



IBM Software Group

# DB2 UDB 9, the best Database for SAP

Günter Weber, [weberg@de.ibm.com](mailto:weberg@de.ibm.com)

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™

© 2005 IBM Corporation

# 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.



# Agenda

- DB2 and SAP
- Optional:
  - ▶ XML Support
  - ▶ Maintenance
  - ▶ Partitioning
  - ▶ Security
  - ▶ SQL Features
  - ▶ Data Compression



IBM Software Group

# DB2 and SAP

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™

© 2005 IBM Corporation

# DB2 und SAP Ankündigungen

April 2005 – [www.ibm.com/solutions/sap/us/en/news](http://www.ibm.com/solutions/sap/us/en/news)

## SAP and IBM Expand Partnership, Introduce Optimized DB2 Database

**COPENHAGEN, Denmark — April 26, 2005** — SAP AG (NYSE: SAP) and IBM (NYSE: IBM) today unveiled an optimized version of the **industry-leading DB2 Universal Database to help customers ease configuration, enhance performance and increase availability of their SAP® solutions running on DB2**. The announcement was made at SAPPHIRE® '05, being held in Copenhagen, Denmark, April 26-28

Mai 2006 - [www.sap.com/germany/company/press/archive/press\\_show.epx?ID=3568](http://www.sap.com/germany/company/press/archive/press_show.epx?ID=3568)

## Neue Version von DB2 für mySAP All-in-One optimiert

**SAP und IBM haben auf der internationalen Kundenkonferenz SAPPHIRE '06 in Orlando eine weltweite Kooperation angekündigt, um mittelständischen Unternehmen eine noch engere Integration zwischen mySAP All-in-One-Lösungen und der nächsten Version von DB2 – Codename Viper – zu bieten.**

**Aufgrund der engen Integration der mySAP All-in-One-Lösungen mit der neuen IBM-Datenbankversion empfiehlt SAP ihren Mittelstandskunden explizit den Einsatz von DB2 Viper (V9).**

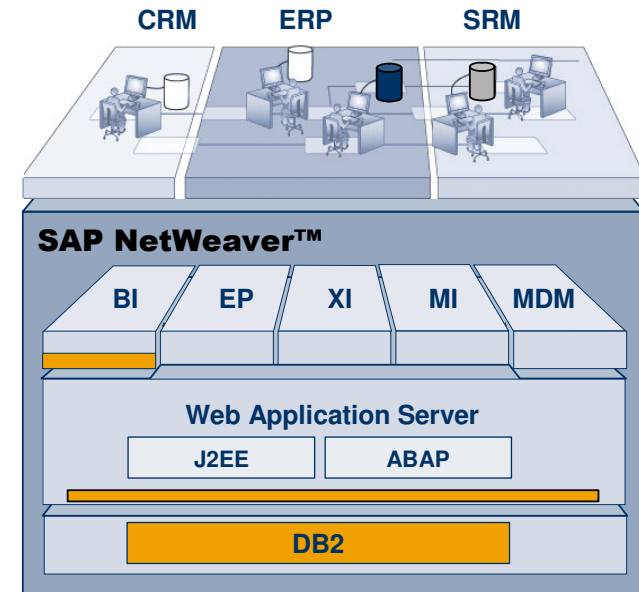
SAP Pinnacle Awards Recognize IBM DB2  
**“2005 Best Technology Solution”**



## SAP Anforderungen an die Datenbank

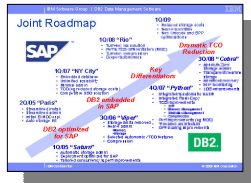
- Alle Anwendungen nutzen die **Plattform Abstraktion**, die vom SAP Web Application Server zur Verfügung gestellt wird
  - ▶ Alle SAP NetWeaver Komponenten sind Datenbank-unabhängig (**mit Ausnahme von SAP BI**)
  - ▶ SAP Anwendungen wissen nicht auf welcher Datenbank sie laufen
  
- Die **optimale Datenbank für SAP Kunden** bietet folgendes
 

<ul style="list-style-type: none"> <li>▶ beste Integration mit SAP</li> <li>▶ überlegene Performance</li> <li>▶ unlimitierte Skalierbarkeit</li> </ul>	<ul style="list-style-type: none"> <li>▶ permanente Innovation</li> <li>▶ beste Produktqualität</li> <li>▶ niedrigste Kosten</li> </ul>
--	---



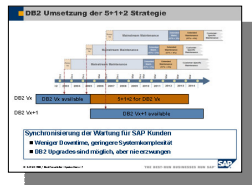
Die „**DB2 optimiert für SAP**“ Initiative basiert auf diesen Eckpfeilern

# Drei Bereiche der Optimierung von SAP auf DB2



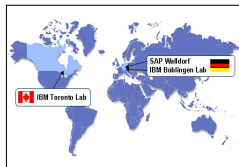
## Technische Innovation

*Gemeinsame Technologie-Roadmap über 2008 hinaus.*



## Produktintegration

*Ein Produkt, eine Wartungsstrategie, integrierter Service*



## Partnerschaft

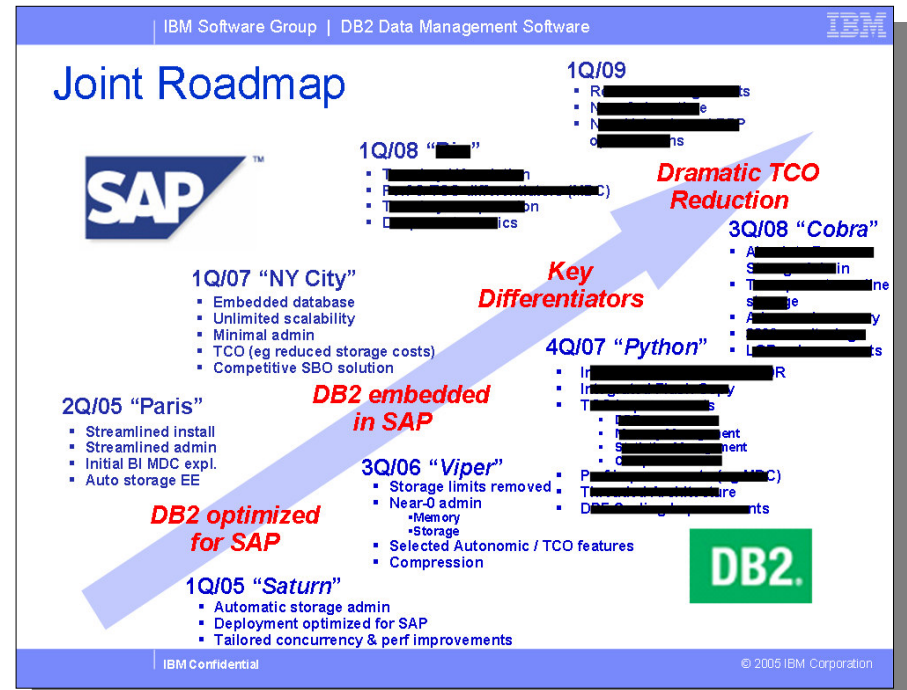
*Gemeinsame SAP + IBM Teams auf allen Ebenen*

**Diese Art der Partnerschaft existiert nur zwischen SAP und IBM**

# Technische Innovation

- Jedes „Optimized for SAP“ DB2 Release ist abgestimmt mit dem SAP Produktplan.
- Das zugehörige SAP Release nutzt alle DB2 Innovationen voll aus.
- Rückwärtsfreigabe: Jedes „Optimized for SAP“ DB2 Release wird zeitnah für alle noch in der 5+1+2 Wartung befindlichen SAP Releases freigegeben.

▶ Jeder SAP Kunde kann direkt von den Innovationen profitieren

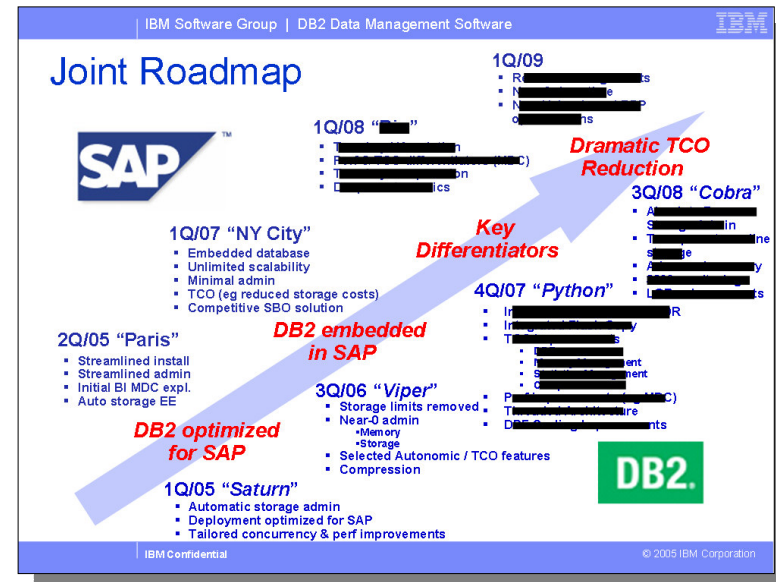


Ziel: „Administrationsfreie Datenbank mit uneingeschränkter Skalierbarkeit und Verfügbarkeit“



# Status der „Optimized for SAP“ Produkt-Releases

- **DB2 V8.2.2 verfügbar seit April 2005**
- **DB2 9 (Viper) verfügbar seit 28. Juli 2006 und für SAP freigegeben seit 30. August 2006**
- **DB2 Python Planung abgeschlossen, Shipment geplant für 2007**
- **DB2 Cobra Planung ist im Gang**

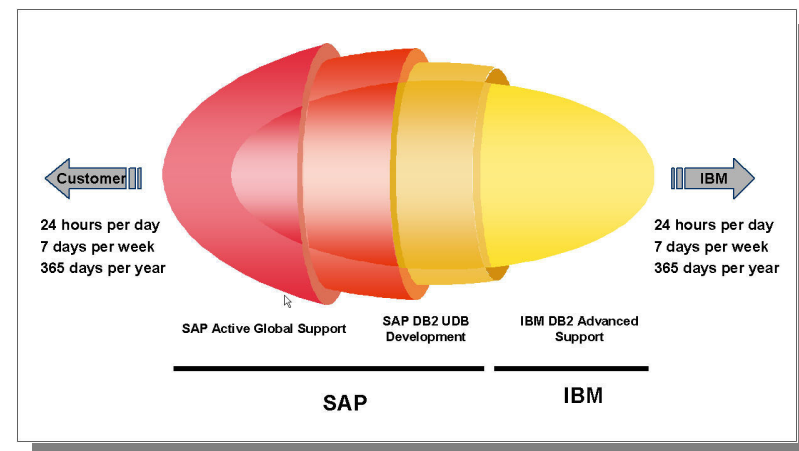
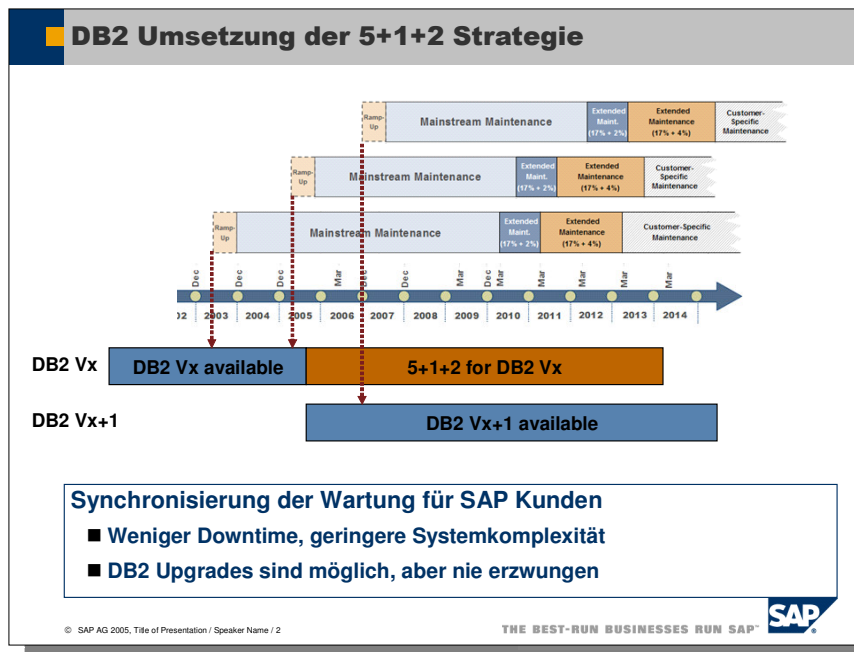


- **Gemeinsame Produktdefinition**
- **Obligatorischer Design-Review und Genehmigung durch SAP für alle SAP-relevanten DB2 Features bevor die Implementierung beginnt**
- **Entwicklungsbegleitendes Testen und Verifikation über den gesamten DB2 Entwicklungszyklus – lange vor der Freigabe an Kunden**

# Produktintegration

## SAP auf DB2 ist ein vollständig integriertes Produkt

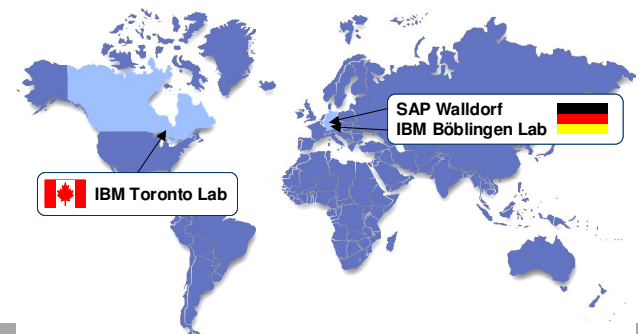
- ▶ Alle Kunden erhalten **Support aus einer Hand** über SAP
- ▶ IBM und SAP **validieren** alle DB2 Releases und Fix-Packs um höchste Qualität sicherzustellen (Integration Centre Toronto)
- ▶ Neue DB2 Releases sind bei Verfügbarkeit sofort (oder wenige Tage später) für SAP unterstützt
- ▶ **DB2 Wartungszyklen abgestimmt** mit der **SAP 5+1+2** Wartungsstrategie. **Keine erzwungenen Datenbank-Upgrades** aufgrund von DB2 Wartungsende während SAP Wartungsperiode



# Partnerschaft

**SAP auf DB2 ist ein SAP+IBM Gemeinschaftsprojekt seit 1993:**

- ▶ **Gemeinsames SAP+IBM *Entwicklungsteam* in Walldorf**
  - Entwicklung von SAP Code
  - Entwicklung von DB2 Code für SAP Line Items
  - Development Support und Zusammenarbeit mit DB2 Service
- ▶ **Gemeinsames *SAP IBM Integration Center* in Toronto**
  - Integration neuer DB2 Releases mit existierenden SAP Releases
  - DB2 QA – jeder DB2 Fix Pack und jedes DB2 Release wird intensiv mit SAP Software getestet, bevor die Freigabe für Kunden erteilt wird
  - Development Support und Zusammenarbeit mit DB2 Service
- ▶ **Intensive Zusammenarbeit mit *DB2 Entwicklung* Toronto**
  - Regelmäßige Abstimmung
  - Gemeinsame Planung
  - Design Review



# DB2 9 (Viper) – Überblick neuer Funktionen

## ■ Vereinfachte Administration

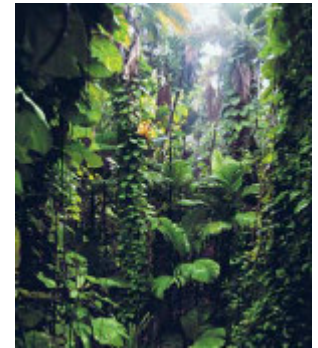
- ▶ Silent-Install – gleichzeitige Installation von SAP und DB2
- ▶ Installation beliebig vieler DB2 Kopien auf einem Rechner für einfachere Upgrades und Einspielen von Patches
- ▶ Thin-Client – keine Notwendigkeit für Installation des DB2 Clients auf SAP Applikationsservern

## ■ Verbesserte Ressourcen-Nutzung und erhöhte Performance

- ▶ Selbst-optimierende Hauptspeicherverwaltung
- ▶ Autonome Plattenspeicherverwaltung auch bei verteilten Datenbanken (mit DB2 DPF)
- ▶ Datenkomprimierung - Reduktion der Plattenkapazität um bis zu 75% und schnellere Abfrageprozesse
- ▶ MDC Erweiterungen

## ■ Hochverfügbarkeit

- ▶ Verbesserter Restore-Prozess (Wiederanlauf nach Unterbrechung)
- ▶ Verbesserte Diagnose-Tools für noch schnellere Problemerkennung (Systemkonfiguration und Performanceengpässe)
- ▶ Vordefinierte HA Lösung basierend auf Tivoli System Automation (inkl. kostenfreier TSA-Lizenz)



## DB2 – Führend bei 3-Tier SAP SD Benchmark\*

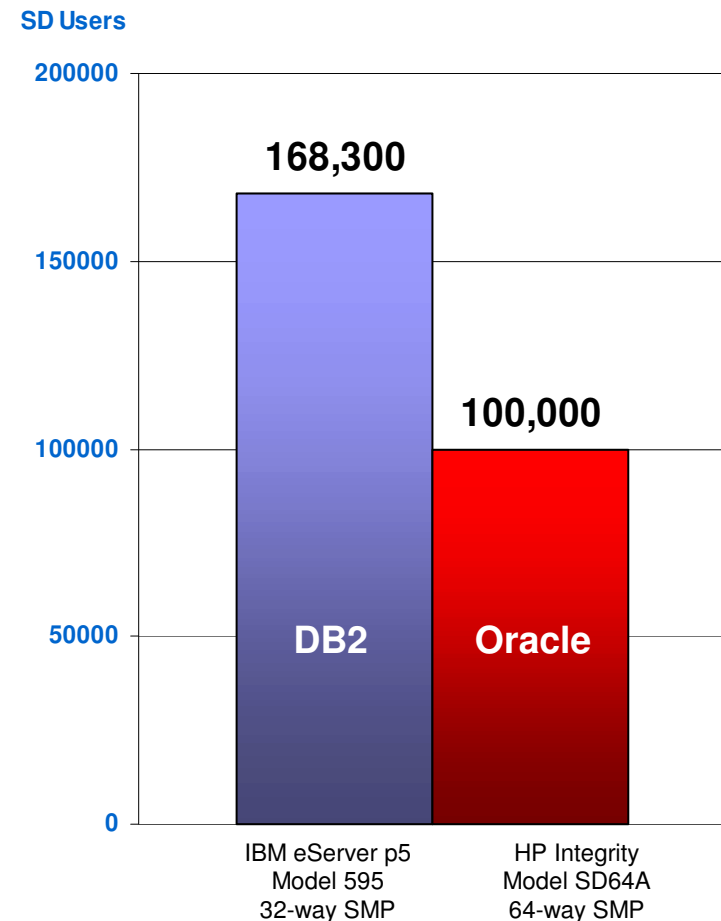
**Benchmark repräsentiert eine 3-Tier SAP R/3 Umgebung, in der der Datenbank-Server auf einem separaten Rechner betrieben wird**

- Zeigt das Leistungsvermögen von DB2 durch Nachweis der Skalierbarkeit bis zu 168,300 SAP SD User

**Ergebnis: 68% mehr Leistung mit DB2**

- DB2 auf IBM pSeries 32-Wege p5 595
- Oracle auf HP Integrity Model SD64A, 64-Wege SMP

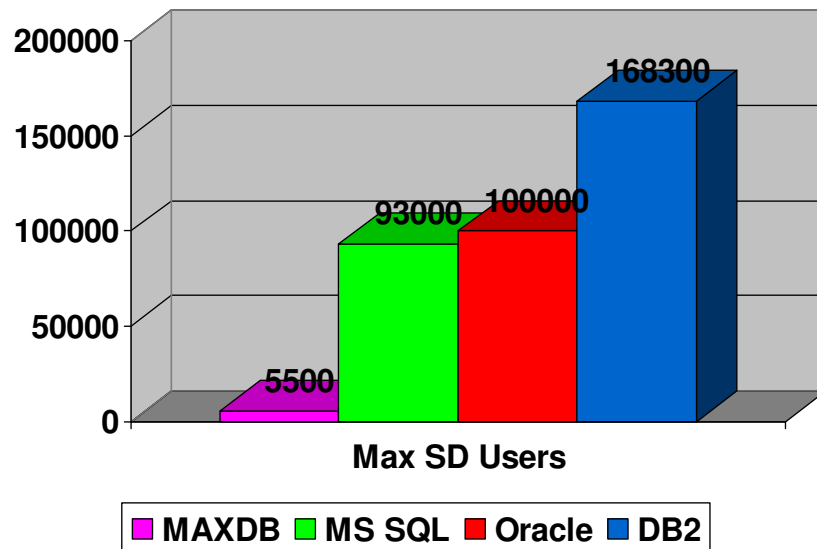
**SAP Sales & Distribution,  
R/3 Release 4.7, 3-Tier Benchmark  
Best Result by Database Vendor**



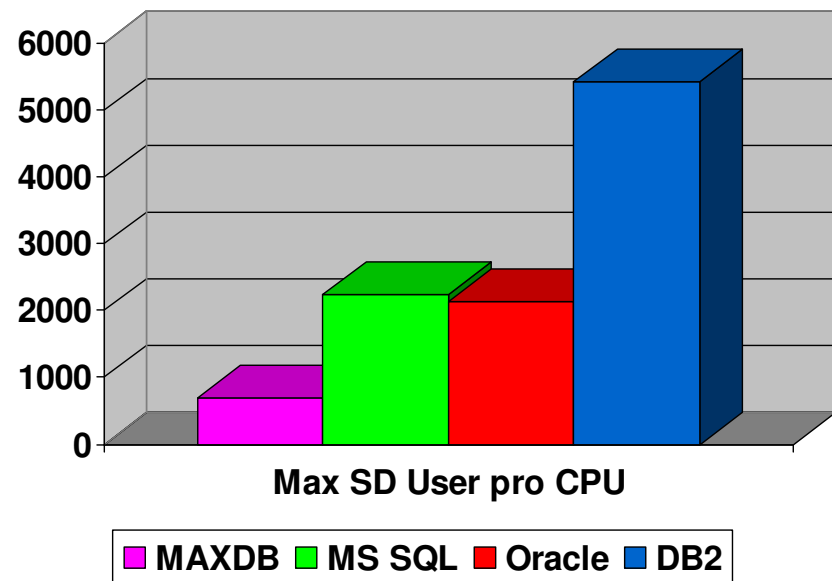
\*Results referenced are current as of December 15, 2005.  
For the latest SAP results, visit <http://www.sap.com/benchmark>.

