What's new in DB2 for i

Rob Bestgen
(bestgen@us.ibm.com)
IBM - DB2 for i Consultant

DB2 for i – Enhancements delivered via DB2 PTF Groups

TR2-timed Enhancements
- Create OR REPLACE table
- JSON - DB2 Store Technology Preview
- SQE Performance improvements
- And more...

TR3-timed Enhancements
- LIMIT and OFFSET
- Guardium V10 and other database security monitoring enhancements
- SQE Performance improvements
- More IBM i Services
- New SQL built-in functions
- Enhancements for SAP on i clients

TR4-timed Enhancements
- Inlined UDTFs
- Trigger (re)deployment
- More IBM i Services
- New DB2 built-in Global Variables
- Enhanced SQL Scalar functions
- Evaluation option for DB2 SMP & DB2 Multisystem

Enhancements in 7.3:
- Temporal Tables
- Generated columns for auditing
- New OLAP built-ins
- Raised architecture limits
- New support for partitioned tables
- More IBM i Services
- All TR-timed enhancements

www.ibm.com/developerworks/ibmi/techupdates/db2
IBM i 7.3 & DB2 for i

DB2 for i and IBM i 7.3 – Reasons to Upgrade

Major enhancements in DB2 for i deliver significant client value:

- **Temporal Tables – History of rows**
  Data-centric, easily deployed, robust SQL point-in-time capability

- **Online Analytical Processing (OLAP) built-in functions**
  Adding more analytics capabilities directly into DB2 for I

- **Generated Columns for auditing – Row level identity**
  Let DB2 maintain the who, what, & how a row came to be

- **And… all the TR-timed enhancements delivered to IBM i 7.2**
  IBM i Services, VARCHAR_FORMAT, built-in global variables, and many more
With Temporal Tables, you can answer time-based questions:

- Who was the client rep as of two years ago?
- Who were the client reps over the last five years?
- Produce an inventory report using a different point in time
DB2 for i – DB2 for i Managed History

Accessing Data
- SELECT

Current

System Managed

History

Modifying Data
- INSERT
- UPDATE
- DELETE

Modifying Data
- INSERT
- UPDATE
- DELETE (DBE Only)

Configuring a Temporal Table

ALTER TABLE employee
ADD COLUMN instance_begin
TIMESTAMP(12) NOT NULL GENERATED ALWAYS AS ROW BEGIN
ADD COLUMN instance_end
TIMESTAMP(12) NOT NULL GENERATED ALWAYS AS ROW END
ADD COLUMN transaction_id
TIMESTAMP(12) GENERATED ALWAYS AS TRANSACTION START ID
ADD PERIOD SYSTEM_TIME (instance_begin, instance_end)

CREATE TABLE employee_history LIKE employee

ALTER TABLE employee ADD VERSIONING USE HISTORY TABLE employee_history

Establish birth/death of a row
Create history table
Enable Temporal tracking
DB2 for i and Temporal Tables

With Temporal Tables, you can:

- **Who was the client rep as of two years ago?**
  ```sql
  SELECT CLIENT_REP FROM ACCOUNTS
  FOR SYSTEM_TIME AS OF CURRENT_TIMESTAMP – 2 YEARS
  ```

- **Who were the client reps over the last five years?**
  ```sql
  SELECT CLIENT_REP FROM ACCOUNTS
  FOR SYSTEM_TIME FROM CURRENT_TIMESTAMP – 5 YEARS
  TO CURRENT_TIMESTAMP
  ```

- **Produce an inventory report using a different point in time**
  ```sql
  SET CURRENT TEMPORAL SYSTEM_TIME ’2016-03-22 17:00:00’;
  CALL GENERATE_INVENTORY_REPORT();
  ```
SQL Query Engine and OLAP Specification Extensions

On-Line Analytical Processing (OLAP) specifications provide the ability to return ranking, row numbering, and other aggregate function information as a scalar value in a query result. RANK, DENSE_RANK, and ROW_NUMBER were added in IBM i 6.1.

```sql
SELECT workdept, lastname, salary,
    RANK() OVER(PARTITION BY workdept ORDER BY salary DESC) rank,
    DENSE_RANK() OVER(PARTITION BY workdept ORDER BY salary) dense_rank,
    ROW_NUMBER() OVER(ORDER BY workdept, lastname) row_number
FROM employee ORDER BY workdept, lastname
```

<table>
<thead>
<tr>
<th>WORKDEPT</th>
<th>LASTNAME</th>
<th>SALARY</th>
<th>RANK</th>
<th>DENSE_RANK</th>
<th>ROWNBR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AOO</td>
<td>HAAS</td>
<td>52,750.00</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>AOO</td>
<td>HEMMINGER</td>
<td>46,500.00</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>AOO</td>
<td>LUCCHESI</td>
<td>46,500.00</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>AOO</td>
<td>O'CONNELL</td>
<td>29,250.00</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>AOO</td>
<td>ORLANDO</td>
<td>29,250.00</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
</tbody>
</table>

