



(\*)



# IBM DB2 for z/OS

## DB2 Version 10(X)

### (Übersicht)

(\*) ist eingetragenes Warenzeichen der IBM International Business Machines Inc.

## DB2 Version 10 (Überblick)

**Der Slogan der IBM:** „DB2 10 for z/OS –  
Im Einsatz, wo andere längst aufgeben...“



A word cloud featuring various attributes of DB2 10 for z/OS. The words are arranged in a non-linear fashion, with some oriented vertically and others horizontally. The most prominent words are 'bewährt' (proven) in large green letters, 'effizient' (efficient) in teal, 'hochverfügbar' (highly available) in grey, and 'zuverlässig' (reliable) in green. Other visible words include 'skalierbar' (scalable), 'flexibel' (flexible), 'elegant', 'einfach' (simple), and 'belastbar' (loadable). The background of the word cloud is a faint, light blue graphic of a globe or a similar abstract shape.

## DB2 Version 10 (Überblick)

### Ziele der IT-Unternehmen heute:

- ✦ **Niedrige “operating costs”** durch Reduktion der CPU Zeiten
- ✦ Gleichzeitig: **Aufbau und Verwaltung einer starken Basis** für “service-oriented architecture” (SOA) und XML Einsatz
- ✦ DBAs müssen eine **verbesserte DB-Performance, Skalierbarkeit und Verfügbarkeit** anbieten können und trotzdem das **“memory management” reduzieren**, sodass Wachstum zu einem einfachen Prozess werden kann
- ✦ **DB2 10 for z/OS bietet Innovation** in folgenden Schlüsselbereichen:
  - **Verbesserte Effizienz im operationalen Umfeld**
  - **unübertroffene Flexibilität bei “business-critical” Informationen**
  - **Schnelles Anwendungs- und “warehouse-Deployment “ bei entsprechendem Wachstum**
  - **Verbesserte Query und Reporting Funktionen**

### Hier folgt ein **genereller Überblick über DB2 10 for z/OS** mit folgenden Schwerpunkten:

- **„Executive summary“**
- **Vorteile beim Einsatz von DB2 10**
- **Verbesserungen in den Subsystemen**
- **Neue Applikationsfunktionen**
- **Verbesserungen bei Operation und Performance**

## DB2 Version 10 (Überblick)

### Executive Summary

- Verbesserungen in **Performance**, **“scalability”**, **“availability”**, **“security”** und **“application integration”**
- **System z 64-bit Architektur** – dadurch CPU-Einsparungen (5-10%) und Performance-Verbesserungen und so Reduzierung der “total cost of ownership”.
- **Die DB2 10 “CPU reduction features”** bieten verbesserte Funktionen und so können Firmen jeden einzelnen Faktor analysieren, um so die Basis zu verbessern
  - Erste Tests von DB2 10 zeigen, dass viele Verbesserungen die “runtime” Umgebung optimieren und so den CPU Bedarf um ca. 5-10% senken –
  - mit weiteren möglichen 10% Reduzierungspotential, wenn neue Funktionen eingesetzt werden
  - und bis zu 20% Reduktion für neue “workloads”
- **DB2 10 kann 10 mal mehr “concurrent users” verwalten** – bis zu 20,000 Benutzer
- **Kapazität, Skalierbarkeit und Performance sind weiterhin die Stärke von DB2 for z/OS.** Durch extensive Nutzung der **System z “64-bit architecture”**, bietet DB2 Verbesserungen in allen Bereichen des Systems – effizienteres “optimizing”, “in-memory work files”, höhere Parallelverarbeitungsfähigkeiten, neue SQL/XML Zugriffspfade etc. - ohne Änderungen an den Applikationen...
- **DB2 10 “features” verbessern auch das “business processing”**, die Datenverfügbarkeit und die “ease of use” Eigenschaften durch Optionen wie “online database changes“, mehr Parallelität bei Utilities und einfachere Administrationsverfahren

## DB2 Version 10 (Überblick)

### Executive Summary

- **DB2 10 DB Systeme sind immer verfügbar.** Transaktionen werden selbst dann ausgeführt, wenn Änderungen auf dem Datenbanksystem erfolgen. Diese neuen Änderungsverhalten zusammen mit zusätzlichen Optionen für mehr parallele Utility-Verarbeitung führen zu einem kleineren Administrationsfenster und einem geringeren Zeitaufwand für alle Prozesse.
- **Verbesserungen in “Security”, regulärer Compliance und den Audit Fähigkeiten** sind ebenfalls Bestandteil von DB2 10. Die neue “security” erweitert das in DB2 9 eingeführte “role-based model”. DB2 10 bietet granulare Authorisierung, die den Datenzugriff von der Administration einer Applikation, einer Datenbank und dem System trennen kann. DB2 10 besitzt für bestimmte Rolleneinstellungen in der “security” eine hohe Flexibilität und verhindert gleichzeitig Datenzugriffe von nicht-authorisierten Programmen und/oder Administratoren. Das “role-based security model” kombiniert mit “label-based row” und der Zugriffskontrolle auf Spalten, sowie das “masking” bzw. die Verschlüsselung von sensiblen Informationen, verbessern letztlich die Sicherheit der DB-Umgebung entscheidend. Alle “features” offerieren genauere Kontrolle, mehr Flexibilität in der Sicherheitshierarchie mit entsprechenden Implementierungsvarianten von “compliance“-Vorgaben.
- **Applikationsintegration und Portabilität** profitieren von den DB2 10 SQL und XML Verbesserungen.
- **SQL Verbesserungen in DB2 10** ermöglichen der DB2 Familie eine einfachere Portierung von DBMS-Produkten anderer Hersteller in die DB2 10 for z/OS Umgebung. International tätige Unternehmen, werden die Verbesserungen bei “timestamps with time zones”, “Java timestamp compatibility” und “timestamp 12-digit pico-second“-Präzision wegen der weltweiten Verwendungsmöglichkeiten von “transaction timestamps” begrüßen.

## DB2 Version 10 (Überblick)

### Executive Summary

- **Versionsführung von Daten auf aktuellen und “history” Tabellen** wird mit DB2 10 ebenfalls verfügbar. Datenversionierung bietet die Flexibilität, aktuelle operationale Daten auf der Basis vorbestimmter Zeitperioden auf historische Tabellen zu migrieren. “Data versioning “ verbessert die Applikationsperformance und verwaltet historische Daten auch für Audit Vorhaben und bietet bessere Integration in die reguläre “compliance“ Architektur.
- Zusätzlich wurden die **“data warehouse” und “business intelligence” Funktionen** direkt in DB2 10 abgebildet. I. d. R. sind das die neuen “temporal table” Funktionen und die SQL Funktionen zur Kalkulation von gleitenden Summen und gleitenden Mittelwerten. Diese Funktionalität bedient die Grundanforderungen bei der Integration von BI in operationale Applikationen und senkt so den Programmieraufwand.
- **DB2 10 XML Verbesserungen** helfen bei der Applikationsflexibilität, Nutzbarkeit und Performance. DB2 10 bietet eine Reihe von wichtigen XML Performance- und Management-Verbesserungen, z.B:
  - Ersetzen, Löschen und Einfügen von XML Dokument-Knoten
  - Verwalten von mehrfachen Schema-Versionen bei XML Dokumenten
  - Nutzung des “binary pre-tokenized” XML Formats
- Weitere **Verbesserungen sind in den Utilities zu finden. “date time” Datentypen und XML Parameter** werden in SQL Funktionen und Prozeduren unterstützt. So wird XML in jedweder Applikationsarchitektur nutzbar.
- **DB2 10 for z/OS liefert Performance, Skalierbarkeit, Verfügbarkeit und Sicherheit.** Es verbessert ausserdem die Applikationsintegration und regulative “compliance” Anforderungen.

