



Exploring the IBM OmniFind Text Search Server

Kent Milligan
IBM i – ISV Enablement

Sponsored by:

Homerun

all the tools you'll ever need to tune DB2 for i



IBM Information Management software

IBM OmniFind Text Search Server for DB2 for i

- New IBM i product offering: 5733-OMF
 - **No-charge offering**
 - **Requires IBM i 6.1**
- Delivers common DB2 Family text search technology
 - **Advanced, linguistic high-speed searches**
 - **Support enabled for any character-based column**
 - **Search technology also supports Rich Text document formats**
 - Example: LOB columns containing PDF or Microsoft® Word documents
 - IFS documents can be indexed with extra programming
 - **Includes support for 26 different languages**

OmniFind Search Capabilities

- Example Searches:

```
SELECT author, story FROM books
WHERE CONTAINS (story, 'mice chasing cats') = 1
AND pubDate >= '1/1/2000'
```

```
SELECT feedSrc, feedDate,
       SCORE (feedDoc, 'insurance settlement')
FROM newsfeeds
WHERE CONTAINS(feedDoc, 'insurance settlement') = 1
ORDER BY 3 DESC
```

Database Requirements

- Table must have primary key, unique key constraint or ROWID column
 - Physical files are supported
 - Unique-keyed physical file requires usage of ADDPFCST to register key as constraint
- Supported column data types:
 - BINARY, VARBINARY
 - BLOB
 - CHAR, VARCHAR
 - CLOB
 - DBCLOB
 - GRAPHIC, VARGRAPHIC

Database Requirements

- Supported document format types:
 - Plain text
 - XML
 - HTML
 - Adobe PDF
 - Rich Text Format (RTF)
 - JustSystems Ichitaro
 - Microsoft Excel
 - Microsoft PowerPoint
 - Microsoft Word
 - Lotus® 123
 - Lotus Freelance®
 - Lotus WordPro
 - OpenOffice 1.1 & 2.0
 - OpenOffice Calc
 - Quattro Pro
 - StarOffice Calc

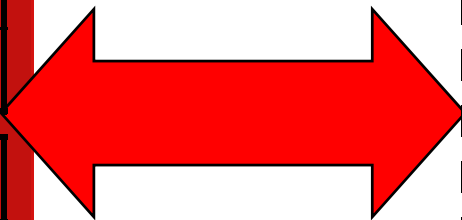
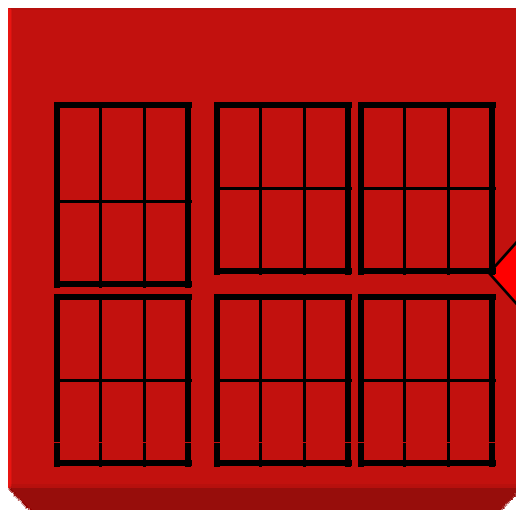
Text Index vs DB2 Index

- Text indexes stored outside of DB2
- Text indexes have delayed maintenance
 - Changes logged to staging table
 - Index maintenance is scheduled
- Text indexes not protected by IBM i SMAPP
- Text indexes utilize different indexing methods
 - Table with VARCHAR(32000) column and 175,000 rows
 - DB2 Index object size: 1.7 GB
 - Text Index object size: 0.1 GB

