# CA Plex support for Modernizing IBM DB2 for i

Re-engineer DDS to DDL (Model API Wizard)
SQL Index, Encoded Vector Index, SQL View
Primary Key, Foreign Key and Check Constraints
Identity Columns, Audit Timestamp
SQL Trigger
SQL Stored Procedures, External Stored Procedures
Stop program-centric and start data-centric programming

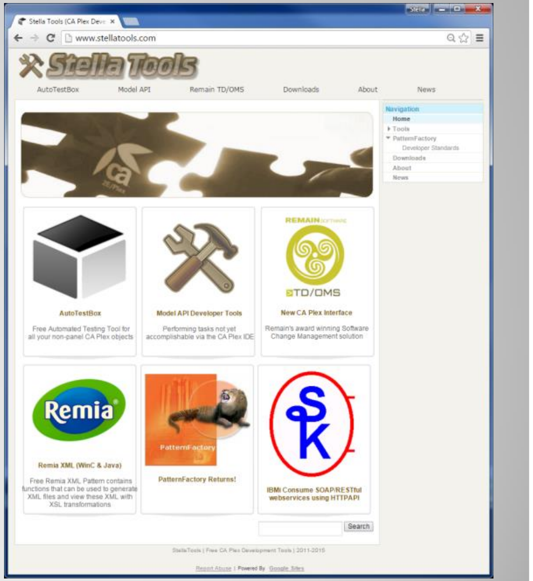
George Jeffcock
Nov 2015

- George Jeffcock, Application Specialist
- CA Plex since 1998
- Based in Helsingborg, Sweden
- Currently looking for the next CA Plex opportunity my Linked in profile

- StellaTools is a personal free time initiative to offer free advice / open source to the CA Plex development community

- Created a framework to allow developers to consume REST/SOAP webservices (XML and JSON format) on the the IBM i without the need to purchase 3rd party tools.
- Created 'AutoTestBox' - An automated testing tool to control the execution of tests while comparing actual outcomes with predicted outcomes.
- Liaised with Remain Software to create a new CA Plex interface to their Eclipsed based Software Change Management solution called TD/OMS
- 15 Model API client tools to perform developer tasks not yet accomplish-able via the CA Plex IDE

- Speaker at the CA 2E/Plex Worldwide Developer Conferences
  - Chicago 2011 - How others do it with CA Plex
  - New York 2013 - How others do Automated Testing



AutoTestBox

Free Automated Testing Tool for all your non-panel CA Plex objects

Model API Developer Tools

Performing tasks not yet accomplishable via the CA Plex IDE

New CA Plex Interface

Remain's award winning Software Change Management solution

Remia XML (WinC & Java)

Free Remia XML Pattern contains functions that can be used to generate XML files and view these XML with XSL transformations

PatternFactory Returns!

IBMi Consume SOAP/RESTful webservices using HTTPAPI

StellaTools | Free CA Plex Development Tools | 2011-2015

Report Abuse | Powered By Google Sites

www.StellaTools.com

- Read: Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between Redbooks, June 2014. (here by referenced as MIiAftDuttUIaEiB)

- Thats it, end of slides!

IBM

Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between

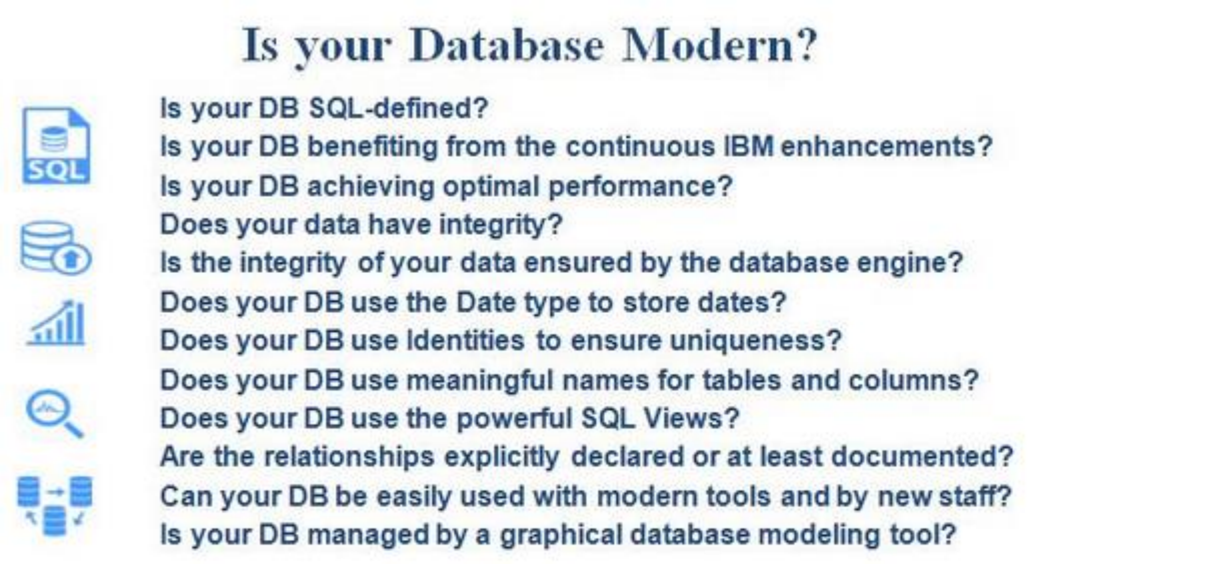
Redbooks

# Download pdf or Buy a copy today

- “For the IBM i community, the need to modernize applications is now an urgent requirement” Page1 MIiAftDuttUIaEiB
- "A Debate: DDS vs. DDL : Having studied the topic, we must conclude that there’s no simple answer. Indeed, there’s not one question here but two. There’s the DDS vs. DDL debate, and then there’s the SQL vs. RPG native I/O debate. You should review the facts and figures for each and make the decisions independently.." September 2008 by Jon Paris, Susan Gantner
- “Starting the modernization process by updating the database is another good approach.” Page221 MIiAftDuttUIaEiB
- An excellent article by IBM’s Dan Cruikshank titled “Modernizing Database Access—The Madness Behind the Methods.” One of his primary conclusions in 2006 is that all new files should be defined with Data Definition Language (DDL).
- SQL vs. DDS -- The good, the bad and the downright ugly An older article by IBM i Paul Tuohy Nov 2004

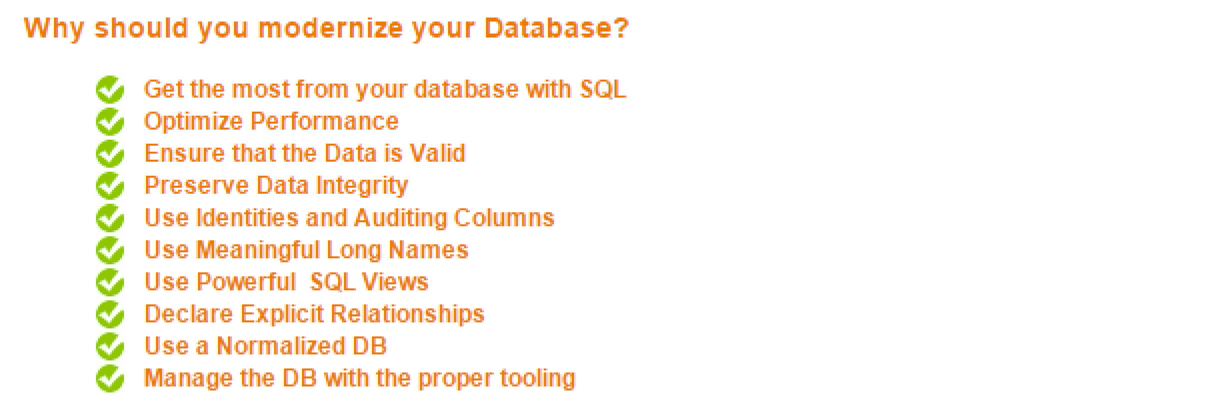
# Background

# Why Update Database?



## Is your Database Modern?

Is your DB SQL-defined?
Is your DB benefiting from the continuous IBM enhancements?
Is your DB achieving optimal performance?
Does your data have integrity?
Is the integrity of your data ensured by the database engine?
Does your DB use the Date type to store dates?
Does your DB use Identities to ensure uniqueness?
Does your DB use meaningful names for tables and columns?
Does your DB use the powerful SQL Views?
Are the relationships explicitly declared or at least documented?
Can your DB be easily used with modern tools and by new staff?
Is your DB managed by a graphical database modeling tool?