## DB2 Version 10 (Überblick)

### 1. DB2 10 – Vorteile/Verbesserungen

- ✚ **DB2 10 kann den gesamten DB2 CPU Bedarf, je nach Einsatz um 5-20% senken – vorausgesetzt, man nutzt so viele Verbesserungen, wie nur möglich.** Viele Einsparungen von CPU sind direkt in DB2 eingebaut und erfordern keinerlei Änderungen an Applikationen. Einige Verbesserungen sind über normale DB2 Aktionen implementiert: neue BIND, Restrukturierung der DB Definitionen, Verrarbeiten der Anwendungen und Utility Verarbeitung.

Die optionalen Reduktionsfunktionen für CPU Verbrauch haben das Potential, die “total cost of ownership “ signifikant zu beeinflussen.

- ✚ **Die Optimierungsverbesserungen reduzieren die Kosten über die Verarbeitung von SQL automatisch durch die Nutzung effizienterer Zugriffspfade:**
  - Einer “range-list index scan” Zugriffsmethode,
  - IN-list “list prefetch” und mehr Parallelität für SELECT und Index INSERTs
  - Verbesserte Nutzung der “work files”
  - Besseres Management bei “record identifier (RID) pool overflow”
  - Verbesserte “sequential detection”
  - Schnellerer Log I/O,
  - Sicherheit in der Evaluation der Zugriffspfade für “static” SQL und
  - Verbesserter Transaktionsablauf im “distributed data facility” (DDF)

## DB2 Version 10 (Überblick)

### 1. DB2 10 – Vorteile/Verbesserungen

- ✚ **Verbesserungen, wie “Hash Access space” und “hash access”, inklusive zusätzlicher Spalten in einem “unique index” und automatische Datenstatistikerstellung können die CPU-Kosten zusätzlich vermindern**
- ✚ **DB2 10 enthält zusätzlich dazu diverse Performance Verbesserungen für Large Objects (LOBs).** Das spart Plattenplatz für kleine LOBs und liefert bessere Performance beim Lesen, Einfügen, Laden und Import/Export von LOBs mit den DB2 Utilities.
- ✚ **Verbesserte Operationen und Verfügbarkeit von DB2 10 trägt ebenfalls zur Kostenreduktion bei.** DB2 10 kann die “system downtime” Kosten durch folgende Funktionalität vermindern:
  - Verbessertes „memory management“
  - Die Möglichkeit 5 bis 10 mal mehr User verarbeiten zu können
  - Die Fähigkeit, “locked rows” zu überlesen
  - Verbesserungen im DB2 Katalog
  - Mehr “online schema” Änderungsfunktionen
  - Mehr „online“ Utilities
- ✚ **DB2 10 hält die Applikation verfügbar, auch wenn mehr User, Anwendungen, Datenbankänderungen und Utilities im System laufen**
- ✚ **Mehr Automatismen im Installationsprozess**
- ✚ **Die Möglichkeit direkt von DB2 V8 auf DB2 10 zu migrieren**
- ✚ **Viele Verbesserungen sind bereits im DB2 10 CM verfügbar, egal von welchem Release man die Migration gestartet hat**



## DB2 Version 10 (Überblick)

### 2. Verbesserungen an den Subsystemen

Wie bei allen vorangegangenen Versionen, profitiert DB2 10 for z/OS von den letzten Verbesserungen in der entsprechenden Plattformumgebung. DB2 10 verbessert die Synergie mit der System z Hard- und Software für bessere Performance, mehr Flexibilität und bessere Funktionalität.

Es gibt neue Funktionen, die Skalierbarkeit und Verfügbarkeit verbessern. Viele der Verbesserungen hängen mit der Aufhebung des 2 GB Limits im “virtual storage” für den DBM1 Adressraum zusammen. Hierfür werden die meisten der DB2 “virtual storage”-Bereiche oberhalb der 2 GB Ebene zugewiesen.

Die Verifizierung des realen Speichers und die Anforderungen an ECSA/ESQA bleiben jedoch bestehen.

#### ✚ „Work file“ Verbesserungen

“Work files” dienen zum Sortieren und für Materialisierung. DB2 10 reduziert die Nutzung der “work files” für kleine SORTs und bietet Unterstützung für große SORTs und “sparse index” Aufbau. Dies kann die Performance verbessern und die Anzahl von “merges” beim SORT vermindern.

In der WORKFILE Datenbank, bietet DB2 10 die Möglichkeit, “partition-by-growth “ TS zu definieren. Dies, um so “in-memory tables” Tabellen für höchste Performance bereitstellen zu können. Die Funktion erlaubt einen dedizierten Definitionstyp für “in-memory” Daten und macht Extra-Administrationsarbeit zum Definieren eines einzelnen TS in dedizierten Bufferpools überflüssig.

DB2 10 erweitert zudem die Satzlänge für “work files” auf 65,529 Bytes, um so besser mit grossen Antwortdaten z.B. auf SORTs und/oder Joins umgehen zu können.

## DB2 Version 10 (Überblick)

### 2. Verbesserungen an den Subsystemen

#### ✚ Bufferpool Verbesserungen

System z hat die “memory”-Allokationen verbessert und DB2 10 nutzt dies in seinem “bufferpool management”. Früher wies DB2 beim “startup” den BP in seiner definierten Größe für alle IX- und TS-Objekte zu.

In DB2 10 weist das System die Menge “memory” für den BP zu, wie Daten in den Pool geladen werden. Diese Vorgehensweise hält den BP auf einer minimalen Größe : auf Applikationanforderung .

DB2 10 reduziert so zusätzlich seine “latch contention” und öffnet die Möglichkeit, größere BPs ( von MB auf GB) für kritische aktive Objekte zu allokiieren. Dies führt zu besseren I/O Raten und mindert die “contention” Situationen.

In “data sharing” Umgebungen, wenn ein Objekt seinen Status von oder nach “inter-system read / write” ändert, vermeidet DB2 10 ein “scanning” des BP, der das Objekt enthält und **spart so CPU**.

System z10® verwaltet mit der 1 MB Page große BPs besser. BP Zuweisungen können schnell sehr groß werden und DB2 10 kann die z10 1 MB Page nutzen, um das “system paging” auf einem Minimum zu halten.

