Power Systems

IBM

DB2 for IBM i

What's new with IBM i 7.1 TR9 & IBM i 7.2 TR1

Scott Forstie
DB2 for i – Business Architect
forstie@us.ibm.com

IBM

DB2 for i

- Standards compliant
- Secure
- Scalable
- Functionally Advanced
- Excellent Performance
- Easier to use
- Easier to maintain

Value Proposition

Continual Investment and Innovation

V5R1

SQL triggers
Java Functions
DRDA DUW TCP/IP
2 GB LOBs
1 Terabyte Table
Journal Minimal Data
Two-phase over TCP/IP
DDL Journaling
Database Navigator
Generate SQL

V5R2

SQE Stage 1
IASPs
Identity columns
Savepoints
UNION in views
Scalar subselect
UDTFs
DECLARE GLOBAL TEMPORARY TABLE
Catalog views
JDBC V3.0
DRDA Kerberos
Journal Standby

V5R3

Partitioned tables
UFT-8 and UTF-16
ICU sort sequence
MQTs
Sequences
Implicit char/numeric
BINARY/VARBINARY
GET DIAGNOSTICS
DRDA Alias
DECIMAL(63)
SQE Stage 3
Ragged SWA
QDBRPLAY
Online Reorganize

V5R4

WebQuery
SSD Memory Preference
On Demand Performance Center
Health Center
Completion of SQL Core
Scalar fullselect
Recursive CTE
INSTEAD OF triggers
Descriptor area
XA over DRDA
DDM 2-phase
Scrollable cursor
2M SQL statement
1000 tables in a query
SQE Stage 5
Implicit journaling enhancements

6.1

Omnifind
MySQL storage engine
DECFLOAT
Grouping sets /super groups
INSERT in FROM
VALUES in FROM
Extended Indicator Variables
Expression in Indexes
ROW CHANGE TIMESTAMP
Statistics catalog views
CLIENT special registers
SQE Stage 6
DDM and DRDA IPv6
Deferred Restore of MQT and Logicals
Environmental limits

7.1

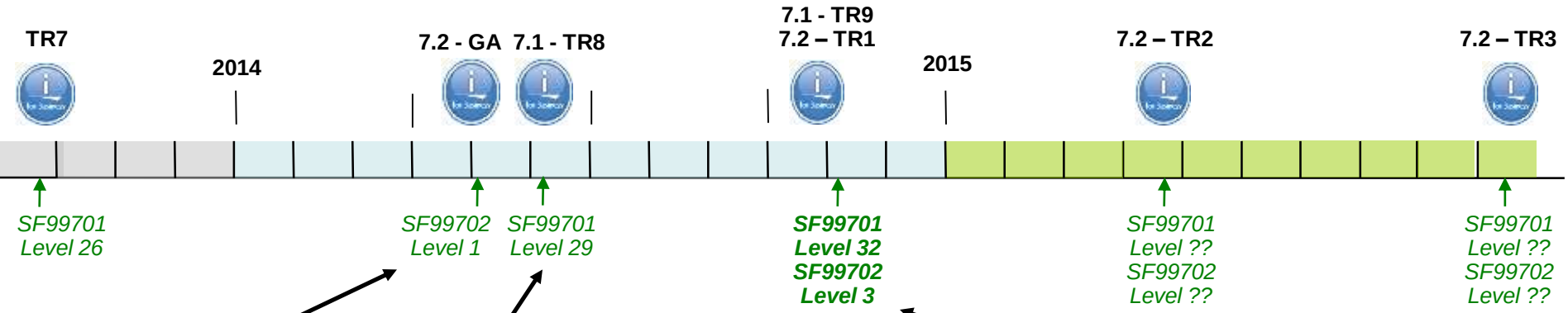
XML Support
Encryption enhancements (FIELDPROCs)
Result set support in embedded SQL
CURRENTLY COMMITTED
MERGE
MQ Functions
Global variables
Array support in procedures
Partition table enhancements
Three-part names and aliases
SQE Logical file support
SQE Adaptive Query Processing
EVI enhancements
Inline functions
CREATE OR REPLACE

7.2

Row and Column Access Control
XMLTABLE
CONNECT BY
TRANSFER OWNERSHIP
Named arguments and defaults for parameters
Obfuscation of SQL routines & triggers
Array support in UDFs
Timestamp precision
Multiple-action Triggers
Built-in Global Variables
Record movement between partitions on UPDATE
1.7 Terabyte Indexes
Health Center – Non-database limits
Navigator Graphing and Charting
SQE I/O Costing model improvement
TRUNCATE

DB2 for i – Enhancements delivered via DB2 PTF Groups

IBM i 7.1 & 7.2



7.2 - GA

- First DB2 PTF Group for 7.2
- All IBM i 7.1 Enhancements thru TR8

7.1 - TR8 timed Enhancements:

- Faster JDBC Toolbox
- Use SQL to programmatically analyze performance
- Generate SQL procedure
- Program & Package statement level statistical catalogs
- And more…

7.1 – TR9 timed Enhancements:

- Pipelined Functions
- Regular Expression Scalar Functions
- New IBM i Services
- Guardium and Performance Monitoring improvements
- Database Navigator Enhancements
- And more…

7.2 – TR1 timed Enhancements:

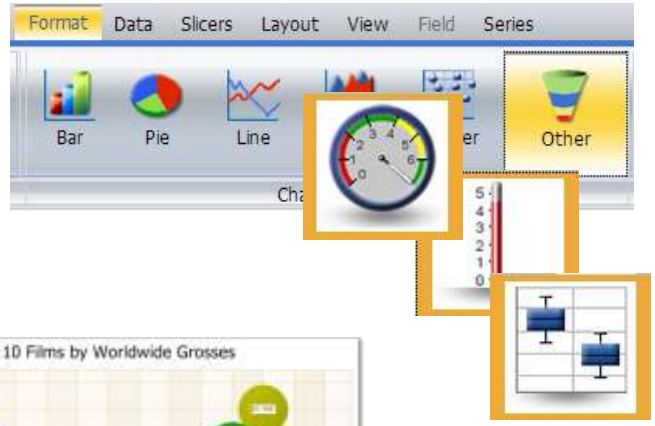
- Everything in TR9. and…
- System Limits Phase 3 - IFS
- CREATE **OR REPLACE** support on MASK and PERMISSION
- DB2 Built-in Global Variables for job name
- And more…

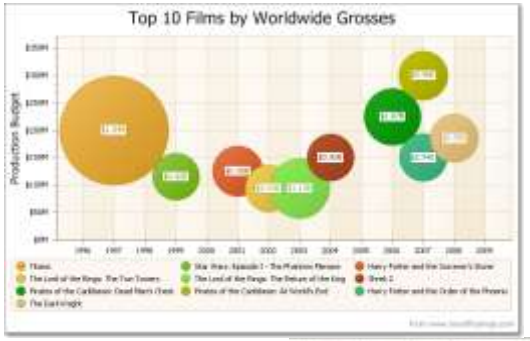
Enhancements delivered by PTF are documented here:

www.ibm.com/developerworks/ibmi/techupdates/db2

New Charts and Maps with Latest Group PTF

- DB2 Web Query contains over 100 different types of visualizations
  - Bar, Pie, Line, 3-D, Thermometers, gas gauges, stock reports, funnels, and more
  - New Visualizations include
    - Heat Maps
    - Bubble Charts
    - Tag Clouds
    - Streamgraph
  - Geo Maps
    - Interactive mapping out-of-the-box
    - 9 Layers of Zoom
    - Translated Countries
    - Adjust Heatscale and Opacity
    - Create customized maps

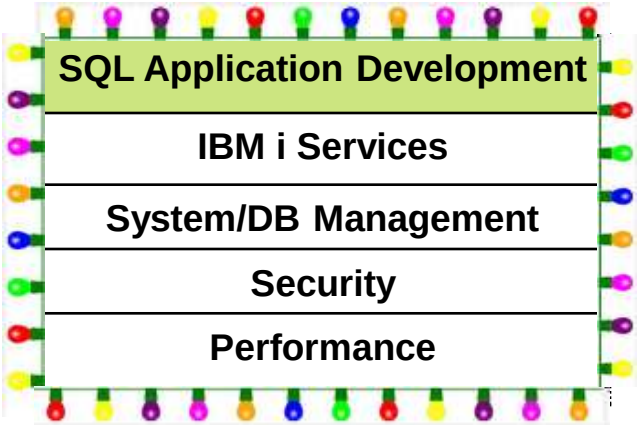




- For more information, go to:
  - www.ibm.com/systems/i/db2/webquery

DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

- **Find and Manipulate Data**
  - Regular Expressions
  - New Scalar functions for padding
- **Deploy complex UDTFs using SQL**
  - Pipelined Table Functions
- **Easy access to environment detail**
  - New DB2 built-in Global Variables
  - API access to DB2 built-in Global variables
- **Improved SQL programmer experience**
  - IBM i Debugger enhanced for SQL routines and triggers
    - STRDBG command and Graphical System Debugger
  - SQL messages changed to return SQL column names
  - RUNSQL support for output listings

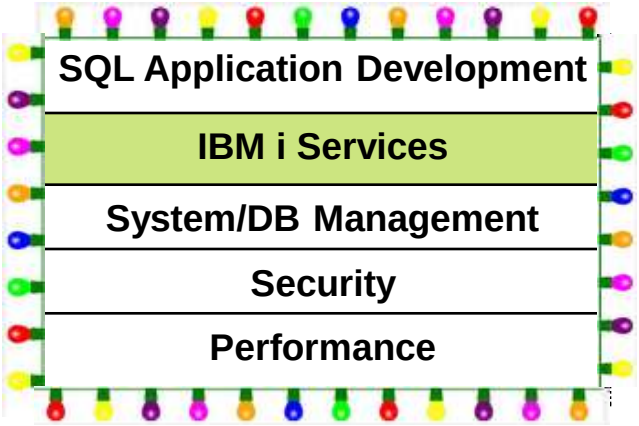


DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

Enhancement	IBM i 7.2	IBM i 7.1	IBM i 6.1
Regular Expressions	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
LPAD & RPAD Scalar Functions	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
Pipelined Table Functions	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
JOB_NAME & SERVER_MODE_JOB_NAME DB2 built-in global variables	DB2 PTF Group SF99702 Level 3	-	-
Debugger support for EVAL within SQL Routines and SQL Triggers	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
Some SQL messages changed to return SQL column names	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
LOCK TABLE ability to target non-*FIRST members	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
RUNSQL control of output listing	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-

DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

- **Journal Catalog**
  - Local and Remote Journal insight
  - Environmental and statistical
- **Library List Catalog**
  - Probe the environment of a connection
  - Easy access to Library & Schema Name
- **Reply List Catalog**
  - Use an SQL query to assess readiness
  - Programmatically find an unused entry
  - Compare production and HA/DR machines for equivalency
- **Job Log Table Function**
  - Incorporate job log queries into application logic
  - Consume job log with failure data collection

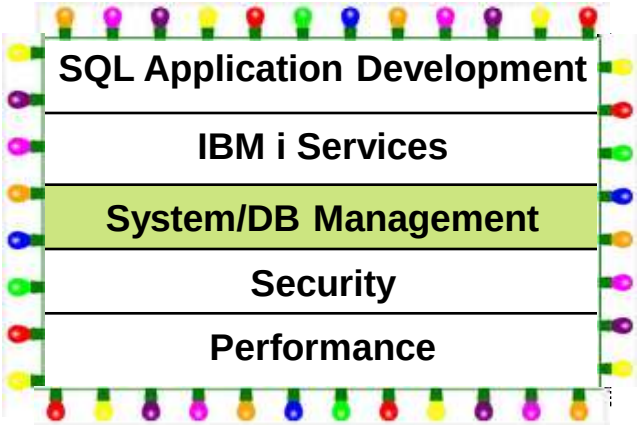


DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

Enhancement	IBM i 7.2	IBM i 7.1	IBM i 6.1
QSYS2/JOURNAL_INFO	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	DB2 PTF Group SF99601 Level 33
QSYS2/REPLY_LIST_INFO	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	DB2 PTF Group SF99601 Level 33
QSYS2/LIBRARY_LIST_INFO	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	DB2 PTF Group SF99601 Level 33
QSYS2/JOBLOG_INFO	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	DB2 PTF Group SF99601 Level 33
SYSTOOLS/GROUP_PTF_CURRENCY	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-

DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

- **Navigator for i**
  - DB dialogs improved
  - Improved performance
- **System Limits Phase 3 – IFS**
  - Discover large IFS Objects
  - Recognize accumulation of IFS objects
  - React to runaway conditions



- **Database Monitor user filtering improved**
  - Choose between 1-10 users or groups to monitor
  - Navigator or Command interface options
  - Leveraged by Guardium Database Activity Monitor

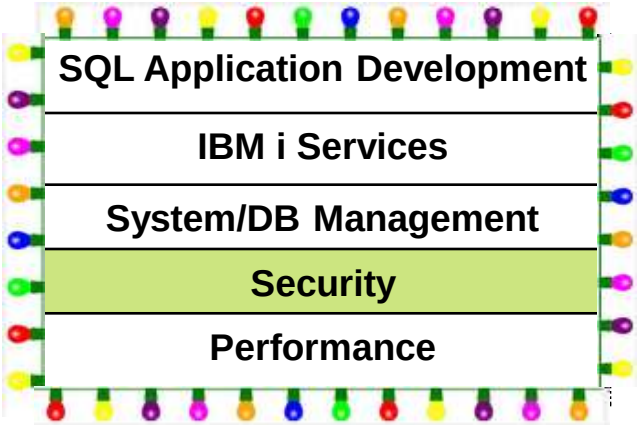
- **PTF Group "currency" view**
  - Query this view and DB2 for i will:
    - Consume a live XML feed from IBM PSP Group information
    - Access the current partition and extract the PTF Group information
    - Use the SQL Query Engine to perform a live comparison

DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

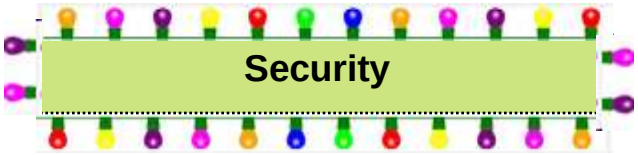
Enhancement	IBM i 7.2	IBM i 7.1	IBM i 6.1
IBM i Navigator – database dialogs improved	HTTP PTF Group SF99713 Level 4	HTTP PTF Group SF99368 Level 30	DB2 PTF Group SF99115 Level 41
System Limits Phase 3 – IFS	DB2 PTF Group SF99702 Level 3	-	-
STRDBMON FTRUSER(1-10)	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
QUSRJOBI() retrieval of DB2 built-in global variables	DB2 PTF Group SF99702 Level 3	-	-
Navigator - SQL Details for Jobs retrieval of DB2 built-in global variables	DB2 PTF Group SF99702 Level 3 & i Access for Windows 7.1 SP SIxxxxx Or HTTP PTF Group SF99713 Level 4	-	-

DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

- **Use DB2 tooling while protecting sensitive data**
  - Database monitor support for secure host variables
  - Navigator and Command Support
- **Row & Column Access Control**
  - Easily re-deploy rule text with **OR REPLACE** (applies to row permissions & column masks)



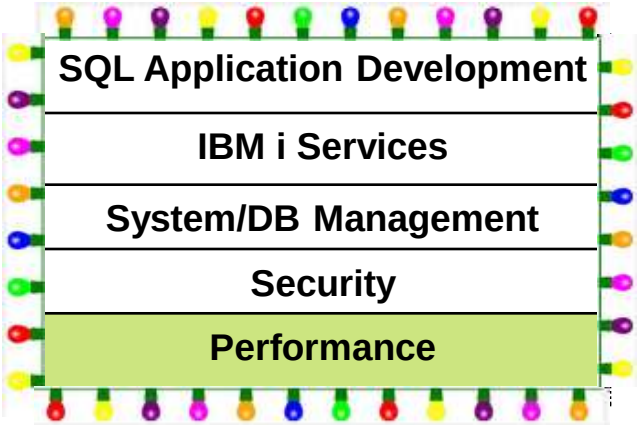
DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1



Enhancement	IBM i 7.2	IBM i 7.1	IBM i 6.1
Allow OR REPLACE on CREATE MASK & CREATE PERMISSION	**DB2 PTF Group SF99702 Level 3**	**-**	**-**
STRDBMON HOSTVAR(*SECURE)	**DB2 PTF Group SF99702 Level 3**	**DB2 PTF Group SF99701 Level 32**	**-**

DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

- **SQL Performance Analysis**
  - Visual Explain enhanced for UDTFs
    - Drill down to see program statements
- **“In Memory” just got easier**
  - SQL language control of preference
  - Full Navigator support
- **More context for Indexes**
  - Easier to gauge index values
  - Catalog and Navigator interfaces
- **Automate collection of DB performance details**
  - Capture & Import into Navigator for i
- **Database monitor efficiency improved**
  - Condensed host variable values
  - Reduced I/O and CPU overhead
  - Navigator or Command Interface options



DB2 for i: Enhancements for IBM i 7.1 TR9 and IBM i 7.2 TR1

Enhancement	IBM i 7.2	IBM i 7.1	IBM i 6.1
Visual Explain UDTF → Specific Program Statements	i Access for Windows 7.1 SP SIxxxxx	i Access for Windows 7.1 SP SIxxxxx	i Access for Windows 7.1 SP SIxxxxx
STRDBMON HOSTVAR(*CONDENSED)	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
Automate DBE tasks for Navigator	DB2 PTF Group SF99702 Level 3	DB2 PTF Group SF99701 Level 32	-
Comprehensive support for 'Keep In Memory'	i Access for Windows 7.1 SP SIxxxxx & DB2 PTF Group SF99702 Level 3	i Access for Windows 7.1 SP SIxxxxx & DB2 PTF Group SF99701 Level 32	-
Index creation date added to Navigator's Schemas & DB2 for i Catalogs	i Access for Windows 7.1 SP SIxxxxx & DB2 PTF Group SF99702 Level 3	i Access for Windows 7.1 SP SIxxxxx & DB2 PTF Group SF99701 Level 32	-

Pipelined Table Functions

Pipelined functions allow the flexibility to code more complex User-Defined Table Functions purely in SQL without having to build & manage a compiled program or service program.

This makes the UDTF much more portable as it eliminates compiler specifics.

Traditional External UDTF

```
CRTSQLCI OBJ(applib/producer) SRCFILE(appsrc/c80)
    COMMIT(*NONE) OUTPUT(*PRINT)
    OPTION(*NOGEN)

CRTCMOD MODULE(applib/producer)
    SRCFILE(qtemp/qsqltemp)
    TERASPACE(*YES) STGMDL(*INHERIT)

CRTSRVPGM SRVPGM(applib/udfs)
    MODULE(applib/producer)
    EXPORT(*SRCFILE) SRCFILE(BUILDLIB/APPEXP)
    SRCMBR(PRODUCER) ACTGRP(*CALLER)

CREATE FUNCTION producer()
    RETURNS TABLE (largest_table_sizes INTEGER )
    EXTERNAL NAME applib.udfs(producer)
    LANGUAGE C PARAMETER STYLE SQL
```

Pipelined SQL UDTF

```
CREATE FUNCTION producer()
 RETURNS TABLE ( largest_table_sizes INTEGER )
 LANGUAGE SQL
BEGIN

FOR LimitCursor CURSOR FOR
  SELECT CURRENT_VALUE FROM
    QSYS2.SYSLIMITS WHERE
    SIZING_NAME = 'MAXIMUM NUMBER OF ALL ROWS'
     ORDER BY  CURRENT_VALUE DESC
    DO
      PIPE CURRENT_VALUE;
  END FOR;

  RETURN;
END;
```

Pipelined Table Functions

Ease of Implementation

- Pipelined table functions are easy to implement. While “set-at-a-time” processing remains the best practice, pipelined functions provide a user friendly alternative.

Greater runtime flexibility

- Pipelined table functions allow the flexibility to programmatically create 'virtual' tables with greater control than SELECT or CREATE VIEW can provide.
  - ❑ UDTF input parameters
  - ❑ Ability to handle errors and warnings
  - ❑ Application logging
  - ❑ References to multiple databases in a single query
  - ❑ Customized join behavior

Big Data / Analytics / Performance

- In those scenarios where only a subset of the result table is consumed, pipelined functions are preferred.

Interoperability / Portability

- Compatible with DB2 for LUW support

Pipelined Table Functions

```
CREATE OR REPLACE FUNCTION Group_check(P_PTF_GROUP_NAME VARCHAR(7) )
RETURNS TABLE
  (V_PTF_GROUP_NAME CHAR(7), V_PTF_GROUP_DESCRIPTION VARCHAR(100),
   V_LEVEL_DETAIL CLOB(1K))
  LANGUAGE SQL  CARDINALITY 2  NOT FENCED  DISALLOW PARALLEL
BEGIN
   DECLARE SkipIt INTEGER;
   DECLARE TARGET_RDB VARCHAR(128);
   DECLARE v_PTF_GROUP_NAME CHAR(7);
   DECLARE v_PTF_GROUP_DESCRIPTION VARCHAR(100);
   DECLARE v_PTF_GROUP_LEVEL INTEGER;
   DECLARE CONTINUE HANDLER FOR SQLEXCEPTION
       BEGIN
         PIPE (NULL, NULL, TARGET_RDB CONCAT ' is not accessible ');
         SET SkipIt = 1;
       END;

SET (TARGET_RDB,SkipIt,v_PTF_GROUP_LEVEL) = ('lpdac710',0,NULL);

SELECT PTF_GROUP_NAME, PTF_GROUP_DESCRIPTION, PTF_GROUP_LEVEL INTO
v_PTF_GROUP_NAME, v_PTF_GROUP_DESCRIPTION, v_PTF_GROUP_LEVEL FROM
lpdac710.QSYS2.GROUP_PTF_INFO WHERE P_PTF_GROUP_NAME = PTF_GROUP_NAME AND
PTF_GROUP_STATUS = 'INSTALLED' ORDER BY PTF_GROUP_LEVEL DESC FETCH FIRST 1 ROWS
   ONLY;
```

Pipelined Example…
Multiple-RDB with error handling

Pipelined Table Functions

```
IF (SkipIt = 0 AND v_PTF_GROUP_LEVEL IS NOT NULL) THEN
 PIPE (v_PTF_GROUP_NAME, v_PTF_GROUP_DESCRIPTION, TARGET_RDB CONCAT ' has level '
   CONCAT lower( v_PTF_GROUP_LEVEL ) CONCAT ' APPLIED');
END IF;
SET (TARGET_RDB,SkipIt,v_PTF_GROUP_LEVEL) = ('MysteryMachine',0,NULL);
SELECT PTF_GROUP_NAME, PTF_GROUP_DESCRIPTION, PTF_GROUP_LEVEL INTO
   v_PTF_GROUP_NAME, v_PTF_GROUP_DESCRIPTION, v_PTF_GROUP_LEVEL
       FROM MysteryMachine.QSYS2.GROUP_PTF_INFO WHERE P_PTF_GROUP_NAME =
   PTF_GROUP_NAME AND PTF_GROUP_STATUS = 'INSTALLED' ORDER BY PTF_GROUP_LEVEL DESC
       FETCH FIRST 1 ROWS ONLY;
IF (SkipIt = 0 AND v_PTF_GROUP_LEVEL IS NOT NULL) THEN
 PIPE (v_PTF_GROUP_NAME, v_PTF_GROUP_DESCRIPTION, TARGET_RDB CONCAT ' has level '
   CONCAT lower( v_PTF_GROUP_LEVEL ) CONCAT ' APPLIED');
END IF;
   RETURN;
END;

SELECT * FROM TABLE(Group_check('SF99701')) A;
```

SELECT * FROM TABLE(Group_check('SF99701')) A - Rchaptf3.rch.stglabs.ibm.com(Dbtest)

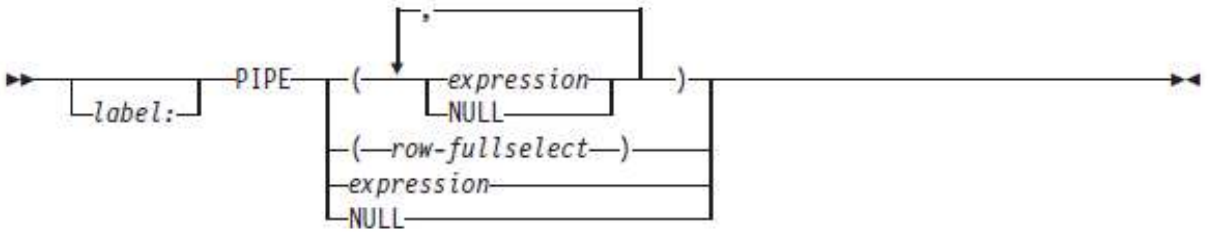
V_PTF_GROUP_NAME	V_PTF_GROUP_DESCRIPTION	V_LEVEL_DETAIL
SF99701	DB2 FOR IBM I ...	lpdac710 has level 30 APPLIED
-	-	MysteryMachine is not accessible

Pipelined Table Functions

Programmer notes…

- To SQE, a pipelined function is no different than other UDTFs
  - ❑ **CARDINALITY** matters – choose an appropriate setting
  - ❑ By default, SQE blocks rows
- If you use the *row-fullselect* capability to return multiple rows, beware that any failure encountered is not eligible to be consumed by a HANDLER
- Input parameters are modifiable, but not visible outside of the UDTF
- The **PIPE** statement cannot be used within embedded SQL and cannot be dynamically prepared. It is valid within SQL table functions only

Syntax



Regular Expressions

A regular expression is a powerful way of specifying a pattern for a complex search. Regular expression searches have been a mainstay in the Unix world for tools like awk, grep, etc.