### Why should you modernize your Database?

- ✓ Get the most from your database with SQL
- ✓ Optimize Performance
- ✓ Ensure that the Data is Valid
- ✓ Preserve Data Integrity
- ✓ Use Identities and Auditing Columns
- ✓ Use Meaningful Long Names
- ✓ Use Powerful SQL Views
- ✓ Declare Explicit Relationships
- ✓ Use a Normalized DB
- ✓ Manage the DB with the proper tooling

- “The fact that you use Plex is an additional "complication" in your case, but also a HUGE benefit, as the fundamental approach behind 2E and Plex is "data centricity".” Aug 2015 Marinus Van Sandwyk, Founder & Chief Technology TEMBO Technology Lab (Pty) Ltd specializes in the development of database modernization solutions for IBM Power Systems running IBM i.
- the 2E model itself is nearly ideal www.databorough.com
- “Re-engineering the database is a vital component for the modernization process. But, you often cannot afford to throw away what you have today.” Page351 MIiAftDuttUIaEiB
- The “do no harm” approach (page 227 Chapter 8 MIiAftDuttUIaEiB)
  - Not as applicable as you have used CA Plex
  - Build your application database today for SQL Server (or Oracle)
- Data-centric development has been successfully achieved by CA Plex for decades for SQL Server and Oracle databases.
- Everything is done in the model
- No unsupported changes to runtime build messages
- CA Plex help docs applicable
- Plex developers already use this method so why would it not be good enough!
- What made a lot of this possible was addition in 2006 of RCDFMT clause to V5R4….its just been waiting for us to harness it…

# The good news is you modelled with CA Plex

# Generation Fix from CA

- Knowledge Base Article: ODBC TABLE FIELD CONSTRAINT DEFINED USING DBMS SCRIPT SCR
  - “Be sure to provide a terminating comma in your SRC object, if required, to separate this field's column definition from any following column or constraint definition.‟ Plex Help
  - ODBC compile error: SQL0104 - **Token ,** was not valid. Valid tokens: LIKE CHECK UNIQUE FOREIGN PRIMARY CONSTRAINT <IDENTIFIER>

- Case 00185182 - Today demo PTF 7.1.026.005 + Fix

- You can use Plex 6.1 but you manually edit the source as we previously have done!

# The Search for Performance

- **Performance should be seen as a Bonus**
- Listen to the IBM Champions Jon Paris and Scott Klement
- “If we are talking about new applications or _significantly_ reworking old ones then SQL is the way to go - period. If we are talking about simply replacing Read/Write/Chain with SQL equivalents then it is often a very bad idea and I have seen major performance issues resulting from this. Many people have tested performance in this area - and do so with each release. To-date all results that I have seen have SQL in second place on single I/O ops. By all means run your own tests. One of the reasons for this is that it is almost impossible to improve SQL performance without also improving "traditional" I/O. The Rochester database folk I worked with were constantly frustrated in their attempts to get SQL performance out ahead of "traditional" because everything they did tended to improve the "traditional" I/O as well. It was a moving target.” Jon Paris - June 2015
- "Sorry, but... why does it matter which one is faster?

  DDS was good enough for us 20 years ago. Nobody said "I'm dumping my AS/400 because DDS is too slow." Computers today are 1000 times as fast as they were 20 years ago, and now you're suddenly worried about whether DDS is fast enough?

  DDL is better because:

  it is more standard -- it's what all database systems use.
  it has more features. For example, try defining a CLOB in DDS.
  it's more flexible. It makes it much easier to take advantage of CCSIDs, nulls, and constraints.
  it leads to better integrity. Checking is done when a new row is inserted to prevent invalid data.
  And, yes, it performs a tiny bit better." Scott Klement 23 May 2012 11:38 PM form LF vs Index: How to test performance

- 2015 not 2005
  - Improvements to the SQE optimizer have virtually eliminated situations where Logical Files would cause DB2 revert to using the CQE optimizer. Jan 2013

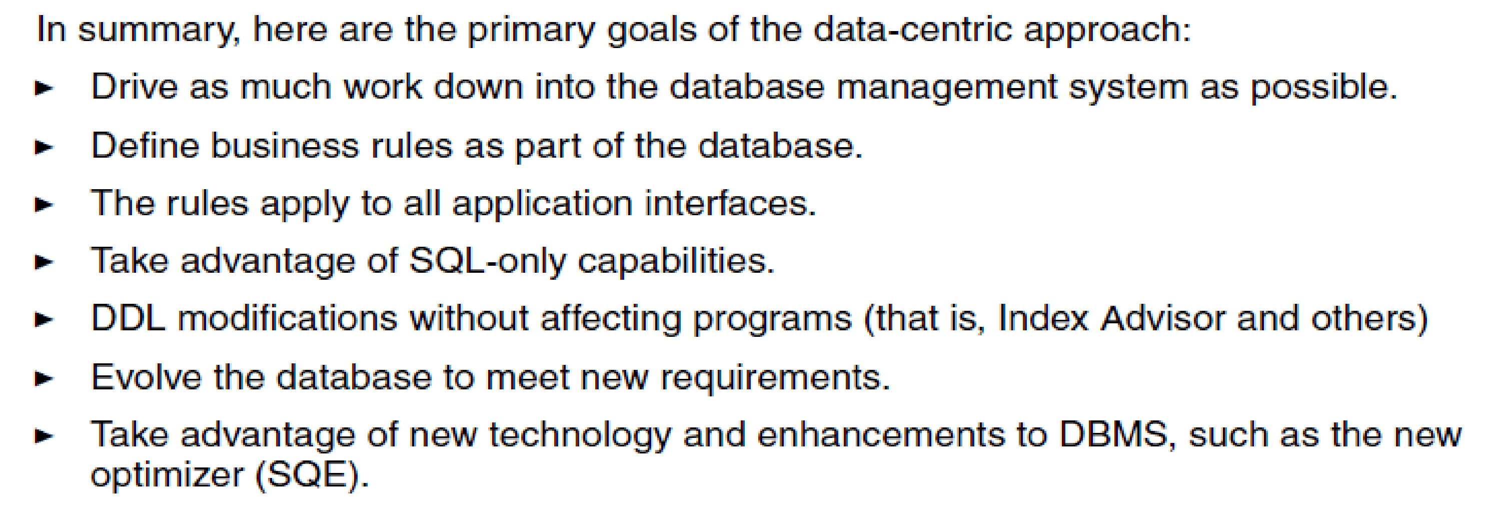
- How to determine which DDS to convert to DDL first Aug 2015 LinkedIn AS/400 Professionals

