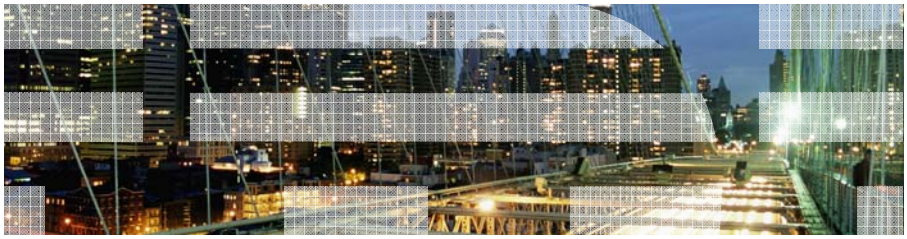
IBM Power Systems

DB2 for i 7.1
- *Hot Off the Grill*

Kent Milligan

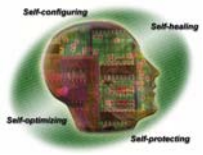
IBM STG Lab Services – DB2 for i CoE



Information Management

DB2 for i Focus Areas

- **The Self Managing Database**
  - Reduced TCO thru automation
  - Integration: Built-in Security and Auditing
- **Trusted Reliability & Scalability**
  - Simplified, best of breed scaling
  - Integrated transaction management
  - Advanced, flexible logging facilities
- **Open for Business**
  - SQL, the strategic interface
  - Latest de facto standards
- **Innovative Applications**
  - SQL & Data-centric programming
  - Move to SOA over time
- **Business Intelligence**
  - Store, manage, and ANALYZE data!
  - End user query and reporting to large scale data warehousing



DB2 for i 7.1 Enhancements

Rapid Application Development

- **SQL & RPG Integration**
- **Stored procedure Result Set consumption**
- **FIELDPROC for transparent column-level encryption**
- **XML Integration**
  - XML data type
  - Annotated XML Decomposition
  - SQL XML Publishing functions
- **Three-part Aliases**
- **Compatibility with DB2 Family & Oracle**
  - MERGE statement
  - Array support & Global Variables
  - REPLACE option on CREATEs
  - Currently Committed supported
- **JDBC & .NET enhancements**

Trusted Reliability

- **Enhanced Remote Journal filtering**
- **Library-level Journaling filtering**
- **IASP spanning transactions**

Performance & Self-Tuning Enhancements

- **SQL Query Engine (SQE) enhancements**
  - Adaptive Query Processing
  - Self-Learning Optimization
  - Inline UDF query rewrite
  - Logical File on FROM support
- **Indexing Advancements**
  - SQL Select/Omit Indexes
  - EVI Aggregates
- **CPYFRMIMPF performance**
- **SSD & In-Memory Database Enablement**
- **OmniFind Text Search Server enhancements**

Simplified Management

- **IBM i Navigator Enhancements**
  - Progress Monitors – Alter Table, Index Build
  - Index Advisor improvements
  - Enhanced Generate SQL capability
  - Object Folder content saves

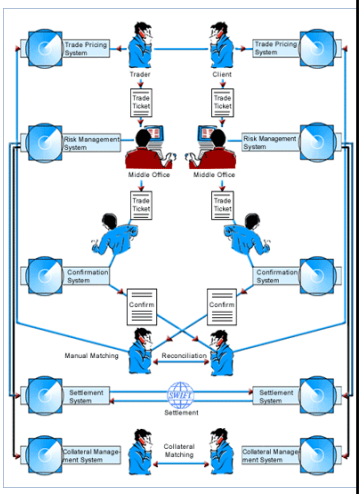
Data Intelligence & Interoperability

- **DB2 Web Query for System i**
  - Excel client support
  - Microsoft SQL Server adapter

