

IBM Software Group

Break Free with DB2

IBM Information Management software

DB2 9.7 Compatibility Features

© 2009 IBM Corporation

Credits:

- Paul Bird
Senior Technical Staff Member, DB2 for LUW
- Robert Harbus
Worldwide DB2 Evangelist
- Development and Support staff of IBM Toronto Lab

Disclaimer/Trademarks

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious, and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

The following terms are trademarks or registered trademarks of other companies and have been used in at least one of the pages of the presentation:

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both: AIX, AS/400, DataJoiner, DataPropagator, DB2, DB2 Connect, DB2 Extenders, DB2 OLAP Server, DB2 Universal Database, Distributed Relational Database Architecture, DRDA, eServer, IBM, IMS, iSeries, MVS, Net.Data, OS/390, OS/400, PowerPC, pSeries, RS/6000, SQL/400, SQL/DS, Tivoli, VisualAge, VM/ESA, VSE/ESA, WebSphere, z/OS, zSeries

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

DB2 9.7 Themes

- **Resource Optimization**
 - Best performance with most efficient utilization of available resources
- **Ongoing Flexibility**
 - Allow for continuous and flexible change management
- **Service Level Confidence**
 - Expand your critical workloads confidently and cost effectively
- **XML Insight**
 - Harness the business value of XML
- **Break Free with DB2**
 - Use the database server that gives you the freedom to choose

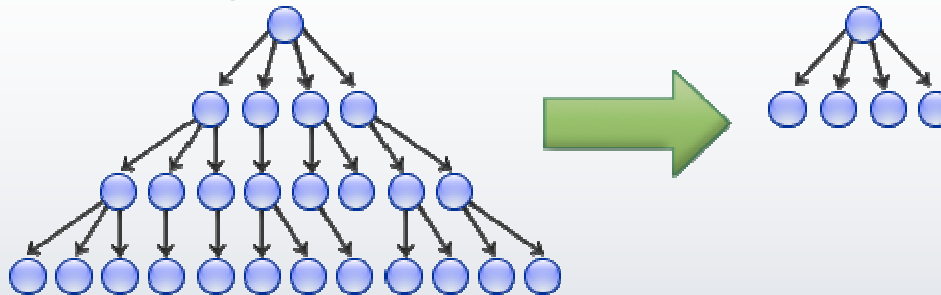
Compression Advancements: Motivation

- DB2 9 table compression provides significant reduction in storage and memory consumption for tables
 - Reductions often 80% and beyond
 - Typically accompanied by overall throughput gains
 - Big win for many customers in reducing TCO and improving overall performance

- Logical next step: apply similar advantages to other key storage/memory consumers
 - Index and Temp Data

Improvements to Compression in DB2 9.7

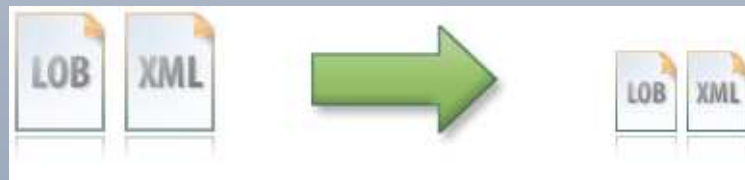
- Multiple algorithms for automatic index compression



- Automatic compression for temporary tables



- Intelligent compression of large objects and XML



Improvements to Compression: Index Compression

- Index pages stored compressed on disk and in bufferpool
- New INDEX compression attribute
 - ▶ CREATE INDEX ... **COMPRESS YES**
 - ▶ ALTER INDEX ... **COMPRESS YES**
- An INDEX will be compressed when CREATED or REORGed, if:
 - ▶ The index compress attribute is YES, or,
 - ▶ The index compress attribute is not set, and the table compression attribute is YES
- Three different compression techniques applies to index pages
 - ▶ Prefix compression
 - ▶ “RIDlist” compression
 - ▶ Slot directory compression
 - ▶ DB2 automatically chooses applicable techniques
- Target: 35%-55%

Compression Savings from SAP Supplied Tables

	Number of Index Pages	Percentage saved
Not compressed	147152	0%
Prefix Compression	102104	30.6%
RID List Compression	99766	32.2%
RID list compression + Variable Slot Directory + Byte based prefix compression	65919	55.2%

Bytes saved = 2.4G - 1.1G = 1.3G

Temporary Table Compression: Overview

- Temp tables will be compressed automatically, if DB2 installation is licensed for data compression
 - ▶ No need to turn on explicitly
- Compression technology is closely modeled after existing table compression
 - ▶ Dictionary based compression
 - ▶ Data size, and row size, thresholds govern if/when compression is used
 - Avoids overhead of creating compression dictionary when potential gains are very small
- db2pd enhancements to monitor temp compression
 - ▶ via `-temptable` option

Temporary Table Compression: Details

- Internal temp tables
 - ▶ These are temp tables automatically created by DB2 in order to run an access plan
 - 'System' temps
 - Spilling sorts
 - ▶ Eligible for compression if the optimizer estimates a large enough
 - Data size (i.e., 100 MB)
 - Row size (i.e., 20 bytes)

- Actually compressed at run-time when
 - ▶ 'System' temps : after data has been inserted to create compression dictionary (~2 MB), subsequent rows inserted into the temp will be compressed
 - ▶ Spilling sorts : dictionary created on first spill; all rows will be compressed
- Target: 40%

Temporary Table Compression: Details

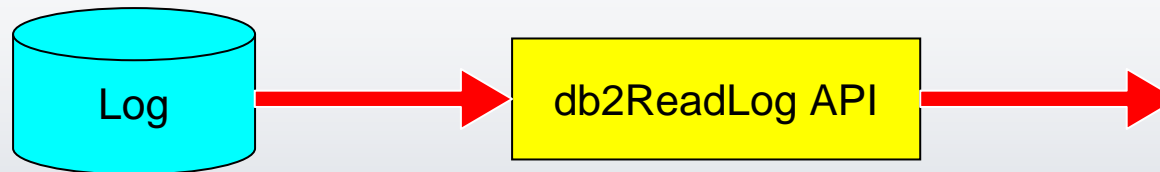
- External temp tables - These are temp tables explicitly created by a user or application
 - ▶ Declared Global Temporary Table
 - ▶ Created Global Temporary Table

- Use an algorithm very similar to automatic dictionary for regular tables
 - ▶ Internal threshold designed as best compromise between compression ratio, and dictionary build speed
 - ▶ Unlike regular tables, the compression dictionary is not stored in the table (it is only stored in memory for temp tables, since it doesn't need to be persisted)