Text Search Server Configuration

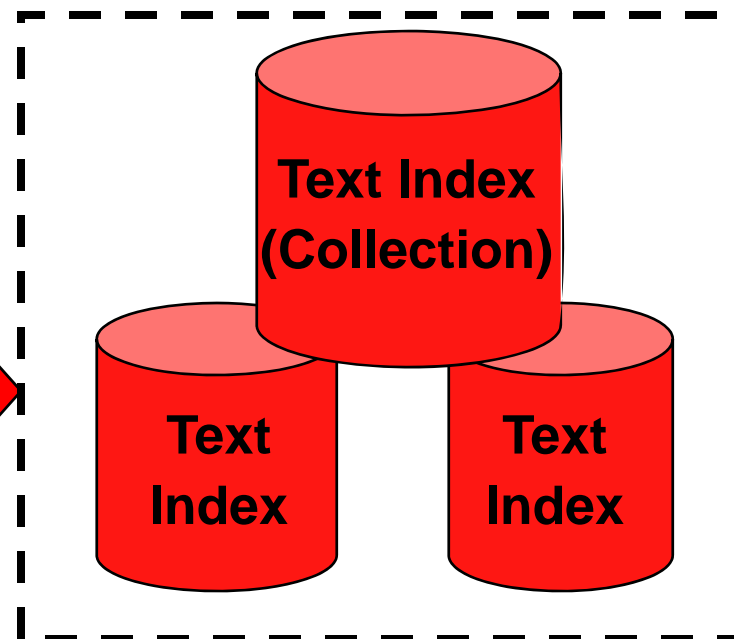
- OmniFind directory structure created during product install

DB2 for i



IFS

/QOpenSys/QIBM/ProdData/TextSearch/



Configuration & Administration Interfaces

- System stored procedures in SYSPROC
 - Starting and ending the text search server
 - SYSTS_START
 - SYSTS_STOP
 - Creating, maintaining, and dropping a text search index
 - SYSTS_CREATE
 - SYSTS_UPDATE
 - SYSTS_DROP