SQL Query Engine and OLAP Specification Extensions

With IBM i 7.3, DB2 for i supports the following OLAP specifications for summarizing and referencing rows within windows of result sets:

- Aggregates (SUM, COUNT, etc.)
- LAG and LEAD
- NTILE
- FIRST_VALUE, NTH_VALUE, and LAST_VALUE
- RATIO_TO_REPORT
- CUME_DIST

DB2 for i 7.3 also added support for statistical analysis which will be covered next.
Scalar Aggregate Functions

Return the detail store information and the total sales by region plus the percentage the store contributed to the total for the region:

```
SELECT store, region, sales, 
    SUM(sales) OVER(PARTITION BY region) region_total, 
    DECIMAL(100*sales / SUM(sales) OVER(PARTITION BY region), 5,2) percentage 
FROM stores ORDER BY region, percentage 
```

<table>
<thead>
<tr>
<th>STORE</th>
<th>REGION</th>
<th>SALES</th>
<th>REGION_TOTAL</th>
<th>PERCENTAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wally</td>
<td>NE</td>
<td>150,000.00</td>
<td>450,000.00</td>
<td>33.33</td>
</tr>
<tr>
<td>Pensk</td>
<td>NE</td>
<td>300,000.00</td>
<td>450,000.00</td>
<td>66.66</td>
</tr>
<tr>
<td>Bobs</td>
<td>NW</td>
<td>100,000.00</td>
<td>1,040,000.00</td>
<td>9.61</td>
</tr>
<tr>
<td>Toms</td>
<td>NW</td>
<td>440,000.00</td>
<td>1,040,000.00</td>
<td>42.30</td>
</tr>
<tr>
<td>Mills</td>
<td>NW</td>
<td>500,000.00</td>
<td>1,040,000.00</td>
<td>48.07</td>
</tr>
</tbody>
</table>

Rolling Sum Scalar Aggregate OLAP example

Return the detail store information and the rolling sum of the store sales:

```
SELECT store, region, sales, 
    SUM(sales) OVER(ORDER BY sales DESC) rolling_sum 
FROM stores ORDER BY rolling_sum 
```

<table>
<thead>
<tr>
<th>STORE</th>
<th>REGION</th>
<th>SALES</th>
<th>ROLLING_SUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caining</td>
<td>SW</td>
<td>770,000.00</td>
<td>770,000.00</td>
</tr>
<tr>
<td>Mills</td>
<td>NW</td>
<td>500,000.00</td>
<td>1,270,000.00</td>
</tr>
<tr>
<td>Toms</td>
<td>NW</td>
<td>440,000.00</td>
<td>1,710,000.00</td>
</tr>
<tr>
<td>Menes</td>
<td>SW</td>
<td>400,000.00</td>
<td>2,110,000.00</td>
</tr>
<tr>
<td>BBB</td>
<td>SE</td>
<td>350,000.00</td>
<td>2,460,000.00</td>
</tr>
</tbody>
</table>
First_Value, Last_Value, Nth_Value – OLAP Aggregate functions

Compare the sales of the current store to the store with the best sales, second best sales, and the worst sales results:

```
SELECT store, sales,
    sales - FIRST_VALUE(sales) OVER (ORDER BY sales DESC RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) behind_1st,
    sales - NTH_VALUE(sales,2) OVER (ORDER BY sales DESC RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) behind_2nd,
    sales - LAST_VALUE(sales) OVER (ORDER BY sales DESC RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) compared_to_last
FROM stores ORDER BY sales DESC
```

<table>
<thead>
<tr>
<th>STORE</th>
<th>SALES</th>
<th>BEHIND_1ST</th>
<th>BEHIND_2ND</th>
<th>COMPARED_TO_LAST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caining</td>
<td>770,000.00</td>
<td>0.00</td>
<td>270,000.00</td>
<td>670,000.00</td>
</tr>
<tr>
<td>Mills</td>
<td>500,000.00</td>
<td>-270,000.00</td>
<td>0.00</td>
<td>400,000.00</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Bobs</td>
<td>100,000.00</td>
<td>-670,000.00</td>
<td>-400,000.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Lag and Lead – Ordered OLAP specifications

Compare the sales of stores within the same region including comparisons to the stores that were adjacent in terms of better and worse sales:

```
SELECT store, region, sales,
    sales - LAG(sales,1) OVER(PARTITION BY region ORDER BY sales) AS prior_diff,
    LEAD(sales,1) OVER(PARTITION BY region ORDER BY sales) - sales AS next_diff
FROM stores ORDER BY region, sales
```