#### ✚ Spezielle „engines“

**DB2 10 Verbesserungen der Parallelverarbeitung:** Mehr Querytypen geeignet für zIIP Przesse.

**DB2 10 RUNSTATS Utility:** Teile des RUNSTATS können auf zIIP laufen – abhängig von den Statistiken.

**DB2 10 BP “prefetch” Aktivitäten:** “Buffer pool prefetch”, inkl. “dynamic”, “list” und “sequential” Prefetch activities, ist zu 100% zIIP fähig in DB2 10.

**„Deferred write“ Aktivitäten:** DB2 fasst zusammen: “sorting by data set”, werden modifizierte Buffer im BP kumuliert - für Updates und asynchrones Schreiben. So werden I/O Aktionen ebenfalls zIIP-fähig.

## DB2 Version 10 (Überblick)

### 2. Verbesserungen an den Subsystemen

#### ✚ **Bessere Nutzung der „Extended 64-bit runtime“ Umgebung**

DB2 10 erreicht mit seiner verbesserten Nutzung des “64-bit environment” ein Verlagern von 80-90% des DB2 Speichers, der bisher “below the bar” war — “working storage”, EDMPOOL und Teile des ECSA— nach “above the 2 GB” und eliminiert Speicherbeschränkungen innerhalb des DB2 DBM1 “address space”.

#### ✚ **Anzahl User auf bis zu 20,000 angehoben**

Durch die Aufhebung der “memory constraints” im System, ist der “virtual memory” nicht mehr kritisch und erlaubt so 5 bis 10 mal mehr “concurrent threads” in einem einzigen DB2 10 Member. Dies erhöht die Anzahl möglicher “threads” und erlaubt zusätzliche DB2 “data sharing members” über die Konsolidierung von LPARs und Members, die früher nur aus Gründen der Benutzeranzahlen angelegt worden waren.

#### ✚ **Verbessertes “data set open” und “close” Management**

DB2 10 nutzt das neue z/OS 1.12 Interface –schnellere OPEN/CLOSEs und mehr “open data sets”.

#### ✚ **Verbesserte Änderungen im „online schema“**

Die besten Funktionen in DB2 10 sind die “online schema change features”. Änderungen an Attributen auf DB2 TS, Index bzw. Tabellenkomponenten sind nun über ALTER und einen möglicherweise notwendigen folgenden REORG möglich. Daraus resultieren Flexibilität in der Administration und höhere Verfügbarkeit der Applikationen . Bei UTS gibt es den ALTER in DB2 10 für :

- „Data set“ Größen
- Table bzw. Index Page Größe
- „Segment size“
- Page Größen (im Buffer pool)

## DB2 Version 10 (Überblick)

### 2. Verbesserungen an den Subsystemen

#### ✚ Verbesserte Änderungen im „online schema“(cont’d)

Man kann alte TS Definitionen auf “partition-by-growth” bzw. “partition-by-range“ UTS „omline“ migrieren. DB2 10 “online schema” Verbesserungen betreffen die Anpassung neuer Attribute über das ALTER Kommando als “pending change”, der später über einen “online REORG” manifestiert werden kann. ALTER und REORG belassen dabei die Tabellen für Anwendungsaktivitäten “verfügbar”, während die Attributänderungen ins System eingebracht werden. Dies vermeidet “downtimes” und entspannt das Problem der Verfügbarkeitsanforderungen bei “mission-critical” und “very large databases“(VLDBs).

Die neue ALTER / REORG Mechanik kennt einen neuen Status auf den DB2-Objekten: **“advisory state” / AREOR**. Er zeigt an, dass Änderungen anstehen.

Da die DB2 Datenbanken und das Transaktionsvolumen weiter anwachsen, wird Zeit für Administration noch mehr zur Herausforderung. Eine Reihe von Verbesserungen bei den DB2 Utilities helfen ebenfalls die “downtimes” zu minimieren, indem die FlashCopy® Funktion während des normalen Betriebs eingesetzt wird.

Bei DB2 10, können die DBAs IX auf Tabellen (auch mit LOBs) erzeugen bzw. mit REBUILD wieder aufbauen, ohne Auswirkung auf die Applikation bzw. “locking downtime”.

#### ✚ Universal „range-partitioned“ TS

DB2 10 unterstützt den “universal range-partitioned” TS auf neue Art und Weise: Als “updated version” des klassischen “range-partitioned” TS mit SEGSIZE Angabe, universellen Merkmalen und Funktionen. Er ist das Migrationsziel für die “alten” RPTS, der nur noch für wenige DB2 Versionen gültig sein wird.

DBA’s sollten also möglichs bald auf den “universal range-partitioned” TS umsteigen , nicht zuletzt wegen der erweiterten Utility Funktionen, der Verfügbarkeit und der Performance-Aussichten.

## DB2 Version 10 (Überblick)

### 2. Verbesserungen an den Subsystemen

#### ✚ Verbesserungen im DB2 Katalog

Der DB2 Katalog und das Directory sind neu strukturiert: Spezielle Strukturen und “links” wurden entfernt.

Mit der Restrukturierung benutzt der DB2 Katalog nun selbst **“universal partition-by-growth” TS, “reordered row format” und “one table per table space”**. Dies heisst: mehr DB2 Katalog TS. Sie sind mit DSSIZE 64 MAXPART 1, “row level locking” und einigen CLOB und BLOB Datentypen definiert, damit sich wiederholende lange “strings” besser zu verwalten sind. Die einheitlichen TS Definitionen erlauben es aber, Katalogtabellen mit Utilities in ähnlicher Weise zu behandeln, wie die TS in einer Applikationsdatenbank.

Zusätzlich dazu wurde der UTSERIAL Lock, der in älteren Versionen immer zu “lock contentions” bei der Utility Verarbeitung führte, eliminiert. Dies und eine Reduktion der “log latch contention”, eine neue Option für lesende Prozesse, die nicht mehr auf INSERTs warten muss und verbesserungen in der Serialisierung von “system thread latching”, reduzieren viele Typen der DB2 “thread processing contention”.

All dies trägt zur Verbesserung der Parallelverarbeitung bei DDL, BIND, Utility und der allgemeinen Verarbeitung innerhalb der Applikationsdatenbanken bei – besonders bei Zugriffen auf den DB2 Katalog.

Ein weitere Verbesserung liefert die Fähigkeit, ein neues Log in ein System einzuhängen, während das Subsystem weiter läuft. Dieses ist sofort verfügbar – ohne DB2 “recyclen” zu müssen.

#### ✚ MEMBER CLUSTER Option

DB2 versucht Daten in “clustering sequence” in eine Tabelle einzufügen. Bei “data-sharing” Systemen mit hoher INSERT Rate, kann dies “locking contention “ in einem TS über mehrere “data-sharing members” hinweg, verursachen. Dies bedeutet extra CPU Zyklen zum Vermeiden von “deadlocks” und “extended response time”.