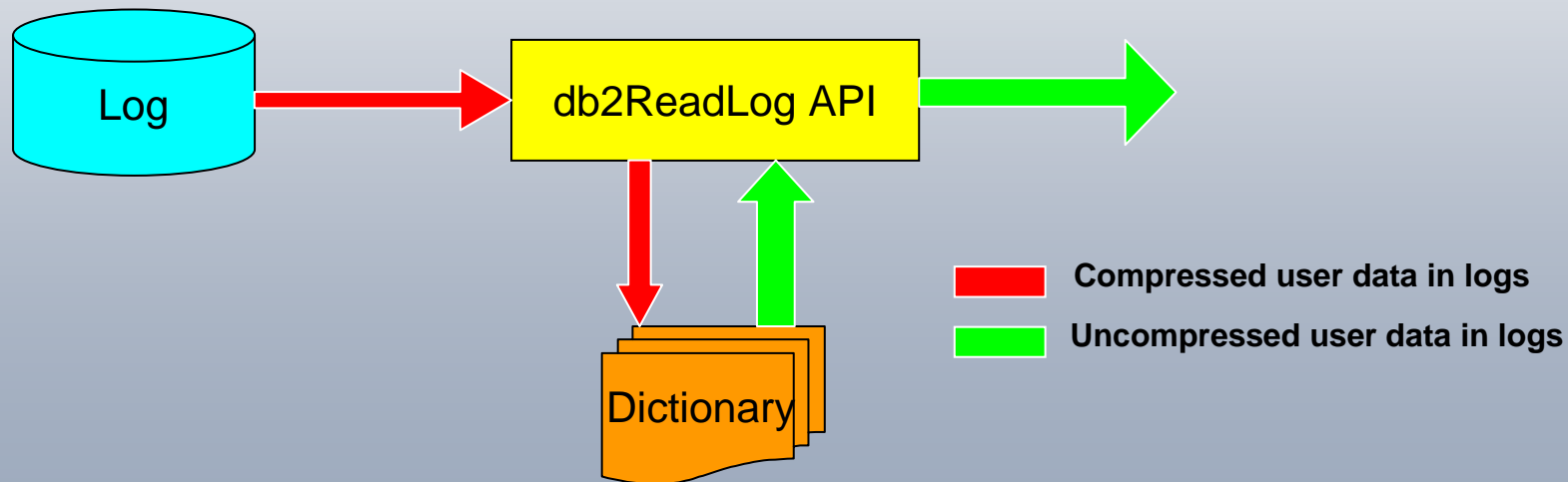
Compression and Replication

- db2ReadLog API will now decompress log data before returning log records

DB2 9.5 with iFilterOption ON

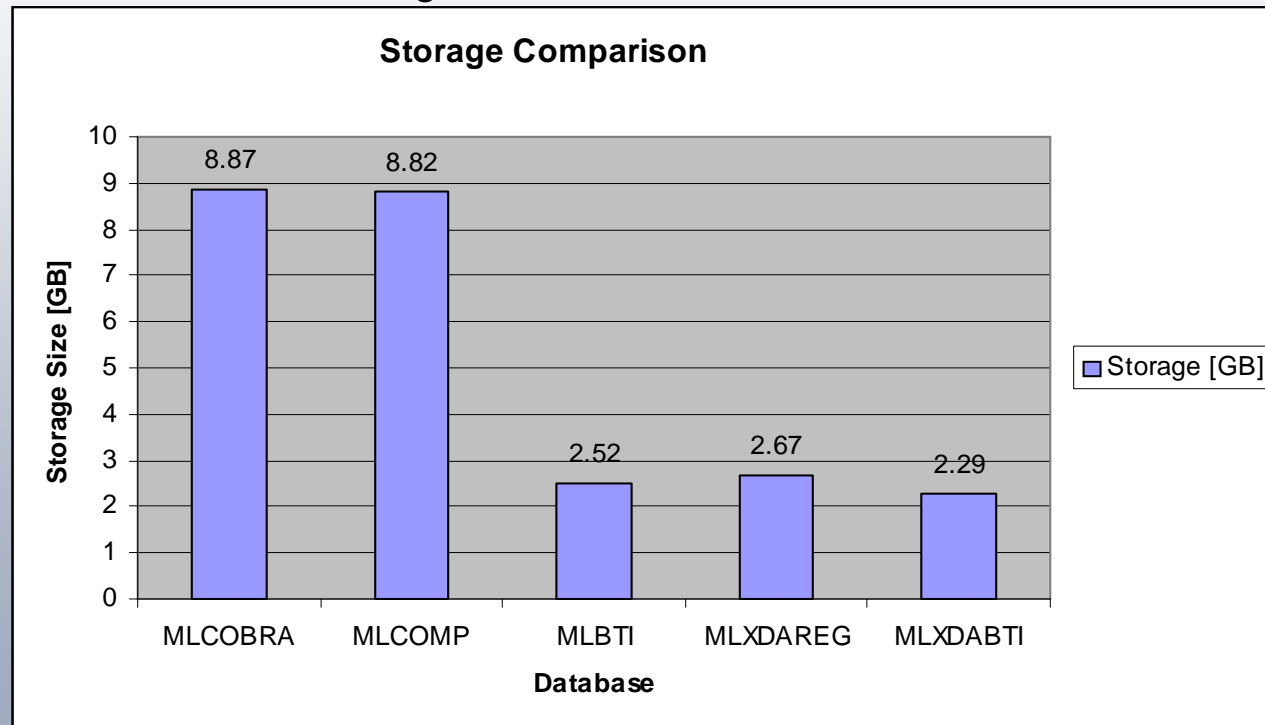


DB2 9.7 with iFilterOption ON



XDA (XML) Compression

- XML docs in XDA object can also be compressed
 - ▶ Enablement is via the table COMPRESS attribute
 - ▶ Static dictionary approach
 - ▶ Classic/'Offline' reorg table based

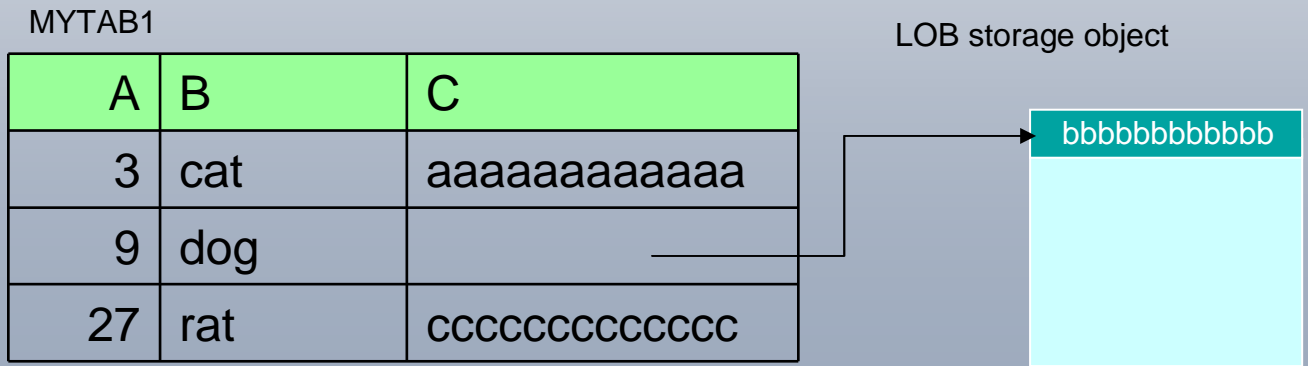


Overall Compression Ratio: 70% (Storage size reduces from 8.87GB to 2.67GB)

Large Object Inlining

- Instead of strictly storing LOB data in the LOB storage object, the LOBs, if sufficiently sized, can be stored in the formatted rows of the base table
 - ▶ Dependant on page size, the maximum length a LOB can be to qualify for inlining is 32 669 bytes
 - ▶ LOB inlining is analogous to XML inlining for XML data, introduced in V9.5
 - ▶ I/O cost is reduced since this data get buffered along with the base table data
 - ▶ XML/LOBs inlined within the base table data can be compressed

```
create table mytab1 (a int, b char(5), c clob inline length 1000)
```



XML Enhancements

- Business intelligence for XML

- ▶ XML in data partitions (DPF)
- ▶ XML in range partitions
- ▶ XML in database views

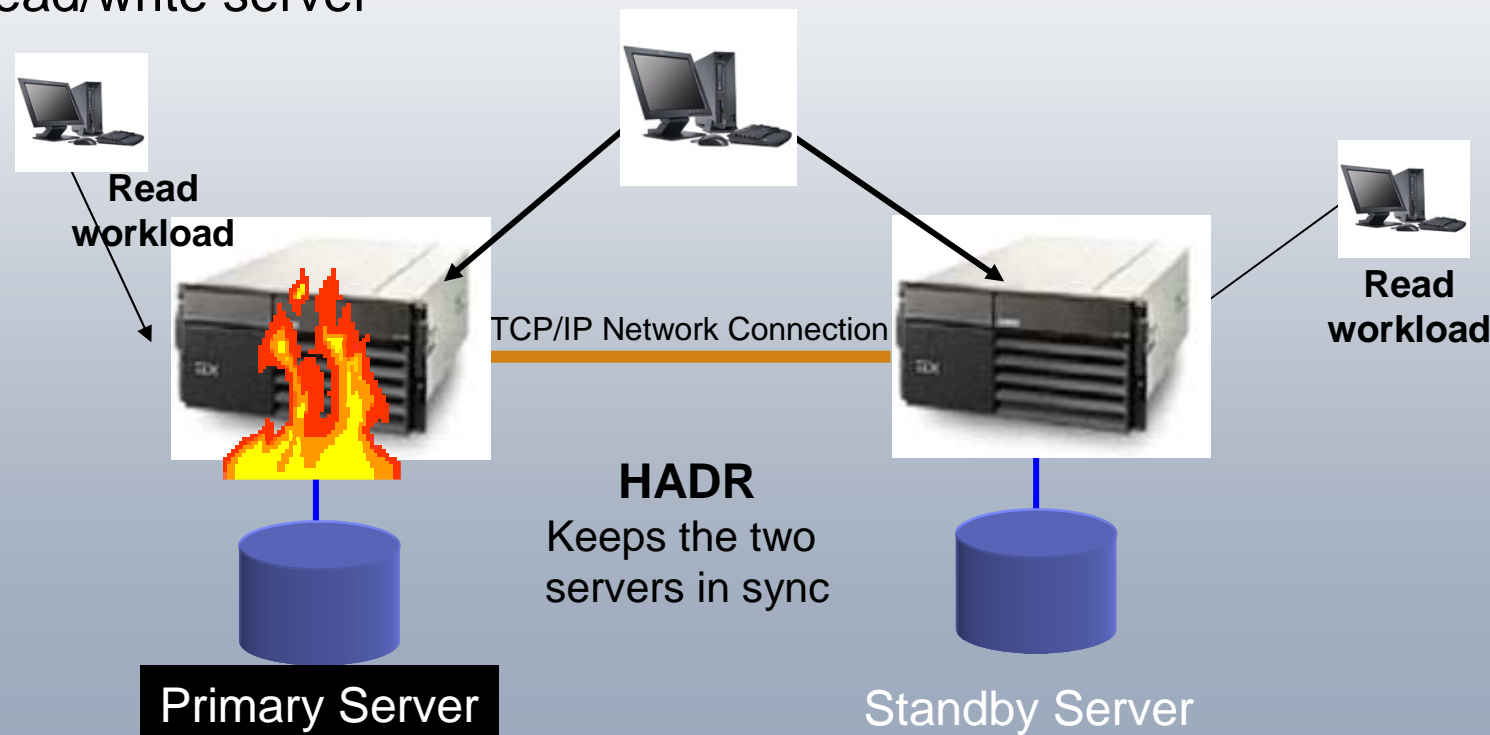
- Additional Features

- ▶ XDA Compression
- ▶ Online index reorg
- ▶ Online XML index create
- ▶ XML in UDF
- ▶ XML in MDC
- ▶ Bulk Decomposition
- ▶ XML DBA Functions
- ▶ Net.Search Extender on DPF, MDC, and Range Partitioned tables