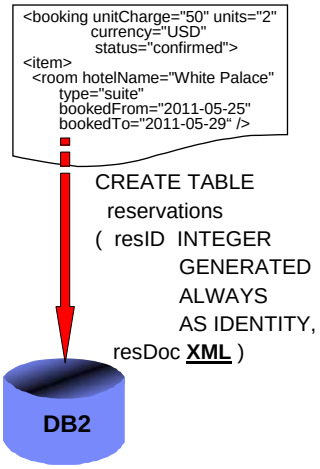
XML Integration

XML Integration with DB2

- **Rich XML Support within DB2 for i – integrated solution that replaces DB2 XML Extender product**
  - New XML data type to simplify storage and retrieval of XML documents
    - XML data access protected with rock-solid DB2 security
    - XML covered by Database Backup and Recovery processes
  - Annotated decomposition of XML documents into DB2 columns
  - Generate XML document with SQL-XML publishing functions
- **IBM OmniFind Text Search Server provides advanced, high-speed search capabilities for stored XML documents**
  - Scope searches to specific elements of an XML document: /book/title[. contains("winning")]
  - XQuery interface not yet supported



XML Data Type



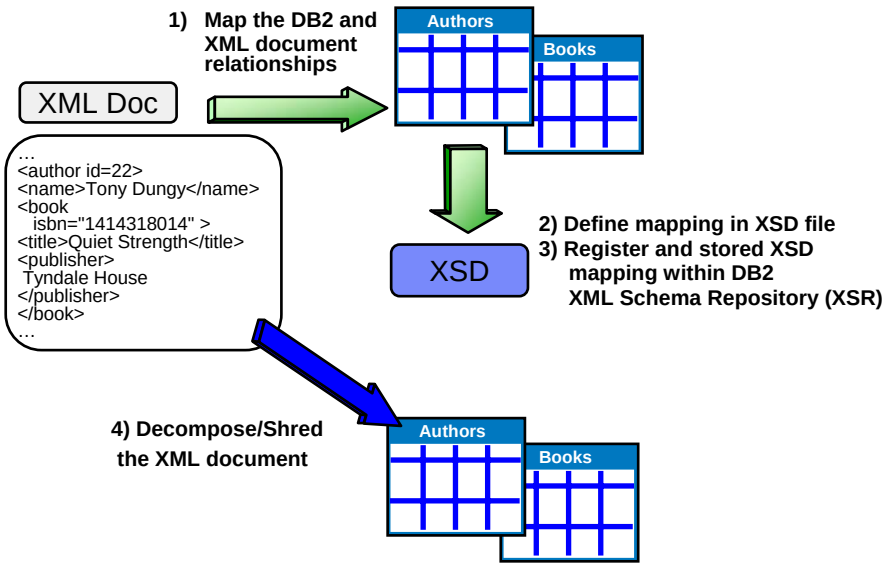
- New XML data type
  - Support XML values up to 2 GB
  - Type can be used for column, parameter, and host variable values
- XML Schema-based validation supported

```
INSERT INTO reservations(resdoc)
 VALUES(XMLVALIDATE(
        XMLPARSE(DOCUMENT
           GET_XML_FILE('/dir1/r6.xml'))
        ACCORDING TO XMLSCHEMA
        ID mylib.resschema) )
```

- XML File Reference variables enable simple export XML documents to IFS

```
D MY_XMLFILE    S       SQLTYPE(XML_CLOB_FILE)
/free
   MY_XMLFILE_NAME= '/out1.xml';
   MY_XMLFILE_NL = 9;
   MY_XMLFILE_FO = SQFCRT;
   exec sql SELECT resDOC INTO :MY_XMLFILE
            FROM reservations WHERE resID=1;
 /end-free
```

Annotated XML Document Decomposition



SQL XML Publishing Example – XMLELEMENT & XMLATTRIBUTE

- Generate XML values for employees celebrating 25th anniversary:

```
SELECT XMLSERIALIZE(
  XMLELEMENT(NAME "employee", XMLATTRIBUTES(e.empno as "id"),
  XMLELEMENT(NAME "Name", e.firstnme ||' '|| e.lastname),
  XMLELEMENT (NAME "Extension",e.phoneno),
  XMLELEMENT (NAME "DeptNo", d.deptno))  AS CLOB(100) ) as “XMLResult“
FROM employee e, department d
WHERE e.workdept = d.deptno AND
  YEAR(CURRENT DATE) –
    YEAR(hiredate) = 25
```