Function or Predicate	Description
REGEXP_LIKE predicate	Searches for a regular expression pattern in a string and returns True or False
REGEXP_COUNT	Returns a count of the number of times that a pattern is matched in a string
REGEXP_INSTR	Returns the starting or ending position of the matched substring
REGEXP_SUBSTR	Returns one occurrence of a substring of a string that matches the pattern
REGEXP_REPLACE	Returns a modified version of the source string where occurrences of the pattern found in the source string are replaced with the specified replacement string

Select the employee number where the last name is spelled LUCCHESSI, LUCHESSI, or LUCHESI from the EMPLOYEE table without considering upper or lower case letters.

```
SELECT empno
FROM employee
WHERE REGEXP_LIKE(lastname,'luc+?hes+?i','i')
```

Regular Expressions

```
-- 7.1 version
CREATE OR REPLACE FUNCTION FindHits(v_search_string CLOB(1M), v_pattern
   varchar(32000) )
RETURNS TABLE (website_reference varchar(512))
LANGUAGE SQL
BEGIN
DECLARE V_Count INTEGER;
DECLARE LOOPVAR INTEGER DEFAULT 0;
SET V_Count = REGEXP_COUNT(v_search_string, v_pattern,1,'i');
IF v_pattern IS NULL OR LENGTH(v_pattern) = 0 THEN
   SET v_pattern = '(\w+\.)+((org)|(com)|(gov)|(edu))';
END IF;
WHILE LOOPVAR < V_Count DO
  SET LOOPVAR = LOOPVAR + 1;
  PIPE( REGEXP_SUBSTR(v_search_string,v_pattern, 1, LOOPVAR, 'i') );
END WHILE;

RETURN;
END;

SELECT * FROM TABLE(FindHits('Are you interested in any of these colleges:
   isu.EDU or www.umn.Edu? We could even visit WWW.wisc.edu if we have time.'))
   A;
```

Regular expression & Pipeline function example for IBM i 7.1

WEBSITE_REFERENCE
isu.EDU
www.umn.Edu
WWW.wisc.edu

Regular Expressions

```
-- 7.2 version
CREATE OR REPLACE FUNCTION FindHits(v_search_string CLOB(1M), v_pattern
   varchar(32000) DEFAULT '(\w+\.)+((org)|(com)|(gov)|(edu))' )
RETURNS TABLE (website_reference varchar(512))
LANGUAGE SQL
BEGIN
DECLARE V_Count INTEGER;
DECLARE LOOPVAR INTEGER DEFAULT 0;
SET V_Count = REGEXP_COUNT(v_search_string, v_pattern,1,'i');

WHILE LOOPVAR < V_Count DO
  SET LOOPVAR = LOOPVAR + 1;
  PIPE( REGEXP_SUBSTR(v_search_string,v_pattern, 1, LOOPVAR, 'i') );
END WHILE;

RETURN;
END;


SELECT * FROM TABLE(FindHits('Are you interested in any of these colleges:
   isu.EDU or www.umn.Edu? We could even visit WWW.wisc.edu if we have time.'))
   A;
```

Regular expression & Pipeline function example for IBM i 7.2

WEBSITE_REFERENCE
isu.EDU
www.umn.Edu
www.wisc.edu

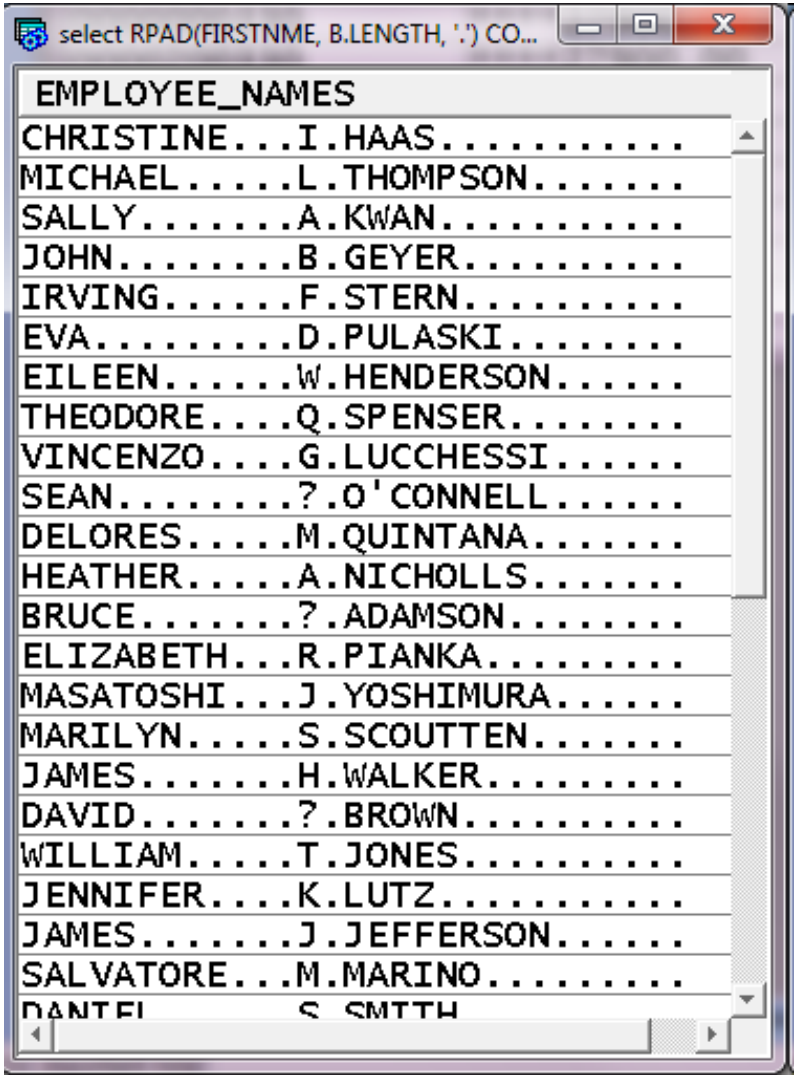
Regular Expressions vs OmniFind

Query Choice / Considerations	Regular Expressions	OmniFind Text Search Server for DB2 for i
Enablement	7.1 - TR9 7.2 - TR1	No charge option (5733OMF) available at 6.1 and higher
Finding matches	REGEXP_LIKE predicate	CONTAINS function
Search Target	Expression that returns a character string, graphic string, numeric, or datetime	Column within a table, Spool files, IFS stream files, and more
Search mechanism	Pattern matching (grep, awk, etc…)	Linguistic Variations Keyword, phrase matching, keyword variations (“mice” for “mouse”)
Performance consideration	No ability to use indexes for implementation	Text search index
Maintenance consideration	None	Index build & update(s)
Complementary services	REGEXP_COUNT, REGEXP_INSTR, REGEXP_SUBSTR & REGEXP_REPLACE	SCORE
Resources	SQL Reference	https://ibm.biz/IBMi_OMF

Controlled padding using LPAD & RPAD

- **Push column formatting logic into your SQL queries**

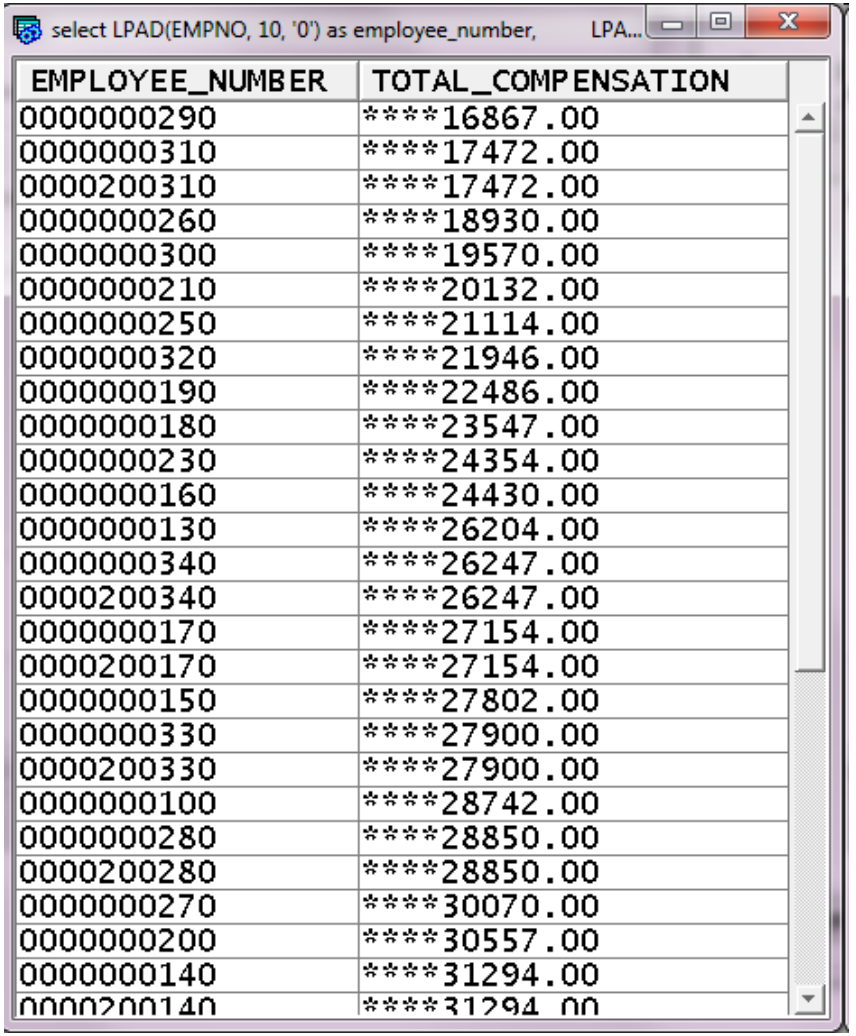
```
SELECT
RPAD(FIRSTNME, B.LENGTH, '.') CONCAT
CASE WHEN MIDINIT IS NULL OR
MIDINIT = ' ' THEN '?' ELSE
RPAD(MIDINIT, C.LENGTH, '.') END
CONCAT '.‘ CONCAT
RPAD(LASTNAME, D.LENGTH, '.')
AS Employee_Names
FROM toystore5.employee A,
(SELECT LENGTH FROM QSYS2.SYSCOLUMNS WHERE
TABLE_NAME = 'EMPLOYEE' AND COLUMN_NAME = 'FIRSTNME'
AND TABLE_SCHEMA = 'TOYSTORE5') B,
(SELECT LENGTH FROM QSYS2.SYSCOLUMNS WHERE
TABLE_NAME = 'EMPLOYEE' AND COLUMN_NAME = 'MIDINIT'
AND TABLE_SCHEMA = 'TOYSTORE5') C,
(SELECT LENGTH FROM QSYS2.SYSCOLUMNS WHERE
TABLE_NAME = 'EMPLOYEE' AND COLUMN_NAME = 'LASTNAME‘
AND TABLE_SCHEMA = 'TOYSTORE5') D
```



Controlled padding using LPAD & RPAD

- **Works with many data types**

```
SELECT
LPAD(EMPNO, 10, '0')
as employee_number,
LPAD(salary + bonus + comm, 12, '*')
as total_compensation

FROM toystore5.employee
ORDER BY total_compensation;
```



EMPLOYEE_NUMBER	TOTAL_COMPENSATION
0000000290	****16867.00
0000000310	****17472.00
0000200310	****17472.00
0000000260	****18930.00
0000000300	****19570.00
0000000210	****20132.00
0000000250	****21114.00
0000000320	****21946.00
0000000190	****22486.00
0000000180	****23547.00
0000000230	****24354.00
0000000160	****24430.00
0000000130	****26204.00
0000000340	****26247.00
0000200340	****26247.00
0000000170	****27154.00
0000200170	****27154.00
0000000150	****27802.00
0000000330	****27900.00
0000200330	****27900.00
0000000100	****28742.00
0000000280	****28850.00
0000200280	****28850.00
0000000270	****30070.00
0000000200	****30557.00
0000000140	****31294.00
0000200140	****31294.00

SQL column names in messages

- SQL Column names are much easier to understand than Field names
- SQL Language constructs like **FOR SYSTEM NAME** and **FOR COLUMN** make it easier to recognize and understand failures, because the name is chosen instead of system generated
- A behavior change has been made, to give preference to SQL column names in failure messages.

The list of SQL messages changed to return SQL column names is:
SQL0190, SQL0196, SQL0197, SQL0404, SQL0406, SQL0407, SQL0415

Now…

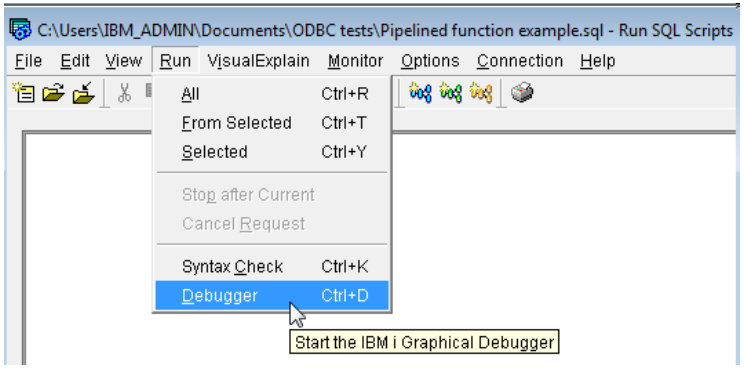
```
> update still_the_one_to_see.table_with_long_name set employee_surname_value =
'Jones-Smith-ExtraSpecial-Calafragilistic' where employee_first_name = 'Bob'

SQL State: 22001
Vendor Code: -404
Message: [SQL0404] Value for column or variable EMPLOYEE_SURNAME_VALUE too long.
Cause . . . . . :   An INSERT, UPDATE, MERGE, SET, VALUES INTO, or GET DIAGNOSTICS
statement specifies a value that is longer than the maximum length string that can
be stored in EMPLOYEE_SURNAME_VALUE. The length of EMPLOYEE_SURNAME_VALUE is 30 and
the length of the string is 40. Recovery  . . . :   Reduce the length of the string
from 40 to a maximum of 30 and try the request again.
```