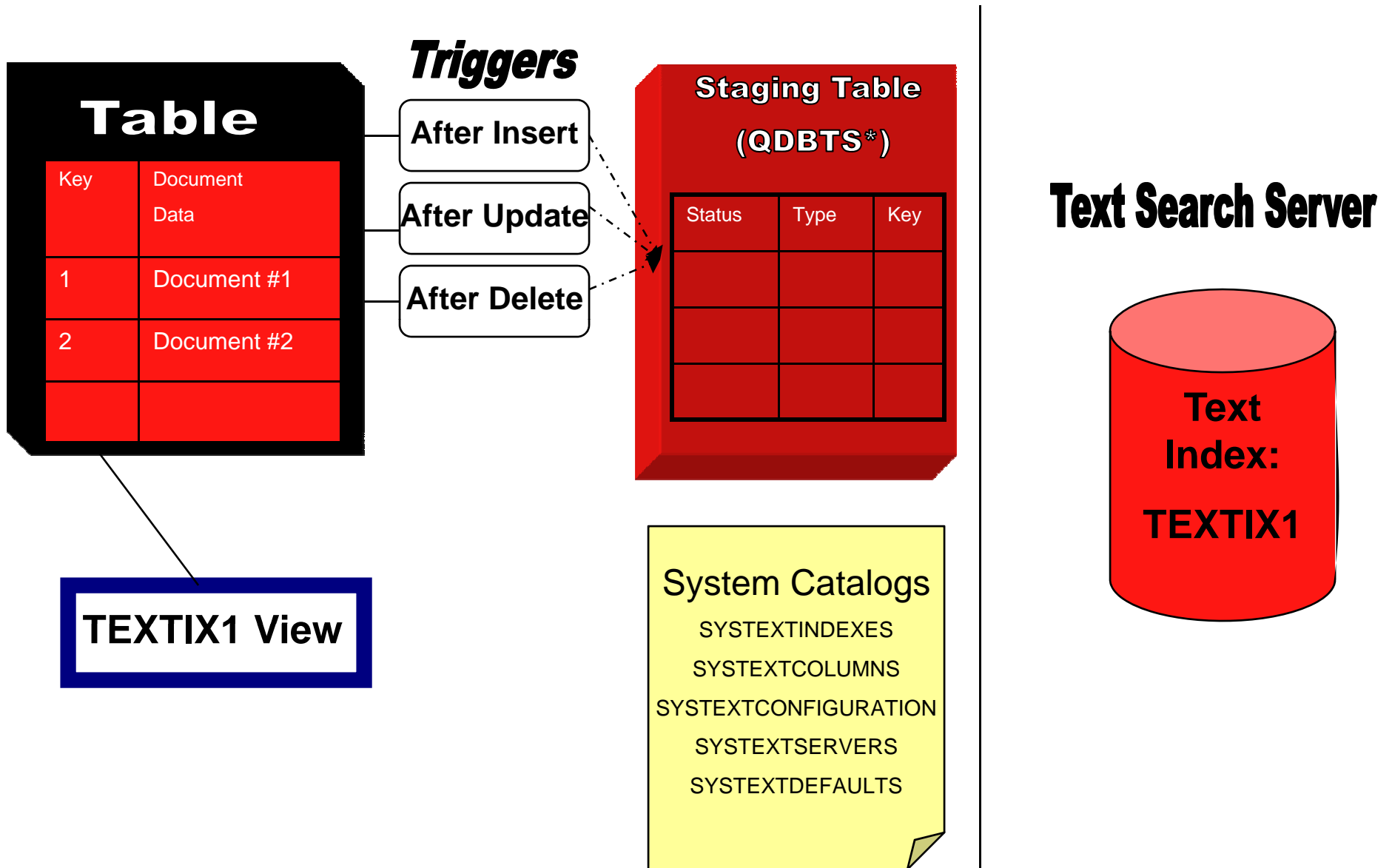
Start and Stop Text Search Processing

- `systs_start(servernum)`
 - Starts the specified text search server
 - Default is all local servers
 - This function must be called before using other text search function.
- `systs_stop(servernum)`
 - Stops the specified text search server
 - Default is all local servers
 - Changes are still logged in the staging table, even when the server is stopped.

Text Index Creation

- A text index is created by a call to SYSTS_CREATE
 - Object created in IFS text server directory
 - Text index initially has no data
- When a Text Index is created, the following DB2 objects are created:
 - A view is created with the same name as the text search index
 - A staging table is created in the QSYS2 schema
 - After Insert, After Update, and After Delete triggers are added to the base table
 - The system catalogs are updated with information about the new index

