



IBM Software Group

DB2 for z/OS Product Direction and Technology Futures

DB2 Information Management Software



Mike Bracey
Systems Engineer
DB2 for z/OS
IBM United Kingdom Ltd

@business on demand software

Disclaimer and Trademarks

Information contained in this material has not been submitted to any formal IBM review and is distributed on "as is" basis without any warranty either expressed or implied. Measurements data have been obtained in laboratory environment. Information in this presentation about IBM's future plans reflect current thinking and is subject to change at IBM's business discretion. You should not rely on such information to make business plans. The use of this information is a customer responsibility.

IBM MAY HAVE PATENTS OR PENDING PATENT APPLICATIONS COVERING SUBJECT MATTER IN THIS DOCUMENT. THE FURNISHING OF THIS DOCUMENT DOES NOT IMPLY GIVING LICENSE TO THESE PATENTS.

TRADEMARKS: THE FOLLOWING TERMS ARE TRADEMARKS OR ® REGISTERED TRADEMARKS OF THE IBM CORPORATION IN THE UNITED STATES AND/OR OTHER COUNTRIES: AIX, AS/400, DATABASE 2, DB2, e-business logo, Enterprise Storage Server, ESCON, FICON, OS/390, OS/400, ES/9000, MVS/ESA, Netfinity, RISC, RISC SYSTEM/6000, iSeries, pSeries, xSeries, SYSTEM/390, IBM, Lotus, NOTES, WebSphere, z/Architecture, z/OS, zSeries,  

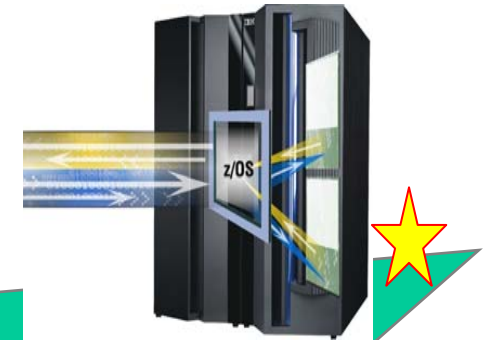
The FOLLOWING TERMS ARE TRADEMARKS OR REGISTERED TRADEMARKS OF THE MICROSOFT CORPORATION IN THE UNITED STATES AND/OR OTHER COUNTRIES: MICROSOFT, WINDOWS, WINDOWS NT, ODBC, WINDOWS 95

For additional information see ibm.com/legal/copytrade.phtml



Technology Evolution with Mainframe Specialty Engines

★ Building on a strong track record of technology innovation with specialty engines, IBM intends to introduce the System z9 Integrated Information Processor



IBM System z9 Integrated Information Processor (IBM zIIP) planned for 2006

Designed to help improve resource optimization for eligible data workloads within the enterprise



System z9 Application Assist Processor (zAAP) 2004

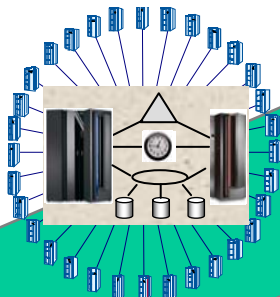
Incorporation of JAVA into existing mainframe solutions



Integrated Facility for Linux (IFL) 2001

Support for new workloads and open standards

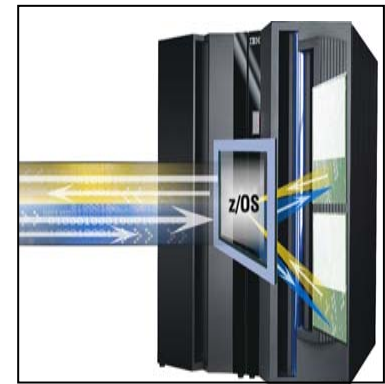
Centralized data sharing across mainframes



Internal Coupling Facility (ICF) 1997

New IBM System z9 Integrated Information Processor (IBM zIIP)

- New specialty engine for the System z9 mainframe (planned for 2006) designed to help:
 - Customers integrate data across the enterprise
 - Improve resource optimization and lower the cost of ownership for eligible data serving workloads
- z/OS manages and directs work between the general purpose processor and the zIIP
 - Number of zIIPs per z9-109 not to exceed number of standard processors. No changes anticipated to DB2 for z/OS V8 applications
- DB2 for z/OS V8 will be first user of the zIIP with
 - System z9 109
 - z/OS 1.6 or later
 - DB2 for z/OS V8
- Webcast replay ibm.com/servers/systems/z/2006/



DB2 V8 and IBM zIIP can add value to database work

- Portions of the following DB2 for z/OS V8 workloads may benefit from zIIP*:

1 - ERP, CRM, Business Intelligence or other enterprise applications

- Via DRDA over a TCP/IP connection



2 - Data warehousing applications*

- Requests that utilize star schema parallel queries

3 - DB2 for z/OS V8 utilities*

- Internal DB2 utility functions used to maintain index maintenance structures

* The zIIP is designed so that a program can work with z/OS to have all or a portion of its enclave Service Request Block (SRB) work directed to the zIIP. The above types of DB2 V8 work are those executing in enclave SRBs, of which portions can be sent to the zIIP.

V9 DB2 Technology Themes

- ❑ **Enable high-volume transaction processing for next wave of Web applications**
- ❑ **Extend the lead in transaction processing availability, scalability and performance**
- ❑ **Reduce cost of ownership and zSeries-specific skill needs**
- ❑ **Improve data warehousing and OLTP reporting**