Before…

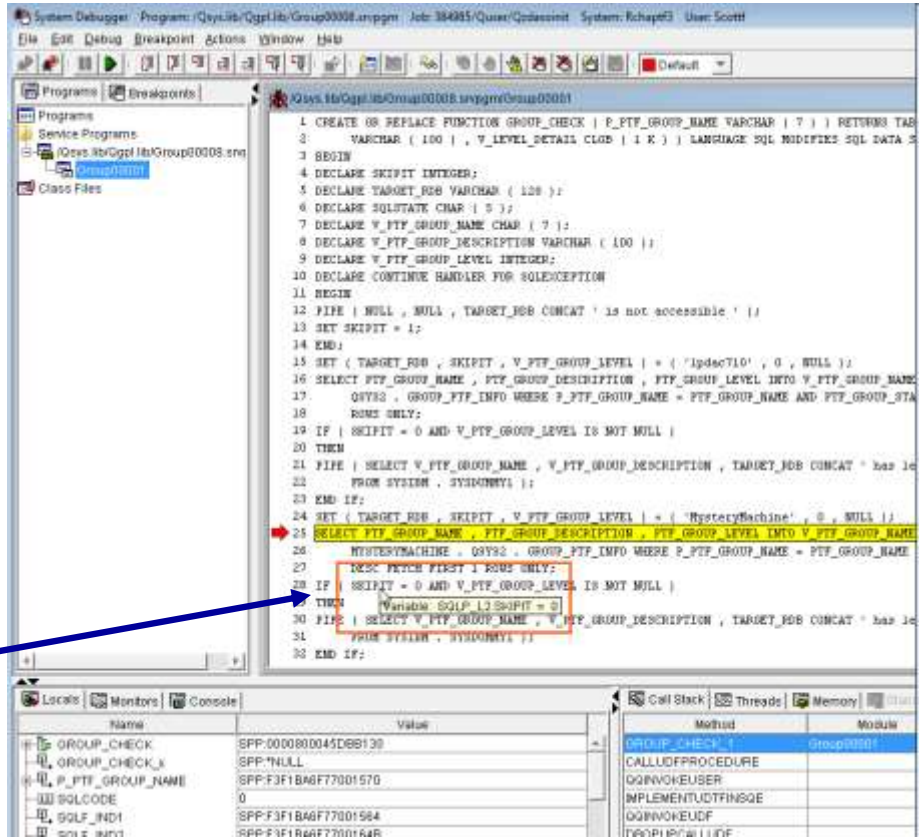
```
> update still_the_one_to_see.table_with_long_name set employee_surname_value =
'Jones-Smith-ExtraSpecial-Calafragilistic' where employee_first_name = 'Bob'

SQL State: 22001
Vendor Code: -404
Message: [SQL0404] Value for column or variable EMPLO00002 too long. Cause . . . . .
:   An INSERT or UPDATE statement or a SET or VALUES INTO statement or a GET
DIAGNOSTICS statement specifies a value that is longer than the maximum length
string that can be stored in EMPLO00002. The length of EMPLO00002 is 30 and the
length of the string is 40. Recovery  . . . :   Reduce the length of the string from
40 to a maximum of 30 and try the request again.
```

IBM i Debugger enhanced for SQL

- STRDBG (Green Screen) & IBM i Debugger (Graphical) can be used to debug SQL procedures, functions and triggers
- Since DB2 for i generates the underlying ILE C code, debug has been complex
- In TR7, the debugger added support for stepping through SQL source
- With TR9, the debugger is enhanced to support EVAL for some SQL variables



New…



IBM i Services

DB2 for i Built-in Global Variables

- The qualified job name of the current connection is easily accessed
- When SQL Server Mode is used, the job name of the application instance which owns the connection is accessed through SERVER_MODE_JOB_NAME
- Use these variables to deploy advanced logic in triggers, RCAC rules, and more

New with IBM i 7.2 SF99702 Level 3

Available with base IBM i 7.2

Variable name	Schema	Data Type	Size
JOB_NAME	**QSYS2**	**VARCHAR**	**28**
SERVER_MODE_JOB_NAME	**QSYS2**	**VARCHAR**	**28**
CLIENT_IPADDR	SYSIBM	VARCHAR	128
CLIENT_HOST	SYSIBM	VARCHAR	255
CLIENT_PORT	SYSIBM	INTEGER	-
PACKAGE_NAME	SYSIBM	VARCHAR	128
PACKAGE_SCHEMA	SYSIBM	VARCHAR	128
PACKAGE_VERSION	SYSIBM	VARCHAR	64
ROUTINE_SCHEMA	SYSIBM	VARCHAR	128
ROUTINE_SPECIFIC_NAME	SYSIBM	VARCHAR	128
ROUTINE_TYPE	SYSIBM	CHAR	1

DB2 for i Built-in Global Variables

Column masking example:

```
CREATE OR REPLACE MASK SSN_MASK ON TOYSTORE2.EMPLOYEE
   FOR COLUMN SSN
 RETURN CASE
 WHEN (QSYS2.JOB_NAME LIKE '%QZDAS%INIT%')
     THEN 'XXX-XX-' CONCAT
          SUBSTR(SSN,8,4)
 ELSE SSN END ENABLE;

ALTER TABLE TOYSTORE2.EMPLOYEE
 ACTIVATE COLUMN ACCESS CONTROL;

SELECT LASTNAME, EMPNO, SSN
 FROM TOYSTORE2.EMPLOYEE ORDER BY 1;
```

Protect sensitive data using Built-in Global Variables

SELECT LASTNAME, EMPNO, SSN FROM TOYSTORE2.EMPLOY...

LASTNAME	EMPNO	SSN
ADAMSON	000150	XXX-XX-0015
ALONZO	200340	XXX-XX-0034
BROWN	000200	XXX-XX-0020
GEYER	000050	XXX-XX-0005
GOUNOT	000340	XXX-XX-0034
HAAS	000010	XXX-XX-0001
HEMMINGER	200010	XXX-XX-0001
HENDERSON	000090	XXX-XX-0009
JEFFERSON	000230	XXX-XX-0023
JOHN	200220	XXX-XX-0022
JOHNSON	000260	XXX-XX-0026
JONES	000210	XXX-XX-0021
KWAN	000030	XXX-XX-0003
LEE	000330	XXX-XX-0033
LUCCHESSI	000110	XXX-XX-0011
LUTZ	000220	XXX-XX-0022
MARINO	000240	XXX-XX-0024
MEHTA	000320	XXX-XX-0032
MONTEVERDE	200240	XXX-XX-0024
NATZ	200140	XXX-XX-0014
NICHOLLS	000140	XXX-XX-0014
O'CONNELL	000120	XXX-XX-0012
ORLANDO	200120	XXX-XX-0012

Tracking Important System Limits – Phase 3

Phase 3 – Instrument object limits for some IFS file systems

File systems instrumented for object limits:

- **"root" (/) file system**
- **Open systems file system (QOpenSys)**
- **Document library services file system (QDLS)**
- **User-defined file systems (UDFSs)**

Discover trends, high consumers and more

Limit description	Limit value
Number of objects linked in a directory	0
Maximum number of directories linked in a directory	1,000,000
Maximum number of file system objects in *SYSBAS ASPs	2,147,483,647
Maximum number of file system objects in an independent ASP	2,147,483,647
Maximum number of document library objects in a folder	65510
Number of document library objects in the system ASP	0
Maximum number of document library objects in a user ASP	349,000
Maximum number of bytes in a stream file	1,099,511,627,776
Maximum number of bytes in a document	2,147,483,647

Tracking Important System Limits – Phase 3

Limit description	Limit ID	Limit value	Floor	Increment
Number of objects linked in a directory	18402	0	100,000	10,000
Maximum number of directories linked in a directory	18403	1,000,000	1,000	1,000
Maximum number of file system objects in *SYSBAS ASPs	18404	2,147,483,647	100,000	10,000
Maximum number of file system objects in an independent ASP	18405	2,147,483,647	100,000	10,000
Maximum number of document library objects in a folder	18406	65510	1,000	500
Number of document library objects in the system ASP	18407	0	100,000	10,000
Maximum number of document library objects in a user ASP	18408	349,000	100,000	10,000
Maximum number of bytes in a stream file	18409	1,099,511,627,776	16,777,216	1,048,576
Maximum number of bytes in a document	18409	2,147,483,647	16,777,216	1,048,576

http://iprodeveloper.com/systems-management/ondemand-tracking-important-system-limits-ibm-i

http://iprodeveloper.com/systems-management/gain-big-insights-db2-i-system-limits-phase-2

Tracking Important System Limits – Phase 3

```
-- Examine the top consumers of IFS object limits
SELECT SIZING_NAME, IFS_PATH_NAME, OBJECT_TYPE,
CURRENT_VALUE, USER_NAME, LASTCHG
 FROM QSYS2.SYSLIMITS
WHERE LIMIT_CATEGORY = 'FILE SYSTEM'
 ORDER BY CURRENT_VALUE DESC
```

SELECT SIZING_NAME, IFS_PATH_NAME, OBJECT_TYPE, CURRENT_VALUE, USER_NAME, LASTCHG FROM QS ... - Lp91ut27.rch.stglabs.ibm.com(Lp91ut27)

SIZING_NAME	IFS_PATH_NAME	OBJECT_TYPE	CURRENT_VALUE	USER_NAME	LASTCHG
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/tmp/brms/flightrec	*STMF	288990181	DBPGMR	2014-09-22 06:07...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/tmp/brms/qbrms	*STMF	222281197	QBRMS	2014-09-22 01:46...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	51770672	QPGMR	2014-09-25 22:04...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	50722063	QPGMR	2014-09-25 22:02...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	49673479	GUOQI	2014-09-25 03:29...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	30409113	SQBATSECO	2014-10-01 11:56...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	29360519	SQBATUSER	2014-10-01 09:29...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	28311932	DBSECOFR	2014-10-01 07:54...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	27263335	QSECOFR	2014-10-01 07:30...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	26214725	DBSECOFR	2014-10-01 06:48...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/HL	*STMF	26193173	QQSECOFR	2014-10-01 06:55...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/QSR/QSR/FR	*STMF	25166123	DBSECOFR	2014-10-01 06:05...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.CF	*STMF	23068672	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/file.cf2	*STMF	23068672	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	23068672	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/file.df2	*STMF	23068672	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.CF	*STMF	22020096	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	22020096	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.CF	*STMF	20971520	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	20971520	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.CF	*STMF	19922944	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	19922944	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	18874368	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	17825792	SQBATSECO	2014-10-01 05:35...
MAXIMUM NUMBER OF BYTES IN A STREAM FILE	/LBMV5FR01/SUBDIR1/FILE.DF	*STMF	16777216	SQBATSECO	2014-10-01 05:35...

QSYS2.JOURNAL_INFO – View

- Use SQL to retrieve detail for local & remote journals
- Information from **QjoRetrieveJournalInformation()** API, **RJRN0100** format.
- The view returns Key 1 & 3 information, one row == information about one journal.

Key	Input Type	Field
1	N/A	Journal receiver directory information
2	CHAR(10)	Journaled object information
3	CHAR(38)	Remote journal information
4	CHAR(10)	Auxiliary storage pool device name

```
-- Which remote journals are the most heavily used?
SELECT JOURNALED_OBJECTS, A.* FROM QSYS2.JOURNAL_INFO A WHERE
NUMBER_REMOTE_JOURNALS > 0 AND JOURNALED_OBJECTS IS NOT NULL
ORDER BY JOURNALED_OBJECTS DESC

-- Find journals nearing the limit of journaled objects:
SELECT * FROM QSYS2.JOURNAL_INFO WHERE
JOURNALED_OBJECT_LIMIT = '*MAX250K' AND
JOURNALED_OBJECTS > 200000
```

QSYS2.JOURNAL_INFO – View

- Use this new source of information to achieve better journal management
- Recognize conditions that require attention

```
-- which remote journals fell the farthest behind this week?
SELECT MAXIMUM_TIME_BEHIND, MAXIMUM_BEHIND_TIMESTAMP,
ESTIMATED_TIME_BEHIND, TOTAL_SIZE_JOURNAL_RECEIVERS,
 RTRIM(ATTACHED_JOURNAL_RECEIVER_LIBRARY) CONCAT '/' CONCAT
RTRIM(ATTACHED_JOURNAL_RECEIVER_NAME) AS JrnName, A.*
FROM QSYS2.JOURNAL_INFO A WHERE
MAXIMUM_BEHIND_TIMESTAMP > CURRENT TIMESTAMP - 7 DAYS AND
MAXIMUM_TIME_BEHIND > 0 AND MAXIMUM_TIME_BEHIND IS NOT NULL
ORDER BY MAXIMUM_TIME_BEHIND DESC FETCH FIRST 10 ROWS ONLY
```

SELECT MAXIMUM_TIME_BEHIND, MAXIMUM_BEHIND_TIMESTAMP, ESTIMATED_TIME_BEHIND, TOTAL_SIZE_JO ... - Z1014p12.rch.stglabs.ibm.com(Z1014p12)

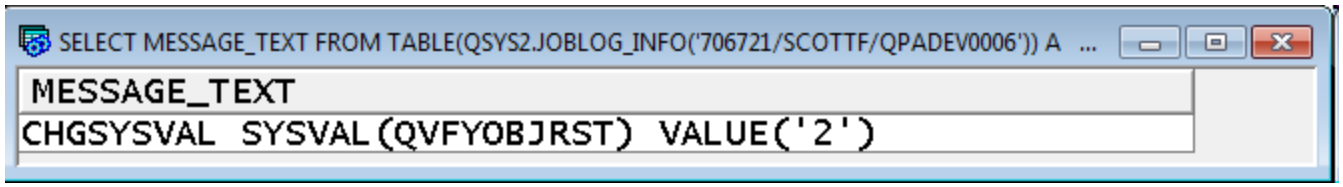
MAXIMUM_TIME_BEHIND	MAXIMUM_BEHIND_TIMESTAMP	ESTIMATED_TIME_BEHIND	TOTAL_SIZE_JOURNAL_RECEIVERS	JRNNAME
11190	2014-10-02 12:04:11.00...	0	872	BHSARCASM/RCV1
8540	2014-10-02 12:09:12.00...	8540	872	BHPIZZAZZ/RCV1
8360	2014-10-02 12:06:07.00...	0	908	BHSARCASM/RCVA1
5320	2014-10-02 12:16:54.00...	0	872	BHSPARKLE/RCVA1
4370	2014-10-02 12:37:44.00...	4370	11376	BHSRC37/RCV17

QSYS2.JOBLOG_INFO – UDTF

- Job logs contain essential information
  - Commonplace to examine multiple messages to understand a single failure
  - User command activity - REQUEST messages appear in the job log
  - Job logs are frequently requested by IBM Service
- The JOBLOG_INFO() UDTF provides a new service for application developers and system managers
- A single parameter indicates the target job.
- '*' can be used to indicate use the current job as the target.

