

Let Your APEX Application Find Everything

"Fuzzy Search" with Oracle Text

Dimitri Gielis
APEX R&D

12c

ORACLE®

Copyright © 2014 Oracle and/or its affiliates. All rights reserved. |

Search challenges in Business Applications

- Need to search based on different criteria
- Need fuzzy / error-tolerant searching
- Need performance also on huge amounts of data
- Need an easy interface



Google Search

I'm Feeling Lucky

Oracle TEXT

Oracle TEXT: What is that?

- Fulltext index engine within the Oracle Database
- Part of all database editions – no separate license required
- Provides linguistic and fuzzy search features
- Searches documents and relational data

Let's start simple ...

1. Search documents

Search Documents

1. Have documents in your table
 - XMLTYPE, VARCHAR2, CLOB, BLOB
 - Filtering available for binary data (Office, PDF)
2. Create a text for your table

```
create index ix_text_index on docs_tab (doc)
indextype is CTXSYS.CONTEXT
parameters ('FILTER CTXSYS.AUTO_FILTER')
```

3. Create the APEX report

```
select * from docs_tab
where contains(doc, :P1_SEARCH) > 0
```

 Oracle TEXT Sample Application Log Out

 Home

 Upload Documents

 Text Retrieval

 Customers Power Search

 Index Maintenance

Text Retrieval



[Geodaten_final.doc](#)
Engine" (Abbildung 6). Diese **XML**-Anfrage enthält alles, was
Document Size: 96KB

[RSS_und_OracleDB.doc](#)
URL), die eine bestimmte **XML**-Datei bereitstellt....description> </item> : Listing 1:
XML-Code des RSS-Feeds „DOAG Termine
Document Size: 67KB

[SQLInjection.pdf](#)

Full text search with SQL: Functionality

- SQL CONTAINS-Function
- Relevance-Ranking based on token frequency: SCORE
- Visualizuation
 - Markup,
 - Highlighting
 - "Keyword-in-Context"

```
select score(1), doc
from docs_tab
where CONTAINS(
    doc,
    'Software AND Oracle',
    1
)>0
```

Oracle TEXT Queries

- Combine Query Terms

... where contains(text, 'dog AND cat') >0

- Use Wild Cards

... where contains(text, 'do% AND c_t') >0

- Powerful Name Searching

... where contains(text, 'NDATA(name, lary elisn)') >0

- Fuzzy matching

... where contains(text, '?apeks') >0

- Multi Lingual Stemming

... where contains(text, '\$runs') >0

Oracle TEXT Queries

- NEAR Operator

... where contains(text, 'near(dog, cat), 4') >0

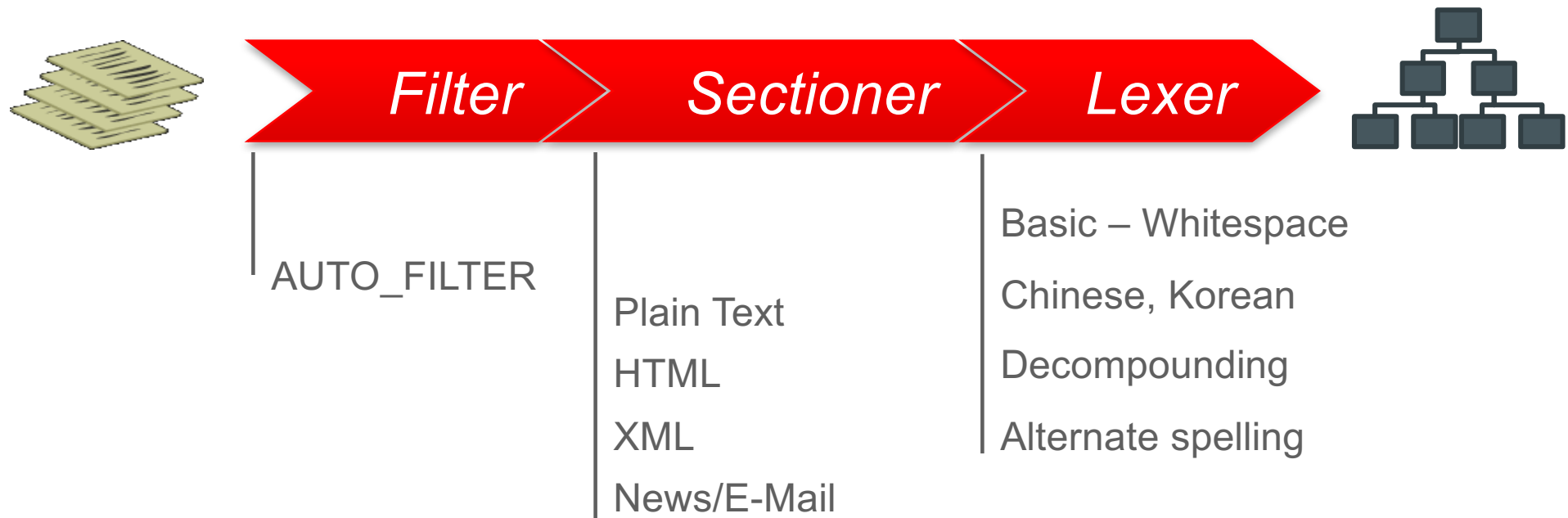
- Search within Sections or Paragraphs (XML!)