- "Pragmatically, I have found little benefit from a performance perspective to moving from traditional RLA from RPG" Sept 2015 Crispin Bates, Senior Application Development Analyst at IBM

# The Search for Performance

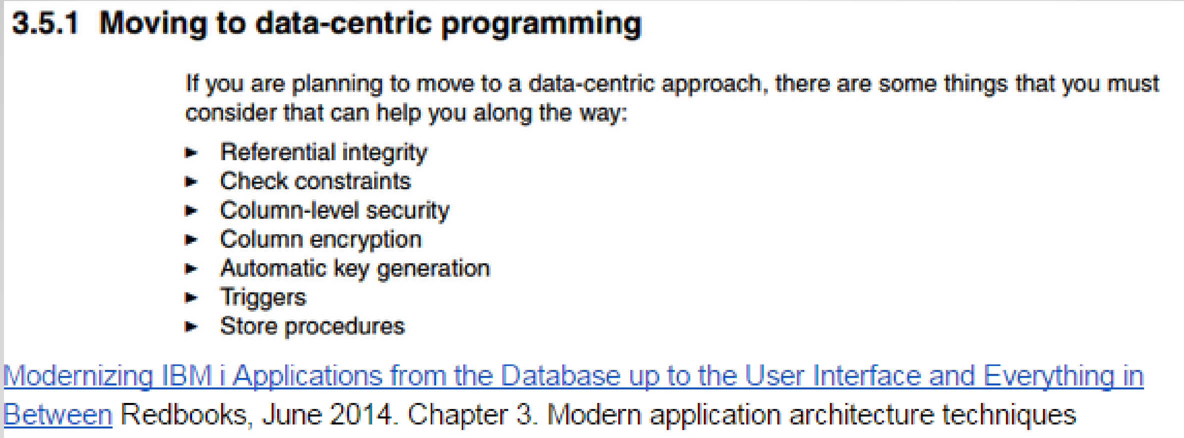
# Stop program-centric programming

- Data Centric Application Architecture  Jim Ritchhart March 2013



In summary, here are the primary goals of the data-centric approach:

- Drive as much work down into the database management system as possible.
- Define business rules as part of the database.
- The rules apply to all application interfaces.
- Take advantage of SQL-only capabilities.
- DDL modifications without affecting programs (that is, Index Advisor and others)
- Evolve the database to meet new requirements.
- Take advantage of new technology and enhancements to DBMS, such as the new optimizer (SQE).



### 3.5.1 Moving to data-centric programming

If you are planning to move to a data-centric approach, there are some things that you must consider that can help you along the way:

- Referential integrity
- Check constraints
- Column-level security
- Column encryption
- Automatic key generation
- Triggers
- Store procedures

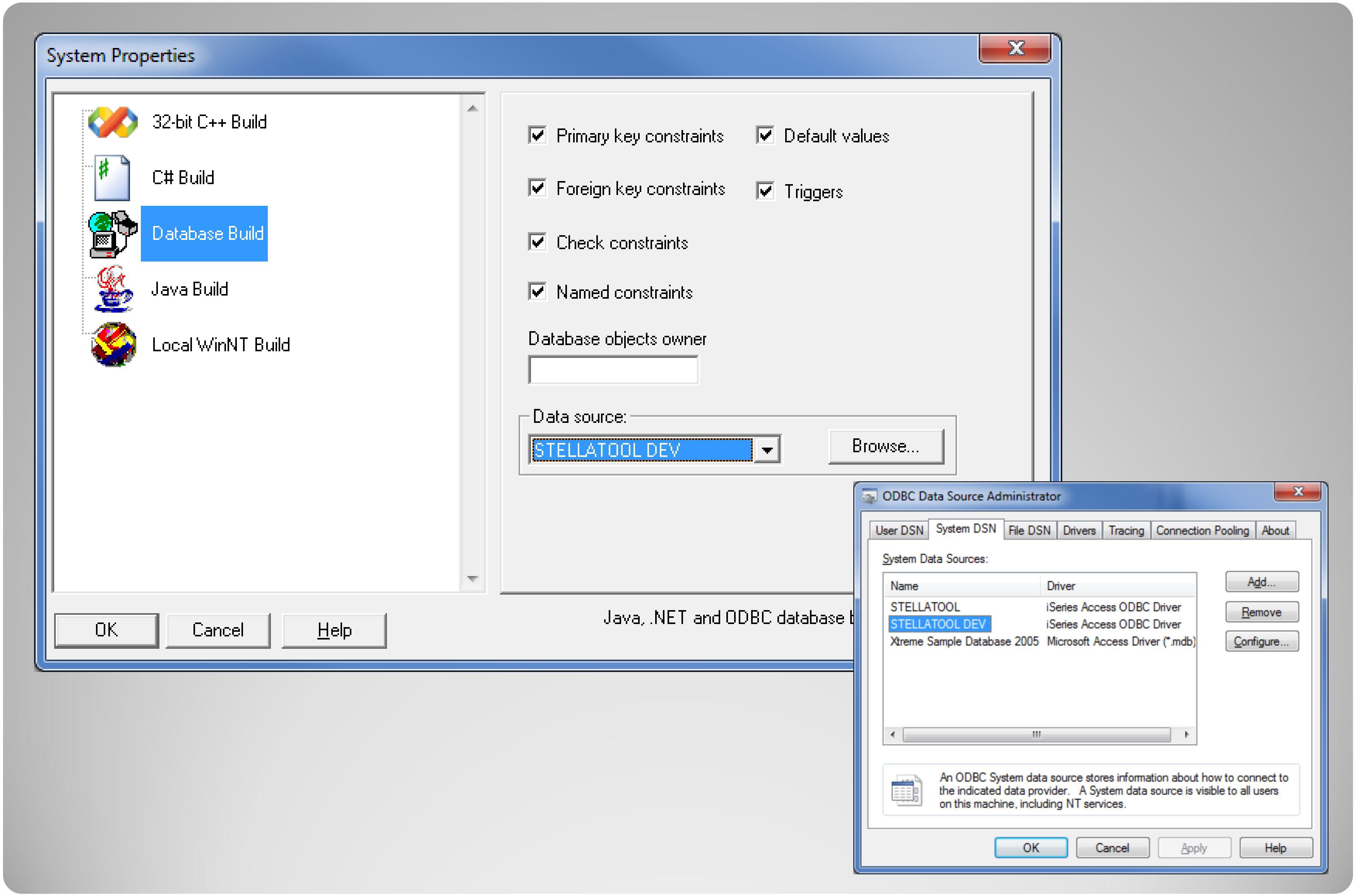
Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between Redbooks, June 2014. Chapter 3. Modern application architecture techniques

# CA Plex for Open Database

- “CA Plex for Open Database consists of the design environment and the Open Database generator and already supports Data Definition Language when generating schema to implement SQL database such as SQL Server, Oracle or ODBC database.” CA Plex Help
  - The trick here is we can use ODBC so that the Table/View Plex objects are generated in DDL and built via the data source specified in the Local System’s Database Build options

- There is a learning curve and CA Plex has a nice help page ‘Performance Considerations for an SQL Database’

- As preparation ,if you are not familiar with Data Modeling in CA Plex using Database Management System Scripts please read Plex help Chapter 5 Data Modeling . DBMS Scripts (SQL Only)  and FLD DBMS type NME verb