DB2 10 unterstützt nun den Parameter MEMBER CLUSTER und erlaubt so “space-allocation” für INSERTs – ohne Rücksicht auf auf die “clustering sequence” nehmen zu müssen. (?).

## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

Die Standard- und anpassbaren Funktionen in DB2 10 offerieren verbesserte “business analytics” Funktionen und reiche Basis für “customization”. **“Built-in pureXML”, “LOB” und “open SQL” skalare und “table” Funktionsschnittstellen** bieten so flexibel, dass sie fast jede Anpassung an “business requirements” zulassen.

“Temporal tables” und die Fähigkeit, historische Daten automatisch zu archivieren, erlauben ein aktives Datenmanagement, verbessern die Performance, Auditing und “Compliance” Funktionalität eingeschlossen.

DB2 for z/OS unterstützt weitere Funktionen zum “deployment” von “data warehouses“ auf System z.

DB2 9 for z/OS unterstützt XML über pureXML und eine “hybrid database engine”. DB2 10 for z/OS erweitert diesen XML Support mit mehr Funktionalität und weiteren Performanceverbesserungen.

#### ✚ Höhere Präzision beim “timestamp” Datentyp für Java Applikationen

DB2 10 erhöht die Präzision des TIMESTAMP Datentyps. Die neue TIME ZONE sensitive Funktionalität erweitert die Kompatibilität und Funktionalität für alle möglichen Applikationstypen.

Der neue TIMESTAMP Typ unterstützt bis zu 12 Stellen der fraktionalen Sekundenangabe (Picosekunden) und ist als “default” gleichzeitig auch der Java “default” mit 6 Stellen. Diese darstellung hilft der Kompatibilität in JAVA und innerhalb der DB2 Familie, sowie dem SQL Server. Applikationen können über ein Spezialregister bei der CURRENT TIMESTAMP Funktion das Format selbst bestimmen.

#### ✚ Support for TIMESTAMP WITH TIMEZONE

Der Datentyp TIMESTAMP WITH TIMEZONE entspricht dem TIMESTAMP mit 12-Stellen und nutzt den neuen Industriestandard UTC (Coordinated Time), der die GMT ersetzt. Dies hilft Applikationen beim Vergleich von Geschäftsdaten bezogen auf weltweite Zeitangaben: Banken, Börsen, Versicherungen etc.



## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✚ „Extended indicator variable“

DB2 10 führt eine **“extended indicator variable”** ein. Sie öffnet einen Weg, bei einem INSERT, UPDATE und MERGE Statement, darzustellen, dass für eine Spalte **kein Wert** vorgesehen ist.

**Beispiel:** Der Wert -5 innerhalb einer “enabled extended indicator” Variablen zeigt den DEFAULT Wert. Sind die “extended indicator “ Variablen nicht im SQL Package zugelassen, dann bedeutet -5 den NULL Wert. Ansonsten bedeutet -5, dass die Variable UNASSIGNED ist, ignoriert und so behandelt wird, als ob sie im SQL Statement nicht existieren würde.

Diese “extended indicator variables” sind **typisch für Java Applikationen** und hilfreich bei dynamischem SQL und variablen SQL Statements, in denen die Anzahl möglicher “host variable” Parameters unbekannt bleibt, bis die Transaktionslogik komplett ist und das SQL ausgeführt wird.

Diese Funktion betrifft vor allem die Applikationen, die mehrere SQL Statements ausführen müssen, bevor Variable für ein weiteres SQL verfügbar gemacht werden können. Nun werden diese “Mehrfach-SQLs” konsolidiert. Ist der Spaltenwert unbekannt, so kann der Wert der Hostvariablen über das Schlüsselwort UNASSIGNED den entsprechenden Spalten zugewiesen werden.

Dieses “feature” ist besonders wichtig für Applikationen, die dynamisches SQL einsetzen und den Dynamic Statement Cache mit vielen Kopien von meist denselben SQL Statements verstopfen.

#### ✚ Erweiterter Support beim impliziten “casting”

“Implicit casting” bedeutet die automatische Konvertierung unterschiedlicher Datentypen, um sie kompatibel zu machen. DB2 verbessert diese Funktion, indem es numerische Datentypen nun auch implizit auf “character” bzw. grafische Formate umwandelt und umgekehrt. Früher musste dies manuell geschehen. Hiermit wird DB2 for z/OS auch kompatibler zu fremden DBMS.

## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✚ Verbesserte „scalar function“

DB2 10 verbessert seine Kompatibilität mit fremden Datenbanksystemen, indem sich SQL “scalar” und “table” Funktionen verändern. Diese “built-in functions” werden von Applikationen genutzt, um die OLAP Funktionalität, wie SUM, AVG, SIN, COS etc. zu verbessern. Sie alle liefern nur **ein Ergebnis** zurück.

“**Non-inline SQL scalar functions**”, die Logik und zusätzliche Applikationsfunktionalität anbieten, erhöhen auch die Flexibilität. Dies wiederum fördert die Kompatibilität der DB2 Familie – untereinander und nach “ausen”. DB2 unterstützt auch Mehrfach-Versionen und bietet ein “source code management” für diese Funktionen – entweder auf der Basis der Parameter Liste, der Routinoption oder ihrem “function body”. Die Funktionen beinhalten auch “fallback”-Möglichkeiten auf “alte” Versionen ohne “rebind” / “recompile”.

DB2 10 unterstützt SQL “user-defined table” Funktionen, die Migrationen von fremden DBMS erleichtern. DB2 “table functions” werden flexibler, da sie ein einzelnes Tabellenresultat liefern, das auf vielen unterschiedlichen Parametertypen, wie LOBs, “distinct types” und “transition tables” basieren kann.

#### ✚ SQL/PL Verbesserungen

DB2 9 bietet **native Unterstützung der SQL “procedural language”** ohne die Anforderung zuerst ein C Programm aus der SQL Prozedur erzeugen zu müssen, die als “external stored procedure” ausgeführt wird. DB2 9 SQL Prozeduren können direkt innerhalb der DB2 “engine” abgearbeitet werden und führen so zu einer besseren “runtime execution” und zur Speicherung direkt im DB2 Katalog.

Die Abarbeitung des “native code” innerhalb von DB2 erleichtert auch das “debugging”, “deploying” und die Verwaltung von “SQL procedural versions” über mehrere Server hinweg. Die Speicherung direkt im DB2 Katalog verbessert die Änderungskontrolle und die Verwaltbarkeit wie bei “normalen” Anwendungsmodulen.



## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✚ SQL/PL Verbesserungen(contn'd)

Die Verbesserungen an SQL/PL umfassen “SQL Table functions”, “nested compound SQL” Statements innerhalb einer Prozedur und das RETURN Statement, das nun ein Resultat als “**set of a SELECT SQL Statement**” zurückgeben kann. Die SQL/PL erlaubt nun alle bekannten Verarbeitungstypen.