Output for XMLResult:

```
<employee id="000010">
  <Name>JENNA HAAS</Name>
  <Extension>0420</Extension>
  <DeptNo>A00</DeptNo>
</employee>
------------------------------
<employee id="000050">
  <Name>JOSH GEYER</Name>
  <Extension>1103</Extension>
  <DeptNo>E01</DeptNo>
</employee>
```

SQL XML Publishing Example - XMLAGG

- Generate count and XML value for parts with specified type:

```
SELECT COUNT(*) AS PartCnt,
 XMLSERIALIZE(
  XMLELEMENT(NAME "Parts", XMLATTRIBUTES(parttype AS "type"),
    XMLAGG(
       XMLELEMENT(NAME "pid", partid)  ORDER BY  partid)
  ) AS CLOB(130)) AS PartList
FROM parts WHERE parttype IN ('C01', 'E21')
GROUP BY parttype
```

PartCnt	PartList
2	<Parts type="C01"><pid>000130</pid><pid>200140</pid> </Parts>
3	<Parts type="E21"><pid>000320</pid><pid>100330</pid> <pid>200340</pid></Parts>

SQL Enhancements

Result Set Integration – Embedded SQL & SQL Routines

- Programmers can now directly integrate stored procedure result sets with embedded SQL & SQL Routines
  - Key Enabler Statements: ASSOCIATE LOCATOR & ALLOCATE CURSOR
  - Optionally, DESCRIBE PROCEDURE & DESCRIBE CURSOR statements can be used to dynamically determine the number and contents of a result set

```
...
DECLARE sprs1 RESULT_SET_LOCATOR VARYING;
CALL GetProjs(projdept);
ASSOCIATE LOCATOR (sprs1) WITH PROCEDURE GetProjs;
ALLOCATE mycur CURSOR FOR RESULT SET sprs1;
SET totstaff=0;
myloop: LOOP
  FETCH mycur INTO prname, prstaff;

  IF row_not_found=1 THEN
    LEAVE fetch_loop;
  END IF;
  SET totstaff= totstaff + prstaff;
  IF prstaff > moststaff THEN
    SET bigproj = prname;
    SET moststaff= prstaff;
  END IF;
END LOOP;
CLOSE mycur;
...
```

PROJNAME	PRSTAFF
GENERAL ADMIN ...	6.00
PAYROLL PROGR...	2.00
PERSONNEL PRO...	1.00
ACCOUNT PROGR...	2.00

Easy Integration!

Stored Procedure Enhancements

- Expressions on CALL statement

  CALL myprocedure (1, UPPER(company_name), company_discountrate*100)

- ARRAY support for SQL Routines
  - Enables exchange of data collections
  - ARRAY element limited to simple data types
  - ARRAY type can be used as parameter for SQL Routine or a local variable
  - Interfaces supporting SQL Routine ARRAY parameters:
    - JDBC
    - SQL Routines
  - Examples:

    CREATE TYPE partids AS CHAR(3) ARRAY[10];
    CREATE TYPE intarray AS INTEGER ARRAY[5];

Stored Procedure Enhancements – ARRAY Example

- Return part type and quantity for the specified collection of parts

```
CREATE OR REPLACE PROCEDURE list_parts
    (IN inparts partids, OUT part_qty intarray)
DYNAMIC RESULT SETS 1
LANGUAGE SQL
BEGIN
 DECLARE cur1 CURSOR FOR SELECT t.id,part_qty,part_type
  FROM parts, UNNEST(inparts) AS t(id) WHERE t.id = part_id;

  IF CARDINALITY( inparts )>5 THEN
    SIGNAL SQLSTATE '38003'
      SET MESSAGE_TEXT='Too many parts';
  END IF;

  SET part_qty = (SELECT ARRAY_AGG(part_qty)
                  FROM parts,UNNEST(inparts) AS t2(id)
                  WHERE t2.id = part_id);
  OPEN cur1;
END;
 ...
SET myparts = ARRAY['W12','S55','M22'];
CALL list_parts(myparts, outqty);
 ...
```