Beta Programme Announced 2nd May

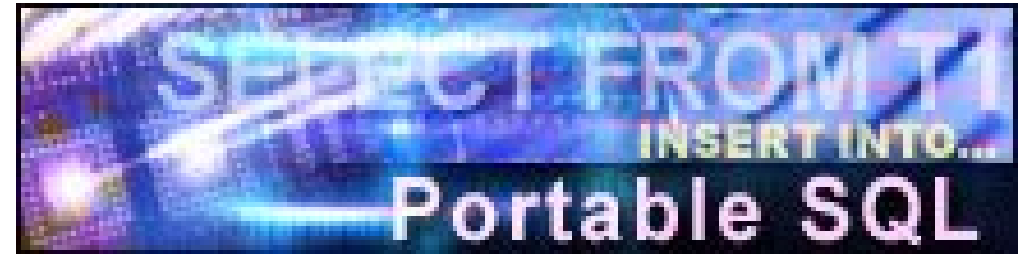


DB2 SQL

z z/OS V7

common

LUW Linux, Unix & Windows V8.2



z { Range partitioning

c
o
m
m
n { Inner and Outer Joins, Table Expressions, Subqueries, GROUP BY, Complex Correlation, Global Temporary Tables, CASE, 100+ Built-in Functions, Limited Fetch, Insensitive Scroll Cursors, UNION Everywhere, MIN/MAX Single Index Support, Self Referencing Updates with Subqueries, Sort Avoidance for ORDER BY, and Row Expressions, Call from trigger, statement isolation

L
U
W { Updateable UNION in Views, ORDER BY/FETCH FIRST in subselects & table expressions, GROUPING SETS, ROLLUP, CUBE, INSTEAD OF TRIGGER, EXCEPT, INTERSECT, 16 Built-in Functions, MERGE, Native SQL Procedure Language, SET CURRENT ISOLATION, BIGINT data type, file reference variables, SELECT FROM UPDATE, DELETE & MERGE, multi-site join, 2M Statement Length, GROUP BY Expression, Sequences, Scalar Fullselect, Materialized Query Tables, Common Table Expressions, Recursive SQL, CURRENT PACKAGE PATH, VOLATILE Tables, Star Join Sparse Index, Qualified Column names, Multiple DISTINCT clauses, ON COMMIT DROP, Transparent ROWID Column, FOR READ ONLY KEEP UPDATE LOCKS, SET CURRENT SCHEMA, Client special registers, long SQL object names, SELECT from INSERT

DB2 SQL

z z/OS V8

common

LUW Linux, Unix & Windows V8.2



Z

Multi-row INSERT, FETCH & multi-row cursor UPDATE, Dynamic Scrollable Cursors, GET DIAGNOSTICS, Enhanced UNICODE for SQL, join across encoding schemes, IS NOT DISTINCT FROM, Session variables, range partitioning

C

O

M

M

O

N

Inner and Outer Joins, Table Expressions, Subqueries, GROUP BY, Complex Correlation, Global Temporary Tables, CASE, 100+ Built-in Functions including SQL/XML, Limited Fetch, Insensitive Scroll Cursors, UNION Everywhere, MIN/MAX Single Index Support, Self Referencing Updates with Subqueries, Sort Avoidance for ORDER BY, and Row Expressions, 2M Statement Length, GROUP BY Expression, Sequences, Scalar Fullselect, Materialized Query Tables, Common Table Expressions, Recursive SQL, CURRENT PACKAGE PATH, VOLATILE Tables, Star Join Sparse Index, Qualified Column names, Multiple DISTINCT clauses, ON COMMIT DROP, Transparent ROWID Column, Call from trigger, statement isolation, FOR READ ONLY KEEP UPDATE LOCKS, SET CURRENT SCHEMA, Client special registers, long SQL object names, SELECT from INSERT

L

U

W

Updateable UNION in Views, ORDER BY/FETCH FIRST in subselects & table expressions, GROUPING SETS, ROLLUP, CUBE, INSTEAD OF TRIGGER, EXCEPT, INTERSECT, 16 Built-in Functions, MERGE, Native SQL Procedure Language, SET CURRENT ISOLATION, BIGINT data type, file reference variables, SELECT FROM UPDATE, DELETE & MERGE, multi-site join

DB2 SQL

z z/OS V9

common

LUW Linux, Unix & Windows V8.2



Z

Multi-row INSERT, FETCH & multi-row cursor UPDATE, Dynamic Scrollable Cursors, GET DIAGNOSTICS, Enhanced UNICODE for SQL, join across encoding schemes, IS NOT DISTINCT FROM, Session variables, range partitioning, TRUNCATE, DECIMAL FLOAT, VARBINARY, optimistic locking, FETCH CONTINUE, ROLE, MERGE

C
o
m
m
o
n

Inner and Outer Joins, Table Expressions, Subqueries, GROUP BY, Complex Correlation, Global Temporary Tables, CASE, 100+ Built-in Functions including SQL/XML, Limited Fetch, Insensitive Scroll Cursors, UNION Everywhere, MIN/MAX Single Index Support, Self Referencing Updates with Subqueries, Sort Avoidance for ORDER BY, and Row Expressions, 2M Statement Length, GROUP BY Expression, Sequences, Scalar Fullselect, Materialized Query Tables, Common Table Expressions, Recursive SQL, CURRENT PACKAGE PATH, VOLATILE Tables, Star Join Sparse Index, Qualified Column names, Multiple DISTINCT clauses, ON COMMIT DROP, Transparent ROWID Column, Call from trigger, statement isolation, FOR READ ONLY KEEP UPDATE LOCKS, SET CURRENT SCHEMA, Client special registers, long SQL object names, SELECT from INSERT, UPDATE, DELETE & MERGE, INSTEAD OF TRIGGER, Native SQL Procedure Language, BIGINT, file reference variables, XML, FETCH FIRST & ORDER BY in subselect and fullselect, caseless comparisons, INTERSECT, EXCEPT, not logged tables

L
U
W

Updateable UNION in Views, GROUPING SETS, ROLLUP, CUBE, 16 Built-in Functions, SET CURRENT ISOLATION, multi-site join, MERGE

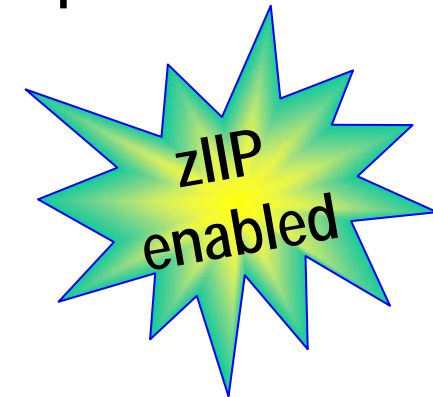
DB2 for z/OS V9 SQL, DB2 family & porting