<table>
<thead>
<tr>
<th>STORE</th>
<th>REGION</th>
<th>SALES</th>
<th>PRIOR_DIFF</th>
<th>NEXT_DIFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bobs</td>
<td>NW</td>
<td>100,000.00</td>
<td>-</td>
<td>340,000.00</td>
</tr>
<tr>
<td>Toms</td>
<td>NW</td>
<td>440,000.00</td>
<td>340,000.00</td>
<td>60,000.00</td>
</tr>
<tr>
<td>Mills</td>
<td>NW</td>
<td>500,000.00</td>
<td>60,000.00</td>
<td>-</td>
</tr>
<tr>
<td>Targe</td>
<td>SW</td>
<td>140,000.00</td>
<td>-</td>
<td>260,000.00</td>
</tr>
<tr>
<td>Menes</td>
<td>SW</td>
<td>400,000.00</td>
<td>260,000.00</td>
<td>370,000.00</td>
</tr>
<tr>
<td>Caining</td>
<td>SW</td>
<td>770,000.00</td>
<td>370,000.00</td>
<td>-</td>
</tr>
</tbody>
</table>
SQL Query Engine and OLAP Specification Extensions

With IBM i 7.3, DB2 for i also supports the following OLAP specifications for statistical analysis:

- Correlation
- Covariance
- Percentile
- Median
- Linear regression (slope, intercept, etc.)


*Regression analysis* is a statistical process for estimating the relationships among variables.

Correlation, Covariance and Covariance_Samp Aggregate function

Use correlation and covariance to analyze the relationship between salary and bonus for each department:

```
SELECT workdept,
    CORRELATION(salary, bonus) correlation,
    COVARIANCE(salary, bonus) covariance,
    COVARIANCE_SAMP(salary, bonus) covariance_samp
FROM employee GROUP BY workdept ORDER BY workdept
```

<table>
<thead>
<tr>
<th>WORKDEPT</th>
<th>CORRELATION</th>
<th>COVARIANCE</th>
<th>COVARIANCE_SAMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>A00</td>
<td>0.976023</td>
<td>1,743,000</td>
<td>2,178,750</td>
</tr>
<tr>
<td>B01</td>
<td>-</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>C01</td>
<td>0.999835</td>
<td>574,437</td>
<td>765,916</td>
</tr>
<tr>
<td>D11</td>
<td>0.775424</td>
<td>240,454</td>
<td>264,500</td>
</tr>
<tr>
<td>E21</td>
<td>0.910221</td>
<td>68,944</td>
<td>82,733</td>
</tr>
</tbody>
</table>
Enhanced data-centric auditing – with autogenerated columns

• Autogenerated columns are a very powerful building block for data-centric programming in that they direct the database to automatically generate column values.

• Prior to IBM i 7.3, DB2 for i supported:
  • IDENTITY columns (which are very good for surrogate primary keys)
  • ROW CHANGE TIMESTAMP (which records the time whenever a row is changed)

• The SQL syntax GENERATED ALWAYS prevents anyone from modifying those column values, including a knowledgeable hacker.

• IBM i 7.3 includes support for additional options:
  – DATA CHANGE OPERATION (I/U/D)
  – Special register
  – Built-in Global Variable
Autogenerated columns – DATA CHANGE OPERATION

- **DATA CHANGE OPERATION** is a one character value recording the last data change:
  - I = Insert
  - U = Update
  - D = Delete

- These work well with temporal tables in that history table will provide a timeline of what changes were made and when.
  - The Delete record will be included if the temporal table was configured with the ON DELETE ADD EXTRA ROW clause.

```sql
ALTER TABLE fact_table
ADD COLUMN audit_type_change CHAR (1)
GENERATED ALWAYS AS (DATA CHANGE OPERATION)
```

Autogenerated columns – special registers

- Special registers can be used to record information about the user making the change and/or the application environment.

- Client registers can be set by the application to provide additional application information.
  - **CURRENT SERVER** contains the currently connected server.
  - **SESSION_USER** and **USER** contain the user profile currently in use which identifies who is making a change to the database.

```sql
ALTER TABLE fact_table
ADD COLUMN audit_app_client_userid VARCHAR(255)
GENERATED ALWAYS AS (CURRENT_CLIENT_USERID)
ADD COLUMN audit_user VARCHAR(128)
GENERATED ALWAYS AS (SESSION_USER)
```
Autogenerated columns – built-in global variables

- Built-in global variables are managed by the system and provide additional environmental information.