HADR Read on Standby

- Turn your disaster recovery hardware investment from seldom used servers to a reporting server giving you more insight
- High Availability Disaster Recovery (HADR) now supports read only reporting
- During failover, DB2 seamlessly turns the read only standby into a read/write server



Optimize Performance with Workload Management

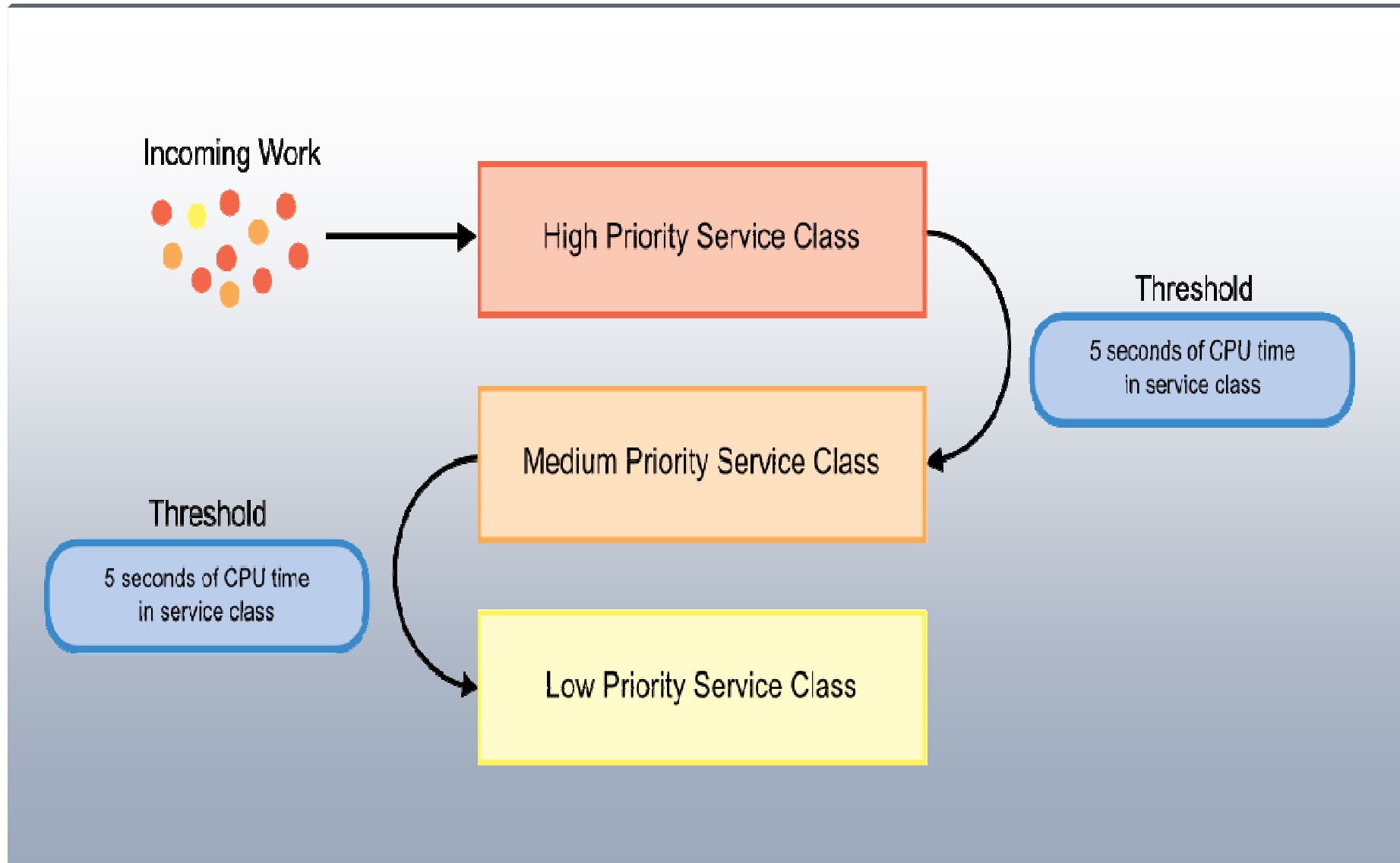
- Help ensure you meet your Service Level Agreements
 - ▶ Create controls ahead of time
 - ▶ Override them on the fly
 - ▶ Create controls that adjust to the changing workload priorities throughout the day
- Lower your costs by automating resource allocation and utilization
 - ▶ Control for both applications and users
 - ▶ Establish controls based on business priority
- Workload management
 - ▶ Part of database engine
 - ▶ Request management
 - ▶ Resource management



Improved Functionality for WLM in DB2 9.7

- Reduced the granularity for time thresholds to 1 minute (from 5)
- Introduce a new aggregate threshold, AGGSQLTEMPSPACE, to control usage of system temporary table space for a service class
- Allow the collection of aggregate activity data at workload level
- Buffer Pool I/O priority
- Linux WLM integration
- Wildcard capabilities for workload definition attributes
- New high water marks for CPU Time and Rows Read
- Introduce support for a “Tiered approach” to WLM

Tiered Approach to WLM



Schema Evolution: Challenges Today

Scenario:

- A table is decomposed into two tables
- A view (V1) must be updated accordingly:

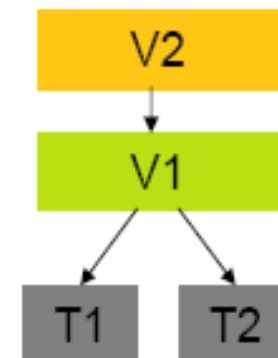
```
DROP VIEW V1 RESTRICT
```

```
CREATE VIEW V1 AS (SELECT ... FROM T1, T2 ...)
```

- Another view (V2) references V1

Problem:

- The database will prevent the drop of V1 because it would make V2 invalid, even though the second statement would make V2 valid again.



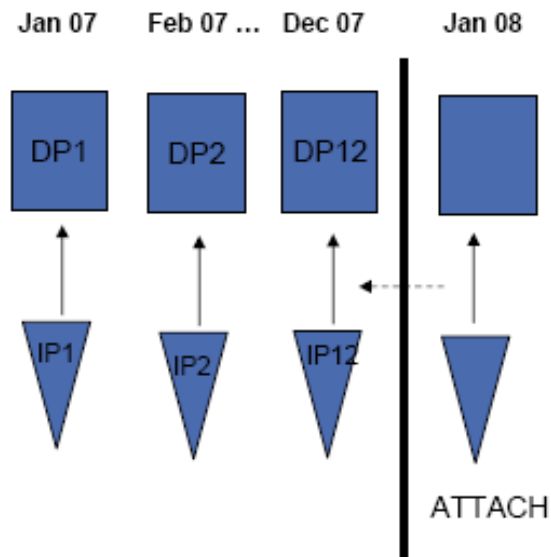
Schema Evolution: DB2 9.7

- Relax the object dependency model
 - Allow changes that affect dependent objects to succeed
 - Automatically revalidate dependent objects
 - **ALTER TABLE**, ALTER COLUMN, DROP COLUMN, RENAME COLUMN
 - **CREATE OR REPLACE** ALIAS, FUNCTION, NICKNAME, PROCEDURE, SEQUENCE, TRIGGER, VARIABLE, VIEW
 - **DROP** FUNCTION, NICKNAME, PROCEDURE, SEQUENCE, TABLE, TRIGGER, TYPE, VARIABLE, VIEW, TABLE

- Implement “missing” features:
 - RENAME COLUMN
 - Support CREATE OR REPLACE syntax for views, functions, triggers, etc.
 - Allow additional data type changes via ALTER COLUMN
 - Between any types SQL runtime can cast

Partitioned (Local) Indexes on Range (Table) Partitioning

- Will support the ability to create local indexes
- This will relieve current issues associated with Roll-In processing, mainly the global index maintenance and associated logging
- Allows for reorg table at the range partition level
 - ▶ Improved ease of use with respect to range level compression



- Ability to create local (partitioned) index
- Unique index must be superset of partition key

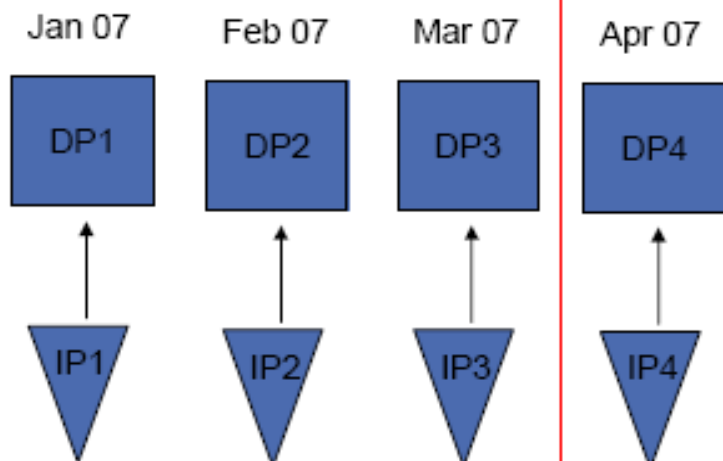
Example:

```
CREATE INDEX pINX1 on SALES  
  (sales_date, partID) PARTITIONED
```