# SAP SD 3-Tier Benchmark Ergebnisse

## Maximal erreichte User-Zahl



## Maximal erreichte User-Zahl pro CPU

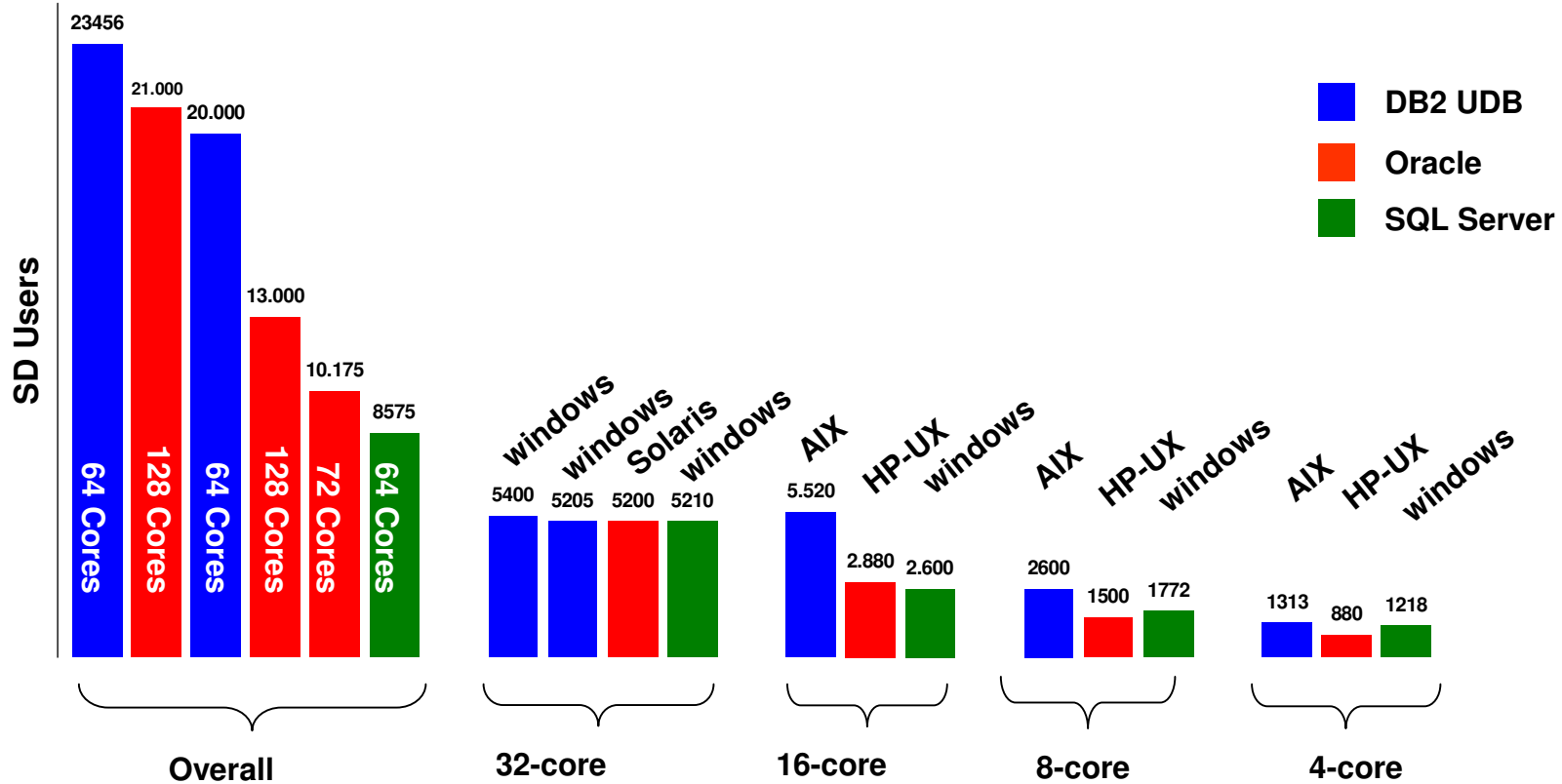


Quelle: [www.sap.com/benchmark](http://www.sap.com/benchmark) - Status: 2. Juni 2006

# SAP 2-tier SD Benchmark-Ergebnisse



★ **!! DB2 9 führend bei SAP 2-tier SD-Benchmark !!**



Results current as of July 01, 2006, Check <http://www.sap.com/benchmark> for latest results

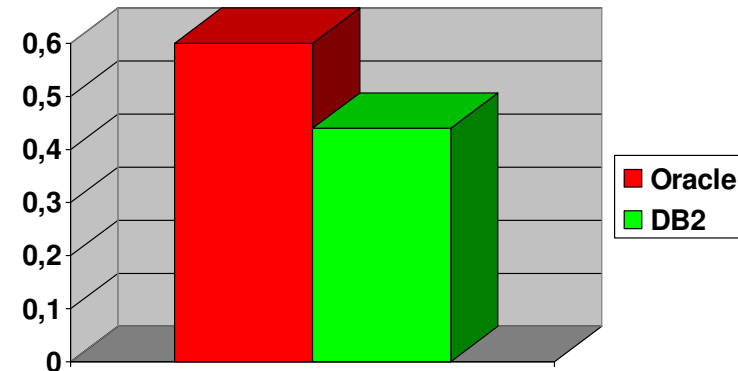
# SAP Performance-Vergleich Oracle vs. DB2

## Beispiel 1 – Kunde rku.it

Messungen vor und nach Migration von Oracle zu DB2 (auf gleicher pSeries Hardware):

- DB2 um 40% schneller bei Dialog
- DB2 um 40% schneller bei Batch

Dialog: Antwortzeiten

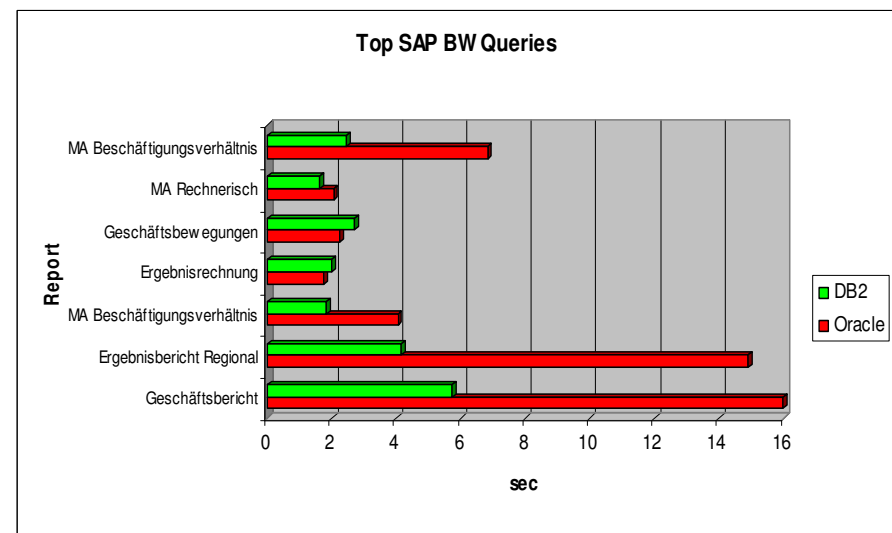


## Beispiel 2 – Grosses deutsches Unternehmen

Testcase: Laufzeit-Messungen von verschiedenen SAP Business Warehouse Queries mit Transaktion RSRT

- DB2 im Schnitt 2,42 mal schneller
- DB2 bis zu 3,6 mal schneller

Top SAP BW Queries

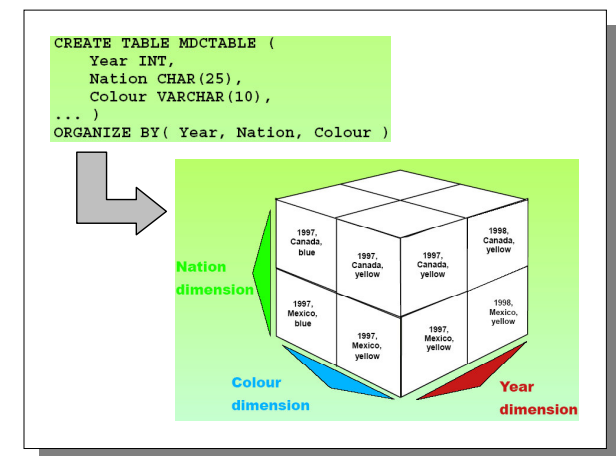
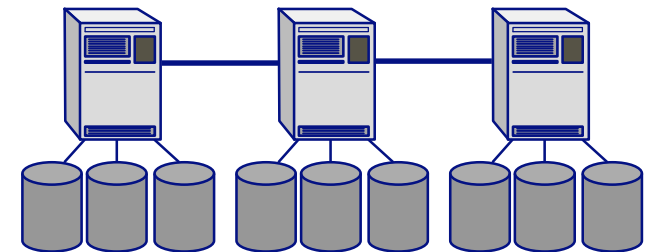




# DB2 Alleinstellungsmerkmale für SAP BI

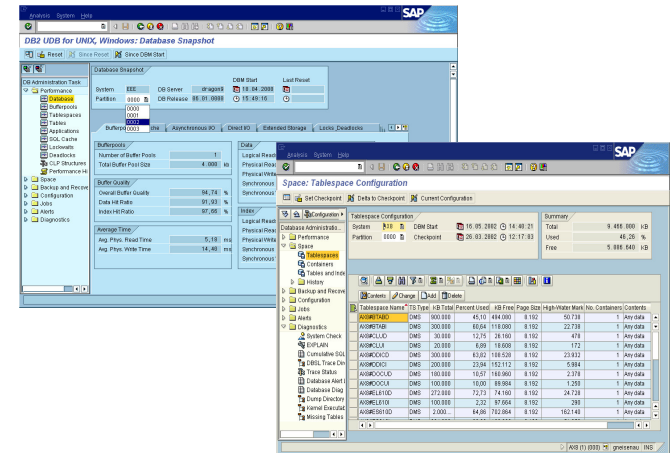
- DB2 "Database Partitioning Feature" (DPF)
  - ▶ Lineare Scale-Out Skalierbarkeit durch Shared-Nothing Architektur (parallele Datenbank mit separate Daten für jeden DB2 Knoten)
  - ▶ Ausgereifte Technologie – in fast allen grossen SAP BI Installationen mit DB2 wird DPF genutzt
  - ▶ Unterstützt seit SAP BW 2.0
  - ▶ Nur für DB2 verfügbar
  
- DB2 "MultiDimensional Clustering" (MDC)
  - ▶ Performance Steigerung für SAP BI Queries bis zum Faktor 8 durch Multi-Dimensional Clustering
  - ▶ Schnelle roll-in / roll-out Möglichkeit
  - ▶ Kein Administrationsaufwand
  - ▶ Nur für DB2 verfügbar

Database Server Cluster



## „State of the art“ Datenbank-Administration mit DB2

- SAP WebAS enthält integrierte Administrations- und Monitoring-Tools für DB2
  - ▶ Voller Multi-Partition Unterstützung
  - ▶ Graphische Oberfläche – umfassend und einfach
- Online-Backup & Online-Datenbank- & Tabellen-Reorganisation



### Kundenaussagen:

- „The actual number that I use for reduction of support personnel is **35% less with DB2 versus Oracle.**“

*Jim Claxton (CSC, Betreiber für SAP Landschaft von Dupont)*

- „Wir waren überrascht, wie schnell wir DB2 beherrschten. Neben der einfachen DB2 Benutzeroberfläche war dafür auch die wertvolle Unterstützung durch das DB2/SAP Support Team der IBM verantwortlich.“

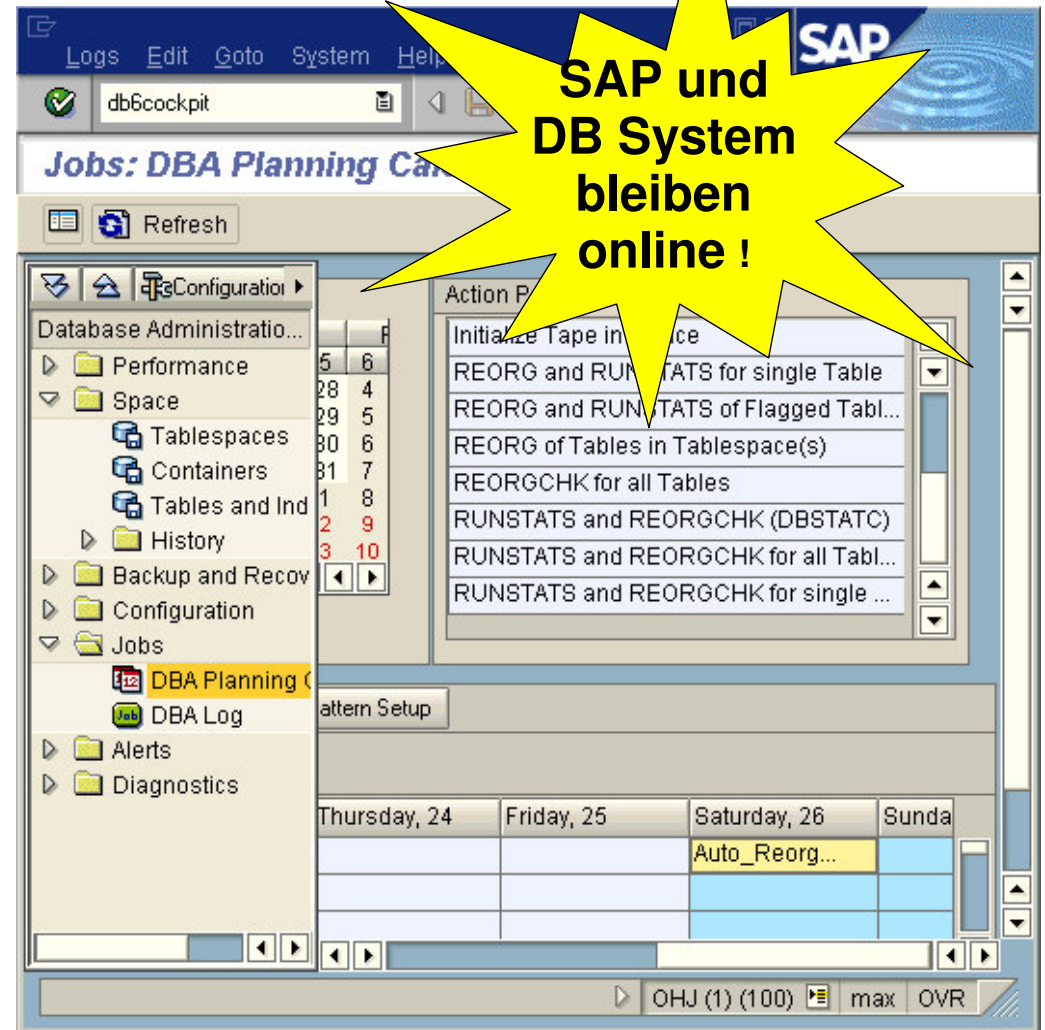
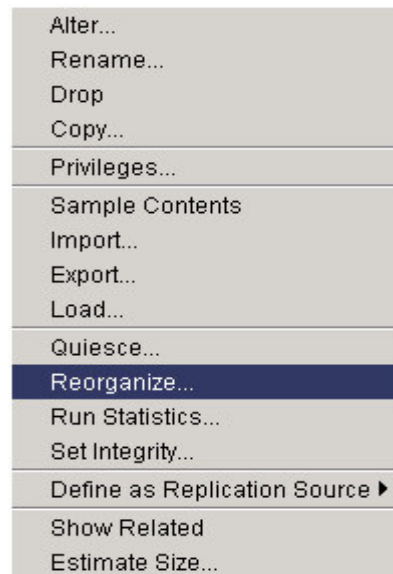
*Manfred Leistner, E.G.O. (migriert von Oracle nach DB2)*

## Konfiguration und Administration

- DB2 pre-tuned for SAP workloads (DB2\_WORKLOAD=SAP)
- Significant enhancements into DB2's self-tuning capabilities
  - ▶ Self-managing storage
  - ▶ Self-tuning configuration
  - ▶ Automatic REORG, RUNSTATS and BACKUP
- Unlimited growth and capacity
- Significant savings in disk space usage (approx. 70%) and I/O bandwidth through row compression
- Ultimately the need for tuning of any DB2 configuration parameter will go away – they will all be just AUTOMATIC

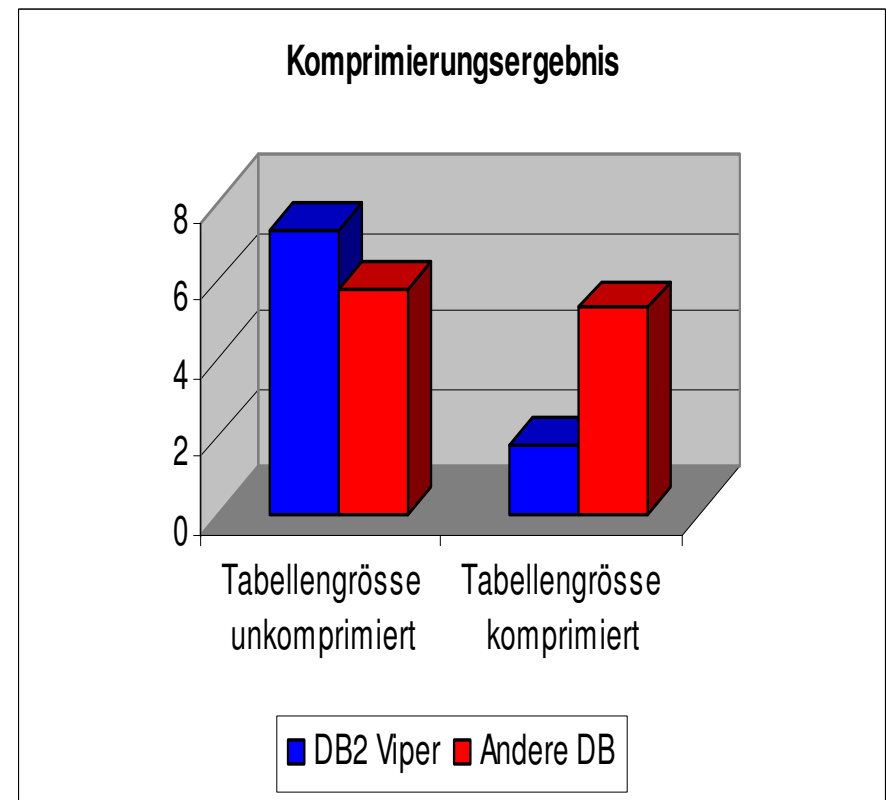
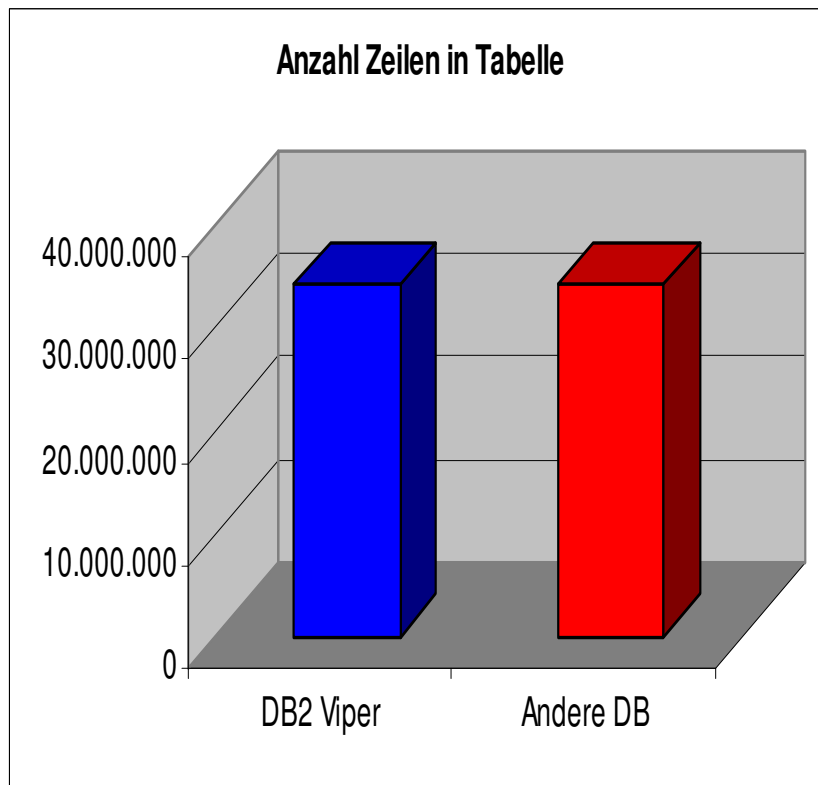
# DB2 Online Tabellen Reorganization