- You can use these to monitor things like which job or which IP address is being used to make a change to the database.

```
ALTER TABLE fact_table
ADD COLUMN audit_job_name VARCHAR(28)
    GENERATED ALWAYS AS (QSYS2.JOB_NAME)
ADD COLUMN audit_client_IP VARCHAR(128)
    GENERATED ALWAYS AS (SYSIBM.CLIENT_IPADDR)
```
Authority Collection

Problem: How do you lock down authority without breaking an application?

IBM Solution: Build a utility that captures pertinent authority data as apps run
- Included as part of the base OS (v7r3)
- Collection covers all native IBM i file systems
- Focus on capturing only unique instances of the authority check
- Run-time performance, while the collection is active, will degrade 2-3%
- Storage consideration for long running authority collection
- Make collection easily accessible from SQL (DB2)

• The collection includes key pieces of information including:
  – “What authority is required for this authority check”

How to Use Authority Collection

Trace security enforcement for a user

Four actions:
1. Start (STRAUTCOL)
2. End (ENDAUTCOL)
3. Display (Access for Web or QSYS2.AUTHORITY_COLLECTION)
4. Delete (DLTAUTCOL)

Reduce data collection using filters:
Users
Libraries
Objects
Object Types
File System Objects
Exclude OS Programs, internal objects, etc…
Details Collected

- Object name
- Library name
- ASP device
- Object type
- SQL name
- SQL object type
- SQL schema name
- Path name and object name
- Authorization list for the object
- Required authority
- Current authority
- Authority source for the user that satisfies the authority request
- Adopted authority indicator (adopt was used to satisfy the authority request)
- Current adopted authority
- Adopted authority source
- Adopting program name and indicator (adopting program that was used to satisfy the authority request)
- Adopting program library
- Adopting program object type (*PGM or *SRVPGM)
- Adopting program owner
- Stack info (most recent invocation and most recent user state invocation including procedure name and statement)
- Job name
- Job user
- Job number
- Current job user profile
- Group profile and indicator (group profile that was used to satisfy the authority request)
- Date and time of authority check

Authority Collection – Interrogate the collected data

Leverage Navigator (Web) for basic data investigation
Authority Collection – Use DB2 to interrogate the collected data

Determine whether excess authority exists for a user:

```
SELECT SYSTEM_OBJECT_NAME, DETAILED_REQUIRED_AUTHORITY, DETAILED_CURRENT_AUTHORITY FROM QSYS2.AUTHORITY_COLLECTION A
WHERE AUTHORIZATION_NAME = 'SCOTTF' AND SYSTEM_OBJECT_SCHEMA = 'TOYSTORE' GROUP BY SYSTEM_OBJECT_NAME, DETAILED_REQUIRED_AUTHORITY, DETAILED_CURRENT_AUTHORITY ORDER BY 1,2,3;
```

DB2 for i & Priced Options
Try before you buy!

DB2 Symmetric Multiprocessing – Option 26
DB2 Multisystem – Option 27

The IBM Lab Services DB2 for IBM i team has the ability to allow you to evaluate either of these options for up to 70 days, for no charge.

This is a simpler, no strings attached, way to evaluate these valuable database options.

Available for all IBM i 7.x releases

Interested?

Contact
Rob Bestgen (bestgen@us.ibm.com) or
Scott Forstie (forstie@us.ibm.com)

DB2 for i

Limits & Scalability
ALTER TABLE ATTACH and DETACH Partitions

ALTER TABLE DETACH PARTITION allows for the efficient roll-out of a partition that is no longer needed to be kept online.

- **ALTER TABLE DROP PARTITION** – Delete the data
- **ALTER TABLE DETACH PARTITION** – Retain the data, in a new single partition table

```
ALTER TABLE orders DETACH PARTITION p2011 INTO Archived_OrdersTable
```

Raised architectural limits in IBM i 7.3

- **Maximum # of objects in a Library**
  - 360,000 → 1,000,000
- **Maximum # of parameters in a DB2 procedure**
  - 1,024 → 2,000
- **Maximum # of parameters in a DB2 function**
  - 90 → 2,000
- **Maximum # of return columns from a DB2 table function**
  - 1024 → 8,000
- **Maximum # of members referenced in an SQL view**
  - 256 → 1,000
Ease of use

Prior to IBM i 7.3:
DLTLIB TOystore5
or
DROP SCHEMA TOystore5

With IBM i 7.3:
DROP SCHEMA TOystore5 CASCADE

CASCADE indicates that no inquiry message(s) should be sent

IBM i 7.2 TR4 & DB2 for i
(included in DB2 for i 7.3 of course)
Inlined SQL Table Functions (UDTFs)