Die DB2 10 SQL/PL Verbesserungen bieten neben der Kompatibilität mit anderen RDBMS auch Verbesserungen in der Nutzung der verschiedenen Datentypen wie XML als Parameter und – mit Einschränkungen - von “scrollable cursors”. Damit einher geht auch die Verbesserung bei Migrationen , Performance, Verfügbarkeit und Skalierbarkeit.

#### ✚ Unterstützung von OLAP: Sich verändernde Simmenwerte, Durchschnitte und Aggregate

**Diese OLAP Funktionen sind nun diekt im DB2 SQL verankert** und führen zu unmittelbaren Resultaten in “work files” und entsprechender Performance bei den “table functions” der OLAP Aktivitäten. “Moving sums”, “averages” und “aggregates” sind allgemein häufig genutzte Funktionen in DWH Applikationen. Sie realisieren verschiedene Gruppen von zeitabhängigen oder lokationsgebundenen Daten aus Produktverkäufen, Lagerorten bzw. anderen gemeinsamen Kriterien.

Diese OLAP Funktionalität ist weiter verbessert über skalare , anpassbare Tabellenfunktionen bzw. die neuen “temporal tables” Funktionen, die den Aufbau eines Datenfensters für die o.g. DWH Funktionen leicht machen.

Zudem können diese SQL-Erweiterungen - “moving sums”, “averages” und “aggregates” - auch in Ausdrücken, “select lists” bzw. ORDER BY Statements eingesetzt werden.

## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✚ “Bi-temporal” Queries und ihre Benefits

DB2 10 bietet eine **temporäre Datenfunktion über BUSINESS\_TIME und SYSTEM\_TIME** Tabellenperioden – Definitionen eingestellt, oft als *“time travel queries”* bezeichnet.

Die Periodendefinitionen werden für “temporal table” Definitionen verwendet, um so “system-maintained”, “period-maintained” bzw. “bi-temporal” (d.h. sowohl “system” als auch “period maintained”) Datenspeicher zu verwalten. Diese “temporal data tables” werden automatisch gepflegt und immer dann, wenn das entsprechende Zeitkriterium erreicht ist, auf eine Historientabelle archiviert.

Die PERIOD SYSTEM\_TIME bzw. PERIOD BUSINESS\_TIME Definition über zwei Spalten legt die temporäre Periode für die Daten in einer Tabelle fest:

- SYSTEM\_TIME beschäftigt sich mit der Zeit, zu der die Daten ins System gelangten
- BUSINESS\_TIME betrachtet die “business transaction” / die “business” relevante Zeitperiode der Daten

Diese Definitionen kontrollieren Kriterien für die die Daten in einer Tabelle verfügbar sein und wann sie auf die zugehörige “history table” geschrieben werden sollen. Beim **Einsatz beider Definitionen - PERIOD SYSTEM\_TIME und PERIOD BUSINESS\_TIME** – besitzt eine Tabelle **bi-temporale Kriterien**.

Beim Parameter **BUSINESS\_TIME WITHOUT OVERLAPS** können die “temporal tables” alle Transaktions-Timestamps eindeutig machen, indem sie die TIMESTAMP -Präzision auf Picosekundenbasis setzen. Diese Funktionalität unterstützt global eingesetzte Systeme mit der **sicheren Eindeutigkeit von “business timestamp transactions”**.

## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✚ Zugriff auf aktuell “committed data”

Mit 5 bis 10 mal mehr parallelen “threads” innerhalb eines DB2 “members”, zielt DB2 10 auf eine signifikante Verbesserung der Parallelverarbeitungsfähigkeit von Applikationen. DB2 10 besitzt hierfür eine eigene Package Option als BIND Parameter:

DB2 10 bietet den Parameter **CURRENTACCESSRESOLUTION** mit **USECURRENTLYCOMMITTED** und **WAITFOROUTCOME** Einstellungen an. Diese Parameter überschreiben die DB2 Subsystem Parameter **EVALUNC** und **SKIPUNCI** und helfen dem Package möglichst schnell parallele Aktionen zu erledigen.

- **USECURRENTLYCOMMITTED** sagt dem System, dass “rows”, die gerade eingefügt werden, ignoriert werden sollen und nur “rows”, die bereits “committed” wurden verwendbar sind. Diese Klausel ist mit den Package BIND “isolation level” Einstellungen verknüpft: “Cursor Stability” / “Read Stability”.
- **WAITFOROUTCOME** gibt an, dass anstehende “scans” auf “commit” bzw. “rollback” warten müssen, wenn Daten sich in einem UPDATE/DELETE-Prozess befinden. “Rows”, die eingefügt werden, können nicht übergangen werden.

Diese unterschiedlichen “settings” machen eine Applikation flexibel in der Entscheidung - beim Handling von hochparallelen “web transactions” – zu warten bzw. “uncommitted data” zu verarbeiten.

#### ✚ „Plan stability“, Package-Schutz

Die “plan stability” in DB2 9 bietet die Möglichkeit, neue Versionen von DB2 “application packages” zu testen: Sie offeriert die Möglichkeit mehrerer “package copies” und des Umschaltens auf vorangegangene Kopien. Ist der “access path” des aktuellen Package ineffizient, so kann der DBA mit einem **einfachen REBIND SWITCH Kommando** auf die “alte” Package-Version zurückgreifen.

## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✦ „Plan stability“, Package-Schutz(contn'd)

DB2 10 for z/OS liefert dazu eine neues “framework”, das zu existenten und zukünftigen “features” im Rahmen der Sicherstellung des “query access path”, passt. Ein “access path repository” enthält den Test der Query, den “access path” und die “optimization options”. Das Repository unterstützt die Versionierung über Kopien von “access paths” und den zugehörigen Optionen pro Query.

#### ✦ „pureXML“ Verbesserungen

In DB2 10 kann **XML fast überall in SQL Variable, “scalar functions”, SQL “table functions” und SQL Prozeduren** verwendet werden. DB2 10 pureXML beinhaltet viele Verbesserungen, die die Gesamtperformance von XML erhöhen und ein einfacheres “XML schema management,” zulassen.

Anfangen von der XML “schema validation” in DB2 10, bis hin zur Manipulation jedes Teils eines XML Dokuments bietet DB2 10 umfangreiche Änderungen.

- **Das XML Schema** muss nicht mehr angegeben sein. Db2 behandelt die “XML schema validation” über eine “built-in function”, die die XML Schemas aus den “instance documents” ableitet. DB2 nutzt dazu “schema registration timestamp” und “schema location hints”.
- **Bei SQL Statements mit XQuery “updating expressions”**, können einzelne oder mehrere XML “document nodes” eingefügt, modifiziert bzw. gelöscht werden. Es können auch Datenwerte aktualisiert werden.
- Ein eigener **XML “type modifier”** legt “constraints” auf XML Spaltendaten und testet und validiert die XML “column data” gegen die “schema definition” Information. Der “type modifier” kann auf ältere XML Schemas ge-ALTERed werden, sodass die XML “column types” dort validiert werden können.