- XML
- MERGE
- SELECT FROM UPDATE, DELETE, MERGE
- TRUNCATE
- INSTEAD OF TRIGGER
- BIGINT, VARBINARY, DECIMAL FLOAT
- Native SQL Procedure Language
- Optimistic locking
- LOB File reference variable & FETCH CONTINUE
- FETCH FIRST & ORDER BY in subselect and fullselect
- INTERSECT & EXCEPT
- ROLE & trusted context
- Many new built-in functions, caseless comparisons
- Index on expression
- Improved DDL consistency
- CURRENT SCHEMA

Native SQL Procedural Language

- Eliminates generated C code and compilation
- Fully integrated into the DB2 engine
- Extensive support for versioning:
 - VERSION keyword on CREATE PROCEDURE
 - CURRENT ROUTINE VERSION special register
 - ALTER ADD VERSION
 - ALTER REPLACE VERSION
 - ALTER ACTIVATE VERSION
- BIND PACKAGE with new DEPLOY keyword



Optimistic Locking Support

- Built-in timestamp for each row or page
 - Automatically updated by DB2
 - Allows simple timestamp predicate to validate that row has not changed since last access
- Eliminates need for complex predicates on WebSphere CMP updates, improves performance



XML Processing Paradigms

XML has become the “data interchange” format between B2B/B2C, inter- and intra-enterprise environments.

XML View Of Relational Data

- SQL data viewed and updated as XML
 - Done via document shredding and composition
- DTD and Schema Validation

XML Documents As Monolithic Entities

- Atomic Storage And Retrieval
- Search Capabilities

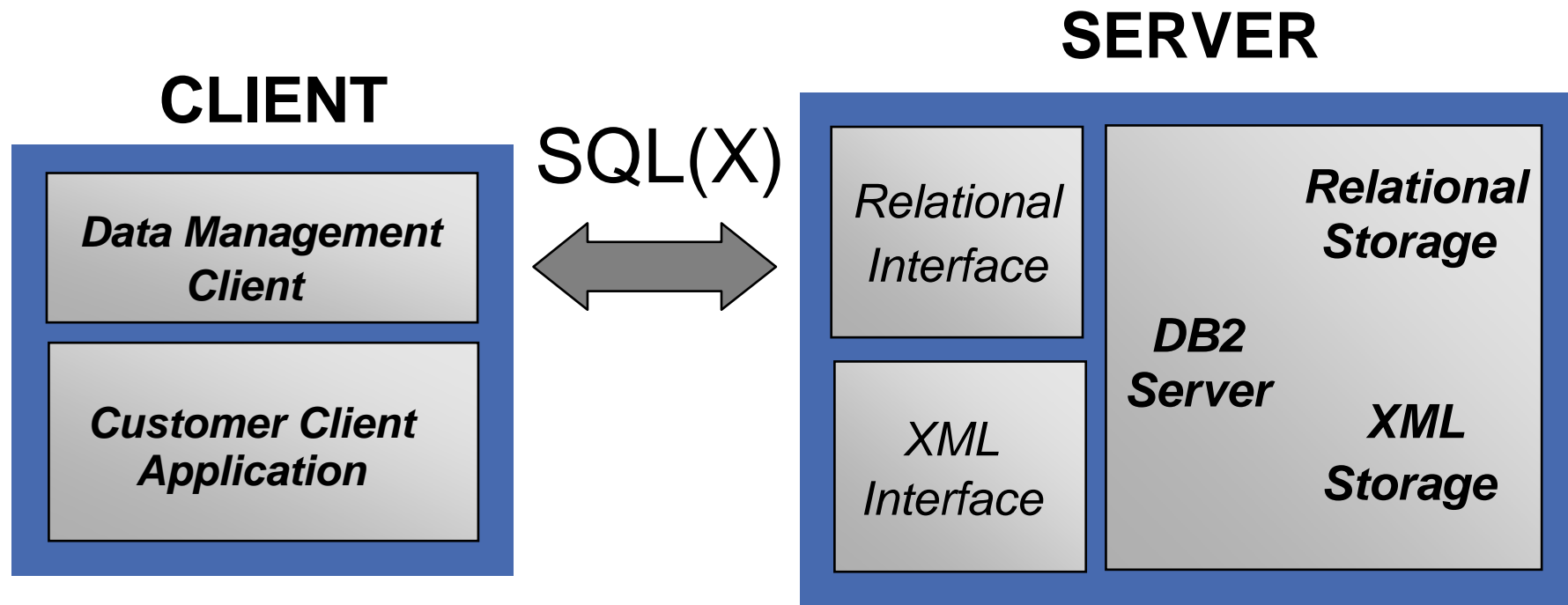
XML As A Rich Data Type

- Full Storage and Indexing
- Powerful Query Capabilities



XML Capabilities Inside the Engine

Performance, Performance, Performance



Native storage Schema Index functions utilities

XML Support

- Support XML data type
 - Store the XML document natively
 - DDL --
 - CREATE/ALTER Table with XML type column
 - Implicitly create XML Auxiliary objects (tablespace/table/index) - one per XML column
 - Index support
 - Created by users
 - uses XPath to determine which nodes in the XML document to index.
- ```
CREATE INDEX dependentName ON deptTable(deptDocs)
GENERATE KEY USING XMLPATTERN
'/department/empl/dependent/name' ATOMIC AS SQL
VARCHAR(20);
```
- INSERT/UPDATE/DELETE
    - INSERT with VALUES and SUBSELECT
    - No Subdocument update





# XML Support -- Query

- Enhanced V8 XML Constructors (XML Publishing Functions)
- SQL/XML Functions and Predicates
  - XMLParse - Convert a XML text to XML value
  - XMLSerialize - Converts XML to character type
  - XMLQuery - executes an XPath expression against an XML value.

```
SELECT XMLQUERY ('//item[USPrice = $price]' PASSING
PO.POrder,
```

```
T.price AS "price") FROM PurchaseOrders PO, T;
```