- Scalar SQL functions have inlining support, but it is restrictive.
- SQL Table functions (UDTF) processing has been enhanced to allow SQL UDTFs with a single RETURN statement to be eligible for inlining by SQE.
  - Must be NO EXTERNAL ACTION
  - Referenced objects e.g. tables, must exist at create function time
- When a function is inlined, the invoking query is combined with the query included on the function's RETURN statement. The query engine just runs this single, composite query.

```
CREATE OR REPLACE FUNCTION LICCHK(EXPIRATION_DATE_TO_CHECK DATE)
RETURNS TABLE(RTN_PRODUCT_ID VARCHAR(7), RTN_LICENSE_TERM VARCHAR(6),
RTN_RELEASE_LEVEL VARCHAR(6), RTN_LICENSE_EXPIRATION DATE,
RTN_PRODUCT_TEXT VARGRAPHIC(50) CCSID 1200)
LANGUAGE SQL
NO EXTERNAL ACTION
NOT DETERMINISTIC
RETURN SELECT PRODUCT_ID, LICENSE_TERM, RELEASE_LEVEL, PRODUCT_TEXT,
LICENSE_EXPIRATION FROM QSYS2.LICENSE_INFO
WHERE LICENSE_EXPIRATION <= EXPIRATION_DATE_TO_CHECK;
```

No Lock Option

**Challenge:** Data access blocks changes to the file
**Response:** Add ALLOW_DDL_WHILE_OPEN QAQQINI control
**Benefit:** Ability to deploy file changes without quiescing activity
**Support:** Applies to CREATE TRIGGER, ALTER TRIGGER, DROP TRIGGER, COMMENT ON TRIGGER, and LABEL ON TRIGGER, ADDPFTRG, RMVPFTRG, and CHGPFTTRG
DB2 for i – Application Development

**SQL enhancements**
- New DB2 Built-in Global Variables
  - QSYS2.PROCESS_ID
  - QSYS2.THREAD_ID

**Query enhancements**
- Enhanced Scalar functions
  - TIMESTAMP_FORMAT
    - Many new format elements
  - VARCHAR_FORMAT
    - Numeric to Varchar
    - Character to Varchar
    - Timestamp to Varchar - new format elements
  - TRUNCATE
    - Make 2nd argument optional, default is to remove fractional element
  - ROUND
    - Make 2nd argument optional, default is to remove fractional element

---

### DB2 for i Built-in Global Variables

- Use these variables to deploy advanced logic in triggers, RCAC rules, logging, …

<table>
<thead>
<tr>
<th>Variable name</th>
<th>Schema</th>
<th>Data Type</th>
<th>Size</th>
<th>SF99702 Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCESS_ID</td>
<td>QSYS2</td>
<td>INTEGER</td>
<td>-</td>
<td>Level 11</td>
</tr>
<tr>
<td>THREAD_ID</td>
<td>QSYS2</td>
<td>BIGINT</td>
<td>-</td>
<td>Level 11</td>
</tr>
<tr>
<td>JOB_NAME</td>
<td>QSYS2</td>
<td>VARCHAR</td>
<td>28</td>
<td>Level 3</td>
</tr>
<tr>
<td>SERVER_MODE_JOB_NAME</td>
<td>QSYS2</td>
<td>VARCHAR</td>
<td>28</td>
<td>Level 3</td>
</tr>
<tr>
<td>CLIENT_IPADDR</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>128</td>
<td>7.2 base</td>
</tr>
<tr>
<td>CLIENT_HOST</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>255</td>
<td>7.2 base</td>
</tr>
<tr>
<td>CLIENT_PORT</td>
<td>SYSIBM</td>
<td>INTEGER</td>
<td>-</td>
<td>7.2 base</td>
</tr>
<tr>
<td>ROUTINE_SCHEMA</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>128</td>
<td>7.2 base</td>
</tr>
<tr>
<td>ROUTINE_SPECIFIC_NAME</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>128</td>
<td>7.2 base</td>
</tr>
<tr>
<td>ROUTINE_TYPE</td>
<td>SYSIBM</td>
<td>CHAR</td>
<td>1</td>
<td>7.2 base</td>
</tr>
<tr>
<td>PACKAGE_NAME</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>128</td>
<td>7.2 base</td>
</tr>
<tr>
<td>PACKAGE_SCHEMA</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>128</td>
<td>7.2 base</td>
</tr>
<tr>
<td>PACKAGE_VERSION</td>
<td>SYSIBM</td>
<td>VARCHAR</td>
<td>64</td>
<td>7.2 base</td>
</tr>
</tbody>
</table>
**TIMESTAMP_FORMAT**

- Transform character data into timestamp values in one step:

```
SELECT
    TIMESTAMP_FORMAT('2457270 01-00-13', 'J SS:MI:HH24')
    ,TIMESTAMP_FORMAT('SEPTEMBER:2015:04 01-00-13', 'MONTH:YYYY:DD SS:MI:HH24')
    ,TIMESTAMP_FORMAT('September:2015:04 01-00-13', 'Month:YYYY:DD SS:MI:HH24')
    ,TIMESTAMP_FORMAT('SEP:015:04 01-00-13', 'MON:Y:DD SS:MI:HH24')
    ,TIMESTAMP_FORMAT('sep:5:04 01-00-13', 'MON:Y:DD SS:MI:HH24')
FROM SYSIBM.SYSDUMMY1;
```

**VARCHAR_FORMAT**

Simplified formatting of values including the currency symbol, group separator, and decimal point. (Denoted as L, G, and D with values based on the CPX8416 message.)