Example 1: Find the most recently executed command in a target job

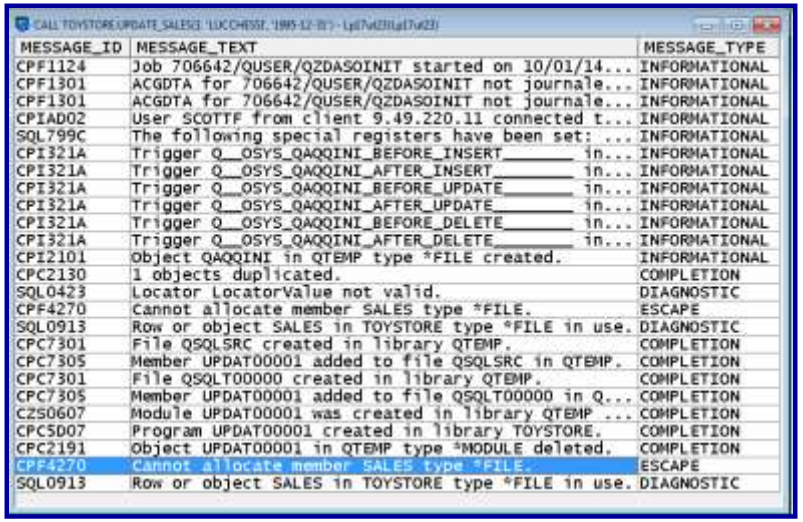
```
SELECT MESSAGE_TEXT FROM
   TABLE(QSYS2.JOBLOG_INFO('706721/SCOTTF/QPADEV0006')) A
  WHERE A.MESSAGE_TYPE = 'REQUEST'
  ORDER BY ORDINAL_POSITION DESC
  FETCH FIRST 1 ROW ONLY
```



QSYS2.JOBLOG_INFO – UDTF

Example 2: Automatic consumption of job log in an application

```
CREATE TABLE APPLIB.Joblog_Detail AS
    (SELECT * FROM TABLE(QSYS2.JOBLOG_INFO('*')) A)
    WITH NO DATA;

CREATE OR REPLACE PROCEDURE TOYSTORE.UPDATE_SALES
  (IN P_NEW_SALES INTEGER, IN P_SALES_PERSON VARCHAR(100),
   IN P_SALES_DATE DATE)
LANGUAGE SQL MODIFIES SQL DATA
BEGIN

Mainline: BEGIN
  DECLARE EXIT HANDLER FOR SQLEXCEPTION
   BEGIN
    INSERT INTO APPLIB.Joblog_Detail
      SELECT * FROM TABLE(QSYS2.JOBLOG_INFO('*')) A
   END;

  UPDATE TOYSTORE.SALES
   SET SALES = SALES + P_NEW_SALES
    WHERE SALES_PERSON = P_SALES_PERSON
     AND SALES_DATE = P_SALES_DATE;
END Mainline;
END ;
```



MESSAGE_ID	MESSAGE_TEXT	MESSAGE_TYPE
CPF1124	Job 706642/QUSER/QZDASOINIT started on 10/01/14...	INFORMATIONAL
CPF1301	ACGDTA for 706642/QUSER/QZDASOINIT not journale...	INFORMATIONAL
CPF1301	ACGDTA for 706642/QUSER/QZDASOINIT not journale...	INFORMATIONAL
CPIAD02	User SCOTTF from client 9.49.220.11 connected t...	INFORMATIONAL
SQL799C	The following special registers have been set: ...	INFORMATIONAL
CPI321A	Trigger Q__OSYS_QAQQINI_BEFORE_INSERT_____ in...	INFORMATIONAL
CPI321A	Trigger Q__OSYS_QAQQINI_AFTER_INSERT_____ in...	INFORMATIONAL
CPI321A	Trigger Q__OSYS_QAQQINI_BEFORE_UPDATE_____ in...	INFORMATIONAL
CPI321A	Trigger Q__OSYS_QAQQINI_AFTER_UPDATE_____ in...	INFORMATIONAL
CPI321A	Trigger Q__OSYS_QAQQINI_BEFORE_DELETE_____ in...	INFORMATIONAL
CPI321A	Trigger Q__OSYS_QAQQINI_AFTER_DELETE_____ in...	INFORMATIONAL
CPI2101	Object QAQQINI in QTEMP type *FILE created.	INFORMATIONAL
CPC2130	1 objects duplicated.	COMPLETION
SQL0423	Locator LocatorValue not valid.	DIAGNOSTIC
CPF4270	Cannot allocate member SALES type *FILE.	ESCAPE
SQL0913	Row or object SALES in TOYSTORE type *FILE in use.	DIAGNOSTIC
CPC7301	File QSQLSRC created in library QTEMP.	COMPLETION
CPC7305	Member UPDAT00001 added to file QSQLSRC in QTEMP.	COMPLETION
CPC7301	File QSQLT00000 created in library QTEMP.	COMPLETION
CPC7305	Member UPDAT00001 added to file QSQLT00000 in Q...	COMPLETION
CZS0607	Module UPDAT00001 was created in library QTEMP ...	COMPLETION
CPC5D07	Program UPDAT00001 created in library TOYSTORE.	COMPLETION
CPC2191	Object UPDAT00001 in QTEMP type *MODULE deleted.	COMPLETION
CPF4270	Cannot allocate member SALES type *FILE.	ESCAPE
SQL0913	Row or object SALES in TOYSTORE type *FILE in use.	DIAGNOSTIC

QSYS2.REPLY_LIST_INFO – view

- One reply list handles system wide automatic response to messages
- Now, SQL can be used to:
  - Compare (exception join) the configuration of two machines
  - Determine whether a specific sequence number is already in use
  - Confirm whether setup is complete
- REPLY_LIST_INFO matches the Work Reply List Entry (WRKRPYLE) command behavior of allowing *PUBLIC users to view the reply list information

Example:

```
SELECT * FROM QSYS2.REPLY_LIST_INFO
    WHERE message_ID like 'CPA%'
```

SELECT * FROM REPLY_LIST_INFO where message_ID like 'CPA%' - Lp24ut27.rch.stglabs.ibm.com(Lp24ut27)

SEQUENCE_NUMBER	MESSAGE_ID	COMPARISON_DATA	COMPARISON_DATA_OFFSET	REPLY_DATA	DUMP_JOB
10	CPA0700	NONE	0	D	YES
3200	CPA32B2	NONE	0	I	NO
6666	CPA7025	TOYSTORE never fully saved.	24	I	NO
7025	CPA7025	NONE	0	I	NO

QSYS2.LIBRARY_LIST_INFO – view

- With direct access to the library list, SQL users can tap into the library list detail to:
  - Programmatically review the environment
  - Know when the library list needs to be adjusted
  - Derive information about the libraries
    - ❑ IASP Number
    - ❑ Schema vs Library name mapping
    - ❑ User vs System vs Product libraries
    - ❑ Above all else, order of libraries searched for unqualified objects

Example:
SELECT * FROM QSYS2.LIBRARY_LIST_INFO

select * from qsys2.library_list_info - Rchaptf3.rch.stglabs.ibm.com(Dbtest)

ORDINAL_POSITION	SCHEMA_NAME	SYSTEM_SCHEMA_NAME	TYPE	IASP_NUMBER	TEXT_DESCRIPTION
1	QSYS	QSYS	SYSTEM	0	System Library
2	QSYS2	QSYS2	SYSTEM	0	System Library...
3	QHLPSYS	QHLPSYS	SYSTEM	0	-
4	QUSRSYS	QUSRSYS	SYSTEM	0	System Library...
5	QIWS	QIWS	PRODUCT	0	-
6	toystore_located_on_a_hilltop	"toys0001"	USER	33	COLLECTION - c...
7	QGPL	QGPL	USER	0	General Purpos...
8	QTEMP	QTEMP	USER	0	-

SYSTOOLS.GROUP_PTF_CURRENCY View

```
SELECT * from SYSTOOLS.GROUP_PTF_CURRENCY
  WHERE PTF_GROUP_RELEASE = 'R720'
  ORDER BY ptf_group_level_available -
  ptf_group_level_installed DESC
```

select * from SYSTOOLS.GROUP_PTF_CURRENCY WHERE PTF_GROUP_RELEASE = 'R720' ORDER BY ptf_gr ... - Lp89ut27.rch.stglabs.ibm.com(Lp89ut27)

PTF_GROUP_CURRENCY	PTF_GROUP_ID	PTF_GROUP_TITLE	PTF_GROUP_LEVEL_INSTALLED	PTF_GROUP_LEVEL_AVAILABLE	PTF_GROUP_LAST_UPDATED_BY_IBM
UPDATE AVAILABLE	SF99716	720 Java	1	3	09/04/2014
UPDATE AVAILABLE	SF99702	720 DB2 for IBM i	1	2	07/31/2014
UPDATE AVAILABLE	SF99713	720 IBM HTTP Ser...	2	3	08/15/2014
INSTALLED LEVEL IS CURRENT	SF99720	Current Cumulati...	14101	14101	06/05/2014

Current or behind on service?

PTF Group Info

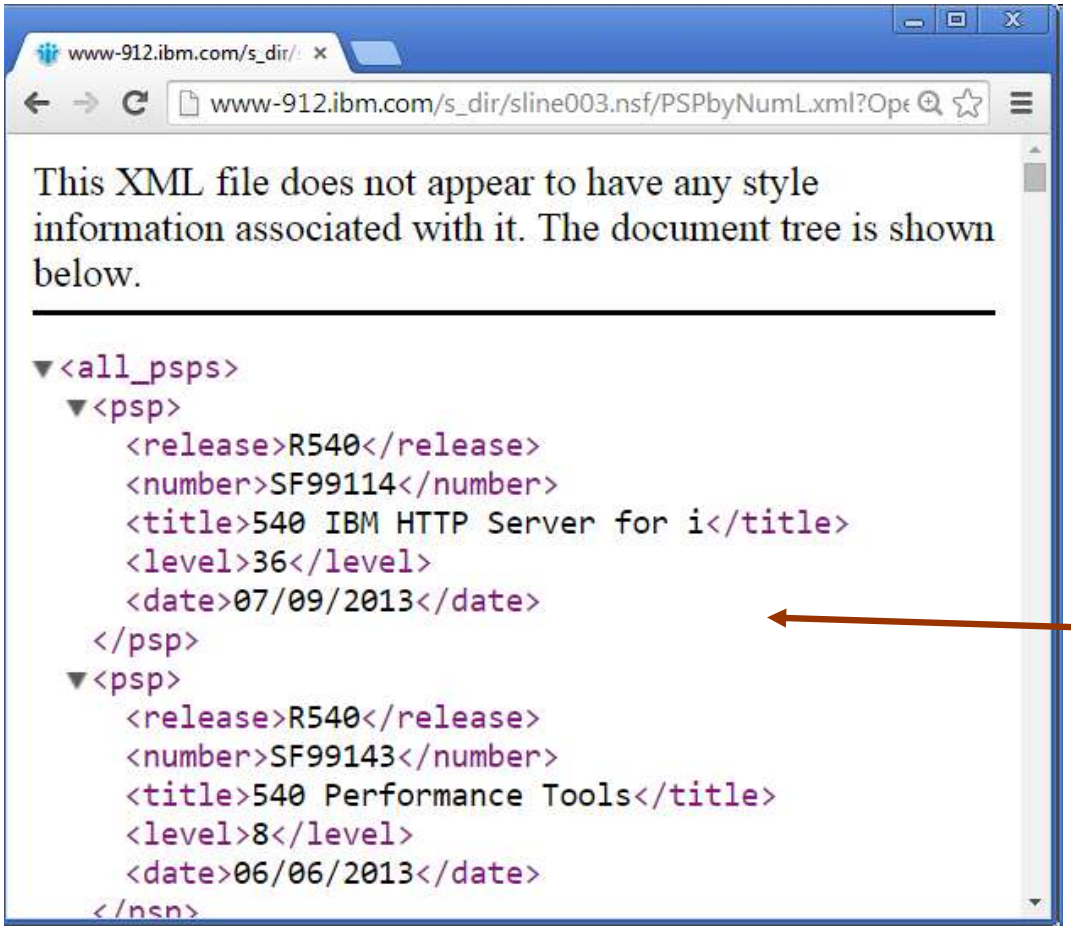
Level installed on this partition

Level available from IBM

Date that IBM last updated this group

SYSTOOLS.GROUP_PTF_CURRENCY View

http://www-912.ibm.com/s_dir/sline003.nsf/PSPbyNumL.xml?OpenView&count=500



XML namespace & structure

SYSTOOLS.GROUP_PTF_CURRENCY View

Study the XML structure to define the data to the HTTP function.

