Searching your SQL data

Harm Wibier

Data Access Worldwide

DataAccess WORLDWIDE

Why a search?

- Navigation options in are key to a good usability
- Google has raised the standards
- Machines have the capacity for full text searches

In DataFlex Applications

- Quickly navigate to a specific entity
  - Central search function in the application
- Easilly select parent records
  - Find records in a select
  - Find records in a prompt list

Chinook database

Artist
- ArtistId
- Name

Album
- AlbumId
- ArtistId
- Title

Track
- TrackId
- AlbumId
- Name
- Composer

DataAccess WORLDWIDE
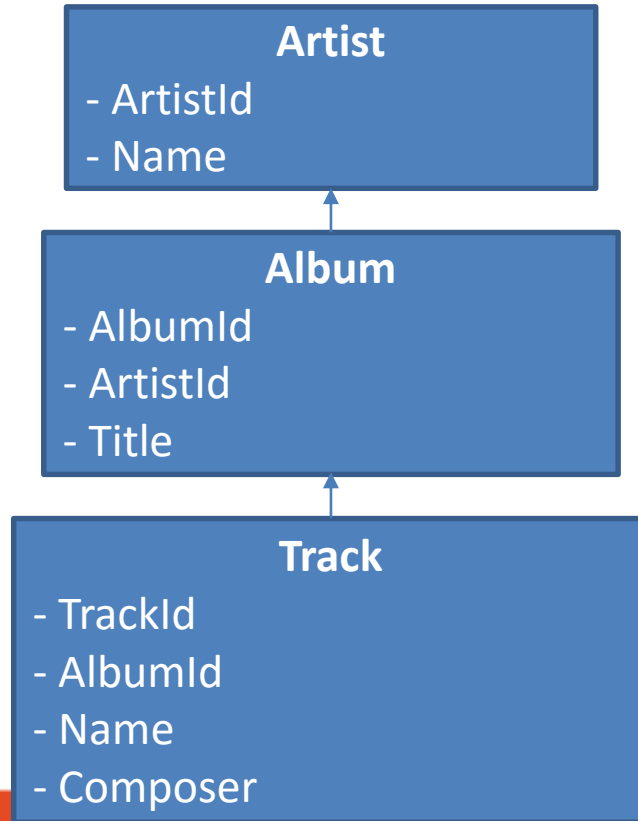
SQL Filters

- Filters in the WHERE clause of a SQL query

```
SELECT * FROM Track WHERE Track.UnitPrice > 1
```

Data Dictionary SQL Filters

- Set psSQLFilter to define
- Enable using pbUseDDSQLFilters

```
Object oTrack_DD is a Track_DataDictionary
    Set psSQLFilter to "Track.UnitPrice > 1"
    Set pbUseDDSQLFilters to True
End_Object
```

psSQLFilter vs Constrain

- Increased performance
  - ❑ Filtering offloaded to the server
  - ❑ No unnecessary records loaded to the client
- Use advanced SQL filtering options

DataAccess WORLDWIDE

SQL LIKE

- Filter based on textual expressions
- Use wildcards like % and _

SELECT * **FROM** Track **WHERE** Name **LIKE** '%Dirty%'

Search filter using LIKE

- ## Multiple keywords

  Name **LIKE** '%Bad%' **AND** Name **LIKE** '%Boy%'

- ## Multiple fields

  (Name **LIKE** '%Bad%' **OR** Composer **LIKE** '%Bad%'**) AND**
  (Name **LIKE** '%Boy%' **OR** Composer **LIKE** '%Boy%'**)**

DataAccess WORLDWIDE

Look at a single table search using LIKE filters set on the DD.

SAMPLE 1

Global helper functions

- StrJoinFromArray aStrings sDelimeter
  - ❏ Join an array of strings together
- StrSplitToArray sString sDelimeter
  - ❏ Splits a string into an array of strings

SQL helpers on DataDictionary

- SqlStrLike iField sSearch

  - Generates a safe like string for the specified field

- SqlStrFileFieldName iField

  - Gets the SQL field name based on a field number

- SqlEscapedStr

  - Escapes a string for usage inside a SQL Filter

OnConstrain in WebApps

- Triggered during synchronization process
- Make sure that it doesn't change directly on the client
    - Do not read values directly from UI
- Can cause synchronization issues
    - Especially pre 18.2

DataAccess WORLDWIDE

SQL Views

- Virtual tables based on resultset of a SQL Query

    CREATE VIEW TrackNames **AS**
    SELECT Name **FROM** Track

- Accessible as a table

    SELECT * FROM TrackNames

- Add to your filelist as a table

DataAccess WORLDWIDE

Multi table search

- Generate a SQL view concatenating the tables
    - Use SQL UNION statement
    - Put search text in a single field
- Access the view like a regular table in DataFlex

Look at using a view to search multiple tables.