- SELECT revenue FROM annual; 13945832.01

- SELECT VARCHAR_FORMAT( revenue, 'L999G999G999D99') FROM annual; 

  $13,945,832.01  

- SELECT VARCHAR_FORMAT( revenue, '999G999G999D99MIL') FROM annual;  

  13,945,832.01 €
IBM i Services for SQL users

Services in 2014:
- JOURNAL_INFO
- LIBRARY_LIST_INFO
- REPLY_LIST_INFO
- JOBLOG_INFO
- SYSTMPSTG (7.2)
- DB2 for i built-in Global Variables (7.2)
- JOB_NAME (7.2)
- SERVER_MODE_JOB_NAME (7.2)
- SYSTOOLS.GROUP_PTF.CURRENCY

Services in 2015:
- ACTIVE_JOB_INFO
- SCHEDULED_JOB_INFO
- SERVER_SBS_ROUTING
- SET_SERVER_SBS_ROUTING
- ORA_AUTHENTICATION_ENTRY_INFO
- JVM_INFO
- SET_JVM

Services in 2Q/2015:
- DRDA_AUTHENTICATION_ENTRY_INFO
- SYSTEM_STATUS_INFO
- LICENSE_INFO
- OBJECT_LOCK_INFO
- RECORD_LOCK_INFO
- OUTPUT_QUEUE_ENTRIES
- MEDIA_LIBRARY_INFO
- NETSTAT_INFO
- NETSTAT_JOB_INFO
- NETSTAT_INTERFACE_INFO
- NETSTAT_ROUTE_INFO
- SYSTOOLS.GROUP_PTF_DETAILS

Services in 2Q/2016:
- OUTPUT_QUEUE_INFO
- ENVIRONMENT_VARIABLE_INFO
- SERVICES_INFO
- Services & SQL dependency management
- Enhanced NETSTAT services
- Enhanced SET_SERVER_SBS_ROUTING
- Enhanced System Limits
- Enhanced DISPLAY_JOURNAL
- Enhanced OBJECT_STATISTICS (7.3 only)

Services in 2Q/2015:
- MEMORY_POOL_INFO
- SYSTEM_STATUS_INFO
- OBJECT_LOCK_INFO
- RECORD_LOCK_INFO
- OUTPUT_QUEUE_ENTRIES
- MEDIA_LIBRARY_INFO
- NETSTAT_INFO
- NETSTAT_JOB_INFO
- NETSTAT_INTERFACE_INFO
- NETSTAT_ROUTE_INFO
- SYSTOOLS.GROUP_PTF_DETAILS

New IBM i Services
- QSYS2.SERVICES_INFO
- QSYS2.ENVIRONMENT_VARIABLE_INFO
- QSYS2.OUTPUT_QUEUE_INFO

Enhanced IBM i Services
- QSYS2.DISPLAY_JOURNAL – New input parameters
- QSYS2.SET_SERVER_SBS_ROUTING – Add four more servers
- QSYS2.SET_SERVER_SBS_ROUTING – Add Allow Rollover configuration control
- QSYS2.NETSTAT_INFO – New columns
- QSYS2.NETSTAT_JOB_INFO – New columns
- QSYS2.NETSTAT_INTERFACE_INFO – New columns
- QSYS2.NETSTAT_ROUTE_INFO – New columns
- QSYS2.OBJECT_STATISTICS (7.3 only) – New audit, journal, and change timestamp columns
- System Limits – Phase 4
  o New columns in QSYS2.SYSLIMITS
  o Add new object limit: MAXIMUM EXTENDED DYNAMIC PACKAGE SIZE
  o Change the increment for max # of members from 100 to 50

http://ibm.biz/DB2foriServices

New & Improved IBM i Services
Be More Productive with SQL
SERVICES_INFO – Catalog of Services!