OUTPUT

ID	PART_QTY	PART_TYPE
W12	25	KSR
S55	124	KSR
M22	125	MNG

out_qty Array:
[1] = 25
[2] = 124
[3] = 125

SQL Global Variables

- Enables simpler sharing of values between SQL statements and SQL objects (Triggers, Views, etc) across the life of a job/database connection

- Example – Cache User Information

```
CREATE VARIABLE gvdept INTEGER DEFAULT
    (SELECT deptno FROM employee WHERE empuserID = USER);

CREATE VIEW filtered_employee AS (
  SELECT firstname, lastname, phoneno FROM employee WHERE deptno = gvdept);

 …

SELECT firstname, phoneno FROM filtered_employee;
```

MERGE Statement

- Allows application to use a single SQL statement to Update, Delete, or Insert into a table based on values from a source table/query
- Simplifies applications trying to merge detailed transaction data into a summary file
  - Typical processing…
    - Perform INSERT when transaction type does NOT yet exist in summary file
    - Perform UPDATE when transaction type does exist in summary file to add to the total for that type

MERGE Statement – Syntax Details

- Merge rows into the Account table. Updating the balance from the set of transactions against an account ID and Inserting new accounts from that do not already exist

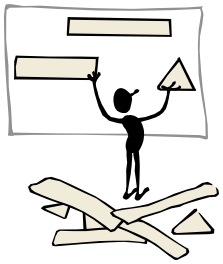
```
MERGE INTO account AS a
    USING
      (SELECT id, SUM(amount) sum_amount FROM trans GROUP BY id) AS t
    ON a.id = t.id
  WHEN MATCHED THEN UPDATE SET balance = a.balance + t.sum_amount
  WHEN NOT MATCHED THEN
          INSERT (id, balance) VALUES (t.id, t.sum_amount)
```

Additional SQL Enhancements

- REPLACE Option for CREATE statements
  - Eliminates need for the Drop statement
  - Preserves existing object dependencies & privileges!
  - Supported objects: Alias, Function, Procedure, Sequence, Trigger, Variable, View

```
CREATE OR REPLACE ALIAS myAlias FOR schema.tab1
```

- ALTER TABLE Enhancements
  - ADD BEFORE column
  - Identity Column support for existing columns
- MQ Integration Functions
  - MQSEND, MQRECEIVE,
- RPG & SQL Long Name Integration
  - New ALIAS keyword on externally described data structures enables direct usage of long SQL column names!



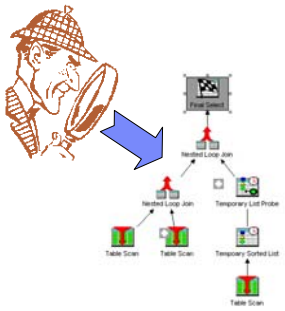
Performance Enhancements

DB2 Performance Enhancements

- SQL Query Engine (SQE) Enhancements
  - Support for Logical File on FROM clause
  - Performance advancements
    - Background Self-Learning Query Optimization
    - Adaptive Query Processing
    - Global Statistics Cache
    - Inline User-Defined Function rewrite
- SQE Indexing Advancements
  - Optimizer awareness of SQL Select/Omit Indexes
  - Encoded Vector Index Aggregate support
- Improved CPYFRMIMPF performance (6.1 & 5.4 PTFs)
- DB2 Object-level performance
  - SSD Media Preference and Random/Sequential Usage Statistics
  - OVRDBF … **REUSEDLT(*NO)** for faster Inserts/Writes
  - In-Memory Database Enablements

```
CHGPF … KEEPINMEM(*YES)
CHGLF … KEEPINMEM(*YES)
```


SQE Adaptive Query Processing



- Real-time self-learning query optimization
  - Enables query plan to be changed while query is running
  - Plan adjustments & query restart completely transparent to the application
- Intelligent monitor agents automatically assigned to each query by SQE
  - Monitoring starts after 2 seconds
  - Periodically polling measures progress against estimates and other plan assumptions
- Real-time plan adjustments can include
  - Change in join order
  - Utilization of a new index
  - …

No user interaction required!