- XMLCast - Cast XML to other types or other types to XML
- XMlexists - a predicate, which returns TRUE if the XPath expression evaluates to a non-empty sequence

```
SELECT PO.pid FROM PurchaseOrders PO, T
```

```
WHERE XMLEXISTS('//item[USPrice = $price]'
PASSING PO.POrder, T.price AS "price")
```



# XML Support (last one!)

- XPATH supported features from XPath 2.0:
- Utility Support
  - LOAD/UNLOAD, CHECK DATA/INDEX, COPY, REBUILD, RECOVER, REORG, etc.
- XML Schema Support
  - XSR – XML Schema Repository
  - Tables to store XML schemas
  - Stored procedures to register XML schemas
- DSN\_XMLVALIDATE() SQL/XML function
  - Test XML values for validity against XML schema
  - Obtain default values and schema normalized values from XML schema
- XML decomposition using annotated XML schema



# Decimal Floating Point

- New datatype DECFLOAT
  - Well suited to typical customer financial calculations
  - Similar to “calculator” mathematics
    - Eliminates rounding errors by using base 10 math
    - Has up to 34 digits of precision
    - Floating point convenience with fixed point precision!!!
  - Hardware support will be provided in the next zSeries processor generation (new IEEE standard)
    - Software emulation provided for other models



# MERGE

- Array MERGE operation
- Targets OLTP applications like SAP

```
MERGE INTO account AS T
USING VALUES (:hv_id, :hv_amt) FOR 5 ROWS AS S(id,amt)
ON T.id = S.id
WHEN MATCHED THEN
 UPDATE SET balance = T.balance + S.amt
WHEN NOT MATCHED THEN
 INSERT (id, balance) VALUES (S.id, S.amt)
NOT ATOMIC CONTINUE ON SQLEXCEPTION
```

# SQL Improvements – Family Compatibility

- **INSTEAD OF** triggers
- **SELECT FROM UPDATE**
- **SELECT FROM DELETE**
- **SELECT FROM MERGE**
- **BIGINT, BINARY** and **VARBINARY** data types
- **ORDER BY** and **FETCH FIRST** in subselect



# DDF Improvements

- 64-bit exploitation by DDF
  - Special “shared private” with xxxDBM1 to eliminate many of the data moves on SQL operations
- Support for IPv6 and SSL
- VTAM definition is now optional
- Prepare for elimination of PRIVATE protocol requester
  - Includes tools for identifying which packages need to be bound at remote servers



# DB2 V9 Themes

- ❑ **Enable high-volume transaction processing for next wave of Web applications**
- ❑ **Extend the lead in transaction processing availability, scalability and performance**
- ❑ **Reduce cost of ownership and zSeries-specific skill needs**
- ❑ **Improve reporting**





# Schema Evolution – Database Definition On Demand

- Fast replacement of one table with another
- Rename column and index
- Alter index to change page size
- Table space that can add partitions, for growth
- Improve ability to rebuild an index online
- Online reorganization with no BUILD2 phase
- Modify early code without requiring an IPL
- Alter table space and index logging
- Create & alter STOGROUP SMS constructs



# CLONE Tables

- Allows fast replacing production data without renames and rebinds
  - A capability to support online load replace
- **CREATE TABLE** to create a Clone Table
  - All indexes are also cloned
  - Table and Index data are not copied
  - Base and Clone tables share the same table space and index names
  - Underlying data sets are differentiated by a data set instance number



# CLONE Tables...

- A clone table can only be created
  - On a single table in a table space (partitioned or non-partitioned)
  - No RI or Trigger on the base table
  - No new index can be created on a base table that has a clone
- Use insert or load to populate clone tables
- Utilities (except RUNSTATS) can operate on clone tables with a new CLONE keyword



## TRUNCATE Statement

- Allows fast delete of all rows in a given table (simple, segmented, or partitioned)
- Very useful for nightly refresh of summary tables, warehouses, etc.

```
TRUNCATE TABLE TABLE-NAME
```

```
< DROP STORAGE | REUSE STORAGE >
```

```
< RESTRICT WHEN DELETE TRIGGERS |
 IGNORE DELETE TRIGGERS >
```

```
< IMMEDIATE >
```



# Partition by Growth & Universal Table Space

- New partitioning scheme:
  - Single table tablespace, where each partition contains a segmented pageset (allows segmented to increase from 64GB to 16TB or 128 TB with 32K pages)
- Partition By Growth
  - Eliminates need to define partitioning key and assign key ranges
  - A new partition is created when a given partition reaches DSSIZE (defaults to 64G)
  - Retains benefits of Utilities and SQL parallelism optimizations for partitioned tables



# CREATE TABLE ... APPEND(YES)

- New APPEND option:
  - Maximizes performance for “INSERT at end”
  - Avoids overhead of attempting to preserve clustering sequence
  - CREATE or ALTER table



# Relief for Sequential Key INSERT

- New index page sizes: 8K, 16K and 32K
  - Fewer page splits for long keys
  - More key values per page
- INSERT at the end of the key range used to result in 50% free space in each index page
  - Enhanced support dynamically adapts page split boundary to minimize wasted space in index pages





# LOB Performance/Scalability

- Elimination of LOB locks – LRSN and page latching is used instead for consistency checks
- New network flows for delivering LOBs
  - JDBC, SQLJ, and CLI will let server determine whether to flow LOB values or LOCATORs based on size thresholds
  - Significant reduction in network traffic
  - Greatly reduces frequency of FREE LOCATOR statements



# Other Performance / Availability Items

- Insert performance APPEND INDEX LOG
  - INDEX on expression
  - Not logged table space
- LOB performance, function, scalability
- CPU reductions in LOAD and REORG
- Online REBUILD INDEX
- Improved varying length performance
- FETCH FIRST n ROWS improvements
  - Can now be specified in a subquery or fullselect
  - ORDER BY now exploits FETCH FIRST n ROWS, so that work files are not created (less I/O)



# DB2 V9 Themes

- Enable high-volume transaction processing for next wave of Web applications**
- Extend the lead in transaction processing availability, scalability and performance**
- Reduce cost of ownership and zSeries-specific skill needs**
- Improve data warehousing and OLTP reporting**



# Introduction to z/OS and the IBM System z Mainframe Environment

**Summary:** This class will teach the basics of the z/OS operating system, the practical skills to navigate and work in a z/OS environment and the zSeries platform. It will discuss the job roles that are available, the fundamentals of the z/OS operating system, the languages in the application programming environment, and the concepts of the zSeries hardware. There will be hands-on labs to view system setup, create data sets, run system utilities, and work with z/OS resources, environments, and applications.