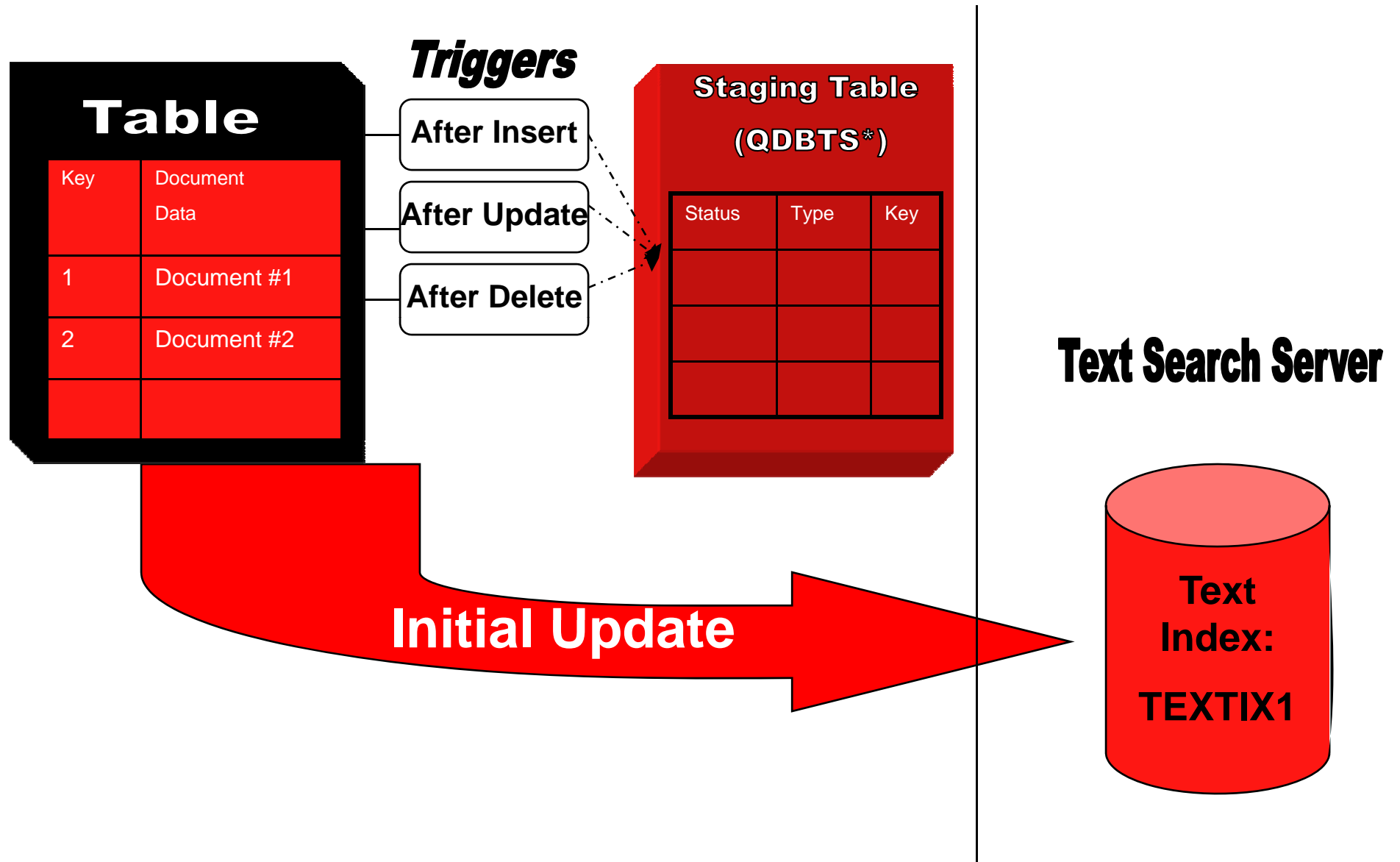
Objects Created by SYSTS_CREATE for a Text Index – TEXTIX1