System Properties

- 32-bit C++ Build
- C# Build
- Database Build
- Java Build
- Local WinNT Build

☑ Primary key constraints ☑ Default values

☑ Foreign key constraints ☑ Triggers

☑ Check constraints

☑ Named constraints

Database objects owner:

Data source: STELLATOOL DEV

Java, .NET and ODBC database b

ODBC Data Source Administrator

System Data Sources:

| Name | Driver |
|---|---|
| STELLATOOL | iSeries Access ODBC Driver |
| STELLATOOL DEV | iSeries Access ODBC Driver |
| Xtreme Sample Database 2005 | Microsoft Access Driver (*.mdb) |

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

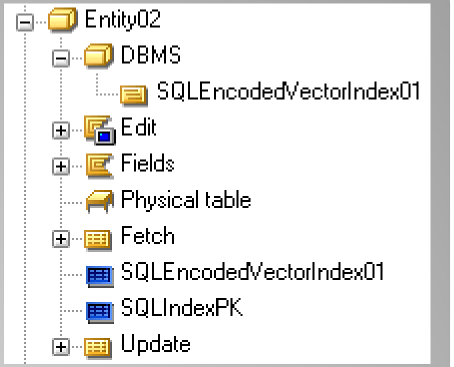
# G&B Settings

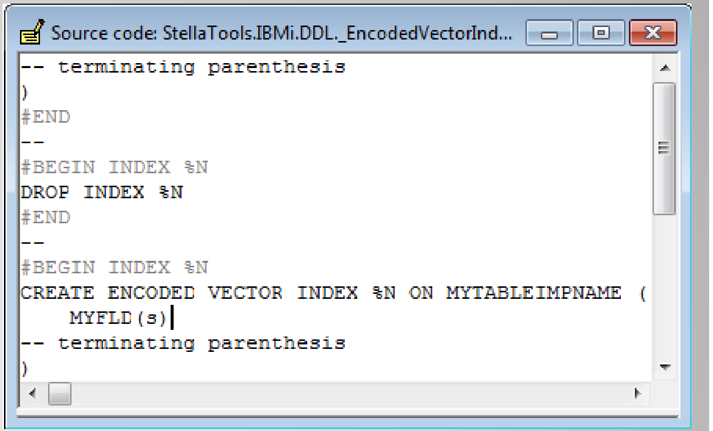
# 1. Reverse engineering DDS to DDL

- IBMi.Examples.Modernization.Entity01
  - Generate Data Definition Language (QSQGNDDL) API
    - Please Note new: QSYS2.GENERATE_SQL Stored Procedure
  - Run SQL Statements (RUNSQLSTM)
  - Major difference between an SQL table and a physical file created with DDS is the point at which data validation occurs. For a DDS physical file, the data is validated as data is read. For SQL, data is validated as it is written May 2005 Dan Cruikshank
  - DSPFD FILE(STELLATOOL/MYEXP1T)
    - SQL file type . . . . . . . . . . . . . . . . . : TABLE
    - Maximum members . . . . . . . . . . . . . . . : MAXMBRS 1
    - Reuse deleted records . . . . . . . . . . . . : REUSEDLT *YES
  - But not in the CA Plex model which arguably misses the point off DDL support in 2015
    - A straight conversion of DDS equals No Key Constraints, No Foreign Key Constraints, No Check Constraints (if like in the vast majority of cases constraints have not been added manually by Add Physical File Constraint (ADDPFCST) command )

# 2. Tuning with SQL Index and EVI

- IBMi.Examples.Modernization.Entity02
  - IBMi.DDL._IndexPK
    - Index name Triple required
    - Inherits – Icon to Distinguish
    - Inherits – index SYS Yes
    - Inherits – Unique SYS Yes
    - Inherits – type Sys Table_access
    - Inherits – language SYS ODBC
    - Does not inherit explicit Record Format
      - (DSPFD FILE(STELLATOOL/MYEXP2IX1)).
      - Therefore LF will share page size
  - IBMi.DDL._EncodedVectorIndex01
    - Index name Triple required
    - Inherits – Icon to Distinguish
    - Inherits – index SYS Yes
    - Inherits – Unique SYS No
    - Inherits – type Sys Table_access
    - Inherits – language SYS ODBC
    - Inherits – index DBMS script SRC
      - Drops SQL Index, Creates your EVI using your source



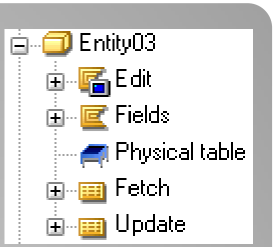


```
-- terminating parenthesis
)
#END
--
#BEGIN INDEX %N
DROP INDEX %N
#END
--
#BEGIN INDEX %N
CREATE ENCODED VECTOR INDEX %N ON MYTABLEIMPNAME (
    MYFLD(s)
-- terminating parenthesis
)
```

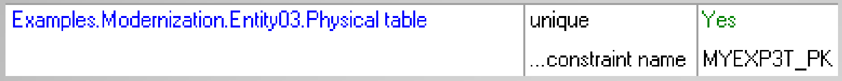
- Note: No DDS defined PF
- Note: Share increased logical page size LF
- Note: Use Index Advisor in System i Navigator

- Read: IBM DB2 for i indexing methods and strategies
- Read: DB2 Indexing: Tips, Tricks, & MIPS

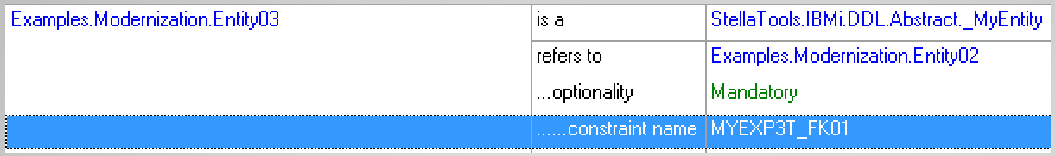
# 3. Convert DDS PF to SQL Table



- IBMi.Examples.Modernization.Entity03
  - PRIMARY KEY CONSTRAINT (MYEXP3T_PK )
  - FOREIGN KEY CONSTRAINT (MYEXP3T_FK01)
  - ”...constraint name NME” CA Plex Help

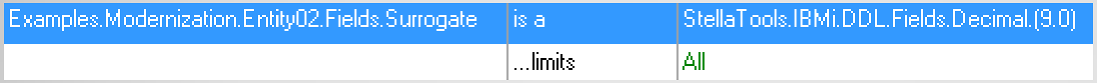


| | | |
|---|---|---|
| Examples.Modernization.Entity03.Physical table | unique | Yes |
| | ...constraint name | MYEXP3T_PK |



| | | |
|---|---|---|
| Examples.Modernization.Entity03 | is a | StellaTools.IBMi.DDL.Abstract._MyEntity |
| | refers to | Examples.Modernization.Entity02 |
| | ...optionality | Mandatory |
| | ......constraint name | MYEXP3T_FK01 |

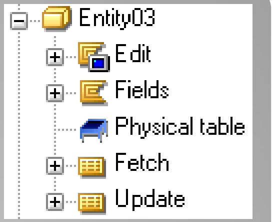
- Mix and Match DDS and DDL
  - SQL0538 - The FOREIGN key in constraint MYEXP3T_FK in STELLATOOL not same as the parent key.