– **Audience:** Computer professionals, especially those who are fairly new to the IT arena, who will move from another operating system to z/OS but have no knowledge of z/OS. The class will be especially useful for those customers who are or will be consolidating servers onto the mainframe.

– **Duration:** 5.0 days

– **Class dates:**

– June 19 - 23, 2006

**Location:** IBM Hursley, United Kingdom

– July 17 - 21, 2006

**Location:** IBM Stuttgart, Germany

– September 25 - 29, 2006

**Location:** IBM Hursley, United Kingdom

– October 16 - 20, 2006

**Location:** IBM Stuttgart, Germany

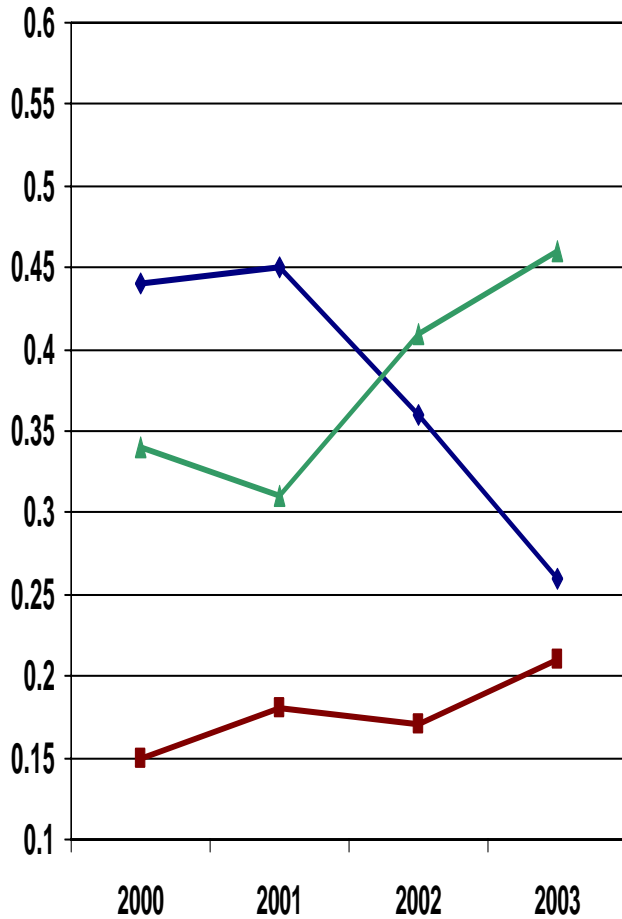
– November 13 - 17, 2006

**Location:** IBM Stuttgart, Germany

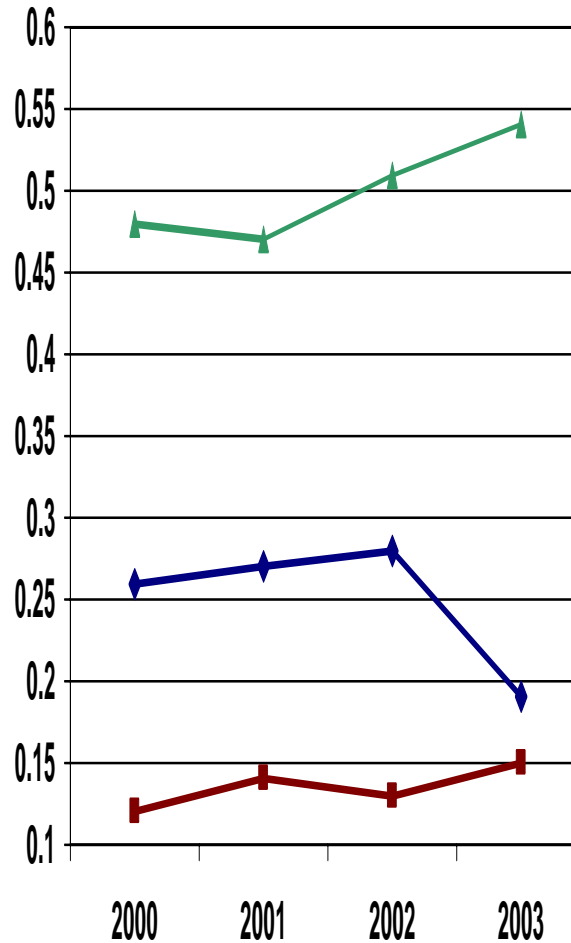


# Cost Of Ownership Trends

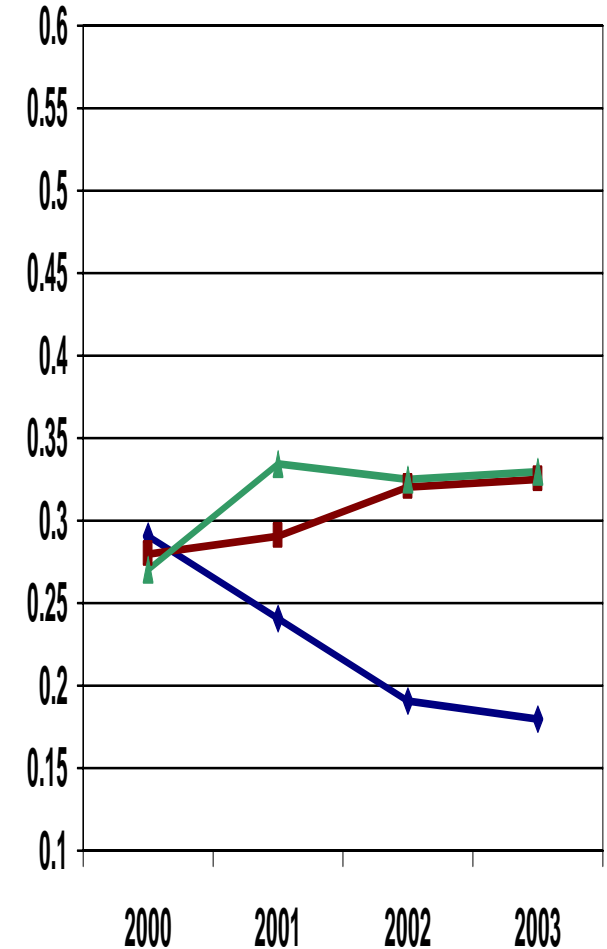
## UNIX



## INTEL



## Mainframe



- ▲— % Personnel
- ◆— % Hardware
- % Software

Meta Group Survey Data

# Compliance/Auditing Pressure

- Regulatory compliance initiatives are impacting IT organizations in most countries/industries, and changing fast
  - Sarbanes-Oxley
  - Basel II
  - FDA: Food and Drug Administration 21 CFR Part 11
  - COPPA: Children's Online Privacy Protection Act of 2000
  - DPA: Data Protection Act (UK)
  - HIPAA: Health Insurance Portability and Accountability Act of 1996