HTTP→XML document structure

```
   FROM   \
     XMLTABLE('/all_psps/psp' PASSING XMLPARSE(DOCUMENT  \
SYSTOOLS.HTTPGETBLOB('http://www-912.ibm.com/s_dir/sline003.nsf/PSPbyNumL.xml?OpenView&count=500' , ''))
      COLUMNS PSP_RELEASE CHAR(5)          PATH 'release',   \
              PSP_NUMBER  CHAR(7)          PATH 'number',    \
              PSP_TITLE   VARCHAR(1000)    PATH 'title',     \
              PSP_LEVEL   INTEGER          PATH 'level',     \
              PSP_DATE    CHAR(10)         PATH 'date'       \
         ) PSPS  \
```

TCP/IP Enablement:

'www-912.ibm.com' maps to 129.42.160.32

Enablement

IBM i TCP/IP configuration Technote:

http://www-01.ibm.com/support/docview.wss?uid=nas8N1018980

White papers:

- https://ibm.biz/XMLandDB2fori
- https://ibm.biz/HTTPandDB2fori

Developer resources

SYSTOOLS.GROUP_PTF_CURRENCY View

```
CREATE OR REPLACE VIEW SYSTOOLS/GROUP_PTF_CURRENCY FOR SYSTEM NAME GRPPTFCUR (
   PTF_GROUP_CURRENCY FOR COLUMN GRP_CRNCY   ,
   PTF_GROUP_ID FOR COLUMN GRP_ID      ,
   PTF_GROUP_TITLE FOR COLUMN GRP_TITLE   ,
   PTF_GROUP_LEVEL_INSTALLED FOR COLUMN GRP_LVL     ,
   PTF_GROUP_LEVEL_AVAILABLE FOR COLUMN GRP_IBMLVL ,
   PTF_GROUP_LAST_UPDATED_BY_IBM FOR COLUMN GRP_LSTUPD ,
   PTF_GROUP_RELEASE FOR COLUMN GRP_RLS     ,
   PTF_GROUP_STATUS_ON_SYSTEM FOR COLUMN GRP_SYSSTS )
   AS
   SELECT CASE WHEN ACTUAL.GRPPTF IS NULL THEN 'PTF GROUP DOES NOT EXIST ON ' CONCAT CURRENT SERVER WHEN
   PSPS.PSP_NUMBER IS NULL THEN 'PSP INFORMATION NOT AVAILABLE' WHEN ACTUAL.GRPPTF = PSPS.PSP_NUMBER AND
   ACTUAL.PTF_GROUP_LEVEL = PSPS.PSP_LEVEL THEN 'INSTALLED LEVEL IS CURRENT' WHEN ACTUAL.GRPPTF = PSPS.PSP_NUMBER
   AND ACTUAL.PTF_GROUP_LEVEL < PSPS.PSP_LEVEL THEN 'UPDATE AVAILABLE' WHEN ACTUAL.GRPPTF = PSPS.PSP_NUMBER AND
   ACTUAL.PTF_GROUP_LEVEL > PSPS.PSP_LEVEL THEN 'PSP IS DOWNLEVEL -  ' CONCAT ACTUAL.PTF_GROUP_STATUS END
   PTF_GROUP_CURRENCY, COALESCE(PSPS.PSP_NUMBER, ACTUAL.GRPPTF) PTF_GROUP_ID, COALESCE(PSPS.PSP_TITLE,
   ACTUAL.PTF_GROUP_DESCRIPTION) PTF_GROUP_TITLE, ACTUAL.PTF_GROUP_LEVEL PTF_GROUP_LEVEL_INSTALLED,
   PSPS.PSP_LEVEL PTF_GROUP_LEVEL_AVAILABLE, PSPS.PSP_DATE AS PTF_GROUP_LAST_UPDATED_BY_IBM,
   COALESCE(PSPS.PSP_RELEASE, ACTUAL.PTF_GROUP_TARGET_RELEASE) PTF_GROUP_RELEASE, ACTUAL.PTF_GROUP_STATUS
   PTF_GROUP_STATUS_ON_SYSTEM FROM
   XMLTABLE('/all_psps/psp' PASSING XMLPARSE(DOCUMENT
   HTTPGETBLOB('http://www-912.ibm.com/s_dir/sline003.nsf/PSPbyNumL.xml?OpenView&count=500' , ''))
   COLUMNS PSP_RELEASE CHAR(5) PATH 'release', PSP_NUMBER CHAR(7) PATH 'number', PSP_TITLE VARCHAR(1000) PATH
   'title', PSP_LEVEL INTEGER PATH 'level', PSP_DATE CHAR(10) PATH 'date' ) PSPS RIGHT OUTER JOIN
   ( SELECT SUBSTR(PTF_GROUP_NAME, 1,7) AS GRPPTF, PTF_GROUP_LEVEL, PTF_GROUP_STATUS, PTF_GROUP_DESCRIPTION,
   PTF_GROUP_TARGET_RELEASE FROM     ( SELECT PTF_GROUP_NAME,PTF_GROUP_LEVEL, PTF_GROUP_STATUS,
   PTF_GROUP_DESCRIPTION, PTF_GROUP_TARGET_RELEASE,
   RANK() OVER (PARTITION BY PTF_GROUP_NAME ORDER BY PTF_GROUP_LEVEL DESC) AS INSTALLED_NUMBER
   FROM QSYS2/GRPPTFINFO WHERE PTF_GROUP_STATUS = 'INSTALLED') A  WHERE A.INSTALLED_NUMBER = 1 ) ACTUAL ON
   (ACTUAL.GRPPTF = PSPS.PSP_NUMBER)
```

SQL Source

Automated management of Plan Cache detail

More procedures added to the roster…

- QSYS2/IMPORT_PLAN_CACHE() procedure
- QSYS2/REMOVE_PLAN_CACHE() procedure
- QSYS2/IMPORT_EVENT_MONITOR() procedure
- QSYS2/REMOVE_EVENT_MONITOR() procedure

Example usage:

```
CREATE OR REPLACE PROCEDURE SNAP_AND_IMPORT()
LANGUAGE SQL
BEGIN
DECLARE SNAP_NAME CHAR(10);
DECLARE OLDEST_SNAP_NAME CHAR(10);
DECLARE SNAP_COMMENT VARCHAR(100);

SET SNAP_NAME = 'SNAP' CONCAT DAYOFYEAR(current date) CONCAT
   SUBSTR(YEAR(current date),3,2);
SET OLDEST_SNAP_NAME = 'SNAP' CONCAT DAYOFYEAR(current date - 60 days) CONCAT
   SUBSTR(YEAR(current date - 60 days),3,2);
CALL QSYS2.DUMP_PLAN_CACHE_topN('SNAPSHOTS', SNAP_NAME, 100);
CALL QSYS2.IMPORT_PC_SNAPSHOT('SNAPSHOTS', SNAP_NAME, 'Top 100 Queries-'
   CONCAT CHAR(CURRENT DATE));
CALL QSYS2.REMOVE_PC_SNAPSHOT('SNAPSHOTS', OLDEST_SNAP_NAME);
END;
```

Automated Life Cycle

- **Capture**
- **import**
- **Removal**

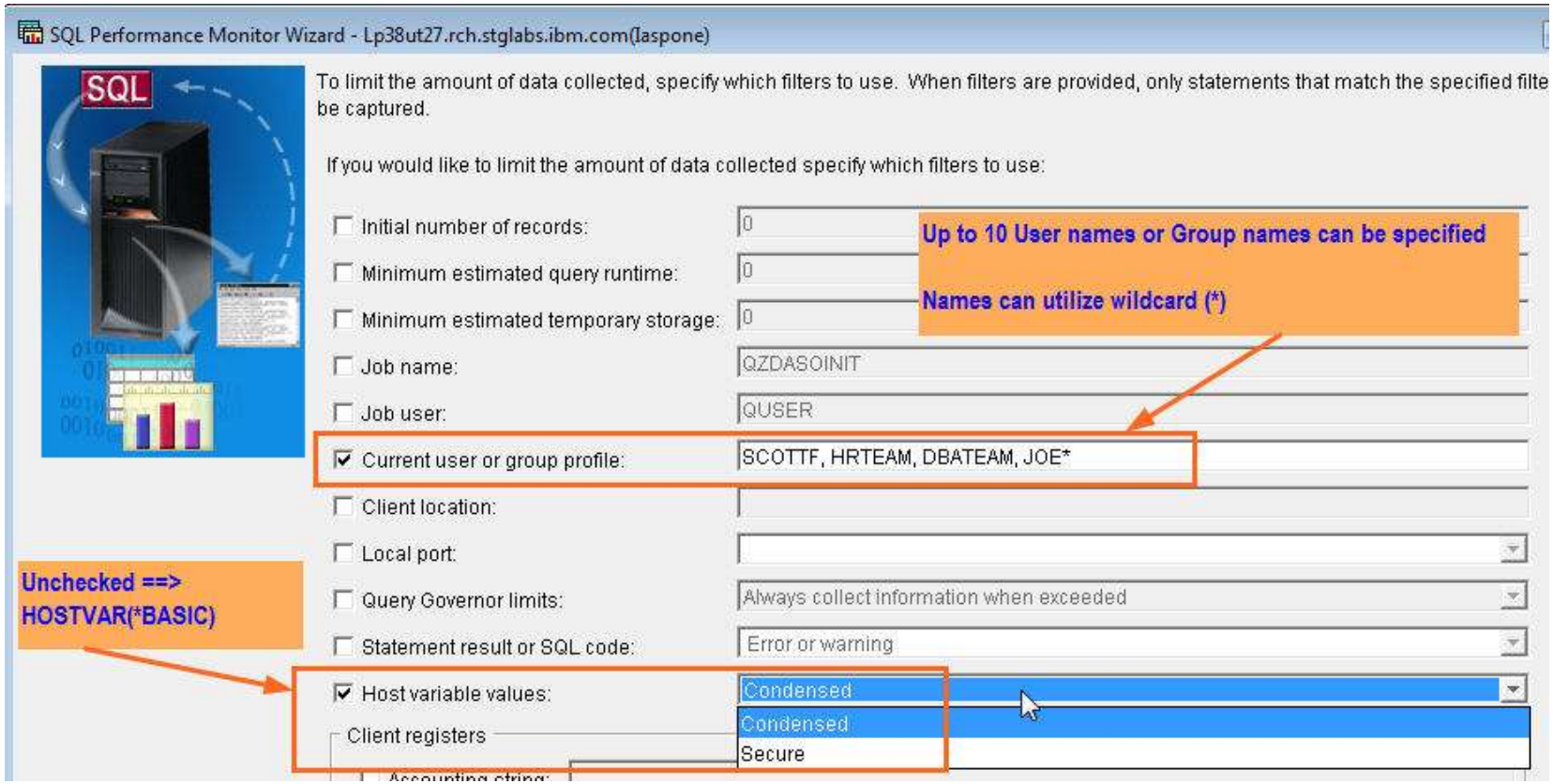
Navigator Enhancements

Navigator – what database users need to know

What are the choices?	**iAccess for Windows** (aka System i Navigator)	**IBM Navigator for i**
Where does it run?	**Windows PC Install**	**Browser** Served from IBM i 6.1, 7.1 & 7.2
Recent service level?	IBM i Access Windows Service Pack 7.1 – SI53584 ➔ TR8	IBM HTTP SERVER FOR i PTF Group: 7.2 - SF99713 Level 3 ➔ TR8 7.1 - SF99368 Level 29 ➔ TR8 6.1 - SF99115 Level 40
Database commonality	Most features are identical, including IBM i TR8 and IBM i 7.2 enhancements	Most features are identical, including TR8 enhancements
Database differences	Run SQL Scripts Visual explain	PDI Perspectives OmniFind administration
Webpage to watch	www-03.ibm.com/systems/power/software/i/access/windows_sp.html	www-912.ibm.com/s_dir/SLINE003.NSF/PTFbyNumber/SF99368 www-912.ibm.com/s_dir/SLINE003.NSF/PTFbyNumber/SF99115 www-912.ibm.com/s_dir/SLINE003.NSF/PTFbyNumber/SF99713
Next (planned) Update	December 31, 2014 ➔ IBM i 7.1 TR9 & IBM i 7.2 TR1	December 31, 2014 ➔ IBM i 7.1 TR9 & IBM i 7.2 TR1

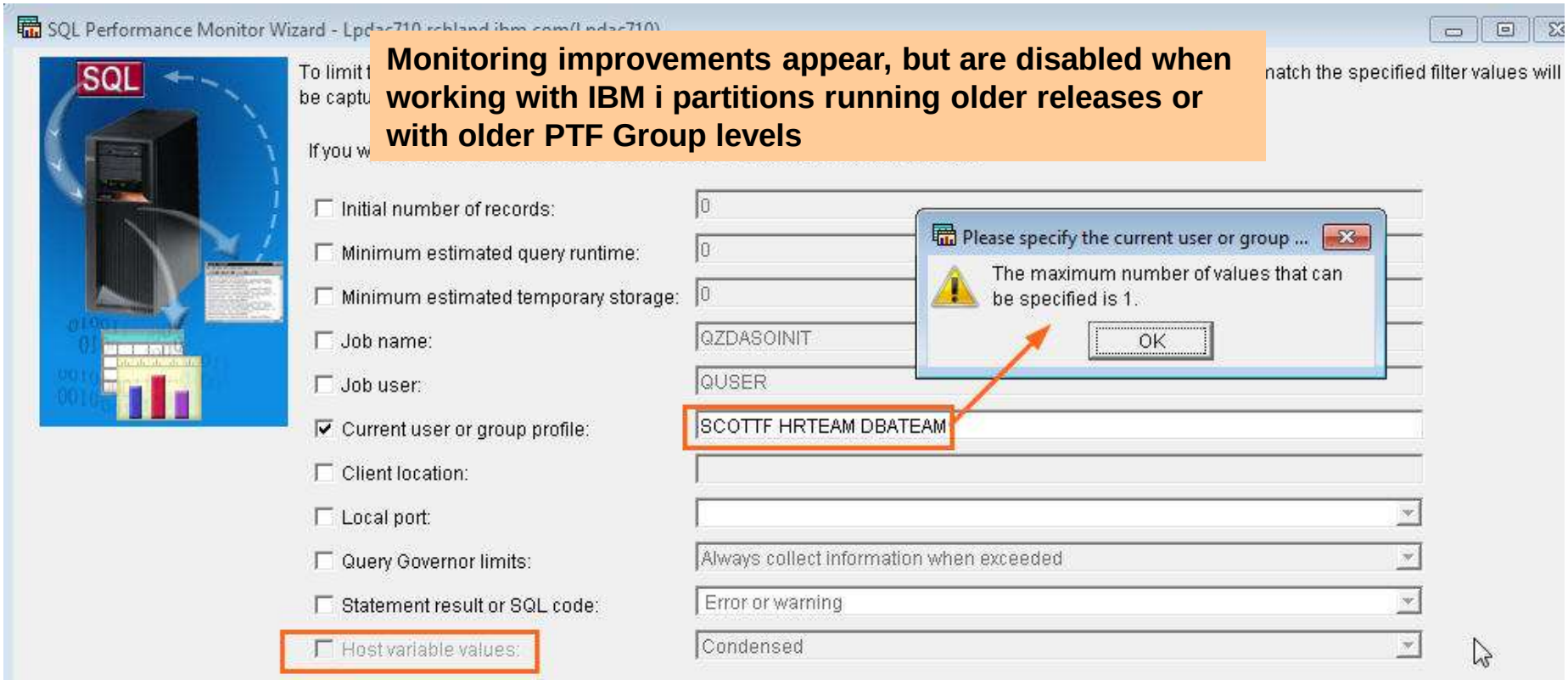
Navigator → SQL Performance Monitors

- SQL Performance Monitors
  - New HOSTVAR (*BASIC / *CONDENSED / *SECURE) control
  - Enhanced 'Filter by User' control, support for 1-10 user or group names