... where contains(text, 'dog WITHIN TITLE') >0

- Thesaurus support

... where contains(text, 'SYN(dog,[thes]' >0)

Oracle TEXT Indexing process



Oracle TEXT: Index Data Storage

Table	Contents
DR\$... \$I	Token table: Contains all Tokens and information about their occurrence in documents
DR\$... \$K	Mapping table ROWID → DOCID
DR\$... \$N	<i>"Deleted list"</i> : Contains all deleted DOCIDs
DR\$... \$R	Mapping table DOCID → ROWID
DR\$... \$P	<i>Substring index to support "%{TOKEN}" queries</i>
DR\$... \$S	SDATA table: contains structured data for queries with <, > or BETWEEN
DR\$... \$E	Oracle12c: <i>XQuery fulltext</i> – contains XML namespace information
DR\$... \$G	Oracle12c: <i>Near Realtime Index</i>
DR\$... \$D	Oracle12c: <i>Save Copy feature</i>

Configure the Oracle TEXT Index: Preferences

- Preferences are parameter collections for Oracle TEXT indexes
- Maintenance with the CTX_DDL package
 - CREATE_PREFERENCE
 - SET_ATTRIBUTE
 - DROP_PREFERENCE
- Different preference types for different purposes
 - DATASTORE, STORAGE, LEXER, WORDLIST, SECTION GROUP ...

Lexer Preferences: An example for the German language

```
begin
  CTX_DDL.create_preference('DE_LEXER', 'BASIC_LEXER');
  CTX_DDL.set_attribute('DE_LEXER', 'ALTERNATE_SPELLING', 'GERMAN');
  CTX_DDL.set_attribute('DE_LEXER', 'COMPOSITE', 'GERMAN');
  CTX_DDL.set_attribute('DE_LEXER', 'MIXED_CASE', 'NO');
  CTX_DDL.set_attribute('DE_LEXER', 'BASE_LETTER', 'YES');
  CTX_DDL.set_attribute('DE_LEXER', 'INDEX_STEMS', 'GERMAN');
end;
```

Create a TEXT index using Preferences

```
create index ft_employee_suche on employees_search(last_name)
indextype is ctxsys.context
parameters('
  datastore      employee_ds
  section group  employee_sg
  wordlist       employee_wl
  lexer         employee_lx
  stoplist      ctxsys.empty_stoplist
  memory 500M
')
/
```