  - PIPEDA: Personal Information Protection and Electronic Documents Act (Canada)
  - SEC Rule 17a-4: Records to be preserved by certain exchange members, brokers, dealers
  - USA Patriot Act: Uniting and Strengthening America by Providing Tools Required to Intercept and Obstruct Terrorism of 2001
- Focus is on both external threats (hackers) and internal employees



## Security in DB2 for z/OS V9

### Some key implementations

- Data Encryption
- Roles
- Network Trusted Contexts
- Instead of Triggers
- Improved auditing
- Secure Socket Layer





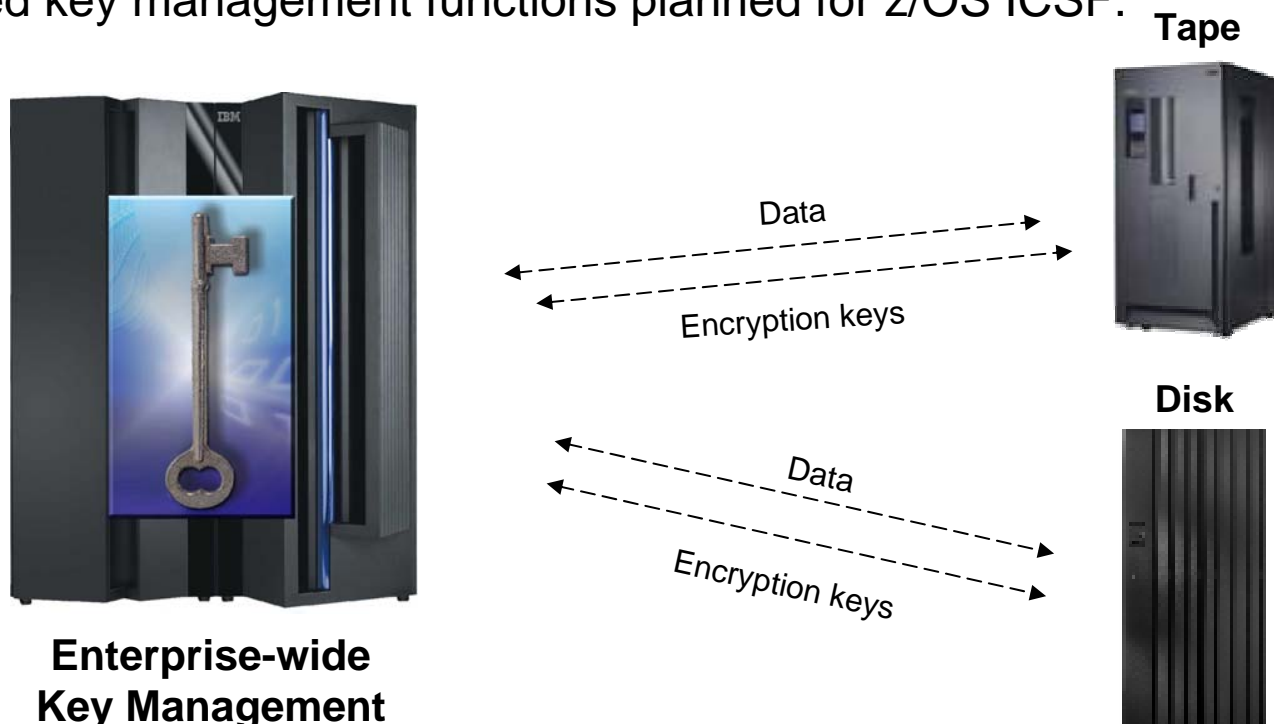
# Protecting data on disk

- We will allow encryption for the key disk resources used by DB2:
  - Tables
  - LOBs
  - Indexes
  - Image copies
  - Logs
  - Archive logs



## Future Directions – Extending Encryption to IBM TotalStorage

- Statement of Direction: To address customers' growing concern with data security, IBM is announcing a statement of direction for the development, enhancement and support of encryption capabilities within storage environments such that the capability does not require the use of host server resources.
- This includes the intent to offer, among other things, capabilities for products within the IBM TotalStorage® portfolio to support outboard encryption and to leverage the centralized key management functions planned for z/OS ICSF.