- This catalog contains a row for every IBM i Service and DB2 for i Service
- Use this catalog to programmatically detect support details and more

```
--
-- Description: Review all the Security related IBM i Services
--
SELECT * FROM QSYS2.SERVICES_INFO
WHERE SERVICE_CATEGORY = 'SECURITY';
```

<table>
<thead>
<tr>
<th>SERVICE_SCHEMA_NAME</th>
<th>SERVICE_NAME</th>
<th>SQL_OBJECT_TYPE</th>
<th>OBJECT_TYPE_NAME</th>
<th>SYSTEM_OBJECT_NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSYS2</td>
<td>USER_INFO</td>
<td>VIEW</td>
<td>*FILE</td>
<td>USER_INFO</td>
</tr>
<tr>
<td>QSYS2</td>
<td>FUNCTION_INFO</td>
<td>VIEW</td>
<td>*FILE</td>
<td>FCN_INFO</td>
</tr>
<tr>
<td>QSYS2</td>
<td>FUNCTION_USAGE</td>
<td>VIEW</td>
<td>*FILE</td>
<td>FCN_USAGE</td>
</tr>
<tr>
<td>QSYS2</td>
<td>GROUP_PROFILE_ENTRIES</td>
<td>VIEW</td>
<td>*FILE</td>
<td>GROUPLIST</td>
</tr>
<tr>
<td>QSYS2</td>
<td>SQL_CHECK_AUTHORITY</td>
<td>SCALAR FUNCTION</td>
<td>*SRVPGM</td>
<td>SQL_C00001</td>
</tr>
<tr>
<td>QSYS2</td>
<td>SET_COLUMN_ATTRIBUTE</td>
<td>PROCEDURE</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>QSYS2</td>
<td>DRDA_AUTHENTICATION_ENTRY_INFO</td>
<td>VIEW</td>
<td>*FILE</td>
<td>DRDA_AUTHE</td>
</tr>
</tbody>
</table>
ACS – Plan for database features

iAccess for Windows 7.1 includes server-aware DB2 for i features such as RCAC, Temporal and more...

### ACS – Plan for database features

<table>
<thead>
<tr>
<th>Navigator Feature</th>
<th>IBM I ACS Version 1.1.5.6 (December 2016)</th>
<th>IBM I ACS Version 1.1.6.0 (January 2017)</th>
<th>IBM I ACS Version 1.1.7.0 (December 2017)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run SQL Scripts</td>
<td>Internal Full support</td>
<td>Add CTI, Transmitter, Statement Identifier, DB2 channel manager, and others</td>
<td>SQL Add statement identifier</td>
</tr>
<tr>
<td>Analyzer</td>
<td>Internal Full support</td>
<td>No changes needed</td>
<td>No changes needed</td>
</tr>
<tr>
<td>Graphical Debugger</td>
<td>Internal Full support</td>
<td>No changes needed</td>
<td>No changes needed</td>
</tr>
<tr>
<td>Visual Style</td>
<td>Not supported</td>
<td>Internal Full support</td>
<td>No changes needed</td>
</tr>
<tr>
<td>Show Statements</td>
<td>Not supported</td>
<td>Not supported</td>
<td>No changes needed</td>
</tr>
<tr>
<td>Compare Monitors</td>
<td>Not supported</td>
<td>Internal Full support</td>
<td>No changes needed</td>
</tr>
</tbody>
</table>

Below is a list of those features that will only exist within Navigator for i (Browser):

- Scheme Finder
- Index Advisor
- DB2 Health Center
- Database Maintenance
- Transactions
- CrossFind Text Search
- View Journal Entries

Below is a list of those database features that will only exist within IBM Access for Windows:

- Database Navigator Maps
- Task Actions - View Contents and DB2 Contents

More great features coming into ACS for the database user

http://www.ibm.com/support/docview.wss?uid=nas8N1019797
DB2 Web Query Update: Version 2.2

- DB2 Web Query (5733-WQx) enhancements just keep on coming!
  - New Version 2.2
  - No charge upgrade from previous versions (with SW Maintenance)
  - New orders will get V2.2. Upgrade from any previous DB2 Web Query version
  - Required for IBM i 7.3!!
  - Supports all IBM i 7.x releases
  - Previous versions of Web Query are not supported on IBM i 7.3
  - Note: Support for V1.1 ends in September 2016

- Expands Heterogeneous database access
  - Support for MySQL, Postgres, or generic JDBC Driver to get data from Oracle and others
    - Requires DB2 Web Query Standard Edition

- New Developer Workbench Client
  - No charge upgrade for licensed owners
  - Many new usability enhancements for working with meta data, dashboards or BI Applications

- Responsive Dashboards
  - Auto adjust dashboard views based on window size or perspective
  - Target specific browsers for testing to eliminate surprises
  - Design for MOBILE devices with new Mobile Layout canvas and enhancements


---

DB2 Web Query Update: Data Migrator ETL Extension

- DataMigrator ETL (Extract, Transform, Load) Tool
  - DB2 Web Query family product (5733-WQM)
  - Replicate data, build data warehouses, data marts, or operational data stores for analytics or other purposes
  - Data Replication supports both Journal Receivers and SQL processes
  - Data transforms, data profiling, job scheduling built in