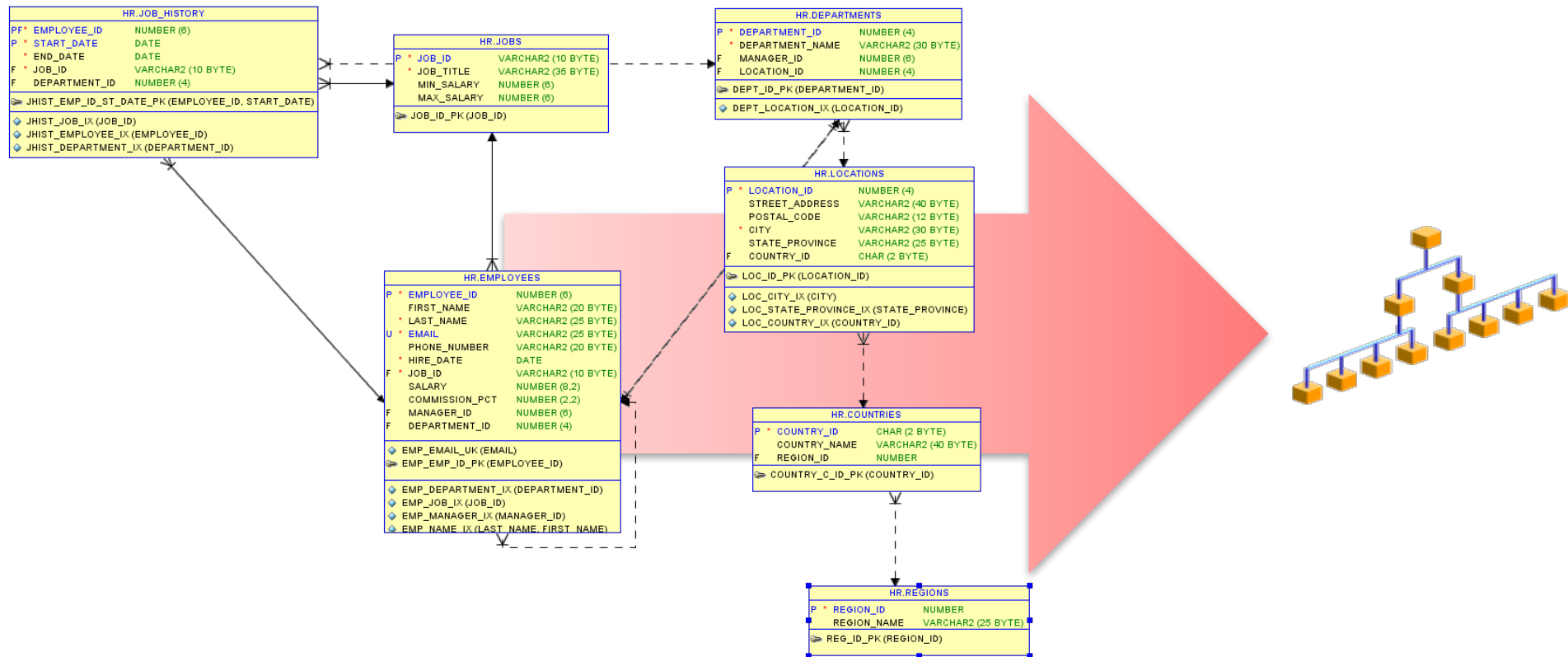
Two important things ...

- A text index is *asynchronous*. DML is not reflected immediately
 - Index synchronization required
 - On Commit, Manual (CTX_DDL.SYNC_INDEX), Regular intervals (DBMS_SCHEDULER)
 - Optimization in Oracle12c: *Near Realtime Index*
- A text index has a complex structure and needs maintenance
 - Index *fragmentation* leads to decreased performance
 - Monitoring and Index optimization required

OK, can search documents, now let's move on ...

2. Search table data

Oracle TEXT for relational data ...?

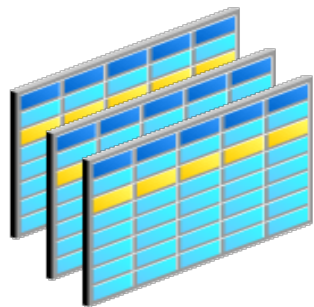


Oracle TEXT Index on plain relational data ...?

- Oracle TEXT wants to index *documents*
- Oracle TEXT **datastore preference** determines where these come from
 - DIRECT_DATASTORE (indexed column) is the default
 - FILE_DATASTORE, URL_DATASTORE for external sources
 - MULTICOLUMN_DATASTORE grabs data from other table columns
 - **USER_DATASTORE** for a "PL/SQL" source
- We can turn relational data to a "virtual" document with PL/SQL

How does a USER_DATASTORE work ...?