- Online + Einfach
- Unter der Kontrolle von DB2 DBMS & SAP DB6COCKPIT



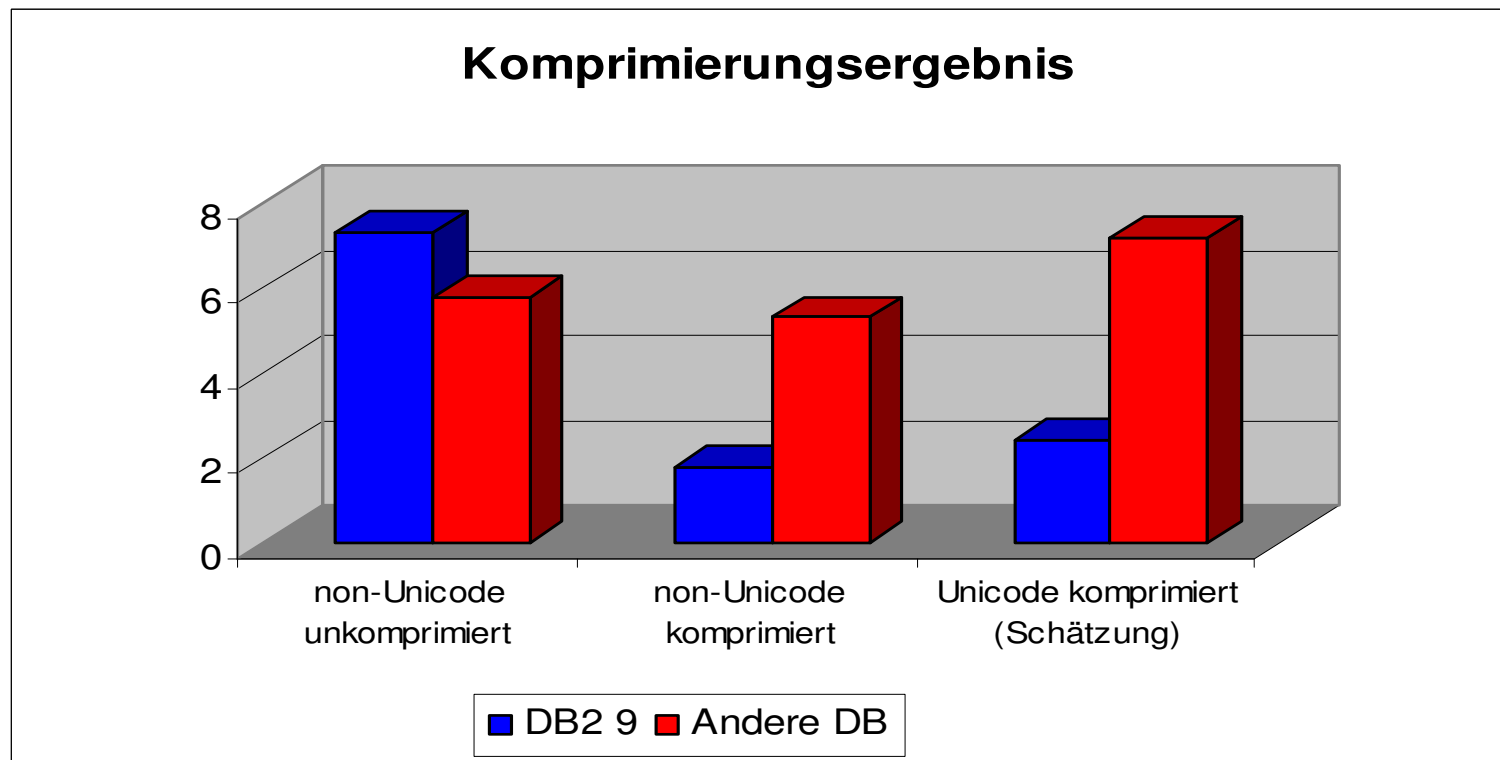
# DB2 Datenkomprimierung – Labor-Test

- Ergebnisse bei Komprimierung der mySAP Tabelle STXH
  - DB2 Komprimierung: 75%
  - Oracle Komprimierung 10%



# DB2 Datenkomprimierung – Labor-Test

- Komprimierungsergebnis mySAP Tabelle STXH (34 Millionen Zeilen)
  - Komprimierungsrate: 75% bei DB2, 10% bei Oracle
  - Platzbedarf nach Komprimierung: DB2 benötigt ein Drittel der Oracle-Grösse



\*) Schätzung für Unicode-Wachstum basiert auf den Aussagen der SAP bzgl. durchschnittlichen Wachstum bei Verwendung von UTF-8

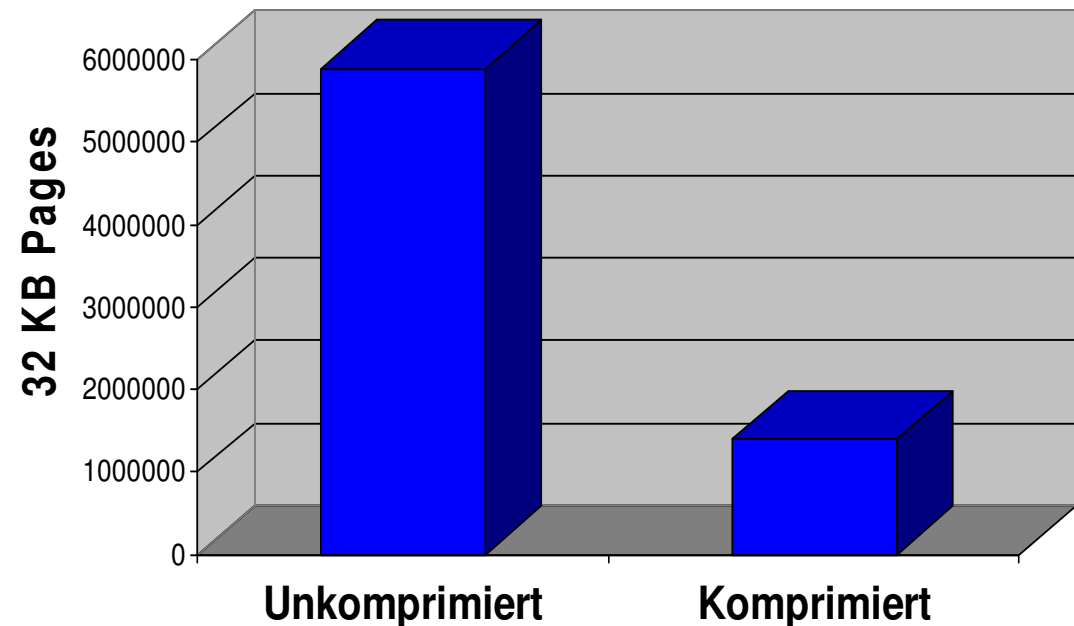
# DB2 Datenkomprimierung - Kundentest

- Konkretes Beispiel bei einem DB2 9 Beta Kunden:
  - ▶ **76% geringeres Datenvolumen bei komprimierter Tabelle**
    - Signifikante Reduktion bei Storage Kosten realisierbar, z.B. 70 Euro pro "Managed GB" und Jahr
  - ▶ Query Performance Verbesserung von ca. 40% aufgrund geringerer Anzahl von I/Os
  - ▶ Erwartete Reduktion des Datenvolumens der gesamten Datenbank bei ca. 40-50%

<u>Compression Type</u>	<u>32KB Page Count</u>	<u>Space Required on Disk</u>
Keine Komprimierung	5893888	179.9GB
Zeilen-Komprimierung	1392446	42.5GB

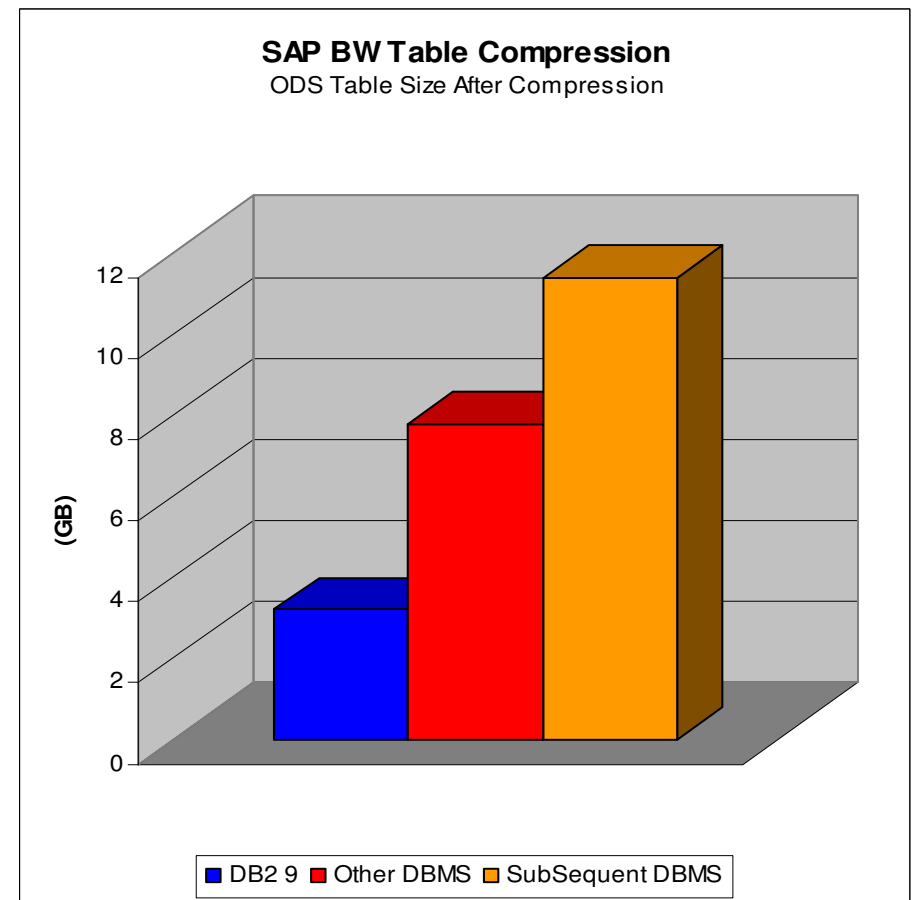
**% Pages Saved: 76.4%**

**T1 Komprimierung - Tabelle unkomprimiert 179,9 GB**



## DB2 9 Komprimierung für SAP BW Tabelle - Labortest

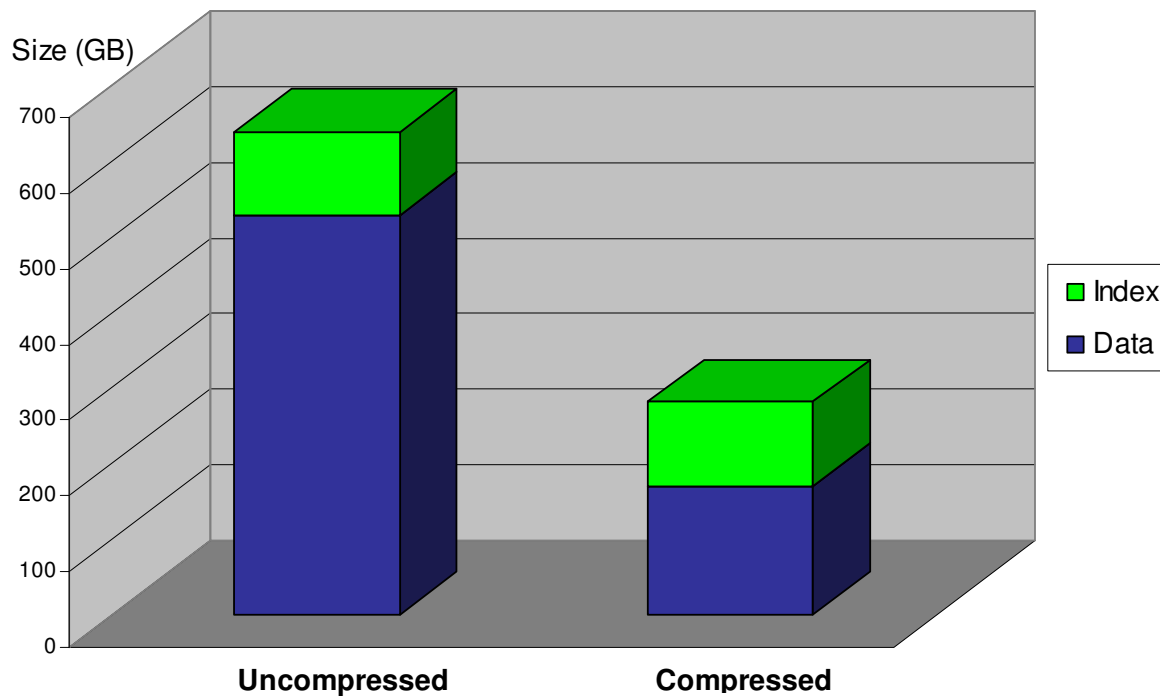
- Grösse einer FACT Tabelle mit 19 Millionen Zeilen nach Komprimierung (in GB)
  - ▶ **DB2 9:** 3,2
  - ▶ *Nicht-IBM DB #1* 7,8
  - ▶ *Nicht-IBM DB #2* 11,4
- DB2 9 bietet überlegene Komprimierung
  - ▶ 60% kleiner als *Nicht-IBM DB #1*
  - ▶ 72% kleiner als *Nicht-IBM DB #2*





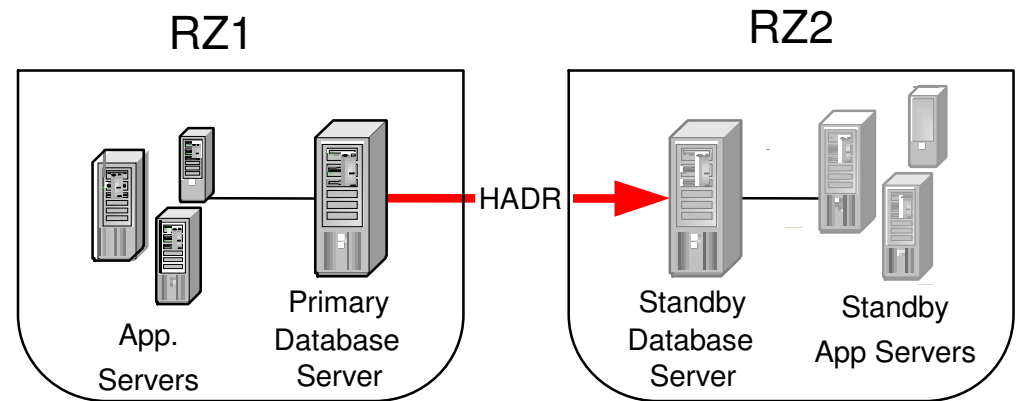
## DB2 9 Komprimierung – reale Kundenumgebung

- Betrachtung der 39 grössten Datentabellen und der zugehörigen Indextabellen
- Gesamt-Komprimierungs-Rate mit DB2 9: **56 %**

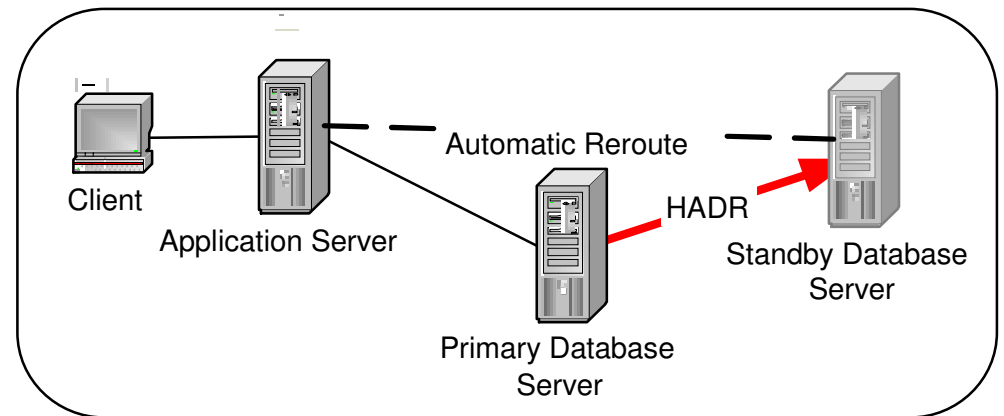


# High Availability Disaster Recovery (HADR)

- **Integrales DB2 UDB Feature für höchste Verfügbarkeit (keine Zusatzkosten)**
- Herausforderung
  - ▶ 24 x 7 Verfügbarkeit (auch im Katastrophenfall)
- Lösung
  - Offsite Disaster Recovery:
    - ▶ Ausfallsicherung ohne Datenverlust
  - Onsite Standby:
    - ▶ Einspielen von Software Updates ohne die Datenbank neu zu starten
- Vorteil
  - ▶ Tagesgeschäft läuft ohne Störung weiter
  - ▶ Einfach zu implementieren und zu verwalten
  - ▶ Automatic Client Reroute



Offsite Disaster Recovery



Onsite Standby

*Übernahmezeit von 11 Sekunden bei Test in SAP R/3 Umgebung mit 600 Benutzern !*

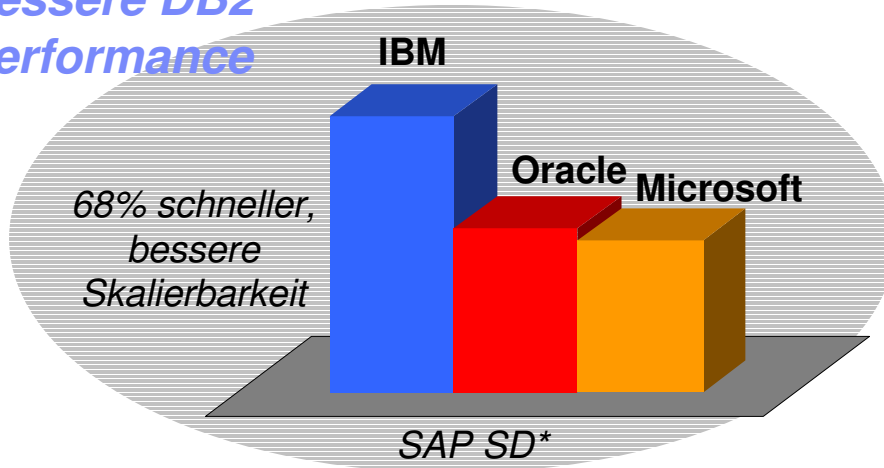
# Technologische Stärken im Überblick

- **DB2** – führende Performance bestätigt durch SAP Benchmarks
  - ▶ Cost-based Optimizer, Multiple Buffer Pools, Index Advisor, Configuration Advisor, ....
  - ▶ Neu in V9: Self-tuning Memory
- **DB2 HADR** – (High Availability & Disaster Recovery)
  - ▶ Hochverfügbarkeits- und Disaster-Recovery-Lösung mit Fail-over Zeiten im Sekundenbereich
  - ▶ Bestandteil der DB2 Enterprise Server Edition (ohne Zusatzkosten)
- **DB2 DPF** (Data Partitioning Feature)
  - ▶ Partitionierung der Datenbank innerhalb eines einzigen Servers oder über mehrere Server hinweg
  - ▶ Horizontale Skalierbarkeit für höchste SAP BI Performance
- **DB2 MDC** (Multi Dimensional Clustering)
  - ▶ Clustering für SAP NW BI Multi-Dimensionale Modelle (InfoCubes und Aggregate)
  - ▶ Verbesserte Star Schema Query Performance
  - ▶ Bestandteil der DB2 Enterprise Server Edition (ohne Zusatzkosten)
- **Neu in V9: DB2 Storage Optimization** (Compression)
  - ▶ Reduziert benötigte Storage Kapazität für SAP Datenbank um bis zu 75%
  - ▶ Performance Verbesserung um bis zu 40% aufgrund geringerer Datenmenge
- **State-of-the-Art Datenbank-Administration mit DB2**
  - ▶ Grafische Oberfläche, umfassend und einfach, integriert in SAP WebAS
  - ▶ Online DB-Administration (Backup, Reorg, ...)

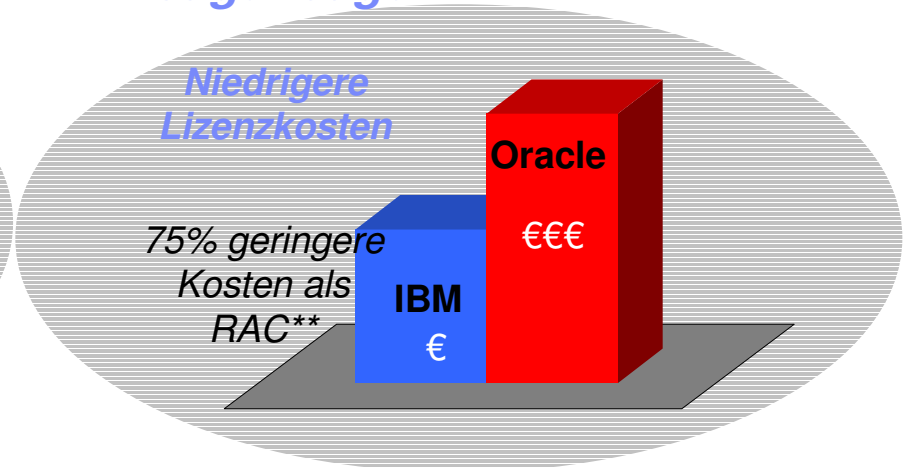


# DB2 Stärken im Vergleich

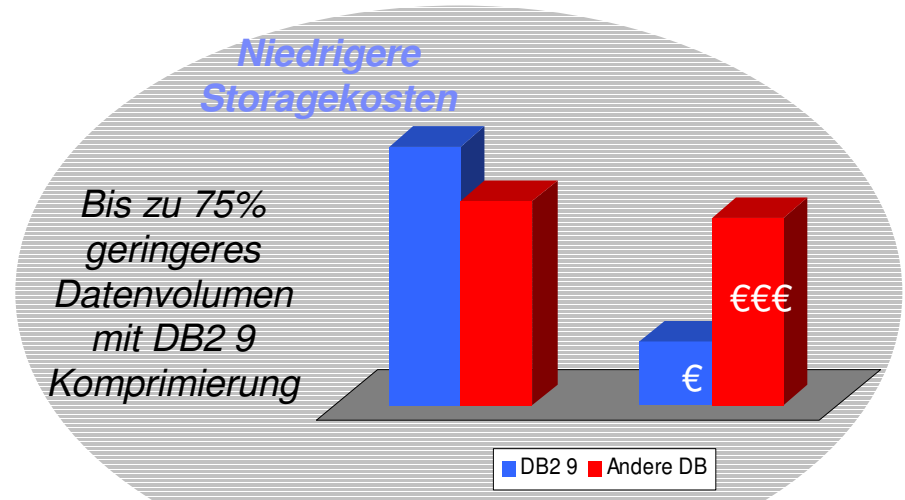
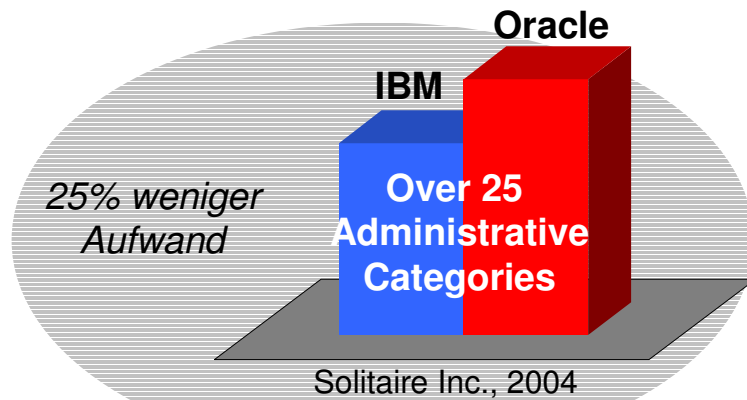
## Bessere DB2 Performance



## DB2 ist günstiger



## DB2 benötigt weniger Administrationsaufwand



\* SAP SD 3-Tier Benchmark, Results as of July 28, 2006. For further information see <http://www.sap.com/benchmark>, Sales and Distribution Benchmark SD, 3-Tier

\*\*As of 09/29/06. DB2 UDB and Oracle RAC Prices from SAP Price List (Section Database Prices).