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

# Database ROLES

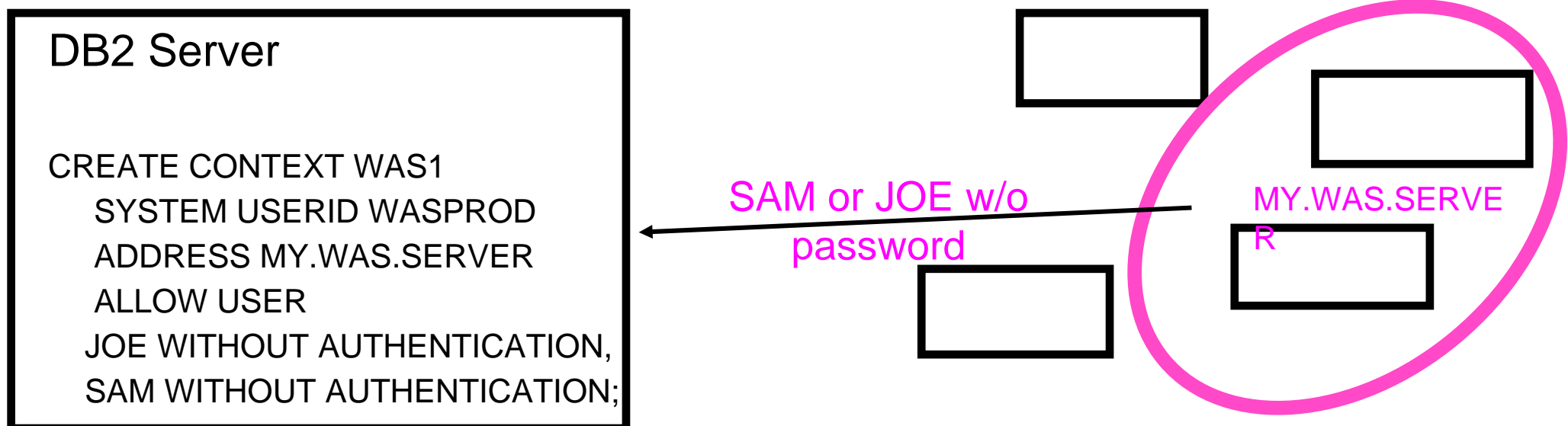
- ROLE is a “virtual authid”
  - Assigned via TRUSTED CONTEXT
  - Provides additional privileges only when in a trusted environment using existing primary AUTHID.
  - Can optionally be the OWNER of DB2 objects

```
CREATE ROLE PROD_DBA;
GRANT DBADM ... TO PROD_DBA;

CREATE TRUSTED CONTEXT DBA1 ...
 DEFAULT ROLE PROD_DBA OWNER(ROLE);
```

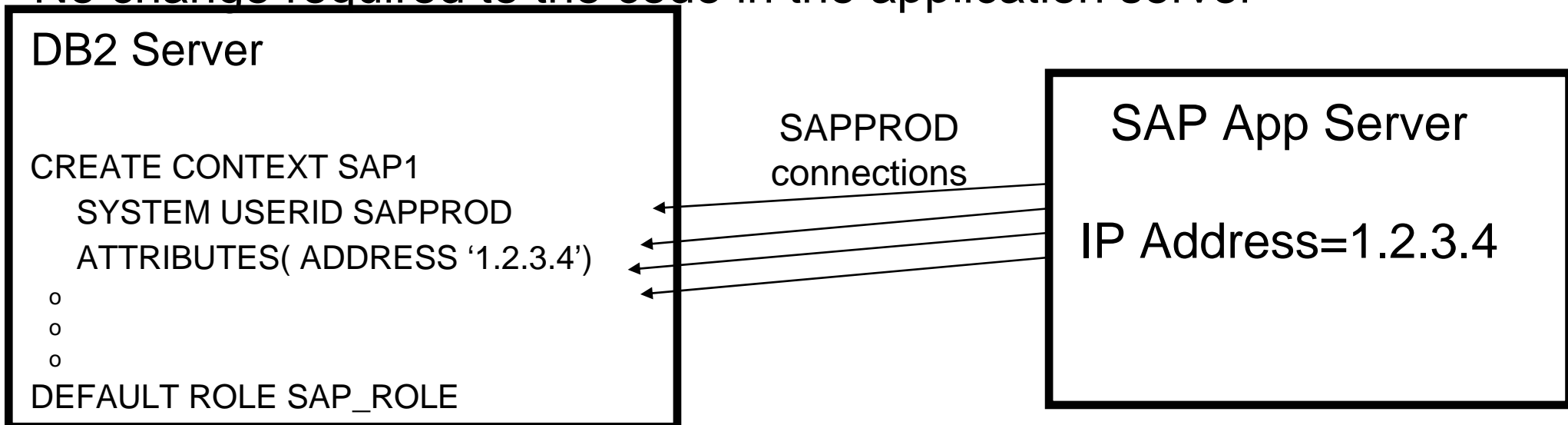
# Trusted Security Context

- Identifies “trusted” DDF, RRS Attach, or DSN application servers
- Allows selected DB2 authids on connections without passwords
  - reduces complexity of password management
  - reduces need for an all-inclusive “system authid” in app servers
  - more visibility/auditability of which user is current running
  - enables mixed security capabilities from a single app server



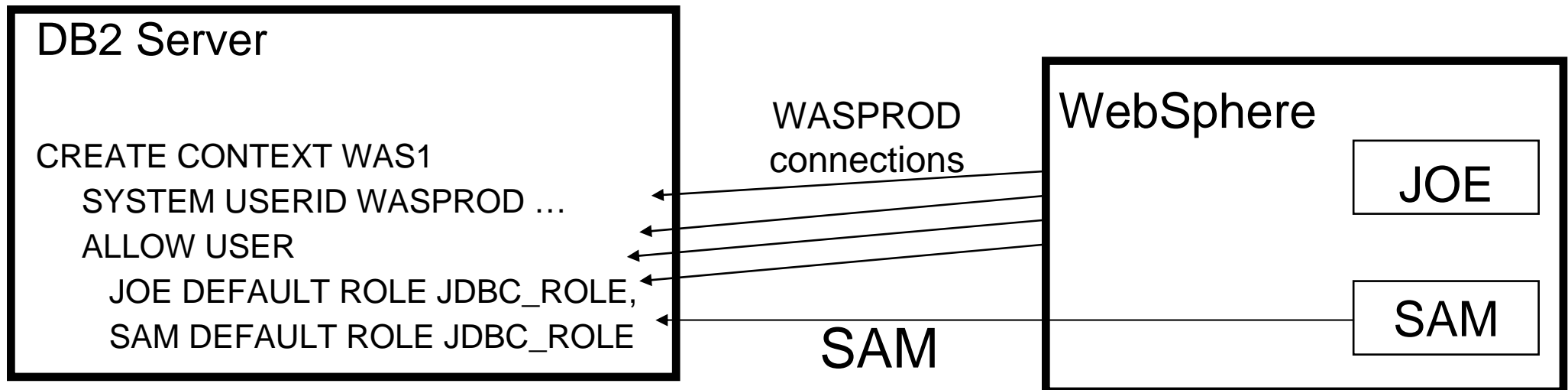
# Example 1: ROLES and Trusted Context used to Secure App Servers