SAMPLE 2

SQL Full Text Search

- Full text search indexing

- Separate service

- Available on all MS SQL versions
  - ❑ Not always installed by default

- Create catalog
  - ❑ Add fields

ChinookSearch
- Database Diagrams
- Tables
- Views
- Synonyms
- Programmability
- Service Broker
- Storage
  - Full Text Catalogs
    - MusicFTC
    - MusicViewFTC
  - Partition Schemes
  - Partition Functions
  - Full Text Stoplists
  - Search Property Lists
- Security
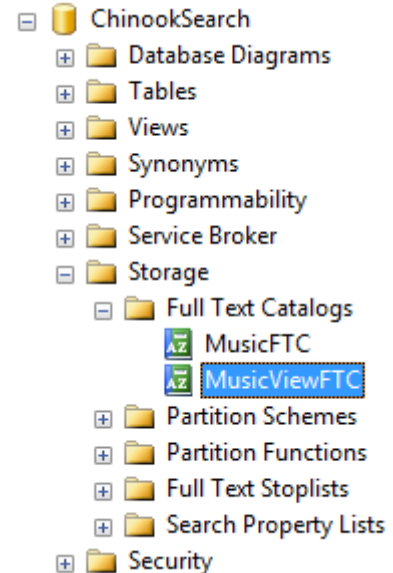
Why SQL full text search?

- Better performance with large amounts of data
- Language enhancements
- Relevance rating
- Highly configurable
  - Scheduled population

FREETEXT function

- Returns true of false
- Filter on keywords
- Columns must be part of full text catalog

SELECT * FROM Track **WHERE** FREETEXT **((**Name**,** Composer**),** 'Body Count'**)**

Look at using the FREETEXT filter.

SAMPLE 3

DataAccess WORLDWIDE

FREETEXTTABLE function

- Returns a table with KEY and RANK columns
  - Key refers to the primary key of the searched table
  - Rank indicates the relevance of the found row
- Use in FROM clause
- JOIN actual data on KEY

SELECT * **FROM FREETEXTTABLE (**Track**, (**Name**,** Composer**),** 'Body Count'**) AS** FTS **ORDER BY RANK DESC**;

Full Query

SELECT

        TrackId**,**

        Track**.**Name **AS** Name**,**

        Composer**,**

        Album**.**Title **AS** Album**,**

        Artist**.**Name **AS** Artist**,**

        FTS**.Rank AS** SearchRank

FROM Track

        INNER JOIN FREETEXTTABLE (Track**, (**Name**,** Composer**),** 'Body Count'**) AS** FTS

                ON Track**.**TrackId **=** FTS**.[KEY]**

        LEFT JOIN Album **ON** Track**.**AlbumId **=** Album**.**AlbumId

        LEFT JOIN Artist **ON** Album**.**ArtistId **=** Artist**.**ArtistId

ORDER BY RANK DESC;

Use embedded SQL

- Manually filled list
    - Implement OnManualLoadData
        - Execute search statement
        - Read result set into tWebRow[]
- Use GridRefresh to fill

Single table full text search sorted on rank.

SAMPLE 4

Stored procedure

- Creates a procedure on the SQL database
- Call using embedded SQL API
- Pass filter as parameter

```
CREATE PROCEDURE SearchMultiTable
@SearchString nvarchar(255)
AS
…
RETURN
```
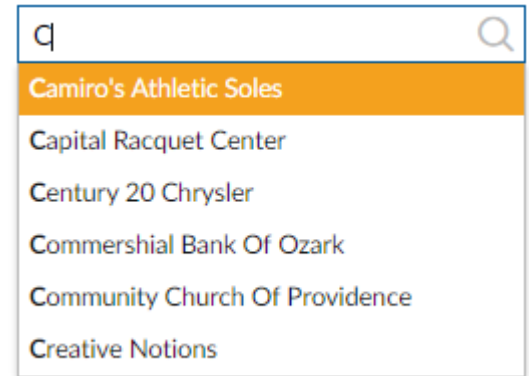
Multi table text search using UNION and embedded SQL sorted by rank.

SAMPLE 5

SuggestionForm

- Did you know the suggestionform?
  - Available for windows and web
  - Ideal for finding parents without lookup
  - Easy to configure using Entry_Item
  - Full text find using LIKE on MS SQL
  - Extendable

DataAccess WORLDWIDE

Extract keywords

- Use MS SQL system function
  - sys.dm_fts_index_keywords

```sql
SELECT * FROM sys.dm_fts_index_keywords(db_id('ChinookSearch'), object_id('dbo.Track'))
WHERE display_term LIKE ('ama%')
```

Search with keyword suggestions.

SAMPLE 6

Comparison

LIKE filters

- Easier to build
  - DD Filter
  - Data Aware Lists
  - Full control in DF
- Take advantage of views

Full Text Search

- Harder to build
  - Embedded SQL
  - Manually filled list
  - Configuration on the database
- Smarter results
- Better performance on large data sets

Questions?

THANK YOU FOR YOUR TIME!