Navigator → SQL Performance Monitors

- Enhancement requires both Navigator update and DB2 PTF Group.
- If you have an up level client and down level IBM i, you will see the following…

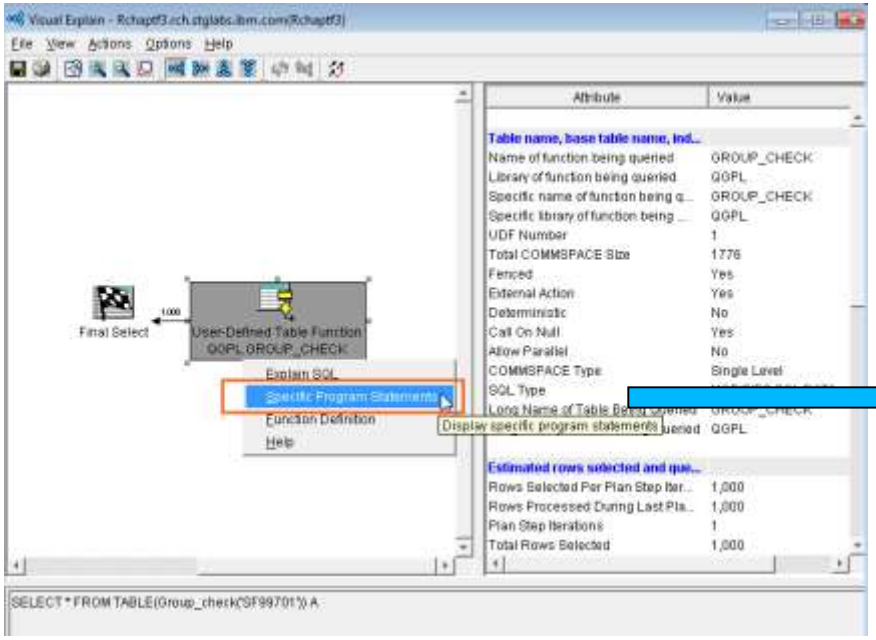


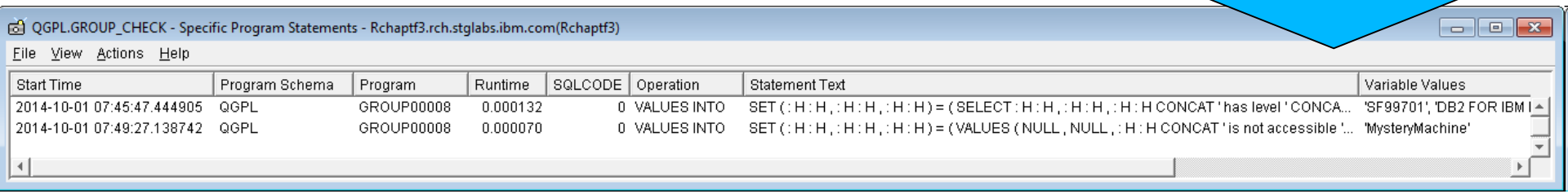
Power Systems

Navigator → Visual Explain of UDTFs

Visual Explain is enhanced to allow you to “drill down” into activity within UDTFs

- Works against Performance Monitors, Snapshots or the live SQL Plan Cache
- Returns statements executed within the UDTF, after the start time of the VE’d query

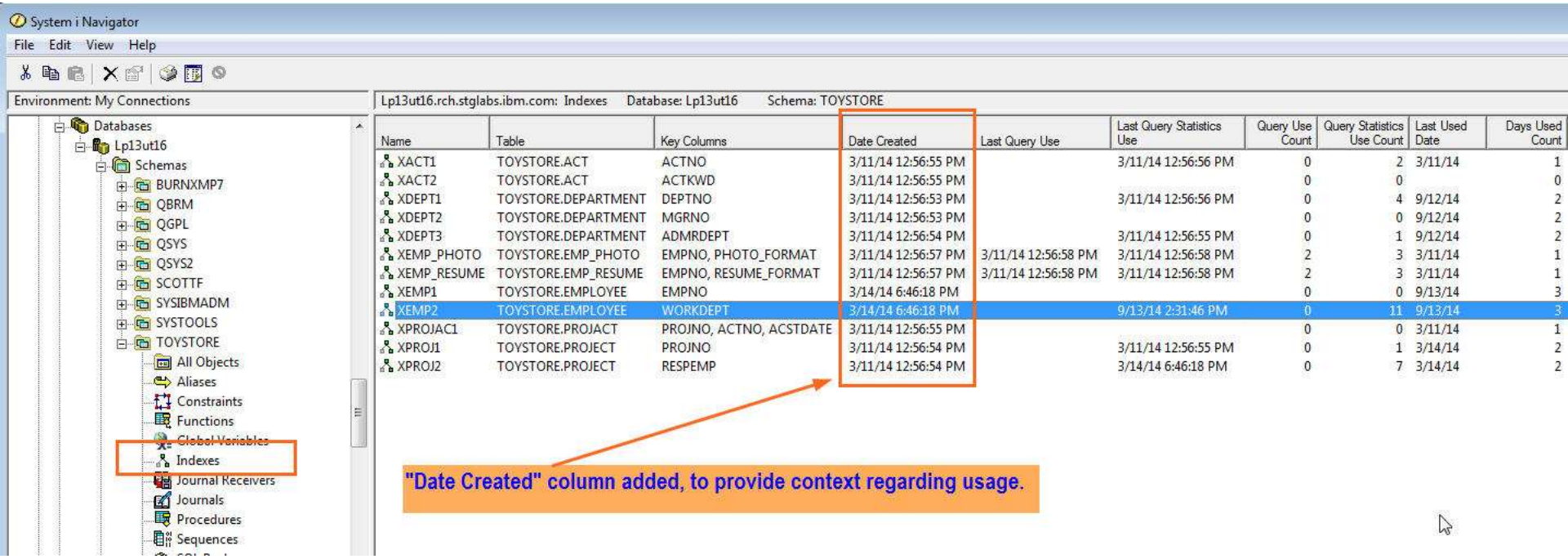




Navigator → Schemas Folder and Indexes

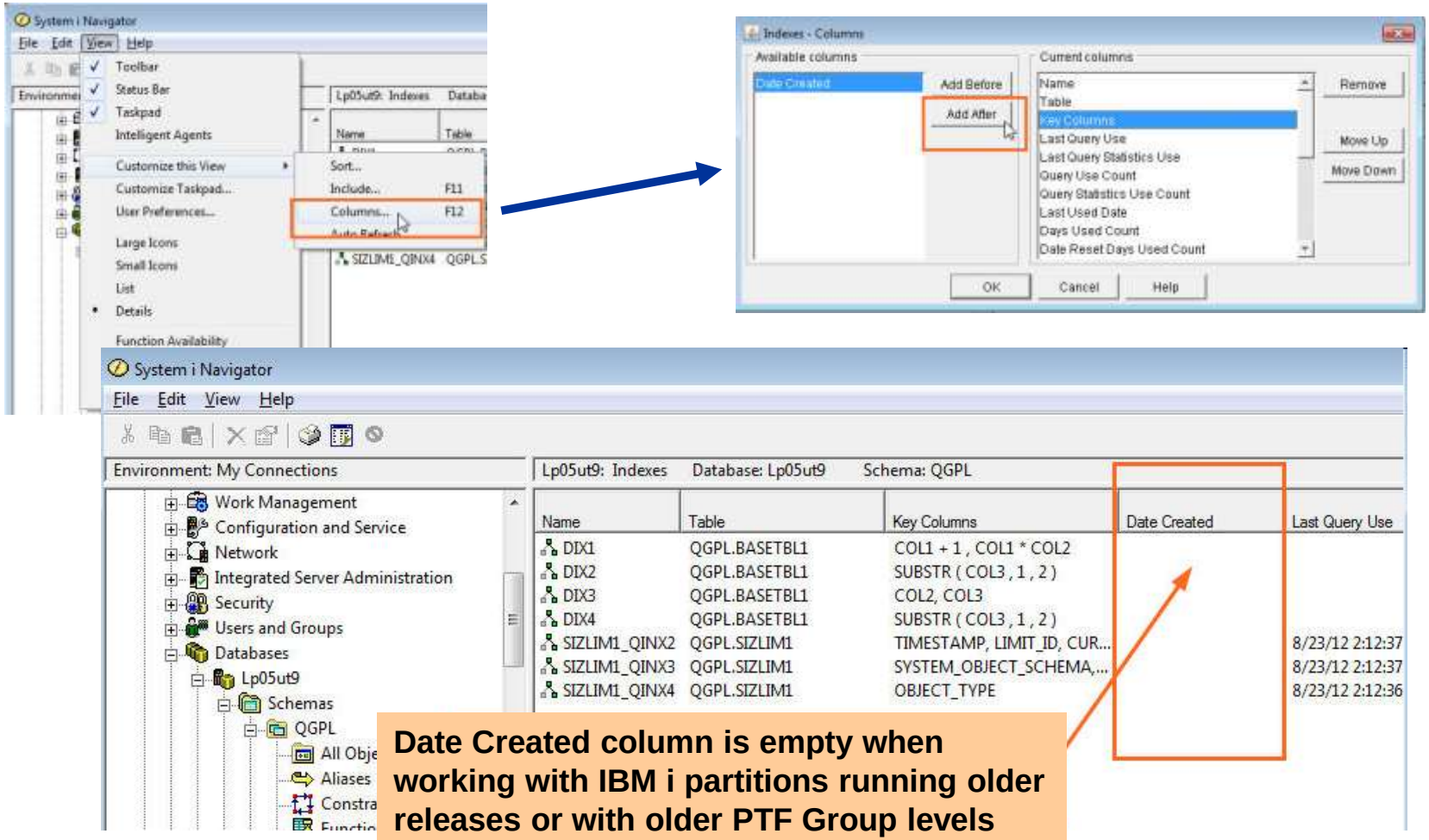
Working with existing indexes via Navigator includes usage detail, making it possible to quickly gauge the value of an existing index.

- The addition of the 'Date Created' column provides more context for the usage statistics



Navigator → Schemas Folder and Indexes

If you have ever customized this dialog, you have to use the Columns… control to add the new column

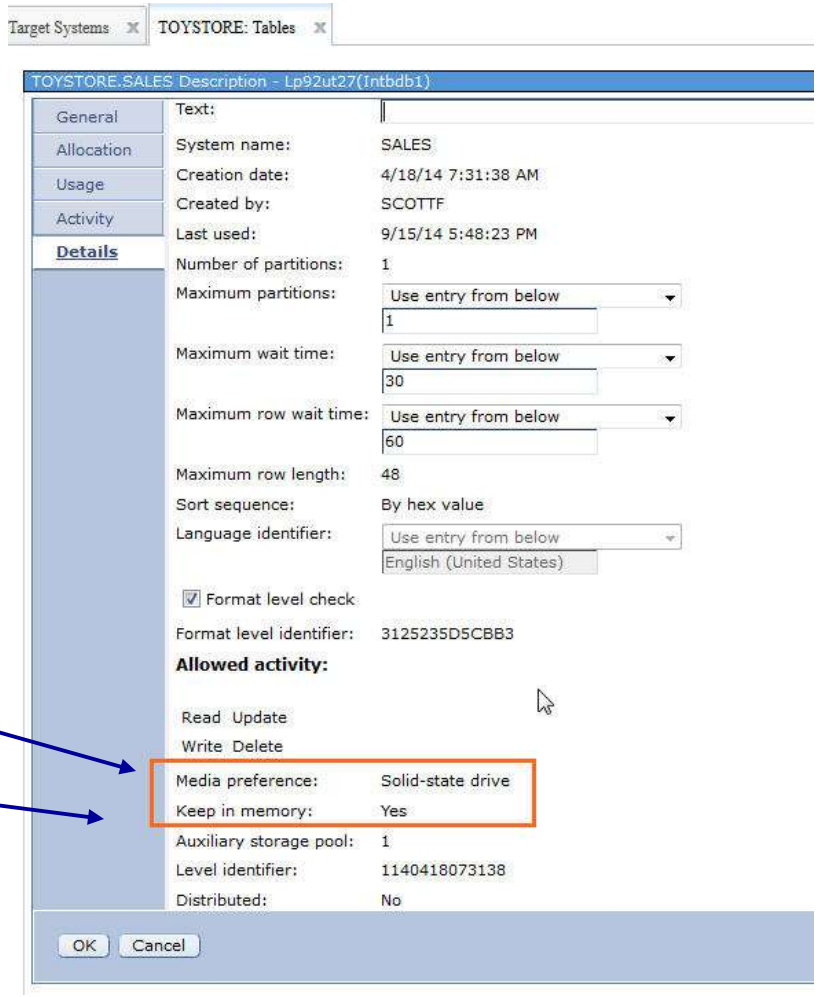


Navigator → Schemas Folder and Object Definition

Whether the object is a table, index, or partition, Navigator can be used to assess the Media Preference & Memory Preference.