DB2 SSD (Solid State Disks) Enablement

- SSD can improve performance for some DB2 objects
  - Large amount of random data access and…
  - Data that is read many times, but written less frequently
- DB2 interfaces enhanced to allows a user to indicate an SSD *media preference* on table, index, physical file, and logical file
  - SQL: UNIT SSD clause for object and partition
    - CREATE/ALTER TABLE
    - CREATE INDEX
  - CL: UNIT(*SSD) parameter
    - CRTPF, CRTLF, and CRTSRCPF
    - CHGPF, CHGLF, and CHGSRCPF

5.4 - Database Group SF99504 #23
6.1 - Database Group SF99601 #10

- ALTER and CHGPF/LF interfaces support asynchronous movement of data and indexes
- Key DB2 7.1 Addition - New random and sequential statistics for tables and indexes

Associated Bank

Moving DB2 tables to SSD reduced month end batch run time by **40%! ***

# of SAS Disk Drives	# of SSDs	Batch Run Time
72	0	4:22
72	8	2:43
60	4	2:48

*http://www.ibmsystemsmagpowersystemsibmidigital.com/nxtbooks/ibmsystemsmag/ibmsystems_power_200909/index.php#/16

IBM OmniFind Text Search Server Enhancements

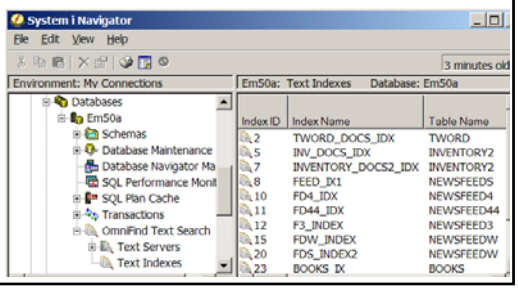
- Product (5733-OMF) originally released after GA of IBM i 6.1
  - Common DB2 Family text search support
  - Supports text columns and text documents (PDF, DOC, PPT, …)
  - High-speed, advanced linguistic searches

    CONTAINS(feedDoc, 'California insurance settlement') = 1

    CONTAINS(textFld, 'Man wins claim') = 1
- OmniFind 7.1 Enhancements
  - Enhanced XML search support
    - Date and Date Time comparisons:

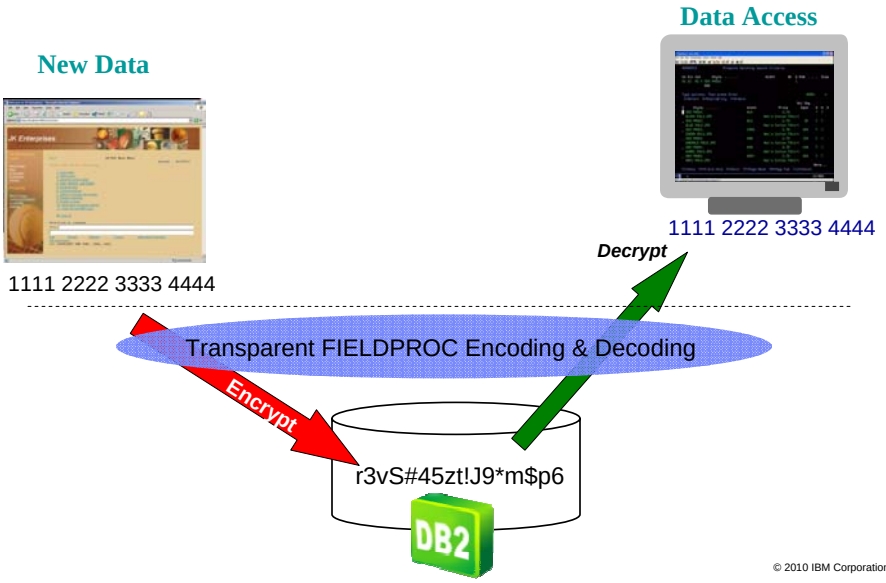
      `/Book[pubDate > xs:date("2005-04-15")]`
    - Numeric comparisons:

      `/Book[Cost <= 59.95]`
    - Namespace specific searches
  - Enhanced Save / Restore capabilities
  - Graphical text index management



Column-Level Encryption

FIELDPROC - Seamless Column-Level Encoding and Decoding



FIELDPROC Programs – What can they do?