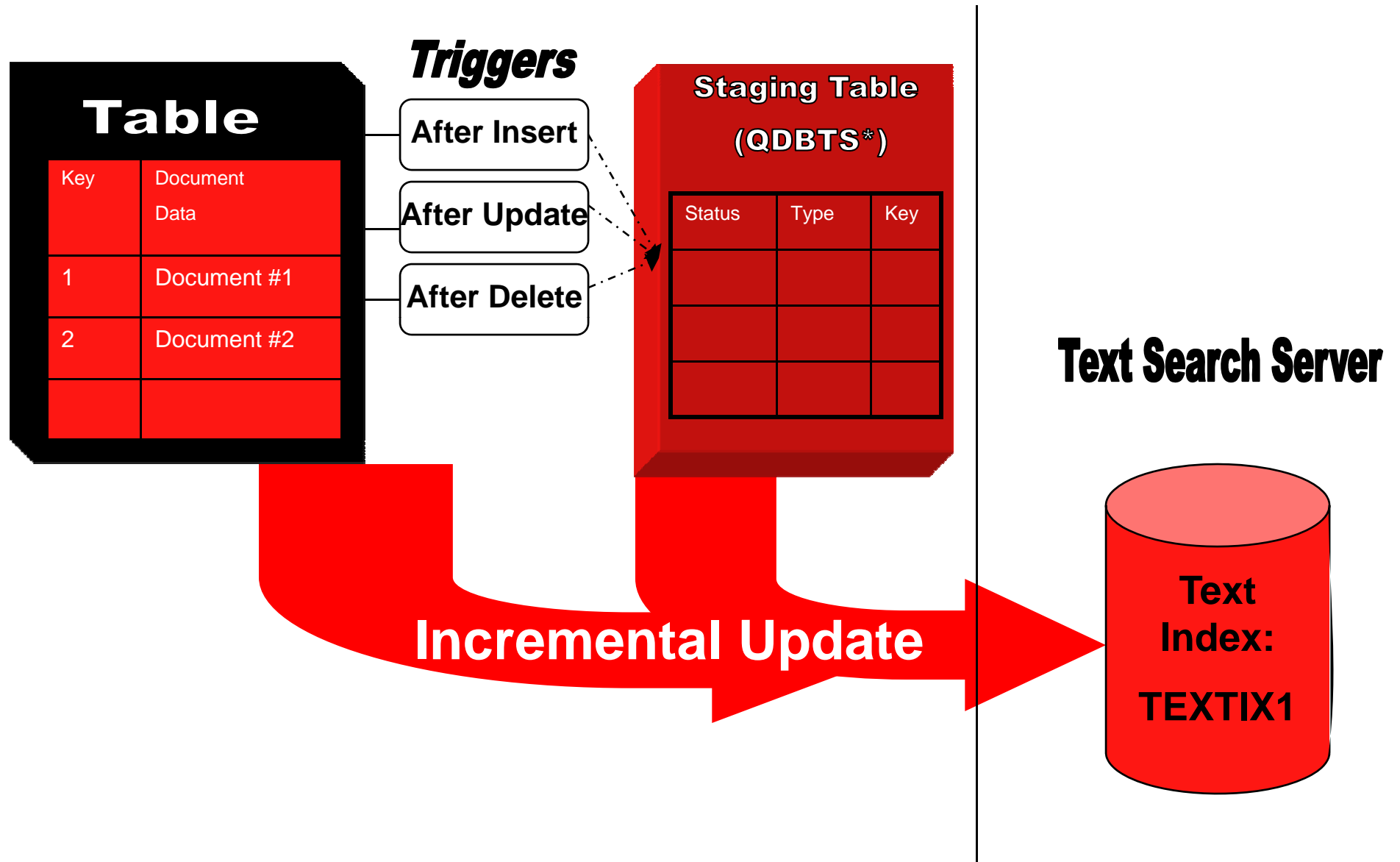
Text Index Maintenance

- Index maintenance controlled with SYSTS_UPDATE procedure
 - Initial update processes all text strings or text documents
 - Future index updates are incremental
 - Changes logged in the staging table are processed
 - IBM i Job Scheduler entries created to perform incremental updates
 - Minimum update threshold can be specified at creation time to only perform index updates when threshold exceeded