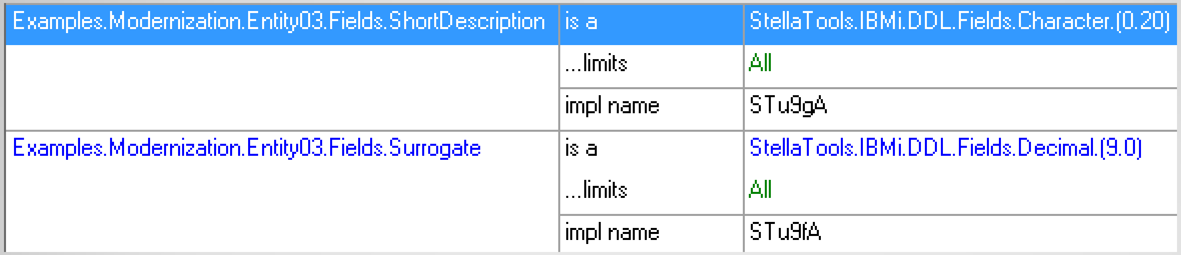
| | | |
|---|---|---|
| Examples.Modernization.Entity02.Fields.Surrogate | is a | StellaTools.IBMi.DDL.Fields.Decimal.(9.0) |
| | ...limits | All |

  - SQL0573 - Table MYEXP2T in STELLATOOL does not have a matching parent key.
    - ADDPFCST FILE(STELLATOOL/MYEXP2T) TYPE(*PRIKEY) KEY(STU91A) CST(MYEXP2T_PK)

# 3. Convert DDS PF to SQL Table



- IBMi.Examples.Modernization.Entity03
  - IBMi.DDL._Table
    - Inherits Icon to Distinguish
    - Inherits language SYS ODBC
  - IBMi.DDL.Fields
    - Inherit DBMS type triple
      - %char% %decimal%(19)
    - Inherit DBMS script SRC
      - NOT NULL, DEFAULT Values
      - fields defined in **DDS** are assumed **not** to be **NULL**-Capable
      - Decimal Lengths



| | | |
|---|---|---|
| Examples.Modernization.Entity03.Fields.ShortDescription | is a | StellaTools.IBMi.DDL.Fields.Character.(0.20) |
| | ...limits | All |
| | impl name | STu9gA |
| Examples.Modernization.Entity03.Fields.Surrogate | is a | StellaTools.IBMi.DDL.Fields.Decimal.(9.0) |
| | ...limits | All |
| | impl name | STu9fA |

- Now its in the CA Plex model

# Where is my Source?

- Don’t be tempted to allow your deployment process to dictate the way you code.
  - If your deployment process is not fit for purpose then change it, not the way you code i.e. program to data centric programming.

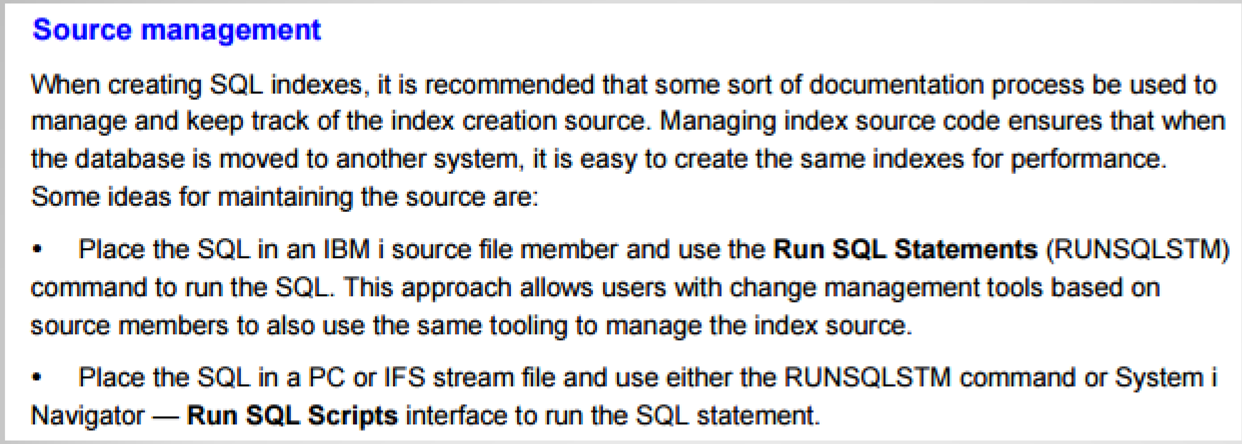
- IBMi.GenerateAndBuildObjs
  - Object Browser Model API Add-In Tool
  - PlexAPILib30.IPlexAPI.Build
  - Generate Data Definition Language (QSQGNDDL) API

- IBMi.SrcGenDDL.Implemented.GenerateDDL

GenerateAndBuildObjs

    - [StellaTools Options]
    - ;Overwrite Source Physical File name from Build file such as QDDSSRC to your own DDL file
    - DDLTbl_sourcefile=

- Index creation: Best practices - Source management (IBM DB2 for i indexing methods and strategies)



**Source management**

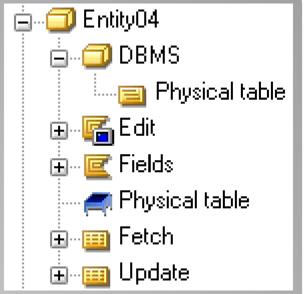
When creating SQL indexes, it is recommended that some sort of documentation process be used to manage and keep track of the index creation source. Managing index source code ensures that when the database is moved to another system, it is easy to create the same indexes for performance. Some ideas for maintaining the source are:

- Place the SQL in an IBM i source file member and use the **Run SQL Statements** (RUNSQLSTM) command to run the SQL. This approach allows users with change management tools based on source members to also use the same tooling to manage the index source.
- Place the SQL in a PC or IFS stream file and use either the RUNSQLSTM command or System i Navigator — **Run SQL Scripts** interface to run the SQL statement.

- “The reality is the database defines itself” Bob Cancilla May 7, 2015

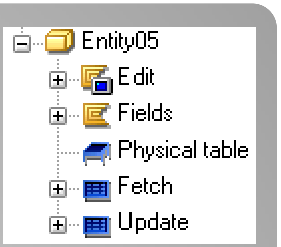
# 4. Column Labels

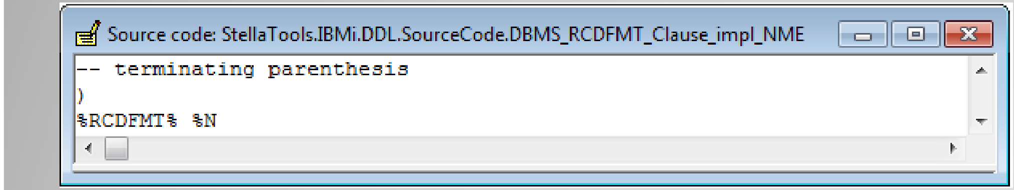
- IBMi.Examples.Modernization.Entity04



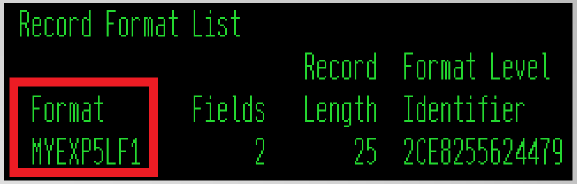
- IBMi.DDL._Table_DBMS_Script_Labels
  - Labels On Table
  - Labels On Column (Text & Heading)
- IBMi.DDL._Table_DBMS_Script_Labels&Alias
  - Wow! I Could Have Had Long Column Names! by Ted Holt November 16, 2011
  - A method to support 'Long Column Names' is to Drop then Create TABLE in DBMS script copying the generated source while adding the FOR COLUMN clauses
  - IBM to support RENAME COLUMN for IBM i DB2 and not just the Windows/Linux DB2 version would help a lot.
- Keep in perpesctive the relative infrequency of File Changes.