- Partitioned index is the default
- RUNSTATS will collect at the table level as well as at partition data and index level
- SYSIBM.SYSDATAPARTITIONS is extended to include:
 - CARD, OVERFLOW, NPAGES, FPAGES, AVGROWSIZE, PCTROWSCOMPRESSED

Partitioned (Local) Indexes on Range (Table) Partitioning

Maximize availability during maintenance



- **Maintenance Concerns:**
 - How can I re-compress a partition?
 - Trickle feed can cause page split for local index?
 - How can I update partition level stats?
- **Solution**
 - Partition level reorg

Transportable Table Spaces and Table Moves

- Easily and quickly move data between QA, test, and production systems
 - ▶ Efficiently move schema between databases
 - ▶ Easy to set up multiple test environments
 - ▶ New table space can have different properties
 - ▶ Larger page size, different extent size, etc.
 - ▶ No downtime—keep tables online
 - ▶ Extract DDL and other dependent objects
 - ▶ Directly reference containers of table space in target database



Sparse MDC Tables

- Storage in an MDC table is tracked through a 'block map': which extents have data and which don't
- When a block is emptied, the storage remains with the table and is available for later reuse
 - ▶ Currently, the only Classic/'Offline' table reorg can free this storage
- New option on reorg table command to not reorg this table but reclaim these empty blocks/extents
 - ▶ This storage is freed back to the tablespace and hence is available for use by any object in the tablespace
 - ▶ During processing, this new capability allows concurrent write to the table and occurs with a minimum amount of time and resource

Tablespace Extent Remapping: Remedy for “HWM Issues”

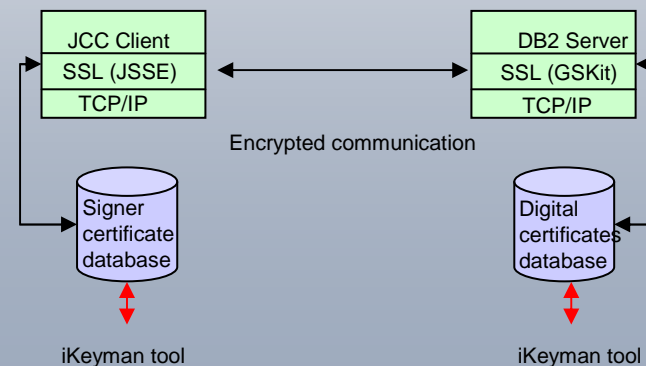
- High Water Marks will always exist
- A HWM can become an ‘issue’ when there is free space below it
 - ▶ ALTER TABLESPACE... REDUCE will lower the HWM and reclaim free space below it by transparently ‘shuffling’ the contents of the tablespace to lower positions in the tablespace
- All new DMS tablespaces will be created with this ‘new’ underlying infrastructure
- Previous (Release migrated) tablespaces will not have this capability; tables must be moved to the new version

Separation of Duties

- Extend the scope of the current SECADM authority to be able to fully manage security and be grantable to roles and groups as well
 - ▶ Remove the capability to grant and revoke the DBADM and SECADM authorities from the SYSADM authority and vest it in SECADM
 - ▶ Remove the implicit DBADM authority from the SYSADM authority
- DBADM can be defined with no capability to grant and revoke privileges or to access any table data
- New Privileges
 - ▶ WLMADM - Manages WLM objects
 - ▶ SQLADM - Responsible for SQL tuning
 - ▶ EXPLAIN - Grants the ability to explain and prepare SQL statements
 - ▶ DBADM Grants – DB2 no longer grants the extra CONNECT, LOAD, CREATETAB, BINDADD, LOAD, CREATE_EXTERNAL_ROUTINE, IMPLICIT_SCHEMA, QUIESCE_CONNECT privileges

Secure Socket Layer (SSL) Support

- DATA_ENCRYPT uses DES with 56 bit encryption keys
 - ▶ DES is no longer the industry standard encryption algorithm
 - ▶ Most auditors would require AES encryption
- SSL is the industry standard for data in-transit encryption
 - ▶ Provides AES encryption
 - ▶ Provides data integrity
- Federal Information Processing Standards (FIPS) 140-2 compliance



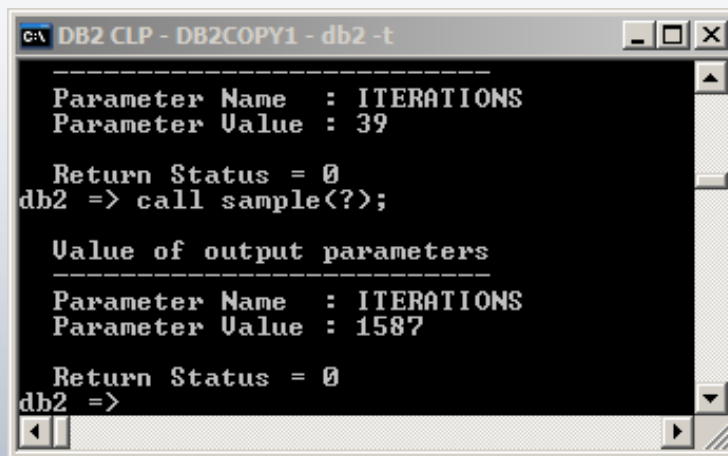
SQL Enhancements

- New and improved scalar functions
 - ▶ ADD_MONTHS, DAYNAME, DECFLOAT_FORMAT ...
- Parameterized timestamps
 - ▶ Timestamps have a precision between 0 and 12 with improved datetime arithmetic
- Public synonyms
 - ▶ Allow shared objects to be accessed by a common, unqualified name
- Implicit casting
 - ▶ Assign or compare strings to numeric types and untyped NULLs
- Created global temporary tables
- Truncate table
 - ▶ Drops all rows without logging
- Autonomous Transactions
 - ▶ Procedure defines work to be done outside the current transaction



Statement Concentrator

- Optionally replace literals with parameter markers
 - ▶ Increases section sharing and reduces compilation
 - ▶ Reduces number of statements to be compiled

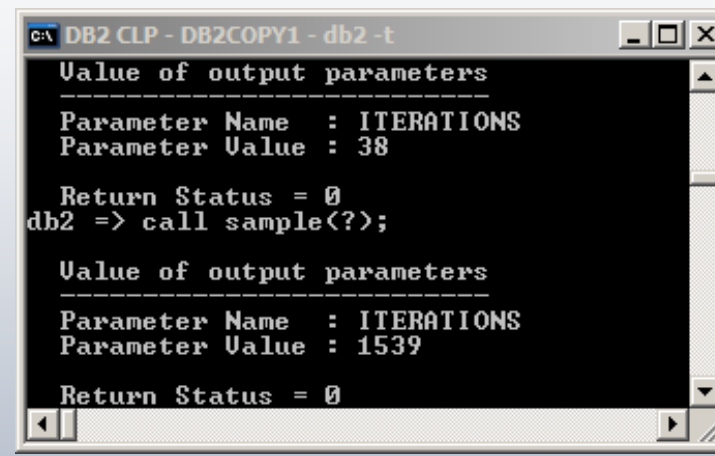


```
C:\> DB2 CLP - DB2COPY1 - db2 -t
-----
Parameter Name : ITERATIONS
Parameter Value : 39

Return Status = 0
db2 => call sample(?);

Value of output parameters
-----
Parameter Name : ITERATIONS
Parameter Value : 1587

Return Status = 0
db2 =>
```

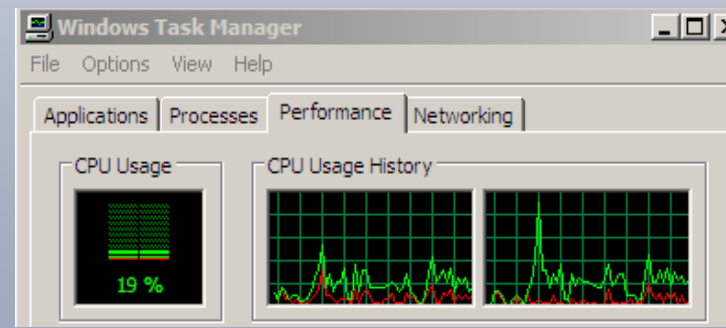
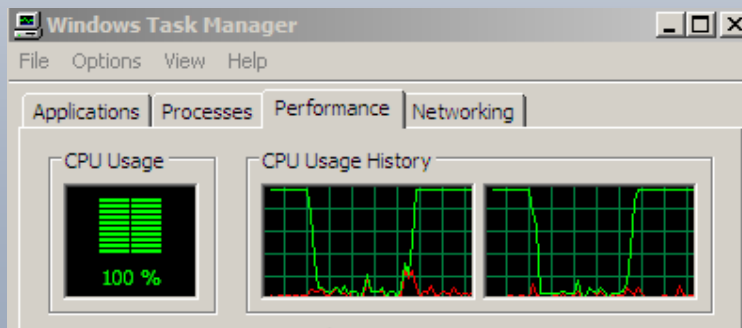


```
C:\> DB2 CLP - DB2COPY1 - db2 -t
-----
Value of output parameters
-----
Parameter Name : ITERATIONS
Parameter Value : 38

Return Status = 0
db2 => call sample(?);

Value of output parameters
-----
Parameter Name : ITERATIONS
Parameter Value : 1539

Return Status = 0
```