# SAP DB2 Kunden auf Linux, Unix und Windows

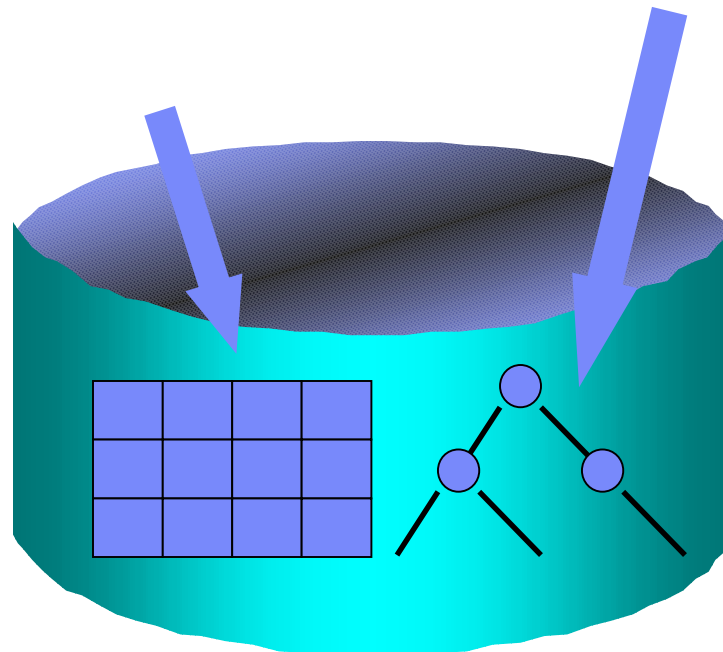


## Native XML Storage

- Must store XML in parsed hierarchical format (similar to the DOM representation of the XML infoset)

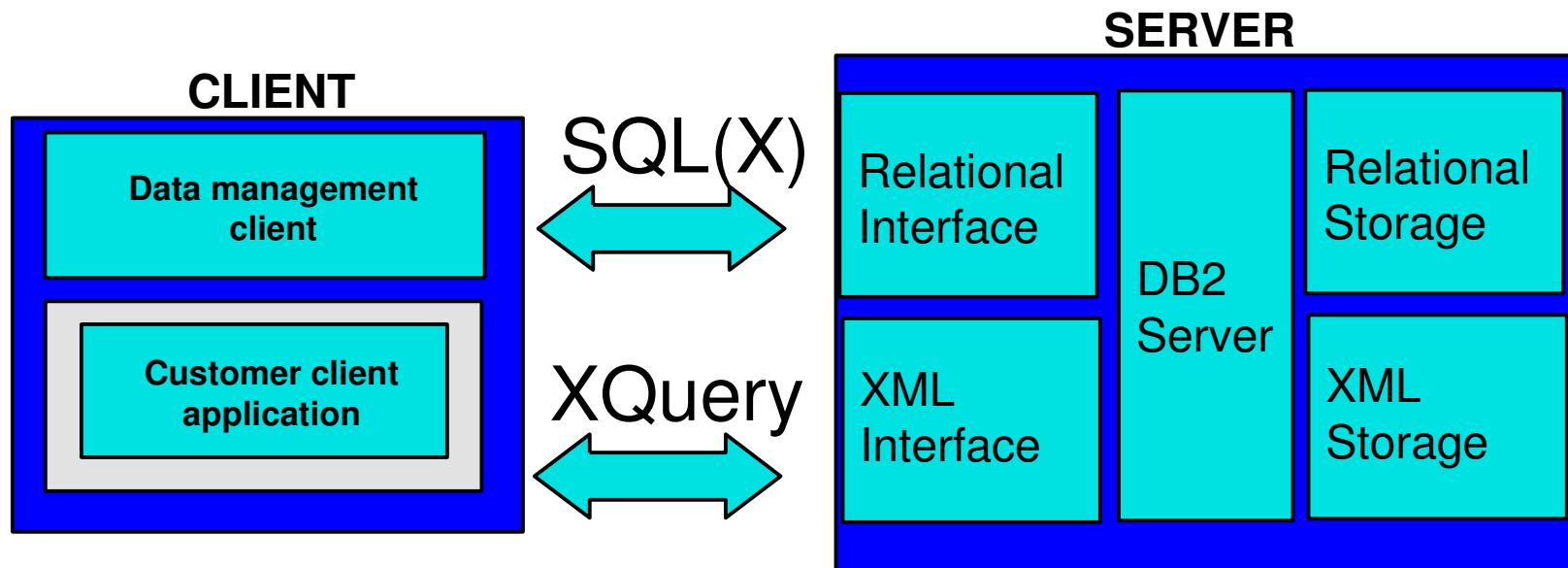
```
create table dept (deptID char(8), ..., deptdoc xml);
```

- Relational columns are stored in relational format (tables)
- XML columns are stored natively
- XML stored in UTF8
  - ▶ Unicode Database !



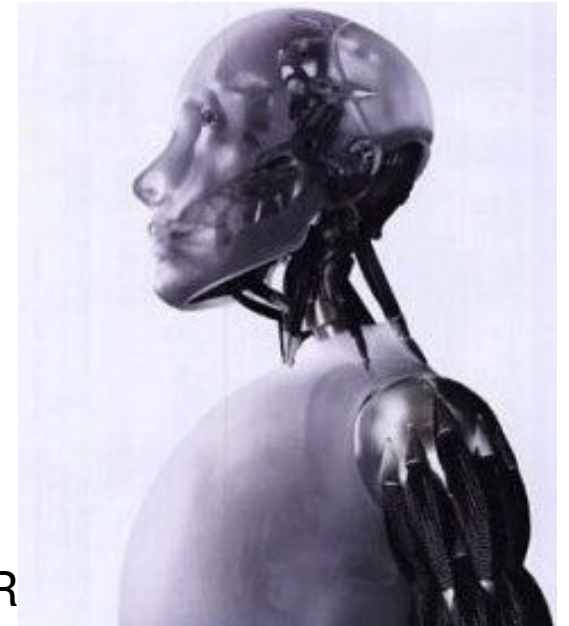
## XML in DB2

- XML capabilities "inside" the engine
- "Feels" relational and/or XML
- Both SQL flavor and fully XML flavor
- XML \*is\* DB2 internals - XML Extender becomes one with the data engine



## Automation Automatically!

- Enable many of the DB2 autonomic computing features by default.
- Examples:
  - ▶ Configuration Advisor (2 second tuning)
  - ▶ Adaptive Self Tuning Memory
  - ▶ Automatic data statistics collection.
- Better defaults for I/O Cleaners and I/O servers
  - ▶ Default for NUM\_IOSERVERS (3) and NUM\_IOCLEANER AUTOMATIC
  - ▶ Values calculated at database startup time
    - IOCLEANERS calculated based on number of CPUs and partitions
    - IOSERVERS calculated based on parallelism settings of all the tablespaces



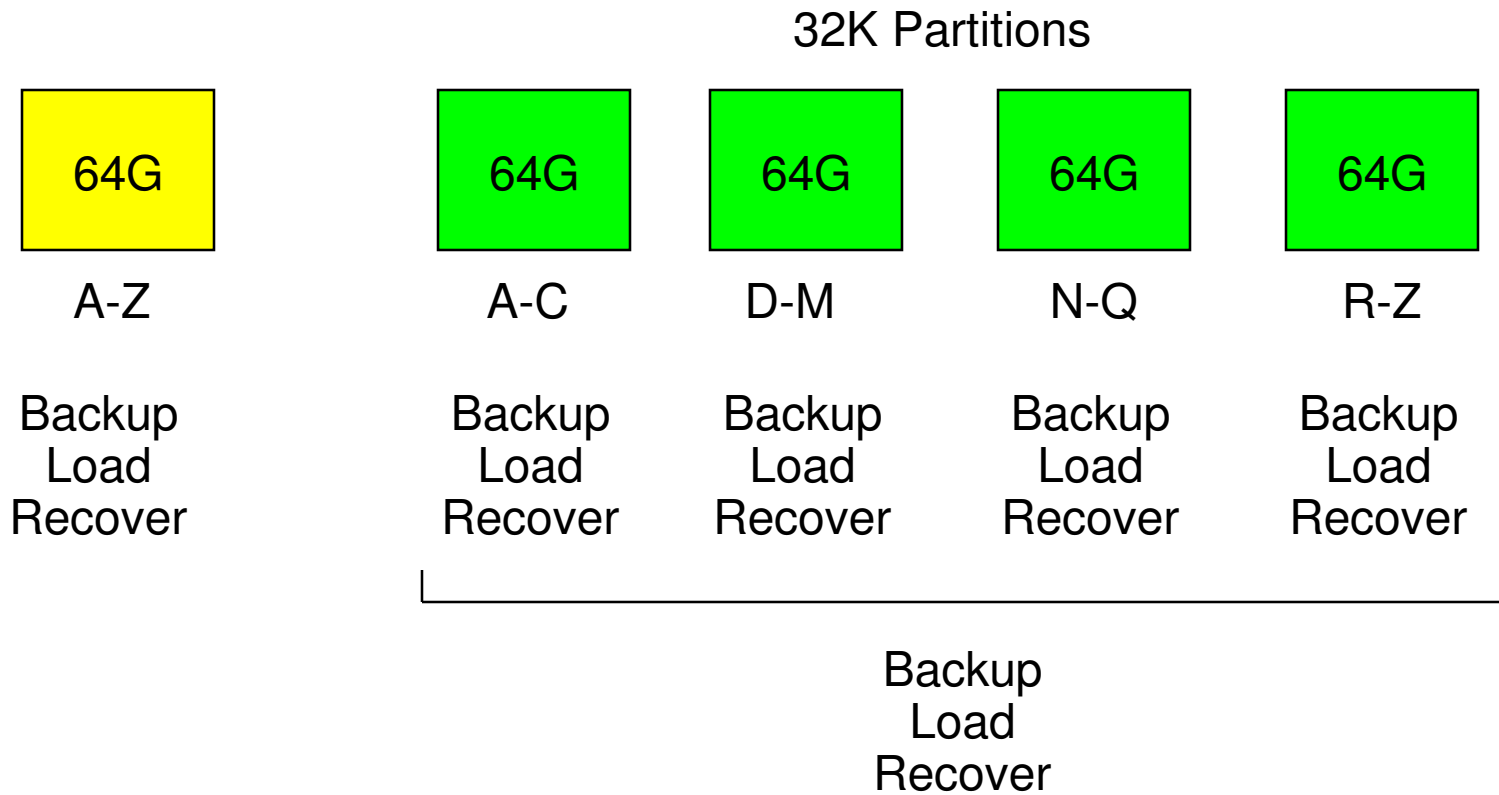


## Memory Tuning in Viper - STMM

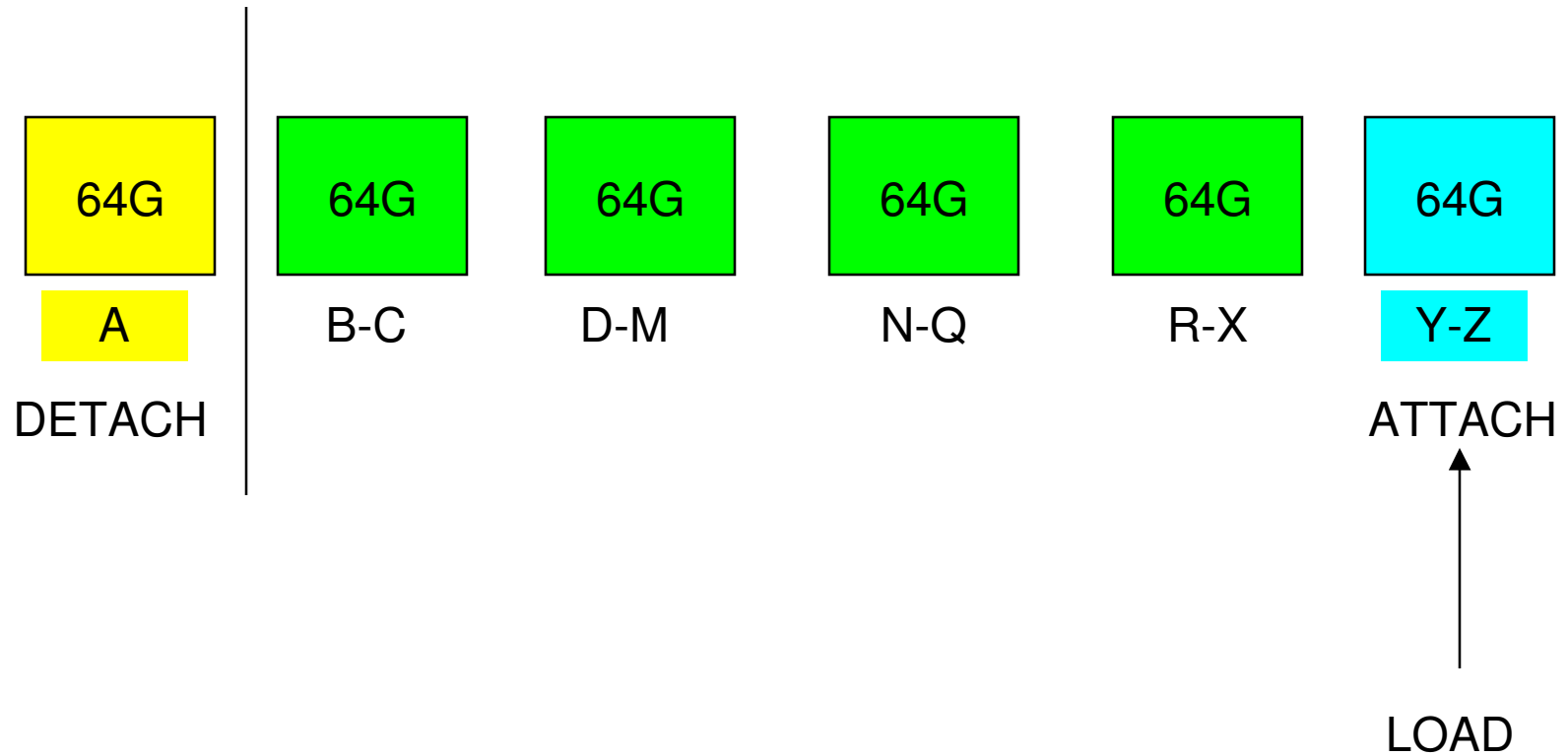
Viper will introduce a revolutionary memory tuning system called the “**Self Tuning Memory Manager**” (STMM)

- Works on main database memory parameters
  - ▶ Sort, locklist, package cache, buffer pools, and total database memory
- Hands-off online memory tuning
  - ▶ Requires no DBA intervention
- Senses the underlying workload and tunes the memory based on need
- Can adapt quickly to workload shifts that require memory redistribution
- Adapts tuning frequency based on workload
- Is able to tune multiple databases and instances on the same box at the same time
- Works in non-partitioned and in partitioned (DPF) environments