- IBMi.Examples.Modernization.Entity05
  - IBMi.DDL.PATTERNS._RelationalTable_NoDDS
    - Inherits – Icon to Distinguish
    - Inherits – index SYS Yes
    - Inherits – unique SYS Yes
    - Inherits – type Sys Table_access
    - Inherits – language SYS ODBC
    - Inherits – contains ALL
    - Inherits – index DBMS script SRC  DBMS_RCDFMT_Clause_impl_NME
      - Imp Name Record Format



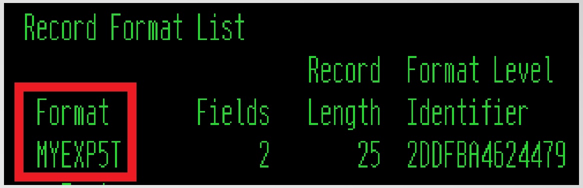

Source code: StellaTools.IBMi.DDL.SourceCode.DBMS_RCDFMT_Clause_impl_NME

```
-- terminating parenthesis
)
%RCDFMT% %N
```

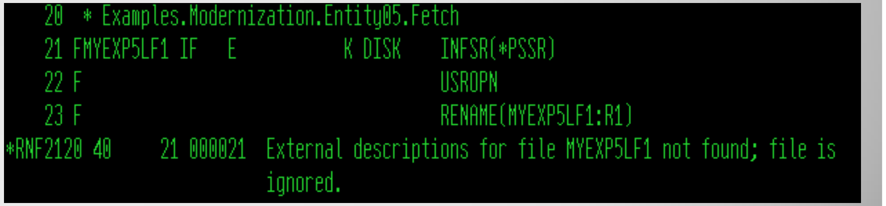


```
Record Format List
                        Record  Format Level
 Format        Fields   Length  Identifier
 MYEXP5LF1          2       25  2CE8255624479
```

- Why RCDFMT Clause?
  - By default Index will use Table Record Format
  - View Variable generates F-Spec based on View record format



```
Record Format List
                        Record  Format Level
 Format        Fields   Length  Identifier
 MYEXP5T            2       25  2DDFBA4624479
```

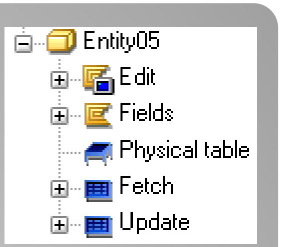


```
   20  * Examples.Modernization.Entity05.Fetch
   21 FMYEXP5LF1 IF   E           K DISK    INFSR(*PSSR)
   22 F                                     USROPN
   23 F                                     RENAME(MYEXP5LF1:R1)
*RNF2120 40     21 000021  External descriptions for file MYEXP5LF1 not found; file is
                           ignored.
```

# 5. Entity with no DDS

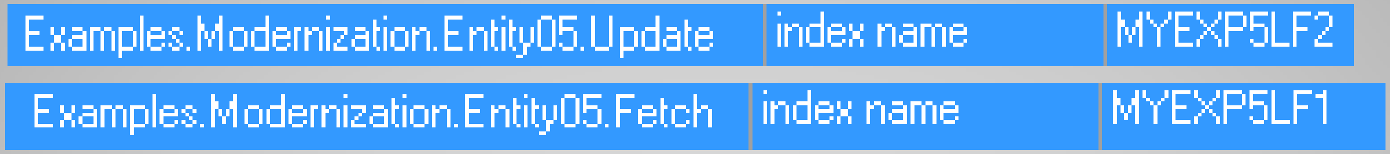
# 5. Entity with no DDS

- IBMi.Examples.Modernization.Entity05



Entity05
Edit
Fields
Physical table
Fetch
Update

  - Index Name Triples MYEXP5LF1,MYEXP5LF2



| | | |
|---|---|---|
| Examples.Modernization.Entity05.Update | index name | MYEXP5LF2 |
| Examples.Modernization.Entity05.Fetch | index name | MYEXP5LF1 |

  - BTW an SQL index cannot be specified in an SQL Statement, but it can be used like any keyed logical file with native I/O, i.e. it can be specified within the F-Specs. Birgitta Hauser

  - Conversely: “Do not open SQL objects (Tables / Indexes / Views) in F specs in your program” Read p256 MIiAftDuttUIaEiB

- Further Examples
  - StellaTools.AutoTestBox.Data (Level V2.0 DDL)
  - \StellaToolsV2_0\DDS2DDL\LIBRARYBOOKS

- Make sure Parent or Referred To Files (DDS or DDL) or in your target compile IBMi library or added through a Library List entry on the ODBC data source. If not you do not get a failure message to the Plex Message Log (you can see the error under ODBC Trace thou)

- Surrogate names.
  - ODBC generator uses an object name if no Imp Name is found which is good but DDS generator does not due to length limitations
    - DROP Table and add Surrogate Names See DBMS Script: StellaTools.AutoTestBox.Data.SurrogateSystem.DBMS.Physical table (Level V2.0 DDL)
    - Add Imp name to library object
    - Replace Field (groan..)

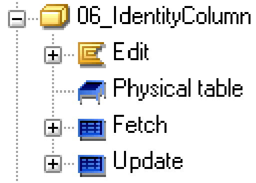
- If you already use the ODBC generator and worried the additional DBMS scripts will harm your existing implementation think about creating a new level
  - AutoTestBox is implemented this way as DB already supports SQL Server so a new level was created for the DDL IBMi support (Level V2.0)

- Journaling - SQL7905 - Table MYEXP3T in STELLATOOL created but was not journaled. ODBC Warning: [IBM][System i Access ODBC-drivrutin][DB2 for i5/OS]
  - Create Journal Receiver (CRTJRNRCV) (http://itknowledgeexchange.techtarget.com/itanswers/journal-in-as400/))
  - Create Journal (CRTJRN)
  - Start Journal Physical File (STRJRNPF)
  - Automatic journaling using QDFTJRN

# Modernization Tips

- IBMi.Examples.NewDevelopment.01_IdentityColumn
  - IBMi.DDL.Fields._IdentityColumn

```
GENERATED ALWAYS AS IDENTITY (
START WITH 1, INCREMENT BY 1
NO MINVALUE NO MAXVALUE
NO CYCLE NO ORDER
CACHE 20 )
```



```
⊟ 06_IdentityColumn
   ⊞ Edit
     Physical table
   ⊞ Fetch
   ⊞ Update
```

IBMi.DDL.SourceCode.DBMS_IdentityColumn_PostCommaFix

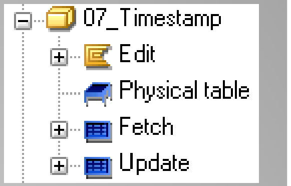
IBMi.DDL.SourceCode.DBMS_IdentityColumn_PreCommaFix

- Be aware of Identity Column becoming an attribute of an entity that is the primary key of another entity

# 1. Auto-generated column support

# 2. Audit TimeStamp

- IBMi.Examples.NewDevelopment.02_Timestamp



- 07_Timestamp
  - Edit
  - Physical table
  - Fetch
  - Update

  - IBMi.DDL.Fields.TimestampInsert

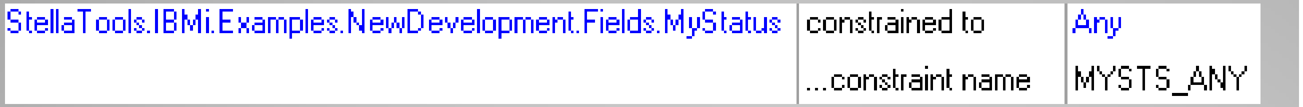
```
NOT NULL
DEFAULT CURRENT_TIMESTAMP,
```