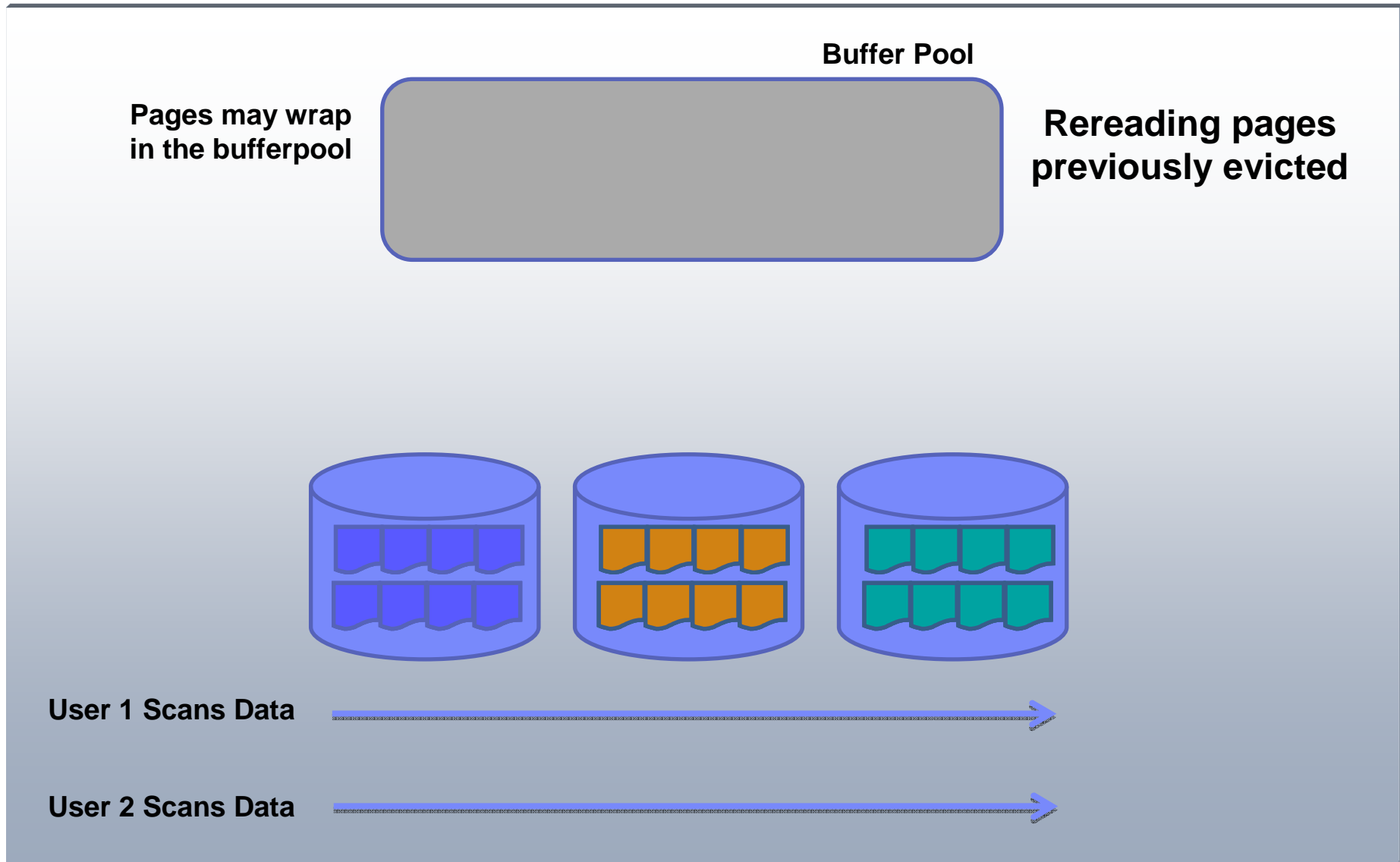


DB2 9.7 Delivers Even Faster Warehousing with Scan Sharing

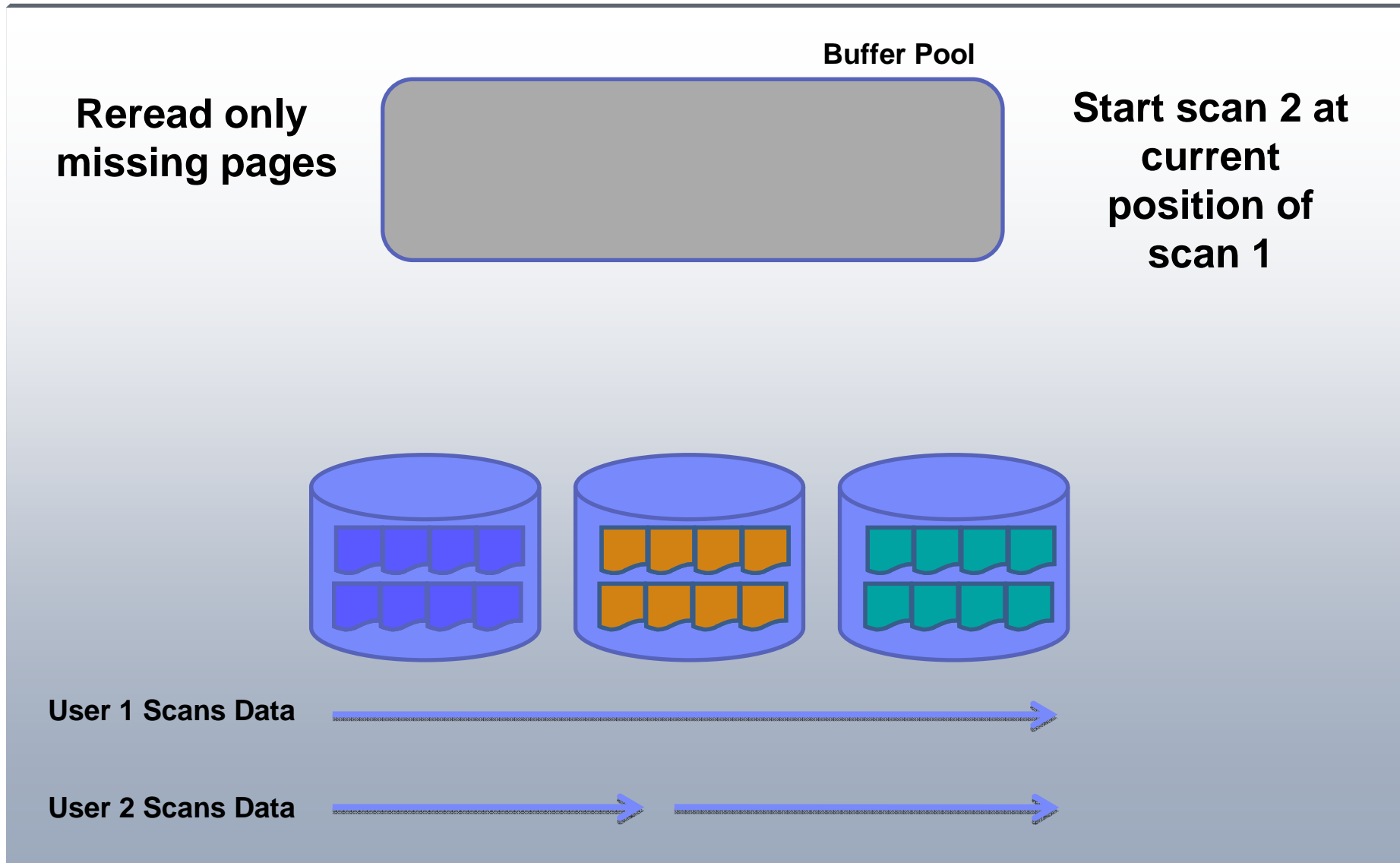
■ Scan Sharing

- ▶ Improves the performance of multi user workloads
- ▶ Queries work together reading the same pages at the same time
- ▶ New scan will start based on current scan position
- ▶ When it reaches end of file it will wrap and finish when it reaches the starting point
- ▶ All Automatic – no DBA intervention required

Multiple Scanners pre DB2 9.7



Multiple Scanners with DB2 9.7



SQL PL Enhancements in DB2 9.7

- Full SQL PL support across the board
- Modules
 - ▶ Bundle of several related objects
- Row and Collection Types
- Cursor and result-set handling
 - ▶ Pass and return cursors
- Data type anchoring
 - ▶ Keep procedural variables in sync with table
- Associative Arrays
 - ▶ Use free-formed text to index into an array
- CALL statement
 - ▶ Default values, named parameters
 - ▶ Can skip parameters with default values in CALL

	DB2 9.5	DB2 9.7
SPs	✓	✓
UDFs	✗	✓
Triggers	✗	✓
Anon. Block	✗	✓

```
DECLARE empSalary ANCHOR employee.salary;
```

```
CREATE TYPE arrType2 AS  
    INTEGER ARRAY[VARCHAR(100)];
```

```
CREATE PROCEDURE P1(x INT, y INT DEFAULT 1)  
    CALL P1(10)  
    CALL P1(10, DEFAULT)  
    CALL P(y => 100, x => 200)
```

Currently Committed Locking

- Readers don't block writers (readers avoid locking)
- Writers don't block readers (readers bypass locks)
- Enabling other vendors application to DB2 required significant effort to re-order table access to avoid deadlocks



	Blocks ->	Reader	Writer
DBMS Snapshot Isolation	Reader	No	No
	Writer	No	Yes

	Blocks ->	Reader	Writer
Pre-DB2 9.7CS Isolation	Reader	No	Maybe
	Writer	Yes	Yes

	Blocks ->	Reader	Writer
DB2 9.7 CS Isolation w/CC	Reader	No	No
	Writer	No	Yes

- Log based
- No management overhead
- No performance overhead

Alternate Vendor Support

- Support other vendor's SQL
 - ▶ Easy for developers to query DB2
- Native execution of all SQL statements
 - ▶ Fast performance for queries
- Easily import other vendor's schemas
 - ▶ Easy for developers to set up DB2
- Support other vendor's concurrency models
 - ▶ Easy for developers to use DB2
- Support flexible data typing
 - ▶ Easy for developers to work with DB2
- And more...