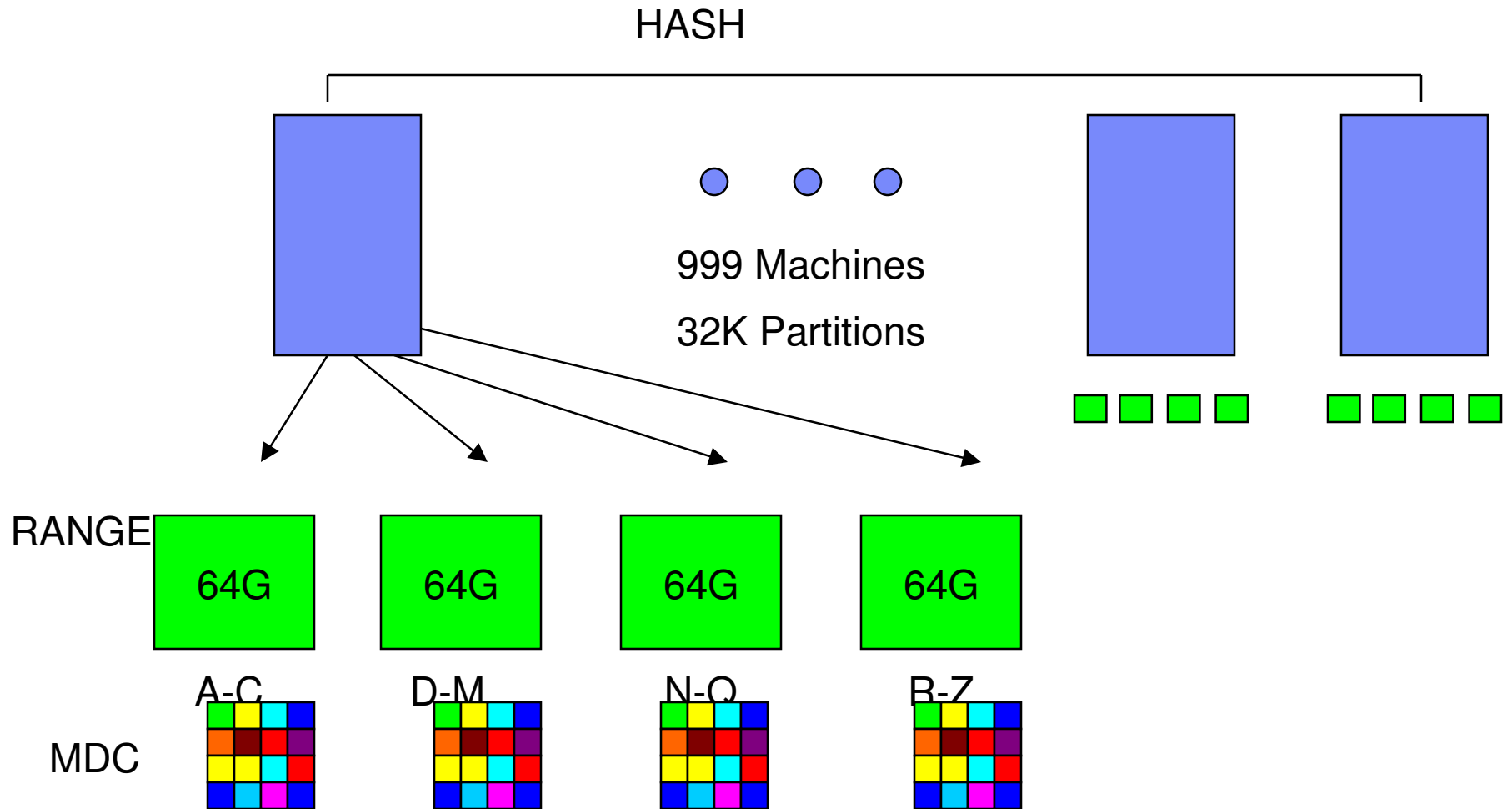
# Range Partitioning



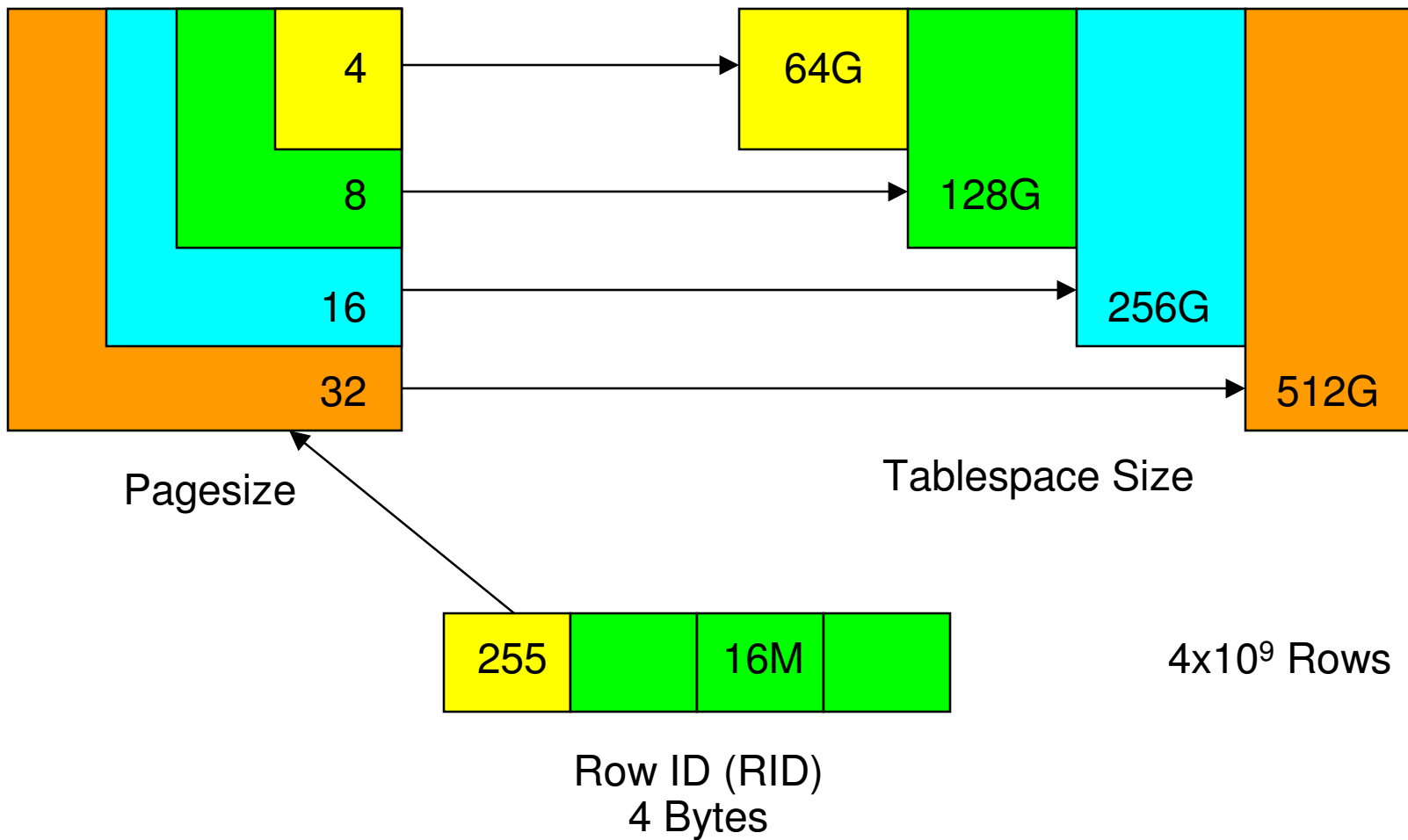
# Range Partitioning



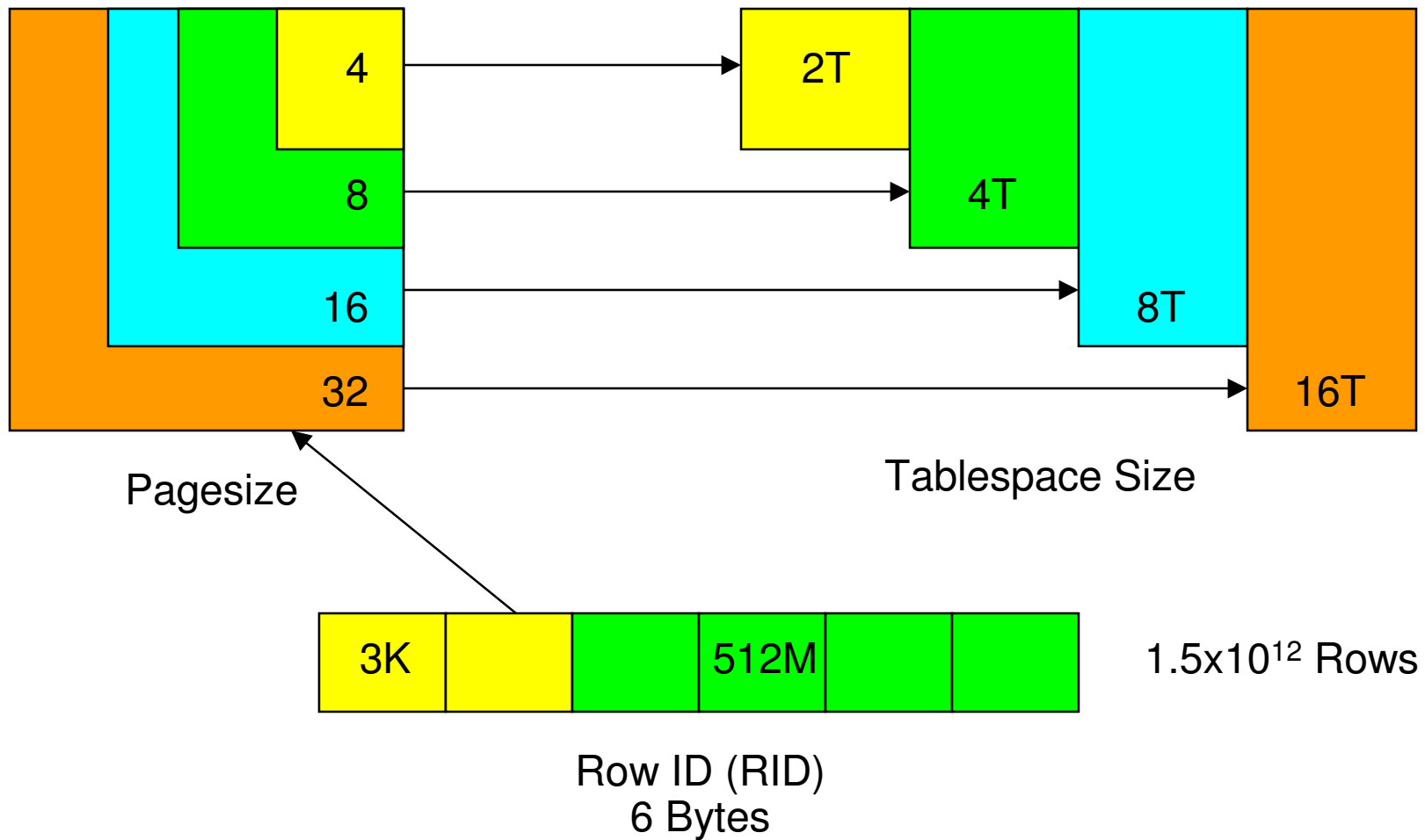
# Hybrid Partitioning



# Current Tablespace Design



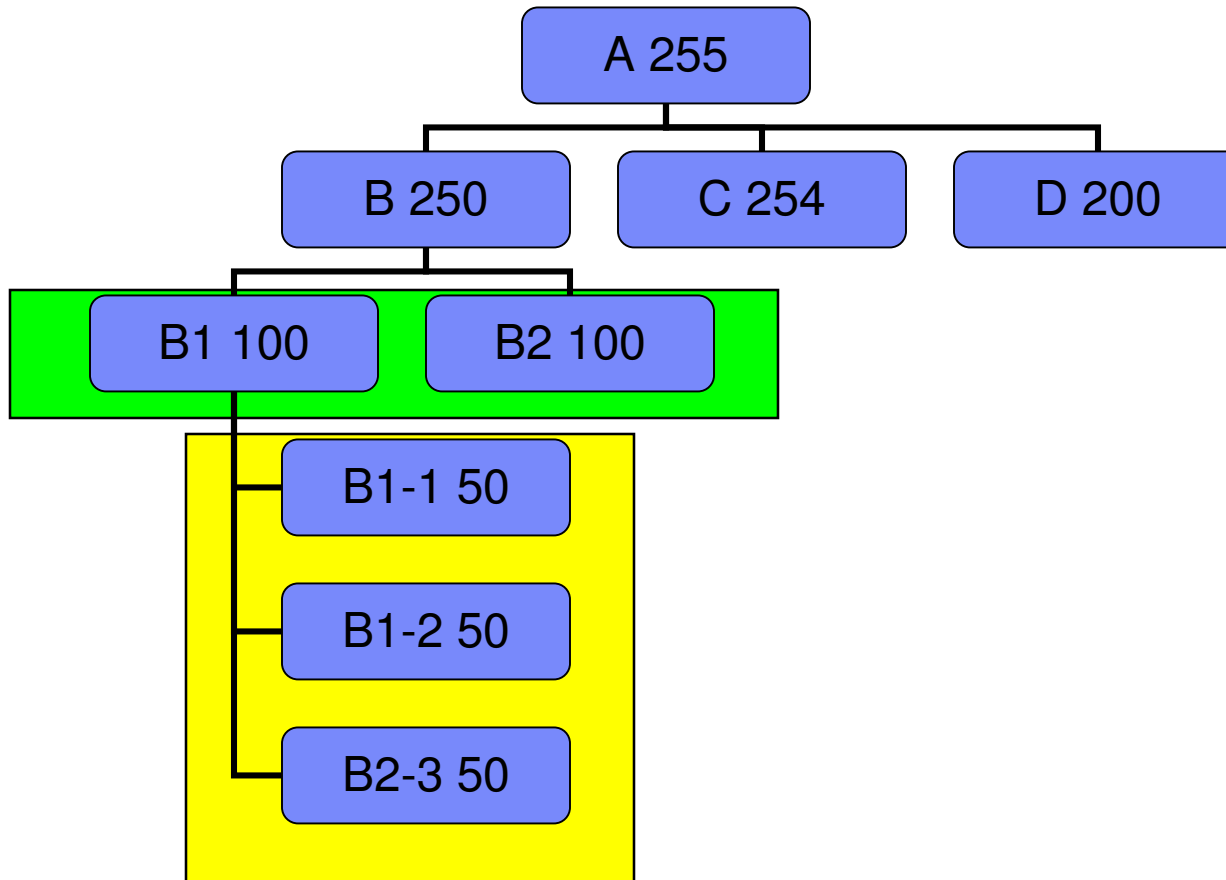
# New Tablespace Design



## Security - Label Based Access Control

- Label Based Access Control for security on a finer granularity level
  - ▶ Security on table, column and row
- Label Based Access Control (LBAC)
  - ▶ A **label** is associated with user sessions, columns and data rows
  - ▶ **Rules** for comparing user and data labels provided allow access controls to be applied at the appropriate level
  - ▶ Labels may consist of multiple **components**
    - Three types available: set, array or tree
  - ▶ **Policies** define label components and access rules that make up a security label
  - ▶ Row labels appear as a single additional column in a protected table, regardless of the number of label components
  - ▶ User labels are granted by a security administrator

# LBAC Hierarchy – Tree

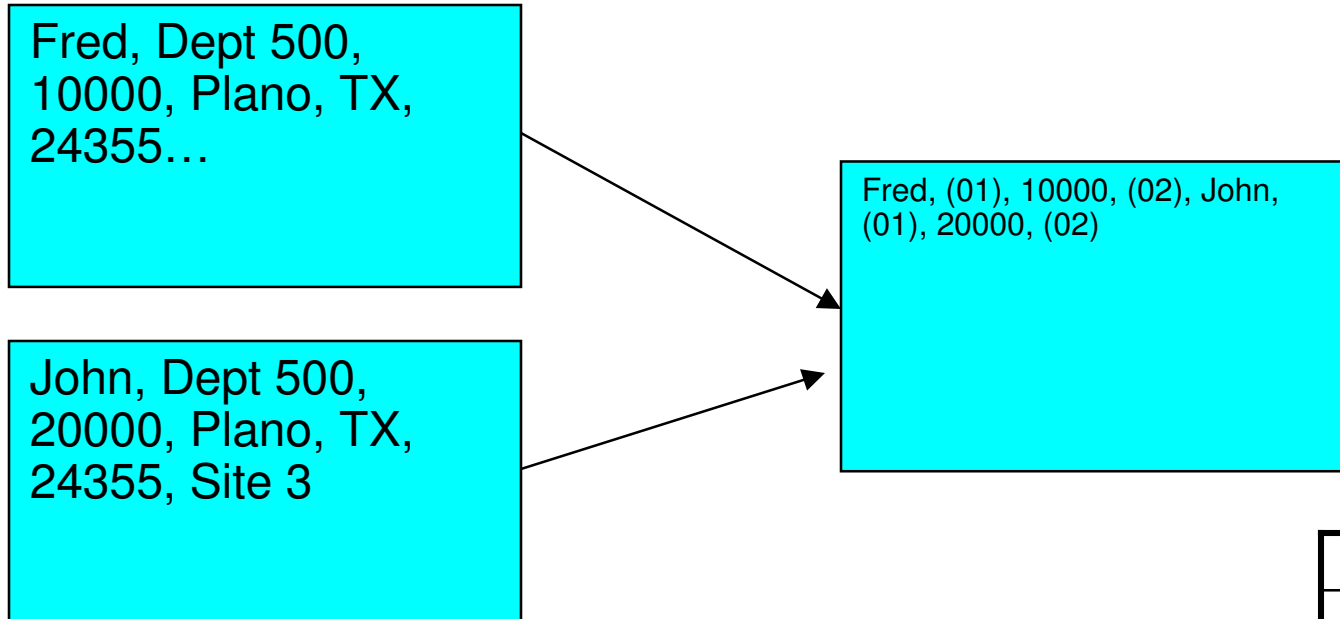




## Row Compression

- Dictionary based - symbol table for compressing/decompressing data records
- Lempel-Ziv (LZ) based algorithm (static dictionary)
- Dictionary per table stored within the permanent table object (~74KB in size)
- Data resides compressed on pages (both on-disk and in bufferpool)
  - ▶ Significant I/O bandwidth savings
  - ▶ Significant memory savings
  - ▶ CPU costs: Rows must be decompressed before being processed for evaluation
- Log data from compressed records in compressed format
- Does not compress rows where no storage saving is realized for that row
- Repeating patterns within the data (and just within each row) is the key to good compression. Text data tends to compress well because of reoccurring strings as well as data with lots of repeating characters, leading or trailing blanks

# Row Compression Using Side Tables



01	Dept 500
02	Plano, TX, 24355
...	...

- Side tables contain repeated information from the rows
- Can be across column boundaries or within columns

# Fragen / Diskussion...





IBM Software Group

# XML Support

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™

© 2005 IBM Corporation

# XML Technology

## ■ XML Technology

- ▶ XML = Extensible Markup Language
- ▶ Self-describing data structures
- ▶ XML Tags describe each element and their attributes

## ■ Benefits

- ▶ Extensible
  - No fixed format or syntax
  - Structures can be easily changed
- ▶ Platform Independent
  - Not tied to any platform, operating system, language or software vendor
  - XML can be easily exchanged
- ▶ Fully Unicode compliant

```
<? xml version="1.0" ?>
<purchaseOrder id='12345' secretKey='4x%$^'>
  <customer id="A6789">
    <name>John Smith Co</name>
    <address>
      <street>1234 W. Main St</street>
      <city>Toledo</city>
      <state>OH</state>
      <zip>95141</zip>
    </address>
  </customer>
  <itemList>
    <item>
      <partNo>A54</partNo>
      <quantity>12</quantity>
    </item>
    <item>
      <partNo>985</partno>
      <quantity>1</quantity>
    </item>
  </itemList>
</purchaseOrder>
```

# Market Projections

- XML Storage is a high growth area

Figure VI.2: Market Size by XML Data Store Solution Type

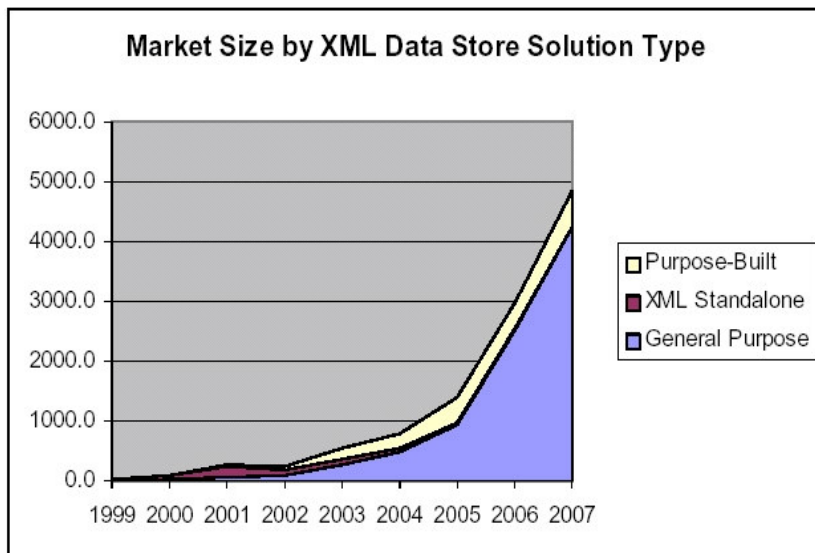
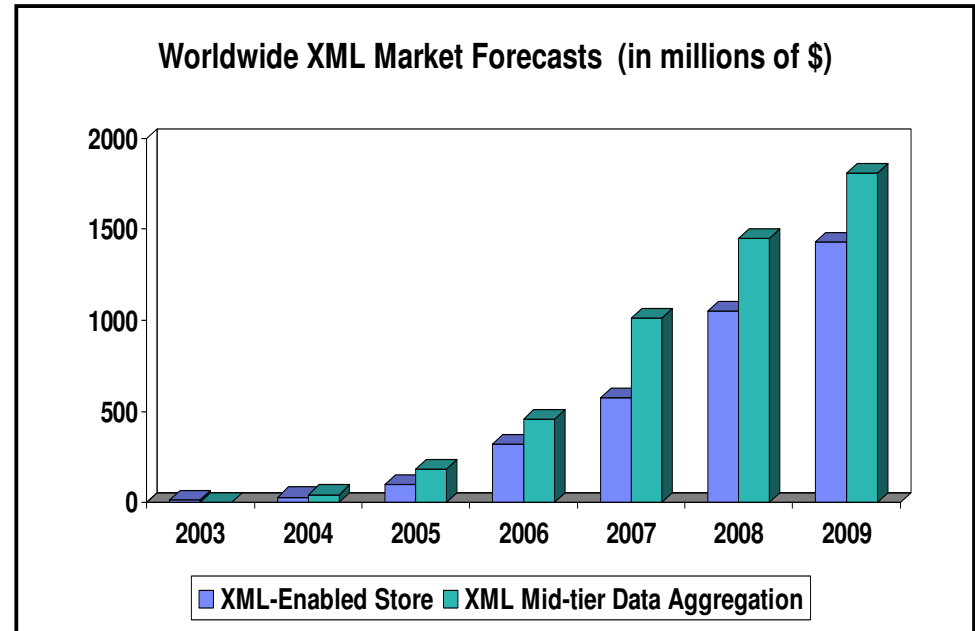


Chart Sources: XML Market Opportunities, Forecasts and Strategies, 2004-2009  
 WnterGreen Research Inc. ZapThink



- XML database revenue to grow at twice the rate of the total database market

- IDC

Worldwide Enterprise Database Management Systems  
 Software Forecast Update, 2003-2007

# XML – The Change is VERY Fundamental

- Relational is a data model:
  - Relations (tables)
  - Attributes (columns)
  - Set based w/ some sequences
  - Strict schema

POID	CustomerID	ItemID
12	1	2
162	3	4
162	3	5

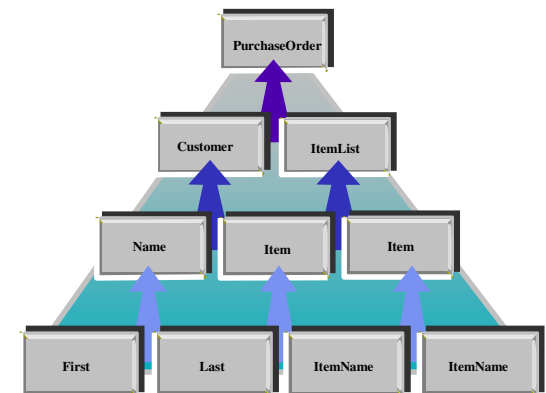
  

Id	LastName	FirstName	Street	City	State	Zip
1	Pirahesh	Hamid	1 Harry Rd	San Jose	CA	95141
3	Selinger	Pat	555 Bailey Ave	San Jose	CA	95141

ItemID	Name
2	#6 wire nut
5	Small Walrus
4	Apollo moon rocket

- XML is a data model:
  - Nodes (elements, attributes, comments, etc.)
  - Relationships between nodes
  - Sequence based w/ some sets
  - Flexible schema

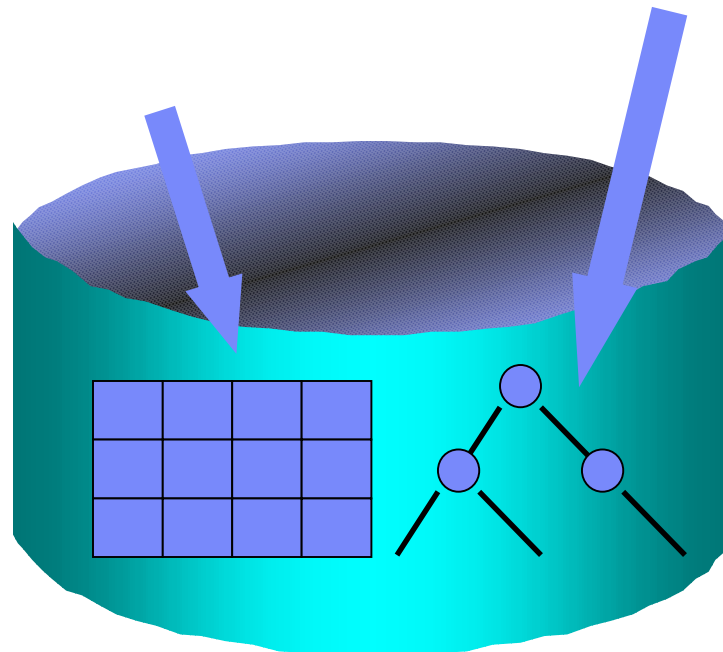


## Native XML Storage

- Must store XML in parsed hierarchical format (similar to the DOM representation of the XML infoset)

```
create table dept (deptID char(8), ..., deptdoc xml);
```

- Relational columns are stored in relational format (tables)
- XML columns are stored natively
- XML stored in UTF8
  - ▶ Unicode Database !