  - IBMi.DDL.Fields.TimestampLastUpdate

```
NOT NULL
FOR EACH ROW ON UPDATE
AS ROW CHANGE TIMESTAMP,
```

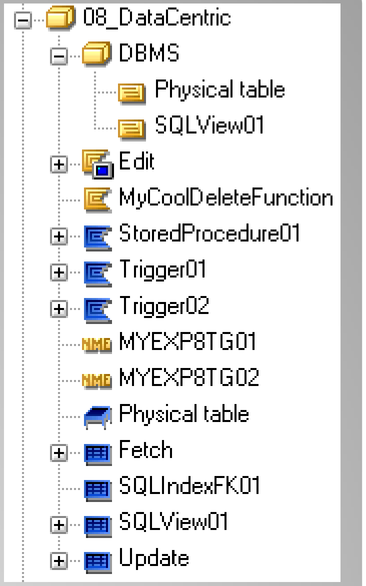
# 3. Data-Centric

- CONSTRAINT CHECK

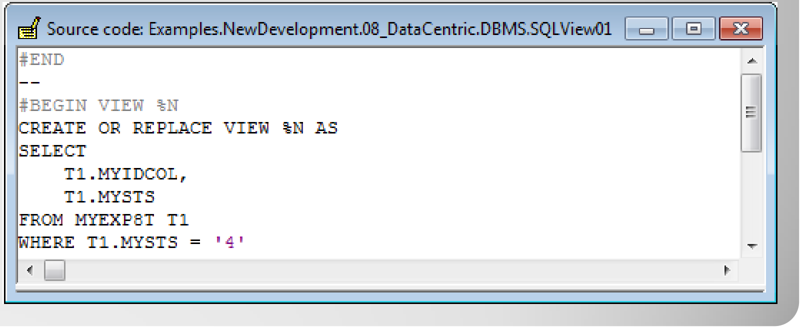


| | | |
|---|---|---|
| StellaTools.IBMi.Examples.NewDevelopment.Fields.MyStatus | constrained to | Any |
| | ...constraint name | MYSTS_ANY |

- PRIMARY KEY CONSTRAINT (MYEXP8T_PK )
- FOREIGN KEY CONSTRAINT (MYEXP8T_FK01)

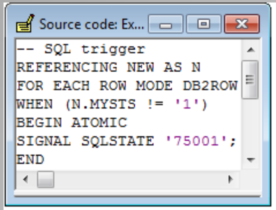


- IBMi.DDL._View01
    - IBMi.Examples.Modernization.Entity02.DBMS. SQLView01
    - Inherits – index SYS No

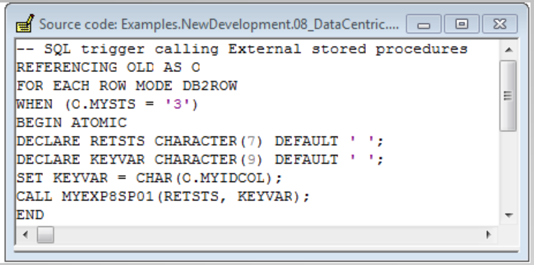

Source code: Examples.NewDevelopment.08_DataCentric.DBMS.SQLView01

```
#END
--
#BEGIN VIEW %N
CREATE OR REPLACE VIEW %N AS
SELECT
    T1.MYIDCOL,
    T1.MYSTS
FROM MYEXP8T T1
WHERE T1.MYSTS = '4'
```

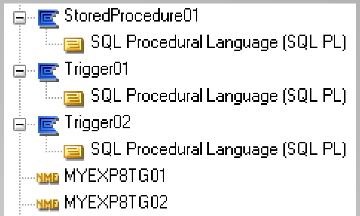
- SQL trigger

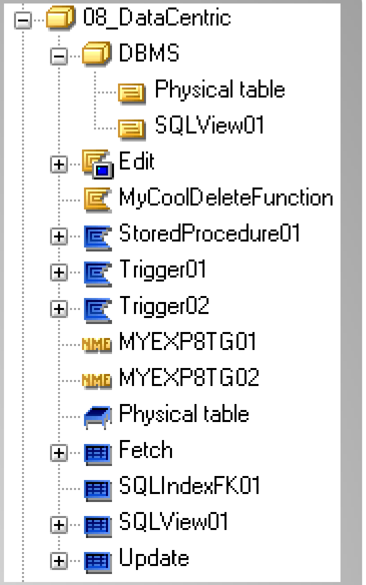


```
-- SQL trigger
REFERENCING NEW AS N
FOR EACH ROW MODE DB2ROW
WHEN (N.MYSTS != '1')
BEGIN ATOMIC
SIGNAL SQLSTATE '75001';
END
```



```
-- SQL trigger calling External stored procedures
REFERENCING OLD AS O
FOR EACH ROW MODE DB2ROW
WHEN (O.MYSTS = '3')
BEGIN ATOMIC
DECLARE RETSTS CHARACTER(7) DEFAULT ' ';
DECLARE KEYVAR CHARACTER(9) DEFAULT ' ';
SET KEYVAR = CHAR(O.MYIDCOL);
CALL MYEXP8SP01(RETSTS, KEYVAR);
END
```


StoredProcedure01
SQL Procedural Language (SQL PL)
Trigger01
SQL Procedural Language (SQL PL)
Trigger02
SQL Procedural Language (SQL PL)
MYEXP8TG01
MYEXP8TG02

