
OTNセミナー オンデマンド

掲載コンテンツカテゴリ(一部抜粋)
Database 基礎
Database 現場テクニック
Database スペシャリストが語る
Java
WebLogic Server/アプリケーション・グリッド
EPM/BI 技術情報
サーバー
ストレージ

100以上のコンテンツをログイン不要でダウンロードし放題

データベースからハードウェアまで充実のラインナップ

毎月、旬なトピックの新作コンテンツが続々登場

例えばこんな使い方

- 製品概要を効率的につかむ
- 基礎を体系的に学ぶ/学ばせる
- 時間や場所を選ばず（オンデマンド）に受講
- スマートフォンで通勤中にも受講可能

毎月チェック！

コンテンツ一覧 はこちら
http://www.oracle.com/technetwork/jp/ondemand/index.html

新作＆おすすめコンテンツ情報 はこちら
http://oracletech.jp/seminar/recommended/000073.html

OTNオンデマンド

ORACLE

オラクルエンジニア通信

オラクル製品に関わるエンジニアの方のための技術情報サイト

技術資料 — インストールガイド・設定チュートリアルetc. 欲しい資料への最短ルート

アクセスランキング — 他のエンジニアは何を見ているのか？人気資料のランキングは毎月更新

特集テーマ Pick UP — 性能管理やチューニングなど月間テーマを掘り下げて詳細にご説明

技術コラム — SQLスクリプト、索引メンテナンスetc. 当たり前の運用/機能が見違える!?

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

オラクルエンジニア通信

oracletech.jp

ITエンジニアの皆様に向けて旬な情報を楽しくお届け

製品/技術情報
Oracle Databaseっていくら？オプション機能も見積れる簡単ツールが大活躍

セミナー
基礎から最新技術までお勧めセミナーで自分にあった学習方法が見つかる

スキルアップ
ORACLE MASTER！試験頻出分野の模擬問題と解説を好評連載中

Viva! Developer
全国で活躍しているエンジニアにスポットライト。きらりと輝くスキルと視点を盗もう

http://oracletech.jp/

oracletech

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

Oracle Direct

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。
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.co.jp/inq_pl/INQUIRY/quest?rid=28

※フォームの入力にはログインが必要となります。
※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
ご登録の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

Hardware and Software
Engineered to Work Together


ORACLE®