- Most existing application servers connect to DB2 using userid/password pairs:
  - Significant exposure if someone steals the userid/password!!!
- Trusted Context and ROLES can be used to limit exposure:
  - GRANTS to SAP\_ROLE can be restricted so that they are only valid when used by a valid SAP app server IP address
- No change required to the code in the application server



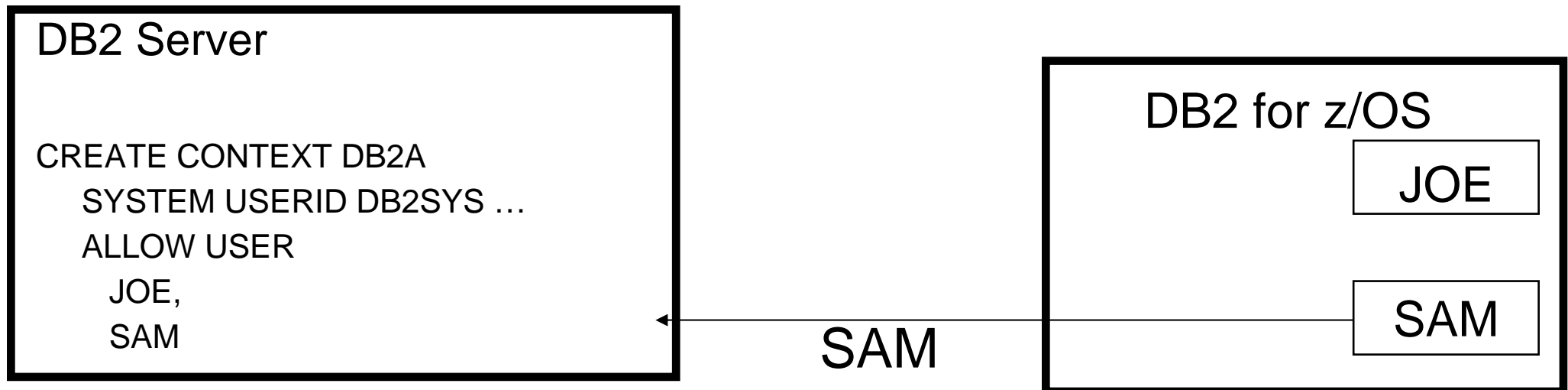
# Example 2: ROLES and Trusted Context for Dynamic SQL Auditing

- Better auditing controls:
  - GRANT dynamic SQL privileges to a ROLE
  - End user identity can be delegated directly to DB2 without granting dynamic SQL privileges directly to the end user
  - End user passwords can be optional.
  - No added complexity for administration of GRANTS, while retaining the ability to audit the end user's identity!!!



# Example 3: ROLES and Trusted Context for Already-Verified DRDA

- Can be used to establish already-verified TCP/IP connections:
  - Improves ability to replace SNA connections with TCP/IP
  - Communication Database is used to identify trusted connections and specify “system userid” for the Trusted Context
  - End user identity is automatically propagated from one DB2 system to the other.



## Example 4: ROLES and Trusted Context to Secure DBA Activities

- Many customers are concerned about DBA access to sensitive customer data. DB2 v9 can help by enabling an auditable DBA process:
  1. Grant DBA privileges to a ROLE
  2. Start audit trace for that ROLE
  3. When a DBA needs to perform a system change:
    - Use Trusted Context to assign DBA ROLE to person
    - DBA is given request and performs activity
    - Revoke Trusted Context
  4. Have another person review the audit trace





# Auditing: DB2 Trace Filtering

- New filtering capabilities for `–START TRACE` that `INCLUDE` or `EXCLUDE` based on these keywords:
  - `USERID` -- client userid
  - `WRKSTN` -- client workstation name
  - `APPNAME` -- client application name
  - `PKGLOC` -- package `LOCATION` name
  - `PKGCOL` -- package `COLLECTION` name
  - `PKGPROG` -- `PACKAGE` name
  - `CONNID` -- connection ID
  - `CORRID` -- correlation ID
  - `ROLE` – end user's database `ROLE`



# Volume-based COPY/RECOVER

- FlashCopy technology used to capture entire content of disk volumes
- RECOVER modified to enable object-level recovery from volume FlashCopy
- Eliminates labor associated with setting up COPY jobs for each database / table space



# Converged TEMP Space

- Single source for all temporary space in DB2, replacing: DSNDB07, temp databases, work file database
- Access is virtualized for small amounts of data, eliminating cost of work file creation (reduced CPU and I/O)
- Supports 4K and 32K page sizes, with automatic selection of the appropriate page size



# DDL Porting Improvements

- Automatic selection of DATABASE and TABLESPACE when DDL omits these keywords
- Automatic CREATE of UNIQUE index for PRIMARY KEY
- Deprecated simple table space, default to segmented



# DB2 V9 Themes

- ❑ **Enable high-volume transaction processing for next wave of Web applications**
- ❑ **Extend the lead in transaction processing availability, scalability and performance**
- ❑ **Reduce cost of ownership and zSeries-specific skill needs**
- ❑ **Improve data warehousing and OLTP reporting**



# Data Warehousing, Reporting and Optimizer Improvements

- SQL enhancements: INTERSECT, EXCEPT, RANK, caseless comparisons, cultural sort, ...
- Index improvements: index on expression
- Improved Optimization statistics: Histogram
- Optimization techniques
  - Cross query block optimization
  - Generalize sparse index & in-memory data cache method
  - Dynamic Index ANDing for Star Schema
- Analysis: instrumentation & Optimization Support