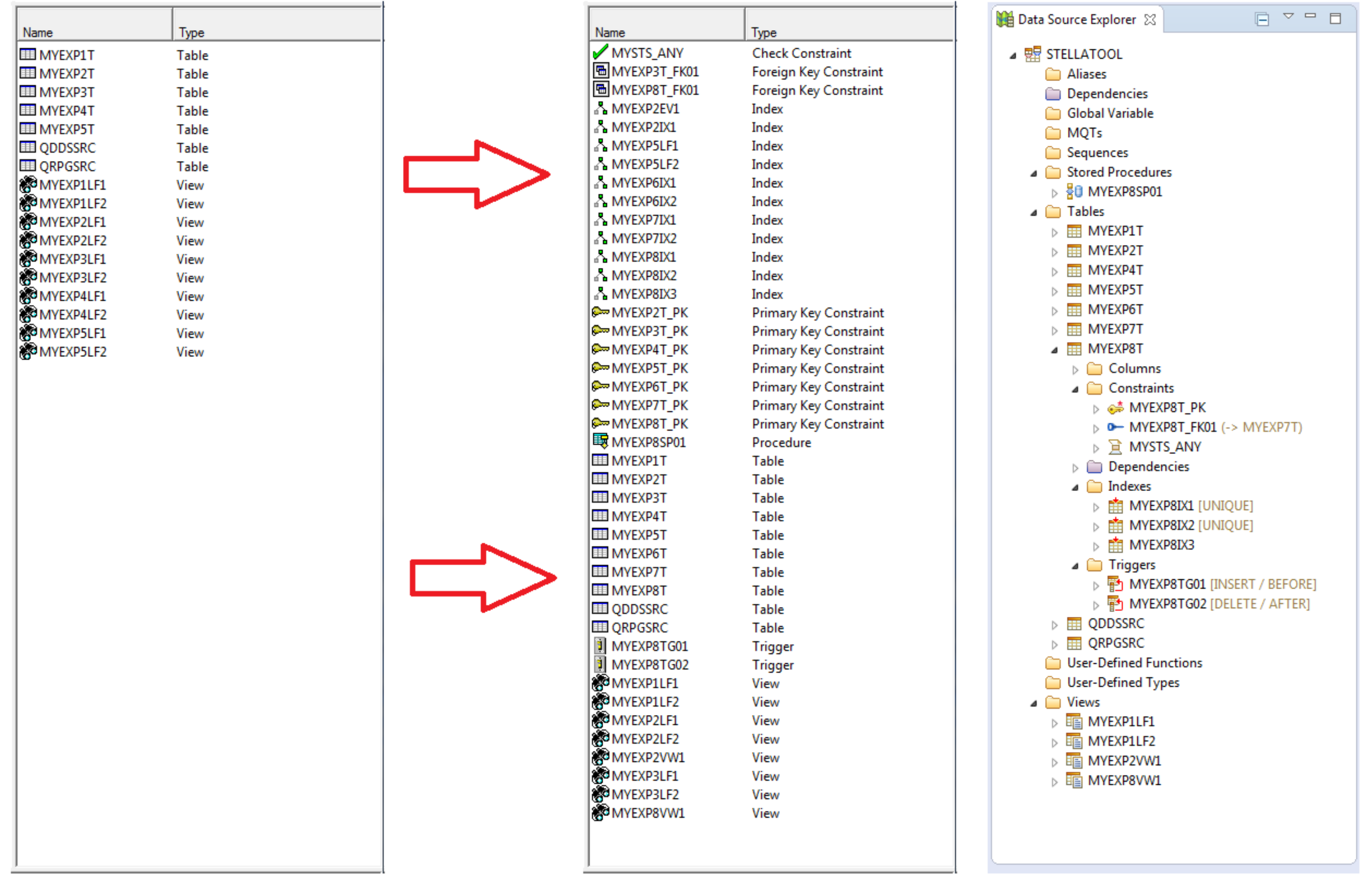


- SQL stored procedures, External stored procedures
  - Sql Result Set from Plex generated RPG?

- Read: Stored Procedures, Triggers, and User-Defined Functions on DB2 Universal Database for IBM I

# 3. Data-Centric

- "The REAL challenge is how to manage all the DB2 components COHESIVELY, an integrated management environment. In a MODERN application you end up with LOTS of Lego blocks‟ Aug 2015 Marinus Van Sandwyk



| Name | Type |
|---|---|
| MYEXP1T | Table |
| MYEXP2T | Table |
| MYEXP3T | Table |
| MYEXP4T | Table |
| MYEXP5T | Table |
| QDDSSRC | Table |
| QRPGSRC | Table |
| MYEXP1LF1 | View |
| MYEXP1LF2 | View |
| MYEXP2LF1 | View |
| MYEXP2LF2 | View |
| MYEXP3LF1 | View |
| MYEXP3LF2 | View |
| MYEXP4LF1 | View |
| MYEXP4LF2 | View |
| MYEXP5LF1 | View |
| MYEXP5LF2 | View |

| Name | Type |
|---|---|
| MYSTS_ANY | Check Constraint |
| MYEXP3T_FK01 | Foreign Key Constraint |
| MYEXP8T_FK01 | Foreign Key Constraint |
| MYEXP2EV1 | Index |
| MYEXP2IX1 | Index |
| MYEXP5LF1 | Index |
| MYEXP5LF2 | Index |
| MYEXP6IX1 | Index |
| MYEXP6IX2 | Index |
| MYEXP7IX1 | Index |
| MYEXP7IX2 | Index |
| MYEXP8IX1 | Index |
| MYEXP8IX2 | Index |
| MYEXP8IX3 | Index |
| MYEXP2T_PK | Primary Key Constraint |
| MYEXP3T_PK | Primary Key Constraint |
| MYEXP4T_PK | Primary Key Constraint |
| MYEXP5T_PK | Primary Key Constraint |
| MYEXP6T_PK | Primary Key Constraint |
| MYEXP7T_PK | Primary Key Constraint |
| MYEXP8T_PK | Primary Key Constraint |
| MYEXP8SP01 | Procedure |
| MYEXP1T | Table |
| MYEXP2T | Table |
| MYEXP3T | Table |
| MYEXP4T | Table |
| MYEXP5T | Table |
| MYEXP6T | Table |
| MYEXP7T | Table |
| MYEXP8T | Table |
| QDDSSRC | Table |
| QRPGSRC | Table |
| MYEXP8TG01 | Trigger |
| MYEXP8TG02 | Trigger |
| MYEXP1LF1 | View |
| MYEXP1LF2 | View |
| MYEXP2LF1 | View |
| MYEXP2LF2 | View |
| MYEXP2VW1 | View |
| MYEXP3LF1 | View |
| MYEXP3LF2 | View |
| MYEXP8VW1 | View |

Data Source Explorer

```
◢ STELLATOOL
    Aliases
    Dependencies
    Global Variable
    MQTs
    Sequences
  ◢ Stored Procedures
    ▷ MYEXP8SP01
  ◢ Tables
    ▷ MYEXP1T
    ▷ MYEXP2T
    ▷ MYEXP4T
    ▷ MYEXP5T
    ▷ MYEXP6T
    ▷ MYEXP7T
    ◢ MYEXP8T
      ▷ Columns
      ◢ Constraints
        ▷ MYEXP8T_PK
        ▷ MYEXP8T_FK01 (-> MYEXP7T)
        ▷ MYSTS_ANY
      ▷ Dependencies
      ◢ Indexes
        ▷ MYEXP8IX1 [UNIQUE]
        ▷ MYEXP8IX2 [UNIQUE]
        ▷ MYEXP8IX3
      ◢ Triggers
        ▷ MYEXP8TG01 [INSERT / BEFORE]
        ▷ MYEXP8TG02 [DELETE / AFTER]
    ▷ QDDSSRC
    ▷ QRPGSRC
    User-Defined Functions
    User-Defined Types
  ◢ Views
    ▷ MYEXP1LF1
    ▷ MYEXP1LF2
    ▷ MYEXP2VW1
    ▷ MYEXP8VW1
```

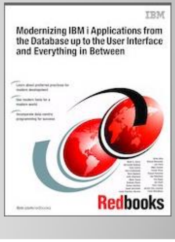
# 3. Data-Centric

# Problem Shooting

- Tracing ODBC Calls – Plex Help
  - "You can set up ODBC to trace all ODBC API calls made by an application. The contents of this file can help identify the last ODBC calls that were made before an error occurred."
  - "You can also use the contents of this file to determine problems during compilation of tables and views from within CA Plex."
- How To Generate an ODBC Trace with ODBC Data Source Administrator

# Conclusion – Enjoy the Journey

- Add PF Constraint (ADDPFCST)
  - Cautionary tale. Why was it not used?
- Download BlockFetchSQL (included in StellaTools)
- Look at DB2 SQL Procedural Language
  - "Separation of the business logic from the high-level application programs by using stored procedures that are written in SQL or host-based languages (that is, RPG) containing SQL statements (also known as external stored procedures)" Read p356 MIiAftDuttUIaEiB
- Create auto-generated keyed column Tables (Identity Column)
  - Then support Natural Keys by Unique Indexes
  - But read Natural vs Surrogate Keys
- Read again!: Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between Redbooks, June 2014.
- Above all Start the journey



IBM

Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between

Redbooks