ON SSD

IN MEMORY

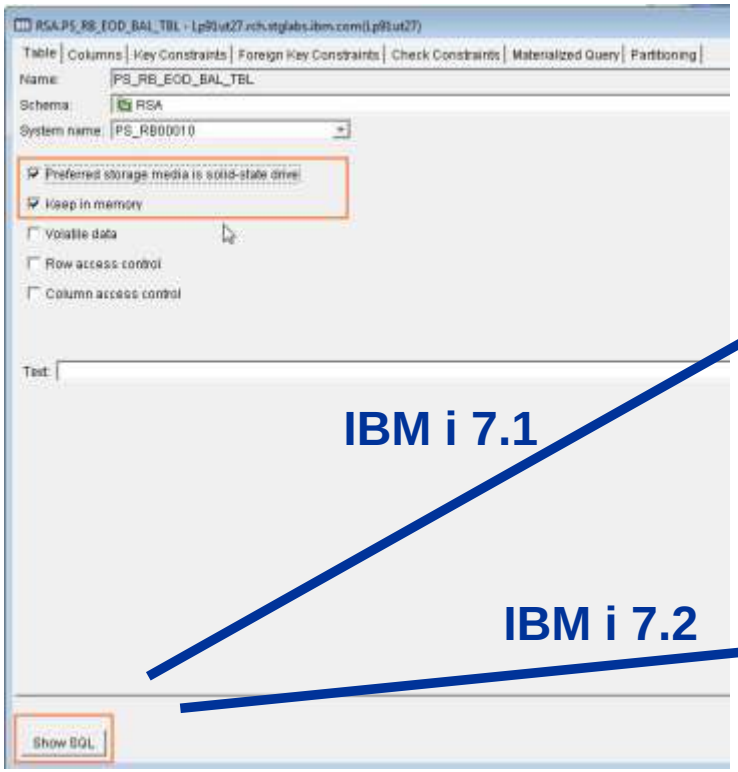


Navigator → Schemas Folder and Object Description

Use the Description action to see the Media and Memory Preference

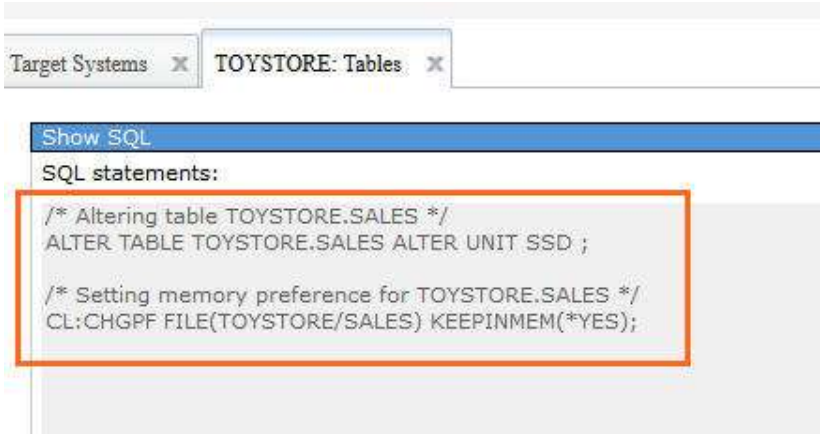
You can change the settings and either:

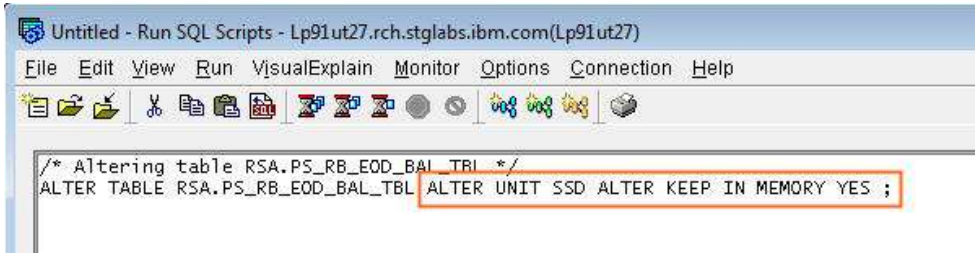
- Complete the change by selecting ‘Ok’
- Select the ‘Show SQL’ button to see the equivalent SQL



IBM i 7.1

IBM i 7.2



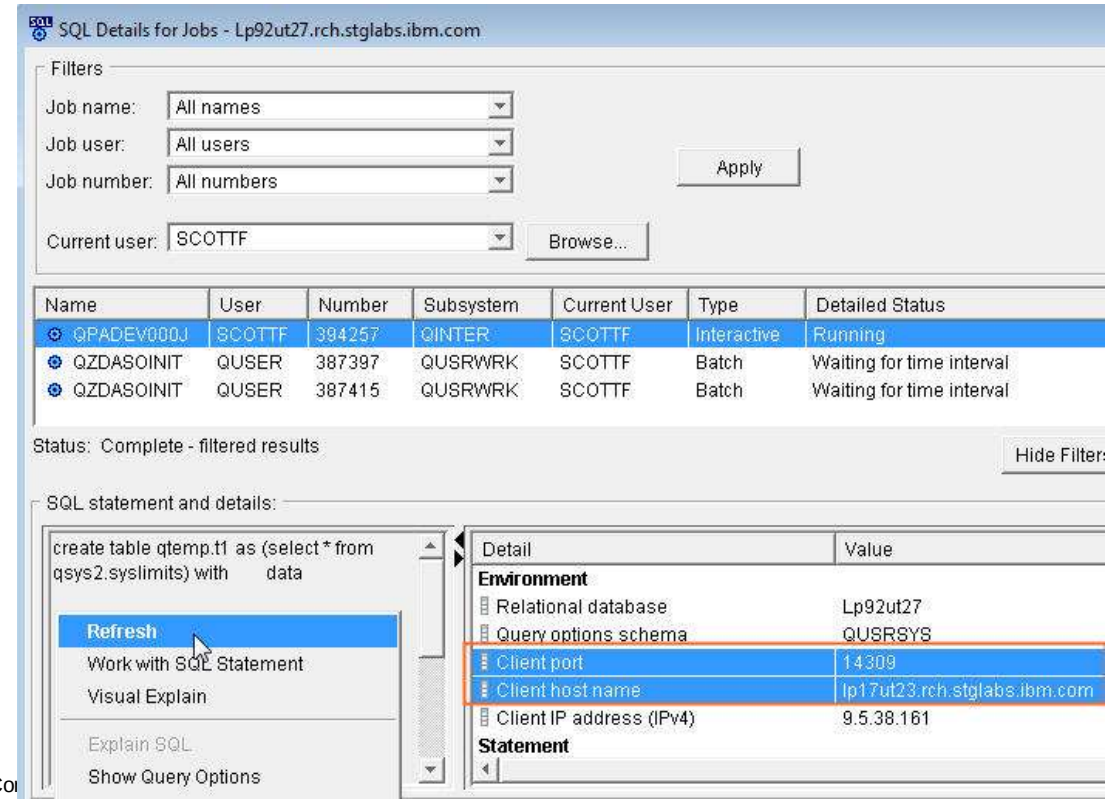


Navigator → SQL Details for Jobs

SQL Details for Jobs is enhanced to show more insight into the environment of the target job.

The Client port and Client host name values match what that job would see if it used the DB2 Built-in Global variables SYSIBM.CLIENT_PORT & SYSIBM.CLIENT_HOST

Note: Right-click on the statement text window to see the Refresh and other actions



Backup

DB2 for i – Enhancements in TR9 / TR1

Navigator enhancements for database

- SQL Performance Monitors
  - New HOSTVAR (*BASIC / *CONDENSED / *SECURE) control
  - Filter by 1-10 user or group names
- Visual Explain
  - New 'Specific Program Statements' action from UDTF icons
- Schemas folder → Indexes
  - New 'Date Created' Column
- Schemas folder → Table or Index Definition
  - Full observance of Memory Preference
  - Full observance of Media Preference
- Schemas folder → Table or Index Description
  - Full control over Memory Preference
  - Full control over Media Preference
- SQL Details for Jobs → Environment Detail
  - Addition of 'Client port'
  - Addition of 'Client host name'

Improved tooling for DBEs & Performance Analysts

IBM i Navigator (Web) enhancements

- Re-factored database dialogs
- Performance improvements

Improved Web Navigator for all database users

QSYS2/JOURNAL_INFO – View

Column Name	Data Type	Length
JOURNAL_NAME	CHARACTER	10
JOURNAL_LIBRARY	CHARACTER	10
ASP_NUMBER	INTEGER	
JOURNAL_ASPGRP	CHARACTER	10
ATTACHED_JOURNAL_RECEIVER_NAME	CHARACTER	10
ATTACHED_JOURNAL_RECEIVER_LIBRARY	CHARACTER	10
MESSAGE_QUEUE	CHARACTER	10
MESSAGE_QUEUE_LIBRARY	CHARACTER	10
DELETE_RECEIVER_OPTION	VARCHAR	3
DELETE_RECEIVER_DELAY	INTEGER	
JOURNAL_TYPE	CHARACTER	10
JOURNAL_STATE	CHARACTER	10
NUMBER_REMOTE_JOURNALS	INTEGER	
REDIRECTED_RECEIVER_LIBRARY	CHARACTER	10
JOURNAL_TEXT	CHARACTER	50
MANAGE_RECEIVER_OPTION	CHARACTER	10
MANAGE_RECEIVER_DELAY	INTEGER	
REMOVE_INTERNAL_ENTRIES	VARCHAR	3
REMOVE_FIXED_LENGTH_DETAIL	VARCHAR	3
RECEIVER_MAXIMUM_SIZE	CHARACTER	10
MINIMIZE_ESD_FOR_DATA_AREAS	VARCHAR	3
MINIMIZE_ESD_FOR_FILES	VARCHAR	19
JOURNAL_CACHE	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_JOB_NAME	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_USER_NAME	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_PROGRAM_NAME	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_PROGRAM_LIBRARY	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_SYSTEM_SEQUENCE_NUMBER	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_REMOTE_ADDRESS	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_THREAD_ID	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_LOGICAL_UNIT_OF_WORK_ID	VARCHAR	3
FIXED_LENGTH_DATA_INCLUDES_TRANSACTION_ID	VARCHAR	3

1) Data common to all journals
2) Data unique to local journals
3) Data unique to remote journals

JOURNALED_OBJECT_LIMIT	CHARACTER	10
JOURNALED_OBJECTS	INTEGER	
JOURNALED_FILES	INTEGER	
JOURNALED_MEMBERS	INTEGER	
JOURNALED_DATA_AREAS	INTEGER	
JOURNALED_DATA_QUEUES	INTEGER	
JOURNALED_IFS_OBJECTS	INTEGER	
JOURNALED_ACCESS_PATHS	INTEGER	
JOURNALED_COMMITMENT_DEFINITIONS	INTEGER	
JOURNALED_LIBRARIES	INTEGER	
JOURNAL_RECOVERY_COUNT	INTEGER	
REMOTE_JOURNAL_TYPE	CHARACTER	10
JOURNAL_DELIVERY_MODE	CHARACTER	10
LOCAL_JOURNAL_NAME	CHARACTER	10
LOCAL_JOURNAL_LIBRARY	CHARACTER	10
LOCAL_JOURNAL_SYSTEM	CHARACTER	8
LOCAL_JOURNAL_ASPGRP	CHARACTER	10
SOURCE_JOURNAL_NAME	CHARACTER	10
SOURCE_JOURNAL_LIBRARY	CHARACTER	10
SOURCE_JOURNAL_SYSTEM	CHARACTER	8
SOURCE_JOURNAL_ASPGRP	CHARACTER	10
LOCAL_RECEIVER_SYSTEM	CHARACTER	8
SOURCE_RECEIVER_SYSTEM	CHARACTER	8
ACTIVATION_TIME	TIMESTAMP	6
ESTIMATED_TIME_BEHIND	INTEGER	
MAXIMUM_TIME_BEHIND	INTEGER	
MAXIMUM_BEHIND_TIMESTAMP	TIMESTAMP	6
JOURNAL_ENTRY_FILTERING	VARCHAR	3

What's New with DB2 Web Query

- New Simplified Packaging
  - Only 2 options: Express or Standard Edition
    - You probably own the rights to Express Edition already
    - Standard adds Report Distribution, APIs, MS SQLServer adapter and run time group licensing
- New Security Center
  - Role based authorizations
- Run Time Environment
  - Control library list processing at run time; create user exit (e.g., to maintain an audit log)
- Simple Dashboards (on top of already existing dashboard builder)
  - Personal Dashboards; Mobile Dashboards
- Lots More!
  - Single Sign On
  - DB2 Family Access (DB2/z, DB2 for Linux/Unix/Windows)
  - Easier integration with existing or customized apps with REST web services application extension