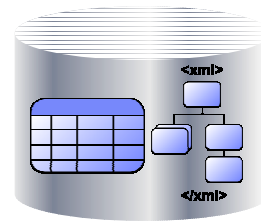




# XML in DB2 UDB Viper



**SQL Person...** "I see a world class RDBMS that also supports XML"



**DB2 with XML Support**



**XML Person...** "I see a world class XML repository that also supports SQL"

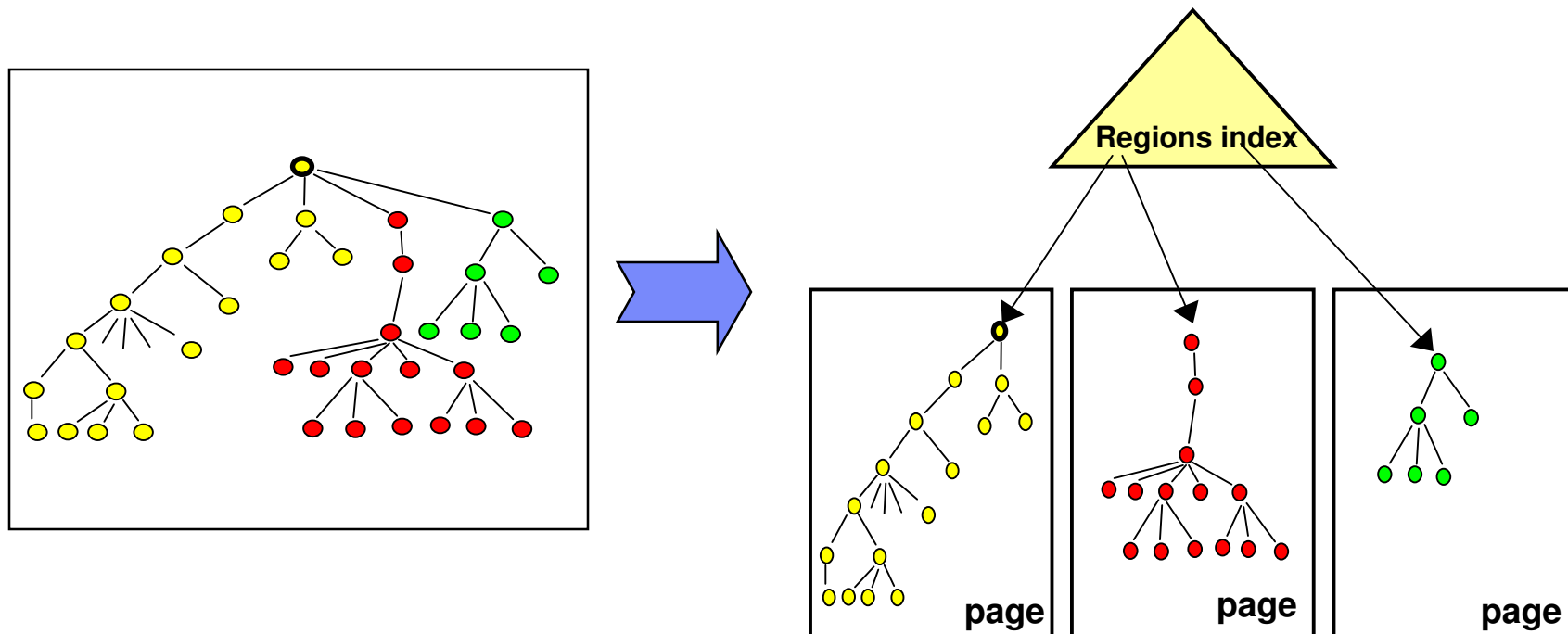
## XML integrated in all facets of DB2!

New XML applications benefit from:

- Ability to seamlessly leverage relational investment
- Proven Infrastructure that provides enterprise-class capabilities

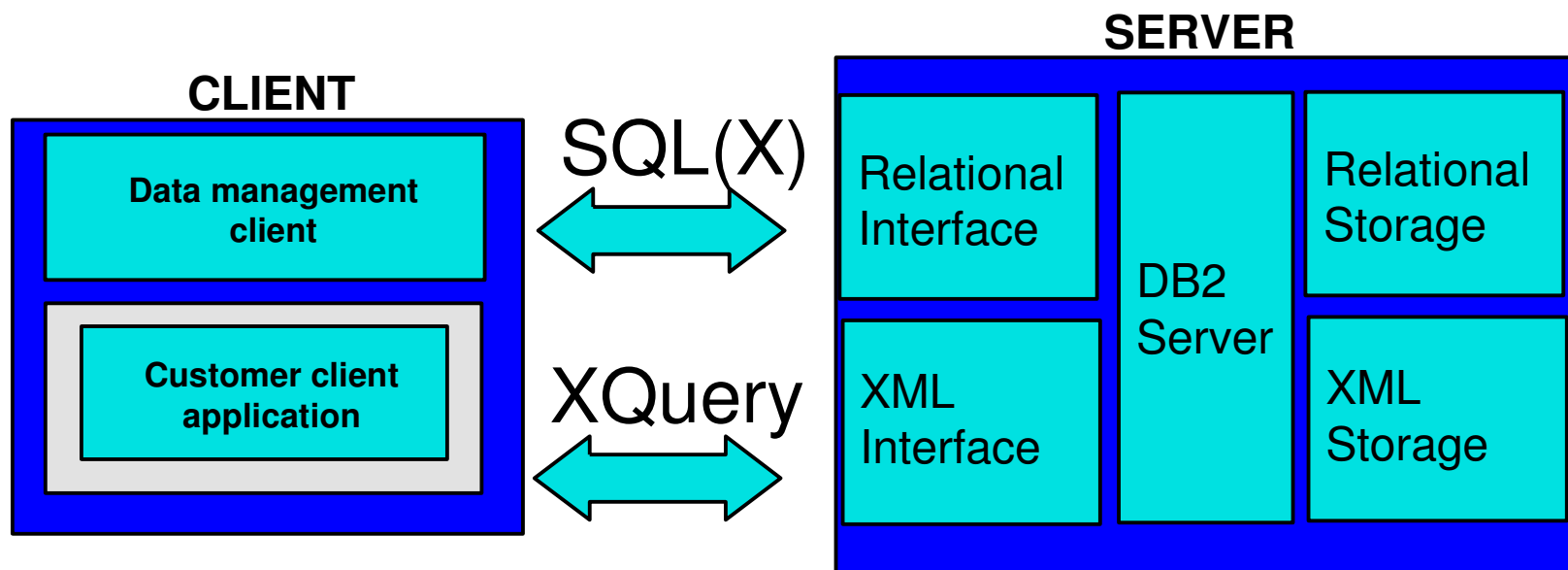
# Document Storage

- Node hierarchy of the XML Documents stored on DB2 pages
- Documents that don't fit on 1 page are split into pages/regions
- System managed Regions Index points to document parts



## XML in DB2

- XML capabilities "inside" the engine
- "Feels" relational and/or XML
- Both SQL flavor and fully XML flavor
- XML \*is\* DB2 internals - XML Extender becomes one with the data engine



## You will see:

- New XML data type for columns  
`create s1.t1 (c1 int, c2 xml)`
- New language bindings for that type  
cobol, c, java, etc..
- New XML indexes  
`create ix1 on s1.t1(c2) generate keys using pattern '/dept/emp/@empno'`
- An XML schema/DTD repository
- Support for XQuery as a primary language as well as:
  - Support for SQL within XQuery
  - Support for XQuery with SQL
  - Support for new SQL/XML functions
- Performance, scale, and everything else they expect from a DBMS

# Language Bindings



- Java Example

```
PreparedStatement stmt1 = con.prepareStatement("Select deptdoc from dept where id = '001' ");
```

```
ResultSet rs = stmt1.executeQuery();
```

```
rs.next();
```

```
// Get the first returned document as a string
```

```
String xmlString = rs.getString(1);
```

```
// As a binary stream
```

```
InputStream is = rs.getBinaryStream(1);
```

```
// As an XML object
```

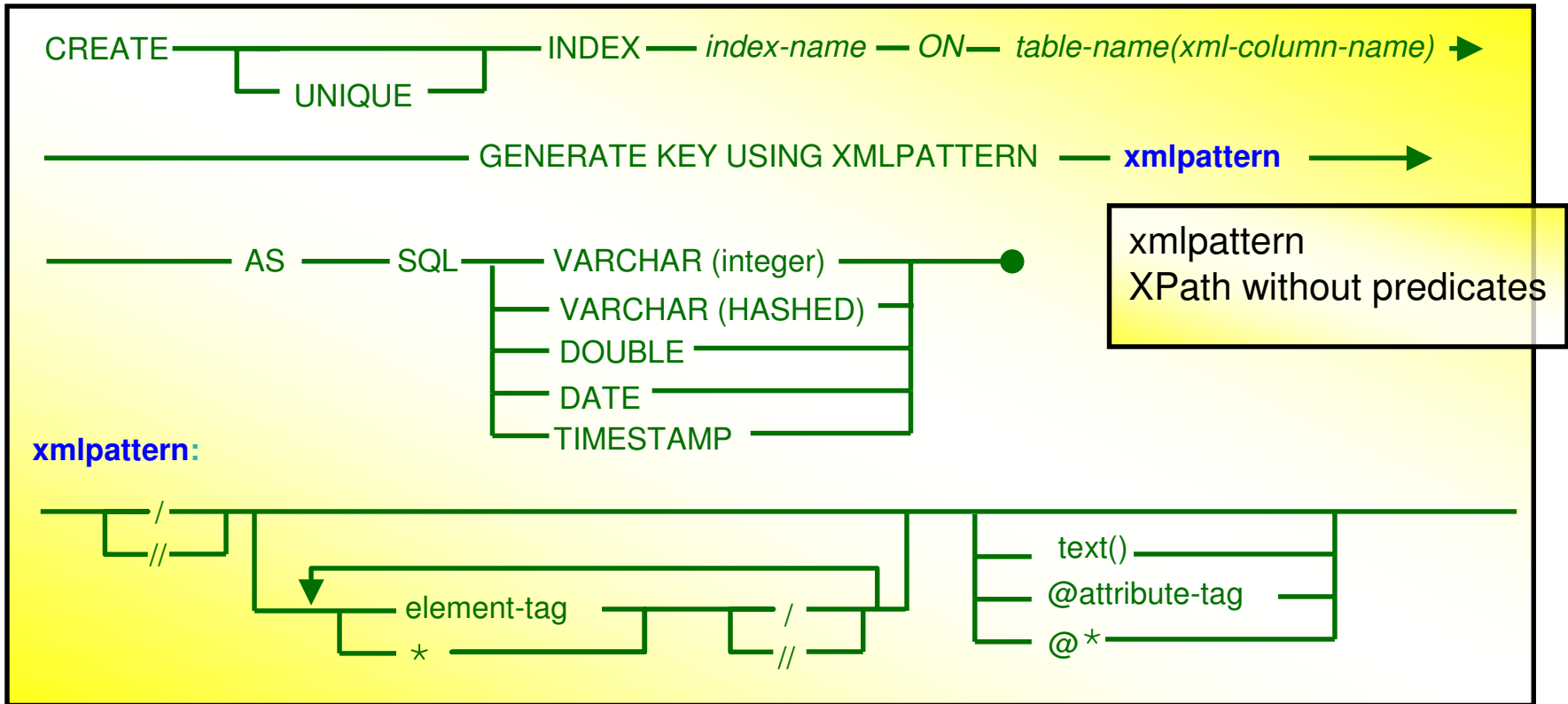
```
// The XML object class supports access methods for String, Byte and Streams
```

```
com.ibm.db2.jcc.DB2Xml xml = (com.ibm.db2.jcc.DB2Xml) rs.getObject (1);
```

```
rs.close();
```

```
stmt1.close();
```

# Indexing



## Document Retrieval using SQL

- Retrieve XML documents

Select i, **deptdoc** from dept



- Retrieve some XML documents based on SQL Predicate

Select deptnum, **deptdoc** from dept

**Where deptnum like 'PR%'**

## Document Query using XQuery

- Full power of XQuery and XPath 2.0
  - ▶ Includes FLWOR nested FLWOR etc
- XQuery over all documents in a column
  - ▶ **FOR \$d in `db2-fn:xmlcolumn('dept.deptdoc')`...**
- XQuery over documents selected using SQL predicates
  - ▶ **FOR \$d in `db2-fn:sqlquery("select deptdoc from dept where deptID LIKE 'PR%' ")`...**





# What is XQuery ?

A query language designed for XML data...  
 ...and supported in DB2 Viper.

**XML  
Schema**

[www.w3.org/  
XML/Schema](http://www.w3.org/XML/Schema)

**XQUERY**

<p><b>Expressions</b>  <a href="http://www.w3.org/TR/xquery">www.w3.org/TR/xquery</a></p>	<p><b>XPath 2.0</b>   <a href="http://www.w3.org/TR/xpath20/">www.w3.org/ TR/xpath20/</a></p>
<p><b>Functions &amp; Operators</b>  <a href="http://www.w3.org/TR/xquery-operators/">www.w3.org/TR/xquery-operators/</a></p>	
<p><b>XQuery 1.0 &amp; XPath 2.0 Data Model</b>  <a href="http://www.w3.org/TR/query-datamodel/">www.w3.org/TR/query-datamodel/</a></p>	

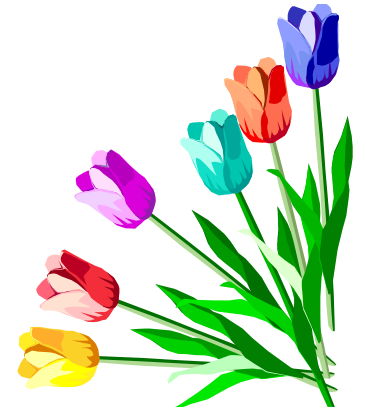


## XQuery: The FLWOR Expression

- **F**OR: iterates through a sequence, bind variable to items
- **L**ET: binds a variable to a sequence
- **W**HERE: eliminates items of the iteration
- **O**RDER: reorders items of the iteration
- **R**ETURN: constructs query results

```
create table dept(deptID char(8), deptdoc xml);
```

```
for $d in db2-fn:xmlcolumn('dept.deptdoc')/dept
let $emp := $d//employee/name
where $d/@bldg > 95
order by $d/@bldg
return <EmpList>
      {$d/@bldg, $emp}
</EmpList>
```



```
<dept bldg=101>
  <employee id=901>
    <name>John Doe</name>
    <phone>408 555 1212</phone>
    <office>344</office>
  </employee>
  <employee id=902>
    <name>Peter Pan</name>
    <phone>408 555 9918</phone>
    <office>216</office>
  </employee>
</dept>
```



IBM Software Group

# Maintenance

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™

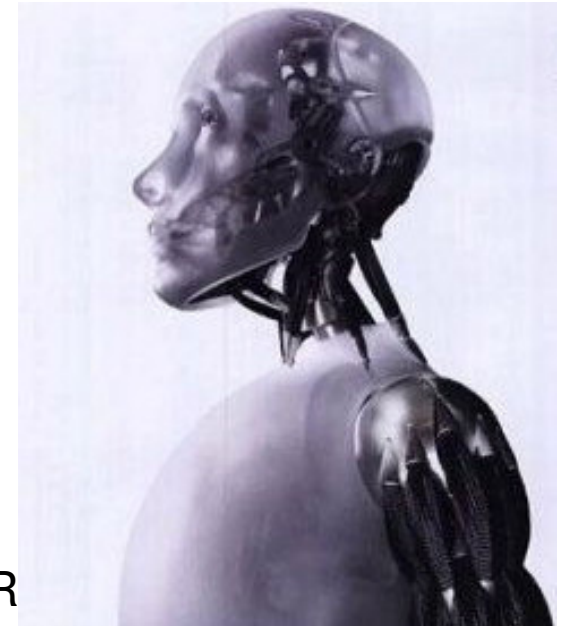
© 2005 IBM Corporation

# Installation

- Reduce installation complexity
  - ▶ Allow non-admin user to install using Elevated Privileges
    - non “root” install on Unix/Linux
    - non “admin-user” install on Windows
  - ▶ Multiple DB2 versions and fixpacks on the same Windows system
- Multiple instances/versions/fixpacks for maintenance
  - ▶ On Windows, Linux, and UNIX
- Uninstall
  - ▶ Allow full uninstall on Windows

## Automation Automatically!

- Enable many of the DB2 autonomic computing features by default.
  
- Examples:
  - ▶ Configuration Advisor (2 second tuning)
  - ▶ Adaptive Self Tuning Memory
  - ▶ Automatic data statistics collection.
  
- Better defaults for I/O Cleaners and I/O servers
  - ▶ Default for NUM\_IOSERVERs (3) and NUM\_IOCLEANER AUTOMATIC
  - ▶ Values calculated at database startup time
    - IOCLEANERS calculated based on number of CPUs and partitions
    - IOSERVERS calculated based on parallelism settings of all the tablespaces



# Backup and Restore

- Table function to list files in a database
  - ▶ Used to automate support of split mirror backup/recovery
  
- Restartable Recovery
  - ▶ Re-issuing RECOVER command will pick up where it left off
  - ▶ Ability to change Point In Time in either direction
  
- Redirected Restore Script builder
  - ▶ Build a redirected restore script from a backup image

## DB2 Simplified Storage Administration

- User specifies a group of storage devices for DB2, DB2 allocates and grows table consumption of storage on demand.
  - ▶ New to the “Saturn” release of DB2
  - ▶ Intended as a “single point of storage management” for table spaces
  - ▶ Create a database and associate a set of storage paths with it
  
- AUTOMATIC STORAGE table spaces
  - ▶ No explicit container definitions are provided
  - ▶ Containers automatically created across the storage paths
  - ▶ Growth of existing containers and addition of new ones completely managed by DB2

## Memory Tuning in Viper - STMM

Viper will introduce a revolutionary memory tuning system called the “**Self Tuning Memory Manager**” (STMM)

- Works on main database memory parameters
  - ▶ Sort, locklist, package cache, buffer pools, and total database memory
- Hands-off online memory tuning
  - ▶ Requires no DBA intervention
- Senses the underlying workload and tunes the memory based on need
- Can adapt quickly to workload shifts that require memory redistribution
- Adapts tuning frequency based on workload
- Is able to tune multiple databases and instances on the same box at the same time
- Works in non-partitioned and in partitioned (DPF) environments



## STMM and the buffer pools

- Trades memory between buffer pools based on relative need
  - ▶ New metrics determine where memory is most needed such that total system time is reduced
- Zero, one or more buffer pools can be set to **AUTOMATIC**
  - ▶ In newly created Viper databases, all buffer pools default to
- Decreasing the buffer pools can take a lot of time
  - ▶ Must write out all dirty pages in memory being freed
  - ▶ If pages are in use the resize may wait on locks
  - ▶ A large percentage of tuning time could be spent on alter buffer pools
    - Not necessarily a concern, just something to keep in mind
- STMM can be set off individually for each bufferpool

# STMM and DATABASE\_MEMORY

- STMM tunes DATABASE\_MEMORY if it is set to AUTOMATIC or a numeric value
  - ▶ If set to AUTOMATIC, memory is taken from, and returned to, the OS if required by the database
    - DBA need not know how much memory to allocate to DB2
    - This is the default for newly created Viper databases
  - ▶ If set to a numeric value, memory is given to AUTOMATIC heaps up to the numeric value
    - Allows DBA to set total memory consumption for the database
    - DB2 will then distribute the memory to optimize performance
  - ▶ If set to COMPUTED, no DATABASE\_MEMORY tuning will occur
    - When database starts, memory requirements are computed based on the heap configuration
    - Once the database starts, the database shared memory set is allocated based on the computation
    - Version 8 AUTOMATIC behavior

## Tuning with DPF

- Some assumptions are made on the underlying system
  - ▶ All data nodes are similarly distributed and require similar memory
  - ▶ All physical nodes contain the same number of logical nodes
    - Or all logical nodes require the same amount of memory
      - Logical nodes/memory ratio must be consistent
- Why the assumptions?
  - ▶ Tuner tries to maintain a consistent configuration on all nodes
- Individual nodes can be uncaged from STMM (i.e. for catalog node)

## Scenarios where STMM shines

### Bufferpool Tuning

- Difficult to tune memory when there are multiple buffer pools
- Works so well that STMM is being used to tune benchmark systems in house

### Varying workloads

- Periods with high transaction throughput
- Periods with long running transactions or online utilities

### Unknown memory requirements

- Tunes quickly enough to bring production systems from out of the box configuration to optimal in an hour or less
- Performs several weeks of manual (trial and error) tuning every hour
- Will stop tuning automatically when it reaches optimal configuration



IBM Software Group

# Partitioning

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™

© 2005 IBM Corporation

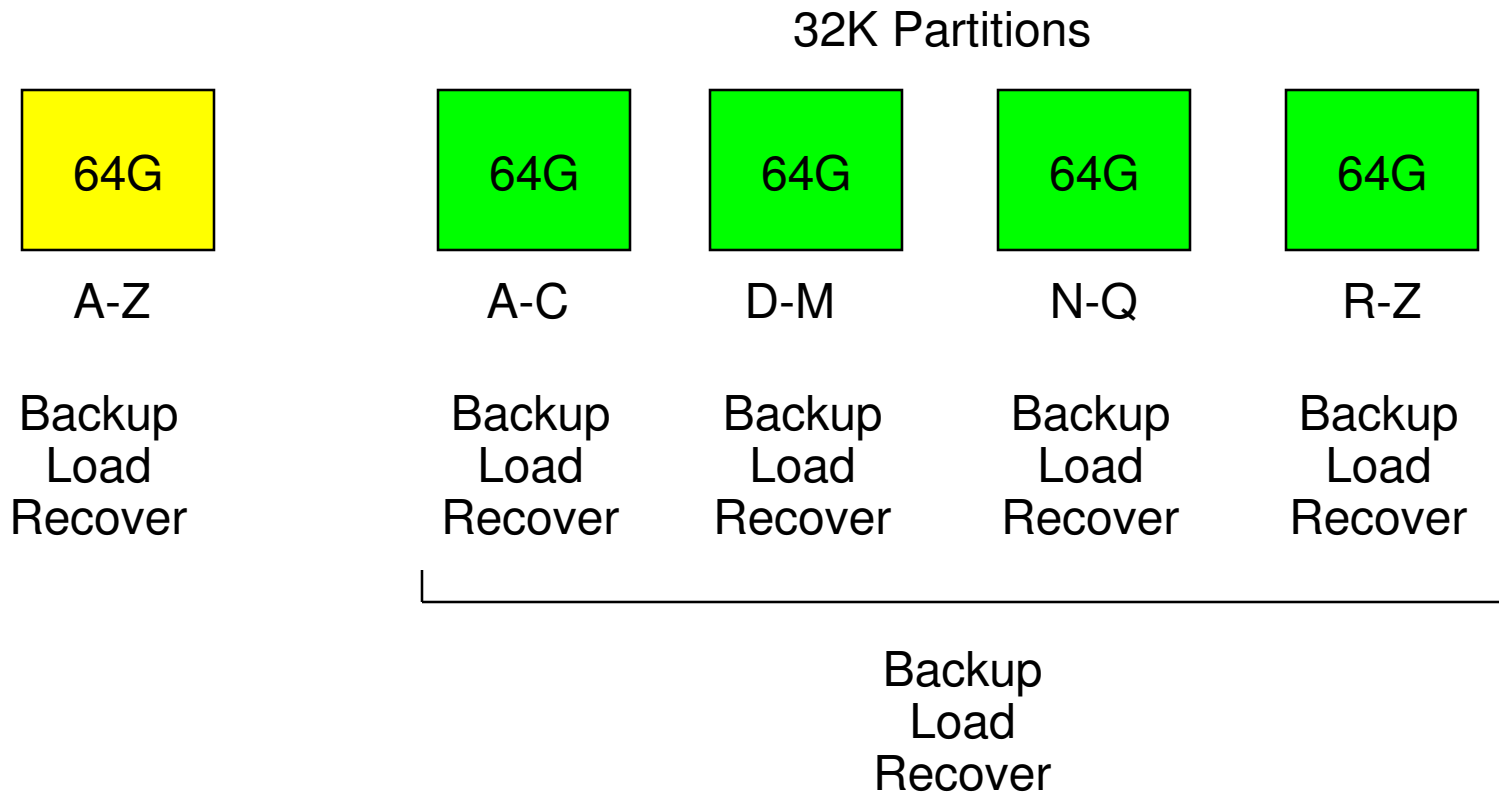
# Range Partitioning Thoughts

- What is Range Partitioning ?
  - ▶ Storing a table in more than one physical object, across one or more tablespaces
  - ▶ Each tablespace contains a range of the data that can be found very efficiently
  