Extended data types support in DB2 9.7

Type	Comment
NUMBER	Exploits P6 hardware accelerated DECFLOAT
VARCHAR2	NULL = "", trailing blank sensitive collation
TIMESTAMP(n)	0 (date + time) <= N <= 12 (date + time + picoseconds)
"DATE"	Year to seconds, SYSDATE
BOOLEAN	In procedural code
INDEX BY	Associative arrays in procedural code
VARRAY	Regular arrays in procedural code
Row Type	In procedural code, VARRAY, INDEX BY
Ref Cursor	Allows passing, and predefining of cursors

TIMESTAMP WITH TIMEZONE and **INTERVAL** in future release

Extended functions in DB2 9.7

Function	Comment
Conversion and Formatting	TO_CHAR, TO_DATE, TO_TIMESTAMP, TO_NUMBER, TO_CLOB
Datetime arithmetic	EXTRACT, ADD_MONTHS, ...
String manipulation	INITCAP, RPAD, LPAD, INSTR, REVERSE, ...
Misc	DECODE, NVL, LEAST, GREATEST, BITAND

Enhanced SQL in DB2 9.7

Feature	Comment
CONNECT BY	Tree walk recursion, includes helper functions
(+)-join	Old style OUTER JOIN syntax
DUAL	Equivalent to SYSDDUMMY1
ROWNUM	Pseudo column syntax for ROW_NUMBER()
NEXTVAL/CURRVAL	Pseudo column syntax for sequences
MINUS	A synonym for EXCEPT
Unnamed inline views	Optional correlation names for subqueries
TRUNCATE table	
Public synonym	For table, sequence, module/package
CREATEd temp table	Temp table with persistent definition

Enhanced PL/SQL Features in DB2 9.7

Function	Comment
All logic	IF, WHILE, :=, etc..
EXCEPTION	Try/catch handling
User Defined Exceptions	Define conditions with or without SQLCODEs
Constant variables	Variables that cannot be set
FOR over range	Step through numbers
over SELECT	Step through result set of query
over cursor	Step through result set of cursor
%TYPE	Anchored scalar data types
%ROWTYPE	Anchored row types
BULK COLLECT/FETCH	Aggregate result set into array
FORALL	Pipe array into SQL statement
AUTONOMOUS transaction	Executes a procedure in an independent TX

No object local type definitions (package)

Enhanced PL/SQL in DB2 9.7

Area	Comment
Anonymous block	New also in SQL PL dialect
Scalar function	
Procedure	
Package	Known as MODULE in DB2
Trigger	

No pipelined table function support at this point.

Extended PL/SQL Package in DB2 9.7

Feature	Comment
CREATE PACKAGE	Defines prototypes and public objects
CREATE PACKAGE BODY	Defines content and private objects
Replace package body	Replace body without losing prototypes or public objects
PKG [BODY] VARIABLE	Public/private variables
CURSOR	Public/private cursors
TYPE	Public/private types
EXCEPTION	User defined exceptions
SYNONYM ON PACKAGE	Public synonyms

- DB2 shreds package and body into individual *module* objects
- External management view is preserved

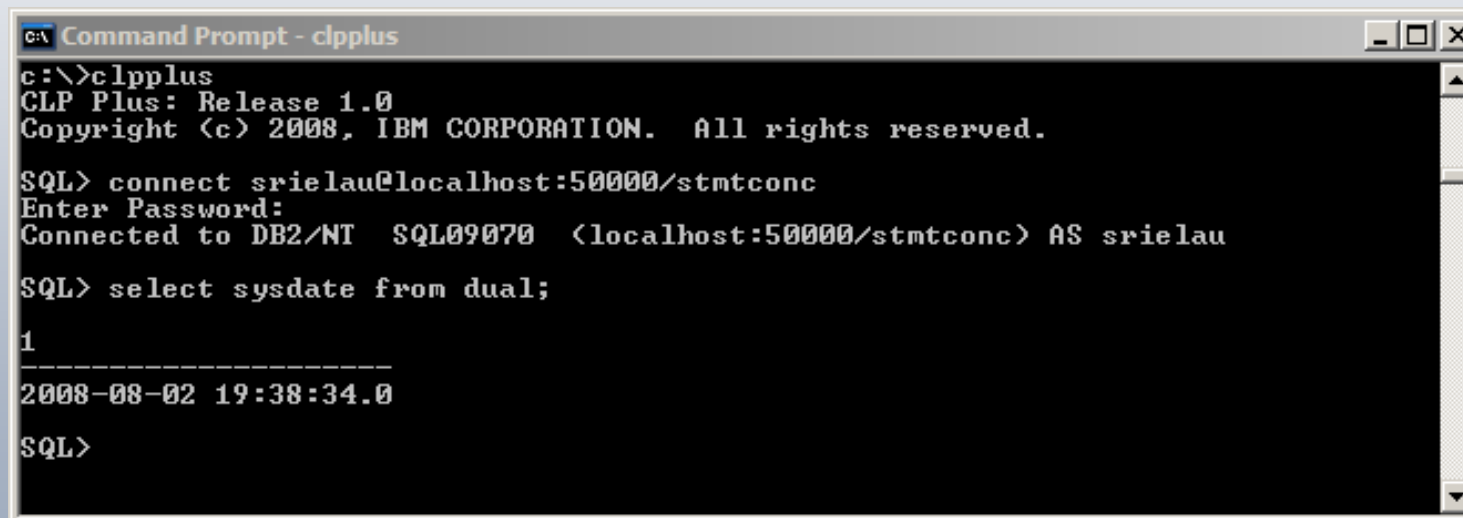
Built-in package libraries in DB2 9.7

Feature	Comment
DBMS_OUTPUT	“print debugging” and simple reporting
UTL_FILE	Server side I/O API
DBMS_ALERT	Cross session semaphoring
DBMS_PIPE	Cross session data pipe
DBMS_JOB	Job scheduler
DBMS_LOB	Alternate API to DB2 native LOB functions
DBMS_SQL	Alternate API to PREPARE/EXECUTE
DBMS_UTILITY	Misc functions and procedures
UTL_MAIL	Server API to email
UTL_SMTP	Server API to SMTP

Option to add more libraries as needed.

Using SQL*Plus scripts in DB2 9.7

- CLPPlus
 - ▶ SQL*Plus compatible command
 - ▶ Variable substitution
 - ▶ Column formatting
 - ▶ Simple reporting
 - ▶ Control variables



```
CA\ Command Prompt - clpplus
c:\>clpplus
CLP Plus: Release 1.0
Copyright (c) 2008, IBM CORPORATION. All rights reserved.

SQL> connect srielau@localhost:50000/stmtconc
Enter Password:
Connected to DB2/NT  SQL09070  (localhost:50000/stmtconc) AS srielau

SQL> select sysdate from dual;

1
-----
2008-08-02 19:38:34.0

SQL>
```

Extended SQL in DB2 9.7?

SQL	→	DB2 9.7
Concurrency Control	→	No change
DBMS SQL	→	No change
PL/SQL	→	No Change
Packages	→	No Change
Built-in packages	→	No Change
JDBC	→	No Change
SQL*Plus Scripts	→	No Change

It's easier to move to DB2. Changes are the exception. Not the rule.