SYSTS_UPDATE Initial Update for a Text Index – TEXTIX1



SYSTS_UPDATE Incremental Update for a Text Index – TEXTIX1



Text Search Index Example

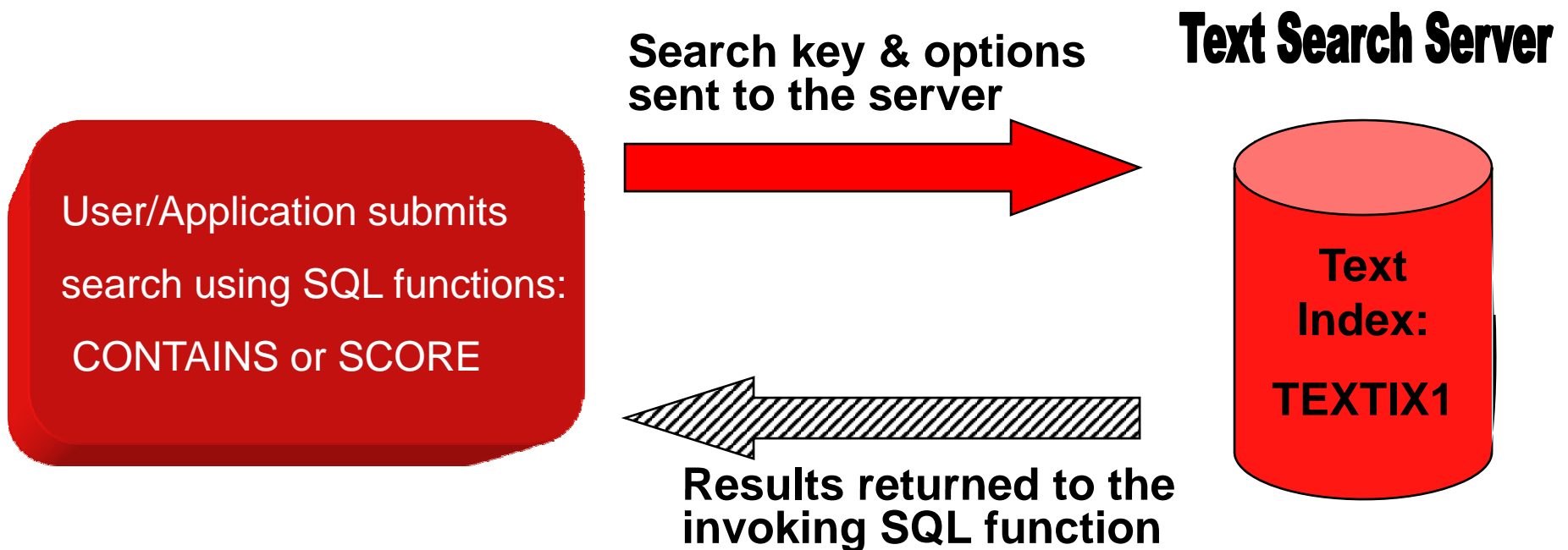
- CALL SYSPROC.SYSTS_CREATE(
 'myschema',
 'resumes_indx',
 'myschema.resumes(applicant_resume)',
 'FORMAT INSO
 UPDATE FREQUENCY D(*) H(0) M(0)')
- FORMAT values: TEXT,HTML,XML,INSO
- UPDATE FREQUENCY controls index maintenance

Text Search Index Example – External Documents

```
CALL SYSPROC.SYSTS_CREATE(  
    'myschema',  
    'externaldoc_index',  
    'myschema.docarchive( getIFSfile(filename_column) )',  
    'FORMAT INSO'  
)
```

```
CREATE FUNCTION getIFSfile(VARCHAR(2000))  
    RETURNS CLOB(2G) AS LOCATOR  
    LANGUAGE C++  
    EXTERNAL NAME 'LIB1/PGM1(getIIFSfile)'  
    PROGRAM TYPE SUB  
    DETERMINISTIC  
    PARAMETER STYLE SQL
```


Search Processing



*****NOTE: CONTAINS and SCORE functions only supported by the SQL Query Engine (SQE)**

CONTAINS function

- **CONTAINS(*column-name*, *search-argument*, *options*)**
 - Column-name: column over which the text search index is built
 - Search-argument: text being searched for
 - Options: optional parameter, can be used to modify the query language or activate synonym matching
 - Function Output (integer value):
 - 1 - match was found
 - 0 - no match found