- Why ?
  - ▶ Increase table capacity limit
  - ▶ Increase large table manageability
  - ▶ Improve SQL performance through partition elimination
  - ▶ Provide fast & online data roll-in and roll-out
  - ▶ Converge towards Informix functionality
  - ▶ Family compatibility with DB2 on zOS

# Terminology (Don't Let It Confuse You)

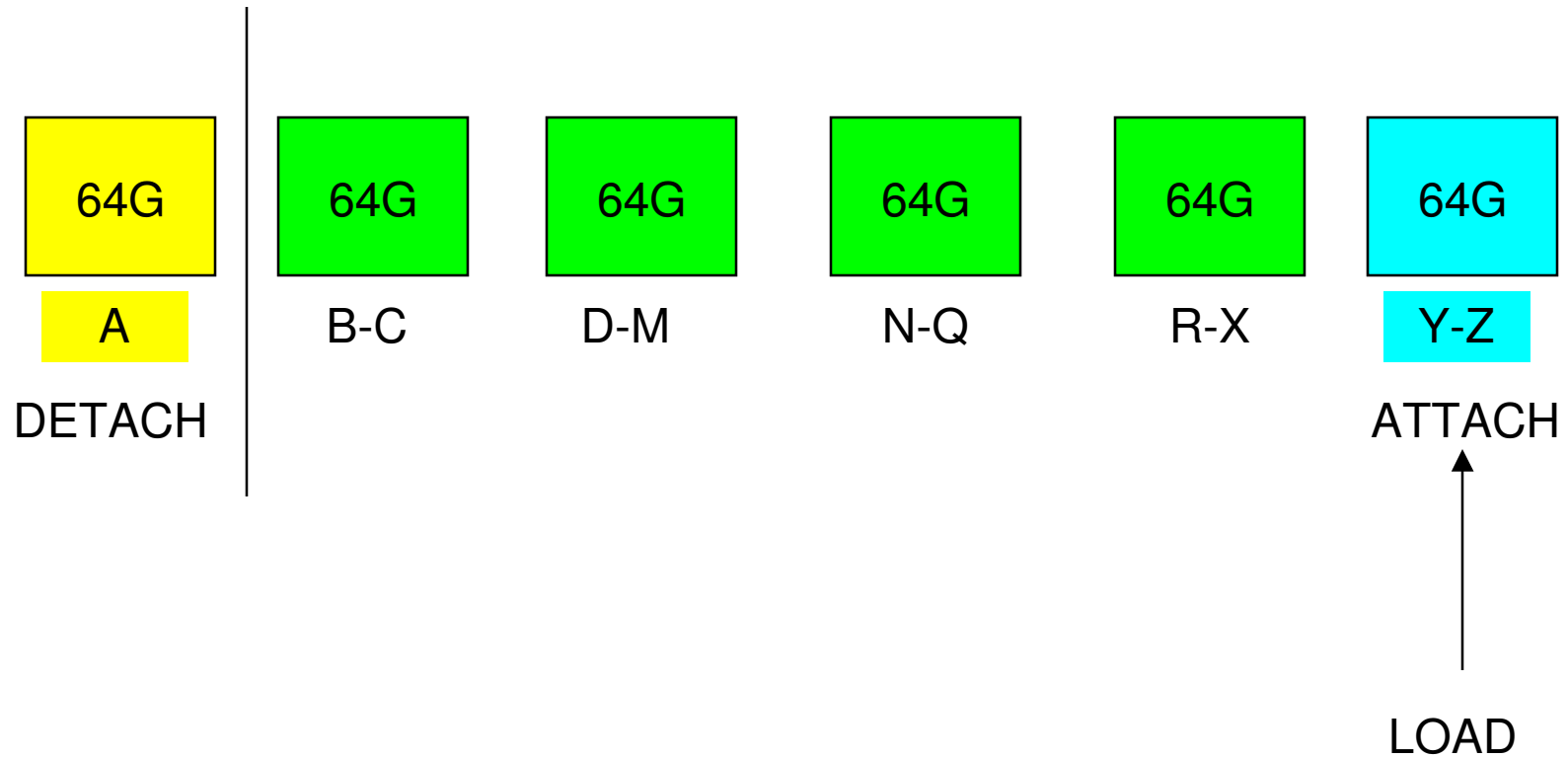
- **DATABASE PARTITIONING**
  - ▶ Distributing data by key hash across logical nodes of the database (aka DPF)
- **DATABASE PARTITION**
  - ▶ An individual “node” of a database that is using database partitioning (aka a DPF node)
- **TABLE PARTITIONING (new)**
  - ▶ Splitting data by key range over multiple physical objects within a logical database partition
- **RANGE or DATA PARTITION (new)**
  - ▶ An individual “range” of a table partitioned using table partitioning
  - ▶ Represented by an object on disk
- **PARTITION ELIMINATION**
  - ▶ When the optimizer determines that some data partitions do not need to be scanned
- **MULTI DIMENSIONAL CLUSTERING**
  - ▶ Organizing data in table (or range of a table) by multiple key values (aka MDC)
- **Database Partitioning, Table Partitioning and Multi Dimensional Clustering can be used simultaneously on the same table**

# Range Partitioning

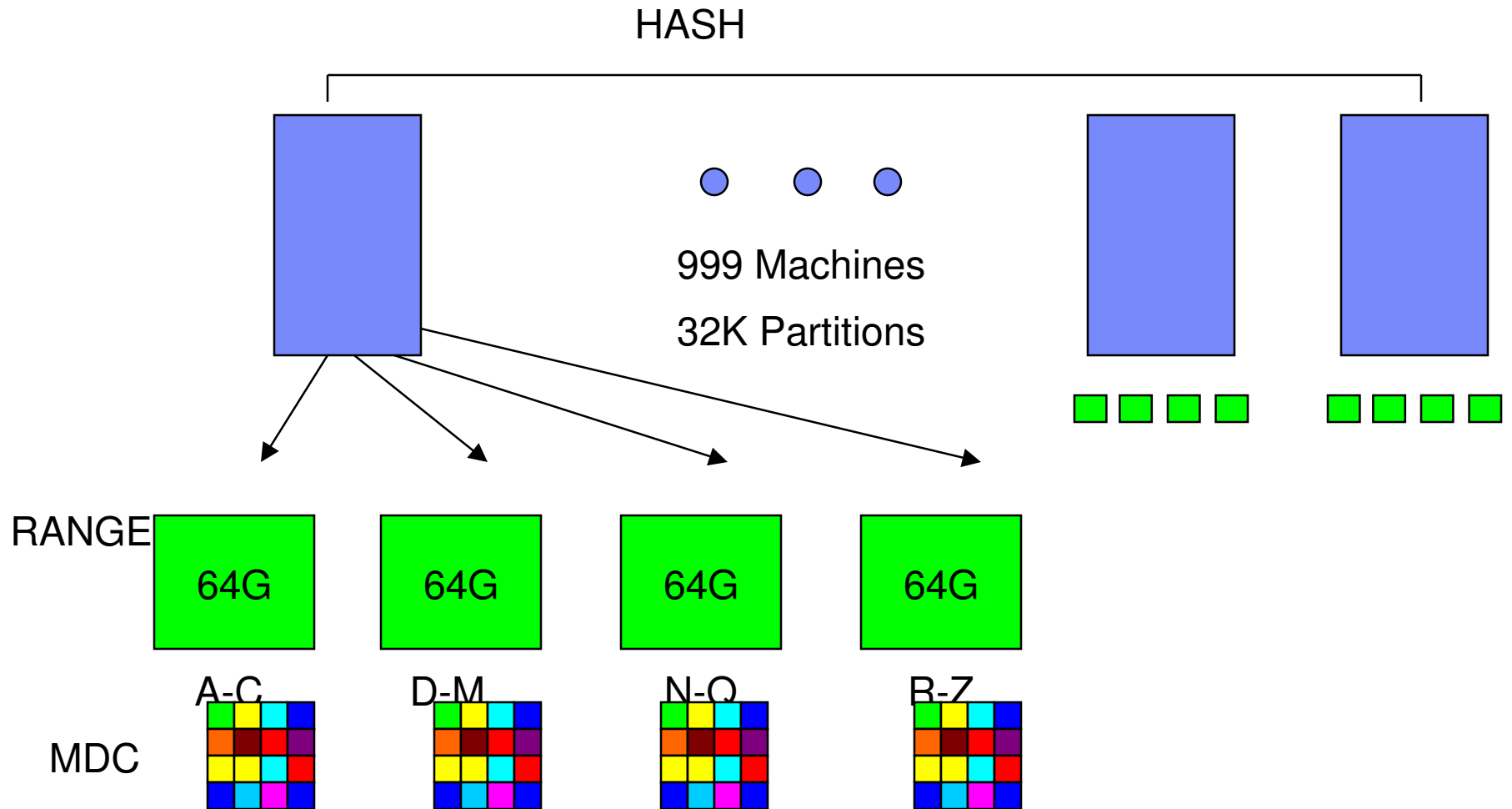




# Range Partitioning



# Hybrid Partitioning



# Table Partitioning

- Example:

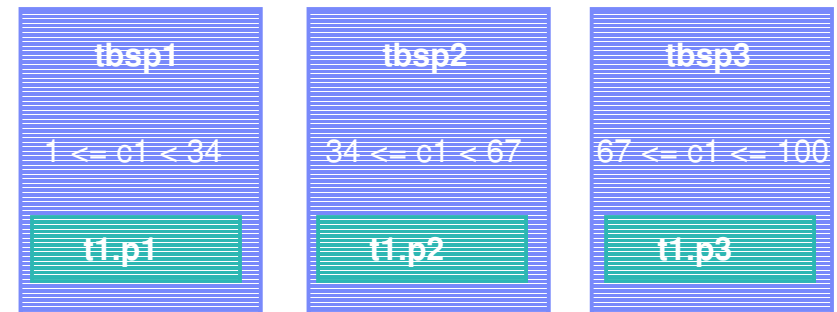
```
CREATE TABLE ORDERS (... DELIVERY DATE)
  IN TBSP1, TBSP2, TBSP3, TBSP4
  PARTITION BY RANGE (DELIVERY)
  STARTING FROM ('JAN-01-2005')
  EVERY (3 MONTHS) ENDING ('DEC-31-2005')
```

- Syntax also supports clauses where specific values can be specified

TBSP1	TBSP2	TBSP3	TBSP4
JAN	APR	JUL	OCT
FEB	MAY	AUG	NOV
MAR	JUN	SEP	DEC

# Creating a Range Partitioned Table: Overview

- Short and long forms
- Partitioning column(s)
  - ▶ Must be base types (eg. No LOBS, LONG VARCHARS)
  - ▶ Can specify multiple columns
  - ▶ Can specify generated columns
- Notes
  - ▶ SQL0327N The row cannot be inserted because it is outside the bounds
  - ▶ Special values, MINVALUE, MAXVALUE can be used to specify open ended ranges



## Short Form

```
CREATE TABLE t1(c1 INT) IN tbsp1, tbsp2, tbsp3
PARTITION BY RANGE(c1)
(STARTING FROM (1) ENDING(100) EVERY (33))
```

## Long Form

```
CREATE TABLE t1(c1 INT)
PARTITION BY RANGE(a)
(STARTING FROM (1) ENDING(34) IN tbsp1,
ENDING(67) IN tbsp2,
ENDING(100) IN tbsp3)
```

## Create Table: Defining Ranges

- Use STARTING ... ENDING ... to specify ranges
  - ▶ This example creates 4 ranges

```
CREATE TABLE sales(sale_date DATE, customer INT, ...)

PARTITION BY RANGE(sale_date)
(STARTING '1/1/2000' ENDING '3/31/2000',
 STARTING '4/1/2000' ENDING '6/30/2000',
 STARTING '7/1/2000' ENDING '9/30/2000',
 STARTING '10/1/2000' ENDING '12/31/2004');
```

## Create Table: Open Ended Ranges

- Use MINVALUE and MAXVALUE to specify open ended ranges
  - ▶ In this example, the first range holds everything before the year 2000
  - ▶ Think of these as positive and negative infinity

```
CREATE TABLE sales(sale_date DATE, customer INT, ...)
PARTITION BY RANGE(sale_date)

(STARTING MINVALUE ENDING '12/31/1999',
 STARTING '1/1/2000' ENDING '3/31/2000',
 STARTING '4/1/2000' ENDING '6/30/2000',
 STARTING '7/1/2000' ENDING '9/30/2000',
 STARTING '10/1/2000' ENDING '12/31/2004');
```

## Create Table: Naming Partitions

- You can name partitions via the PART or PARTITION keyword
  - ▶ Provided name overrides system generated name
  - ▶ Used to specify partition on partition level operations like DETACH (more on this later)

```
CREATE TABLE sales(sale_date DATE, customer INT, ...)
  PARTITION BY RANGE(sale_date)

(PART      rest  STARTING MINVALUE,
 PARTITION q1   STARTING '1/1/2000',
 PARTITION q2   STARTING '4/1/2000',
 PARTITION q3   STARTING '7/1/2000',
 PARTITION q4   STARTING '10/1/2000' ENDING '12/31/2004');
```

## Create Table: Short Syntax

- Use **STARTING ... ENDING ... EVERY** to quickly define ranges
  - ▶ Simple way of creating many partitions quickly and easily
  - ▶ Appropriate for equal sized ranges based on dates or numbers
  - ▶ This example creates 20 data partitions, one for each quarter

```
CREATE TABLE sales(sale_date DATE, customer INT, ...)
PARTITION BY RANGE(sale_date)
(STARTING '1/1/2000' ENDING '12/31/2004' EVERY 3 MONTHS);
```



## Partitioning on Multiple Columns

- Multiple columns can be specified in the PARTITION BY clause
  - ▶ Analogous to multiple columns in an index key
- Ranges cannot overlap
  - ▶ Use MDC if you need to define grids or cubes

```
CREATE TABLE sales(year INT, month INT, ...)
PARTITION BY RANGE (year, month)

(STARTING (2000, 1) ENDING (2000, 6),
 STARTING (2000, 7) ENDING (2001, 3),
 STARTING (2001, 4) ENDING (2001, 9),
 STARTING (2001, 10) ENDING (2002, 12));
```

2000	1	2	3	4	5	6	7	8	9	10	11	12
2001	1	2	3	4	5	6	7	8	9	10	11	12
2002	1	2	3	4	5	6	7	8	9	10	11	12

# Storage Mapping : Mapping Ranges to Tablespaces

- With the long syntax,....
  - ▶ You explicitly specify a tablespace for each partition
  - ▶ In this example, data in the 1Q/2000 will be placed in tbsp1, 2Q/2000 in tbsp2, 3Q/2000 in tbsp3



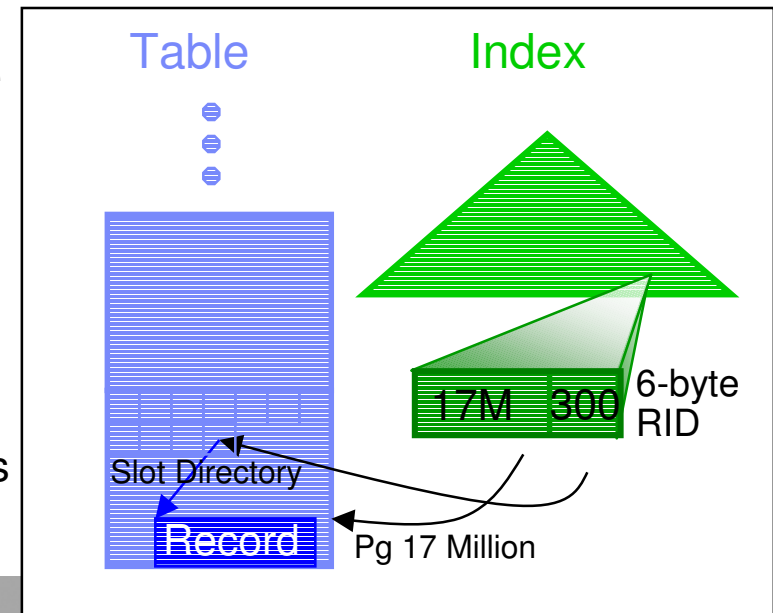
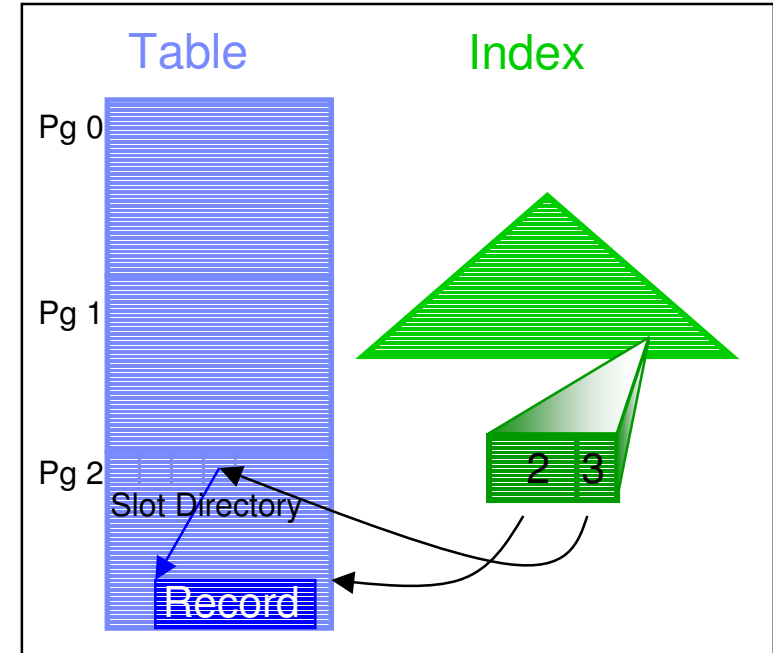
```
CREATE TABLE sales(sale_date DATE, customer INT, ...)
  PARTITION BY RANGE(sale_date)
  (STARTING MINVALUE                               IN tbsp1,
   STARTING '3/1/2000'                             IN tbsp2,
   STARTING '6/1/2000' ENDING '9/30/2000'         IN tbsp3);
```

## Large Row Identifiers

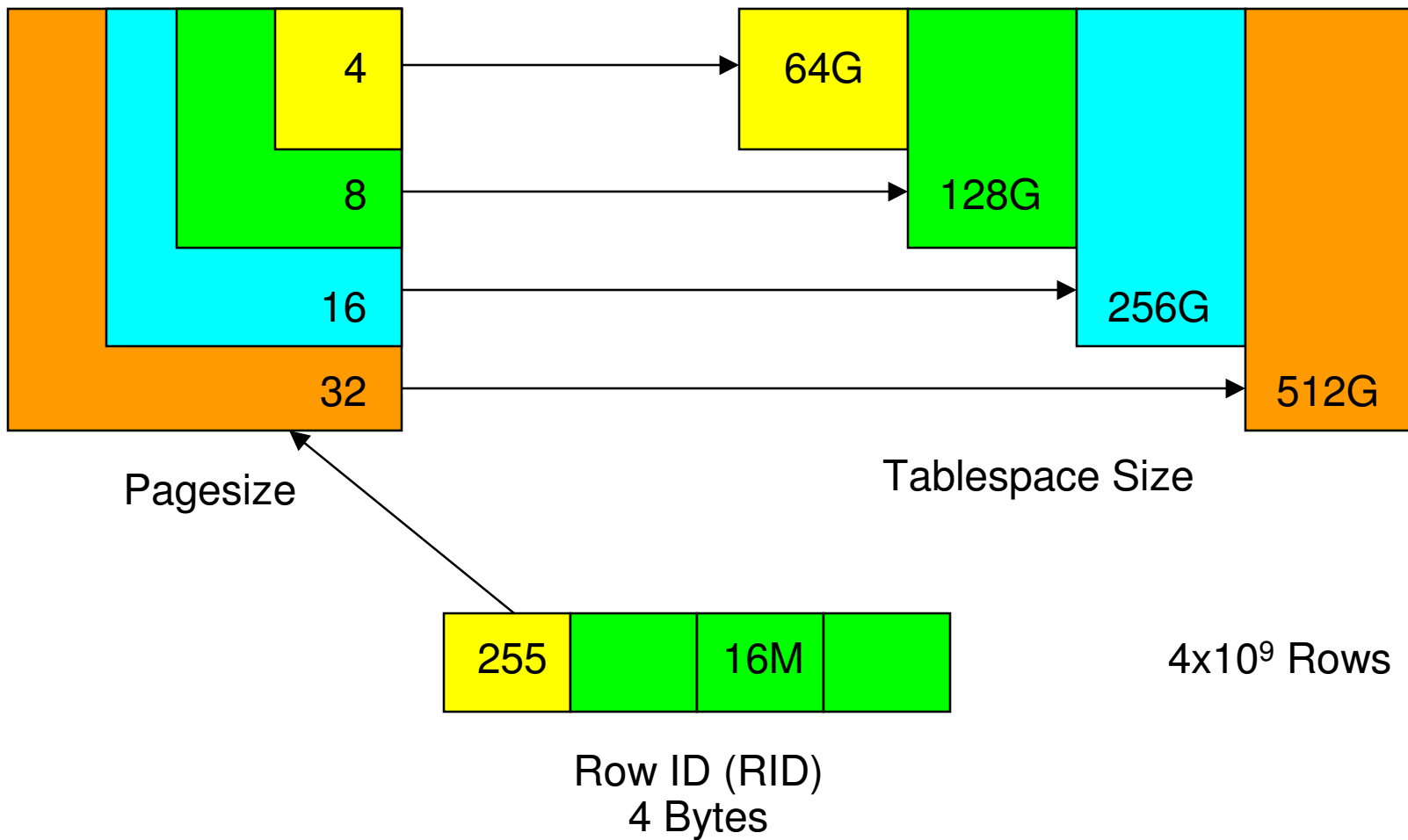
- Increase table size limits and rows per page
  - ▶ Tablespace level definition
  - ▶ DMS Tablespace only