## DB2 Version 10 (Überblick)

### 3. Applikationsfunktionen

#### ✚ „pureXML“ Verbesserungen(cont'd)

**DB2 10 unterstützt nun XML “date”- und “time” Typen.** Sie werden in XML Indexes zulässig und der “timestamp” wurde ebenfalls auf höhere Präzision ausgedehnt. XML “time”- und “date”-Arithmetik wird ebenfalls möglich.

DB2 10 verbessert die XML Performance mit der Unterstützung von **“binary XML” Objekten**. Dies hilft vor allem der Server- und Applikationsinteraktion. DB2 nutzt dazu **“pre-tokenized” Format- und Längen- definitionen**, die Einfluss auf Performance und einfachere Verwendung in Java Applikationen haben.

“Binary XML” besitzt zusätzliche “flexibility features”, wie “String IDs”, die Texte gleichen Inhalts mit einem “integer identifier” kennzeichnen. So wird die Größe von XML reduziert – zugunsten der Performance.

DB2 10 bietet einen neuen **LOB\_INLINE\_LENGTH** Installationsparameter . Er setzt den “default” für die Anzahl Bytes, die für “inline LOBs” gespeichert werden können. Eine minimierte LOB Länge bzw. eine vordefinierte Standardlänge führt zu besserem I/O Verhalten, besserem “Buffering” und **optimiert die Nutzung des “inline LOB space“**.

Administratoren haben nun die Option die Definition der LOB- bzw. der XML “data sets” und der IX zu verzögern, was zu Platzeinsparungen und zur Möglichkeit des SELECT und FETCH durch die Applikation führt. Die LOB / XML “data sets” und ihre IX werden erst allokiert, wenn der erste INSERT erfolgt.

Der DBA kann ausserdem das Utility CHECK DATA einsetzen, um die **Konsistenz zwischen dem XML Schema, den Dokumentdaten und den XML Index-Daten** zu prüfen.

## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Unterstützung der “row” und “column access control”

**DB2 10 kann seine “security” auf “rows und “columns” abbilden.** Dies ergänzt die bereits implementierte “security” Methoden im Privilegienmodell. Nach Aktivieren des “row” / “column level access control” wird jedes SQL Statement gegen diese Sicherheitsregeln gegengeprüft – egal welcher User das SQL-Statement absetzt.

Dies trägt dazu bei, genauere Sicherheitsbedingungen für alle Daten zu definieren. Das “role-based security” Modell, kombiniert mit diesem “feature” und der Maskierung bzw. der Verschlüsselung sensibler Informationen, verbessert letztendlich die Sicherheit für das gesamte “database environment”.

#### ✚ Real time Statistics Stored Procedures

Da der Optimizer über die “access paths” die Performance nachhaltig verbessert, sind die momentanen Statistiken enorm wichtig. **DB2 10 beinhaltet eine Reihe von “stored procedures”, die Tabellen und IX-Statistiken überwachen und automatisch erneut erstellen können.**

Die Prozeduren helfen in sich ständig ändernden Umgebungen “access paths” automatisch zu erstellen, indem sogen. “index filtering statistics” für SQL WHERE Klausel Prädikate verwendet werden, um die besten “access path” Entscheidungen treffen zu können. Diese DB2 10 “stored procedures” nehmen die Verantwortung in großen und dynamischen Objekten von den DBA’s und helfen die AP-Performance zu sichern.

#### ✚ Skip level migration

DB2 10 for z/OS ermöglicht die Migration nicht nur aus DB2 9 NFM sondern auch direkt von DB2 V8 NFM. Dieser Vorgang läuft unter dem Begriff: **“skip level migration”**. **Die Planung und die Lernanforderungen** betreffen trotzdem die neuen Funktionen und die Restriktionen beider Releases.



## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

Technische Innovation hilft bei “security” und der Verbesserung der “auditing” Funktionen.

Die DB2 Utilities Suite for z/OS liefert den vollen Support für die Verbesserungen in DB2 10 for z/OS und die Integration mit den “storage systems” Funktionen, wie “FlashCopy”, Neue “sort options” (DB2 Sort), Backup und Restore.

Die Performance in DB2 10 for z/OS wird gesteigert über verminderte Regression und Möglichkeiten CPU Zeiten zu reduzieren. Für die Nutzer relationaler DBMS ist die Performance von höchster Bedeutung. DB2 10 for z/OS kann den CPU Bedarf sofort um 5-10% senken – ohne Änderung an den Programmen. Weitere CPU Bedarfe können um bis zu 20% gesenkt werden, wenn man alle DB2 10 Verbesserungen im “new-function mode” (NFM) nutzt.

#### ✚ Security und reguläre „compliance“

**DB2 10 verbessert, die aus DB2 9 bekannte “role-based security” mit zusätzlichen administrativen und weiteren granulären Autorisierungen und Privilegien.** Dies wiederum ist hilfreich bei separater Administration und datenzugriffen, die nur ein Minimum an angemessener Autorisierung erfordern.

Das “authorization level” SECADM sieht vor, dass die AUTH “access to the tables” verwaltet wird, während die Erstellung, Änderung und das Löschen von Tabellen verboten ist. Eine geänderte DBADM AUTH ermöglicht es, Administrationsfunktionen ausüben zu können, ohne Datenzugriff zu haben.

DB2 10 implementiert “audit” und “regulatory compliance” über eine neue “audit” Policy, die eine Reihe von Kriterien anbietet, um möglichen Mißbrauch und sich überschneidende Autorisierungen im System zu verhindern. Dies hilft dem Management, der Administration und dem geschäftlichen Umfeld.

## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Performance ist durchgängig im Produkt verankert

Laut IBM **liefert DB2 10 for z/OS mit seiner durchgehenden Reduktion des CPU-Verbrauchs die besten Performanceverbesserungen seit DB2 Version 2.1.** “Availability”, “scalability”, “security”, “compliance”, “application integration”, XML und SQL alles enthält auch Performanceverbesserungen. Dies verbessert das operationale Umfeld, erleichtert die Administration und senkt die “total cost of ownership”(TCO).

Performance ist also das Hauptziel von DB2 10. Nach der Migration auf die neue Version und dem REBIND der Applikationen, können diese bereits von den Vorteilen der DB2 10 profitieren. DB2 10 nutzt die 64-bit Architektur in vollem Umfang, die Optimierung bietet neue “access paths” und Performanceverbesserungen, die unabhängig von diesen sind.

**“Rebinding” ist auch einfacher geworden.** Das Problem der Regression kann einfach über “plan stability” gelöst werden. Eingeführt in DB2 9 und weiterentwickelt in DB2 10 bietet diese “DB2 10 plan stability” “fallback”, und “package level” DSNZPARM Einstellungen und damit den DBA’s die REBIND-Möglichkeiten ohne Performance Risiken.