CONTAINS example

- Find matches on exact string 'New product interest' in the COMMENT column, use a host variable to pass search string.

```
char search_arg[100];
```

```
...
```

```
EXEC SQL DECLARE C1 CURSOR FOR  
    SELECT custkey FROM customers WHERE  
    CONTAINS(comment, :search_arg) = 1  
    ORDER BY CUSTKEY;
```

```
EXEC SQL SET :search_arg = "New product interest";
```

```
EXEC SQL OPEN C1;
```

SCORE Function

- **SCORE(*column-name*, *search-argument*, *options*)**
 - Column-name: column over which the text search index is built
 - Search-argument: text being searched for
 - Options: optional parameter, can be used to modify the query language or activate synonym matching
 - Function Output:
 - Value between 0 and 1, up to 3 decimal points
 - Higher value indicates a better match on the specified search

SCORE Example

- Find those thesis reports that discuss programming from a performance or parallel perspective along with the normalized score rating

```
SELECT projID, projAuthor,  
       INTEGER(SCORE(thesis,  
                   'programming AND (parallel OR performance)*100'  
                   ) AS relevance  
FROM projects  
WHERE  
CONTAINS(thesis,  
         'programming AND (parallel OR performance')=1  
ORDER BY relevance DESC
```

Search argument options

- Simple Search
 - Enter one or more query terms, default operator is AND
 - Other logical operators: OR, NOT
- Use the minus sign (-) to exclude terms
 - To show all documents with terms “SQL performance”, but not “Oracle”, search argument is:
SQL performance – Oracle
- Surround exact phrases in double quotes
 - To find hits for the product name DB2 Web Query, search argument is: “DB2 Web Query”

Search argument options

- Wildcard character (*) helps find documents when the exact spelling is not known, or many variations are desired.
 - ‘Sh* Gree*’
Would return: Shonn Greene, Shawn Green, etc
 - ‘John * Kennedy’
 - Would return: John F Kennedy or John Fitzgerald Kennedy, but NOT John Kennedy

Advanced search options

- Score Customization with $\wedge n$
 - Weights specific terms more than others
 - DB2 AND “IBM i” $\wedge 5$
 - “IBM i” will be weighted more in the score results than ‘DB2’
- XML search
 - A subset of the XPATH query language is supported
 - Search for text contained in specified XML element/tag

Search Example

- Find all of the applicants who have RPG experience in their resume.
 - **“RPG” search term delimited, so only matches returned for upper-case**

```
SELECT first_name || ' ' || last_name  
FROM resumes  
WHERE  
    CONTAINS(applicant_resume, "RPG") = 1
```

Search Example

- Find resumes with a match on 'software engineer' in Spanish

```
SELECT first_name || ' ' || last_name, email_address,  
       SCORE(applicant_resume, 'ingeniero de software',  
             'QUERYLANGUAGE=es_ES')  
FROM myschema.resumes  
WHERE  
      CONTAINS( applicant_resume, 'ingeniero de software',  
              'QUERYLANGUAGE=es_ES') = 1  
ORDER BY 3 DESC
```

Additional Information

- OmniFind Documentation
Search for “OmniFind PDF” at: <http://publib.boulder.ibm.com/infocenter/systems>
- DB2 for i Websites
 - Home Page: ibm.com/systems/i/db2
 - DeveloperWorks Zone: ibm.com/developerworks/db2/products/db2i5OS
 - Porting Zone: ibm.com/servers/enable/site/db2/porting.html
- Newsgroups
 - USENET: comp.sys.ibm.as400.misc, comp.databases.ibm-db2
 - System i Network DB2 Forum - <http://systeminetwork.com/isnetforums/forumdisplay.php>
- Education Resources - Classroom & Online
 - ibm.com/systemi/db2/gettingstarted.html
 - ibm.com/partnerworld/wps/training/i5os/courses
- DB2 for i Publications
 - White Papers: ibm.com/partnerworld/wps/whitepaper/i5os
 - Online Manuals: ibm.com/systemi/db2/books.html
 - DB2 for i Redbooks: ibm.com/systemi/db2/relredbooks.html