# Large Record IDs

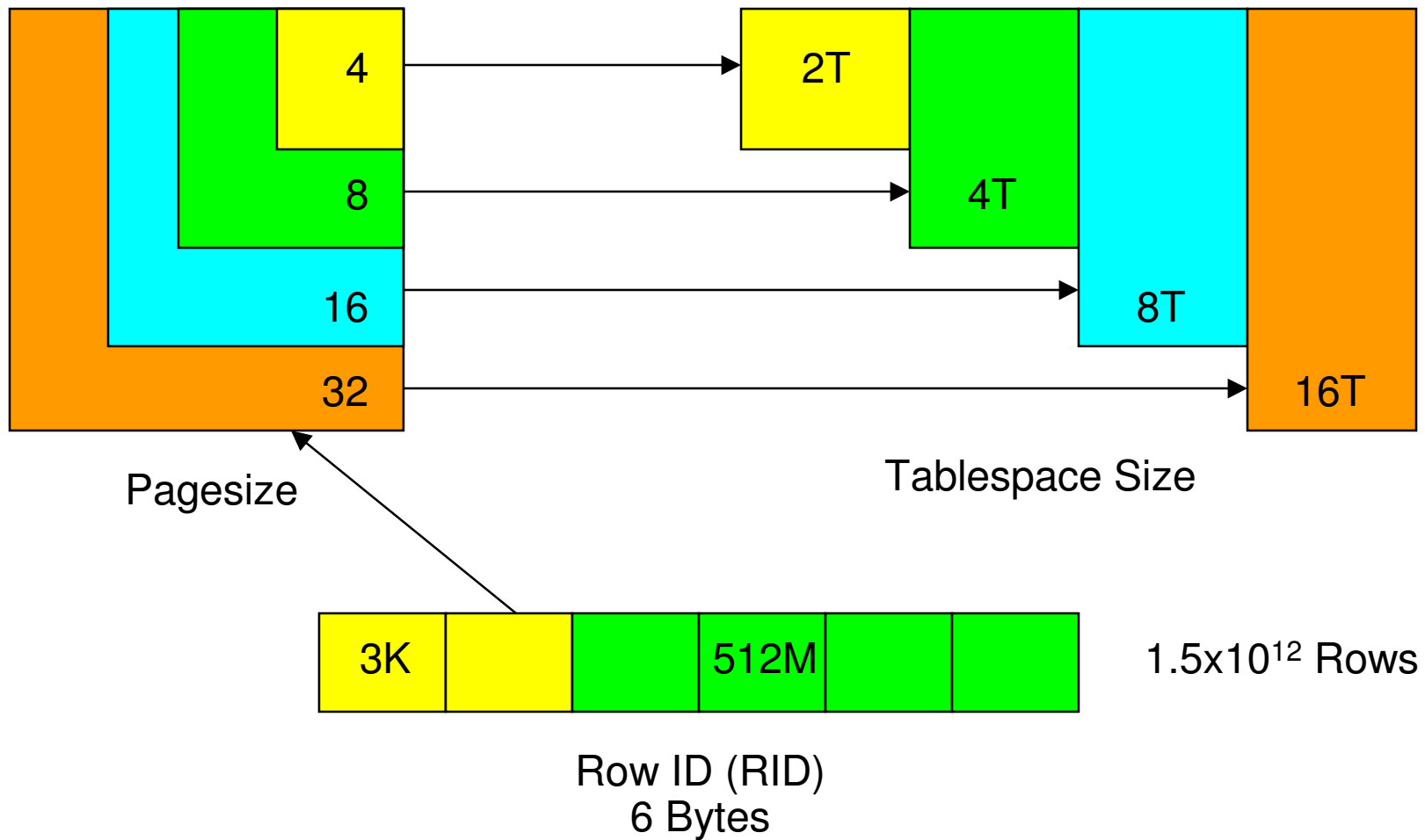
- Before Viper
  - ▶ RIDs are 4 bytes: 3 byte page number and 1 byte slot number
  - ▶ Default table space data type was REGULAR
  - ▶ Tables (data part) could not be placed in LARGE table spaces
  
- In Viper
  - ▶ RIDs can be 6 bytes: 4 byte page number and 2 byte slot number
  - ▶ Enabled by creating a table in a LARGE tablespace (Viper will allow tables to be created in LARGE tablespaces -> previously not possible)
  - ▶ LARGE is now the default type for new DMS and AUTO STORAGE tablespaces
  - ▶ Note: the size of RIDs in indexes is dependent on the size of RIDs in the table (it has nothing to do with the type of table space where the index resides)



# Current Tablespace Design



# New Tablespace Design





IBM Software Group

# Security

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™

© 2005 IBM Corporation

## Security - Label Based Access Control

- Label Based Access Control for security on a finer granularity level
  - ▶ Security on table, column and row
- Label Based Access Control (LBAC)
  - ▶ A **label** is associated with user sessions, columns and data rows
  - ▶ **Rules** for comparing user and data labels provided allow access controls to be applied at the appropriate level
  - ▶ Labels may consist of multiple **components**
    - Three types available: set, array or tree
  - ▶ **Policies** define label components and access rules that make up a security label
  - ▶ Row labels appear as a single additional column in a protected table, regardless of the number of label components
  - ▶ User labels are granted by a security administrator



# LBAC – Label Component SET

- **Set** – collection of elements where the order in which those elements appear is not important
  - ▶ e.g. { 'US', 'UK', 'DE' }
  - ▶ User has label { 'US' }
  - ▶ User has label { 'DE', 'UK' }

Label	Value
'US'	1
'US', 'UK'	2
'DE'	3
'UK'	4
'DE', 'UK'	5

Label	Value
'US'	1
'US', 'UK'	2
'DE'	3
'UK'	4
'DE', 'UK'	5

Label	Value
'US'	1
'US', 'UK'	2
'DE'	3
'UK'	4
'DE', 'UK'	5

# LBAC – Label Component ARRAY

- **Array** – ordered set that can be used to represent a simple hierarchy
  - ▶ [ ‘Top Secret’, ‘Secret’, ‘Confidential’, ‘Unclassified’ ]
  - ▶ User has label { ‘Unclassified’ }
  - ▶ User has label { ‘Secret’ }

Label	Value
‘Secret’	1
‘Confidential’	2
‘Unclassified’	3
‘Top Secret’	4
‘Secret’	5

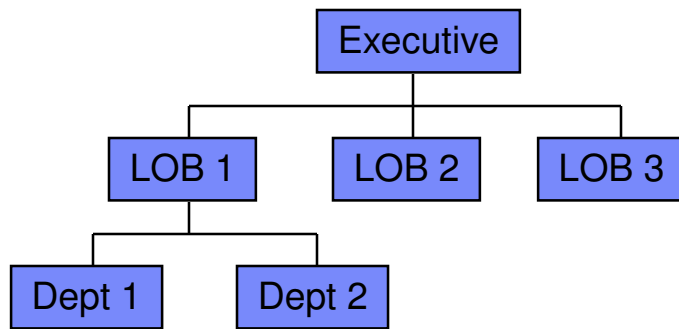
Label	Value
‘Secret’	1
‘Confidential’	2
‘Unclassified’	3
‘Top Secret’	4
‘Secret’	5

Label	Value
‘Secret’	1
‘Confidential’	2
‘Unclassified’	3
‘Top Secret’	4
‘Secret’	5



# LBAC – Label Component TREE

- **Tree** – represents a more complex hierarchy that can have multiple nodes and branches



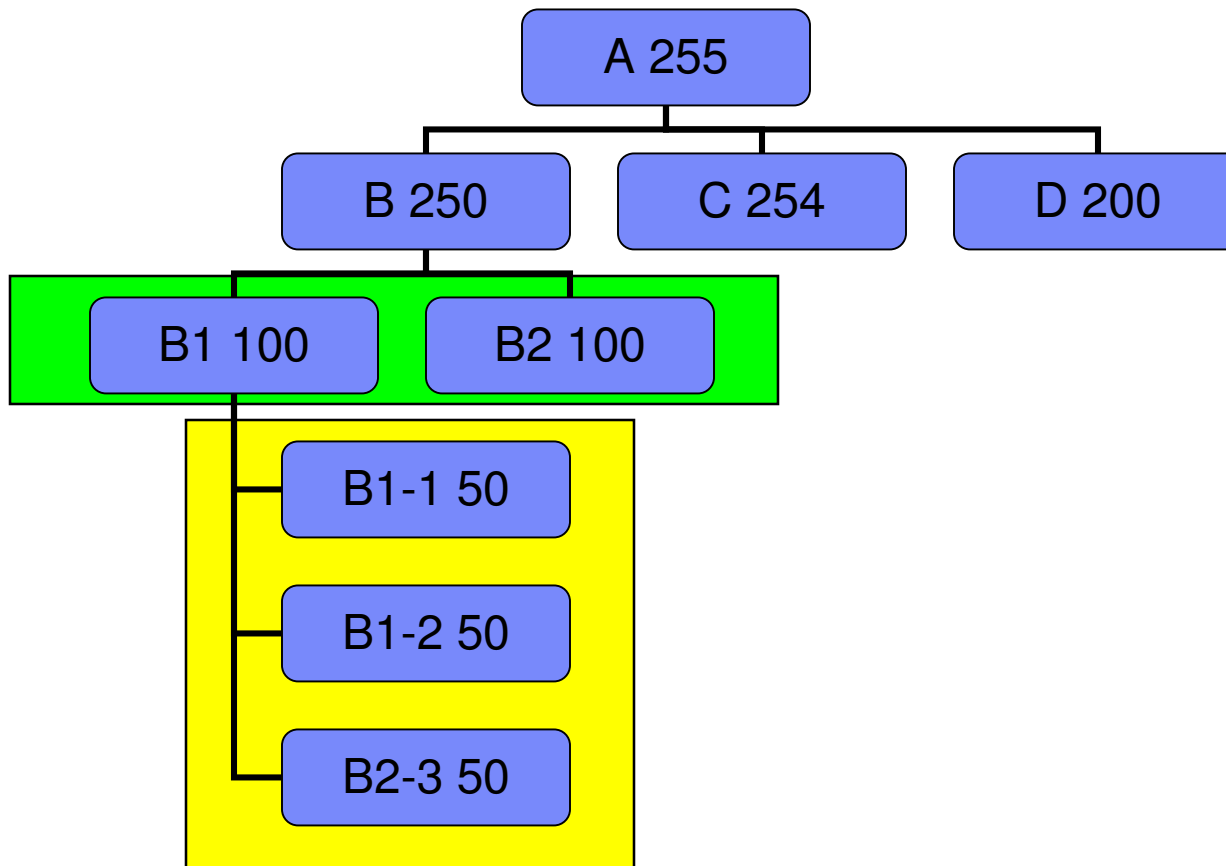
Label	Value
'Dept 1'	1
'Dept 2'	2
'Lob 1'	3
'Executive'	4
'Lob 2'	5

- ▶ User has label { 'Lob 1' }
- ▶ User has label { 'Dept 2' }

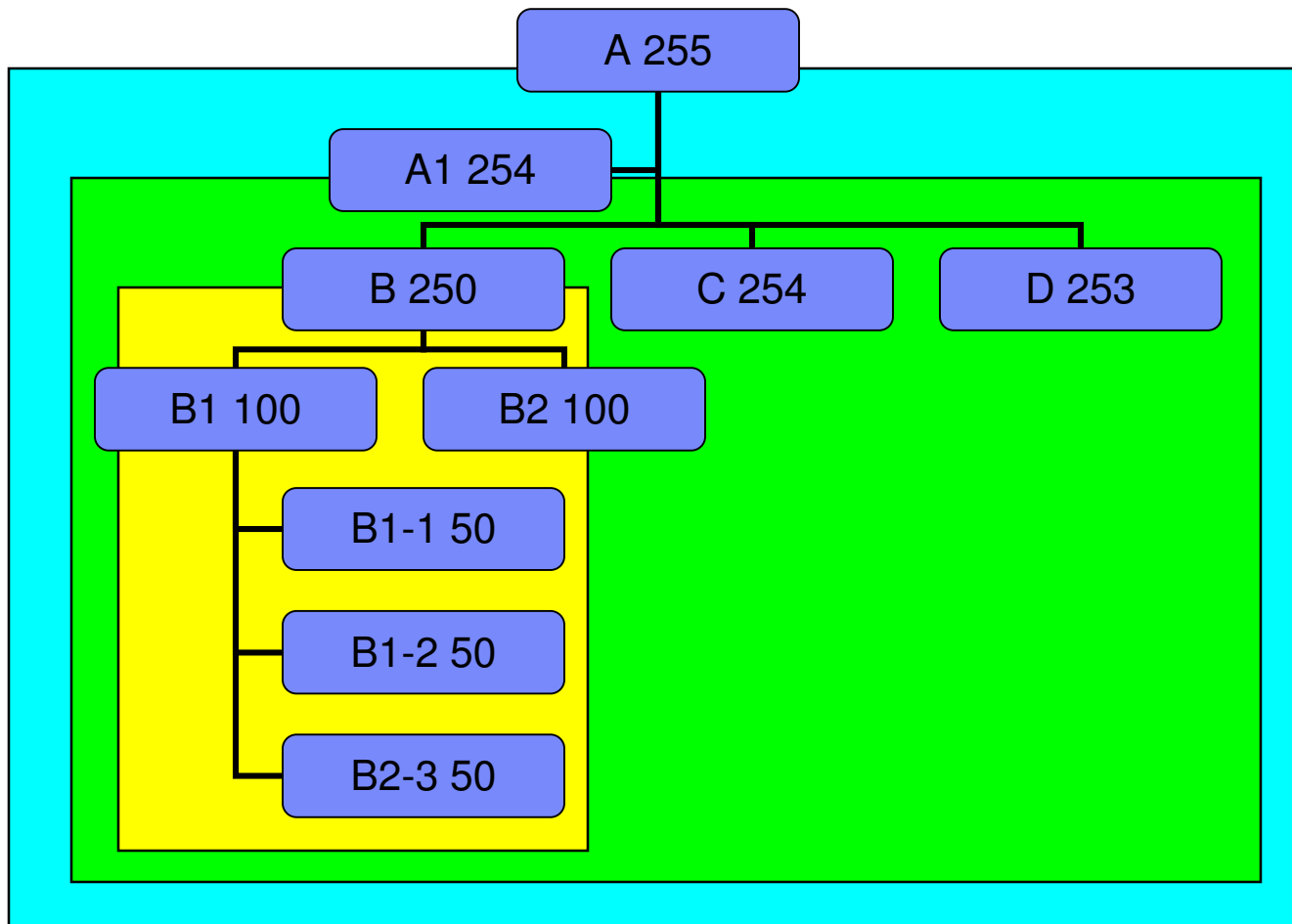
Label	Value
'Dept 1'	1
'Dept 2'	2
'Lob 1'	3
'Executive'	4
'Lob 2'	5

Label	Value
'Dept 1'	1
'Dept 2'	2
'Lob 1'	3
'Executive'	4
'Lob 2'	5

# LBAC Hierarchy – Update



# LBAC Hierarchy – Read



# LBAC Query

```
SELECT * FROM EMP  
WHERE SALARY >= 50000
```

User Level = 100

No LBAC	LBAC	ID	SALARY
Red		255	60000
Red	Green	100	50000
Red	Green	50	70000
		50	45000
		60	30000
Red		250	56000
Red		102	82000
Red	Green	100	54000
		75	33000
		253	46000
Red	Green	90	83000
Red		200	78000
		105	45000

## LDAP Authentication via Security Exit !

- DB2 does not have its own mechanism for maintaining userids and passwords or userid group memberships
  - ▶ All authentication in DB2 is managed either through the underlying operating system or an external security system such as Kerberos
- This feature will allow the customers to create their own authentication mechanisms to handle:
  - ▶ group membership
  - ▶ authentication on the client side
  - ▶ authentication on the server



IBM Software Group

# SQL features

**DB2.** Data Management Software



**ON** DEMAND BUSINESS™



## Less Limits

- Support for larger index key parts and number of columns
- Support for >18 Char Function Name
- Increase identifier limits to 128 bytes

Version	Length of index key parts	# of columns in index key
Pre-Viper	1024	16
Post-Viper	1024 – 4K page	64
	2048 – 8K page	64
	4096 – 16 K page	64
	8192 – 32 k page	64

# Materialized Query Table Improvements

- Explain Un-used MQTs
  - ▶ Lists MQTs which were not used and the reason for their exclusion
- Mismatched Elements and Expression Support
  - ▶ Allow for expressions with different order of operation to be considered for MQT selection
  - ▶  $C = A + B$  is equivalent to  $C = B + A$
- Efficient handling of  $A=B$  OR (A IS NULL AND B IS NULL) predicates



IBM Software Group

# Data Compression

**DB2.** Data Management Software



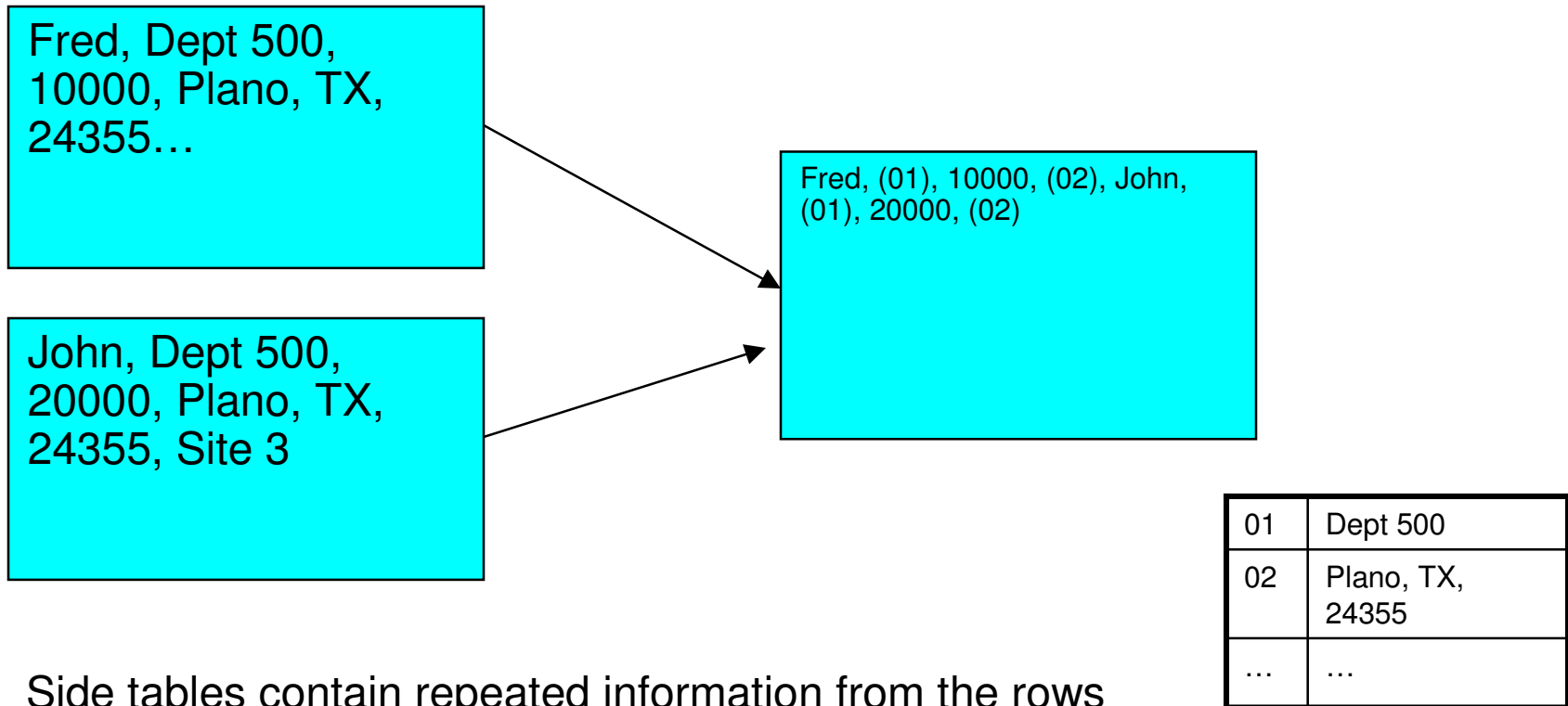
**ON** DEMAND BUSINESS™

© 2005 IBM Corporation

## Row Compression

- Dictionary based - symbol table for compressing/decompressing data records
- Lempel-Ziv (LZ) based algorithm (static dictionary)
- Dictionary per table stored within the permanent table object (~74KB in size)
- Data resides compressed on pages (both on-disk and in bufferpool)
  - ▶ Significant I/O bandwidth savings
  - ▶ Significant memory savings
  - ▶ CPU costs: Rows must be decompressed before being processed for evaluation
- Log data from compressed records in compressed format
- Does not compress rows where no storage saving is realized for that row
- Repeating patterns within the data (and just within each row) is the key to good compression. Text data tends to compress well because of reoccurring strings as well as data with lots of repeating characters, leading or trailing blanks

# Row Compression Using Side Tables



- Side tables contain repeated information from the rows
- Can be across column boundaries or within columns

# Questions?