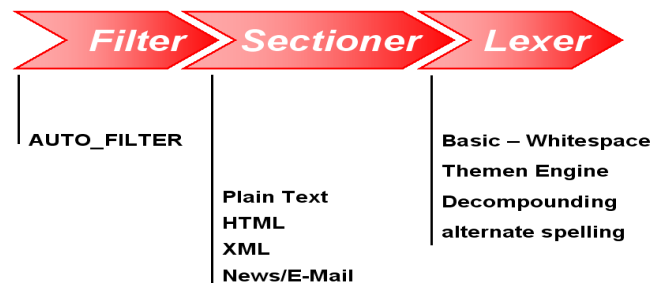
- A PL/SQL procedure collects the data to be indexed



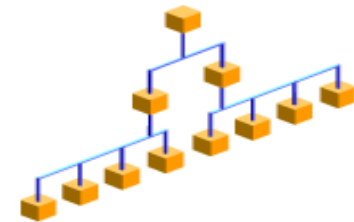
Table(s)



PL/SQL



Oracle TEXT



Index

DEMO

Step 1: Create separate "search table"

Create a separate search table

- This is not a must – you *can* do this
- Advantages
 - Search result list is a simple SQL query – no Joins, no Computations
 - Strict separation of full text search from relational schema
 - Full text maintenance does not affect other operations
 - Search functionality can stay online even when tables are in maintenance
- Disadvantages
 - Sync between data tables and search table is required → Programming
 - Space allocation

DEMO

Step 2: Create User Datastore procedure

PL/SQL based USER_DATASTORE: Some hints and tips

- You have multiple interfaces
 - Return VARCHAR2 → FAST!
 - Return a CLOB
 - Return a BLOB → has to be "filtered" like PDF or Office documents
- Your procedure will be called for each table row, so be fast!
 - You know, you are below 32K ...? Use VARCHAR2, not CLOB!
 - Using CLOB? Initialize *one* temp LOB, hold in a package and re-use
 - Write the most efficient SQL you can
 - Write fast, not nice code, i.e. generate XML by String concatenation.

Oracle TEXT User Datastore: Things to consider

- We created a ***search table***. How to keep it in sync with the data?
 - Trigger, Manual, Regular intervals with DBMS_SCHEDULER
 - Oracle TEXT index must also be synchronized
- How current is my "Search Engine" ...?
 - Regular sync
 - Simple processes – data changes are
 - Can perform powerful search on all data
 - Data changes are not immediately reflected
 - How long does it take until Google finds your Website ...?

DEMO

Step 3: Create Oracle TEXT preferences

DEMO

Step 4: Create Oracle TEXT index



Neena Kochhar

Administration Vice President
(Executive) in Seattle

Phone: 515.123.4568
E-mail: NKOCHHAR
City: Seattle, 98199
Manager: Steven King

NK

Nancy Greenberg

Finance Manager (Finance) in Seattle

Phone: 515.124.4569
E-mail: NGREENBE
City: Seattle, 98199
Manager: Neena Kochhar

NG

Hermann Baer

Public Relations Representative (Public
Relations) in Munich

Phone: 515.123.8888
E-mail: HBAER
City: Munich, 80925
Manager: Neena Kochhar

HB

Shelley Higgins

Accounting Manager (Accounting) in
Seattle

Phone: 515.123.8080
E-mail: SHIGGINS
City: Seattle, 98199
Manager: Neena Kochhar

SH

Jennifer Whalen

Administration Assistant
(Administration) in Seattle

Phone: 515.123.4444
E-mail: JWHALEN

JW

Susan Mavris

Human Resources Representative
(Human Resources) in London

Phone: 515.123.7777
E-mail: SMAVRIS

SM

Neena Kochha
Administration V
(Executive) in Sea
Phone: 515.123.4
E-mail: NKOCHH
City: Seattle, 9819
Manager: Steven

Jennifer Whalen
Administration A
(Administration) i
Phone: 515.123.4
E-mail: JWHALEN

```
select first_name || ' ' || last_name as card_title,
       :
from employees_search
where contains(
    last_name,
    '<query>
    <textquery>
    <progression>
      <seq>definescore((neen), RELEVANCE*5) within (FULL_NAME)</seq>
      <seq> (neen%) within (FULL_NAME)</seq>
      <seq>?(neen) within (FULL_NAME)</seq>
      <seq>NDATA(neen WITHIN (ND_FULL_NAME)</seq>
      <seq> neen</seq>
      <seq> (neen%)</seq><seq>?(neen)</seq>
    </progression>
    </textquery>
    </query>', 1
) > 0 and rownum <= 50 order by score(1) desc
```



dgielis@apexrnd.be

<https://www.apexrnd.be>

Based on presentation of Carsten Czarski of Oracle

ORACLE®

Copyright © 2014 Oracle and/or its affiliates. All rights reserved. |