- Developers have freedom to implement virtually any column encoding & decoding scheme
  - Encryption (3rd party solutions: Linoma Software and others)
  - Data compression
  - Text normalization
  - …
- FieldProc program requirements
  - Program must be an ILE program object & contain no SQL
  - Handle 3 different events:
    - Fieldproc registration to define encoded attributes
    - Write operations to encode data
    - Read operations to decode data

FIELDPROC Programs – When are they used?

- FieldProc Write/Encode Events
  - SQL Insert, Update, & Merge statements
  - Native record-level writes
  - Query searches: WHERE cardnumber='1111222233334444'
  - "Writing" CL Commands: CPYF, RGZPFM, STRDFU, …
  - Trigger Processing
    - Fieldproc processing occurs after BEFORE triggers
    - Fieldproc processing occurs before AFTER triggers
- FieldProc Read/Decode Events
  - SQL Select & Fetch
  - Native record-level writes
  - "Reading" CL commands: CPYF, RGZPFM, DSPPFM, …
  - Trigger processing

FIELDPROC Programs – How do they get called?

- FieldProc Registration interface - SQL !

```
CREATE TABLE ccstore(
  custid  CHAR(5),
  cardnum CHAR(16) FIELDPROC mylib/ccpgm,
  cardexp DATE )

ALTER TABLE orders ALTER COLUMN cardnum
                   SET FIELDPROC mylib/ccpgm
```

- FieldProc Removal

```
ALTER TABLE orders ALTER COLUMN cardnum
                   DROP FIELDPROC
```


FIELDPROC - Implementation Considerations

- Consider utilization of 3rd party encryption solutions for complete implementation
- Consider converting physical file DDS definitions to SQL before using SQL ALTER TABLE for Fieldproc registration
  - CHGPF can accidentally remove Fieldproc registrations!
  - Techniques exist for avoidin program recompiles after SQL conversion
- Consider pitfalls of conditional decoding (ie, masking) solutions

Additional Information

- DB2 for i Websites
  - Home Page: **ibm.com/systems/i/db2**
  - DeveloperWorks Zone: ibm.com/developerworks/db2/products/db2i5OS
  - Porting Zone: ibm.com/partnerworld/i/db2porting
- Newsgroups & Forums
  - USENET: comp.sys.ibm.as400.misc, comp.databases.ibm-db2
  - DeveloperWorks: https://www.ibm.com/developerworks/forums/forum.jspa?forumID=292
  - System i Network DB2 Forum: http://forums.systeminetwork.com/isnetforums/
- Education Resources - Classroom & Online
  - ibm.com/systemi/db2/gettingstarted.html
  - ibm.com/partnerworld/wps/training/i5os/courses
- DB2 for i Publications
  - White Papers: ibm.com/partnerworld/wps/whitepaper/i5os
  - Online Manuals: ibm.com/systems/i/db2/books.html
  - DB2 for i Redbooks (http://ibm.com/redbooks)
    - Getting Started with DB2 Web Query for System i (SG24-7214)
    - OnDemand SQL Performance Analysis … in V5R4 (SG24-7326)
    - Preparing for and Tuning the SQL Query Engine on DB2 for i5/OS (SG24-6598)
    - Modernizing iSeries Application Data Access (SG24-6393)

➡Are you experiencing performance problems?

➡Are you using SQL?

➡Are you getting the most out DB2 for i?

➡Need help?

IBM DB2 for i Center of Excellence

- ✓ Database modernization
- ✓ DB2 WebQuery
- ✓ Database design, features and functions
- ✓ DB2 SQL performance analysis and tuning
- ✓ Data warehousing and Business Intelligence
- ✓ DB2 for i education and training

Contact: Mike Cain mcain@us.ibm.com
IBM Systems and Technology Group
Rochester, MN USA