#### ✚ Der Optimizer Zugriffspfad “range-list index scan”

DB2 10 offeriert einen neuen **“application access path” mit Namen “range-list index scan”**. Dieser verbessert das SQL “processing” gegen einen IX, wenn mehrere WHERE Klauseln denselben IX referenzieren sollten. **Beispiel:** Besitzt ein SQL mehrere OR bedingungen, die denselben IX betreffen, so erkennt der Optimizer dies und durchsucht den IX nur einmal, unter Beachtung der IX Reihenfolge – anstatt einen “multiple index access” durchzuführen, der am Schluss einen SORT benötigt. Dieses Vorgehen verringert die Anzahl der Einträge in der RID List entries, was wiederum I/O’s und CPU Einsparungen mit sich bringt.



## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Der Optimizer Zugriffspfad “range-list index scan”(contn’d)

Man kann diesen “access path” nutzen, wenn folgende Bedingungen zutreffen:

- Alle WHERE Klauseln mit mehrfachen OR Statements betreffen dieselbe Tabelle
- Jedes OR Prädikat besitzt letztendlich ein “matching predicate”
- Alle OR Prädikate werden auf denselben IX “gemapped”.

Diese Art WHERE Klausel ist typisch für bestimmte Applikationstypen: “searching”, “scrolling” etc.

#### ✚ Optimizer setzt mehr auf Parallelität

**DB2 10 verbessert bestehende “access paths” über Parallelverarbeitung.** Dieses extra entworfene “feature” eliminiert bisher existierende Restriktionen, erhöht die Arbeitslast, die an die zIIP Prozessoren gegeben werden kann und verteilt die Last gleichmäßiger auf die “parallel tasks”. Diese Verbesserungen eröffnen zusätzliche Gründe, warum man Parallelverarbeitung für die eigene Umgebung freischalten sollte.

Parallelverarbeitung kann in folgenden SQL Query-Typen stattfinden:

- „Multi-row fetch“
- „Full outer joins“
- „Common table expressions“ (CTE) Referenzen
- „Table expression“ Materialisierung
- „table functions“
- CREATE GLOBAL TEMPORARY TABLE (CGTT)
- “work file” aus einer “view” Materialisierung

## DB2 version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Optimizer uses more parallelism (contn'd)

Diese neuen DB2 10 CP Parallelverarbeitungsmechanismen sind aktiv, sobald die SQL Explain PLAN\_TABLE PARALLELISM\_MODE Spalte ein "C" enthält-

Diese Verbesserungen sind auch aktiv in folgenden SQL Situationen, wenn:

- der Optimizer "index reverse scan" für eine Tabelle auswählt
- eine SQL Subquery in einen Join transformiert wird
- DB2 sich entschliesst einen "multiple column hybrid join" mit SORT auf "composite" durchzuführen
- die führende Tabelle "sort output" ist und der Join zwischen der "leading table" und einer zweiten Tabelle ein "multiple column hybrid join" ist

Weitere DB2 10 Optimierungs- und Zugriffsverbesserungen sind ebenfalls für die Applikationsperformance zuträglich. In DB2 10 wurden die Techniken "index lookaside" und "sequential detection" weiterentwickelt.

#### ✚ List prefetch improves index access

**"List prefetch" wird in DB2 10 oft für den Zugriff auf "index leaf"- und "non-leaf"-Pages eingesetzt.** In früheren DB2 Versionen - bei disorganisiertem IX, IX mit grossen Lücken zwischen den "non-leaf"-Pages etc. - wurde das sequentielle Lesen von "non-leaf"-Pages über hohe synchrone I/O-Raten ziemlich langsam. DB2 10 nutzt nun die "non-leaf"-Page Information für einen "list prefetch" auf den "leaf pages". Dies wiederum macht die meisten der "synchronous I/Os" überflüssig und die "I/O waits", die durch die "gaps" in den "non-leaf pages" während des sequentiellen Lesens entstehen, entfallen nun. Diese "list prefetch" Verarbeitung hilft bei langlaufenden Queries, die vom Zugriff auf die "non-leaf page" abhängig sind und auch bei IX-orientierten Utilities, wie "partition-level" REORG, REORG INDEX, CHECK INDEX und RUNSTATS.

## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Verbesserte „sequential detection“

Immer wenn DB2 einen IX nutzt, der eine hohe „high cluster ratio“ besitzt und dann die „rows“, die über den „index scan“ qualifiziert wurden, liest, verwendet es typischerweise den „dynamic prefetch“ auf die Daten.

**Der „Dynamic prefetch“ besitzt einen „sequential detection“ Algorithmus**, dessen Ziel es ist, 'synchrone I/Os für „clustered pages“ zu vermeiden. In DB2 10 wurde dieser Algorithmus robuster, was zu weniger „synchronous“ I/Os führt, besonders, wenn die „cluster ratio“ unter 100% liegt.

#### ✚ Verbessertes Log I/O

**DB2 10 vermindert die Anzahl Log I/Os** und arbeitet mit verbessertem „I/O overlap“ beim „dual logging“.

#### ✚ Optimizer ändert den „access path“ wegen RID Pool „overflows“ nicht mehr

**DB2 10 verbessert ebenfalls das Handling von SQL Statements, die eine hohe Datenrate wegen List-Verarbeitung aufweisen.** Diese Art der Verarbeitung betrifft grosse Anzahlen von „record IDs“ (RIDs), was manchmal zum Überlaufen des RID Pool führt. Früher führte dieser „RID pool overflow“ zu einer Veränderung der SQL-Zugriffsmethode – meist TS Scan. Heute wird die RID Liste beim Überlauf auf „work file resources“ geschrieben und die Verarbeitung wird fortgeführt.

Dies hilft „table space scans“ zu vermeiden – mit hohen „elapsed times“ und „performance overhead“.

#### ✚ Optimizer erledigt mehr SQL im „stage 1“ Umfeld

**DB2 10 kann nun „scalar functions“ und andere „stage 2“ Prädikate in „stage 1“ verlagern.** Für „indexed columns“ bedeutet dies, dass der Optimizer diese „stage 2“ Prädikate als „non-matching screening“ Prädikate verarbeiten kann und so die Anzahl von „data rows“, die gelesen werden müssen reduziert. Dies verbessert die „query elapsed time“ und damit die gesamte Performance.

## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Optimizer entscheidet, welcher Zugriff der sicherste ist

In früheren Versionen von DB2 evaluiert der Optimizer das SQL WHERE Prädikat, die verfügbaren Tabellenindexe und verschiedene Statistiken, um den effizientesten Pfad zum Zugriff zu finden. Mit den Verbesserungen in DB2 10 **analysiert der Optimizer zusätzliche Filterfaktorvariable** - betreffend SQL "range predicates", die Ungleichverteilung der Daten, die Verwendung von "parameter markers", "host variables" bzw. Literalen, deren Werte nicht bekannt sind.

Bei der Wahl zwischen unterschiedlichen IX entstehen Situationen, in denen diese unbekannt Filterfaktorvariablen die Kostenschätzungen zwischen unterschiedlichen Zugriffspfaden entscheidend beeinflussen. Die Analyse, welche der Filterfaktorvariablen bekannt sind, führt DB2 zur Erkenntnis, welcher der IX Pfade der sicherere ist. **DB2 kann so einen IX Zugriff wählen, der zwar höhere Kosten verursacht, aber eine sicher bestimmbare "runtime performance" voraussagt.** Diese Analyse ist besonders dann wichtig, wenn die beste verlässliche Performance in unterschiedlichen Programmiersprachen und entsprechenden Vielfalt der "parameter markers", Literale und "host variable" gefordert wird.