DB2 has extended its SQL Syntax to support statement familiar to professionals working with other database software, eg Oracle

Number of Supported PL/SQL Statements

- SQL in procedural code:

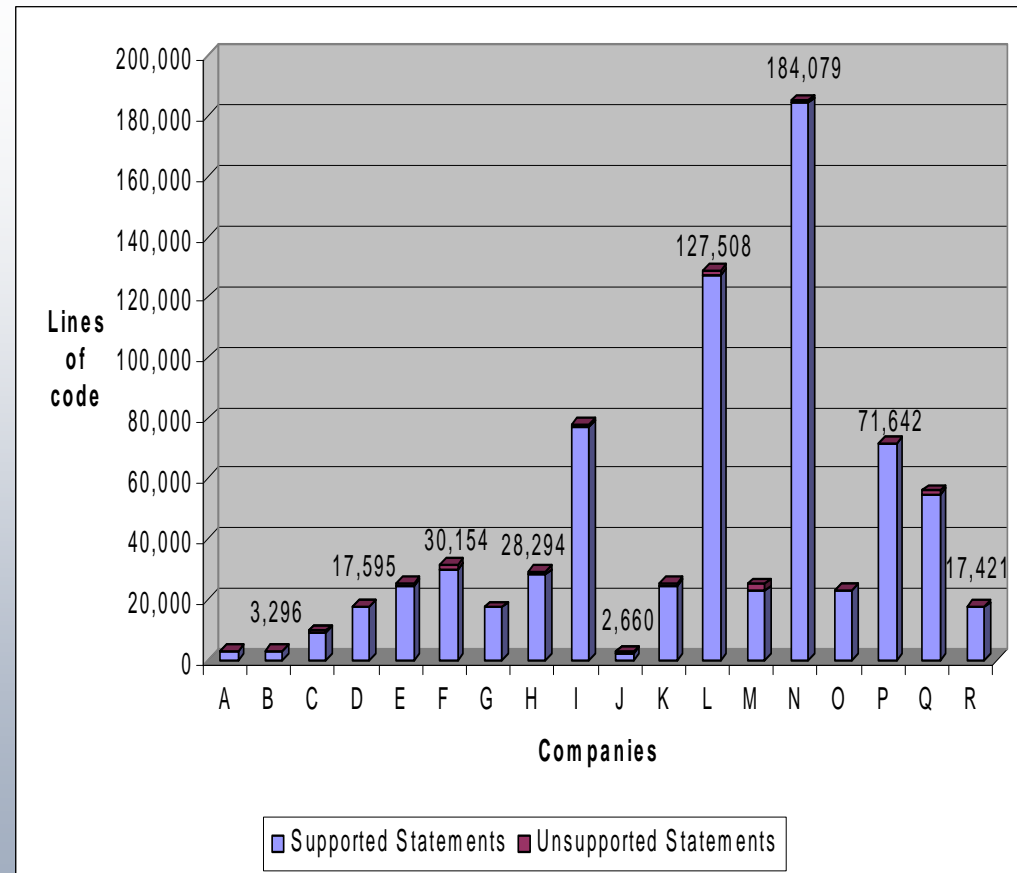
- ▶ Functions
- ▶ Built-in functions
- ▶ Procedures
- ▶ Packages
- ▶ Triggers

- Use measurement tool

- ▶ 18 companies agreed

- Variety of applications:

- ▶ Big (185k statements)
- ▶ Small (2k statements)

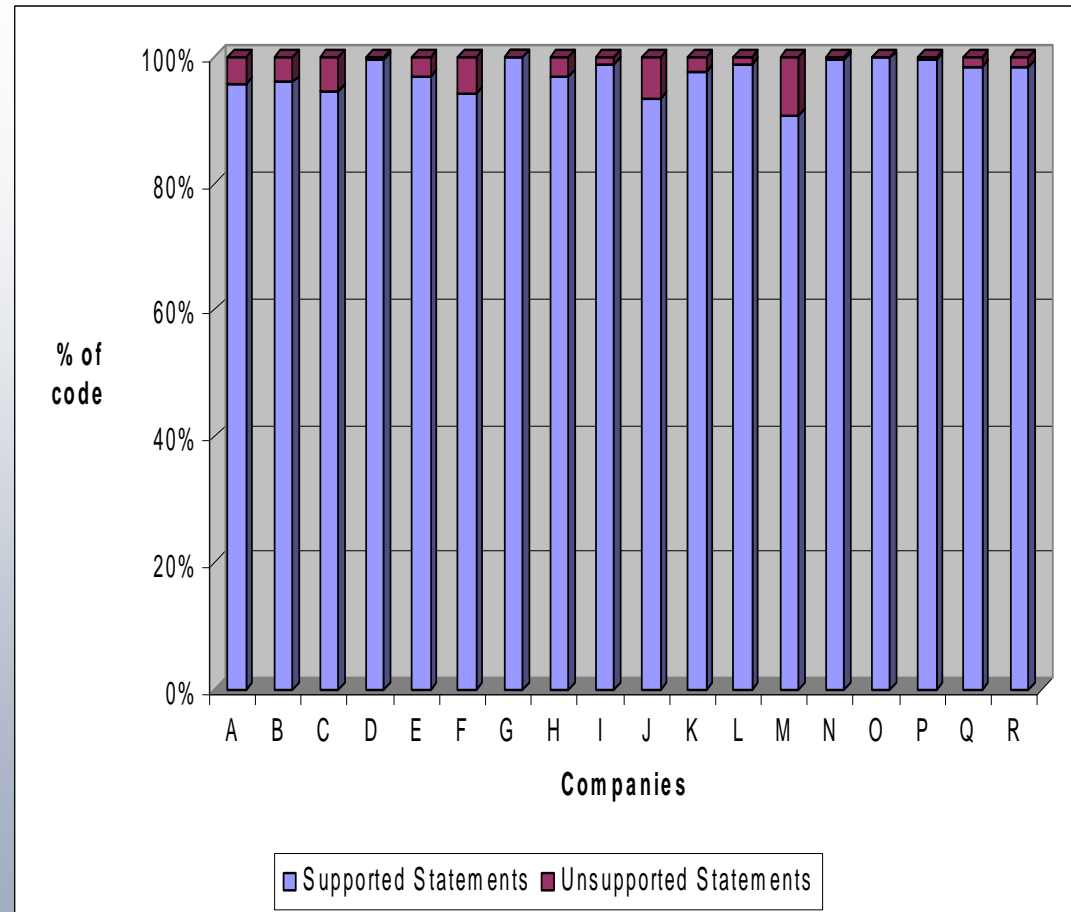


Percentage of Supported PL/SQL Statements

- Variety of participants:
 - ▶ Different industries
 - ▶ Different solutions
 - ▶ Different app sizes
 - ▶ Different countries

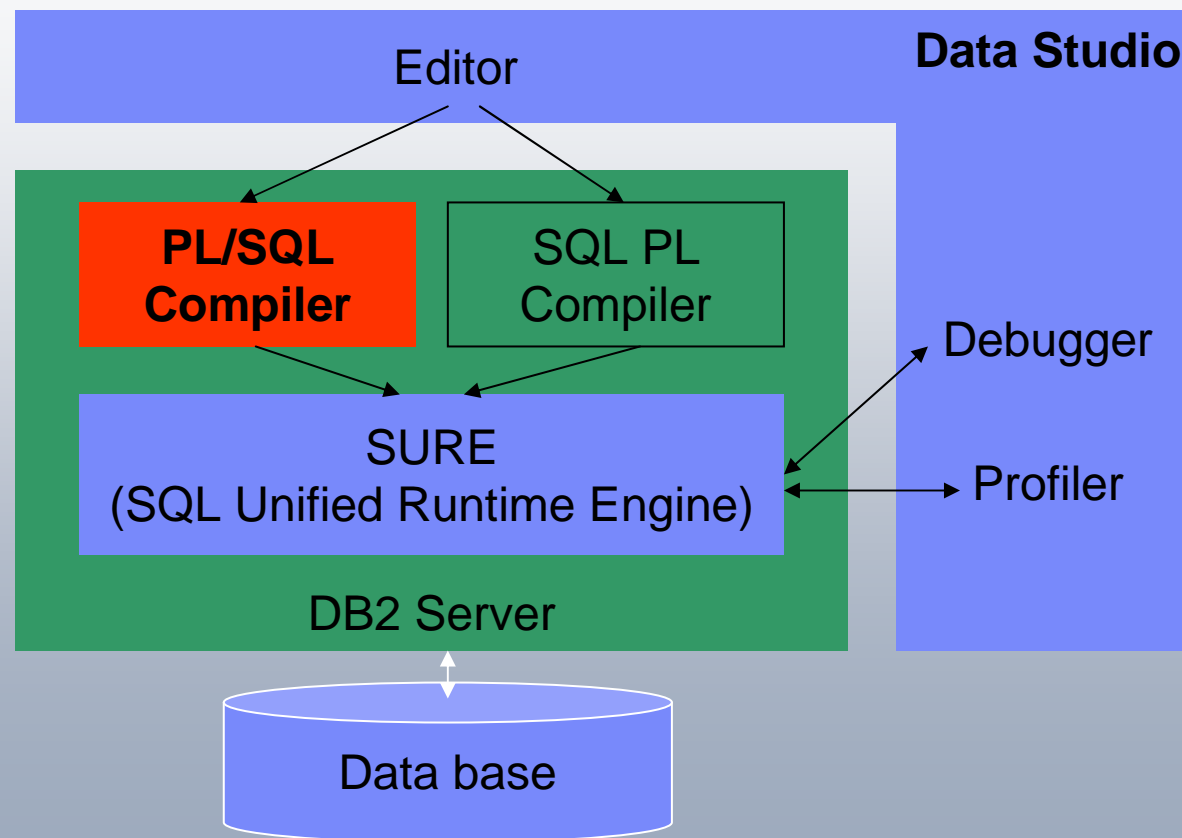
- PL/SQL supported:
 - ▶ > 750,000 lines of code