- NEW with V2.2
  - Support for MySQL, Postgres, or generic JDBC Driver to pull data from Oracle and others
    - Requires DB2 Web Query Standard Edition

- Why?
  - Consolidate data from multiple systems/databases
  - Isolate and optimize analytics workloads from production systems
  - Cleanse and transform data for accurate, speedy, self service analytics

- Considerations:
  - Replace sunsetting Data Propagator with DataMigrator
  - Complete Data Warehouse Solution for as low as $15K (U.S. List price)
  - NOTE: NOT a replacement for HA tools
Welcome to the Waitless World

Links

- DB2 Web Query for i Marketing Website
  - ibm.biz/db2webqueryi
- DB2 Web Query for i Wiki
  - ibm.co/db2wqwiki
- DB2 Web Query Getting Started Enablement
  - https://ibm.biz/db2wqconsulting
- DataMigrator Info
  - https://ibm.biz/DB2WQDATAMIGRATOR
- Video Demonstrations
  - Wizards: https://ibm.biz/DB2WQWizards
  - End User: https://ibm.biz/db2wqreportingdemos
  - Getting Started: https://ibm.biz/db2wqgettingstarteddemos

- Or reach out to the DB2 Web Query team at QU2@us.ibm.com

DB2 for i & Programmer Resources
DB2 for i – SQL Programming Resources

Essential resource for SQL & DB2 for i database application development
Draft published: March 2016

www.redbooks.ibm.com/redpieces/abstracts/sg248326.html

DB2 for IBM i Resources

- DB2 for IBM i homepage: www.ibm.com/systems/power/software/i/db2

- DB2 for IBM i wiki: ibm.biz/Bd4fFb

Welcome to the home page for the DB2 for i Wiki. Here you will find a variety of information from the leading experts for DB2 for i within IBM.
DB2 for IBM i Lab Services

• Facilitated workshops covering current state, requirements, future state, possible solutions, implementation best practices, and formulation of a strategic roadmap:
  • RCAC
  • Temporal Tables
• Customized consulting workshops
  • Advanced SQL and Datacentric Programming
  • SQL Performance Best Practices, Monitoring and Tuning
• Consulting on any DB2 for i topic

For more information, contact mcain@us.ibm.com

Thank You!

www.ibm.com/developerworks/ibmi/techupdates/db2
This document was developed for IBM offerings in the United States as of the date of publication. IBM may not make these offerings available in other countries, and the information is subject to change without notice. Consult your local IBM business contact for information on the IBM offerings available in your area.

Information in this document concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquiries, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" without any warranties or guarantees either expressed or implied. All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client’s credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice. IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies.

All prices shown are IBM’s United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply. Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies. All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied. All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client’s credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice. IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies. All prices shown are IBM’s United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply. Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies. All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied. All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client’s credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice. IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies. All prices shown are IBM’s United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply. Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment. Revised September 26, 2006.
Notes on performance estimates

rPerf for AIX

rPerf (Relative Performance) is an estimate of commercial processing performance relative to other IBM UNIX systems. It is derived from an IBM analytical model which uses characteristics from IBM internal workloads, TPC and SPEC benchmarks. The rPerf model is not intended to represent any specific public benchmark results and should not be reasonably used in that way. The model simulates some of the system operations such as CPU, cache and memory. However, the model does not simulate disk or network I/O operations.

- rPerf estimates are calculated based on systems with the latest levels of AIX and other pertinent software at the time of system announcement. Actual performance will vary based on application and configuration specifics. The IBM eServer pSeries 640 is the baseline reference system and has a value of 1.0. Although rPerf may be used to approximate relative IBM UNIX commercial processing performance, actual system performance may vary and is dependent upon many factors including system hardware configuration and software design.

Note that the rPerf methodology used for the POWER6 systems is identical to that used for the POWER5 systems. Variations in incremental system performance may be observed in commercial workloads due to changes in the underlying system architecture.

All performance estimates are provided "AS IS" and no warranties or guarantees are expressed or implied by IBM. Buyers should consult other sources of information, including system benchmarks, and application sizing guides to evaluate the performance of a system they are considering buying. For additional information about rPerf, contact your local IBM office or IBM authorized reseller.

========================================================================

CPW for IBM i

Commercial Processing Workload (CPW) is a relative measure of performance of processors running the IBM i operating system. Performance in customer environments may vary. The value is based on maximum configurations. More performance information is available in the Performance Capabilities Reference at: www.ibm.com/systems/i/solutions/perfmgt/resource.html

Revised April 2, 2007