#### ✚ Verbesserung der „Dynamic statement cache“ ATTRIBUTES

**Eine der wichtigsten DB2 10 Systemverbesserungen ist die Fähigkeit, Varianten von SQL innerhalb des "dynamic statement cache" miteinander zu kombinieren.** Mit der neuen ATTRIBUTES Klausel im PREPARE SQL Statement, kann DB2 10 erkennen, dass ein SQL gleich ist – ausgenommen die WHERE Klausel Literal - Werte. Dies hilft DB2 zu erkennen, dass diese Statements im "cache" bereits existieren und so die "cache resources", die vorher generiert wurden, wiederverwendet werden können. Dies führt dazu, dass DB2 unnötige Katalogaktivitäten vermeidet, wie z.B. Objektverifikation, "access path" – Erstellung für weitere SQL Statements. So kann auch "cache space" für "reuse" anderer SQL's freigemacht werden, was ebenfalls zur Performanceverbesserung und schnellerer "transaction response time" beiträgt.

## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Verbesserter DDF „transaction flow“

**Die Applikationsperformance und der “network transaction” Verkehr wird optimiert, wenn SELECT Statements mit FETCH 1 ROW ONLY kodiert sind.** DB2 erkennt dieses SELECT mit FETCH 1 ROW ONLY Klausel und fasst die API “calls” und “messages” zusammen, um deren Anzahl im Netz zu reduzieren.

Die Änderung der FETCH 1 ROW ONLY Klausel verbessert auch die Performance von JDBC und CLI APIs. Nach dem Lesen der Querydaten veranlasst die FETCH 1 ROW ONLY Klausel eine API-”default“-Aktion, die die Ressourcen schliesst. DB2 macht dies, ohne Rücksicht darauf, ob ein CURSOR WITH HOLD definiert ist und informiert den API “driver” alle weiteren FETCH / CLOSE Anforderungen abubrechen. Damit wird die Anzahl und der Aufwand der “transaction network messages” minimiert, u.a. auch die “locking contentions“.

#### ✚ “Hash space” und Zugriff

**DB2 10 bietet nun einen neuen Zugriffstyp – genannt “hash access”, unterstützt über einen speziellen “hash space”.** DB2 nutzt dazu einen internen “hash” Algorithmus mit der Referenz auf die Datenlokation. Manch-mal reduziert eine “hash access” den Zugriff auf einen einzigen I/O, mit Einsparungen in der CPU Last und der Reduktion der Applikations-”Response time”. Queries die volle “key equal” Prädikate zur Existenzprüfung einsetzen, wie z.B. “KUNDENNUMMER” oder “PRODUKTNUMMER” sind Kandidaten für “hash”-Zugriffe. Man kann daneben natürlich weitere IX haben, um andere Zugriffsaspekte zu unterstützen.

Die Definition des “hash space” erfordert ein oder mehrere Spalte(n) für “direct hash access keys”. Jede Tabelle mit “hash access” braucht dazu einen “hash space” bzw. “hash space” Partitions.

## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ “Hash space” und Zugriff(contn’d)

Es gibt aber auch Nachteile bei der Nutzung des “hash access”. Parallelverarbeitung ist nicht möglich, traditionelles “clustering” ist für “hash data” nicht erlaubt. Ein “hash access” kann aber auch sehr nützlich und vorteilhaft sein: bei eindeutigen object IDs, XML document IDs und anderen direkten Schlüsseln.

#### ✚ Eleganterer Selektionsalgorithmus für „insert page candidates“

**DB2 10 ändert den Algorithmus zur Auswahl einer Page, in die eine neue “row” eingefügt werden soll, wenn dieser “key” nicht dem “cluster index” folgt.** Anstatt die Page des nächsthöheren “keys” zu wählen, wählt DB2 10 die Page des nächstniedrigeren Schlüssels. Wird nun eine Serie von “rows” mit einem sequentiell aufsteigenden “key” am gleichen Einfügepunkt des “cluster index” eingestellt, so führt der neue Algorithmus dazu, dass DB2 dieselbe Page auswählt, in der vorher der “free space” gefunden wurde. Dieser Algorithmus kann die Zahl der notwendigen “getpages” zur Lokalisierung des “free space” reduzieren und gleichzeitig “gutes clustering“ erreichen.

#### ✚ INCLUDE von “non-unique” Spalten in einem “unique” Index

**Eine der “schema” Änderungen , die jede Applikation brauchen kann, ist die Möglichkeit mehr Spalten als nötig in einen “unique Index” zu stellen.** Dieses DB2 10 “feature” erlaubt die Inklusion von “non-unique” Spalten in einer Definition von “unique” Indexes. Vor dieser Weiterentwicklung benötigte man einen IX für die “unique constraint” und einen weiteren IX für “non-unique columns”. Mit der neuen CREATE / ALTER INCLUDE Klausel kann eine “unique IX” Definition zusätzliche Spalten, ausserhalb der eindeutigen, enthalten.

Diese Funktionalität eliminiert extra I/O für andere IX mit ähnlichen Spalten(-folgen) und verbessert den Tabellenzugriff.



## DB2 Version 10 (Überblick)

### 4. Verarbeitungs- und Performance-Verbesserungen

#### ✚ Parallele INSERTs in mehrere Indexe

**DB2 10 verbessert die Insert-Performance mittels höherer Parallelverarbeitungsrate.** Modifiziert ein INSERT SQL eine Tabelle mit mehreren Indexen, so erledigt DB2 10 den “prefetch” mehrerer IX parallel. Bei der Initialisierung der parallelen I/Os für die “multiple indexes”, wartet der Prozess nicht mehr auf synchronen I/Os und reduziert so bereits die Gesamtverarbeitungszeit für den INSERT. Diese Methode senkt auch die Zeit in der “contentions” im System auftreten können und erhöht so den Durchsatz für alle Applikationen.

#### ✚ Utility Verbesserungen

**Utilities Support für RECFM=VBS** und weitreichendere und integrierte Verwendung der FlashCopy Technologie verbessern die Performance und die Verfügbarkeit bei Utilitynutzung.

#### ✚ Datenkompression

**Ein neuer DSNZPARM ermöglicht die Komprimierung von DB2 SMF Trace Daten.** Dies minimiert die Kosten des Plattenplatzes für wertvolle “accounting” Information.

Es ist nicht notwendig dazu mit REORG ein “dictionary” zu erzeugen. Der Insert tut dies dynamisch.

## DB2 Version 10 (Überblick)

### Executive Summary

1. **DB2 10 – Vorteile/Verbesserungen**
2. **Verbesserungen an den Subsystemen**
3. **Applikationsfunktionen**
4. **Verarbeitungs- und Performance-Verbesserungen**