- Average: 98.43%



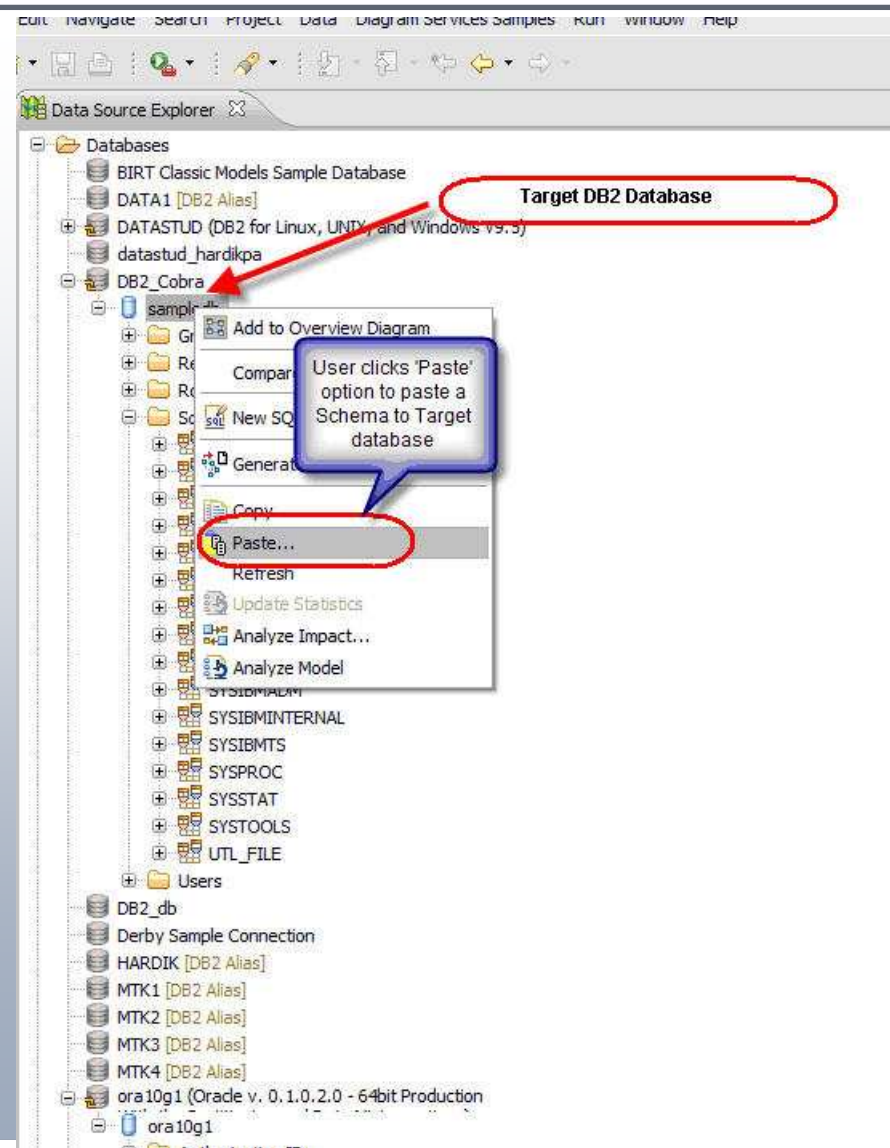
PL/SQL in DB2 9.7

- Built in PL/SQL compiler
- Source level debugging and profiling

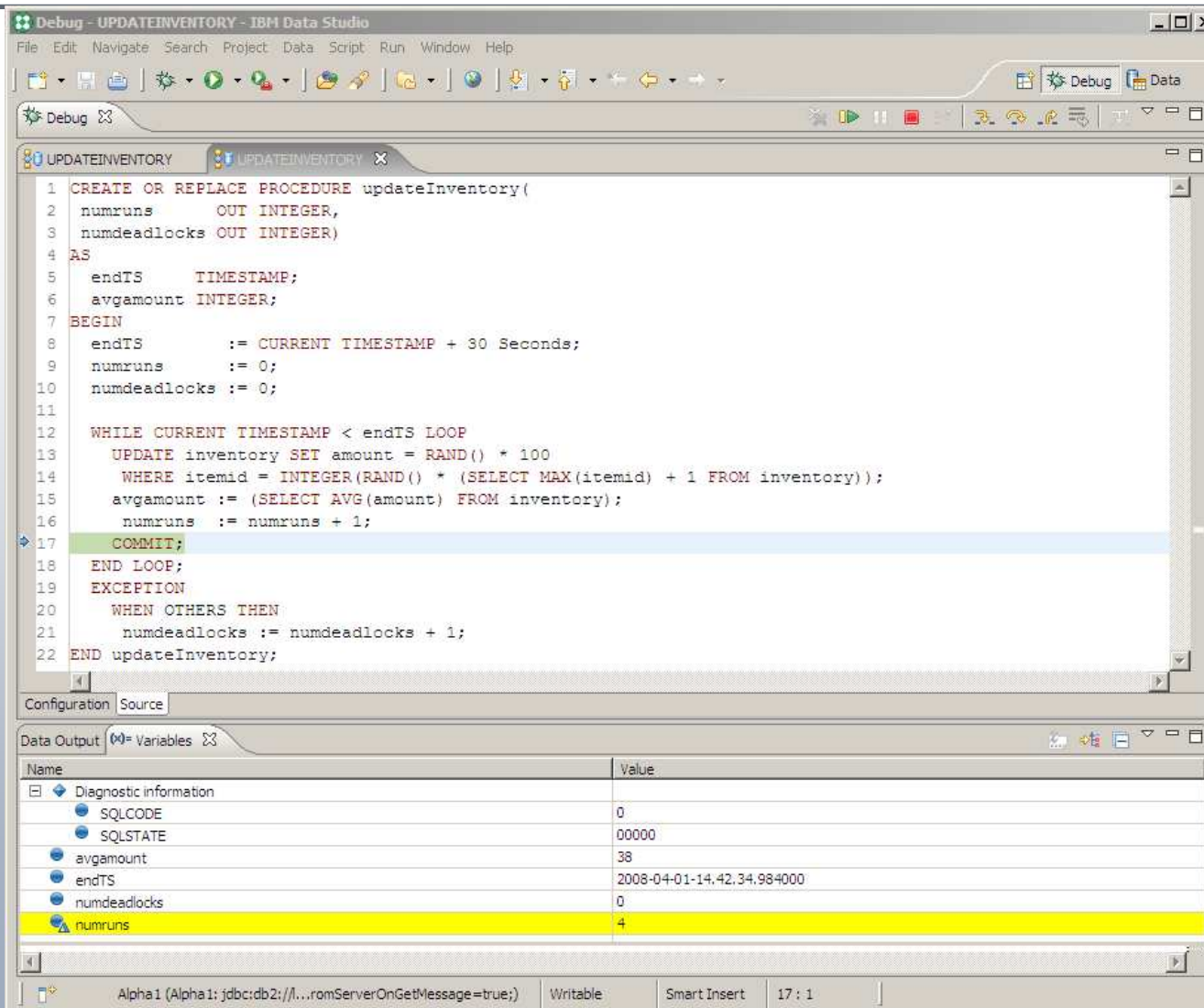


Easily Import Other Vendor's Schemas

- Drag and drop databases between other DBMS – for example Oracle Database and DB2 for quick migrations and trials
- Easily map schemas and data types from other DBMS – for example Oracle Database to DB2
- Automatically map schemas and data types
- Oracle Database developers can quickly start using DB2



Debugging PL/SQL in DB2 9.7





The screenshot shows the IBM Data Studio interface during a PL/SQL debug session. The main window displays the source code of a procedure named 'updateInventory'. The code includes a loop that updates inventory amounts and counts runs and deadlocks. The current execution point is at line 17, 'COMMIT;'. The Data Output window shows the following variables and their values:

Name	Value
Diagnostic information	
SQLCODE	0
SQLSTATE	00000
avgamount	38
endTS	2008-04-01-14.42.34.984000
numdeadlocks	0
numruns	4

At the bottom of the window, the connection details are visible: Alpha1 (Alpha1: jdbc:db2://...romServer:OnGetMessage=true;) | Writable | Smart Insert | 17 : 1

Enabling DB2: Before and After

- 
- Before:
 - Map schema and data types
 - Move data
 - Translate (semi-automated)
 - ▶ Triggers
 - ▶ Procedures
 - ▶ Functions
 - ▶ Anonymous blocks
 - Translate SQL (manual)
 - Debugging
 - Tuning
 - Parallel production
 - Cut over
 - After:
 - Import schema and data types
 - Move data
 - Runs natively
 - Handle exceptions
 - Debugging
 - DB2 auto tunes
 - Parallel production
 - Cut over
- 
- Now takes days instead of months!**

IBM DB2 for Linux, UNIX and Windows



Lowest TCO

*Unparalleled Automation
Deep Compression
Lightning Fast*

Simple to Run

*Flexible Development
Industry leading XML support*

Most Reliable

*World class audit & security
Easy High Availability
Workload Management*

Business Performance Advantage



25 of the Top 25 Worldwide Banks



9 of the Top 10 Global Life / Health Insurance Providers



23 of the Top 25 US Retailers



Cost Effective Solutions

